

# Designing a Multi-Slate Reading Environment to Support Active Reading Activities

NICHOLAS CHEN  
University of Maryland

FRANCOIS GUIMBRETIERE  
Cornell University

and

ABIGAIL SELLEN  
Microsoft Research Cambridge

---

Paper has long been the prevalent medium on which the reading activities of knowledge workers and students occur. Paper's dominance can be attributed to the fact that it supports a broad range of these users' diverse reading requirements. Our extensive analysis of the literature indicated that a new class of reading device consisting of an interconnected environment of thin and lightweight electronic slates could potentially unify the distinct advantages of e-books, PCs and Tabletop computers to offer an electronic reading solution providing functionality comparable to paper. This paper presents the design and construction of such a system. In it, we explain how data should be mapped to slates, detail interactions for electronically linking the slates, and describe reading tools that leverage the connectivity between slates. A preliminary study of the system was well received and indicated that such a system has potential as an electronic alternative to paper.

Categories and Subject Descriptors: H.5.2 [**Information interfaces and presentation**]: User Interfaces – Interaction Styles. H5.m [**Information Interfaces and Presentation**]: Miscellaneous.

General Terms: Design, Human Factors

Additional Key Words and Phrases: active reading, electronic books, horizontal display, tablet computer, distributed user interface

**ACM Reference Format:**

---

This work supported by NSF Grants IIS-0812196, IIS-0936105, and a gift from Microsoft Research. Nicholas Chen is funded by a Google Ph.D. Fellowship.

Authors' address: N. Chen and F. Guimbretiere, 301 College Avenue, Information Science, Ithaca, NY 14850, USA; email: nchen@cs.umd.edu, francois@cs.cornell.edu; A. Sellen, 7 J.J. Thomson Avenue, Cambridge CB3 0FB, Great Britain email: asellen@microsoft.com

Additional copyright information goes here.

## 1. INTRODUCTION

The reading activities that knowledge workers and students engage in are considerably more complex than the linear progression through textual material that characterizes reading for leisure. These reading activities involve a diverse mix of linear reading, skimming, annotating, interleaving reading and writing, and switching between documents that are used simultaneously. While there is no single agreed upon term to describe these reading activities where the user is more actively interacting with the material, they have often been mentioned as being components of “work-related reading” [Adler et al. 1998], “active reading” [Adler and van Doren 1972] or “responsive reading” [Pugh 1978]. For brevity’s sake, however, we will refer to these activities simply as *active reading* in this paper. Irrespective of the specific terminology used, most would agree that these activities are well supported by paper and that current electronic reading solutions do not appear to provide the wide range of functionality this style of reading demands. For this reason, it is perhaps not surprising that there has been low uptake of electronic reading devices such as the Amazon Kindle in the professional sector, and that deployments in academic environments have been met with mixed reactions at best [Behler 2009; Young 2009; Thayer et al. 2011].

Researchers who have studied paper use explain that one critical way paper supports active reading is by allowing the reader to work with a large amount of information simultaneously. Tasks like glancing back to re-read, comparing documents, and interleaving reading and writing all benefit from the fact that content is distributed across several sheets of paper or multiple documents. Moreover, since sheets of paper are thin and lightweight, they can be picked up, laid out, and rearranged effortlessly, providing organizational and cognitive benefits that aid reading tasks that involve several documents [O’Hara et al. 2002]. It follows, then, that if electronic devices are to more fully support the reading activities of knowledge workers and students, they must provide similar capabilities that will enable readers to spread out, navigate through, and work with multiple documents or pages at the same time.

One strategy is to simply increase the size of the display to enable readers to see more than one document at a time. This is the approach that large tabletop computers like Microsoft’s Surface [2010] have adopted. Even if we assume that tabletop systems will eventually become somewhat portable, a major drawback is that increasing the size of the display obviously undermines the mobility of the system. In the context of reading, reducing mobility is undesirable as it limits when and where reading can be done [Tashman and Edwards 2011a] and also removes an important element of physicality from the reading process [Marshall 2005].

In this paper, we explore another approach. Our system, *United Slates*, is based on a small collection of moderately sized, but highly portable slate devices. This approach draws inspiration from how paper distributes content across several different sheets that are individually mobile, while at the same time taking advantage of the power of the digital world through the interconnection of these displays or slates. Although each display independently provides features tailored for reading, such as highly readable E-Ink screens and the ability to capture and render freeform ink markings, it is this integration of the separate displays that makes our system truly unique. We present the design and implementation of this new electronic reading system and highlight how it has the potential to more completely support the reading needs of knowledge workers and students than existing digital solutions.

We begin by surveying the literature detailing the reading requirements of students and knowledge workers. This literature review helped inform the goals of the system, and was used to specify the hardware and software requirements of our system. The analysis highlighted the shortcomings of current solutions in supporting key features of reading, thus further bolstering the case for taking a different approach. In particular, our analysis highlighted the fact that commercially available hardware tended to be too heavy and



**Figure 1. Different usage configurations of our multi-slate reading system. The system can be used to support workflows that include the PC, for working with multiple documents, and can adapt for use in diverse environments.**

lacked support for writing. For this reason, we collaborated with an industrial partner to develop custom E-Ink based reading hardware. We describe how we validated our hardware during a 2.5 month long technology trial which was designed to reveal any unforeseen issues (technical or otherwise) that could arise with this kind of multi-screen reading system.

Next, we outline the distributed interface that underpins the unified reading environment we create across slates. Our interface primarily focuses on supporting navigation activities, resolving physically awkward slate interactions, and finding more integrated ways of dealing with documents within the context of the overall active reading workflow. This is done through the interconnectivity across slates, but relies on new interaction techniques designed specifically for use across multiple, connected slates. For instance, we will describe how we support rapid switching between pages across documents, enable quick side-by-side comparison of documents, remote control of operations across slates, support for reading in a variety of environments, and the ability to federate with other electronic devices such as PCs. By streamlining information exchange across devices and providing a method for working across devices without physically switching the device one works with, our system lowers the barriers of working in a multi-device environment (Figure 1).

To get a sense of what aspects of our system offer the biggest gains in terms of supporting reading activities, we conducted a preliminary evaluation of our system by having participants compare using a set of slates that incorporated the multi-slate reading tools to single devices that did not have these tools available. Based on our results, we conclude that our system's ability to fluidly integrate with PCs, to provide rapid non-sequential navigation across many documents, and to provide an integrated solution to working with large collections of documents are the most essential functions to immediately incorporate into reading devices for knowledge workers and students. In summary, the main contribution of our research is showing how the entire reading workflow, ranging from the page to the workspace can be supported through reading

tools that take advantage of the multi-slate setup. It is through this comprehensive support of the reading process that renders our system unique from other systems in the literature. Moreover, we believe that this type of multi-device environment can also be further applied to other computing activities.

## 2. RELATED WORK

### 2.1. Reading Requirements for Knowledge Workers

Reading is a multi-faceted and complex activity and as a result, a great deal has been written about its processes and requirements. The literature makes it very clear that there are many different types of reading. Pugh's [1978] study of how university students read identified five distinct styles of reading in which students engage. These include: receptive reading, the process of linearly progressing through the text without interruption; reading to search for a specific piece of information; acquisition of information without a set goal; reading to get an overview about the general structure of the material; and responsive reading, which is the process of actively engaging with the material in the form of note taking, annotation, and cross-referencing. This analysis is mirrored in Adler et al. [1998] who focuses on the work-related purpose of reading, and highlights the extent to which work-related reading tasks involve writing and annotation as well cross-referencing. The reading activities that involve writing, annotation and cross-referencing are frequently referred to as "active reading" [Adler and van Doren 1972]. It is, however, important to note that active reading tasks do not occur on their own [Thayer et al. 2011]. Rather, these tasks are punctuated with receptive reading and skimming activities. Thus, one cannot support active reading without also supporting these other types of reading.

Another important finding of ethnographic studies was the relationship between reading and writing and the number of documents that are used concurrently. Adler et al.'s [1998] work is informative in this respect, reporting that reading happens more frequently with writing than without, and that reading is largely performed across several documents or displays (e.g. monitor, sheet of paper) concurrently. O'Hara et al.'s [2002] study of a wide variety of professionals and students performing writing tasks also describes the concurrent use of multiple displays. In particular, O'Hara et al. mentions the savings gained by being able to quickly glance across displays or pages. O'Hara et al. also found that the spatial layout of these materials in the workspace served an important role in supporting readers' mental organization of the material. Marshall's work documenting annotation practices in a university setting [Marshall 1997] is also applicable here. In particular, Marshall regards the ability to smoothly integrate annotation with reading to be one of the most essential challenges for any reading system. These findings about annotation, spatial layout of information, and interleaving reading and writing are further corroborated by O'Hara and Sellen's [1997] laboratory study comparing reading on paper to reading on a PC. Taken as a whole, these studies strongly motivate system support for seamless annotation and writing capabilities and the ability to work across multiple documents simultaneously.

More recently, Tashman and Edwards [2011a] conducted a diary study and design workshop to determine how users believe existing active reading practice can be improved. The major findings revealed a tension in which users wished to have more space for their multi-document tasks and more space to view content, but also wanted their reading workspaces to be portable.

### 2.2. Experiences with Electronic Reading Devices

In addition to studies about how people read on paper, researchers have also examined the use of electronic reading technology. From these reports we can glean more insights about how to support reading by examining what aspects of these systems have been successful, and where these systems fall short.

The XLibris [Schilit, Golovchinsky, and Price 1998] reading appliance was an experimental reading appliance specifically designed for active reading. As a research device, researchers have evaluated its use with various groups of people including researchers in a reading group and legal scholars [Marshall et al. 1999; Marshall et al. 2001]. Users in these studies found that XLibris' document presentation, annotation, information extraction, and non-linear and cross-document navigation features facilitated reading activities. On the other hand, the legal scholars who evaluated the prototype expressed a need for the device to better integrate with the work on PCs, leading researchers to wonder whether a laptop with annotation capabilities would be more suitable.

Marshall and Ruotolo [2002] report on the deployment of PDAs for use with course materials for university students, and found that the high portability and availability of the devices and text search capabilities were advantageous. However, the small screens on the devices meant that layout-sensitive texts were difficult to use and random-access navigation on the devices was difficult, particularly in long documents.

Wilson and Landoni's [2003] survey of several different electronic reading devices similarly found that basic text search capabilities on the devices were extremely helpful. However, they also reported that as a whole, electronic devices needed to better support working with multiple documents, and provide awareness about one's place within a large document. Recent deployments of commercial e-books (generally those employing

---

**Where Described in the Literature**

---

**Hardware / Physical Requirements**

Mobility; thin, lightweight, graspable	Marshall and Ruotolo 2002; Marshall 2005; Wilson and Landoni 2003
Readability	Wilson and Landoni 2003; Young 2009
Support for writing	O'Hara and Sellen 1997; Adler et al. 1998; O'Hara et al. 2002; Sellen and Harper 2003; Thayer et al. 2011

---

**Page Level Requirements**

Linear reading	Adler and van Doren 1972; Pugh 1978
Graphics: Page Layout / Illustration	O'Hara and Sellen 1997; Marshall and Ruotolo 2002; Young 2009; Behler 2009;
Superimposed annotations	Adler and van Doren 1972; Marshall 1997; Thayer 2011
Text search	Wilson and Landoni 2003; Young 2009
Glancing back to re-read	Marshall et al. 1999; Marshall et al. 2001
Kinesthetic cues for page-turning	O'Hara and Sellen 1997; Thayer et al. 2011

---

**Document Level Requirements**

Non-sequential navigation	O'Hara and Sellen 1997; Behler 2009; Tashman and Edwards 2011a
Cognitive map (awareness) of content	O'Hara and Sellen 1997; Wilson and Landoni 2003; Thayer et al. 2011
Skimming to get overview	Pugh 1978; Adler et al. 1997
Discovery of topical knowledge	Pugh 1978; Adler et al. 1997
Switch between navigation styles	Thayer et al. 2011

---

**Workspace Level Requirements**

Sorting and triage of documents	Adler et al. 1997; Marshall and Shipman 1997
Spatial layout	O'Hara and Sellen 1997; O'Hara et al. 2002
Extracting information	Adler et al. 1997; Morris Brush and Meyers 2007
Reading from multiple documents	O'Hara and Sellen 1997; O'Hara et al. 2002; Wilson and Landoni 2003; Tashman and Edwards 2011a
Integrating with PC workflows	O'Hara et al. 2002; Morris, Brush and Meyers 2007; Tashman and Edwards 2011a; Thayer et al. 2011

---

**Multi-session Reading Requirements**

Reading in different venues	Sellen and Harper 2003; Thayer et al. 2011; Tashman and Edwards 2011
Filing and archiving	Sellen and Harper 2003
Recreating reading workspace, resuming	Wilson and Landoni 2003; Tashman and Edwards 2011

**Table 1. A summary of the reading requirements for knowledge work, as detailed in previous research.**

electronic paper displays) into the classroom [Young 2009; Behler 2009] have found serious deficiencies in the navigation systems in these devices for classroom reading. These navigation issues often stemmed from the slow refresh rates on the displays. Another problem these studies identified was that illustrations, color ones in particular, were not reproduced with sufficient fidelity. Finally, Thayer et al. [2011] produced an extensive report of Kindle DX use in an academic setting. Thayer et al. likewise found that the device they tested lacked support for the diverse navigation needs of student readers and that the lack of freeform ink annotation capabilities severely limited students' abilities to read responsively.

Table 1 presents a summary of this literature, and in doing so also provides guidance as to the range of features identified as important in the support of active reading. These range from some core features of the devices, to features important at the page, document and workspace level. In addition, there are some more general requirements we distill out to do with the support of document tasks across space and time.

### 2.3. Distributing Material Onto Tangible Surfaces

Our system adopts the strategy of distributing content onto several independent slate devices in light of the importance of supporting freeform annotation alongside multiple simultaneous document use. The technique of distributing content onto separate tangible surfaces has recurred in the literature. Early work like VideoMosaic [Mackay and Pagani 1994] and DigitalDesk [Wellner 1993], were not designed specifically for reading activities, but demonstrated how spatial layout capabilities afforded by independent physical surfaces can be leveraged to provide a richer experience, while at the same time being augmented by computers. PaperWindows [Holman et al. 2005] explores interaction techniques with flexible paper-like displays. PaperWindows makes the assumption that electronic displays will eventually have the same physical properties of paper. In contrast, our system takes a more restrained view of the future and incorporates designs sensitive to the limitations we expect slate-type devices will impose in the near future. The dual-display Codex [Hinckley et al. 2009] provides two screens that can be separated to form an environment with two independent displays. Codex, in particular, explores the interactions that are possible between two screens but does not examine the logistical nor interactional issues that arise when the number of devices that are available is increased or reduced.

In terms of systems specifically designed to support reading, PapierCraft [Liao et al. 2008] enables the use of multiple surfaces by allowing users to move between documents on Tablet PCs and on sheets of digital paper. Information can be shared between surfaces using an inking-based command system. Our system shares similar goals with PapierCraft but explores the setting where all surfaces are electronic and interactive. The Dual-Display e-book [Chen et al. 2008] focused on improving navigation with the inclusion of an extra display, and investigated the benefits of providing a two-page linked reading view. But, like the Codex above, did not consider situations where the number of screens was not equal to two. Most relevant to our present work is Morris, Brush and Meyers' [2007] evaluation of a system composed of three independent Tablet PCs (not functionally linked). Users in this study liked the spatial layout and annotation capability of the tablets, but reported issues that stem from the inability to move information between tablets.

### 2.4. Connecting Slates

Morris, Brush and Meyers' [2007] experience highlights a problem with the default multi-slate configuration, which is that separate slate devices are computationally isolated. The isolation is problematic because it limits the devices' capability to work together in a collaborative fashion. Even though the literature does not provide prior examples of how to construct an interface to unify slate devices in a reading application,

there are several techniques that we are able to adapt for the purposes of connecting our reading devices together.

The most frequently encountered example of using multiple screens is that of multi-monitor computing. However, multi-slate systems are slightly different from traditional multi-monitor systems in that the relative positions of monitors tend to be static and can share a unified coordinate space. In contrast, in our approach, slates are treated as separate devices and separate spaces, and the focus is instead on bridging these spaces.

One important requirement for bridging these spaces is to simplify the movement of information across the spaces. A great deal of work has been done in streamlining data transfer between devices. The most examples most relevant to our work are SyncTap [Rekimoto 2004], Pick-and-Drop [Rekimoto 1997], Synchronized clipboard [Miller and Myers 1999] and Stitching [Hinckley et al. 2004]. The one aspect all of these techniques have in common is that they offer users a fast and direct way of specifying the endpoints of a transfer operation.

Another strategy to help unifying disparate devices is by allowing one device to control another remotely. Remote control spans the spectrum of functionality provided by Virtual Network Computing [Richardson et al. 1998], which provides a full proxy to a different computer system, to systems like Pebbles [Myers 2001] where portions of the input and output stream are redirected to control other devices.

Finally, with improvements in sensing technology, physical connection of devices as a means of specifying relationships between devices has also been explored. Examples include Siftables [Merrill, Kalanithi, and Maes 2007] and ConnecTables [Tandler et al. 2001]. Although we believe that these interactions are valuable, slate devices have appreciable mass and activities that result in having to repeatedly connect or disconnect devices can be cumbersome. As such, we must recognize that physical connection can be useful but may not always be appropriate.

## 2.5. Comparison of E-Reading Technologies

We conclude this section by presenting a summary of how various technologies currently support the reading activities we have found to be central to active reading. Table 2 shows an analysis of the major technological alternatives. When populating the table, we attempted to factor in the latest developments for each platform. Because of its current prevalence in reading, paper is often considered the reference system when it comes to supporting active reading and we indicate it as such in our table. However, in some cases, such as dealing with large documents collection paper suffers from limitations stemming from the cost of distributing, transporting and archiving paper documents [Sellen and Harper 2003].

The traditional personal computer, which includes both desktop and laptops in our analysis, is a versatile tool well suited for composition and editing tasks that often go hand in hand with reading. PCs have robust multi-tasking capabilities along with comparatively large screens making them better suited for cross-referencing, re-reading, and sorting tasks. The lack of pen input was addressed in part by the Tablet PC system. But, the PC's focus on being very general, all-in-one tools results in software interfaces that are more complex and hardware that tends to be bulkier and more difficult to handle.

Although we are not aware of a reading system specifically developed for a tabletop configuration, multi-touch computing surfaces like the Microsoft Surface are probably good candidates to support reading for knowledge work: Tabletops, by virtue of allowing direct manipulation of objects spread across a large surface are suitable for spatial layout and working with several documents at once. Certain tabletop configurations are also able to receive high-resolution pen input. At the same time, even if we take into account future developments like flexible displays, tabletops will still constrain where and how reading is performed. For one, tabletops make it difficult to support the physicality of reading [Marshall 2005] because documents are locked on the surface. Additionally, with

	PC	TabletPC	Tabletop	Pad Appliance	E-book Appliance	Paper
<b>KEY</b>						
	○ low support	◐ moderate support	● good support			
<b>Hardware / Physical Requirements</b>						
Mobility; thin, lightweight, graspable	○	◐	○	●	●	(●)
Readability	●	◐	○	◐	●	(●)
Support for writing <i>(Note 1)</i>	○	●	◐	○	○	(●)
<b>Page Level Requirements</b>						
Linear reading	●	●	●	●	●	(●)
Illustrations	●	●	●	●	○	(●)
Superimposed annotations	○	●	◐	○	○	(●)
Text search	●	●	●	●	●	○
Glancing back to re-read <i>(Note 2)</i>	◐	○	●	○	○	(●)
Kinesthetic cues for page-turning	○	○	○	○	○	(●)
<b>Document Level Requirements</b>						
Non-sequential navigation	●	●	●	●	○	(●)
Cognitive map of content	◐	◐	◐	○	○	(●)
Skimming to get overview	◐	◐	◐	◐	○	(●)
Discovery of topical knowledge	◐	◐	◐	◐	○	(●)
Switch between reading styles <i>(Note 3)</i>	◐	◐	○	○	○	(●)
<b>Workspace Level Requirements</b>						
Sorting documents	◐	◐	●	○	○	(●)
Spatial layout	○	○	●	○	○	(●)
Extracting information	(●)	●	●	◐	○	○
Reading from multiple documents	◐	◐	●	○	○	(●)
Integrating with PC workflows	(●)	◐	◐	○	○	○
<b>Multi-session Reading Requirements</b>						
Reading in different venues	(●)	●	○	●	●	○
Filing and archiving	(●)	●	●	○	○	◐
Recreating reading workspace	◐	◐	◐	◐	◐	○

Notes on the table: 1. Although there are styli for capacitive multi-touch screens, they offer far less accuracy or resolution. 2. Glancing back to re-read a function of how much content can be viewed at once, hence better scores for devices with more screen area. 3. Ability to reconfigure the interface for different reading needs.

**Table 2. Comparison of existing reading solutions' support of reading for knowledge work. Parentheses denote what are generally regarded as the "target" for interaction experience.**

a single large screen, these devices would necessarily require environments that provide a correspondingly large and flat working area.

Pad appliances such as the iPad, depart from the standard laptop/Tablet PC interface by subscribing to a full-screen application model. The majority of appliances these days forego a pen digitizer, instead relying on touch, or imprecise capacitive styli (still considerably less accurate than the inductive digitizers found in TabletPCs) that only support writing to a limited degree. Nevertheless they are proving to be good reading platforms for single documents. Moreover, the multi-touch capability found in almost all of these devices offers many possibilities for enhancing navigation, such as those demonstrated in LiquidText [Tashman and Edwards 2011b].



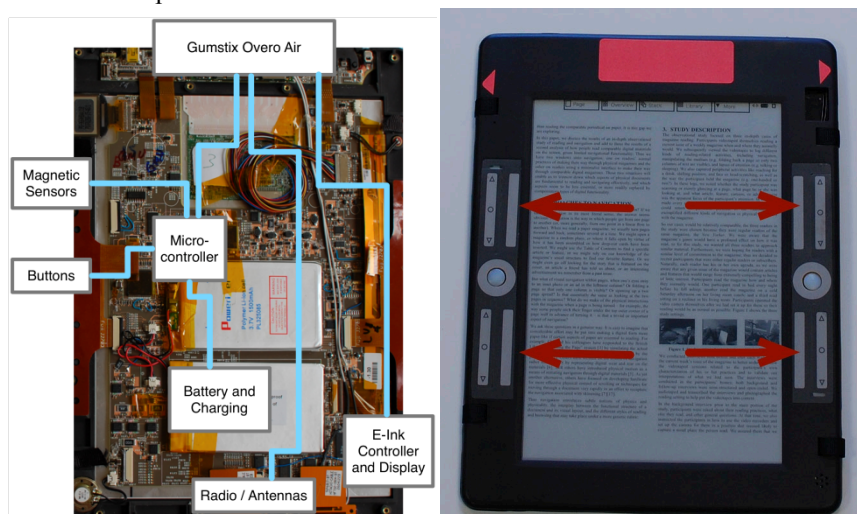
Dedicated electronic reading devices, which include products like the Amazon Kindle, offer highly readable screens, extremely low weight, and long battery lives, making them well-suited for receptive reading. However, the slow refreshing black-and-white displays (from which many of their beneficial physical properties derive) limit the navigation capabilities as well as the content that these devices can reproduce. With the exception of devices from iRex (now defunct) these devices only support simple highlighting and textual annotations.

Looking at the table, we can surmise that a system able to combine the physical properties of a dedicated e-book reader, with the pen input of a Tablet PC, and the multi-document functionality of a large working surface like a tabletop system would significantly advance the state of the art to provide a comprehensive system that can begin to rival the capabilities of paper.

### 3. PRELIMINARY DESIGNS FOR A MULTI-SLATE READING SYSTEM

#### 3.1. Hardware Design Choices

Based on the literature about reading, we knew that at the most basic level, the devices we needed for our system would have to be extremely mobile and easy to handle. The devices also needed to have pen-based annotation capabilities. Such a device did not, and still does not exist commercially. As a result, we were compelled to create a set of prototype reading devices with an industrial partner (Figure 2). These devices had a 9.7” E-Ink electronic paper display (EPD) placed over a Wacom pen digitizer. The EPD offered paper-like readability and extremely low power consumption compared to standard LCD displays. Furthermore, the use of an EPD helped reduce the thickness and weight of the device by virtue of a thinner screen and smaller battery requirements (At 8.5mm thick and 500g, each slate compares favorably to devices like the Amazon Kindle DX and Apple iPad). One tradeoff the use of an EPD entailed was that highly interactive visualizations of the document that are normally possible with LCD screens would not be possible due to the slower refresh rate of the display. Also, the lack of an active backlighting system made placing an overlay to sense touch input impractical, as it would reduce the contrast and display fidelity. Our discussion section details implications of our hardware component choices.



**Figure 2. Left: internal photographs of the slate devices. We have annotated key portions of the photograph to indicate components of interest and how they are connected. Right: Front view of the slate. There are two large, white, circular buttons and four banks of touch sensitive buttons (indicated by arrows).**

The computing and communications capabilities come from a Gumstix Overo Air computer-on-module featuring a Texas Instruments OMAP3 application processor, plus Wi-Fi and Bluetooth radios. The choice of a smartphone-class processor was to reduce power consumption. The dual radios enable the devices to communicate amongst themselves, as well as through the cloud via near-ubiquitous Wi-Fi access points.

Our slates additionally provided a rich set of input and sensing electronics beyond the pen digitizer. Controls on the device include two large buttons, plus four additional tactile buttons that can also capacitively sense finger position and gestures on their surfaces (Figure 2). We replicated the controls around the device to support a variety of grips and reading positions. Inside the device, pairs of magnets and Hall-effect sensors are embedded along the long edges to detect when two devices are placed against each other.

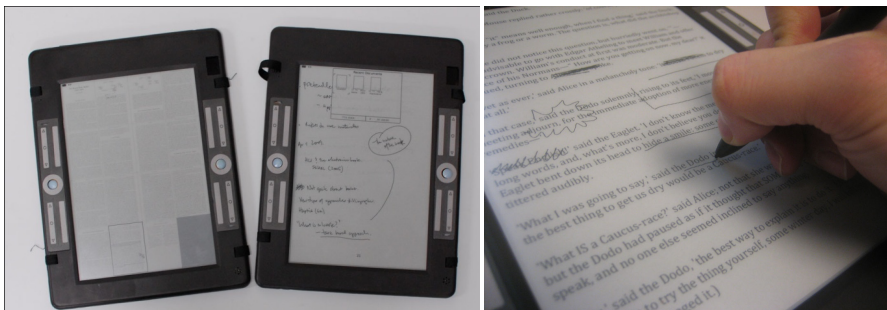
### 3.2. Technology probe

With the exception of Morris, Brush and Meyers' [2007] work, our knowledge of the interactional needs for multi-slate reading derives from what we know about people working with paper. Since slates and paper differ significantly in terms of physical characteristics like weight and thickness, not to mention their interactional capabilities, users most likely would not use slates in the same way they use paper. To better understand how slates would be used for active reading, we conducted a 2.5-month long trial with a technology probe with 13 undergraduate English students in the Spring 2010 semester. In this trial, we first assessed students' reading activities with a single slate. Then, after 1.5-months, we provided a second slate and demonstrated additional features. During the course of the study the students were asked to use one or both devices to carry out the majority of their course readings. We conducted bi-weekly interviews with students to gather feedback. When possible, we modified the software on the slates to reflect the feedback.

#### 3.2.1. Software configuration

Each slate displayed one full page of a document at a time, without margins, so that pages of letter-sized documents could be viewed without scrolling. We pre-loaded the devices with course readings after consulting with the instructor. Students did not have the capability to place their own documents on the devices.

To turn pages, we relied on the hardware button on the slates. For navigation over longer distances we provided a Space-Filling Thumbnail (SFT) [Cockburn, Gutwin, and Alexander 2006] overview of a document, a natural choice given the relatively slow refresh rate of the EPD. On each page, users could apply freeform ink on the pages with a stylus without entering any special inking modes (Figure 3). When more than one device was in use, ink marks made on one slate were transmitted electronically to the other slate so that the annotations on both slates remained consistent. The system also allowed



**Figure 3. Left: Reading device UIs for the technology probe. Left device shows a Space-Filling Thumbnail (SFT) overview of the document. Right device displays a notebook page that has been populated with ink notes. Right: Freeform ink superimposed on a page.**

students to create “notebook” documents, which were special documents with 50 blank pages. Users could ink on these pages and they could also copy and paste regions of pages into the notebook. The clipped regions linked back to the location from which they were extracted so that the user could view the context of the clipping.

### 3.2.2. *Lessons Learned During the Deployment*

Students were particularly enthusiastic about the portability aspects of the device, opting to carry the device in lieu of the course readings on paper. Students also mentioned that the form factor and weight of the device were often preferable to a book because it was easier to operate with a single hand and did not flop around. The free-form ink annotation also received a large number of positive comments. Students reported writing more on the electronic devices because they felt they were not “defacing” the original document due to the perceived reversibility of electronic ink marks. The difference reported was particularly pronounced for books, which many students said they wanted to keep pristine. Furthermore, students mentioned that the rigidity of the device actually made annotating easier than on paper. In fact, the simple presence of a pen was a plus in itself; like readers in O’Hara and Sellen’s [1997] study, many students reported that reading with a pen was useful for helpful for maintaining one’s place in the text. Based on this feedback, we were reasonably confident that the hardware design of the devices—the support for annotation, in particular—were in line with our goals.

Perhaps the biggest shortcoming of the devices we deployed was the fact that the devices did not integrate well with the existing electronic devices students employed. Students desired a way to put documents on the device and also be able to move data off of the device in support of their writing activities. Students reported that the majority of their composition-heavy tasks occurred on their PCs. Additionally, many other daily activities revolved around their PC. Users often reiterated their desire to use the slates in conjunction with their computers. This result corroborates a large number of studies that make observations regarding the integral role PCs play in many reading tasks [Marshall et al. 2001; Morris, Brush, and Meyers 2007; Tashman and Edwards 2011a].

In addition, while users reported that sequential reading was satisfactory on the devices, many users ended up working on the paper versions of documents when they needed to complete writing assignments and review for exams. This was particularly pronounced for large books. Students’ experiences working with one large text that was several hundred pages long proved particularly illustrative of the challenges they faced. For this particular book, we split up each chapter of the book into a different document in order to simplify access to each chapter. So while SFT could be used to navigate within the chapter, navigating across chapters required a lengthier process of changing documents. Student comments indicated that navigating with SFT was bearable but that for their writing and review activities, they frequently needed to consult disparate information found in the other chapters and it was simply easier to perform that type of navigation with the paper book. These comments highlight the fact that our system needed to not only support rapid non-sequential navigation within a single document, but also needed a strategy for providing equally quick access across different documents. The wide range in navigation practices we observed is similar to those that Thayer et al. [2011] noted in their deployment of electronic reading devices.

Finally, even though we only provided users with two slates in the probe, we were able to observe interesting differences between how the students made use of the additional slate. The number of slates a user employed was heavily dependent on task and individual preferences. For example, users preferred only a single slate when they were on the go, but mentioned that multiple slates were useful for content extraction tasks into a notebook. These comments confirm the fact that practical issues like size and weight will limit how many slates a user will be able to use at any given time.

### 3.3. Impact on our final design

The technology probe directed our attention to critical parts of our system that we needed to refine in order to produce a viable reading platform. They include the following key points:

- Slates are not just “hard paper”. The physical size and weight of a slate device, plus the fact that each device multiplexes several pages on a single physical object means that slates are used differently from paper. As a result our design should seek to *transcribe* paper based interactions, not replicate them on the new platform;
- Slates should be tightly interconnected with each other and other tools like laptops. We believed we had addressed Morris et al.’s recommendations by implementing a distributed clipboard function (which users used, and liked), but this was not enough. A successful system must take extensive advantage of the available screen real-estate to simplify interactions with each document. It is also important that the multi-slate systems be easy to use during the text creation phase when users are mostly typing text on the computer;
- The “ideal” number of slates changes over time – The number of slates a user employed was heavily dependent on task, environmental constraints, and individual preferences. As such, a system should not force users into using an unchanging quantity of slates. Rather, it is important for a successful system to offer a way to reconfigure itself and remap content as the number of devices change. This ability is particularly important in light of the related research describing the importance of allowing users to read in a variety of venues.

## 4. DESIGN OF THE “UNITED SLATES” SYSTEM

We now present an overview of our system design drawing together insights from the existing literature, and from our initial deployment of our first prototype multi-slate system. Our treatment of the system can roughly be divided into four parts. The first two parts are concerned with establishing the basic concepts behind the multi-slate environment and consist of how to distribute information to slate devices; and how to move this information around efficiently. These functions are mostly targeted at reading tasks at the page, document, and partially the workspace level (as outlined in Table 1). Then, we turn our attention to additional functionality that enables slates to work with PCs in the workspace and finally, we discuss how our document and workspace model supports reading activities that span multiple reading sessions, venues and devices.

### 4.1. Mapping Information to Slates

The conceptual mapping of information to slate devices has important implications in terms of how successful users will be in working with multiple documents across multiple slates. The mapping we select will determine how many devices a user must have in order to conduct reading activities and how easily users can navigate through the information on a slate.

We first consider the most finely grained mapping in which each page is mapped to a slate. This configuration makes sense because the display of each one of our slates can show one full page of printed content (assuming Letter/A4 sized sheets of paper). However, the disadvantage is the inherent mismatch between the physical characteristics of a slate and a sheet of paper. The fact that for the foreseeable electronic devices will not have the thinness, lightness of a sheet of paper, suggests that this mapping is not ideal.

Going up a level, we might consider a system in which each slate represents a single document. Viewing slates at this level also makes a great deal of sense since grouping pages up into a single unit is a familiar concept. In order to support this model, our system provides facilities to navigate within a document, which include hardware controls for page turning within a document, and Space-Filling Thumbnails of the current document. However, the downside to this approach comes to the fore when considering

larger collections of documents. In particular, working with many documents simultaneously would be difficult since a one to one mapping between documents and slates will require the user to work with too many slates at once.

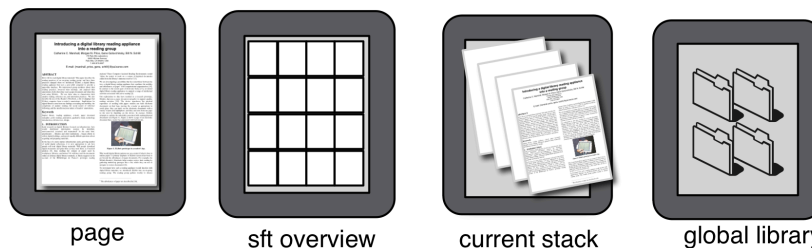
Alternatively, each slate can represent a stack, or collection, of documents. This is the mapping implemented in the “Library” on many reading appliances because it allows a single device to represent a much larger set of documents. This is the central mapping we decided to adopt in our system. On commercial reading appliances, however, there is a single library, and therefore one stack. Having only one stack to hold all of a user’s documents can present a problem when the number of documents is large. As with stacks of paper, increasing the number of items in the stack can make managing the stack or navigating to a particular document in the stack tedious. Thus, our model assumes that a slate represents an *active* stack of documents, along with several inactive stacks that can be swapped in at will. This model resembles how knowledge workers employ a small set “hot documents” that are actively in use, and an archive containing the remainder of the documents [Sellen and Harper 2003]. The inactive documents continue to be available through a global library, which is shared among all slates. The active stack and global library metaphor is additionally useful for the practical interfacing of the slate environment with the outside world. The shared library can be represented as a special DropBox [2011] folder on a PC. To get new documents into the library, users simply drag and drop documents into that folder, which becomes shared with the entire reading environment.

#### 4.2. Continuum of Document Views

Although a slate represents a stack of documents, we still need to be able to view the contents of that slate at the page, document, and stack levels in order to read the page, or navigate through the document. To support viewing the content of each slate at these different levels, each slate in the system presents the active content on the slate using one of four standard views (Figure 4). The first view is a page-level view of the document, the second is a space filling thumbnail (SFT) [Cockburn, Gutwin, and Alexander 2006] overview of the information, the third shows the working set of the slate, and the final is an interface into the shared library across all of the devices. To take full advantage of the available screen real estate, each view is always shown full screen. This approach simplifies the interface by avoiding inefficient tiling, and complicated panning or scrolling maneuvers that would be necessary if multiple documents or views shared the same screen.

The key thinking behind our reading system is that users will make use of more than one slate. The composition of slates displaying different combinations of the four views supports many reading activities, as we will show in the following sections. Furthermore, using multiple devices allow information to be spread out spatially, which allows the slates to conform to users’ mental models of the content, which is important for

#### 4 Levels of Views in Our Reading System



**Figure 4. Our system provides four levels at which users can interact with documents. Our multi-slate interactions enable combinations of these views to support reading activities.**

organizing and keeping track of information during reading activities [O'Hara et al. 2002] The cognitive benefits are potentially reinforced by the fact that the information is on physically separate devices, allowing users to take advantage of physical partitioning between devices [Grudin 2001] to better isolate distinct groupings of information.

#### 4.3. Moving Information Around

While the initial distribution of information across the workspace is important, reading activities additionally require this information to be manipulated and moved around. For instance, on paper, the process of moving information around allows users to access, organize, and compare information. The key realization is that there are many ways information is moved around in the environment. For paper interactions, similar tasks may involve completely different physical manipulations. For instance, while annotating a document placed far away, a user might reach and annotate in-place, temporarily bring the document closer and then put it back, or more permanently reconfigure the workspace. Depending on their needs, users decide on the fly which movements to employ. Every method of moving information imposes some type of cost/benefit tradeoff making certain movement strategies better suited for a particular task than others. Factors that can influence the appropriateness of a particular transfer technique include the speed to perform the transfer, the physical effort required to specify the relationship, and the duration of the interaction (across which the setup costs are amortized).

One important characteristic of paper is that moving information around is synonymous with moving the physical sheets of paper. In the case of our slates, moving the physical devices is not only unnecessary, since we can simply move the information over electronic channels, but undesirable because slates are heavier and thicker than sheets of paper. Instead, what we strive for in our system is to enable users to achieve a similar degree of information movement, but to use electronic functionality to decouple that from physical manipulation of the slates. Nevertheless, electronic interactions have similar tradeoffs of speed, setup cost, etc. as paper.

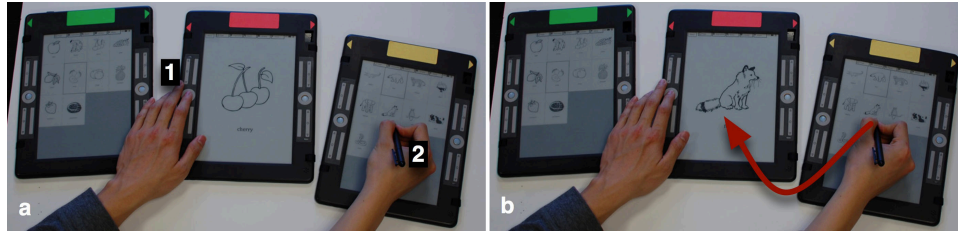
In light of all this, our reading environment provides a range of carefully thought out interactions for performing information movement electronically between the views on our slates. By electronically linking together slates with these interactions, we are able to provide a wide range of electronic tools that enhance reading. We describe these interactions and the features they enable in more detail below.

##### 4.3.1. *Point-to-Point connections between slates*

To provide an unambiguous and fast way to move information between views, we aimed to give the user a direct method of specifying the slates participating in the operation while also taking advantage of bimanual processes present in reading [O'Hara and Sellen 1997]. Having the user directly specify the source and destination is well suited for our application, where the slates are in the vicinity of one-another. Some of the advantages direct interaction brings over a symbolic approach (like picking devices from a list) include reducing confusion about where data is going, side-stepping the complexity of mapping a device to a name, and supporting the use of anonymous displays, which abstract away the idea that each new screen is a separate computer [Rekimoto 1997].

In the resulting technique, the non-dominant hand is first used to designate a target and then the dominant hand, which offers more precision, selects the item to be transferred. One way to visualize the process is that the user's body forms a **Conduit** through which the information is transferred. Conduit operations are particularly well-suited for activities where content needs to be transferred to a device with which the user is actively interacting, since the user's non-dominant hand is already close to, or holding that device.

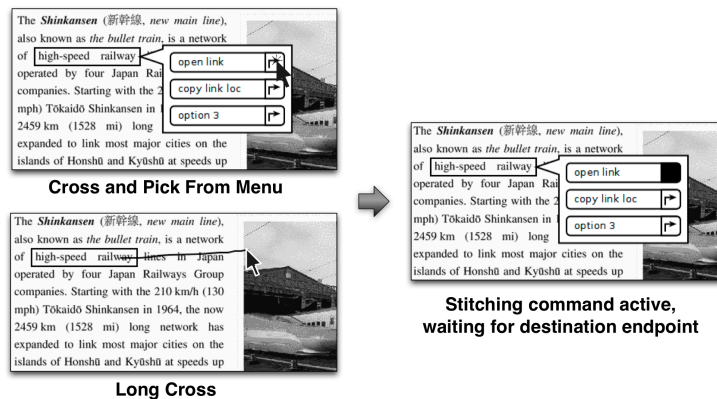
One way we use the Conduit technique to move information between devices is to allow users to navigate to the page corresponding to a thumbnail on a device other than



**Figure 5. Distributed thumbnail navigation using the Conduit interaction technique. Left: We prepare to navigate to a page on the red slate using a thumbnail on the yellow slate by depressing the command button (1), tapping on desired page (2). Right: Result of the operation; note that the user acts as a Conduit through which the data flows.**

the one where the thumbnails are displayed. Retargeting the thumbnail navigation makes it possible for users to have devices that are dedicated for thumbnail overviews. To navigate, the user first depresses one of the large circular buttons on the device where he wishes to view the thumbnail in detail (which we will subsequently refer to as the **command button**), which specifies the destination slate of the command. Then, the user selects one of the page thumbnails shown in the overviews (Figure 5). The slate on which the command button is depressed navigates to the selected page. In this way, the user has the ability to quickly jump to any page to see it in detail, without having to repeatedly switch between page and overview views. With multiple panels of thumbnails, each corresponding to a different document, the user can rapidly access pages across different documents. Another situation where Conduit is used in our system is for transferring documents from the working set of one slate to another via the Stack View.

Although Conduit makes quick transfers possible, one key limitation is that the transfers are asymmetric toward the slate the non-dominant hand controls. Moreover, given its simplicity, it is also not very well suited for commands where parameters are needed at both the origin and destination, such as copy and paste. For these cases our system offers a variant of Stitching [Hinckley et al. 2004]. The basic mechanics of Stitching are the following: First, a UI element on the source slate is crossed through with the command button depressed, which causes a context menu to pop up. Then, the



**Figure 6. The process of activating a Stitching operation with hyperlink UI elements. Users have a choice of selecting from full menu of commands (left, top) or using the shortcut of a long cross (left, bottom). Note the indicator icon is filled while the system waits for a destination endpoint for the command. (Figure is a screenshot from our device simulator and uses desktop cursors for clarity)**

stitching icon corresponding to a command is picked from the menu. Finally, the user taps on the destination slate, optionally specifying additional parameters for the command, at which point the command is executed. For interactional consistency, the source and destination slates can be the same in a Stitching operation.

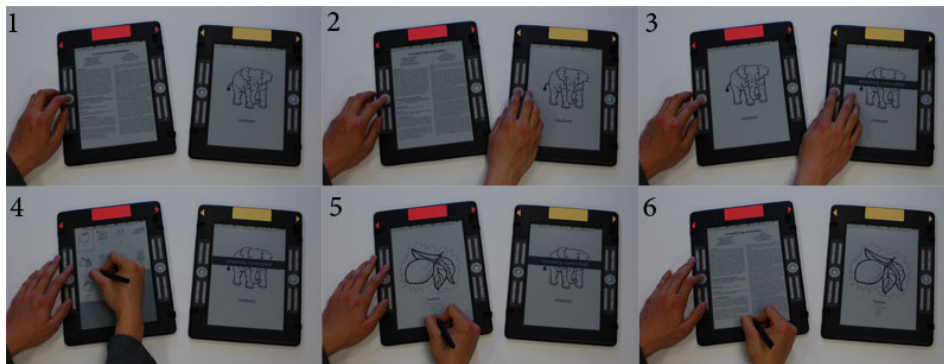
To further speed up stitching operations, our system provides an additional shortcut by allowing the user to cross through UI elements with a long cross to automatically select a default command, rather than having to pick from the menu [Dixon, Guimbretière, and Chen 2008] (Figure 6, left). Other crossing interaction variations can include marking gestures, which would allow expert users rapid access to an even larger set of commands. Lastly, it bears noting that since Stitching is simply a generalized case of Conduit, the two can be mixed and matched. Thus, one can also specify the where a hyperlink should open using Conduit, or Stitch a thumbnail to a device to view the page in detail.

At the page view level, Stitching hyperlinks to another device offers a way to explore hyperlinks on a page without having to navigate away from that page. By keeping the origin of the hyperlink visible, this feature helps maintain context for the newly opened link, supports comparisons between linked documents, and speeds up backtracking by not requiring extra navigation to return to the source of a link. Figure 6 illustrates the process of stitching a link from one slate to another.

#### 4.3.2. Proxying interactions through slates

One interesting feature that can be exploited in a multi-slate system is that since individual slates are functionally identical, one slate can easily double as another. The chief way we apply this functionality is in a remote control feature to provide interaction at a distance.

Since users have trouble stacking up and overlapping slates, they have a tendency to spread them out across a large area on the desktop. Unfortunately, certain interactions such as navigating, writing, or examining a page in detail are difficult to perform when a device is far away. Users can stretch to operate a slate, but that would be uncomfortable for lengthy operations. Users could also pick up the distant slate and move it closer, as we might do with paper. Unfortunately, if a user has a slate in hand already, this requires extra work because it is difficult to hold two slates at once given their size and weight. The remote control feature addresses these problems by allowing the user to use a nearby device as a proxy for a device that is far away. The functionality leverages the fact that users frequently have a device in hand while reading. By using the device in hand *as if* it



**Figure 7. Example of the slate remote control interaction. (1) Press and hold command button on local device. (2) Click command button on remote device (enhancement), (3) Remote screen is frozen and transferred (note the message informing the user that the session is active), (4, 5) Using local device as a proxy, (6) Double click command button to end teleport session.**



was another, we can provide the illusion the user has virtually picked up a distant device and brought it closer.

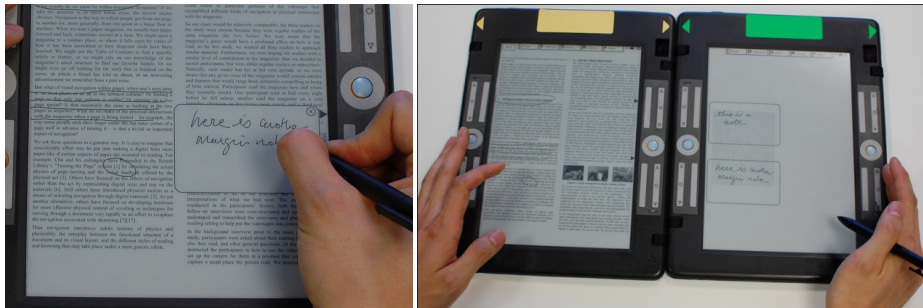
In our system, a Conduit interaction is used to activate the remote control feature. As with all Conduit interactions, the command button on the near device is depressed. Next, the tab corresponding to the view we wish to teleport to the device in hand is picked on the distant device, which sends that particular view to nearby device. When that occurs, the screen on the distant device is frozen and displays a message reminding the user that it is being remotely controlled.

When a remote control session is active, all of the functionality of the system is retained, with the exception of being able to invoke an additional remote control session, as we believed it would be difficult for users to keep track of multiple simultaneous sessions and chains of remote control sessions. In order to enable the normal controls to work transparently during the remote control session, we needed to introduce a special input sequence to terminate the remote control session. Thus, the user ends the remote control session by double clicking the command button on the nearby device, at which point the distant device updates to match the changes made while it was being remotely controlled, and the local device returns to its original state (Figure 7). One enhancement we made to make a remote control session easier to start was to let users push the command button of the distant device instead of aiming for an on-screen target on that device. This enhancement provides users a larger target to acquire, as well as tactile confirmation, which is helpful when the user must reach to access a distant device.

#### 4.3.3. Proximity links between slates

More permanent connections between slates can be useful for reducing repetitive actions to set up relationships between slates. However, invisible electronic links between slates run the risk of confusing the user if the user is not aware of the connection. For instance, Chen et al. [2008] reported that users were confused when displays on a second device changed unexpectedly while the user was working with a different device. These issues are an example of mode confusion. In order to reduce the chance of this type of mode confusion, our system allows more permanent connections to be formed based on slate proximity. This provides strong visual reinforcement that there is a relationship between devices, reducing the chance that slate state changes result in confusion. Although the cost of establishing a proximity-based connection between devices is relatively high, given the physical manipulation required, the resulting implicit connection between devices saves user effort later on, and is worth it for connections that last a long time.

Like the Dual-Display E-Book [Chen et al. 2008], we use proximity links to create a two-page, side-by-side view of a document, which effectively enlarges the amount of content one sees from a document at a given time. Proximity links are also used in our system to streamline the creation and viewing of margin callouts on the page view of a slate. Normally, when using on slate, users can get extra writing space by creating a



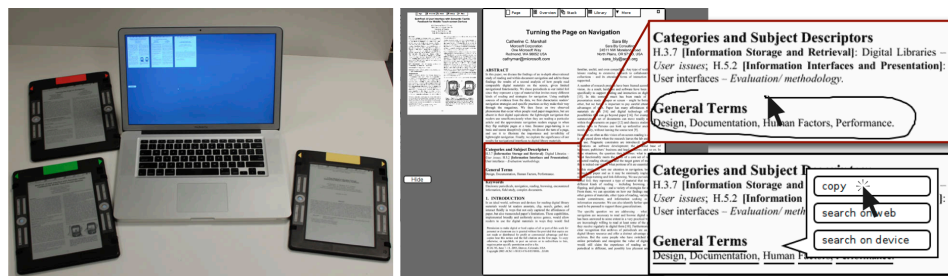
**Figure 8. Left: Single screen margin callout window. Right: Proximity links used to expand the view of a page to include margin notes.**

margin callout with an angle bracket gesture (>) against the edge of the screen. With a single slate, margin callouts need to be called up and viewed one at a time (Figure 8, left). When two slates are connected with a proximity link, the system expands the margin to the second slate; the right slate displays all of the margin callouts associated with the page shown on the left slate (Figure 8, right). Creating new callouts, or editing existing ones also occurs on the right side, eliminating issues of the callout occluding the main text in the single slate case.

#### 4.3.4. Remote Control From PC

As we discovered during our technology probe trial, it is important for users to have their reading system well integrated with their main content creation tool such as a laptop or a PC. Laptops and PCs provide a keyboard and vertical display that faces the user and are generally preferred for composition tasks [Morris, Brush, and Meyers 2007]. Other tasks for which a PC is superior include things like accessing data from electronic archives. As such, we expect that many documents will be marshaled together on the PC first, before making their way onto the slates. While it is an option to attempt to duplicate PC functionalities on the slates themselves, like a Tablet PC, that would necessarily entail adding a great deal of complexity to the interface. Wilson and Landoni [2003] point out that one desirable quality in a reading device is maintaining its simplicity so that operating the device does not begin to interfere with reading activities. Instead, we thought a more prudent approach is for the slate system to complement, rather than replace, the PC in reading environments. One way we handle the requirements of federating with a PC is through a synchronized distributed clipboard [Miller and Myers 1999], of which the PC is a member. The synchronized clipboard supports information extraction tasks by ensuring that material on a slate is sharable to both slates and PCs.

However, the concurrent use of PCs and slates presents another interactional challenge arising from differences in the devices' input modalities. PCs are primarily operated via keyboard and a pointing device, while our slates are controlled via hardware buttons and a stylus. Tasks that involve repeated switches between PCs and slates are slowed considerably from the user having to re-home on different input devices after every switch. To address this problem, we decided to extend our remote control mechanism so that slates can be controlled directly from a PC. Upon activation, our PC interface presents a visualization of all of the slates in the environment (Figure 9). From this visualization, the user can select a particular slate, and then control the slate using a keyboard and mouse using the same communications infrastructure that powers the remote control feature described in the previous section. The PC remote control function allows a user to have full control of a slate environment (including text entry and text selection), but treat it as if it was simply another window on the computer desktop. For example it is possible to excerpt a piece of text very quickly using this feature (Figure 9,



**Figure 9. PC Remote Control example. Left: PC showing overview of slate devices. Right: Detail view of screen on PC, after selecting a slate to control. Insets: Lassoing text, then selecting the copy command, at which point the text is copied to the distributed clipboard. (Mouse pointers are enlarged for more visibility)**

right).

#### 4.4. Stack Management

The functionality we have presented up to this point has been concerned with reading activities that make use of the small set of documents that are in the active stack of each slate. However, reading activities will almost certainly require documents that are not in the active stack, forcing users to modify, save, or replace the contents of the active stack on a slate. In fact, the availability of robust stack management capabilities has far-reaching effects with regards to aiding document organization, providing the ability to suspend and resume reading activities, and even enhancing the mobility of the system. The importance of stack management cannot be understated, which is why we dedicate a separate subsection to describe how the Stack Manager component of our system helps realize these desirable characteristics.

##### 4.4.1. *Aiding Document Organization*

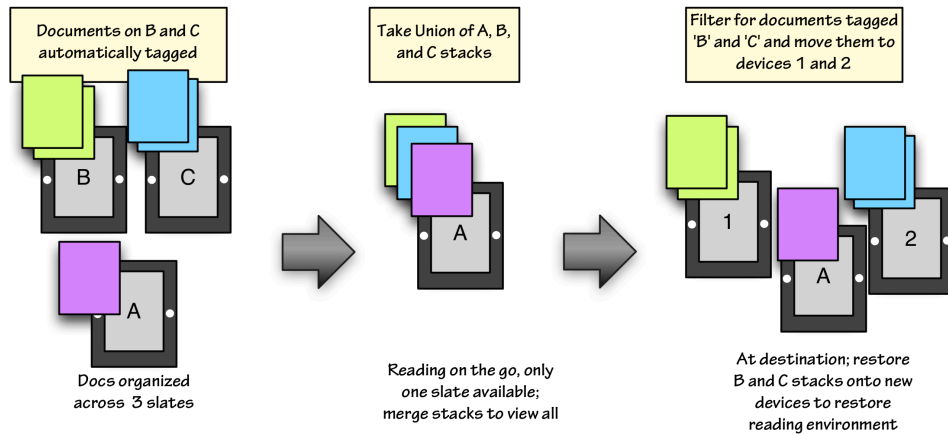
One of the design choices we made in the system is to map several documents to each slate, which we detailed in the first part of this section. Recall that the reason for this decision is our belief that users will probably need to work with more documents than slates available at any given time. Presumably, like paper stacks, the electronic stacks comprise logical organizations of documents so that users can quickly find and access a desired document. However, with a limited number of slates, a slate may need to be repurposed to view a document that may not necessarily be related to the other documents in the stack of that slate. To illustrate, suppose we have two slates, one with a stack of documents about different types cats and one about different types of dogs. At some point, we would like to compare two different dogs, side-by-side, meaning we would have to move a dog document onto the stack of cat documents. After several rounds of this, the stacks on each slate lose their organizational significance.

The Stack Manager addresses the above problem with a tagging mechanism to group together documents in a more permanent fashion. Tagging documents is similar to placing them into folders, except that a document can belong to several tag groups. Moreover, the Stack Manager allows users the ability to quickly specify complex groupings of documents using unions and intersections of tag groups. In a single operation, documents in tag group (or complex grouping) can be added to, or replace, the active stack of a slate. The tagging system allows users to quickly and concisely select documents to put on the active stack of a slate, ensuring that users can rapidly get at the documents they need. As for the above example, it is simple for the user to "clean" the stack of each slate by reloading the contents with documents that have been appropriately tagged.

The reverse process, where a user applies a tag to the documents in a particular stack, is useful as well, especially for sorting tasks. Suppose a user is going through a large set of uncategorized documents and wishes to group related documents together. With the Stack manager, the user can just move documents from the slate with the uncategorized documents, to other slates representing the document categories, which initially have empty stacks. After moving the documents to the appropriate slates, the user can tag the documents in each stack in a single operation. This method of sorting allows the user to leverage spatial layout to sort documents, much like what is possible with paper, and avoids repetition by allowing several documents to be tagged together in batch.

##### 4.4.2. *Suspending and Resuming Reading Activities*

Although we have shown that the Stack Manager is good for organizing documents, its functionality also makes it possible for reading activities to be suspended and resumed at a later time, at a different venue, and with different devices. The fact that the user can restore the environment using different devices increases the mobility of the system since



**Figure 10. Using tag operations to manipulate, save and restore document stacks in order to save and resume reading activities across different environments and times. Although Slate A is always present in all of the venues in the above example, there is no restriction that a common slate be present across different reading environments.**

the user is freed from having to carry the slates he needs around to all of the locations where he reads.

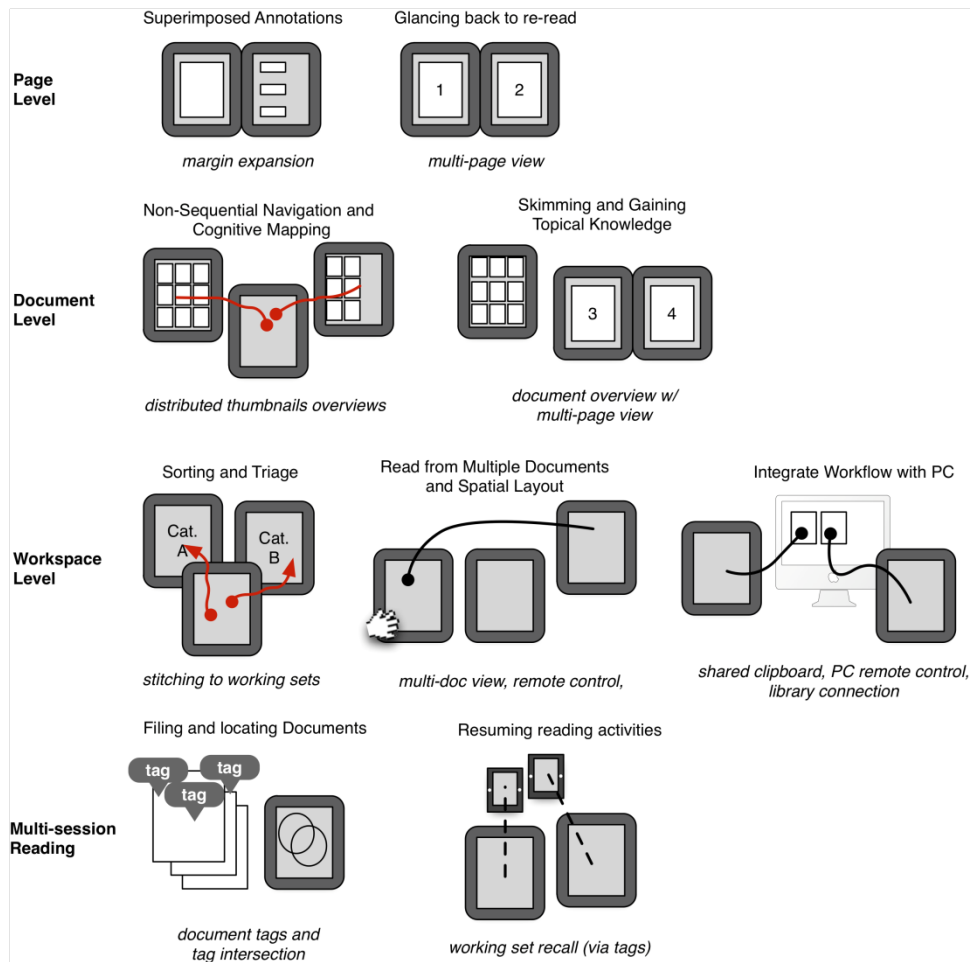
The mechanism that makes this additional feature of the Stack manager possible is the application of automatic, system-generated tags. As documents are placed onto the active stack of a slate, the Stack Manager automatically tags them with a timestamp as well as the name of the device on which it is being viewed. Using these automatic tags, it is possible for users to recall the specific documents used on a device at a particular time, without needing to explicitly tag those documents in advance. If a user wishes to restore a previous reading workspace in a new location he can set up slates in the new venue, filter for the documents he was using previously by time and by slate, and repopulate the slates with those documents. More powerful, however, is the fact the user can restore a reading session using different devices. For instance, if a user is using three slates at work, he can recreate his reading workspace at home using slates he already has at home (Figure 10, rightmost panel). By filtering for the documents he used at work by slate, it is easy to map the contents that were on each work slate to a home slate.

#### 4.4.3. *Dealing With Resource Mismatches*

The Stack Manager further increases mobility by allowing an environment with many slates to gracefully scale down to one with a smaller number of slates. As a result, users are able to continue reading in environments where few slates are available (e.g. no slates there already, user does not wish to transport full complement of slates), or environments where working with many slates is not practical (e.g. public transit). The method to achieve this is simple; the user can take the union of tags corresponding to documents on different slates and redistribute them onto a smaller number of slates (Figure 10, middle panel). While functionality may suffer from the reduced number of displays, this technique ensures that it is still possible to work with documents even if there is a resource mismatch between reading environments.

#### 4.5. Putting Everything Together

In the sections above, we have described many examples of interactions between slates and explain how these interactions can benefit reading activities. In order to give the reader a better idea about how the multi-slate interactions fit within the broad requirements of reading, we place these multi-slate in context with the different reading requirements we established at the beginning of the paper. Figure 11 organizes the



**Figure 11. Multi-slate interactions in the context of the reading requirements presented in Section 2**

features we presented above in a structure matching that of Table 1. As one can see, the beneficial enhancements that the United Slates system extends to the full gamut of reading requirements, from the page level to the multi-session reading level. The functions made possible through multi-slate use, coupled with the special characteristics of the individual slates (e.g. light weight, readability, and support for writing), translate to more complete support of reading requirements than existing electronic reading technologies.

## 5. IMPLEMENTATION

Since we covered the details of our slate hardware in Section 3, we focus now on the software components of the United Slates system, which is composed of the underlying system architecture and the application software. The Gumstix COM we use has a standard ARM-based application processor (TI OMAP3503), for which several standard distributions of Linux are available. As a whole, the system operates like any other Linux system. The main additions made to the operating system were several device drivers that routed data from the microcontroller to user applications and an intermediate layer that presents a virtual Linux frame buffer device to user applications, but modifies the graphics data prior to sending it out to the display controller. Unlike most Linux systems, we opted not to use the X Window system to provide graphics. Instead, to improve

performance, we use a lightweight DirectFB graphics layer to draw to the virtual framebuffer.

### 5.1. System Architecture

In our current system implementation, hardware devices are fully interlinked using a Bluetooth Personal Area Network (PAN). Bluetooth PANs require a master device to route traffic between devices, but this aspect of the network is abstracted away at the OS level. Applications instead simply see a standard IP-based network linking the devices. The reason we adopted this architecture was largely due to the fact it requires no extra hardware, like a wireless access point, which was useful when we performed our technology probe and could not reliably guarantee wireless network access. The one limitation this configuration has is that our master device must always be on. An easy, and transparent modification that would make the slates fully interchangeable would be to use Wi-Fi networks instead.

In our system, each device maintains a copy of each document being viewed. As annotations are applied, only the annotation data needs to be sent to each device. The device on which the annotation is made is responsible for sending out that data to all over the other devices. We use a non-locking opportunistic synchronization scheme (last input in wins) across devices. This scheme works well considering our usage scenario is currently of a single user operating several devices.

In order to implement the stack tagging, saving and restoring features described in our system, a central server storing the state of the various stacks must be available to convey the data to new devices that were not previously present. In the current implementation, we simply placed the server on the Bluetooth master device, because we knew for sure that that particular device would always be present in any configuration. However, in a real-world implementation, the server would be in the cloud and always be accessible. One good way to ensure that devices have ubiquitous network connectivity to each other and to the cloud is through the use of an off-the-shelf mobile Wi-Fi hotspot.

### 5.2. Application Software

The application software on the slates is written in Python and uses the GTK toolkit, along with PyGTK bindings. Inter-device communication is facilitated through Python Remote Objects (PyRO). The tablet federation application on the PC is also written in Python and shares much of the same codebase as the application running on the slates. The application is structured into several view classes, each providing the UI for one level of the system (e.g. page-level, thumbnails, stack manager). A view manager class routes input events to the view that is active. View classes maintain all variables related to their state, and can export the state on demand or temporarily load a new set of state variables. It is through this capability that we implement the remote control features of the system. The system we tested in this paper uses a custom document format consisting of page images organized into directories on the filesystem. Strokes are stored in separate files and overlaid on top of the page images. The system is also able to view PDF documents, in which case, annotations are placed directory into the PDF document. We use the ShortStraw [Wolin, Eoff, and Hammond 2008] algorithm for many of our gesture recognition requirements.

## 6. PRELIMINARY EVALUATION STUDY

As the next step in the iterative development of this system, we decided to carry out an evaluation study in which we asked participant users to describe some of their typical work practices involving active reading, and then to try out some of the features of our system assessing them against the real world tasks they usually carry out. This was not a controlled study aimed at comparing our system against others. Rather, this is an exploratory study in which we aimed to provoke discussion and feedback from potential real users against a grounded understanding of their real work practices. This kind of

study we see as complementary to a controlled assessment and evaluation once the system design is more polished and robust. At this point, we judged it to be more important to get a sense of how our evolving system might fit with real world reading requirements.

For this study, we recruited 12 students from a large University. We did not recruit for specific majors or class level. Our participants consisted of 11 undergraduate students and one graduate student. Participants received \$20 compensation for their participation.

The study was set up as a semi-directed interview in which the participants had the opportunity to perform and compare different kind of interactions on the slates. We started each session with an interview about the reading (e.g. for class, work, or research), which they had undertaken in the previous semester. This was not only to give us an understanding of their practices, but also to get the participants to reflect on the ways in which they really do their work prior to using our system. We asked participants to describe the materials they used, what and why they printed to paper, and the resources they employed when reading and writing, situations where multiple documents were used together, and instances when documents were on different types of media.

### 6.1. Observing Users Interacting With The System

Following the interviews, we explained that the purpose of the study was to assess the efficacy of a set of features that would make use of situations where one might have more than one slate from which to read. First, we outlined the basic capabilities of a single slate and let participants familiarize themselves with controls and inking capabilities of the device. Then, we tested the multi-slate reading support tools we developed in the following order: margin expansion, overview navigation, hyperlink navigation, screen teleport, shared clipboard, PC remote control, and stacks and reading sessions.

The general process we used for testing the tools was to have users perform short tasks that simulated portions of real-world reading activities using each tool. In order to get a better sense of whether our multi-slate interactions provided any benefits, the testing for all but two of the features included an alternative condition where users were asked to perform the same task using a single slate. The single slate condition not only represents the experience of using a contemporary single-screen reading device, but also reflects the user experience of a multi-slate system that did not provide features integrating the slates. The two features where we did not provide an alternative condition were the copy and paste to the PC using the distributed clipboard, and the stack manager because there was no plausible method of performing those tasks efficiently without functionality our system provides. Table 3 provides specific details about the task that we used to test each feature, what we asked users to using the features of the United Slates system, and the procedure for the single slate condition.

Feature	Task	United Slates Procedure	Single Slate Procedure
Margin expansion	Make margin callout annotations on several pages, return and review the annotations	Use two slates that are attached using proximity links so that annotations are created and displayed alongside the main text	Use a single slate to create margin annotations and cycle through annotations one at a time.
Distributed Thumbnail Navigation	Navigate to a different page in Document A (A), navigate to a page in Document B (B), and then navigate back to A	Use a total of three slates, with two slates showing thumbnail overviews of A and B, and one slate for viewing a page in detail. Use the distributed thumbnails to jump to target pages.	Use a single slate to go between A and B and back to A. Users needed to go to the stack view to switch documents, and then use the thumbnail overview in each document to navigate to the target page.
Hyperlink Navigation	Visit a sequence of hyperlinks. The sequence includes backtracking navigation where the next hyperlink to follow is on a previously viewed page.	Perform the navigation using three slates, using a combination of Conduit and Stitching interactions to open links on other devices in the environment.	Use a single slate to visit the sequence of links. Backtracking was performed using a dedicated "Back" button.
Screen Teleport	With a slate in hand, change the page on a different device to a specific target page. Once at target, make ink markings on that page.	Use Screen teleport function to move the display of the distant device to the slate in hand. Perform navigation and inking tasks using slate in hand and send display back to distant device	Reach and pick up the distant slate, perform navigation and inking task and then return the distant slate back to its original location.
Distributed Clipboard	Copy and paste text passages from the slate to a word processing application on the PC	Copy selected text to the distributed clipboard. Paste text on PC using either mouse or keyboard (up to the user).	N/A
Remote Control from PC	Navigate on the two slates alongside the PC so that they are displaying specific target pages.	Use the PC interface to remotely control the slates. Perform navigation without touching the slates.	Navigate by directly operating the two slates.
Stack Manager, Tagging	Place a selection of documents into a newly created stack. Shut off a device, use stack manager to restore that stack.	Use stack management interface and perform tasks using tags.	N/A

**Table 3. Task and procedures we used to test each feature with participants.**

After testing the multi-slate and single slate conditions for each feature, we asked users to compare the single slate method of accomplishing that task and the multi-slate approach. Then we asked participants to describe whether the feature that was just tested would be useful for their personal reading tasks, whether the feature was confusing, the advantages and disadvantages of the feature, and how the feature might be improved to make it better suited for the style of reading they typically performed.



## 6.2. Results

For all of the techniques, users appeared to understand the process of all of the techniques with minimal intervention and explanation on our part. For nearly all of the features, users had no trouble jumping in and performing the interactions we asked them to perform during the experiment. Therefore, it would appear that these techniques would be suitable for further testing as-is in a system outside of the laboratory.

Overall, users were positive about all of the multi-slate functionality we tested with them. The positive feedback is perhaps not surprising given the fact that participants in such studies are generally inclined to be positive about prototype systems in which they see the investigator as having a stake. Therefore, the more interesting takeaways are in the specific comments that participants made with respect to each feature, including their suggestions for improvement.

**Margin expansion** – Participants reported that expanding the space allocated to the margin callouts by physically attaching two devices made creating and viewing annotations easier than the process when using a single screen. Participants mentioned that using this technique meant that callouts no longer occluded the main text, allowing one to read while writing (P4, 7, 9, 10, 11, 12) and that showing all of the notes at once on the second screen made it easy to find pertinent notes (P4, 5, 7, 8, 10, 12). The most requested improvement was to provide facilities to either expand the writing area or to allow writing on the entire of the second device (P6, 7, 8, 9, 11). Participants added that the proximity link system could also be used for other features like automatically searching selected text on a second screen (P7) or automatically displaying documents that the user has associated with a particular page (P2, 8, 12).

**Distributed thumbnail navigation** –Every participant except for P8 mentioned that overview navigation would be very useful for working with multiple documents or locations in tasks such as writing papers, comparing articles, and browsing lecture notes. Most participants (P1, 2, 3, 5, 7, 8, 9, 11, 12) pointed out that they liked the simplicity and speed of using the thumbnails compared to the alternative we had participants try, which involved going to the current stack, switching documents, and then navigating to the desired page. Improvements that participants suggested included showing annotations and margin notes more prominently in the thumbnails (P2, 4), and making the process of calling up a thumbnail on a different slate more like drag-and-drop (i.e. providing visual feedback showing the process of moving the thumbnail from one device to another) (P6). P11 mentioned that he probably would not want to use a separate device for each document overview as that could become difficult to manage. Instead, he said he would prefer to “swipe” or otherwise cycle between SFT overviews on one device, while employing an additional device to view the page in detail.

**Hyperlink navigation** – Participants found several reasons why they would want to open a link on a different device. First, opening a link on a second device made sure both the source and target documents are visible, ensuring the user did not get lost. It also reduced the amount of flipping that needed to happen to work with both documents (P1, 2, 6, 9, 10, 11, 12). Cross-device hyperlinking also helped participants focus on reading material, by allowing them to move content from a peripheral slate onto the “main” device that they were holding (P6, 8, 9, 10, 11). One problem that some participants encountered when we introduced Stitching alongside Conduit was confusion about which technique to use when attempting to navigate to a link on a different device. When asked to open a link, some users immediately proceeded to pick up the target device to use Conduit. In these situations, we reminded the participant to use Stitching. Most users, when asked about this behavior, attributed it to having to adjust their mental model to accommodate the different directions information can flow from one slate to another. They also added

that once they updated their mental model, Stitching made sense as well. A few participants (P11, 12) expressed concerns that spreading documents among too many devices could be confusing. Two participants (P10, 12) suggested that a simple list based approach for selecting a target, rather than interacting with the target device itself, might be preferable.

**Screen teleport** –In the non-teleport condition, where users had to manually operate the remote slate, one thing all participants had in did was to place the slate they were holding down on the table before picking up the other slate, confirming the difficulty of holding on to more than one slate at any given time. However, the utility of using one slate to control another split user opinion. Most participants found that using a nearby device to control a remote device to be preferable than the alternative of reaching over to pick up the remote device (P5, 6, 7, 8, 9, 10, 11, 12). P8 described it as being “more efficient—3 presses of a button did the same as reaching over.” However, a contingent of participants (P1, 2, 3, 4) found the feature to be confusing, mentioning that it would be simpler to operate the devices directly, even if it meant physically shifting devices around. These participants said that the prototypes were light enough to move around without being awkward, and indicated that the experience of using the screen teleport feature to be slightly worse than directly moving the slates. Like with the hyperlinks, participants mentioned it would be helpful to pick from a list (P10, 12) rather than having to stretch to interact with the device to proxy to. P12 specifically mentioned that picking from a list would be useful for situations where she moves from reading at her desk to reading on the bed and wants to access a resource that was collocated in the environment, but out of reach.

**Shared PC clipboard** – All participants thought the shared clipboard feature would be helpful, although P8 believed that working with electronic documents directly on his PC would still be preferable. Not surprisingly several participants (P1, 2, 7, 9, 10) pointed out that it was annoying to switch between the slate and the PC. There were some differences in how our participants employed the clipboard, however. Instead of using pull-down menus on the PC, P5, and P11 used keyboard shortcuts to perform the paste operation on the PC side. Since it was possible to press the shortcut keys while holding the pen, these two users did not report issues with having to switch devices.

**Remote Control from PC** – All participants, with the exception of P12, found controlling slates remotely through the PC to be the more preferable option for operating the slates when we asked them to control the slates while performing a simulated writing task on the PC. P12 found it easier to operate the devices directly. Many participants cited the fact that they did not have to switch devices as an advantage (P3, 4, 5, 6, 8, 11). P4 and P8 remarked that the remote control allowed the slate alongside the computer to behave as an extra, portable monitor. Another advantage participants cited as a plus for the remote control functionality was the fact that they were familiar with using a traditional PC (P1, 2, 3, 9, 10). P7 and P10 mentioned that the PC interface made controlling many devices in rapid succession easier. One unforeseen benefit of the PC remote control feature was that turning pages and switching documents on the PC was *much* faster, owing to the faster processor and display on the computer. Participants had mixed feelings about the lack of an accurate rendering of slate location in our visualization. A majority believed that the visualization actually felt that it provided a more organized view of the slates in the environment, allowing one to more easily see and access the slates (P1, 2, 4, 5, 6, 9). But, several participants mentioned that it was somewhat confusing to identify a target slate because the ordered grid of slates in the PC UI did not match the spatial layout of the slates on the desk.

**Stacks and reading sessions** – The vast majority of participants thought that tagging documents and organizing them into tag-based folders to be a good translation of how they currently organize documents. P9 made the comment that the organizational system made going through the documents easier by making it “feel like checking email.” Some users additionally mentioned that they sort documents by time as well, which the time-based tagging supported. In terms of going beyond what they currently do, a number of participants mentioned that they frequently lose track of documents, so the automatic session tracking feature would be helpful (P3, 6, 7, 9, 12). Finally participants confirmed that the ability to save reading sessions to be resumed later on other devices would increase portability (i.e. not have to carry devices everywhere) and also supports the idea of having a “main/master device” (P5, 6, 10, 11, 12). The participants who were not as enthusiastic about the stacks and sessions (P4, 8) mentioned that the use of the tags resulted in too many grouping options. Instead, these users said that they would just prefer to organize documents using traditional folders.

### 6.3. Evaluation Summary

Participants found the interoperability of the slates with the PC to be the most compelling feature, since the isolation of information on paper documents from tasks on the PC was a problem familiar to all of our users. The other feature of our reading system that participants judged to be directly applicable to their current reading needs is the ability to manage large collections of reading material we provide in our stack manager.

One particularly interesting observation was that while users were split about the utility of screen teleport between slates, they were considerably more positive about the application of the same core functionality to control the devices from the PC. This is an interesting finding on several levels. For one, the fact that many users did not find picking up a slate and moving it around onerous provides a data point about the threshold of how heavy a reading device can be before it becomes burdensome to handle. Our devices, which weigh in at 500g, appear to be under this threshold. The discrepancy between slate and PC remote control is enlightening as well. We believe there are two main reasons for the discrepancy. First, there is more work involved to switch from using a PC to using slates than to switch from slate to slate. Second, many users found that performing navigation tasks on the slates using the PC interface provided a better experience than working on the slates directly: on the PC all the slates can be controlled from a single location, and using a keyboard and mouse was fast and familiar.

One final observation of interest was that we noticed was there was a practical limit in the number of slates that could reasonably fit in the immediate working area, especially when a computer is also present. The main reason for the limit is because users do not overlap slates. Therefore, P11’s suggestion to allow cycling through multiple panes of thumbnails on one device is a particularly useful suggestion.

## 7. DISCUSSION

### 7.1. Implications of Our Research

Tashman and Edwards [2011a] identified two major tensions in active reading technologies. The first of these tensions was that users wanted a large amount of space to work with many documents, but also wanted the reading environment to be portable so that they could set up in arbitrary locations. The second tension was that users liked the reading experience of reading from a dedicated device, given their unique affordances, but required the use of their PC for many of the support activities connected to that reading. We believe that the United Slates system we presented offers a potentially effective solution for resolving both of these active reading tensions by providing a dedicated reading solution that enables users to work with many documents at once, provides the annotation qualities and form factor found in dedicated reading devices, while maintaining users’ ability to fluidly work across PC and reading device.

One of the fundamental motivations underpinning our system is to provide more space to work with documents, in ways that are both flexible and optimized for cross-document use. At the same time, we believe that our system is also a more portable solution than other approaches. At the most basic level, transporting a set of slates is a far more practical proposition than transporting a large multi-touch surface. However, with the ability to save and restore reading sessions, users may not need to transport any slates at all, as they can capitalize on slates that are already present at the venue where they wish to read such as their office or a library. One way to further increase the mobility of our system is for venues to provide slate devices as infrastructural tools, much like a projector. The reason this vision is compatible with our system is because the stack manager blurs the lines between public and private devices. Work performed on a shared public device is not lost once the user returns the public device. Furthermore, we can use physical proximity links as a way of associating public devices with the user temporarily in a fast and unambiguous manner. The physical aspect of this type of authentication may also be useful for streamlining authentication and rights management issues.

How the system integrates reading tasks with other activities that occur on other tools in the reading workspace was another requirement for which we expressly designed our interface to support. We recognized that PCs and reading devices have unique niches owing to the significantly different input and output modalities that each support. As such, we believed it was unwise to force users to conform their reading tasks to the PC, or relegate standard computing tasks to a device that offers a smaller screen, and lacks keyboard and indirect pointing control. This belief led us to implement features that streamlined the use of both slates and PC simultaneously. However, during our evaluation an even more interesting phenomenon emerged: we discovered that the PC actually serves as a good platform for performing meta-tasks in the multi-slate environment like quickly setting the state of many devices. Thus, we believe that connecting the multi-slate reading environment with external devices entails much more than making sure that we can move data between slates and PC. Instead, the proper approach is to consider how slates and PCs can further work together to create better interactional options.

## 7.2. Areas for improvement

Although our system, as presented, attempts to address many of the outstanding problems surrounding reading activities, it is clear that there are many enhancements that could be made to the system to improve its performance.

### 7.2.1. *Fast, Touch Capable Displays*

In constructing our system, we were forced to make a number of tradeoffs about what technologies we used for the system. In the end, we elected to use slower, but lighter, and more readable e-paper displays. Practical considerations for things like battery life, arising from our plans for future deployments with these devices, shaped our decisions as well. An implication of this choice was that not only was our screen refresh rate limited, we could not employ direct touch sensing because touch sensing overlays considerably degraded the screen image. Fortunately, these tradeoffs are not fundamental, as a number of viable solutions are around the corner. For instance, the Mirasol [Qualcomm 2011] display promises a fast, full-color display with energy consumption characteristics of current electronic paper. For situations where writing and weight are not critical, it is also important to point out that many elements of our system could be readily implemented on commercial pad appliances like an iPad.

It is clear that faster screens and touch control would both be valuable for improving the interactions on our reading devices. Faster displays would enable our devices to more closely match PCs and pad appliances in reading activities like skimming, where rapid

presentation of content is beneficial. Furthermore, more interactive displays translate to more visualization possibilities for improving users' cognitive awareness of the state of the document, stack or reading environment.

Touch input would open the door to much richer bimanual operation of the devices, allowing them to more closely approximate what is possible on a tabletop computing environment. Projects like FluidText [Tashman and Edwards 2011b] have demonstrated how the combination of highly interactive displays and multi-touch can be used to support active reading. These techniques would be great additions to the system we present in the paper. Moreover, touch input can further extend the expressiveness of the techniques we present in this paper. For instance, with touch, one can directly specify screen positions at both the source and destination slates of a Conduit operation, allowing the fast mode of transfer Conduit provides to be employed in more types of cross-slate operations.

#### 7.2.2. *Slate Localization and Identification*

For certain interactions like opening thumbnails and remote control, participants in our evaluation mentioned that indirect interaction that did not require physical reaching, like picking a device from a list, would be faster. This preference was further reflected in users' enthusiasm about controlling many devices indirectly from the PC, which could be done quickly, without any reaching required. However, in an environment where slates are similar in appearance (and/or showing similar-looking content), relying on indirect interaction could be confusing since users would not be able to reliably correspond a choice in a list to a slate in the environment. Although providing additional feedback or identification cues through multi-monitor window notification techniques [Hoffmann, Baudisch, and Weld 2008] could help alleviate this problem, a better solution might be to rely on slate localization to create an accurate rendering of the devices in the environment.

Slate localization would be a great boon for our system because we could use indirect input to sidestep the problems of reaching. We omitted this functionality because existing systems require infrastructural support at the room level, which would limit slate mobility. However, one possibility is to have the system only be active in selective environments. A better solution, of course, is to have a localization system that does not require any environmental infrastructure. One possibility would be to use acoustic localization technology like that demonstrated in BeepBeep [Peng et al. 2007] to obtain centimeter-level positioning accuracy or localization techniques developed for sensor networks [Broxton, Lifton, and Paradiso 2006].

#### 7.2.3. *Integration With Other Computing Platforms*

While tabletops are not mobile they can still be a great complement to our system when they are available. For instance, tabletops make working with very large quantities of information tractable. Moving things around, creating and destroying items is completely seamless on the tabletop. Slates, as we have shown in this paper, provide a dynamic display in a highly mobile form factor. The two together would support a wide range of reading situations that include working with very large numbers of documents and multi-user reading with a shared display. Another platform for which one should not discount is paper. Paper is both inexpensive and dispensable. Furthermore, there do not appear to be any technologies on the horizon that match the writing experience on paper. Therefore, digital paper technologies like PapierCraft [Liao et al. 2008] would be a good candidate to add to this integrated, multi-device reading environment.

## 8. FUTURE WORK

We believe that we have only scratched the surface with regards to the functionality of the multi-slate reading system. For one, it will be a good idea to incorporate techniques that have been developed for other pen-based platforms like TabletPCs. One avenue we

believe would be especially fruitful to pursue for reading is the inclusion of active note-taking functionality like that presented in InkSeine into the system. InkSeine's in-situ ink search capability and interface for creating arbitrary page layouts of extracted content and hyperlinks would be extremely useful for supporting cognitive mapping, and for locating content.

In light of our observation that there are many other possibilities for slates and PCs to work together as a unified whole, we also believe that it will be important to push further in that area. For one, our current project has largely focused on workflows that occur mostly on the PC, but require the occasional use of slate resources. Scenarios involving the reverse direction are equally likely. To support these reverse scenarios, applications on the PC should be made slate-aware, so that operations on the slates can flow to the PC with less user intervention. Also, to add a level of generic control, it should be possible to temporarily use the slate as a pointing device (i.e. digitizer tablet) to operate the PC from a distance. Also, there is no reason to believe that multi-slate environments should only be confined to reading activities. There are other types of activities for which a thin, pen-based slate system would also be desirable. One candidate is that of design practice, in which pen input is used extensively. More electronic functionality can make the multi-slate system suitable for sketching, and other graphical design work.

In conjunction with added functionality it will be important to extend our understanding of the implications of multi-slate reading through real world deployments with knowledge workers and students. One thing we would like to determine is which multi-slate functions are most applicable to reading in the real world. Also of interest to us is how multi-slate reading systems alter existing reading practice. Lastly, would like to investigate further is how users split their electronic reading requirements when both slates and PCs are available in the environment.

## 9. CONCLUSION

In this paper, we presented the design of a multi-slate system for supporting the complex reading activities of knowledge workers and students. The design of the reading system is informed by an extensive survey of the reading requirements for these users, which indicated that existing reading technologies each supported only a subset of these requirements. Thus, our system combines positive traits of different reading technologies, which include the portability and physicality of e-book appliances, inking capabilities of TabletPCs, and the spatial layout possible with a tabletop computer. The electronic aspects of the system also simplify the process of working with large quantities of documents, reading in different venues, and interfacing with PCs, which tend to be difficult when reading with paper.

Key contributions of our work are the ideas that make the multi-slate design viable for reading activities. These ideas concern the model of mapping many documents onto a limited number of slates, a range of interactions that support of flow of information between slates, and techniques that support reading activities that occur across different venues and using different slates. From these principles, we develop a range of reading tools to support the reading requirements we identified.

The preliminary evaluation of our prototype demonstrates the potential advantages of this system and provides useful feedback to further iterate on its design. Ultimately, we hope to refine this system even more in terms of its interface design, as well as improve on its technical robustness. At that point, the system can be more systematically evaluated in rich, real world reading situations, or even in controlled laboratory settings. All of this, we believe, helps to open up a new category of reading appliances—ones that we hope build upon the affordances of paper while exploiting the power and flexibility of digital tools.

## 10. ACKNOWLEDGEMENTS

We would like to thank Ken Hinckley for his suggestions for improving this paper. Also, Richard Harper, Richard Banks and Bill Buxton provided a number of helpful ideas during the preliminary discussion about this area of investigation.

## 11. REFERENCES

- ADLER, A., GUJAR, A., HARRISON, B.L., O'HARA, K., AND SELLEN, A. 1998. A diary study of work-related reading: design implications for digital reading devices. In *Proceedings of CHI '98*, 241-248. ACM Press/Addison-Wesley Publishing Co., Los Angeles, California, United States.
- ADLER, M.J. AND VAN DOREN, C. 1972. *How to Read a Book*. Simon and Schuster, New York, NY.
- BEHLER, A. 2009. E-readers in Action.
- BROXTON, M., LIFTON, J., AND PARADISO, J.A. 2006. Localization on the pushpin computing sensor network using spectral graph drawing and mesh relaxation. *SIGMOBILE Mob. Comput. Commun. Rev.* 10(1) (January), 1–12.
- CHEN, N., GUIMBRETIERE, F., DIXON, M., LEWIS, C., AND AGRAWALA, M. 2008. Navigation techniques for dual-display e-book readers. In *Proceedings of CHI 2008*, 1779-1788. ACM, Florence, Italy.
- COCKBURN, A., GUTWIN, C., AND ALEXANDER, J. 2006. Faster document navigation with space-filling thumbnails. In *Proceedings of CHI 2006*, 1-10. ACM, Montréal, Québec, Canada.
- DIXON, M., GUIMBRETIERE, F., AND CHEN, N. 2008. Optimal parameters for efficient crossing-based dialog boxes. In , 1623-1632. ACM, Florence, Italy.
- DROPTBOX. 2011. Dropbox - Simplify your life. <http://www.dropbox.com>
- GRUDIN, J. 2001. Partitioning digital worlds: focal and peripheral awareness in multiple monitor use. In *Proceedings of CHI 2001*, 458-465. ACM, Seattle, Washington, United States.
- HINCKLEY, K., DIXON, M., SARIN, R., GUIMBRETIERE, F., AND BALAKRISHNAN, R. 2009. Codex: A Dual Screen Tablet Computer. In *Proceedings of CHI 2009*. ACM, Boston, Massachusetts, USA.
- HINCKLEY, K., RAMOS, G., GUIMBRETIERE, F., BAUDISCH, P., AND SMITH, M. 2004. Stitching: pen gestures that span multiple displays. In *Proceedings of AVI'04*, 23-31. ACM, Gallipoli, Italy.
- HOFFMANN, R., BAUDISCH, P., AND WELD, D.S. 2008. Evaluating visual cues for window switching on large screens. In *Proceeding of the twenty-sixth annual SIGCHI conference on Human factors in computing systems*, 929–938. CHI '08. ACM, Florence, Italy.
- HOLMAN, D., VERTEGAAL, R., ALTOSAAR, M., TROJE, N., AND JOHNS, D. 2005. Paper windows: interaction techniques for digital paper. In *Proceedings of CHI 2005*, 591-599. ACM, Portland, Oregon, USA.
- LIAO, C., GUIMBRETIERE, F., HINCKLEY, K., AND HOLLAN, J. 2008. Papiercraft: A gesture-based command system for interactive paper. *ACM Trans. Comput.-Hum. Interact.* 14(4), 1-27.
- MACKAY, W. AND PAGANI, D. 1994. Video mosaic: laying out time in a physical space. In *Proceedings of the second ACM international conference on Multimedia*, 165–172. MULTIMEDIA '94. ACM, San Francisco, California, United States.
- MARSHALL, C. 1997. Annotation: from paper books to the digital library. In *Proceedings of DL'97*, 131-140. ACM Press.
- MARSHALL, C. 2005. Reading and Interactivity in the Digital Library: Creating an Experience that Transcends Paper. In *Digital Library Development: The View from Kanazawa*, D. MARCUM AND G. GEORGE, Eds. Libraries Unlimited, Westport, Connecticut.

MARSHALL, C.C., PRICE, M.N., GOLOVCHINSKY, G., AND SCHILIT, B.N. 1999. Introducing a digital library reading appliance into a reading group. In *Proceedings of DL '99*, 77-84. ACM, Berkeley, California, United States.

MARSHALL, C.C., PRICE, M.N., GOLOVCHINSKY, G., AND SCHILIT, B.N. 2001. Designing e-books for legal research. In *Proceedings of JCDL '02*, 41-48. ACM, Roanoke, Virginia, United States.

MARSHALL, C.C. AND RUOTOLO, C. 2002. Reading-in-the-small: a study of reading on small form factor devices. In *Proceedings of JCDL '02*, 56-64. ACM, Portland, Oregon, USA.

MERRILL, D., KALANITHI, J., AND MAES, P. 2007. Siftables: towards sensor network user interfaces. In *Proceedings of the 1st international conference on Tangible and embedded interaction*, 75-78. TEI '07. ACM, New York, NY, USA.

MICROSOFT. 2010. Welcome to Microsoft Surface. <http://www.microsoft.com/surface>

MILLER, R.C. AND MYERS, B.A. 1999. Synchronizing clipboards of multiple computers. In *Proceedings of UIST '99*, 65-66. ACM, Asheville, North Carolina, United States.

MORRIS, M.R., BRUSH, A.J.B., AND MEYERS, B.R. 2007. Reading Revisted: Evaluating the Usability of Digital Display Surfaces for Active Reading Tasks. In *Proceedings of TABLETOP '07*, 79-86.

MYERS, B.A. 2001. Using handhelds and PCs together. *Commun. ACM* 44(11), 34-41.

O'HARA, K., TAYLOR, A.S., NEWMAN, W.M., AND SELLEN, A. 2002. Understanding the materiality of writing from multiple sources. *Int. J. Hum.-Comput. Stud.* 56(3), 269-305.

O'HARA, K. AND SELLEN, A. 1997. A comparison of reading paper and on-line documents. In *Proceedings of CHI '97*, 335-342. ACM, Atlanta, Georgia, United States.

PENG, C., SHEN, G., ZHANG, Y., LI, Y., AND TAN, K. 2007. BeepBeep: a high accuracy acoustic ranging system using COTS mobile devices. In *Proceedings of the 5th international conference on Embedded networked sensor systems*, 1-14. SenSys '07. ACM, Sydney, Australia.

PUGH, A.K. 1978. *Silent reading: an introduction to its study and teaching*. Heinemann Educational, October 2.

QUALCOMM. 2011. mirasol® Displays. <http://www.mirasoldisplays.com/>

REKIMOTO, J. 1997. Pick-and-drop: a direct manipulation technique for multiple computer environments. In *Proceedings of UIST '97*, 31-39. ACM, Banff, Alberta, Canada.

REKIMOTO, J. 2004. SyncTap: synchronous user operation for spontaneous network connection. *Personal Ubiquitous Comput.* 8(2), 126-134.

RICHARDSON, T., STAFFORD-FRASER, Q., WOOD, K.R., AND HOPPER, A. 1998. Virtual network computing. *Internet Computing, IEEE* 2(1), 33-38.

SCHILIT, B.N., GOLOVCHINSKY, G., AND PRICE, M.N. 1998. Beyond paper: supporting active reading with free form digital ink annotations. In *Proceedings of CHI '98*, 249-256. ACM Press/Addison-Wesley Publishing Co., Los Angeles, California, United States.

SELLEN, A.J. AND HARPER, R.H.R. 2003. *The Myth of the Paperless Office*. MIT Press.

TANDLER, P., PRANTE, T., MÜLLER-TOMFELDE, C., STREITZ, N., AND STEINMETZ, R. 2001. Connectables: dynamic coupling of displays for the flexible creation of shared workspaces. In *Proceedings of CHI 2001*, 11-20. ACM, Orlando, Florida.

TASHMAN, C.S. AND EDWARDS, W.K. 2011a. Active reading and its discontents: the situations, problems and ideas of readers. In *Proceedings of CHI '11*, 2927-2936. CHI '11. ACM, Vancouver, BC, Canada.

TASHMAN, C.S. AND EDWARDS, W.K. 2011b. LiquidText: a flexible, multitouch environment to support active reading. In *Proceedings of CHI '11*, 3285-3294. CHI '11. ACM, Vancouver, BC, Canada.

THAYER, A., LEE, C.P., HWANG, L.H., SALES, H., SEN, P., AND DALAL, N. 2011. The imposition and superimposition of digital reading technology: the academic potential of



- e-readers. In *Proceedings of CHI '11*, 2917–2926. CHI '11. ACM, Vancouver, BC, Canada.
- WELLNER, P. 1993. Interacting with paper on the DigitalDesk. *Commun. ACM* 36(7), 87-96.
- WILSON, R. AND LANDONI, M. 2003. Evaluating the usability of portable electronic books. In , 564-568. ACM, Melbourne, Florida.
- WOLIN, A., EOFF, B., AND HAMMOND, T. 2008. ShortStraw: A Simple and Effective Corner Finder for Polylines. In *EUROGRAPHICS Workshop on Sketch-Based Interfaces and Modeling*, 33-40.
- YOUNG, J.R. 2009. 6 Lessons One Campus Learned About E-Textbooks. *The Chronicle of Higher Education*, June 4, sec. Technology.