Shared Text Input for Note Taking on Handheld Devices

Laurent Denoue, Patrick Chiu, Tohru Fuse
FX Palo Alto Laboratory
3400 Hillview Ave., Bldg. 4
Palo Alto, CA 94304 USA
{denoue, chiu, fuse}@pal.xerox.com

ABSTRACT
Shared text input is a technique we implemented into a note taking system for facilitating text entry on small devices. Instead of writing out words on the tedious text entry interfaces found on handheld computers, users can quickly reuse words and phrases already entered by others. Sharing notes during a meeting also increases awareness among note takers. We found that filtering the text to share was appropriate to deal with a variety of design issues such as screen real estate, scalability, privacy, reciprocity, and predictability of text location.

Keywords
Shared text input, note taking, handheld devices

INTRODUCTION
The success of Palm and Pocket PC has made handheld computers nearly ubiquitous and many users are starting to plug them into wireless networks, creating new opportunities for networked handheld applications (e.g. [1]). But text input remains a challenge for novice or casual users. Soft keyboards displayed on a screen for use by a stylus, or alternative text entry methods such as Graffiti are slow or hard to learn.

In this paper, we describe a note taking application that implements *shared text input*. Shared text input facilitates rapid input of text on small networked devices by allowing users to reuse text entered by fellow note takers during a meeting or a presentation. In addition, circulating what others are writing makes note taking more lively and increases awareness, adding to the benefits of sharing notes after a meeting (see [3]). To provide more choices and help users get started, the shared text is supplemented with text extracted from related material like past notes, meeting agendas or talk abstracts (see [2] and [4] for related work).

In a typical scenario, users take notes using wireless handheld devices. Others may also be typing in notes on networked laptops. Text notes are sent to the note-taking server, filtered, and sent back to all users (Figure 1). The client application runs in a Web browser and the shared text is displayed under a text area where users enter their notes (Figure 2). An item of shared text may be a word or a short phrase. A simple tap operation on a text item adds it to the note, avoiding the need to type in all the characters individually.

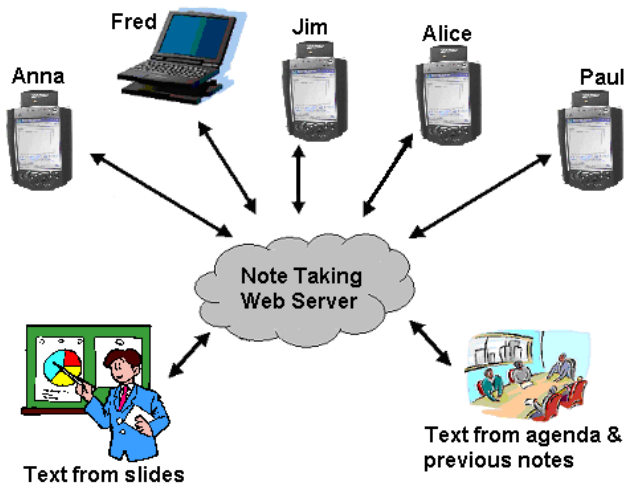

Figure 1. Text is shared in real time between note takers and extracted from related material.

DESIGN ISSUES
Several issues were considered when designing the system, including screen real estate, scalability, privacy, reciprocity, and predictability of text location. Interestingly, all these issues can be dealt with by filtering the text to share.

First, text needs to be filtered because handheld screens are small. The displayed shared text must be limited to only the most recent note items. Longer notes need to be truncated to just the first several words. Text extracted from related material should consist of a short list of the most relevant words and phrases. With a large number of users, scalability becomes an issue and more elaborate techniques like collaborative filtering may be employed (see [5]).

Privacy and reciprocity are closely related in our system. A user can chose different degrees of sharing: 1) do not share text I enter or use, 2) share only text I reuse, 3) share text I reuse and enter, 4) share everything (default option). The degree of sharing then tells the system how much to filter the user's notes.

In order to minimize a user's distraction from the ongoing discussion or presentation, locations for picking up text should be predictable. We chose to separate live text that is

shared during the meeting from text extracted from a corpus of past notes or related material. Each set is identified on the interface between brackets "[]" (see Figure 2), with live text displayed using a bold font. Furthermore, text entered by fellow note takers is displayed in italic to distinguish it from text entered by the user. Other visualization techniques may be applied; e.g. text from different users may be shown in different color.

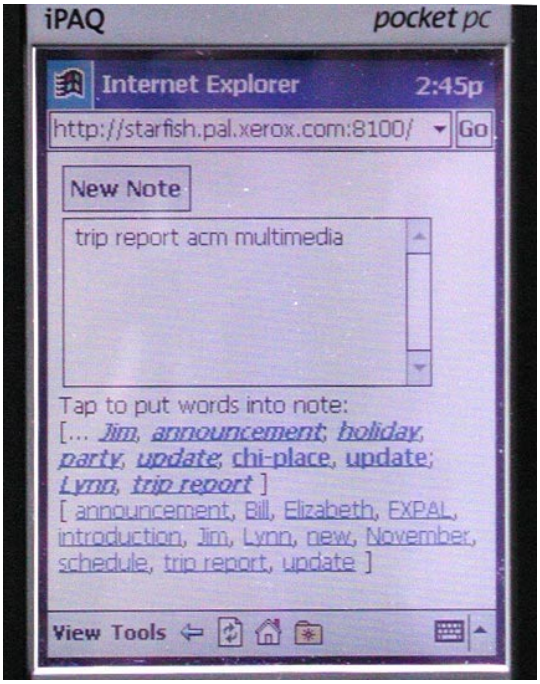

Figure 2. Users can reuse text entered by others with a single tap (in addition to existing text input methods); text is categorized into live text and text extracted from related material

PROTOTYPE TESTING

As part of our iterative design process, we built a prototype and tested it. The note-taking server is a Java servlet and the client runs in Pocket Internet Explorer on wireless iPAQs (see Figure 2). The three authors used this prototype in several staff meetings and seminars for a month. Each user created about ten note items per session. From the notes, the most recent ten words or phrases were displayed for sharing. The text from the related material comprised of the top ten words in document frequency extracted from a corpus of notes from the past year (which were taken on laptops), plus words from the meeting agenda that matched text in the corpus. All users followed the default policy of sharing everything, and after each session the shared notes were posted on the lab's intranet.

We found that all users reused words from the static set of related text, which they sometimes had looked at before the start of a meeting. Users were pleased when they were able to reuse text entered by another person. One user also found it helpful to reuse his own words in the set of live text that came back from his recent notes. Awareness was useful to another user who decided not to enter a note based on the simple fact that someone else had already entered it.

DESIGN IMPROVEMENTS

In the prototype shown in Figure 2, text is shared when the user creates a new note. As a result, other note takers cannot reuse text from a user before this person decides to create a new note. A simple modification is to add a "share note" button but we suspect users will not always remember to share. A better solution is to share the text in real-time as it is entered or reused, and this can be achieved by making the client a Java applet.

We also noticed that all users manually entered words that appeared on the presentation slide, suggesting that text from the current slide should be extracted and shared. A simple approach is to extend the current list of shared text. A more graphical approach is to generate an interactive thumbnail of the current presentation slide from which users can reuse text by taping at the desired location. Even if individual words are not readable on the thumbnail, we expect users to locate them easily because the slide is projected concurrently on a large display in the room.

Finally, we will investigate alternative presentation techniques to display shared text. The current approach with two categories of live and related texts might not scale or prove too slow to locate text items. To address these problems, we are considering the idea of categorizing text by type, including names, events, places and dates.

CONCLUSION

We presented an application of shared text input for note taking. This technique is intended to improve the note taking experience on small devices. The concept of shared text input could benefit other applications running on small devices like Instant Messaging on cell phones. By sharing notes, the system also provides awareness between fellow note takers, which could be even more important in distance learning settings where note takers are distributed.

ACKNOWLEDGMENTS

We thank Bill Schilit and Lynn Wilcox for valuable discussions on this project.

REFERENCES

1. Myers, B. Using handhelds and PCs together. *Commun. ACM 44*, 11, pp. 34-41.
2. Darragh, J., Witten, I., James, M. The Reactive Keyboard: A predictive typing aid. *IEEE Computer* 23, 11, pp. 41-49.
3. Davis, R., *et al*. NotePals: Lightweight note sharing by the group, for the group. *Proceedings of CHI '99*. ACM Press, pp. 338-345.
4. Masui, T. An efficient text input method for pen-based computers. *Proceedings of CHI '98*. ACM Press, pp. 328-335.
5. Resnick, P., *et al*. GroupLens: An open architecture for collaborative filtering of Netnews. *Proceedings of CSCW'94*. ACM, New York, pp. 175-186.