Recent Shortcuts: Using Recent Interactions to Support Shared Activities

John C. Tang[1], James Lin[1], Jeffrey S. Pierce[1], Steve Whittaker[2], Clemens Drews[1]

[1]IBM Research
650 Harry Road
San Jose, CA 95120 USA
{john.tang, jameslin, jspierce, cdrews}@us.ibm.com

[2]The University of Sheffield
211 Portobello Street
Sheffield, S1 4DP, UK
s.whittaker@shef.ac.uk

ABSTRACT

We present an empirical study of teams that revealed the amount of extraneous individual work needed to enable collaboration: finding references to other people, finding files to attach to email, managing incoming email attachments, managing the variety of files used in shared activities, and tracking what work is owed to others. Much of this work involves finding *recently accessed* objects that are needed again in the user's current task focus. These observations led to the design of *Recent Shortcuts*, a tool to help support coordination by making recently used objects easily accessible. Recent Shortcuts enables quick access to people (including groups of people), received attachments, files, and file folders that the user interacted with recently for re-use in the user's current context. Recent Shortcuts makes it easy to use these objects across applications with no additional user input and minimal changes to the user's applications or work practice. Early user experiences with a working prototype led to an extension that integrates recently accessed objects across multiple devices.

Author Keywords

Coordination, recent context, recent documents, Recent Shortcuts, empirical study, multi-device interaction.

ACM Classification Keywords

H.5.3 [Group and Organization Interfaces] Computer-supported cooperative work, H.5.2 [User interfaces] User-centered design.

INTRODUCTION

Recent research has drawn attention to the problem of information fragmentation and how much of our work with computers involves switching among, and integrating across, multiple applications and files [2, 4, 6, 11]. Current computer systems, however, do not support managing activities that involve applications and resources scattered across a computer (or computers). This lack of support has given rise to research exploring how to organize users' interactions according to the *activities* and *tasks* that they want to accomplish, rather than the traditional organization around applications and files [2, 3, 7, 13, 15, 16].

While much of the prior research on activities has focused on individual work (cf. ABC [2], TaskTracer [7], UMEA [13]), we take a collaborative perspective by explicitly focusing on communication and coordination for *shared* activities. We conducted an empirical study of collaborative work practice to understand better how people currently accomplish their activities through the computer. We were particularly interested in the collections of resources and tools that people use, the problems they encounter as they integrate across applications and files, and other challenges introduced by coordinating their activities with others.

We first describe our empirical study and present observations of how people currently accomplish shared activities. Our collaborative focus revealed the amount of individual work needed to find the information used to collaborate with others. We identified opportunities for helping collaborators find that information through objects that they recently accessed, which led to the design of the *Recent Shortcuts* working prototype. Recent Shortcuts makes those objects easily available for re-use, without added user effort or drastic changes in work practice. Initial user feedback on this prototype led to extending Recent Shortcuts to support user interactions across multiple devices and identifying directions for future research.

COMPUTER SCREEN RECORDING

Our empirical study methodology followed in the tradition of video-based observation [17]. Since prior research highlighted the fine-grained nature of activities [6, 11], we used an approach that captured a rich, high-fidelity record of users' interactions with and through their computers. We also wanted to observe users' naturally-occurring computer interactions in the field, with a minimum of disruption to their work. We used a novel approach employing computer screen recording software, rather than video cameras, to capture users' everyday interactions with their computers.

We installed Camtasia Studio screen recording software [5] on users' laptop computers to allow us to unobtrusively capture video records of interactions wherever users

CHI 2007, April 28–May 3, 2007, San Jose, California, USA.
Copyright 2007 ACM 978-1-59593-593-9/07/0004...$5.00.

brought their laptops. Camtasia captured a video of each user's computer screen at 5 frames per second and recorded the surrounding audio through a microphone (embedded or attached to the computer). While screen recording only captures the user actions visible on the computer screen, this is consistent with our interest in supporting users' interactions with and through their computers. Screen recording provided a rich record of users' everyday interactions with their computers for our analysis.

Capturing such a detailed record of users' work (and especially their interactions with others) naturally raised privacy concerns. We mitigated these concerns by restricting access to the videos to the research team and abiding by agreed-upon guidelines on the use of the video data. A key factor was developing a sense of trust with the participants on handling the video data. Tang et el. [18] discuss the advantages, limitations, and privacy concerns of this methodological approach.

We conducted three rounds of data collection where we observed teams of knowledge workers:

- a pair of colleagues at our company, planning a presentation with a deadline the following week
- three members of a core team of over a dozen people at our company, planning for a community service science camp for the next week
- three team members at a university, designing a research experiment for the following month

In each round, we concurrently collected data from team members to observe how they coordinated their work around activities that they share. Their approaching deadlines prompted timely interactions among them, thus increasing the likelihood of capturing coordinating activities. We expected that this collaborative perspective on activities would identify issues not seen in individual activity work.

Each round of data collection with the participants lasted approximately a week, consisting of:

- installing the screen recording software onto their computers and interviewing them about their work
- letting them control recording their computer activity for the next few days
- reviewing selected clips from the recorded data with the participants to elicit their perspectives on it

Over the three rounds involving eight participants (seven females, one male), we collected 67½ hours of video recordings. We qualitatively analyzed the data by indexing the video into episodes relating to activities and collecting examples of specific issues for comparative analysis. In particular, we identified instances where users experienced obstacles in coordinating with others on shared activities. Our interviews with the participants identified additional frustrations they faced in accomplishing their work. We also noted recurring issues that we observed across the different participants.

THE INDIVIDUAL'S WORK IN SHARING WORK

The observations that emerged from our analysis highlighted the amount of extraneous individual effort currently involved in sharing work with other people. By sharing work, we mean exchanging information through email, instant messaging (IM), and shared access to data, as people coordinate working together on shared activities. Our empirical observations identified the amount of individual work needed to enable collaboration, such as:

- finding contact information for people
- finding files to attach to email
- managing attachments that they receive in email
- managing the multiple resources involved in an activity
- tracking what work they owe to other people

We discuss these observations and their design implications for tools to better support coordinating shared activities.

Finding References to People

One repeated observation that emerged in our study was the effort required by users to find contact information for other people in order to communicate and share information with them. Users expended much effort finding the exact reference to people needed to send them email, include them in a calendar appointment invitation, add them to their IM contact list, phone them, etc.

For example, Beth[1] wanted to add a person to a calendar meeting invitation, but could not quite get the right spelling for the calendar system to recognize the person's name. Beth had only recently begun working with the person, who has an Asian name requiring a hyphen in just the right place. After trying several guesses, Beth finally went to look for a prior meeting invitation that involved the person to get the correct spelling of her name. This problem delayed Beth from completing her meeting scheduling task for over three minutes.

In our data, we logged 39 examples of users actively searching for references to people in their previous computer interactions. Users typically found those references by looking back through sent or received email messages or calendar meeting appointments. In most cases, users had recently interacted with the people they were looking for, and had a good idea of where to look. But in other cases, finding references to people meant conducting a search through prior email or the Web to find the right email address or other contact information. Sometimes an online corporate directory was used to look up contact and

[1] All of the names in this paper are pseudonyms.

organizational information about a person, or users even asked another person for the contact information.

Most participants' email clients did have an auto-completion feature that attempted to complete an email address once enough of the name was typed in, but this feature was often unsuccessful. It was especially a problem for those participants who worked at our company, a large enterprise with many highly similar names. Auto-completion also did not work well on names that were hard to recall, especially those with ethnic spellings. Furthermore, users often needed to refer to people across multiple applications, typically requiring a different contact reference, such as writing email to someone with whom they just had an IM chat.

In some cases, users wanted information for a *set* of people, rather than just a single person, and so they searched for a prior email or calendar appointment containing the desired set. These cases represent recurring interactions with a set of people for which users had not yet defined an explicit group alias (e.g., an email distribution list). These ad hoc groups accounted for 18% (7 out of 39) of the searches for people that we observed. When re-using these ad hoc groups, users sometimes manually modified them to repurpose the original list by deleting a name or adding other names to it. Thus, in some cases, the ad hoc group was not the complete set of all the people they wanted to interact with but a useful starting point.

On average, searches for people's contact information occurred every 104 minutes (39 instances spread across 67½ person-hours of video). However, the amount of time spent on these searches and the perceived distraction from the task at hand stood out, even more than the frequency of occurrence. This extraneous and distracting work highlights the opportunity to streamline this process.

One design implication of our observations is that computers could offer a *list of people*, containing both individuals and ad hoc groups with whom users recently interacted, that is easily accessible from anywhere on the computer desktop. This list could be similar to the mechanisms modern computers offer for easily accessing recent files and applications, such as the taskbar and "My Recent Documents" mechanisms in Windows. Indeed, the work involved in finding references to people was in sharp contrast with the relative ease and skill users displayed in finding recently used files.

The list should contain references to people extracted from email, calendar, and IM interactions, as well as potentially more specialized collaborative applications (e.g., shared databases, application sharing sessions). Users need desktop-wide, easy access to such references so that they can use them in their current context (e.g., copying a list of addresses from email into a calendar invitation).

Finding Files to Attach to Email

Exchanging documents, spreadsheets, URLs, presentations, and other objects is also an integral part of collaborating around shared activities. This exchange is commonly accomplished by sending attachments via email [8]. We observed many examples of a particular pattern of activity that immediately preceded attaching a file to an email message, illustrated in the following example.

Paula got an email from a university administrator asking her to review and update an attached course description for a class that she is teaching. She opened the attachment in Microsoft Word, edited it to reflect this year's updates, and saved it, navigating to the specific folder in her file hierarchy where she wanted it located. Having just saved the file in Word, Paula then switched back to her email client (Eudora) to send the updated description back to the administrator. However, when she brought up the file chooser within Eudora to attach the file, it was aimed at an apparently random folder (the one most recently used to attach a file in Eudora). It took her an additional 12 seconds to navigate to the right folder to select the file that she had *just saved* in Word.

Out of 27 observed instances of attaching a file to an email message, the participants had *just worked* on the file in another application 78% of the time (21/27). In most cases, they had just saved the file. But finding the file again to attach it in their email application was laborious and distracting, even though the user immediately switched from saving the file to attaching it. To the user, this sequence should be part of a seamless activity of creating some content and sharing it with others. Yet to the computer, switching from a productivity application to an email client loses any context that the file just saved is likely to be the desired attachment file. While a Windows shortcut to the saved document could be found in My Recent Documents in Windows XP, that shortcut does not help add the corresponding document as an email attachment. Windows allows copying the shortcut, but pasting a shortcut into an email message does not result in attaching the file to it (since Windows shortcuts do not work outside the local desktop).

These observations suggest another opportunity to use *recent temporal context*. When a user needs to access an object in one application, especially a communication application like email, that object is likely to be the file that was just saved or the web page that was just viewed in another application. Providing a list of recently used objects could be a useful mechanism to share information across applications. Such a mechanism essentially extends the "copy and paste" clipboard metaphor to a larger grain size (a file) and a longer history (list of recently used files). While the inconvenience of attaching files in email may not occur frequently or consume large amounts of time, some participants strongly resonated with the annoyance of having to re-find files to attach during our interviews.

Managing Incoming Attachments

The corollary to the work of sending attachments is the work in managing attachments and other pointers *received* via email. Ducheneaut & Bellotti [8] drew attention to the pervasiveness of using email to transmit files via attachments. This practice presents the recipient with the additional work of managing these incoming attachments. We analyzed our data to understand what users do with email attachments that they receive.

Across all three rounds of our study, we observed 39 instances of receiving attachments in email. In 18 cases, recipients immediately opened and viewed attachments, and in three additional cases they opened them shortly after reading the email message (with a few intervening actions to reply to the email or switch applications). In seven cases, recipients subsequently saved the attachment locally after viewing it, while in six instances recipients saved an attachment without even opening it.

In total, recipients saved 13 out of 39 attachments (33%). Of those thirteen, recipients saved four to the desktop rather than to a specific folder in the file hierarchy. Saving to the desktop affords easily finding those files in the near future. Not included in these numbers is a strategy that at least one user had of leaving attachments in her email and using email search as a way of retrieving those attachments (using it much like a local file system).

This need to manage incoming attachments in a way that affords easy future access is consistent with what Bellotti et al. [3] found in their study of email. Their observations led to the Taskmaster prototype, which collected attachments into task threads to enable managing them within an email-based client. Yet, perhaps one reason users in our study saved a third of their received attachments in their file system was their need to use the files again in a context *outside* of email. While we were unable to trace how all of the saved attachments were later used, we saw several examples of users opening these saved attachments outside of email in conjunction with other files that were part of the same activity.

These observations suggest that collecting and providing easy access to *recently received attachments* could be a useful extension to more general recent file mechanisms. The six cases of saving an attachment without even opening it suggest including received attachments for easy future access even if users do not immediately open them. These observations may also apply to other pointers sent through email, such as links to web pages or database entries.

Managing Multiple Resources for Sharing

As previously mentioned, users' activities usually cannot be accomplished within a single application, but involve integrating across applications and resources on a computer [11, 13]. This integration is even more visible when coordinating with others on shared activities. Collaborating often involves email, IM, or other computer-mediated communication tools along with the tools or productivity applications used to create the activity's content.

For example, Paula sent her colleague Racquel an email message that included a description of her plan for revising a paper. The message also included two attachments: an outline of the revised paper structure and the current version of the paper itself, which contained comments and annotations about suggested changes to the paper. Subsequent messages included updated versions of the outline and paper, which Racquel had to organize to keep the collection and sequence of information together. In another example from a different team, Janelle and Mei-li used IM and email to coordinate their joint work on a slide presentation. Each colleague kept their own versions of the file in parallel, requiring one of them to ultimately merge and integrate them together.

Preparing for meetings, especially those that are distributed, also exemplifies the diverse kinds of information that need to be managed. Meetings are often announced by email, usually including an associated calendar appointment. They also often have an agenda, perhaps included as text in the email or as an attachment. Distributed meetings often require a conference call-in number and passcode. Slides or other materials that will be referred to in the meeting are also often circulated, either as attachments in email or pointers to a web page, database, or other shared resource. At the meeting time, all of these materials need to be marshaled together to fully participate in the meeting. Currently, these materials may be scattered across email, calendar, file folders, and other applications and locations in the computer desktop, and the user must manage retrieving them in time for the meeting to start.

While these observations do not strictly revolve around recency of use, they suggest an opportunity of combining temporal clusters of usage along with the people involved in those interactions. For example, a cluster of user interactions (sending/receiving attachments) with a specific set of people (collaborators or meeting participants in the examples above) may help identify relationships among objects that allow inferring a user's activity. Using both temporal and social patterns to help identify user activities is a promising area for further research.

Tracking What Work is Owed to Others

Our empirical data included a few striking examples that illustrate the often subliminal effort needed to track when collaborators owe work to others. People use explicit representations of owed work in some situations (e.g., workflow, to-do lists), and other research found that people use email to implicitly help track the commitments that they owe others [3, 19, 20]. While people often track what work they owe to others mentally, the subliminal effort involved is often revealed when encountering a person to whom some work is owed, which serves to trigger some explanation of the status of the owed commitment.

We saw several examples of encountering a person about an unrelated topic that triggered an explanation about some owed work. In one case, during an IM conversation about one topic, a collaborator wrote "I owe you a response to another email you sent". In another case, receiving email about an unrelated topic prompted a reply to that message, and was immediately followed by a separate email with a plan about when and how she planned to work on a joint paper revision. In two cases, just seeing the person's name in a list on the computer apparently triggered an email or IM to explain the status of owed work.

Computers provide few explicit mechanisms for indicating an expectation of owed work in everyday collaboration. Instead, people tend to socially and contextually negotiate a sense of owing work. One subtle indicator that we observed was differentiating between addressing email using the To and cc fields. We saw several examples of users carefully crafting who to include in each field (often involving cutting and pasting between the fields as the user thought this through). It seems that those included in the To field have a higher responsibility of following through on the message, whereas those in the cc field are often included to be aware of the message without having to act on it.

Scoping recent interactions according to their social context might reduce the effort of tracking owed work. For example, when Mary receives an IM from Joe, it would be useful for Mary to have easy access to all her recent interactions with Joe, including email, attached files, and potentially even upcoming meetings. This would enable Mary to quickly access the information that could help her provide an account for work that she owes to Joe, rather than relying on recall from memory. In this sense, rather than strictly organizing objects according to activities, it is an advantage that recent interactions may cut *across* activities, in that correspondence on one shared activity may trigger an explanation about a different shared activity.

The Value of Recency in Supporting Activities

A common theme that emerged from this set of empirical study observations is the value of providing easy access to *recent interactions* in supporting the current task. Our observations suggest conceptually extending current recency mechanisms in at least three ways:

- Include people, groups of people, and other objects that are often used to coordinate with collaborators
- Extend lists of objects recently used within applications to be available *across* applications
- Extend the copy and paste clipboard buffer concept to include a *list* of recently used files for easy transfer across applications

While modern computer desktops offer mechanisms for easily opening recently used files and applications (such as Recent Items in Mac OS X), our research demonstrates that more context beyond files and applications would be useful. We highlight the importance of including communication objects, such as email addressees, IM screen names, and received attachments, that we observed users searching for in their work but which do not appear in Recent Items.

Furthermore, while many applications offer recency or history mechanisms *within* the application (e.g., URLs in web browsers, email addresses in email clients) we saw the need for access to recent objects *across* applications. For example, working on a file in a productivity application and moving to email to attach that file to share with others crosses application boundaries. Beyond objects recently used within an application, users also need desktop-wide access to objects to support use across applications.

Echoing the popularity of copy and paste for easily transferring snippets between files, our research also identified opportunities for easily transferring files between applications. This is especially visible when attaching files in email, where a list of recently used files may often contain the file users want to attach and share with others.

DESIGNING *RECENT SHORTCUTS*

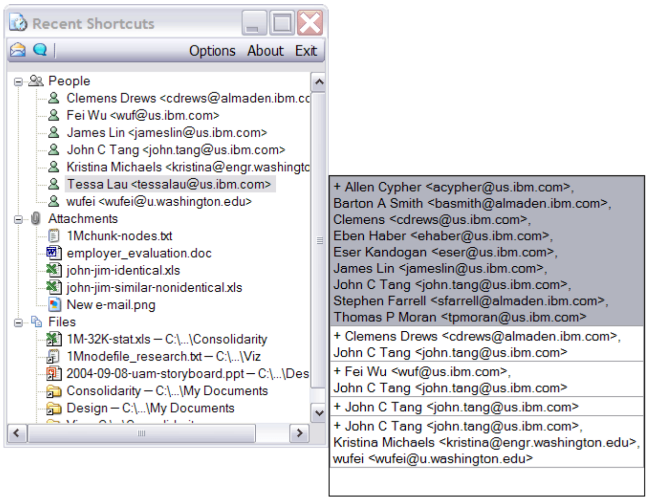
We drew on our empirical observations to design a desktop-wide tool, which we call *Recent Shortcuts*, to provide users easy access to a variety of recent context. We focused in particular on improving access to recent communication objects (e.g., email addresses, received attachments) that participants used in the course of their shared activities. We present our current design along with initial experiences in using the prototype and ideas for future development.

Initial Working Prototype

The Recent Shortcuts prototype tracks and presents three types of recently used objects:

- People — people recently interacted with via email, IM, and calendar and any associated groupings of people
- Attachments — files attached to recently viewed incoming email messages
- Files — recently used files and their containing folders in the file directory hierarchy

These were the types of objects we saw people frequently searching for in our empirical study. Recent Shortcuts tracks the use of these objects and presents the last *n* of them in a list, where *n* is a user preference that can be set for each object type. From the list, users can double-click an item to perform a default action (e.g., open), or they can drag and drop to copy an object into their current activity.



People
- Clemens Drews <cdrews@almaden.ibm.cc
- Fei Wu <wuf@us.ibm.com>
- James Lin <jameslin@us.ibm.com>
- John C Tang <john.tang@us.ibm.com>
- Kristina Michaels <kristina@engr.washingtc
- Tessa Lau <tessalau@us.ibm.com>
- wufei <wufei@u.washington.edu>

Attachments
- 1Mchunk-nodes.txt
- employer_evaluation.doc
- john-jim-identical.xls
- john-jim-similar-nonidentical.xls
- New e-mail.png

Files
- 1M-32K-stat.xls – C:\...\Consolidarity
- 1Mnodefile_research.txt – C:\...\Viz
- 2004-09-08-uam-storyboard.ppt – C:\...\Des
- Consolidarity – C:\...\My Documents
- Design – C:\...\My Documents

+ Allen Cypher <acypher@us.ibm.com>, Barton A Smith <basmith@almaden.ibm.com>, Clemens <cdrews@us.ibm.com>, Eben Haber <ehaber@us.ibm.com>, Eser Kandogan <eser@us.ibm.com>, James Lin <jameslin@us.ibm.com>, John C Tang <john.tang@us.ibm.com>, Stephen Farrell <sfarrell@almaden.ibm.com>, Thomas P Moran <tpmoran@us.ibm.com>

+ Clemens Drews <cdrews@almaden.ibm.com>, John C Tang <john.tang@us.ibm.com>

+ Fei Wu <wuf@us.ibm.com>, John C Tang <john.tang@us.ibm.com>

+ John C Tang <john.tang@us.ibm.com>

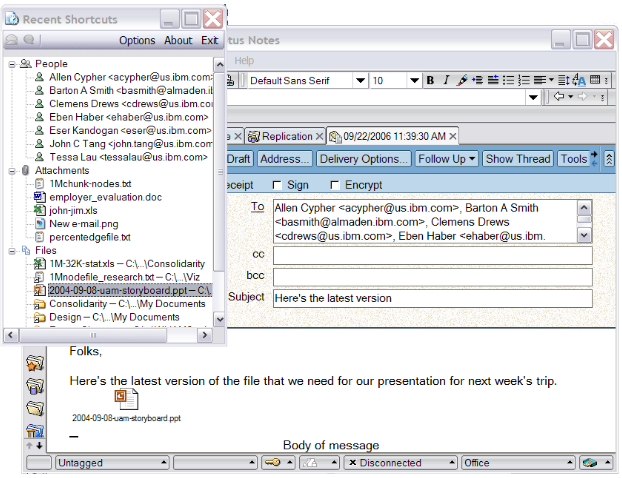
+ John C Tang <john.tang@us.ibm.com>, Kristina Michaels <kristina@engr.washington.edu>, wufei <wufei@u.washington.edu>

Figure 1. Recent Shortcuts window showing people and groups associated with the selected person, attachments received via email, and files and their containing folders.

The *People* section lists recent people (the default is the last 10) with whom the user has interacted via email (including anyone in the From, To, and cc fields) or IM. Selecting any person expands a sub-window that shows the groupings of people that appear together with that person on an email message. Figure 1 shows Recent Shortcuts with one user selected to show the groups associated with that person. By default, Recent Shortcuts shows the last 5 groups for every person. Double-clicking on a person or group will bring up an email compose window addressed to that person or group. Drag and drop will add the email address(es) to applications that recognize that object type (e.g., Mozilla Thunderbird). Alternatively, selecting a person and clicking the IM button (the blue chat balloon icon on the toolbar) will initiate an IM conversation with that person. Selecting a group and pressing the IM button initiates an invitation for a group chat. Integrating with both email and IM facilitates migrating a conversation between IM and email applications. The People section in Recent Shortcuts makes it easier to communicate and share information with others.

The *Attachments* section shows files received as attachments in email messages that the user has recently viewed. By default, Recent Shortcuts shows the last 5 files. Users need not have opened the attachment for it to appear in Recent Shortcuts; they just need to have viewed the email to which it is attached. Users can open these file attachments by double-clicking on them or copy them via drag and drop (e.g., they can drag an attachment directly into a file directory folder). Recent Shortcuts simplifies users' access to incoming attachments by eliminating the need to locate their containing message(s) in an email client or use a file chooser within email to save them.

Similarly, the *Files* section shows the 10 (by default) most recently used files. We draw this list from the same source that Windows uses for its "My Recent Documents" feature. We also display the folder in the file hierarchy that contains each file, as that folder is often a quick way to get to files *related* to those that have been recently used. Users can open files and folders by double-clicking on them and copy files by drag and drop. Recent Shortcuts affords a quick way of accessing a file that was just saved to attach it to an email (e.g., by dragging it from the list into an email message), making it easier to share information with others. This support for drag and drop goes beyond the functionality provided by the Windows "My Recent Documents" mechanism.



People
- Allen Cypher <acypher@us.ibm.com
- Barton A Smith <basmith@almaden.il
- Clemens Drews <cdrews@us.ibm.coi
- Eben Haber <ehaber@us.ibm.com>
- Eser Kandogan <eser@us.ibm.com>
- John C Tang <john.tang@us.ibm.com
- Tessa Lau <tessalau@us.ibm.com>

Attachments
- 1Mchunk-nodes.txt
- employer_evaluation.doc
- john-jim.xls
- New e-mail.png
- percentedgefile.txt

Files
- 1M-32K-stat.xls – C:\...\Consolidarity
- 1Mnodefile_research.txt – C:\...\Viz
- 2004-09-08-uam-storyboard.ppt – C:\.
- Consolidarity – C:\...\My Documents
- Design – C:\...\My Documents

To: Allen Cypher <acypher@us.ibm.com>, Barton A Smith <basmith@almaden.ibm.com>, Clemens Drews <cdrews@us.ibm.com>, Eben Haber <ehaber@us.ibm.

cc:

bcc:

Subject: Here's the latest version

Folks,

Here's the latest version of the file that we need for our presentation for next week's trip.

2004-09-08-uam-storyboard.ppt

–

Figure 2. Using Recent Shortcuts to address an email compose window to a group and add an attachment.

Figure 2 shows an email compose window that a user addressed by double-clicking on a group of people in Recent Shortcuts. Furthermore, the user dragged and dropped a file from Recent Shortcuts into the message to add it as an attachment.

Users can access Recent Shortcuts through a global keyboard shortcut (settable by the user), so that it is easily available from anywhere on the computer desktop. Or, users can leave Recent Shortcuts open all the time, somewhat like an IM buddy list. Recent Shortcuts dynamically updates its lists as the user interacts with files, email messages, and IM chats.

In summary, Recent Shortcuts takes advantage of a user's recent history of computer interactions to support their current activity. It creates lists of recent objects without requiring any additional user effort. This design is based on the *temporal grouping* of activities that we observed in our empirical study where users often re-used objects with which they had recently interacted in their current activity.

Implementation

Figure 3 shows an architecture diagram of the initial Recent Shortcuts prototype. It integrates with the Lotus Notes and Mozilla Thunderbird email clients to collect the email addresses of recent correspondents and incoming email attachments, and with the Lotus Sametime IM client to collect recent IM correspondents. It also looks up people in the corporate directory of our company to associate email

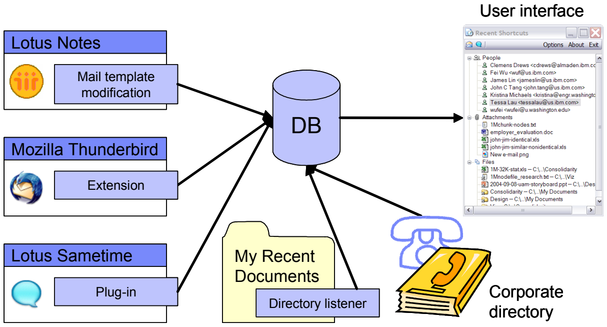


Figure 3. Architecture diagram showing the integration of information from email, IM, Windows, and corporate directory.

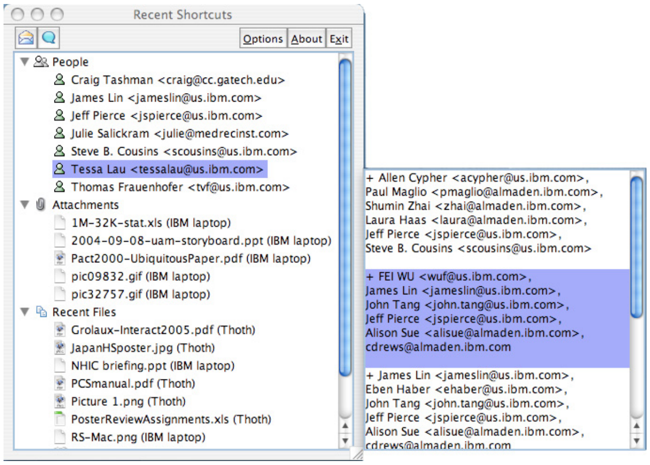
addresses and IM names with people where necessary. It collects information about recently used files and folders from the Microsoft Windows "My Recent Documents" mechanism. It stores all of this information in a database and presents it through a dedicated user interface.

The initial prototype combines a C# Microsoft Windows program for the user interface with multiple application extensions. We currently employ JavaScript to interface with Thunderbird, Java to interface with Sametime, LotusScript (a dialect of BASIC) to interface with Notes, and Visual Basic 2005 to interface with the corporate directory.

Besides using Recent Shortcuts ourselves over the past seven months, we have been able to deploy it to nine users from three different sites of our company who were not directly involved in developing Recent Shortcuts. While we had intended to extensively study usage of the prototype, our initial implementation had performance problems due to the way several users configured their computers. Nonetheless, we did gather preliminary usage experience and feedback. In the next sections we discuss an extension we made to Recent Shortcuts and identify additional areas for future work based on the lessons learned.

Extending to Multiple Devices

While our empirical study revealed the utility of Recent Shortcuts for individuals working with a single computer, our preliminary usage experience suggested that it would also be useful across multiple devices (e.g., desktops, laptops, PDAs, mobile phones). For example, a user might need access on a home computer to files recently worked on or email attachments received on an office computer. Also, references to people with whom users had recently exchanged email on a laptop might be handy to have on a cell phone (e.g., to call to negotiate a last-minute change for an appointment that was arranged via email). Sharing access to recently used files across devices can obviate the need for anticipating and manually transferring files among them. These issues did not arise in our empirical study, in part because some of the participants only used one computer (a laptop that they took with them between home and office) and in part because we only set up one computer for screen recording (to simplify our data collection).



People
Craig Tashman <craig@cc.gatech.edu>
James Lin <jameslin@us.ibm.com>
Jeff Pierce <jspierce@us.ibm.com>
Julie Salickram <julie@medrecinst.com>
Steve B. Cousins <scousins@us.ibm.com>
Tessa Lau <tessalau@us.ibm.com>
Thomas Frauenhofer <tvf@us.ibm.com>
Attachments
1M-32K-stat.xls (IBM laptop)
2004-09-08-uam-storyboard.ppt (IBM laptop)
Pact2000-UbiquitousPaper.pdf (IBM laptop)
pic09832.gif (IBM laptop)
pic32757.gif (IBM laptop)
Recent Files
Grolaux-Interact2005.pdf (Thoth)
JapanHSposter.jpg (Thoth)
NHIC briefing.ppt (IBM laptop)
PCSmanual.pdf (Thoth)
Picture 1.png (Thoth)
PosterReviewAssignments.xls (Thoth)
RS-Mac.png (IBM laptop)

+ Allen Cypher <acypher@us.ibm.com>,
Paul Maglio <pmaglio@almaden.ibm.com>,
Shumin Zhai <zhai@almaden.ibm.com>,
Laura Haas <laura@almaden.ibm.com>,
Jeff Pierce <jspierce@us.ibm.com>,
Steve B. Cousins <scousins@us.ibm.com>

+ FEI WU <wuf@us.ibm.com>,
James Lin <jameslin@us.ibm.com>,
John Tang <john.tang@us.ibm.com>,
Jeff Pierce <jspierce@us.ibm.com>,
Alison Sue <alisue@almaden.ibm.com>,
cdrews@almaden.ibm.com

+ James Lin <jameslin@us.ibm.com>,
Eben Haber <ehaber@us.ibm.com>,
John Tang <john.tang@us.ibm.com>,
Jeff Pierce <jspierce@us.ibm.com>,
Alison Sue <alisue@almaden.ibm.com>,
cdrews@almaden.ibm.com

Figure 4. Recent Shortcuts client running on a Macintosh that also includes objects accessed through other devices.

To extend Recent Shortcuts across multiple devices, we leveraged recent research on providing seamless access to information across any of a user's devices [1]. Each of a user's devices running our extended version of Recent Shortcuts provides information about the people, attachments, and files accessed on it to the user's other devices, and in turn integrates information from those other devices. We implemented our extended Recent Shortcuts client using Java SE, allowing it to run on any Java SE platform, and are working on a Java ME version to run on more portable devices. Figure 4 shows the Recent Shortcuts Java client on a Macintosh, and we have also implemented a client for Linux.

Integrating inputs from multiple devices required some changes to the Recent Shortcuts user interface. The files and attachments in Figure 4 are annotated with the machine on which the user accessed them (Thoth is the machine name of the Macintosh). A file transfer and caching mechanism allows files initially accessed on one device to be available to any device through Recent Shortcuts, even if the original device is currently disconnected.

Extending our prototype to multiple devices required some programming tailored to each platform. For example, we have to interface with the mechanism that each platform provides for tracking recently used files ("Recent Items" in the case of Mac OS X), or create such a service if it does not exist on that platform. Interfacing to email and IM clients on each platform may also require customized work, although using extensions to cross-platform applications (e.g., our Thunderbird extension for email) simplifies that work. Ideally, a uniform application programming interface (API) would enable Recent Shortcuts to access the information it needs from each application on every platform.

Design Driven by User Feedback

Our initial usage experience and feedback identified several potential areas for future work. Besides incorporating additional design implications from our empirical study, areas for further development include:

- including more kinds of objects in Recent Shortcuts
- enable organizing, sorting, or filtering the list of objects in different ways
- revising the way Recent Shortcuts is integrated into the computer desktop

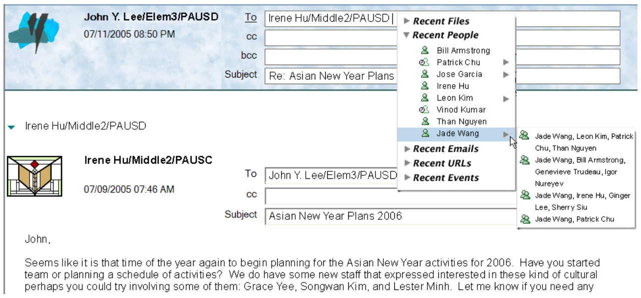
Our initial users of the prototype were quick to suggest other kinds of objects to include in the Recent Shortcuts list, such as recently viewed web pages and email messages (not just addresses). Integrating with address books, another request, would further enable migrating interactions between email, IM, and phone calls (although mechanisms for unifying references to people across email, IM, and phone would be needed).

Some users were disappointed that some interactions with files (e.g., copying an email attachment into a folder, saving a new file but leaving its window open in Microsoft Word, opening a file in a temporary directory from a web download) did not create an entry in the *Files* section of Recent Shortcuts. Since we use the same mechanism Microsoft Windows uses to populate "My Recent Documents", we are limited to the information that it captures concerning recently used files. Our experience shows that there are recent user interactions with files currently overlooked by Windows that should be included.

Adding more kinds of recently used objects or showing more items from the user's interaction history creates a design tension. It makes it more likely that Recent Shortcuts will contain what the user is looking for, but also makes it harder to find that item in the list. This tension suggests the need to explore other ways to sort and filter the items in Recent Shortcuts.

For example, a couple of Recent Shortcuts users requested sorting the list according to recency of use, rather than alphabetically (a design decision we made to leverage the familiarity of ordering in the recency mechanisms on Windows and Mac OS). Another approach would be to integrate other attributes of objects, such as frequency of use or inferred relationship with the current user task, to help sort the items within Recent Shortcuts. While Recent Shortcuts is driven largely by temporal context, other dimensions of context-aware computing research [14] could be explored to further enhance the ability of Recent Shortcuts to anticipate what objects the user might need.

Another way to organize and filter Recent Shortcuts might be to use the user's *current focus of attention*. For example, if the user's cursor is currently positioned in the email address field when she opens Recent Shortcuts, she is most likely looking to insert an email address for a person. Thus, Recent Shortcuts may expand only the list of people, leaving the other object types collapsed, as depicted in Figure 5. This filtering simplifies the list to present the items that are most likely to be useful in the user's current context, but still leaves other items in Recent Shortcuts accessible by expanding groupings or opening up sub-windows. This sorting and filtering would dynamically change as the user's input focus changed. So, when the user's input focus changes to the body of the email message, she may most likely want to insert a file attachment or a URL. Recent Shortcuts would respond by expanding those objects and collapsing the other item types to reduce interface clutter.



John Y. Lee/Elem3/PAUSD
07/11/2005 08:50 PM

To: Irene Hu/Middle2/PAUSD
cc:
bcc:
Subject: Re: Asian New Year Plans

Irene Hu/Middle2/PAUSD

Irene Hu/Middle2/PAUSC
07/09/2005 07:46 AM

To: John Y. Lee/Elem3/PAUSD
cc:
Subject: Asian New Year Plans 2006

John,

Seems like it is that time of the year again to begin planning for the Asian New Year activities for 2006. Have you started team or planning a schedule of activities? We do have some new staff that expressed interested in these kind of cultural perhaps you could try involving some of them: Grace Yee, Songwan Kim, and Lester Minh. Let me know if you need any

Figure 5. Concept screenshot of Recent Shortcuts filtered by the user's current input focus in the email addressing field, which expands "People" for access to individuals and groups.

A deeper integration of Recent Shortcuts into the desktop user experience would also require that applications, the operating system, and other computer resources expose usage information through APIs so that Recent Shortcuts can know what artifacts (files, people, URLs, email messages, etc.) have been recently used. Recent Shortcuts might also need APIs to get the user's current context to automatically filter and sort its items.

Balancing how much history and what kinds of data types to include in Recent Shortcuts with how to sort and filter the collected list and how aggressively to prune it down is an area open for further research. We expect that analyzing real world experience with using Recent Shortcuts will help in further tuning those design decisions.

Related Work

While contemporary operating systems (e.g., Windows XP, Mac OS X) have been integrating recency mechanisms in file choosers, Recent Shortcuts goes beyond them in the variety of recent objects supported, the variety of actions users can perform on those objects, and the ability to use those objects across applications. Besides these recency mechanisms in commercial OS products, there are other ideas and research projects related to Recent Shortcuts.

A number of add-in customization components are available to help users customize quick access to a selected set of files and folders. A representative example is Quick View in the Sidebar of Google Desktop [12]. Besides files, Quick View also includes web links. It allows users to choose between viewing recently accessed and frequently accessed objects, but does not provide access to people and attachments. While Quick View allows users to cut and paste selected items, it does not work with all applications, and users cannot drag items from it to copy. The Google Sidebar API could be used to integrate Recent Shortcuts into the desktop as a Sidebar plug-in.

A converse approach is to integrate communication functionality within other applications that create and view content. "My Recent Documents" and many productivity applications afford sending the file being worked on directly as an email attachment. Modern web browsers also offer sending a web link as email. However, despite the availability of this functionality, we saw no examples of users taking advantage of it in our empirical study. This application-centric integration of email requires users to recast what they think of as an email operation into an interaction initiated from a different application. Such integration is also limited. For example, it does not support the attachment of multiple file types to a single message. It is unclear whether users will become more adept at utilizing such integration over time.

Recent Shortcuts differs from other activity-based research prototypes in two important ways. Activity-centric computing usually involves associating each object or action in a computer with an activity or task. This association enables collating together all resources and sub-activities related to an activity for easy display and computational processing. However, these representations typically require users to change their work practices to designate how each object belongs to which activity. Activity Explorer [16] requires using a new application, and TaskTracer [7] and ABC [2] require users to tag each window with an activity to which it belongs. UMEA [13] implicitly tags user interactions with the currently active project, which the user must proactively select and manage.

Recent Shortcuts, on the other hand, simply tracks and displays objects with which users have recently interacted to exploit any implicit relationship, based on recency, with the user's current task. Since working on an activity often occurs over substantial stretches of time, quick access to recently used objects is likely to be helpful in accomplishing the current task. Passively collecting objects from the user's recent interactions does not require *any* additional user effort or change in work practice, and its lightweight interface provides quick access for easy re-use. Recent Shortcuts could be useful in populating more explicitly labeled activities or transitioning to more activity-centric user work practices.

Furthermore, the technical approach of most prior activity-based prototypes involves extensive modifications at the OS level. As a result, those systems are typically limited to a single OS platform, namely Windows. Our technical architecture captures user interactions at a level that is more easily implemented across OS platforms, as evidenced by our implementations for Windows, Mac OS X, and Linux.

We also contrast our approach with the Lifestreams work [10], which organized the computer desktop experience according to the chronological stream of files that the user encountered. While this metaphor enabled access to recently used files, it was organized solely around files and did not offer fine-grained access to other recently used objects (in particular, people). It also did not integrate with desktop or communications applications.

On the other hand, ContactMap [20] used people as the organizing resource for the computer desktop to help users find and manage social contacts, based in part on recency and frequency of interaction. This design is consistent with our observations of frequent searches for references to people in collaborative work. Our research includes people as one of several types of important recent objects to track in the Recent Shortcuts interface.

Some research prototypes have leveraged being able to index and find information in the stream of information that the user has viewed through a computer desktop. For example, Stuff I've Seen [9] indexes information that a user has previously viewed through the computer desktop and provides a query interface to search for and retrieve that information. They found that people tend to search for items that they have seen recently. Rather than a query interface, Recent Shortcuts presents a dynamically updated list of likely useful items that the user can quickly access, scoped by recency to keep the list of manageable size.

Informed by our empirical study, the design of Recent Shortcuts goes beyond the pieces of functionality that have been demonstrated in current systems and research projects. Recent Shortcuts integrates access to a wide range of recently used computer objects into a single, lightweight interface without requiring additional user effort. Furthermore, our focus on the work needed to coordinate with others on shared tasks enabled us to clearly see the importance of including quick access to the people and groups with which users recently interacted. We see the ways we have integrated people together with attachments and files into Recent Shortcuts as a distinguishing factor from most prior research prototypes that have focused on improving individual productivity *in isolation*.

EVOLVING THE DESIGN OF RECENT SHORTCUTS

We arrived at the Recent Shortcuts design requirements directly from our empirical study, which provided a systematic framework for exposing the frustrations that users frequently experience but often do not explicitly recognize when they coordinate activities. During our post-interviews, several participants quickly recognized how often they were annoyed by the extraneous effort revealed by our observations (e.g., looking back for people's email addresses, navigating around for files to attach to email). Our observations highlighted the individual work that is needed to set up collaborating with others, and led to the design of Recent Shortcuts as a way of easing that work around shared tasks.

Yet the design of Recent Shortcuts also evolved from observations enabled by early experiences with the working prototype. We have extended Recent Shortcuts to work across devices so that users can maintain access to recently used objects even if they switch to a different device.

We expect to continue learning from the deployment of the Recent Shortcuts prototype to gain user experience to extend and refine its design. We also hope to explore future design directions that expand the data types and capabilities of Recent Shortcuts. Our experiences with Recent Shortcuts will help refine a useful tool for supporting coordination around shared activities and inform the design of the activity-centric user experience.

ACKNOWLEDGMENTS

We gratefully acknowledge our eight anonymous participants in our empirical study for trusting us to screen record their work activity for a few days. Furthermore, we thank the nine anonymous trial users of Recent Shortcuts, for bravely working with our prototype code and giving us great feedback on the ideas. This work grew out of the Unified Activities Management project, and we thank Tom Moran, Michael Muller, Sophia B. Liu, and Andy Schirmer for their support and ideas along the way.

REFERENCES

1. Ahn, Juwon & Jeffrey S. Pierce. "SEREFE: Serendipitous File Exchange Between Users and Devices", *Proc. Mobile HCI 2005*, Salzburg, Austria, September 2005, pp. 39–46.
2. Bardram, Jakob E., Jonathan Bunde-Pedersen, and Mads Soegaard, "Support for Activity-Based Computing in a Personal Computing Operating System", *Proc. CHI 2006*, Montreal, Canada, April 2006, pp. 211–220.
3. Bellotti, Victoria, Nicolas Ducheneaut, Mark Howard, and Ian Smith, "Taking email to task: the design and evaluation of a task management centered email tool", *Proc. CHI 2003*, Fort Lauderdale, FL, April 2003, pp. 345–352.
4. Boardman, Richard & M. Angela Sasse, "'Stuff Goes into the Computer and Doesn't Come Out': A Cross-tool Study of Personal Information Management". *Proc. CHI 2004*, Vienna, Austria, April 2004, pp. 583–590.
5. "Camtasia Studio Screen Recorder for Demos, Presentations and Training", (verified December 21, 2006), http://techsmith.com/camtasia
6. Czerwinski, Mary, Eric Horvitz, and Susan Wilhite, "A Diary Study of Task Switching and Interruptions", *Proc. CHI 2004*, Vienna, Austria, April 2004, pp. 175–182.
7. Dragunov, Anton N., Thomas G. Dietterich, Kevin Johnsrude, Matthew McLaughlin, Lida Li, and Jonathan L. Herlocker, "TaskTracer: A Desktop Environment to Support Multi-tasking Knowledge Workers", *Proc. IUI 2005*, San Diego, January 2005, pp. 75–82.
8. Ducheneaut, Nicolas & Victoria Bellotti, "E-mail as habitat: An exploration of embedded personal information management", *Interactions*, 8(5), 2001, pp. 30–38.
9. Dumais, S. T., E. Cutrell, E., J.J. Cadiz, G. Jancke, R. Sarin, and D. C. Robbins, "Stuff I've Seen: A system for personal information retrieval and re-use", *Proc. SIGIR 2003*, Toronto, July 2003, pp. 72–79.
10. Fertig, Scott, Eric Freeman, and David Gelernter. "Lifestreams an alternative to the desktop metaphor", *CHI'96 Conference Companion*, Vancouver, 1996, pp. 410–411.
11. González, Victor M. & Gloria Mark, "Constant, constant, multi-tasking craziness: Managing multiple working spheres", *Proc. CHI 2004*, Vienna, Austria, April 2004, pp. 113–120.
12. "Google Desktop — Features", (Verified, December 21, 2006), http://desktop.google.com/features.html#sidebar
13. Kaptelinin, Victor, "UMEA: Translating interaction histories into project contexts", *Proc. CHI 2003*, Fort Lauderdale, FL, April 2003, pp. 353–360.
14. Moran, Thomas P. & Paul Dourish, "Introduction to This Special Issue on Context-Aware Computing", *Human-Computer Interaction*, 16(2–4), 2001, pp. 87–95.
15. Moran, Thomas P., Alex Cozzi, and Stephen P. Farrell, "Unified Activity Management: Supporting People in eBusiness", *Communications of the ACM*, December 2005, pp. 67–70.
16. Muller, Michael J., Werner Geyer, Beth Brownholtz, Eric Wilcox, and David R. Millen, "One-hundred days in an activity-centric collaboration environment based on shared objects", *Proc. CHI 2004*, Vienna, Austria, April 2004, pp. 375–382.
17. Ruhleder, Karen & Brigitte Jordan, "Capturing Complex, Distributed Activities: Video-Based Interaction Analysis as a Component of Workplace Ethnography", in *Information Systems and Qualitative Research*, A.S. Lee, J. Liebenau, and J.I. DeGross (eds.), London: Chapman and Hall, 1997, pp. 246–275.
18. Tang, John C., Sophia B. Liu, Michael Muller, James Lin, and Clemens Drews, "Unobtrusive but Invasive: Using Screen Recording to Collect Field Data on Computer-Mediated Interaction", *Proc. CSCW 2006*, Banff, Alberta, Canada, November 2006, pp. 479–482.
19. Whittaker, Steve, "Supporting Collaborative Task Management in Email", *Human Computer Interaction*, 20(1–2), 2005, pp. 49–88.
20. Whittaker, Steve, Quentin Jones, Bonnie Nardi, Mike Creech, Loren Terveen, Ellen Isaacs, and John Hainsworth, "ContactMap: Organizing Communication in a Social Desktop", *ACM Transactions on Computer Human Interaction*, 11(4), 2004, pp. 445–471.