Supporting Email Workflow

Gina Danielle Venolia, Laura Dabbish, JJ Cadiz, Anoop Gupta
Microsoft Research, Collaboration & Multimedia Group
One Microsoft Way, Redmond, WA 98052 USA
{ginav; jjcadiz; anoop}@microsoft.com

ABSTRACT
As more people use e-mail at home or on the job, more people have come to experience the pain of e-mail that Denning first wrote about 20 years ago [3]. In this paper, we present data from a field study in our own company to add to the existing body of research about how people use e-mail. We then use these data and prior literature to outline a framework of the five main activities that we believe people use e-mail for. In particular, we focus on two activities that we believe have been under-studied: attending to the flow of messages they arrive, and doing "triage" on a body of new messages. In addition, we outline potential design directions for improving the e-mail experience, with a focus on e-mail clients that group messages and their replies together into threads. We present a prototype of such an interface as well as results from a lab study of the prototype.

Keywords
Email, personal information management, task management, computer-mediated communication

1 INTRODUCTION

In 1982, Peter Denning (then the ACM President) first wrote about the pain of working with e-mail, calling it "The Receiver's Plight" and asking, "Who will save the receivers (of e-mail) from drowning in the rising tide of information so generated?" [3]. 20 years later, we still don't have the answer. Numerous studies have continued to provide data outlining the plight of e-mail users, and it seems the only thing that's changed is that the number of people experiencing this pain has risen dramatically. Feelings first expressed by the ACM president are now headlines in national newspapers: "E-mail overload taxes workers and companies" [10]. Furthermore, the trend isn't slowing. IDC reports that in the year 2000 there were 452 million email mailboxes and approximately 9.7 billion messages exchanged on an average day. In 2005, the numbers are predicted to jump to 983 million mailboxes and 35 billion messages.

Simply put, e-mail has become a place where many of us now live—a habitat, as stated by [5]—and as shown by Whittaker and Sidner [13], this place poorly supports the tasks we need to accomplish. As they note, e-mail has become overloaded: the usage and uses of email go far beyond what we could have imagined twenty years ago, but the interfaces of mail clients have not kept pace. In many ways, e-mail has become a victim of its own success.

2 PREVIOUS EMAIL LITERATURE

Researchers have been studying e-mail for quite some time. Much of the early work on social and organizational aspects of e-mail is summed up well by Sproull and Kiesler [11]. In this paper, we'd like to focus on designing e-mail clients to better support users' needs.

The research on how people work with their e-mail clients includes both studies of current use and studies of prototype interfaces. As noted in the previous section, Denning [3] was the first to note that current e-mail clients did little to help people who received lots of e-mail. Denning proposed several solutions to the problem based on two principles: first, there should always be a special path for people to get urgent, certified, and personal messages; and second, that all other paths should be filtered.

Six years after Denning's paper, Mackay [8] published results from an extensive study of e-mail (based on the Information Lens system built by Malone et al. [9]). Her results included two primary findings: people use e-mail in incredibly diverse ways, and people use e-mail for much more than just basic communication (e.g. task management, task delegation, time management, archiving information for future use). She also found that people generally fell into one of two categories when it came to handling e-mail: archivers or prioritizers. Archivers focused on strategies for making sure that they would see all messages and not miss anything important; prioritizers focused on strategies to limit the time they spent with e-mail so that they could get other work done. In a nutshell, prioritizers controlled their e-mail while archivers were controlled by their e-mail. Mackay also classified people based on whether they were "overwhelmed", "on the edge", or "ok" when it came to handling all their e-mail.

Eight years after Mackay's work, Whittaker and Sidner [13] published their study on e-mail use within Lotus. Like Mackay, they found that e-mail was being used for several tasks in addition to basic communication, calling the phenomenon "e-mail overload." They also studied how people handled e-mail overload when it came to filing messages and classified people as no filers (people who don't clean up their inbox but use searching tools to manage it), frequent filers (people who constantly clean up their inbox), and spring cleaners (people who cleaned up their inbox once every few months).

Five years after Whittaker and Sidner's work, Ducheneaut and Bellotti [5] published their study, which examined e-

mail usage in three organizations. Like the previous studies, they also found that e-mail is used for a variety of tasks. In fact, they found that people used e-mail so often for so many tasks that they called e-mail not just a killer application, but a "serial killer", writing: "It is seriously overloaded and has been co-opted to manage a variety of tasks that it was not originally meant to support."

3 METHODOLOGY

Previous studies of e-mail portray a clear trend: e-mail is being used for a number of tasks that it was never meant to support. Thus, we sought to use the existing literature and our own field study to determine the main activities that people need e-mail clients to support. To accomplish this goal, we studied several employees in our company using three methods: interviews, analysis of message archives, and a survey.

3.1 Structured Interviews

We interviewed ten individuals for the first part of our study. Participants included a systems engineer, a television studio engineer, an encyclopedia editor, a sales representative, an administrative assistant, a game tester, two project managers, and two training coordinators. All interviews were scheduled for one hour in the participant's office, to occur after a period of absence from the computer—first thing in the morning or after a meeting—so there would likely be some new messages waiting. In addition, participants were asked beforehand to refrain reading new messages for the day prior to the interview.

Part of the interview was conducted in a contextual inquiry manner where participants worked with their mail while thinking aloud, while the rest of the interview was conducted in a more traditional manner with the experimenters asking a variety of questions about how often participants checked their mail, their folder hierarchies, how they handled each message, what they liked and disliked about e-mail, and so on.

3.2 Message Archives

We used a tool to collect ten message archives for the second part of our study. Seven of the archives were collected from our interview participants (technical difficulties prevented us from collecting archives from the other three interviewees) and three of the archives were from members of the authors' workgroup. The tool collected all the information in users' e-mail archive including thread structure of messages, folder hierarchy, where messages were filed, whom messages were sent to, etc. The only information that wasn't collected were subject lines and bodies of messages.

3.3 Survey of E-Mail Use

The last method we used in our study was a web survey. Based on our interview findings, we developed a survey asking a variety of questions about what makes a message important or unimportant, how people handle messages when they arrive, how people use e-mail as a task planning tool, how people file messages, and how people retrieve older messages. This survey was sent to approximately 1,500 people via general-interest discussion lists, resulting in 406 completed surveys. The majority of the questions were answered using a 5-point Likert scale where 1 = "strongly disagree" and 5 = "strongly agree".

3.4 The Participants

It's important to note that we work for a software company, thus our study participants are arguably above average when it comes to technical expertise. At the same time, we were careful to include a wide variety of job roles, and we excluded anyone who had ever been involved in the design or development of an e-mail client. All of our participants used Microsoft's Outlook as their e-mail client, which is standard in our company.

4 FIVE E-MAIL ACTIVITIES

Based on our review of the literature and data collection, we have developed a conceptual model of users' activities surrounding email. We have identified five different activities:

Flow: As people are working on other tasks, they want to keep up with the flow of incoming messages as they arrive.

Triage: After people are away from their e-mail for a period of time, they need to catch up and deal with all the e-mail that accumulated while they were away.

Task management: People often use e-mail to remind them what they need to do, and to help them get tasks done.

Archive: People store e-mail so they can refer to it later.

Retrieve: After archiving messages, people need a method of retrieving messages.

While the latter three activities are often discussed in the literature, less attention has been paid to the first two. The remainder of this section will discuss each of these five activities in depth, along with ways that interfaces might better support these activities.

4.1 Flow Activity

As stated by [5], e-mail has now become a habitat that many of us live in. However, as much as we might like to, we can't live in e-mail all the time. Eventually people have to do other work on their computers, and while they do, they like to keep track of incoming messages as they arrive, an activity we call keeping up with the "flow." This desire to be aware of message arrival was clearly indicated in our survey responses. The median response to the statement, "When I'm at my computer and a message arrives, I immediately look at it" was 4 or "agree" (avg=3.7, sd=0.9).

Unlike the other four activities we discuss, the flow activity is typically a secondary background activity that is unrelated to the primary task being performed (writing a document, reading a web page, etc.). Thus, when users receive a new message, a series of tasks is triggered

Factor	Mean
Reply to my message	4.3
Sole recipient	4.2
From Manager	4.2
I'm on TO line	4.1
"High Importance" flag	4.1
From project member	4.0
From direct report	3.9
From management chain	3.7
From peer	3.7
Interesting auto-preview	3.6
Interesting subject line	3.6
To fewer than five	3.5
From family member	3.4
I'm on CC line	3.2
From friend	3.2
Important DL on TO line	3.1
From administrator	3.1
To fewer than ten	3.0
From other person in org	2.9
Important DL on CC line	2.6
To more than ten	2.6
From unknown sender	2.1

Table 1: Factors in message importance. Mean responses to survey questions of the form, "A message is particularly important if…"

revolving around evaluating the message and deciding what action to take.

Unfortunately, the flow task requires too much energy. Currently, Outlook provides three methods of being notified of new mail: playing a sound, displaying an icon in the Windows task bar, or briefly changing the mouse pointer. When users are notified of a new message, they have to stop what they're doing, switch to Outlook, and read the message in order to determine if they need to do anything. When finished, they have to remember what they were doing before and switch back to it. This context switching can be very painful. In fact, several of our interview participants said that when they were stressed or deeply involved in a task, they would ignore Outlook when a new message arrived, turn off new mail notifications, or shut down Outlook altogether.

To support people when they are involved in the flow activity, all they need from their e-mail clients is enough information to decide whether they need to stop what they're doing and deal with the message that just came in. Outlook's e-mail notifications don't support this activity in the following ways:

1) There is often not enough information in the notification to help decide whether the arriving email is important

2) The notification is not based on the priority or importance of the message, i.e. the notification is the same regardless of message

3) The notification is not sensitive to the email device and physical context

4) The notification does not enable UI for few quick common commands – e.g. delete/open/reply – if such a decision can indeed be made based on what is shown in the notification.

Fortunately, solving this problem can be as simple as displaying just a little more information about new messages such that users aren't forced to switch away from what they're doing if they don't need to. Microsoft's Messenger service already does this with new Hotmail messages, but a similar feature doesn't exist for Outlook. Interestingly, when a prototype (unrelated to this project) that provided this feature for Outlook was distributed within our company, the data indicated that this single feature was one of the most popular features, even though it was a relatively minor part of the prototype [2].

Of course, when creating mail notifications, one question is which information to display in the notification. Hotmail's notifications display sender and subject line, but as Table 1 shows, people may benefit from seeing additional information so they can decide whether to deal with the message. A more involved approach is to use an intelligent agent to infer over time what makes a message important to a user. Agents can then use this information to decide whether incoming mail warrants an interruption [6].

4.2 Triage Activity

People often spend blocks of time going through their mail and deciding what to do with all their messages. This time is used to process both unread messages and messages that have been read but saved for later action. We call this activity "triage".

Triage can be triggered by several events. First, nearly all our survey respondents indicated that they performed the triage activity on their inbox after being away form their mail for a while. The median response to, "When I get to work in the morning, the first thing I do is check my inbox" was 5 or "strongly agree" (avg=4.8, sd=0.4). The median response to "When I get back from a meeting, the first thing I do is check my inbox" was also 5 (avg=4.7, sd=0.6). Triage may also be triggered by a full inbox (median=4, avg=4.1, sd=1.1) or by the arrival of an important message (median=4, avg=3.9, sd=1.1). Note that performing triage on a single message as soon as it arrives is essentially the "flow" activity discussed in the previous section.

In our interviews we observed two dominant strategies for approaching the Triage activity: serial (3 of 10 interviewed participants) or prioritized (7 of 10). Participants who used the serial strategy read messages in the order of arrival, while those who used the prioritized strategy either skipped around picking out interesting senders or subject lines, or used sorting to group messages by sender. The dominance of the prioritized strategy was supported in the survey: the median response to "When I have a lot of mail to read through, I skip around to find important messages" was 5 (avg=4.2, sd=1.0).

We believe two reasons underlie the use of the prioritize strategy. First, people have a greater need to keep aware of things that are important to them and that have potential of greater impact on their life. Second, people may not be able

to finish the triage task before they have to attend to some other task, thus people want to deal with the most important messages first.

Thus, the key UI challenge for the triage activity is supporting intelligent/prioritized browsing of pending messages. This includes providing enough information and flexibility in the UI for people to identify messages they consider important, support for dealing with collections of related messages (threads) effectively, and greater support for reading and taking action on these threads. We discuss a prototype for threading messages in section 5.

4.3 Task Management

It's clear that people rely heavily on their e-mail clients to help them keep track of what they need to do. Mackay [8] found this, Whittaker and Sidner [13] found this, Ducheneaut and Bellotti [5] found this, and we found this in our study. 6 of our 10 interview participants used e-mail messages as their to-do lists, and on our survey, the median response to "I keep messages as reminders for later action when I owe a response" was 4 or "agree" (avg=4.3, sd=0.7). People also kept messages that needed to be read later (median=4, avg=4.1, sd=0.8) and messages for which the respondent was owed a reply (median=4, avg=3.9, sd=1.0).

However, the problem we observed is that there's no single successful method provided by Outlook for handling tasks. Although Outlook provides a separate Task list tool, only three of our interview participants used this feature. Furthermore, on our survey, we asked, "If a message needs action but I can't do it right away, I move it to the Outlook Task list". The median response was 2 or "disagree" (avg=2.4, sd=1.3).

In addition to the Task list, Outlook also provides several low-level methods for handling messages that need future action: leave in inbox, mark as unread, flag for follow-up, move to a specific to-do/project folder, move to calendar, and so on. As shown in Figure 1, by far the most popular strategy is keeping everything in the inbox. This strategy was so prevalent that in our interviews, we even observed the same thing Ducheneaut and Bellotti found: people placing non-email related tasks in their inbox by sending themselves mail.

Of course, the problem with keeping everything in the inbox is that the inbox can quickly become swamped with messages, making it difficult to figure out what needs to be done. When we asked, "I can easily tell which messages I have kept as reminders", the median response was 3 or "neutral" (avg=3.2, sd=1.3). Whittaker and Sidner [13] also found this problem and made two suggestions for improving the interface to better support the activity of task management: grouping messages by thread, and allowing people to flag messages such that the system would remind them later about the message. Outlook supports the latter suggestion, but it doesn't appear to be widely used: when

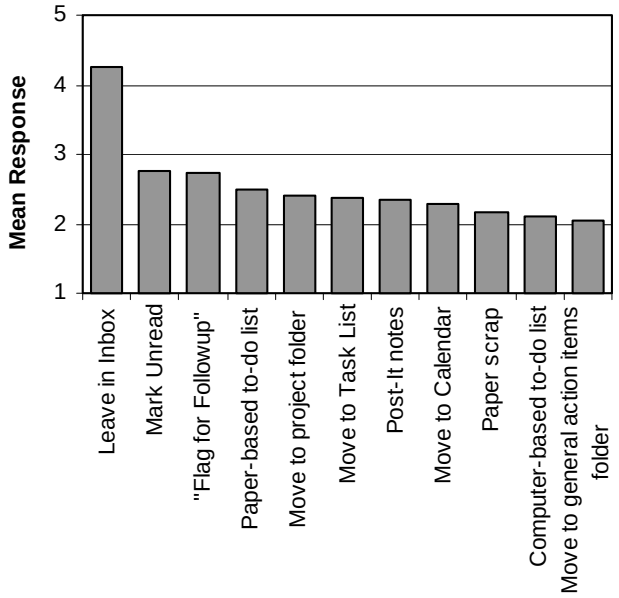

Figure 1: Users task management strategies. Average Likert scale ratings for various mechanisms for turning a message into a task reminder.

we asked, "If a message needs action but I can't do it right away, I use the 'Flag for Follow Up' feature" the median response was 2 or "disagree" (avg=2.7, sd=1.4). We discuss Whittaker and Sidner's other suggestion—grouping messages into threads—in section 4.

4.4 Archive Activity

Eventually, most messages reach a state when they require no further action. Some messages reach this state immediately after they're read, others reach this state after they've been replied to, and others reach this state after some other task has been performed. Once a message requires no further action, the message can be deleted or kept for future use. We call the activity of keeping messages for future use the "archive" activity.

It's clear that archiving messages is a very common activity. In our survey, when we asked, "I organize saved mail into folders" the median response was 5 or "strongly agree" (avg=4.5, sd=0.7). However, the frequency with which people archived messages varied: according to our survey data, 67% of respondents filed daily or weekly, 23% monthly, and 10% rarely or never, corresponding to frequent filers, spring cleaners, and no filers categories discussed in [13]

As shown in Figure 2, people archive messages for a variety of reasons. Clearly the information content of the message is important: the median response to "I try to keep a message easy to find when I may want the information it contains later" was 4 or "agree" (avg=4.3, sd=0.7), but we also found that people tend to file messages when they have objects in them that may be of future use (files, file pointers, web links, etc.). In fact, the second most popular reason for keeping a message is because of attachments it

may contain. [5] reported similar findings, noting that e-mail is now the main method of exchanging documents. They also explained that the popularity of exchanging documents via e-mail has to do with the ability of the message to provide context for the attachment and the ability of the attachments to elaborate and enrich the text in the mail.

Attachments

Unfortunately, attachments pose a problem for today's mail clients. Because many mail systems (like Outlook) keep their data on a central server, size quotas have to be enforced. In our company, mailbox quotas are 100MB, and this limit can be reached quickly if several attachments are sent and received. Thus, users have to delete the whole message, delete the attachment, or save the attachment in the file system and deleting it from the message. In all of these cases, users lose the exact synergy that Ducheneaut and Bellotti [5] emphasize that users really value.

Ideally, we can solve this simply by invoking the massive increase in hard-disk capacities every year so people never have to delete anything, but in managed client-server systems today this remains a complex problem to solve cleanly. The more involved solution is to create mail systems that allow users to store attachments separately from their mail but still maintain links from the message to the file and from the file to the message.

Using Folders

The most common way today's e-mail clients facilitate archiving of messages is with the use of folders. Previous research has examined users' folder structures, and we did the same. For the 10 people whose archives we studied, the average number of folders was 104 folders (min 11 and max 309). These folders were organized in a hierarchy with a typical depth of 2 or 3, but one had 5 levels. Overall, these data indicate that the complexity of folder structures has increased since Whittaker and Sidner's [13] study in 1996 when on average *no filers* had 11.33 folders, *spring cleaners* had 61.43 folders, and *frequent filers* had 70.6 folders. Our numbers are also higher than the numbers reported by Mackay in 1988 [8] where the average number of folders was 33.

Clearly, having so many folders can lead to problems, including folders having too many or too few messages to be useful [13], mail clients enforcing alphabetical ordering of folders, which isn't what users always want [5], having many folders that are no longer useful, and having so many folders that filing often requires scrolling through a long list of folders [8, 13]. Fortunately, users don't seem to have a problem finding a place for messages to go. In our survey when we asked, "When filing a message, I know exactly where it should go," the median response was 4 or "agree" (avg=3.9, sd=1.0).

However, just because people know where to file every message doesn't mean that every message belongs in just one folder. One problem with most programs like Outlook is that they only allow messages to be stored in one folder. The problem becomes more acute when dealing with entire threads of messages. In our analysis of message archives, we found that 23% of all message threads were spread across more two or more folders, mostly because of Outlook's habit of automatically placing one's replies in the "sent items" folder, which guarantees that it's in stored in a separate folder than its parent message.

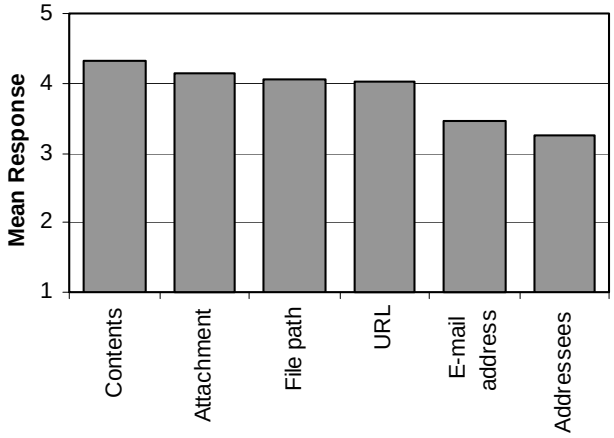

Figure 2: Reasons for filing. Mean responses to survey questions of the form, "I try to keep a message easy to find when I may later want…"

As an aside, we should note that Outlook does allow users to associate multiple category labels with messages, which can be a way around the limitation of only being able to store mail in one folder. However, both the UI to attach and retrieve messages using categories is cumbersome, and they are not used: the median response to "I use the Categories feature in Outlook" was 2 or "disagree" (avg=2.1, sd=1.2).

Rules and Filtering

Rules and filters are methods for assisting users with the activity of archiving mail. These methods have existed for quite a while [8, 9] and are provided by many e-mail clients today, including Outlook. Rules were popular among our survey respondents, with 46% of respondents reporting the use of rules to filter incoming mail by sender, and 69% by discussion list. In fact, 24% of respondents had more than six rules for filing by discussion list.

Unfortunately, while rule-based filing is becoming common, it interacts poorly with the use of email for task management. If task-related messages are automatically filed away in subfolders before users have looked at them and if they explicitly have to go to those subfolders to check pending tasks, people will more likely miss things (following the out-of-sight out-of-mind cliché reported by [13]). One solution is to allow messages to exist in several locations (in the inbox, the to-do folder, and some project specific folder), but as discussed in the previous section, this solution isn't provided by many mail clients. However,

one system that provides this functionality is described next.

Mail without Filing

Filing takes time and today's systems only allow messages to be stored in one place. However, one system developed by researchers at the Compaq SRC labs called Pachyderm [1] introduces a system that solves both of these problems. Pachyderm is based on the notion that there should be no folders (in form of separate storage buckets) and all messages should reside in a single conceptual store. However, users can create folders using standing queries (search commands that are continuously updated). Thus, instead of creating a folder for all mail about project "Gresham", I can create a standing query for all messages sent to the "Gresham" distribution list and all messages containing the word "Gresham" in the subject or body. Users' collections of standing queries can be represented just like the folder hierarchy, with the advantages that no filing is required, and messages can exist in the results from several standing queries. Similar issues have been explored in the document management space by Dourish et al. [4].

4.5 Retrieve Activity

People archive messages because they want to be able to recover them later, thus clearly another main activity people need to do with their e-mail clients is retrieve older messages. Just as with archiving messages, retrieval is a very common activity: when we asked the question, "I never access old messages," the median response was 1 ("strongly disagree").

Clearly your archiving strategy affects your retrieving strategy. If you have a well-formed folder hierarchy and you don't have many problems filing mail in the correct folder (as survey data in the previous section indicated), then retrieving messages should be easy. When we asked, "When I need to access an old message, I first look in one of the folders I've created," the median response was 4 ("agree"). Furthermore, it seems that people didn't have many problems knowing which folder to look in. When we asked, "When I need to access an old message, I know the folder that I filed the message in," the median response was 4 ("agree").

However, one surprising aspect of the data was the extent to which people look for messages in their "sent items" folder. When we asked, "When I need to access an old message, I first look in the Sent Items folder", the median response was 4 ("agree"). This was consistent with the strategy of one interview participant who always deleted messages as soon as he responded to them. He figured that if a message was important enough to look for again, he likely had responded to it, thus there would be a copy in the "sent items" folder. (However, this participant also admitted that he felt comfortable with this strategy because others in his group were extremely good at keeping copies of all important mail, thus he could always ask them for an old message if he couldn't find it. Note that the strategy of depending on others to be good archivers has been found to exist with archives of important paper documents [12].)

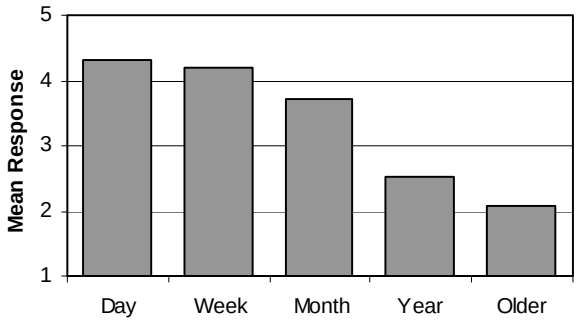

Figure 3: Age of retrieved items. Mean responses to survey questions of the form, "I frequently need to access messages that are a ___ old."

Another interesting finding from our survey with respect to retrieval was the age of messages that tended to be retrieved. As Figure 3 shows, survey respondents believed that as a message got older, the chances of them needing to retrieve the message declined. The implication is that day-old messages are still within the first screen of the inbox, thus a visual scan suffices and no search is needed.

5 TESTING A THREADED E-MAIL CLIENT

At several points in section 4, we noted that grouping messages together that were part of the same reply chain would help alleviate some of the pain experienced by the users of e-mail. In this section we discuss an early prototype of an interface we developed to facilitate the use of threads, along with results from a lab study that tested this interface.

5.1 Why Threads Help

We believe providing a threaded e-mail client has the potential to help users in three main ways. First, displaying a message along with all the replies above and below it in the chain provides *better local context*, which can help users better understand conversations that occur via e-mail. Although this context is somewhat preserved by current e-mail programs when they automatically include the text of all previous messages in replies, this method breaks down when multiple people reply to the same message, creating a complex, branching reply tree.

Second, by making the main unit of display the thread, more items can be displayed at the same time, providing *greater global context*. As noted in previous sections, users' strategies often depend on how many messages they can view at once in the inbox. Thus, by collecting messages into threads, sets of messages that normally would have been displayed on several lines can be displayed on just one line, allowing people to view more items at once.

Third, when users work primarily with threads instead of individual messages, the interface can provide valuable *global operations*. Currently, if I receive five messages

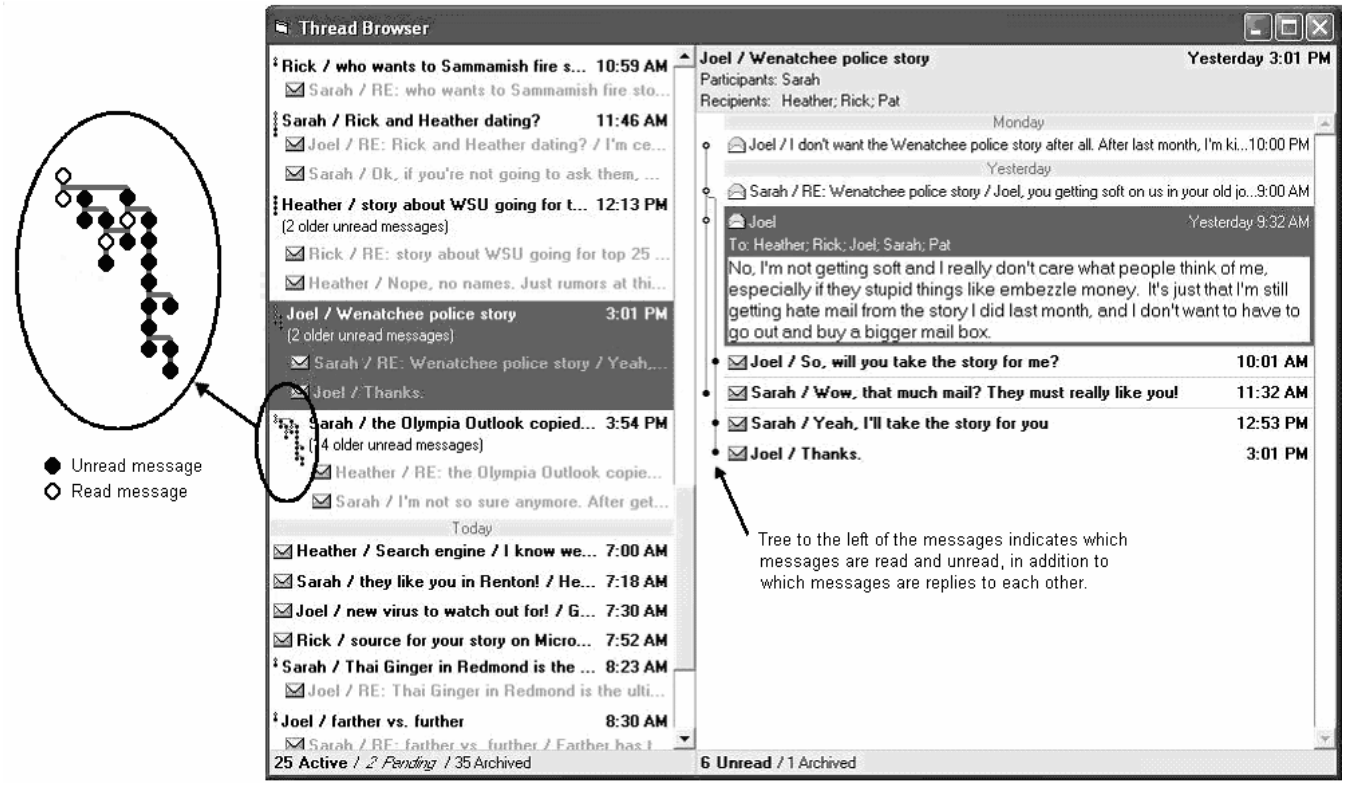

Figure 4: Thread-based mail browser. Messages that are replies to each other are grouped together into one item and displayed on the left. Clicking on one of these items displays all the messages inside the thread on the right. Messages are displayed in one-line preview format on the right, and clicking the message displays the entire message. In both the left and right panes, a thread tree is displayed to help the user determine what the structure of the thread is and how the messages relate to each other.

that are all part of the same thread, I have to perform five sets mouse and keyboard actions to work with all the messages (read, file, delete, etc.). However, if all the messages are grouped together, I only have to perform one set of mouse and keyboard actions. While this may seem like a small benefit, multiplied over the large number of e-mail messages, the benefit translates to a significant saving. In our analysis of user's e-mail archives, we found that 54% (sd=26%) of messages occur in threads of two or more messages (although this may be an underestimate given that people likely delete messages that are parts of threads). In addition, higher-level operations are also possible. For instance, if you start receiving messages on a topic that you're really not interested in, you could "unsubscribe" from the thread such that all current messages would be deleted, as well as all future messages on the same topic. As another example, if you wanted to add someone to the conversation, instead of having to forward the person all current and future messages (or trying to find the one message that was a good summary of the thread), you could "subscribe" the person to a thread such that they would receive all past and future messages.

5.2 The Prototype Interface

To explore benefits of a threaded e-mail client, we built an early prototype using Visual Basic. To begin, we wanted the prototype to support just one of the five activities discussed in section 4, thus we chose the triage activity: we wanted to see how a threaded e-mail client could help people process a very large amount of unread messages. This prototype is shown in Figure 4. The prototype has two panes: the left pane shows the list of threads, and clicking on a thread displays it in the right pane.

The thread list in the left pane is sorted by most recent message in the thread, and visually grouped by day. Each entry in the thread list shows a number of attributes of the thread: a thumbnail of the thread tree, the sender, the subject, the time of the most recent message, and a line with sender and subject per unread message (up to three). In the thread list, a thread tree thumbnail appears to the left of each thread that contains two or more messages. This compact visual representation gives users a high-level sense of the structure of the thread, as well as which parts haven't yet been read by the user.

The right pane consists of a header and a list of messages. The header shows the sender of the first message in the thread, others who have sent messages, others who have received messages in the thread, the title of the original thread message, and the date of the most recent message in the thread. The message list is sorted by message date (oldest to newest), and grouped by day. Each entry in the message list shows an icon indicating its read/unread status, the sender, the subject (if different from the previous), the first few words of the body, and the time. The selected message is expanded inline, showing sender, subject, "to"

addressees, "cc" addresses and the complete body. Just to the left of list is a parallel depiction of the messages as a thread tree so that users can see how the individual messages relate to each other.

One design decision to note is the departure from indenting messages to signify replies. Most Usenet browsers display messages in a thread as an indented tree. The tree display has a couple of flaws: deep trees, the typical shape for email conversations, result in substantial indenting, wasting valuable display space; the newest messages are distributed almost randomly through the list of messages; and when writing mail, it is not uncommon to refer to *any* prior message, not limited to the ancestors in the tree. The tree display destroys the temporal order, making the complete message context difficult for the reader to understand. We chose instead to sort the message list in the right pane by date, avoiding all three problems. In addition, it allows grouping by day, helping to give a sense of the temporal characteristics of the conversation.

5.3 Testing the Prototype

To test our thread interface, we recruited sixteen participants who had used e-mail for their job for at least 6 months and received at least 15 messages on a typical work day. Participants were told that they were a journalist who had just returned from vacation. Their job was to go through 200 e-mail messages that had accumulated and enter all the tasks they had to do in a spreadsheet (the e-mail messages were generated by the experimenters). Participants were given 25 minutes to complete the task. Half the participants were randomly assigned to use the thread interface while the other half used the same interface with threading turned off (the left pane of the interface shown in Figure 4 just showed the list of all 200 messages, and clicking on a message displayed it in the right pane).

In a post-test questionnaire, participants responded to a number of questions on a 5-point Likert scale, where "strongly disagree" was 1 and "strongly agree" was 5. For the question, "I didn't like using this email program to read the messages," the median response of subjects who used the message prototype was 4 (avg=3.6, sd=0.9) while the median response of those who used the thread prototype was 2 (avg=2.3, sd= 0.5). Analysis by a Mann-Whitney U test found this difference to be significant (z=−2.8; p=0.007), thus the thread prototype was preferred.

Users who used the threaded interface also commented that the threads helped them perform their task better. One participant wrote, "All messages referring to one idea were grouped together. Made it easy to read & refer back." Another participant wrote, "I could easily see if something was resolved before I spent time on it myself."

6 CONCLUDING REMARKS

In this paper we have identified five major activities surrounding how people use email. In particular, we've highlighted two activities—keeping up with the flow of incoming messages, and triaging existing messages—that we believe are important, but haven't been widely covered by previous studies. For each activity we have discussed the mismatch between user expectations and what current interfaces support, how the problems have changed (or not) during the past decade, and possible solution directions. It is quite amazing how the majority of problems have remained unchanged and unaddressed, with the main new problem being dealing with attachments. Finally, we've presented an early prototype of a thread-based e-mail client, as well as results from a lab study evaluation. The results demonstrate clear benefits for the triage activity.

REFERENCES

1. Birrell, A., Wobber, E., Schroeder, M., Perl, S. Pachyderm, https://pachyderm.pa-x.dec.com/. Personal communication from Andrew Birrell.
2. Cadiz, J., Venolia, G., Jancke, G., and Gupta, A. (2001). Sideshow: Providing Peripheral Awareness of Important Information. *Microsoft Research Tech Report* MSR-TR-2001-83.
3. Denning, P. (1982). Electronic Junk. *Communications of the ACM*, 25(3), March 1982.
4. Dourish, p., Edwards, K., LaMarca, A., and Salisbury, M. (1999). Presto: An Experimental Architecture for Fluid Interactive Documents Spaces. *ACM Transactions on Computer-Human Interaction*, 6(2).
5. Ducheneaut, N., and Bellotti, V. (2001). Email as Habitat. *Interactions*, September/October 2001.
6. Horvitz, E., Jacobs, A., and Hovel, D. Attention-Sensitive Alerting. (1999). *Proceedings of the Conference on Uncertainty and Artificial Intelligence* (UAI '99).
7. Levitt, M. (2000). Email Usage Forecast and Analysis, 2000-2005. *International Data Corporation*, IDC Report #23011, September 2000.
8. Mackay, W. (1988). Diversity in the Use of Electronic Mail: A Preliminary Inquiry. *ACM Transactions on Office Information Systems*, 6(4), October 1988.
9. Malone, T., Grant, K., Turbak, F., Brobst, S., and Cohen, M. (1987) Intelligent Information-Sharing Systems. *Communications of the ACM*, 30(5), May 1987.
10. Schwartz, J. (2001). E-mail overload taxes workers and companies. *USA Today*, June 26, 2001, pg A-1.
11. Sproull, L., and Kiesler, S. (1991). Connections: New Ways of Working in the Networked Organization. MIT Press: Cambridge, Massachusetts.
12. Whittaker, S., and Hirschberg, J. (2001). The Character, Value, and Management of Personal Paper Archives. *ACM Transactions on Computer-Human Interaction*, 8(2), June 2002.
13. Whittaker, S., and Sidner, C. (1996). Email Overload: Exploring Personal Information Management of Email. *Proceedings of the ACM Conference on Human Factors in Computing Systems* (CHI 1996).