

## **NOTE TO USERS**

**The original manuscript received by UMI contains pages with indistinct and/or slanted print. Pages were microfilmed as received.**

**This reproduction is the best copy available**

**UMI**



**Time-based Management and Visualization of  
Personal Electronic Information**

by

**Kelvin Shek Yiu**

A thesis submitted in conformity with the requirements for the degree of  
Master of Applied Science  
Graduate Department of Electrical and Computer Engineering  
University of Toronto

© Copyright by Kelvin Shek Yiu, 1997.



National Library  
of Canada

Acquisitions and  
Bibliographic Services

395 Wellington Street  
Ottawa ON K1A 0N4  
Canada

Bibliothèque nationale  
du Canada

Acquisitions et  
services bibliographiques

395, rue Wellington  
Ottawa ON K1A 0N4  
Canada

*Your file* *Votre référence*

*Our file* *Notre référence*

The author has granted a non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of this thesis in microform, paper or electronic formats.

The author retains ownership of the copyright in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de cette thèse sous la forme de microfiche/film, de reproduction sur papier ou sur format électronique.

L'auteur conserve la propriété du droit d'auteur qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

0-612-29420-X

## Abstract

TimeStore is a personal information management system that takes advantage of human autobiographical memory. Information such as email messages, tasks, calendar events, and notes are plotted on a two dimensional graph where time is displayed along the x-axis and the names of message sender are listed along the y-axis. Timestore also uses full text searching to replace the need of a semantic hierarchy to categorize information.

This project included a redesign of TimeStore where many of the usability and functionality problems with the previous prototype were resolved. A usability study was conducted using Microsoft Camcorder to capture on screen interactions and users' thoughts spoken aloud for later analysis. The user evaluation indicates that users find time-based visualization and full text searching useful. Suggestions for improvements in TimeStore and the data capture tool are presented in the final chapter.

## Table of Contents

|  |           |
|--|-----------|
| <b>CHAPTER 1 - INTRODUCTION</b>                        | <b>1</b>  |
| History of the TimeStore Project                       | 3         |
| Time-Based Visualization vs. Semantic Hierarchies      | 4         |
| Thesis Outline   | 5         |
| <b>CHAPTER 2 - RELEVANT RESEARCH</b>                   | <b>6</b>  |
| Research in Human Memory                               | 6         |
| Studies of Information Organization and Usage          | 12        |
| Information Retrieval                                  | 16        |
| Time-Based Information Systems                         | 19        |
| Commercial Email Systems                               | 23        |
| Usability Testing Methodologies                        | 26        |
| Relevance for TimeStore                                | 28        |
| <b>CHAPTER 3 - TIMESTORE USER INTERFACE DESIGN</b>     | <b>30</b> |
| The New Interface                                      | 30        |
| A Comparison with the Previous TimeStore Interface     | 37        |
| <b>CHAPTER 4 - THE IMPLEMENTATION OF TIMESTORE 3.0</b> | <b>42</b> |
| <b>CHAPTER 5 - USABILITY TESTING</b>                   | <b>49</b> |
| Methodology  | 50        |
| Initial Interview Questions                            | 52        |
| Daily Usage Questions and Final Interview Questions    | 54        |
| Results of the Initial Interview                       | 56        |
| Results of the Daily Usage and Final Interview         | 57        |
| Lessons Learned about Methodology                      | 62        |
| Factors Affecting TimeStore's Usage                    | 63        |

|   |            |
|---|------------|
| <b>CHAPTER 6 - CONTRIBUTIONS AND FUTURE WORK</b>            | <b>66</b>  |
| Contributions   | 66         |
| Improving the User Interface                                | 67         |
| Proposed Extensions to TimeStore                            | 72         |
| Building A Better Data Collection Tool                      | 74         |
| Conclusion  | 76         |
| <b>APPENDIX A - INITIAL INTERVIEW RESULTS</b>               | <b>78</b>  |
| <b>APPENDIX B - INITIAL USER IMPRESSIONS</b>                | <b>90</b>  |
| <b>APPENDIX C - INDIVIDUAL USER DAILY USAGE DATA</b>        | <b>94</b>  |
| <b>APPENDIX D - USER COMMENTS FROM INFORMAL DISCUSSIONS</b> | <b>114</b> |
| <b>APPENDIX E - FINAL INTERVIEW RESULTS</b>                 | <b>123</b> |
| <b>REFERENCES</b>   | <b>130</b> |

## LIST OF FIGURES

| <i>Number</i> |  | <i>Page</i> |
|---------------|--|-------------|
| Figure 1 -    | Day of week (DOW) errors calculated from the data in figure 2.   | 10          |
| Figure 2 -    | The difference in days between the remembered and actual dates of personal events.                                 | 11          |
| Figure 3 -    | Deviation of remembered month and year of moving to a new home from the actual time of the move.                   | 11          |
| Figure 4 -    | Lifestreams user interface.  | 20          |
| Figure 5 -    | Lifelines showing juvenile records.  | 21          |
| Figure 6 -    | Lifelines showing medical records.   | 22          |
| Figure 7 -    | Eudora Pro 3.0.1.  | 23          |
| Figure 8 -    | Microsoft Outlook.   | 24          |
| Figure 9 -    | Outlook's Journal feature where the user can display all Office documents, faxes, email, etc in a timeline format. | 25          |
| Figure 10 -   | Netscape Navigator 3.01's built-in email client.   | 26          |
| Figure 11 -   | TimeStore main window.   | 31          |
| Figure 12 -   | TimeStore inbox window.  | 32          |
| Figure 13 -   | Create new calendar event dialog box.  | 33          |
| Figure 14 -   | Create task dialog box that associates the task with an email message.   | 34          |
| Figure 15 -   | TimeStore taskbox window.  | 34          |
| Figure 16 -   | TimeStore address dialog box.  | 36          |
| Figure 17 -   | Create mailbox dialog box.   | 37          |
| Figure 18 -   | TimeStore 2.0.   | 38          |
| Figure 19 -   | Message handling data structures.  | 44          |
| Figure 20 -   | Calendar, tasks, and notes handling data structures and files.   | 46          |
| Figure 21 -   | Runtime data structure designed for fast access to message information.  | 47          |
| Figure 22 -   | MS Camcorder.  | 49          |
| Figure 23 -   | Proposed TimeStore main window showing the hierarchical organization of sender names and new icons.                | 68          |
| Figure 24 -   | Proposed message window with additional message navigation controls.   | 71          |



## ACKNOWLEDGMENTS

I wish to thank everyone who has helped me to complete this thesis. In particular, I like to thank the following people and groups:

- Professor Ron Baecker, my thesis supervisor, for providing research directions, for providing many opportunities to demonstrate my work, and for providing feedback at every stage of the thesis, even under severe time constraints.
- NSERC, for funding this research.
- Eric Smith, for his Macintosh programming expertise especially at the beginning when I knew nothing about programming the Macintosh.
- Blair McLeod, for doing a superb job in keeping the lab working, despite everything.
- Byron Long, Joanna McGrenere and Alex Mitchell, for many useful suggestions and help.
- Aleksandra Jovicic, for her help in both programming and testing of TimeStore.
- Leo, Tyrone, Diana, Laureen, Pancy, Howard, and Karen, who have kept me sane throughout the research, for many enlightening conversations about anything and everything, for all their support, and for showing me the light at the end of the tunnel when it was dark.
- Ellen, my mother, for having faith in me from start to finish, and for instilling the value of higher education in me from the very beginning.

# Chapter 1

## Introduction

---

Email overload is a growing problem for many users in the workplace [Fitzmaurice, Baecker, and Moore, 94; Whittaker and Sidner, 96]. Users often have trouble in retrieving messages for use at a later time, or remembering to reply or to act on a particular message. There are two causes for this problem. The first is due in part to the problems associated with maintenance and retrieval in a semantic hierarchical (file and folder) structure and the fact that current email systems only provide functions to create and maintain a semantic hierarchy. The second is that current email systems are designed around the assumption that messages are informational, that they are read upon arrival, and that important messages are filed immediately.

The use of a semantic hierarchy for filing presents many problems when dealing with a large volume of data. Filing data is cognitively intensive but users do not usually notice this because the time spent seems small compared to the time they spend on creating or using the data. Although users preferred to browse through a hierarchy when searching, the maintenance of a hierarchy is time consuming. Maintenance of a semantic hierarchy for a small number of personal files is simple and straightforward. However,

maintaining for an email system is different. Hundreds of new messages can arrive each day, and the user may not have time or want to read all messages. Since the amount of time the user spends on each message can be very small, the filing time becomes a significant portion of the total amount of time the user spends on email.

Furthermore, filing adds no value to the information except that it facilitates efficient retrieval of messages for later use. Since most users keep a majority of messages they may never access again [Silver, 96; Whittaker and Sidner, 96], spending a large amount of time on filing is a waste of time.

Of course, people's email use does not come to a standstill when the number of incoming messages is tremendous. Users have ways to cope with the present volume of data. However, we feel that current email systems do not assist the user as much as they should. Users should be able to retrieve what they need from a large volume of email quickly and effortlessly.

Email systems are being used for task management as well as for communication and filing [Whittaker and Sidner, 96]. The basic assumption that email is a form of asynchronous communication is inaccurate. Research has indicated that email messages are also used for purposes such as document delivery and archiving, work task delegation, storing personal names and addresses, and scheduling appointments.

TimeStore is an email system that uses the time of arrival as the principal arrangement to display electronic mail. Time-based visualization has the potential to complement or replace the traditional email systems. We designed and built TimeStore around the philosophy that the user should not have to do any filing.

## **History of the TimeStore Project**

Our project started with a study on how users organize their computer environment [Fitzmaurice, Baecker, and Moore, 94]. The 14 subjects (Mac, DOS, UNIX, VMS) all used a semantic organization for their filing. File organization was strongly influenced by the visual display of the system. Subjects did use time stamps and date notations were used in naming files and folders. This finding led to the first TimeStore prototype.

Long [94] implemented this prototype to study time-based visualization as an alternative to folders for organizing and retrieving email messages. The original version was an add-on for the Macintosh version of Eudora. Time was represented along the x-axis of a two-dimensional graph, the message senders were listed along the y-axis and messages are displayed as dots.

Subsequently, an Eudora user study and a TimeStore user study [Silver, 96] found that the addition of a time-based system did aid in email retrieval. Users were able to see patterns and trends of correspondence activity and the lack of dots acted as a reminder to contact a specific person or to reply a certain message. As users were reluctant to give up semantic hierarchies, support for Eudora mail folders was added.

In this thesis, we redesigned and reimplemented TimeStore, taking into account many of the problems and suggestions from previous research. We also significantly enhanced its functionality and conducted a usability study of TimeStore with the use of a real-time screen and audio capture tool.

## **Time-Based Visualization vs. Semantic Hierarchies**

Research into dating accuracy in human autobiographical memory indicated that the use of time is a relatively poor retrieval cue [Brewer, 88]. Despite this fact, why use time as the primary method for accessing personal electronic information?

Our goal is to design a system that can file all data automatically while keeping it accessible to the user. In an environment where the filing is not performed by the user, such as in a shared file system [Berlin *et al*, 93], or with a rule-based email filing system [Silver, 96], users often have trouble locating the desired documents. This may be related to the depth of processing [Gleitman, 91] where the user forms retrieval cues due to the mental processing needed to categorize and file the document. When this retrieval cue is no longer present, semantic hierarchies are not as useful in retrieval.

The problem with a semantic hierarchical based automatic filing system is that the user will not have the benefit of forming any retrieval cues about the location of the document. Instead, the user must reconstruct the location using other cues. Consequently, the ease and accuracy of retrieval normally found in a user maintained semantic hierarchy is lost.

Time-based visualization offers the user additional retrieval cues by allowing users to cross reference email messages with personal events. If users are able to recall events accurately, they will be able to use the time of the event as a retrieval cue to find the associated messages. We hypothesize that the automatic filing in a time-based visualization system may overcome the

problems associated with semantic hierarchical based automatic filing systems.

### **Thesis Outline**

The following chapter covers many relevant researches in human memory, information organization, information retrieval, and time-based systems. Chapter 3 gives a detailed description of the TimeStore interface and a comparison with the previous version. Chapter 4 provides a description of the technical implementation in the TimeStore prototype. Methodology and results of the usability study are discussed in detail in chapter 5. Finally, the final chapter concludes with a discussion of the contributions of the thesis, implication of the results and possible future work.

## Chapter 2

# Relevant Research

---

The design of TimeStore is based on research from studies in human memory, information organization and usage, time-based visualization, current commercial email systems, and usability methodology. The following research has helped guide the design of TimeStore.

### **Research in Human Memory**

An effective retrieval system must take advantage of the way human memory operates. The ability to remember is dependent on success in *encoding and retrieval*.

#### ***Encoding***

Theories viewed human memory as two distinct systems: short-term and long-term memories. Information is encoded, and placed into the short-term memory. The encoded information must remain in the short-term memory through a process called *maintenance rehearsal* before entering the long-term memory [Gleitman, 91]. The longer information remains in the short-term memory, the more likely it will be transferred into the long-term memory.

However, some argue that the length of time in short-term memory has little benefit in aiding recall [Craik and Watkins, 73; and Rundus, 77]. One factor that affects recall is the *depth of processing* [Craik and Lockhart, 72]. In one experiment, subjects who used the meaning to encode has a much higher rate of recall than subjects who used shallow encoding such as typeface or colour.

### ***Retrieval***

Recognition of an item is easier than free recall. Factors that affect the ability to recall include how the original information is encoded, how well the information is organized in memory, and how unique information is from one another. The *Encoding Specificity Principle* [Gleitman, 91] states that retrieval is most likely if the context at the time of recall approximates that during the original encoding. Subjects who were presented with a list of words had better recall if they were asked in the same room.

Information that is related to that already in memory has a higher rate of recall. This is called *elaborative rehearsal* and is different from *maintenance rehearsal*. Elaborative rehearsal refers to all mental activities that reorganize of information that enables a more efficient storage in the long-term memory [Gleitman, 91]. This process can also relate information to other information, which enhances the effectiveness of retrieval.

### ***Theories of Forgetting***

Memories tend to decay over time. As new information is recorded in the long-term memory, items that are similar have a poorer chance of recall. This is due to *interference* between similar items [Gleitman, 91]. The information is still held in long-term memory, but is rendered inaccessible. Forgetting can also be caused by a *change in retrieval cues*. Memories that are acquired



at a certain location may experience a decrease in the chance of retrieval when the location has been changed. Therefore, if there are a large number of similar events that occurred in a short period, the user will have a high chance of not remembering any individual event.

### ***Types of Long-term Memory***

In general, there are two types of memory. *Semantic memory* deals with the meanings of words and concepts while *episodic memory* deals events in one's life [Gleitman, 91]. A concept related to episodic memory is *autobiographical memory*, which can be viewed as a superset of episodic memory that includes some aspects of semantic memory for the purpose of storing all events, and facts related to oneself [Conway, 90].

### ***Theories of Autobiographical Memory for Time***

Life is a succession of related events that occur in chronological sequence. Without temporal organization in autobiographical memory, awareness of one's history would be impossible. The following is a list of observations of autobiographical memory [Conway, 90].

1. Recent events are well remembered.
2. Different tasks lead to the sampling of different sets of memories. The focus of the retrieval process depends on the specificity on the retrieval cue.
3. Exact dates are not stored in memory.
4. Distinctive events are remembered, routine events are forgotten. This is consistent with the interference effect.
5. When the date is not known, subjects tend to date events more recently than they actually are (forward telescoping). The error increases with increasing retention period.

Psychological research has suggested various theories for time. The reconstructive theory [Friedman, 93; Larsen, Thompson, and Hansen 96] states that our perception of time in the past is reconstructed from general knowledge about time patterns (i.e., temporal schemata) rather than retrieved [Thompson, Skowronski, Larsen, and Betz, 96]. However, we can learn and remember dates just like any information. Subjects who date diary events reported about 10% of all successful recall of the exact date of an entry [Thompson, Skowronski, and Lee, 88]. On the other hand, research suggests that calendar time is not explicitly represented in memory because dates alone are the worst retrieval cue for recalling events [Brewer, 1988].

| Actions         | %  | Thoughts        | %  |
|-----------------|----|-----------------|----|
| Time            | 25 | Time            | 7  |
| Location        | 33 | Location        | 9  |
| Time + location | 49 | Time + location | 14 |
| Thought         | 58 | Action          | 42 |

Table 1. Recall rates to different type of cues for a person's actions or thought (Conway 1990, based on Brewer 88). For example, using time alone, the percentage of correct recall for actions is 25% while the percentage of correct recall for thoughts is only 7%.

Infrequent location and goal-related actions have a positive effect on recall since they are more distinctive from other memories. Memorability of thought was found associated with high levels of excitement. The retrieval cue is the strongest when the action is coupled with thought (see Table 1).

Studies in dating accuracy suggested that knowledge about cyclic time patterns (such as year, month, etc.) is the major source of information to date events that occurred in the past. Such cyclic schemata is very important for cognition [Friedman, 93]. Subjects in a diary experiment were consistently recalling the day of week accurately at least 60% of the time (see Figure 1) in various studies [Thompson, Skowronski, Larsen, and Betz, 96]. The

accuracy rate for an absolute correct recall was around 45% (see Figure 2). For more distant events (a few years in the past), subjects were able to recall the month correctly in close to 40% of the cases (see Figure 3) [Larsen, Thompson, and Hansen, 96]. It is interesting to note that the day of week was immune to forgetting as separate studies varying in retention time from 1.5 months to 10 months resulted with the same pattern.

Important as cyclic schemata are, human life is linear. Cyclic schemata alone cannot account for the fact that we can recall events in chronological order. Some researchers believe that we can remember certain events with the exact date. Called *landmarks*, such events are the link between personal/subjective and social/objective time.

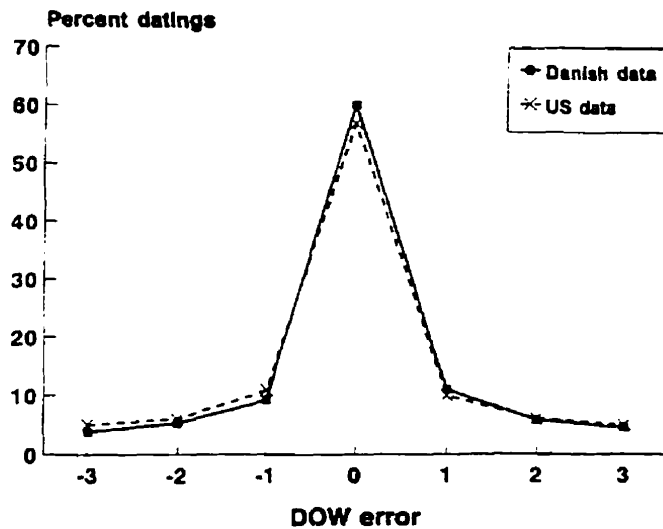


Figure 1 - Day of week (DOW) errors calculated from the data in figure 2. Reproduced from Thompson, Skowronski, Larsen, and Betz, 96. Subjects' recall of the day of week is very accurate to within a day.

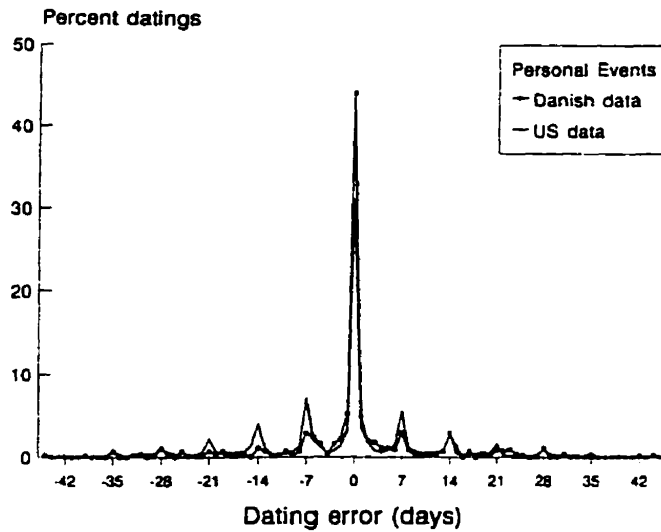


Figure 2 - The difference in days between the remembered and actual dates of personal events, tested 1 week after a 3 to 4 month diary period. Reproduced from Thompson, Skowronski, Larsen, and Betz, 96. Subjects recall of the exact day is not as accurate as the DOW error. Only 45% of the recall were accurate.

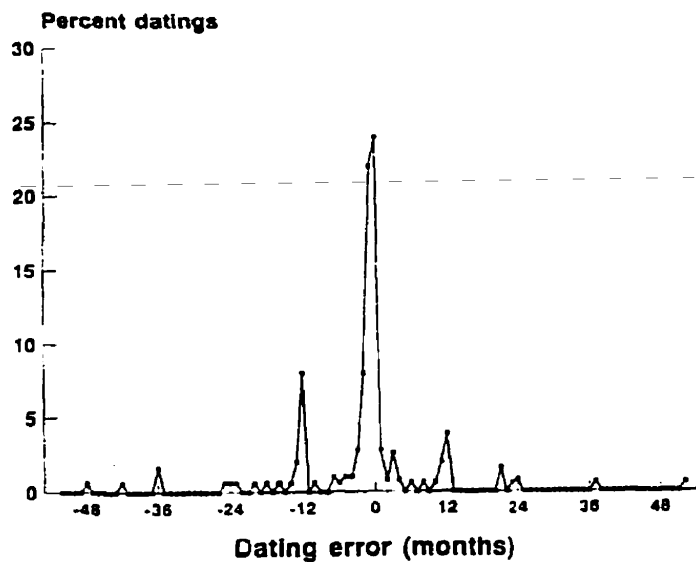


Figure 3 - Deviation of remembered month and year of moving to a new home from the actual time of the move. Reproduced from Thompson, Skowronski, Larsen, and Betz, 96. Subjects remember the month much better than the year.

Barsalou proposed a hierarchical structure that focus on *extended event time lines*. Extended event time lines may run in parallel or overlap and are consist of extended events that may have subparts further down the hierarchy. Schooler [Schooler and Herrmann, 92] suggested that autobiographical memory distinguishes between *periods* (long-term thematic generalizations), *episodes* (sequential narratives of specific events or sets of interrelated events), and *moments* (isolated and often vivid instants in time).

What we can conclude from all the research in human memory is that people's recollection of when an event occurred is not perfect. Therefore, a time-based visualization system must provide ways to cross-reference highly memorable events with other data that may not be as memorable in order to increase the retrieval accuracy. In addition, the system must also expect that the user will not be able to remember the exact date, and therefore may need a few tries to find the data the user needs.

## **Studies of Information Organization and Usage**

Understanding how people organize their electronic data, especially their email messages, is vital in designing a system that organizes data automatically. The following research has helped guide the redesign of TimeStore.

An early study into how people organize their office environment [Malone, 83] was the first to point out the importance of both filing and reminding functions. Organizations ranged from being very messy to very neat. The most important organizational units were files and piles. The main distinction is that files are labeled and arranged in some systematic order

while piles are not intentionally arranged in any order. However, piles are often semi-organized in an inverse chronological order. Subjects also noted difficulty in classifying information for filing and the difficulty often leads to vaguely classified piles on top of desks and tables. It is interesting to note that Malone suggested using time of creation to classify information automatically.

Research on group memory in the TeamInfo prototype [Berlin, Jeffries, O'Day, and Paepcke. 93] illustrated the difficulty in finding a set of category labels that everyone in the group will agree. Berlin described his users as being different along five different dimensions:

1. *Purists* and *proliferators*: Purists store things in one place and are willing to search multiple places for them. Proliferators want to be able to find related items in one place and they are willing to have virtual copies all over the place.
2. *Semanticists* and *syntacticists*: Syntacticists prefer to use episodic clues for retrieval while semanticists prefer to use semantic clues. Episodic clues are acquired from particular events (episodes) from one's own life while semantic clues concern more with the meanings of words and concepts.
3. *Scruffies* and *neatniks*: Scruffies want to minimize the up-front cognitive load of organizing information while neatniks prefer to use fine-grain hierarchical organization to minimize the effort required at search time.
4. *Savers* and *deleters*: Savers keep things around even the subject may not be directly related. Deleters want to delete all items that seem unrelated at the moment.
5. The expected purpose for which the item is saved: The role of the person and his expected future tasks dictate in which category the person files things.

Barreau and Nardi [Barreau and Nardi, 95, Nardi, Anderson, and Erickson, 95] conducted separate studies on file organization among DOS, Windows, OS/2 and Macintosh users. There were many similarities among PC (DOS, Windows, and OS/2) users and Mac users:

1. Preference for searching for files by location. Users preferred to browse lists of files rather than to remember the exact file name or to use text searching. Barreau and Nardi hypothesized that the preference is due to a greater sense of control. The task also engages the mind and body to a greater extent. They also noted that users are more willing to use logic based search techniques when accessing remote machines that are not organized by the user.
2. Location of files served as a reminding function. Users are more likely to place files at locations that they will see often to remind the users to do certain tasks. This is one of the reasons that users prefer location-based filing.
3. Use of ephemeral, working, and archived information. *Ephemeral* information has a very short life and includes things like (some) email, "to do" lists, news articles downloaded from databases, etc. Mac users tend to keep such information on the desktop or in the top directory. This can be a problem if the user has to deal with a large quantity of ephemeral information. *Working* information is used more frequently and usually is related to the users current work needs. Users have no trouble finding such information; they can remember the spatial location. *Archive* information is often only indirectly related to the user's current work. Users reported difficulties in selecting files to place in the archive.
4. Archived files are less important to regular users. The study of archived information has been emphasized since researchers usually archive more information than regular users.

Barreau and Nardi felt that the study of ephemeral information would help regular users deal with the potentially large quantity of ephemeral documents. They also felt archived information can be handled better through tools that were designed with such a purpose in mind.

Fertig, Freeman, and Gelernter [96] support the position of Barreau and Nardi that current systems do not handle ephemeral information well and users prefer to browse a list of files rather than use a text search tool. However, they argued that users preferred location based searching over logical searching using a search tool because of inadequate tools, not because location based searching actively engages the mind. They also argue that users use locations for reminding simply as a coping mechanism. Location based storage are more useful for a small collection, and do not scale well to a large collection of information. The fact that users do not make use of old archived information is because it was not convenient to store and access older information.

Research into how people use email has identified that email systems are used for task management as well as for communication and filing [Whittaker and Sidner, 96]. Task management demands that information relating to current tasks is readily available. Users must be reminded about when to finish their tasks.

Email that tends to be left in the inbox can be categorized into 4 basic types:

1. *To dos* are messages, which assigns a task for the user. Sometimes a task can span over many days and users usually leave such messages in the inbox to remind them for the uncompleted tasks.
2. *To reads* are long informational documents that do not require a reply. Users may not have time to read such messages, and so they are left in the inbox, to remind the user.
3. *Messages of indeterminate status* are informational messages whose significance cannot be determined when it first arrives. The significance of such messages may depend on events after the arrival or subsequent messages. Therefore, the user leaves them in the inbox.



4. *Ongoing correspondences* are messages that are part of incomplete, ongoing threads of conversations. The user may leave such messages in the inbox because they are unable to reply immediately. They may be waiting for messages from other people.

In terms of filing habits, this study confirms what others have found. The filing strategies of users can be categorized into three types. *No filers* do not file messages away. *Frequent filers* usually file messages as they arrive and had a more effective folder structure. Finally *spring cleaners* only file messages sporadically and had a less effective folder structure.

Shneiderman and Plaisant [94] propose that organization be “role-centered” that places the emphasis on the users’ tasks rather than “document-centered”. In their proposed Personal Role Manager, tools should be designed to support users in managing their multiple roles in an organization. Each role includes a vision statement that reminds users of their goals, a list of people which the user interacts with in this role, a task hierarchy, and a schedule.

## **Information Retrieval**

Understanding how people retrieve their electronic data was very important in the redesign of TimeStore.

Erickson and Saloman [91] discuss some of the interface issues of a desktop information retrieval system. They were able to use DowQuest, which is a full-text retrieval system that gives users access to six months of news in the Dow Jones News Service. They found that users had high expectations of the intelligence in the retrieval engine. When the system returns irrelevant articles, the users assumed the system was no good, or no relevant files

existed. They also found that by shielding the user from complex query languages, the user actually lost control of the search in some cases.

Users had two types of questions: *ad hoc* queries for answers to a specific question or *on-going* queries to keep up to date on a particular topic. In dealing with dynamic information such as the Dow Jones News Services, they found that problems arise when the user executes the same query on different dates where the results of the same query will be different. *On-going* queries present more problems because they will generate an increasing number of results over time. These queries must be made more specific. Also, terminology may change over time and the queries must be updated to be effective. Erickson and Saloman called this a mismatch: a static query cannot remain effective in a dynamic database.

The INFOSCOPE [Fischer and Stevens, 91] project looked at the information overload problem with Usenet News. There were roughly 400 newsgroups in 1991. That number has since grown to about 12000. The flow of information consists of four stages:

1. *Send time processing* is the work done to send a message.
2. *Read time processing* is the work done to search for the information of interest.
3. *Storage time* is the time when users save the message according to some retrieval semantics. The range of granularity can be from saving to a flat file, to coarse-grained (a folder structure), to some other fine grained where files can be retrieved using rules.
4. *Question time* is the time it takes a user to retrieve previously stored information.

Fischer and Stevens based their research on the hypothesis that senders will not expend the effort to classify messages but readers will spend limited effort to organize since there is a direct benefit to them. The flow of information can be analyzed using two models. The *situation model* is used to analyze the users' needs in terms of their current task and context, while the *system model* is used to analyze the organization of the information space presented to the user. The task of mapping the user's personal situation model to the global system model consumes cognitive resources.

Programmers must navigate through large amount of information (code, program output) and most do it with ease. Altmann, Larkin, and John [95] suggested a cognitive computation model based on the study of a skilled programmer. The model has three components: knowledge, an encoding mechanism provided by the supporting cognitive architecture, and mechanisms for memory retrieval. Knowledge is divided into three categories: expert, external, and situation. Expert knowledge includes expertise on the task. External knowledge rests in the display, combined with internal knowledge to extend the user's effective memory. Users acquire situation knowledge when they work in an environment and their subsequent actions in navigation often depend on what they saw earlier.

Their findings were the following: Situation is encoded in long term memory automatically as a byproduct of problem solving. Activating such memories requires recreating a working memory context similar to the one in which they were encoded. When a memory was encoded in a situation other than the current, this requires search to find the right context. This approach to memory management, with most of the effort for retrieval occurring at retrieval time, has been used in other models that learn and use recognition

knowledge in situated tasks. Deliberate search of long term memory is a fallback strategy. Such a search uses semantic cues to recreate goal conditions and image cues to recreate display conditions that can trigger memories of previous situations that may lead to navigation.

## **Time-Based Information Systems**

In addition to TimeStore, there are two other projects that we know of that uses time to organize data automatically. In this section, we present a short description for both the Lifestreams and the LifeLines projects.

The Lifestreams project [Carriero, Fertig, Freeman, and Gelernter, 95; Fertig and Freeman, 95; Steinberg, 97] completely rethinks the way we deal with electronic information (see Figure 4). A Lifestream is a collection of information organized into a stream of chunks. Each Lifestream is used to store a particular type of information. Data units (chunks) in the stream are arranged chronologically, but the user can choose another viewing method if needed.

Lifestreams is designed based on five observations:

1. "Names" and "Directories" should be junked. Users should only have to name objects whenever they feel like doing so. Most documents have names that are meaningless for retrieval purposes and naming such data is a waste of the user's time. Directories or folders at the present are too static.
2. The system should provide logic for finding a chunk or group of chunks.
3. The system should provide logic for summarizing or compressing a large group of related chunks to the user a concise overview.

4. "Compatibility" should be automatic. Instead of using the computer as a device for storage and managing information, the computer should be used as a viewing device, like a television. A Lifestream can be accessed from any available "viewport"
5. Explicit file-storage management should be junked.

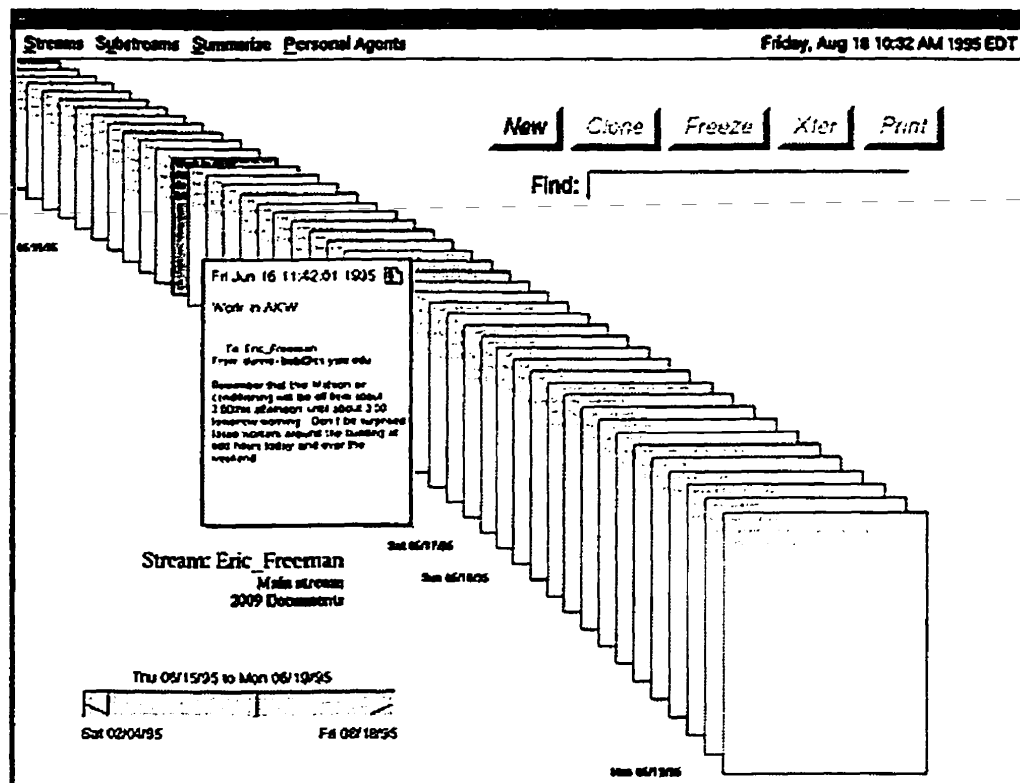


Figure 4 - Lifestreams user interface. Documents are arranged in a "stream" that is organized by time. The slider is used for fast scrolling through the stream. Reproduced from Carriero, Fertig, Freeman and Gelernter [96].

The LifeLines project [Plaisant, Milash, Rose and Widoff, 96] looks at how personal histories, youth records (see Figure 5) and medical records in particular (see Figure 6) can be effectively visualized. It uses graphical time scales with horizontal lines representing data that is more or less continuous

such as legal check ups. Icons are used to represent discrete (less frequent) events. Other line attributes such as thickness and colour are used to represent relationships or to highlight significant events.

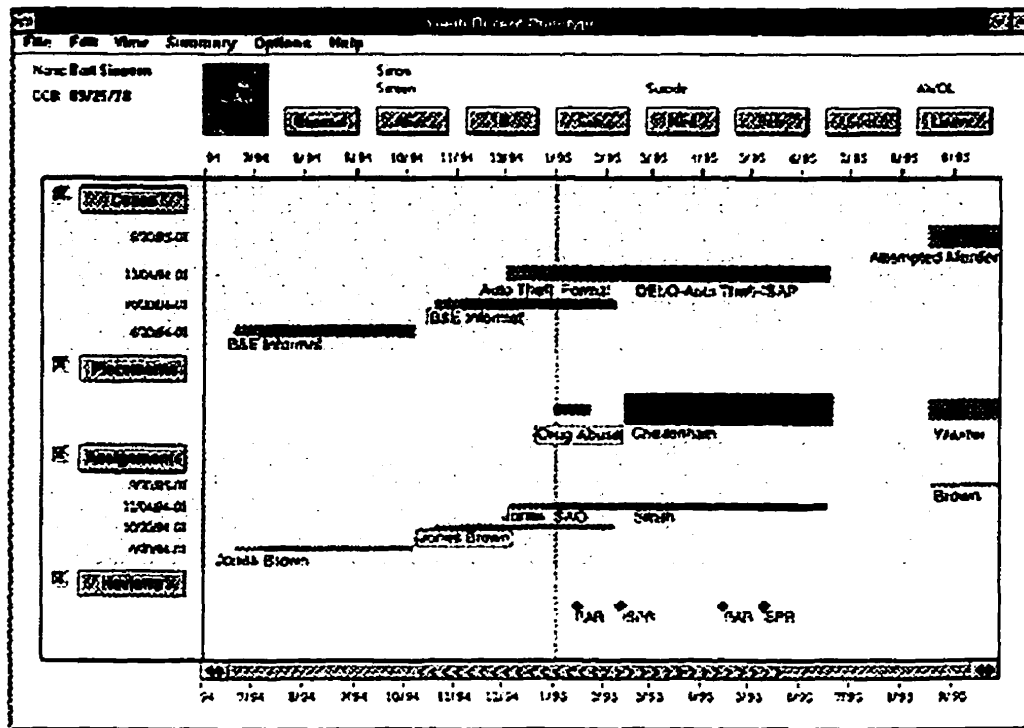


Figure 5 - Lifelines showing juvenile records. This record shows cases, placements, assigned workers and reviews (y-axis). Notice the overview enables the user to zoom into any one part. Reproduced from Plaisant, Milash, Rose and Widoff [96].

LifeLines addresses the problem of information overload by using an overview. Plaisant did not use scroll bars in the initial overview because the entire image may not be viewed and the user may be unaware that a significant part of the data exists. Completeness must have priority over detail. Once the user has recognize a portion of data that he must work with, he can expand that part. LifeLines accomplishes this objective through the use of silhouettes and shadows. LifeLines can organize complex data into a

hierarchy. The user can expand, contract or directly manipulate the headings as required.

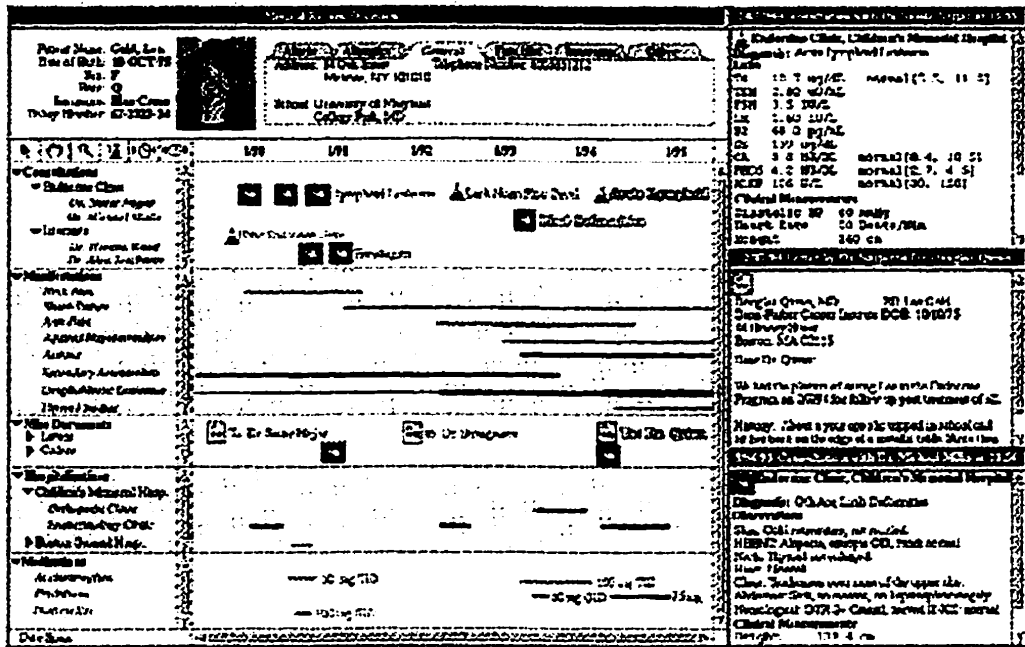


Figure 6 - Lifelines showing medical records. In this application, the line thickness is used to show the severity and dosage. A portion of the complex hierarchy can be collapsed to provide an overview. Reproduced from Plaisant, Milash, Rose and Widoff [96].

Plaisant also did a user study on a prototype of LifeLines. The features missing were:

1. The ability to show future events, such as scheduled events.
2. The display of exact dates.
3. The marking of informal groups of related cases or events.
4. The integration with data entry techniques.

## Commercial Email Systems

In this section, we present three of the most popular email systems on the market today. All of the users in the usability study use one of these systems.

Eudora (see Figure 7) was one of the first email clients available for either the PC or Mac [Qualcomm 97]. It organizes email messages in folders, and the user can specify “filters” where the system will automatically move a message into a specific folder. Each folder is displayed as a list of messages. A click on one of the column labels will cause Eudora to sort the list using that field. The user can open a message by double clicking on a message.

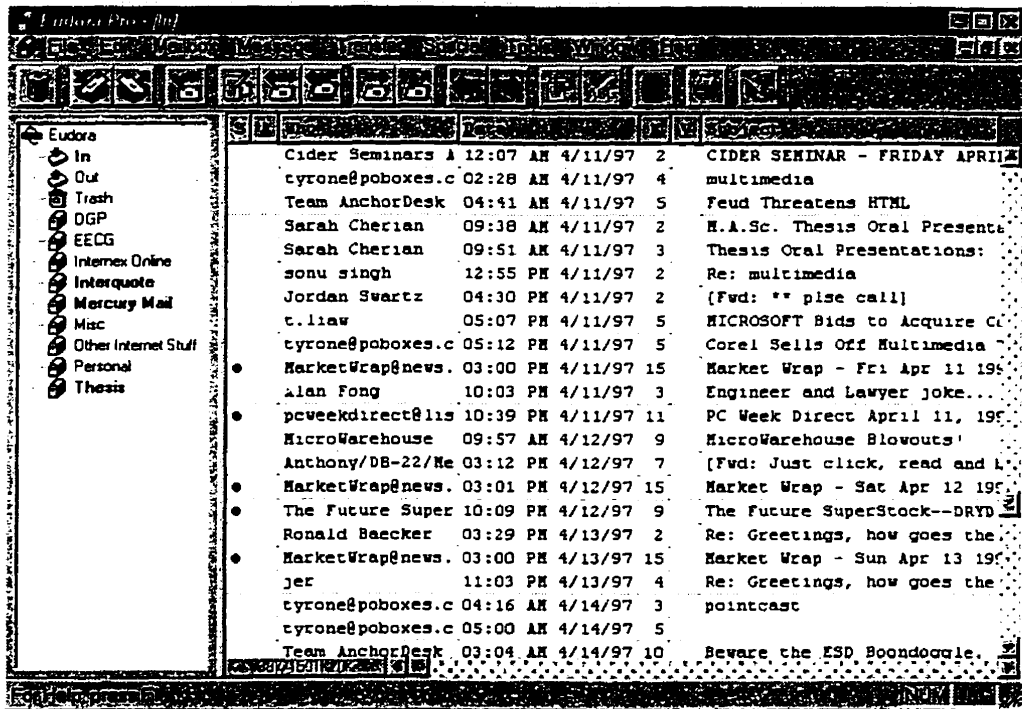


Figure 7 - Eudora Pro 3.0.1. Eudora allows the user to maintain messages in a hierarchical folder structure. Filters are used to automatically file messages into specific folders. In this example, the content of the inbox is shown.



Microsoft Outlook (see Figure 8) is the first commercial system that combines email, task management, contact management, and other personal information into one package [Microsoft 97]. Outlook was released eight months after the first TimeStore 3.0 prototype was built. Like Eudora, it uses a hierarchical folder structure to organize email messages. The most distinctive feature of Outlook is the AutoPreview feature where the first three lines of the body of the message are displayed for unread messages. Outlook also has a feature similar to Eudora's filters where messages can be filed automatically.

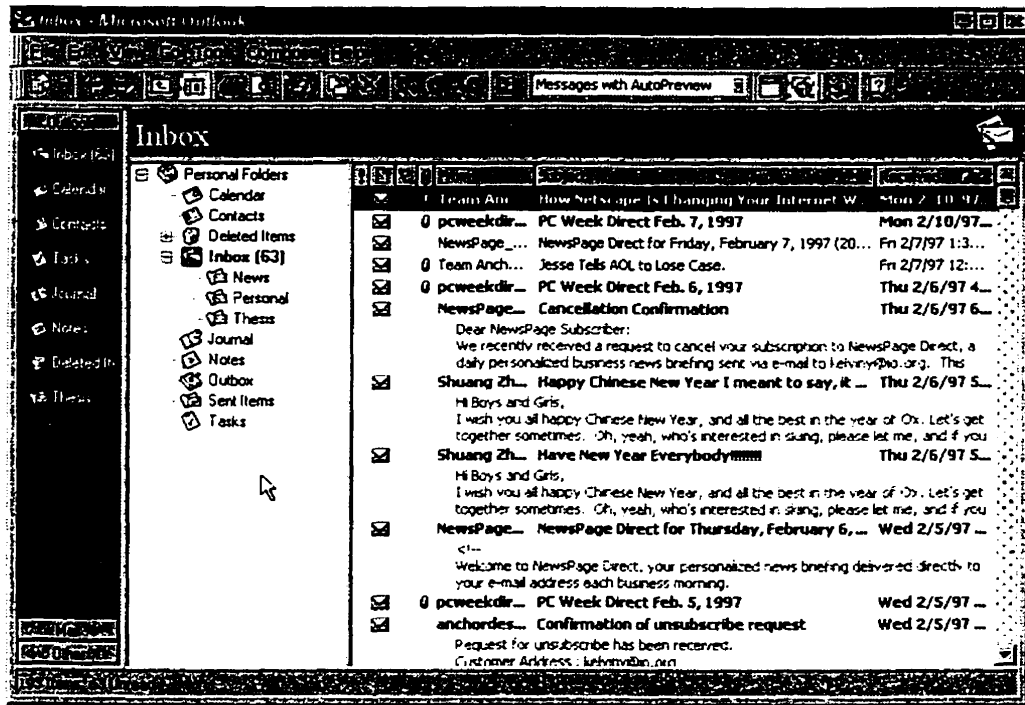


Figure 8 - Microsoft Outlook. Similar to Eudora, messages are organized into folders. The content of each folder is displayed in a list. The bodies of unread messages are shown in the same window of the header so that the user would not have to open the message.

Outlook's Journal feature (see Figure 9) uses a timeline to display email messages, faxes, and other Office 97 documents. Each document is represented by an icon. If there are multiple documents created on the same day, Outlook will display all of them in individual icons listed in the same column. Unfortunately, there is no way to specify how the y-axis is to be used in the display. Outlook does allow the user to group documents of the same type into collapsible sections.

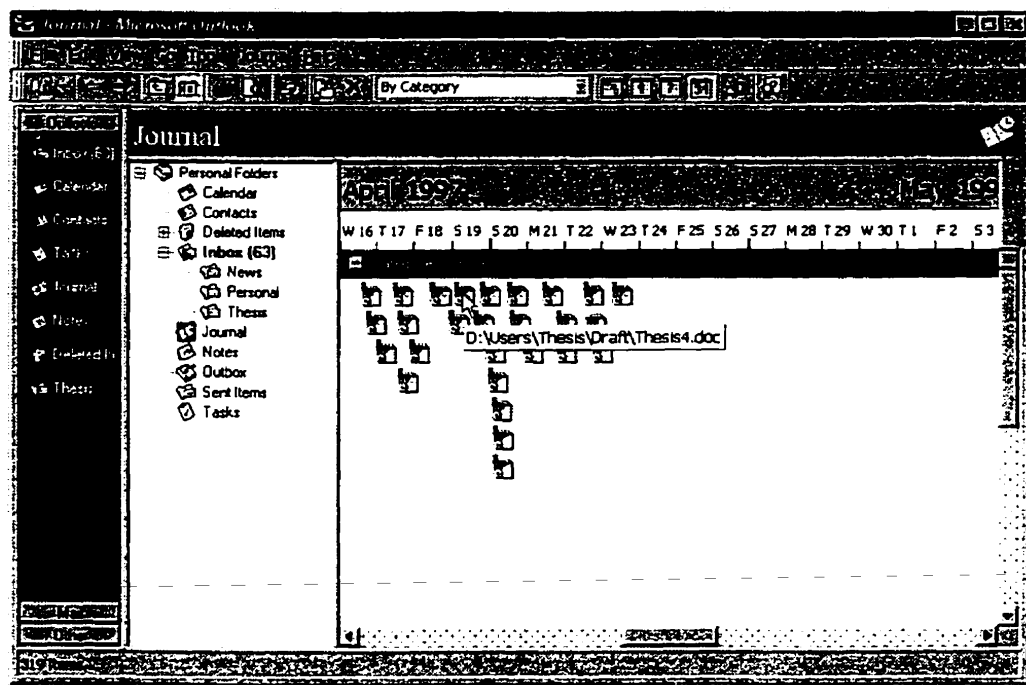


Figure 9 - Outlook's Journal feature where the user can display all Office documents, faxes, email, etc in a timeline format.

Netscape Navigator [Netscape 96] is the most popular browser for the World Wide Web. The built-in email client (see Figure 10) is different from the others. Navigator also allows messages to be organized into a hierarchical folder structure. However, it does not provide an automatic filing feature like

the others. One area in which Navigator excels is in allowing users to browse through their messages. A single click on the message's header will bring up the message in the bottom part of the window.

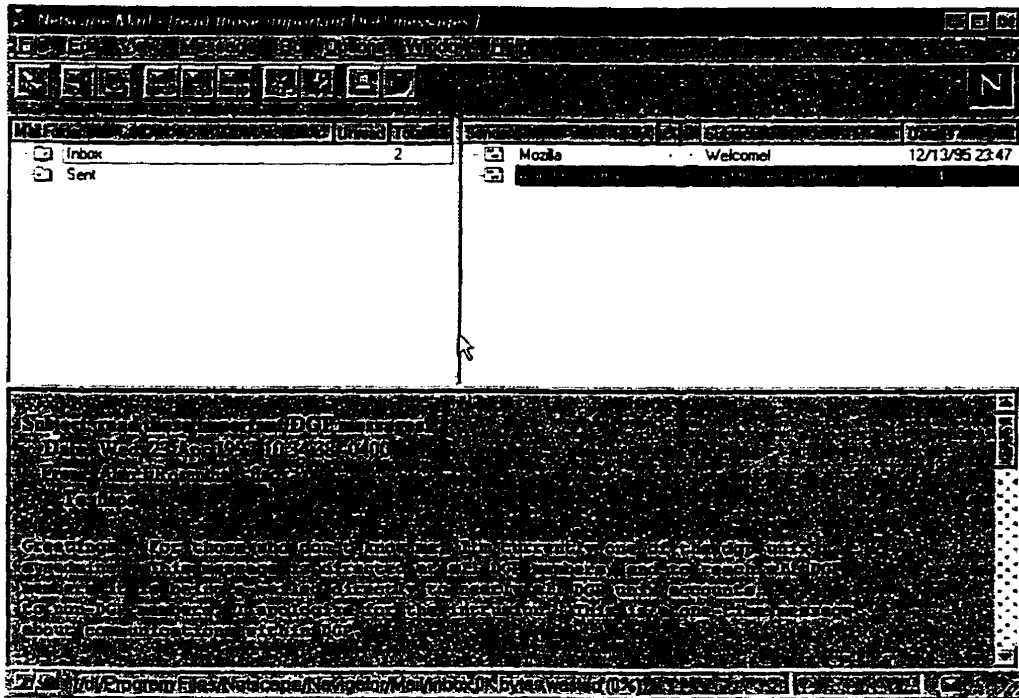


Figure 10 - Netscape Navigator 3.01's built-in email client. The user can open a message by single clicking on a message. This is more efficient than the others but limits the number simultaneous message windows to 1 unless the user opens a separate message window by using a button on the toolbar.

## Usability Testing Methodologies

Usability studies can be categorized into two basic types. One type consists of studies that are performed in a controlled environment where subjects are asked to perform pre-defined tasks. The other evaluates users in their normal work environment and data is gathered using qualitative interviews and

questionnaire. None have tried to observe users using software prototypes for long duration.

Evaluating software in the users' natural environment requires new methods and tools [Posner, Baecker, and Mitchell, 97]. In order to gather data about how actual system usage, the experimenter must ideally be omnipresent, able to reconstruct all details about user actions and system responses, and unobtrusive. This is impossible, but the authors have a number of suggestions that can be used to form the foundation of a new methodology.

1. Build technology with data analysis in mind.
2. Iteratively design the study. Run pilot tests and pretests on "users" as many times as needed and adjust the study as necessary and promptly.
3. Collect redundant data without overloading the user using a variety of methods such as questionnaire, think-aloud protocol, interviews, etc.
4. Save history of software interactions.
5. Use video recording selectively.
6. Record surrounding events in context, such as record think-aloud audio in synchronization.
7. Begin analysis immediately.
8. Use visualization tools for improved analysis.
9. Consider the Internet as an evaluation tool so that users in remote locations can participate in the study.

In the user study of the previous version of TimeStore, Silver [Silver, 96] used a combination of program instrumentation and think-aloud protocol recorded using a tape recorder in an attempt to study real software usage. The program instrumentation consists of recording all user interactions to a

log file. A copy of the user's Eudora TOC (table of contents) files was saved with the log file. The experimenter could then listen to the audio tape and recreate the user's actions with TimeStore. The combination of program instrumentation and think-aloud protocol was very helpful in analyzing users' usage patterns. To improve the methodology, Silver suggested creating a small program that could use the data in the log files and drive TimeStore. Another suggestion was to add digital sound recording capability to TimeStore so that the audio clips could be synchronized with the log playback.

## **Relevance for TimeStore**

Research into information organization such as work done by Barreau and Nardi [95] and Whittaker and Sidner [96] showed that current information systems do not provide adequate support for ephemeral information. TimeStore attempts to address these issues by allowing the user to track their calendar and tasks with their email messages. Support for ephemeral information is provided by the notes feature and the large description field in each task.

In the research of Erickson and Saloman [91], they concluded that users had very high expectations in the intelligence of a full-text search engine. Users are quick to conclude a system is no good if irrelevant results are returned by the system. TimeStore did not directly address the problem with search engines. Rather, TimeStore displays the search result in a timeline, and the user can use other information in the display, such as the date or sender name, to help them locate the message they need.

The use of a timeline allows users to use their autobiographical memory to help them locate events and messages in time. Understanding how human memory uses time as a retrieval cue is vital to an effective design of a time-based visualization system.

Other time-based information systems such as Lifestreams [Carriero, Fertig, Freeman and Gelernter, 95] and LifeLines [Plaisant *et al*, 96] offered insights into how others design time-based information systems. TimeStore draws from their strengths and applies them to the problem of personal information and email.

## Chapter 3

# TimeStore User Interface Design

---

This chapter describes the interaction, display, and major features of TimeStore version 3.0, followed by a comparison with the previous version of TimeStore.

### **The New Interface**

TimeStore 3.0 manages email but also has an additional objective: to provide integral support for task management as well as other personal information where time is the primary method for access.

TimeStore plots information as dots on a two-dimensional graph (see Figure 11). Time is represented on the x-axis and is shown at the top of the window. The timeline is continuous and the user can choose the beginning of the time period to view. Navigation in the timeline is achieved by using the arrow buttons beside the timeline or by dragging the timeline directly. The user can choose between viewing the timeline by month or by day.

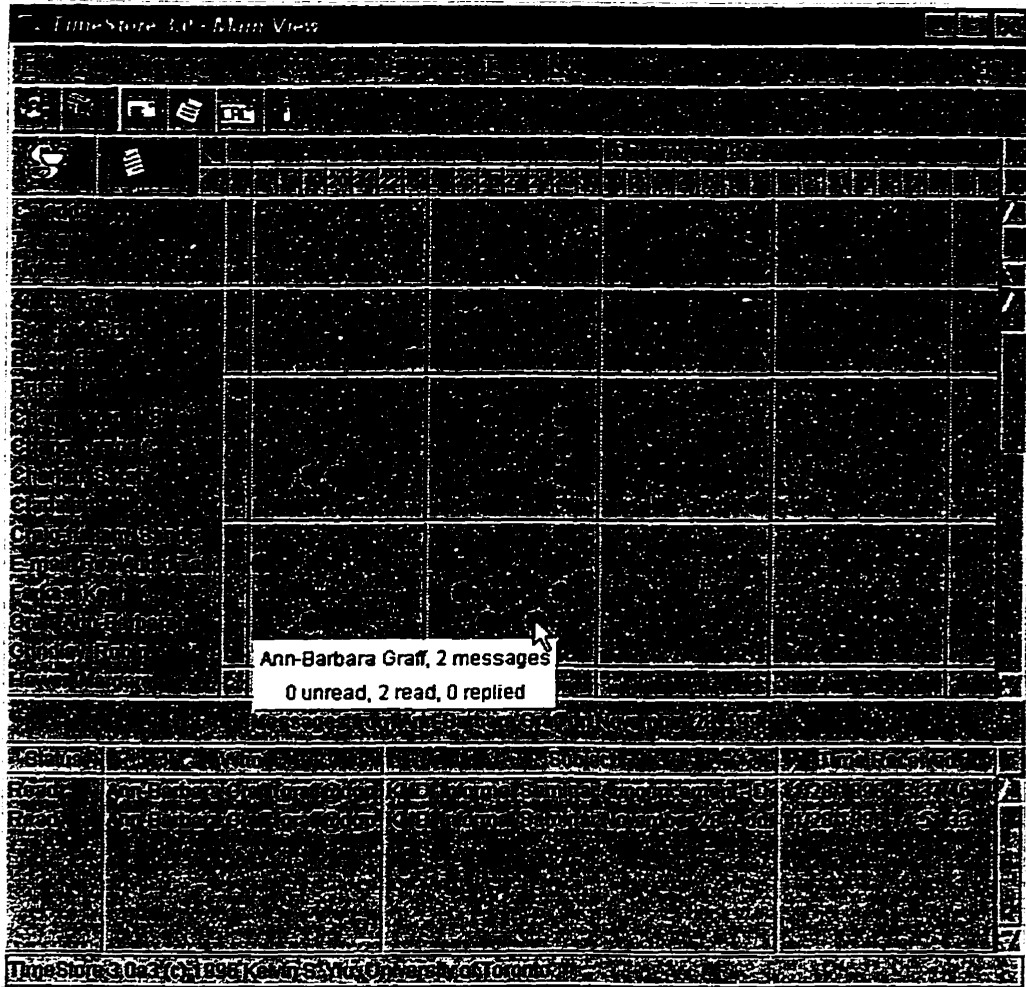


Figure 11 - TimeStore main window.

The display is split into three areas. The top portion displays a calendar, a task list and a notes list. The middle section displays the messages as dots with the sender names listed the y-axis. The bottom section lists the information in detail when the user clicks on a dot. Positioning the mouse pointer over a dot will display a balloon showing the number of read, unread, and replied messages. All unread messages are displayed as red dots in the



main display while messages that have been read or replied are shown in blue.

TimeStore's approach to the inbox is different from most email systems. The inbox is one view into the mailbox where only recent and unread messages are listed (see Figure 12). The content of the inbox is generated dynamically. Messages that have been read automatically disappear into the main display and will not be included in the inbox the next day.

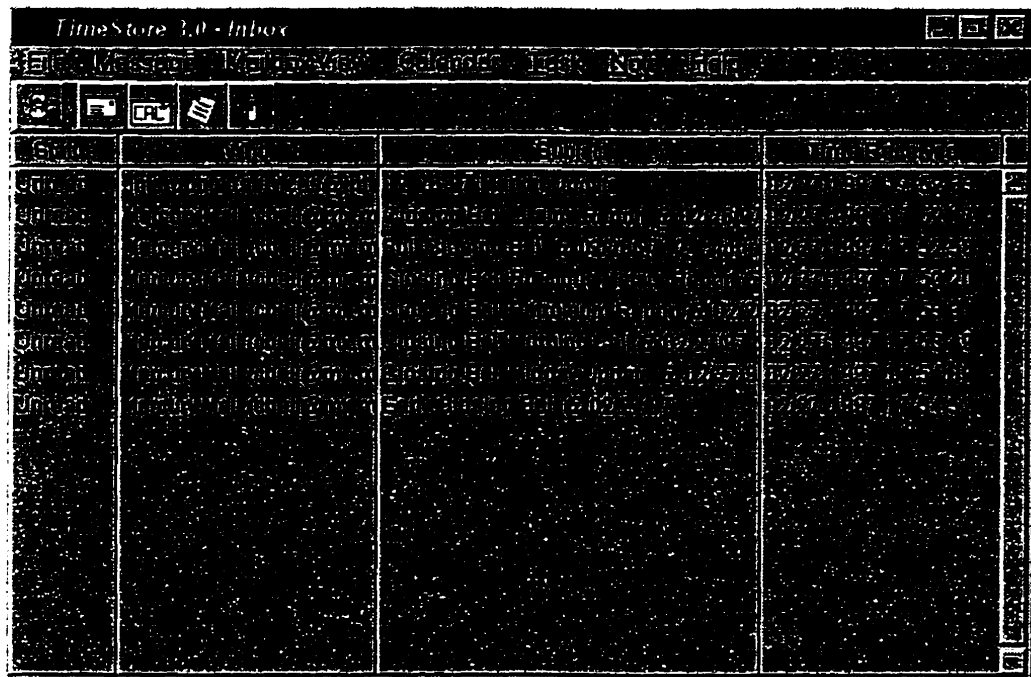


Figure 12 - TimeStore inbox window. Only new and recent messages are shown here.

The calendar is organized by date (see Figure 13). The user can use this to record important dates such as holidays, special events, etc. The description area acts as a notepad for the user to write down other information related to the event. The user can access information in the calendar by clicking on a

dot on the calendar line and double clicking on any one event listed at the bottom of the main window.

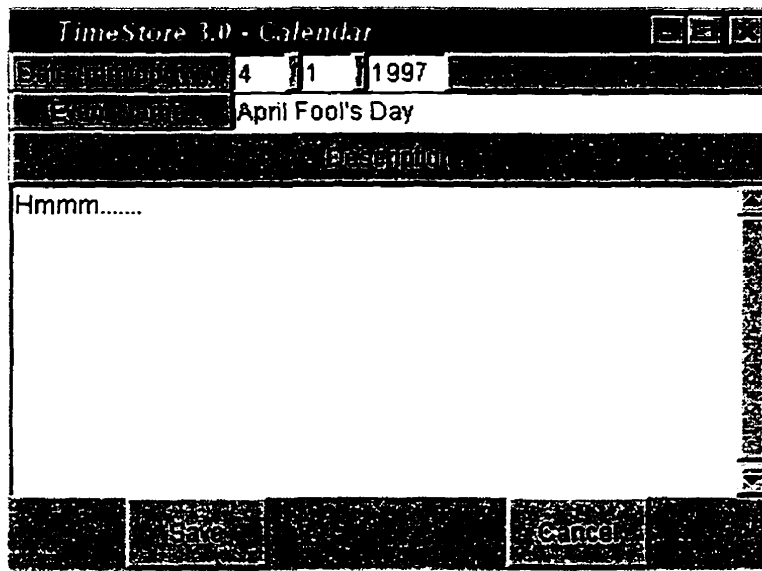


Figure 13 - Create new calendar event dialog box.

TimeStore provides integrated task management by allowing the user to create tasks from within the message window, inbox, and main window. Tasks created this way are associated with a message and the user can record additional information about this task that was not part of the message. The User can click on the "View Mail" button to read the associated message.

The tasks are organized by due date (see Figure 14). The interaction with tasks is same as with the calendar. The main difference between tasks and calendar technically is a status field and incomplete tasks are listed in the taskbox window (see Figure 15). However, we have separated the two because personal events and the things a person has to do are different.

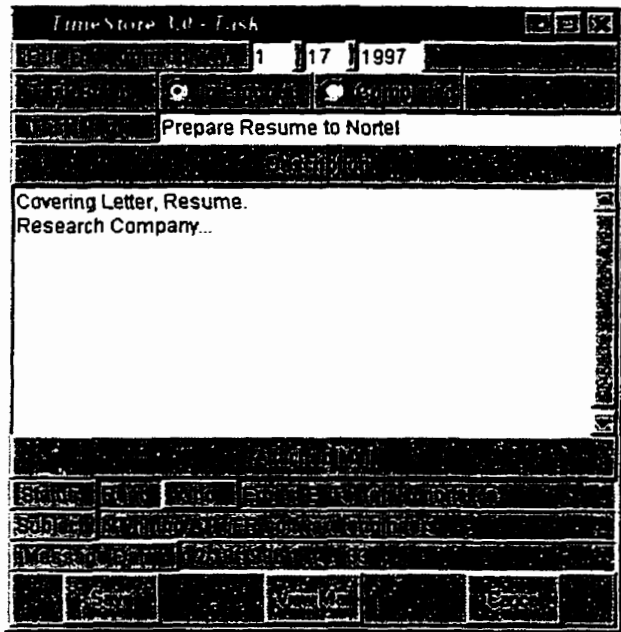


Figure 14 - Create task dialog box that associates the task with an email message.

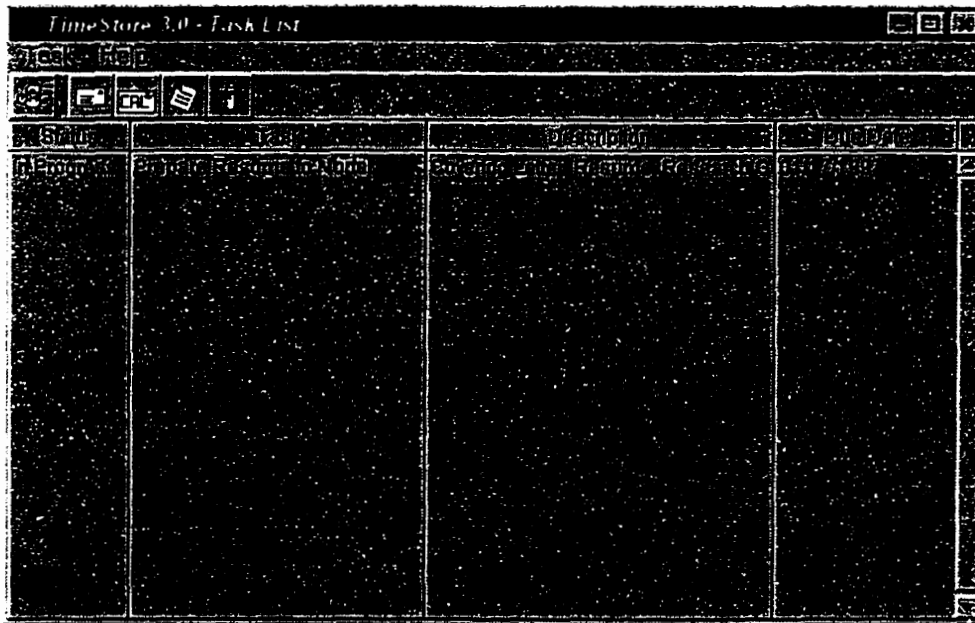


Figure 15 - TimeStore taskbox window.

In addition to creating a task for reminding purposes, TimeStore also allows the user to mark a message. By marking a message, the dot representing that message will appear in the display in red, rather than blue. We feel that the red dot, combined with the name of the sender and the time is enough to remind the user about a particular message. This feature is useful in handling *to read* messages. Also, giving the user the ability to associate messages with tasks provides another opportunity to form a retrieval cue for the message based on an action, which improves the general accuracy in dating messages [Brewer, 88].

Ephemeral information [Barreau and Nardi, 95] that cannot be classified as a task or event can be recorded using TimeStore's notes feature. The note list is organized by time of creation. This organization is adequate since the dating accuracy for short-term information is very good. Also, the user do not have to name the note since naming can be cognitively intensive and poorly constructed names may not aid the retrieval process.

Address book information can be accessed by double clicking a name (see Figure 16). TimeStore automatically creates a record for each message sender when the user decides to enter any address book information. This is in contrast with other email systems where address book entries must be explicitly created. Names can also be merged together by dragging one name to another. This is useful because messages from senders who have many email address can appear under a single name.

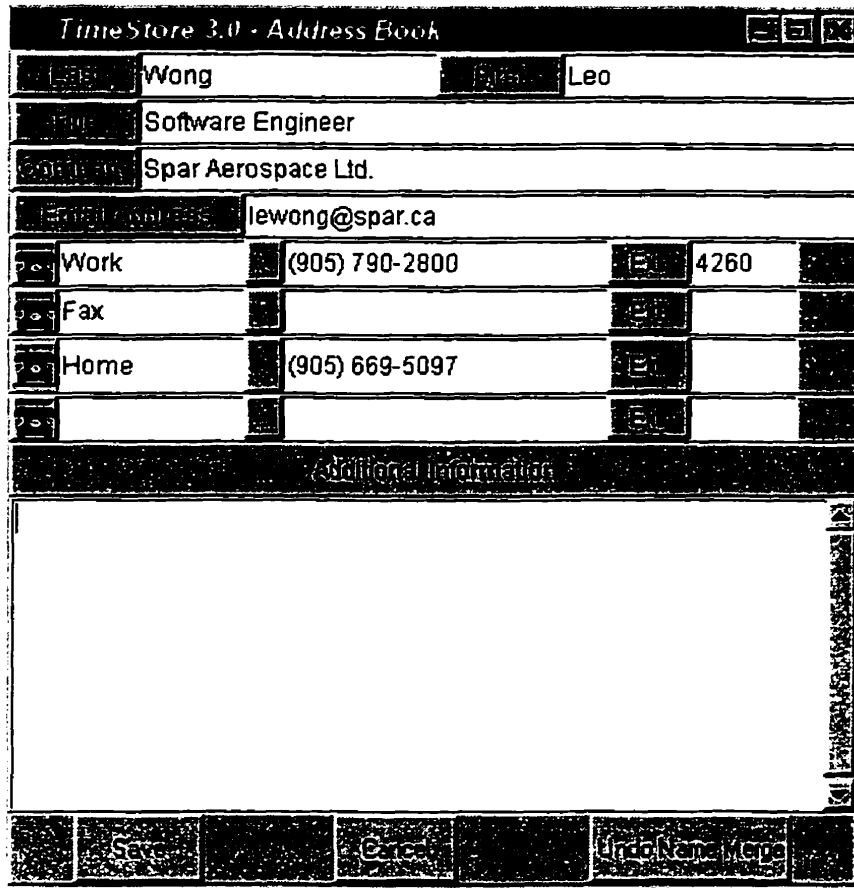


Figure 16 - TimeStore address dialog box. A double click on any name will bring up the address book entry for that sender.

In place of the traditional folders, TimeStore provides a search tool for creating dynamic mail folders. Similar to the substream concept found in the Lifestreams project [Fertig, Freeman, and Gelemter, 96], users can specify keywords to search for in certain fields in the message (see Figure 17) to create mailbox views. The result of such a search is displayed in the same form as the entire mailbox. Moreover, new messages that satisfy the specified criteria will be included automatically, and messages can appear in

multiple views which eliminate the problem that a message can only appear in one folder.

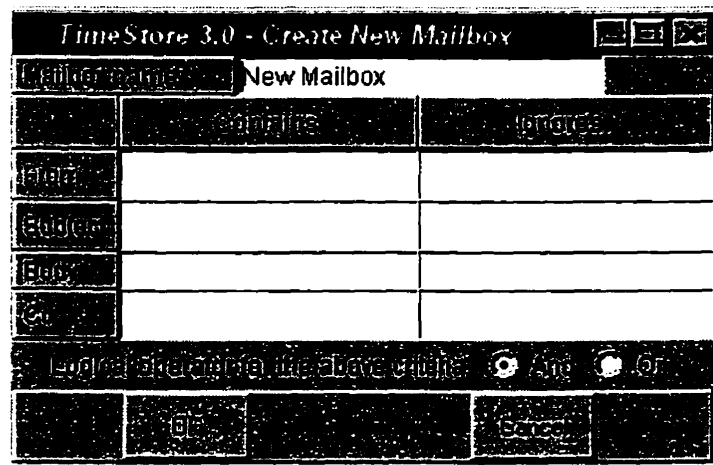


Figure 17 - Create mailbox dialog box. The layout is very simple for quick searches.

### **A Comparison with the Previous TimeStore Interface**

The original TimeStore user interface was developed for an initial study in time-based visualization as an alternative to using a semantic hierarchy for filing and retrieval (see Figure 18). Email was a convenient place to start. Although TimeStore 2.0 was useful in helping users in retrieving older email messages, the original design did not study in sufficient detail how users use email and the interface itself has a number of interaction problems. The following is a list of the problems in the previous version of TimeStore and their resolution in the current version.

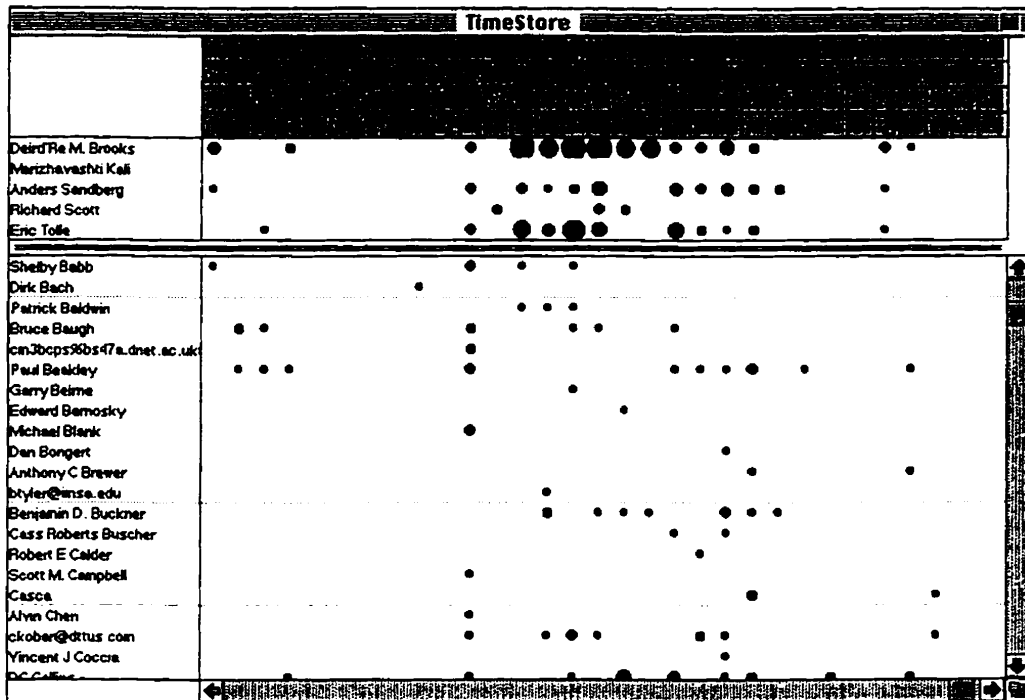


Figure 18 – TimeStore 2.0. The size of the dots represents the number of messages. The section below the timeline is the "most frequent list" of senders.

1. TimeStore 2.0 seems to work well for more distant messages, but not as well for recent messages.

The problem with using a timeline to locate recent messages is that the number of recent messages is relatively small and the user have a very good idea of the location of the message is within a list. The timeline view actually hides information from the user. Therefore, TimeStore 3.0 uses a separate inbox window to show recent and unread messages.

2. TimeStore 2.0 provides no hierarchical organization structure. Currently, user can maintain multiple mailboxes using Eudora Pro - a commercial email package that TimeStore 2.0 depends on for sending and retrieving email. TimeStore 2.0 only provide separate views for the individual mailboxes. Users commented that they would like the ability to create views.

TimeStore 3.0 tries to address this problem by using the search engine to create mailbox views. However, TimeStore 3.0 does not support a hierarchical structure for the mailbox views and does not provide a way to manually include a message into a particular view.

3. One user had trouble remembering when a particular message arrived. He preferred to search for messages based on the position of the message in a list in an Eudora mailbox window.

The problem is related to the fact that autobiographical memory is not 100% accurate and users may not be able to recall when a particular message has arrived. However, TimeStore 3.0 tries to address this problem by using a small list at the bottom of the main window to display messages. If a user is unable to remember exactly when a message arrived but can remember whom the sender was, the user can click on a succession of dots and search for the messages.

4. The ability to show whether a message has been read or not is missing in TimeStore 2.0.

This has been resolved by including the status of the message in the main window and in the inbox.

5. Users reported that the empty areas in TimeStore 2.0 between dots were ambiguous.

This is partially due to the use of different sizes of dots to represent different number of messages. TimeStore 3.0 uses a fixed size dot and the number of messages along with other information is given in a balloon.

6. TimeStore 2.0 lacks visual feedback from clicking on names and dots.

This has been addressed by drawing a black outline around a dot when the mouse cursor is inside the dot and changing the colour of the dot to black when the user clicks on the dot.

7. Changes in the list of names when the user scrolled through time caused some confusion. TimeStore 2.0 only listed names that had sent mail in a particular time period. This is a problem if the user is viewing a time period to search for a person that is not listed. Unlisted names are also used by users as a reminder to get in contact with someone.



The current system does not change the name list when the user moves around in time. However, the user has an option to block out names that they have not contacted recently.

8. The size of the dots in TimeStore 2.0 does not convey a clear meaning as to how many messages it represents.

This information in TimeStore 3.0 is now presented by showing a balloon when the mouse cursor is inside a dot.

9. When the user clicks on a dot in TimeStore 2.0, all of the messages that the dots represent are displayed. This annoyed some users especially if there are a large number of messages.

When the user clicks on a dot, TimeStore 3.0 will only list those messages at the bottom of the main window. The user can open each individual message.

10. In TimeStore 2.0, changing a person's "real name" with a click on the name has caused some problems. Some users expected that by clicking on a name would display all messages from that sender. Some preferred the single click to highlight the entire row of messages, while others preferred to have the change name feature hidden in a menu. Single clicks on the main window that results in a new window popping up are very annoying.

To keep the interface consistent, a single click on the name performs no significant action (it only displays the total number of messages sent by that person on the status bar at the bottom). A double click will bring out the address book feature in which the user can change the name as well as to track data such as address and telephone number for that person.

11. TimeStore 2.0 provided no function to organize email by subject.

Version 3.0 does not attempt to address the problem associated with conversation threads. The user can use the search feature to create a mailbox view with a subject line. In the message window, the user has the option to create a mailbox view with the current subject line as the default used by the search tool.

12. Most users complained that the date ribbon at the top did not seem clickable. Once they understood its function, they felt that the changes in the view were not expected.

This feature has been removed, replaced by the ability to navigate the timeline by dragging the date ribbon. Changing the display granularity is now a menu item.

13. Users expected to be able to scroll continuously through time using the horizontal scroll bar. TimeStore 2.0 does not allow continuous scrolling.

TimeStore 3.0 now supports continuous scrolling of the timeline.

14. Other information in Eudora that the users took advantage of such as the size of message, whether the user has replied or not, label, and priority was not available in TimeStore 2.0.

TimeStore 3.0 only tracks the status of the message. The display of other information is not implemented due to time constraint.

15. Users also suggested that TimeStore 2.0 could be integrated with calendar and address book functionality.

This has been implemented in the version 3.0.

In conclusion, the TimeStore 2.0 interface did not take into account many issues in interaction design. For example, many areas on the screen that will react to mouse clicks were not obvious to many users. The lack of feedback in some operations also bothered some users. We feel that the new interface is a major improvement over the previous version. Users' reactions to the new interface are described in Chapter 5.

## Chapter 4

# The Implementation of TimeStore 3.0

---

The current version (3.0) has been completely rewritten as a cross platform application because we wanted a wide range of users from different computing platforms to be able to try TimeStore.

TimeStore is built on top of Sun Microsystems's Tcl/Tk version 8.0 alpha 2 [Sun 97]. Tcl/Tk is a cross platform scripting language that can be used to implement GUI interfaces very quickly. The functionality of Tcl/Tk applications can also be extended by writing C functions and associating C functions to Tcl/Tk commands. The resulting program consists of a main executable and a number of script files.

We chose Tcl/Tk because of its cross platform abilities, its ease of use, and its ability to add our own C extensions. Tcl/Tk provides a set of API calls for services such as file I/O, sockets, and memory allocation which result in a single C source code that can be compiled for Unix, Windows 95 / NT, and the Macintosh. Only the Macintosh version requires some additional code for proper data initialization. At the time of writing, TimeStore has been tested

on Windows 95, NT, PPC and 68K Macintosh, Sparc and x86 Solaris, and FreeBSD Unix. Theoretically, it should compile on any platform that is supported by Tcl/Tk.

TimeStore only implements the most basic functions of the SMTP and POP3 protocols. Certain features were omitted to facilitate quick software development. Many “convenience” features such as having the option to manage messages on the mail server or the ability to determine which messages on the server are new have been omitted. Therefore, TimeStore must download all messages from the server to determine if the messages are new or old. Local area network users would not notice a large difference. However, dialup users may notice slow mail download if they choose to leave messages on the server.

TimeStore is an independent email system for obvious reasons. Data is kept in nine different files. First, all names and email address along with separate IDs are kept in one file (see Figure 19). All email messages are kept in a single file with an index file that contains information such as a name ID, email address ID, subject, and the received date. Address book, calendar, tasks, and notes are stored in their own separate files except that any large text field is stored in a common text file. Finally, the mailbox view information is stored separately.

At startup, TimeStore loads all index files into memory. Although this scheme is not very efficient, it is adequate to handle the mailstore in the study. The index for each message takes up less than 200 bytes, and so a mailstore with 4000 messages would use about 800K. However, if TimeStore was to be used regularly over a long period of time, then additional support must be built to archive parts of the mailstore.

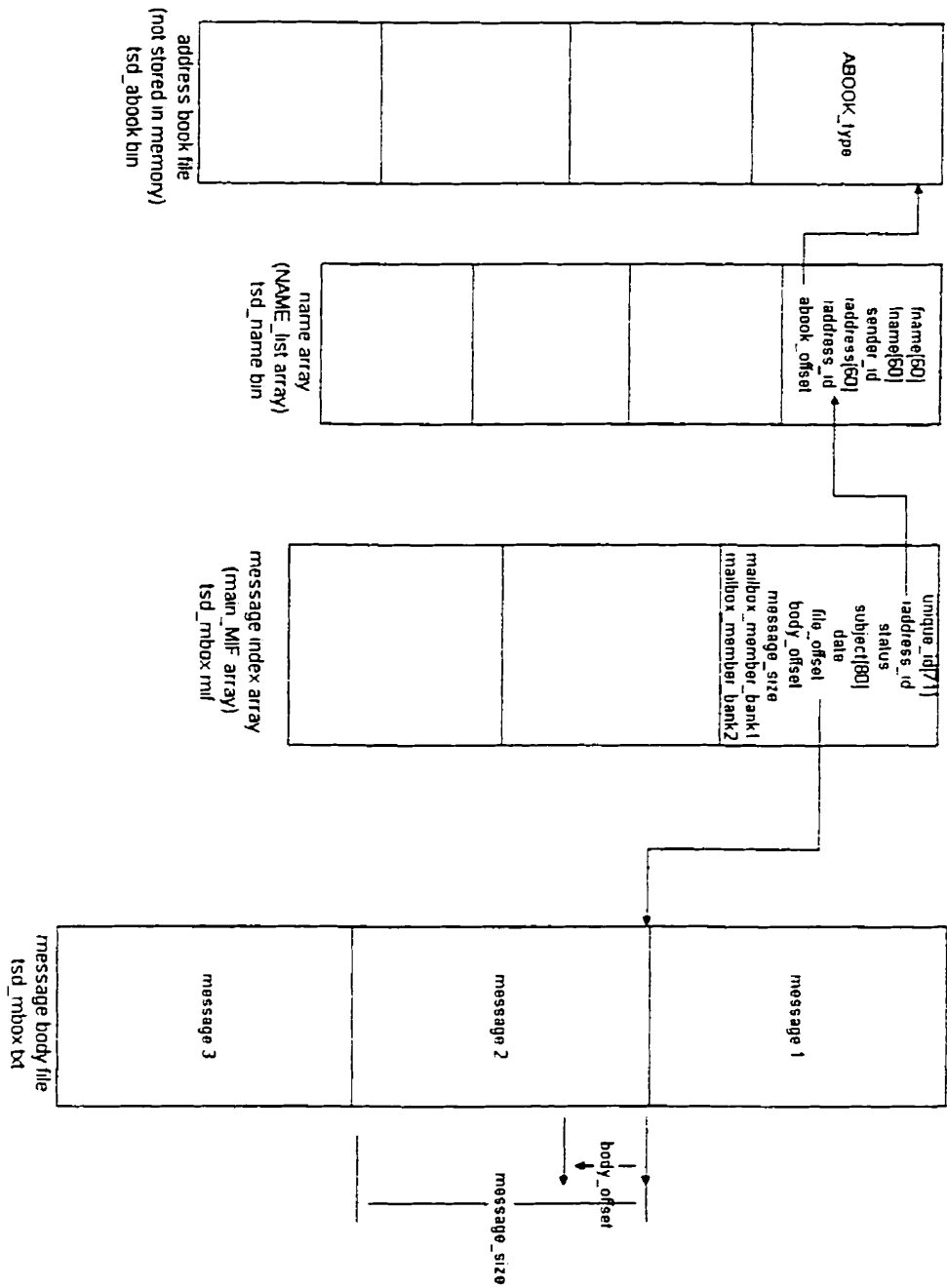


Figure 19 - Message handling data structures. The structures are stored on separate files. Only the NAME\_list and main\_MIF arrays are loaded in memory. The name in parenthesis is the variable name, which can be found in the source code. The name of the data file is shown after the array variable name.

The *iaddress\_id* field in the message index array indexes a name record in the name array. Email addresses are unique and if a sender has more than one email address, he or she would have more than one record in the name file. The *file\_offset* field in a message index record is an offset into the *tsd\_mbox.txt* file, which contains all messages. The *message\_size* field stores the exact length of the message in bytes. The *body\_offset* stores the number of characters to skip from the file offset to the beginning of the actual message content.

The *message\_member\_bank1* and *message\_member\_bank2* fields in the message index record are used to track which mailbox view the message belongs. Each of the two fields is a 32-bit number is used as a bit mask. The mask is calculated from using the formula  $2^{(\text{mailbox view ID})}$ . If the ID is greater than 32 then *message\_member\_bank2* is used. Both fields are initially zeroed.

The calendar, tasks, and notes data structures are organized much as messages are organized (see Figure 20) with one exception. The three data types share a common file *tsd\_misc.txt* to store the variable amount of text in their description fields.

At runtime, an array called the *dgraph* is built from the message index array (see Figure 21). Each element in the *dgraph* array corresponds to one time unit (day or month) and is the head of a linked list. In this linked list, each element represents one sender and has an array of pointers (*dlist*) that references the main message index array. All messages from this sender that was received on this time unit will be listed in the *dlist*. The runtime data structure for the calendar, tasks, and notes are organized in a similar fashion.

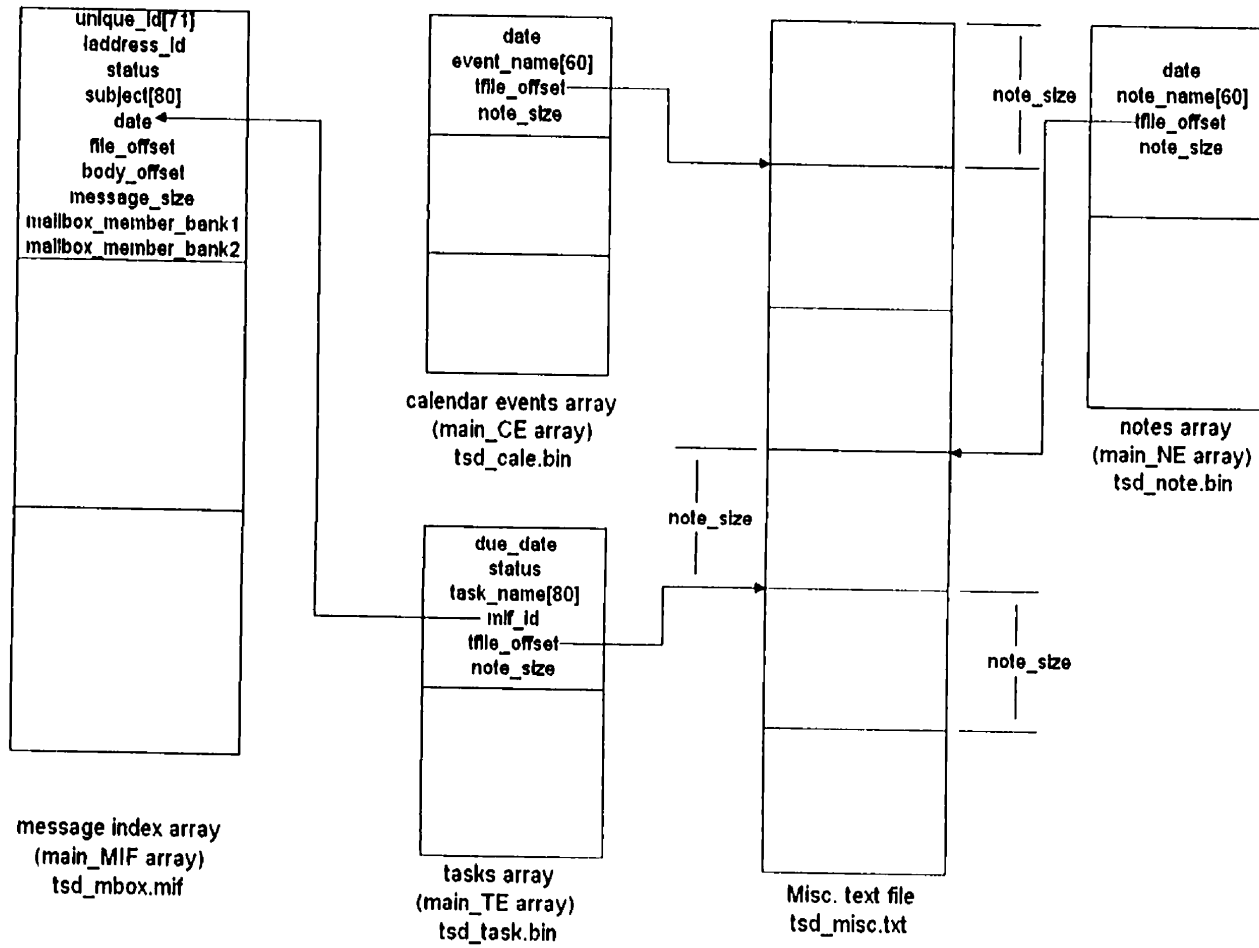


Figure 20 - Calendar, tasks, and notes handling data structures and files. All structures shown are loaded in memory except for the misc. text file. The name in parenthesis is the variable name, which can be found in the source code. The name of the data file is shown after the array variable name.

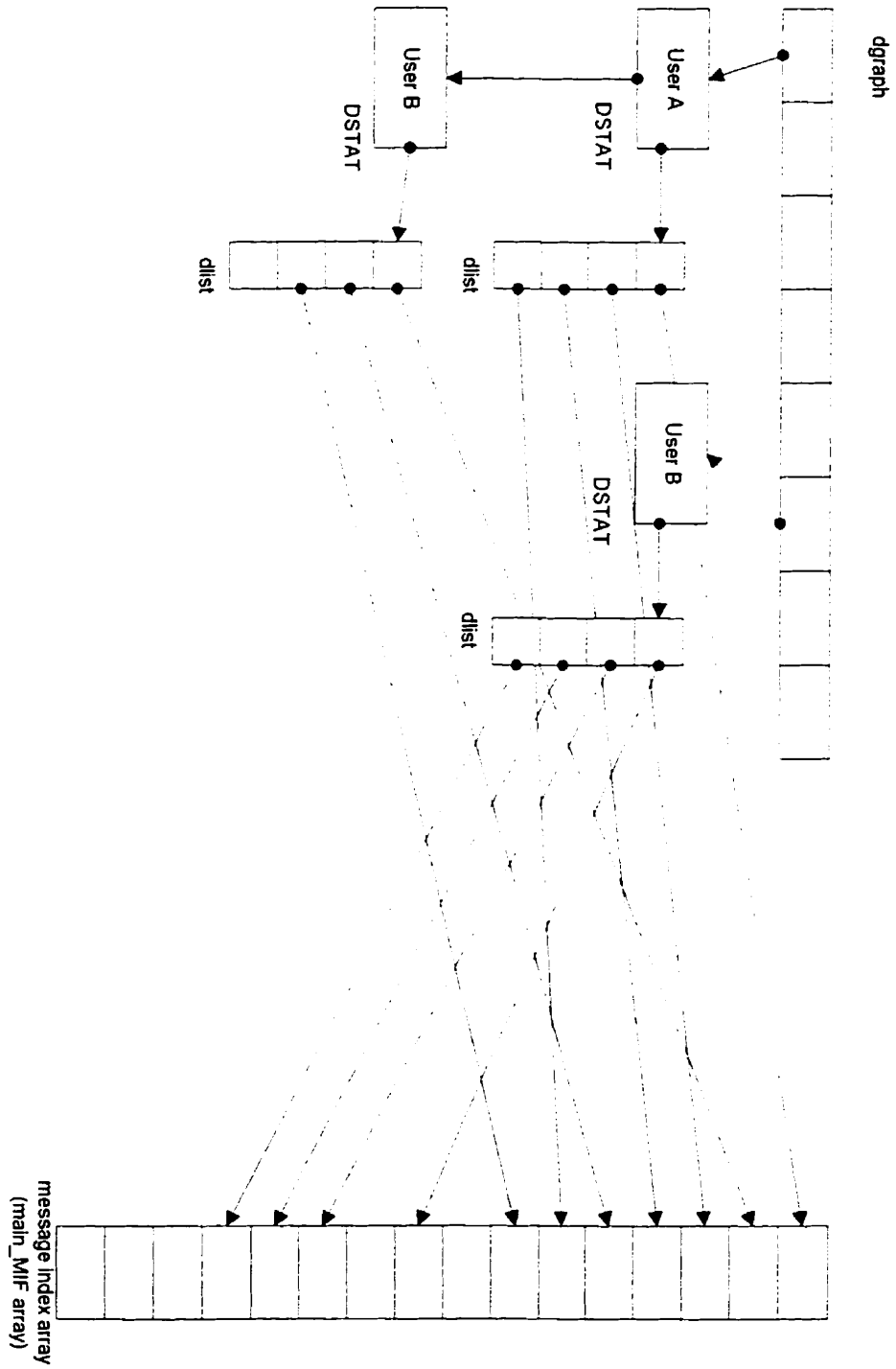


Figure 21 - Runtime data structure designed for fast access to message information.



A separate array of pointers to the *NAME\_list* array is constructed at runtime with only the names that TimeStore needs to display. The use of such a pointer array also eliminated the need to re-sort the main name array. TimeStore's drawing algorithm is very simple. With each element in the *dgraph* array, it follows the linked list down. With each element in the linked list, it determines whether some of the messages referenced are either marked or unread. If so, the dot is drawn in red. The y coordinate of the dot is determined by locating the sender's name (in the name pointer array), and then using this position and multiplying by the height of one line in the listbox.

When the user moves the mouse pointer over a dot, the mouse position is noted. The x coordinate is used to locate the *dgraph* array element. The y coordinate is used to generate the position in the name pointer array, which is then used to get a sender ID. This ID is used to traverse through the linked list to find the specific element, which references the messages represented.

## Chapter 5

# Usability Testing

---

Our objective in the usability testing was to understand whether time-based visualization is useful. Moreover, we also wanted to know if the integration of other personal information into an email system is useful.

Usability data was gathered using a combination of qualitative interviews, audio recorded think-aloud sessions, and screen / audio capture by the user's computer using Microsoft Camcorder (see Figure 22) on Windows 95 [Microsoft 97]. Camcorder is a free utility that can be downloaded from Microsoft's website.



Figure 22 - Microsoft Camcorder. Notice it has a very simple interface for recording movies. However, it lacks a pause button which users would find useful.

Camcorder has a very simple interface. All the user has to do is click on the record button and Camcorder will begin recording immediately. The use of

Camcorder is not automatic and the users have complete control as to when to use it.

Camcorder was originally designed for recording short demonstrations on the PC with optional audio capture. It has an option for creating highly compressed stand alone executable movie files for easy distribution. However, we had a few technical problems with Camcorder. The most severe is that the user cannot disturb the PC when Camcorder is busy compressing a movie. The result can be a corrupted movie. Another problem with Camcorder is that it requires an enormous amount of free disk space on the user's boot drive. Although Camcorder uses a special compressing video codec that can result in highly compressible movie files, it must write out the movie to disk uncompressed during a recording session. The disk space required is roughly 7.7 MB per minute for screen resolution of 1024 by 768 pixels and 16 bits per pixel. The same 1 minute file can be compressed to about 500K using standard utilities such as WinZip. A discussion of possible improvements to Camcorder for use with usability research can be found in the next chapter. Note that Camcorder must be installed first because the movie files make use of the compressing video codec that is installed with Camcorder.

## **Methodology**

We wanted to test the system with users from a variety of backgrounds. The usability study for the previous version of TimeStore was limited to Macintosh users using Eudora. The current version of TimeStore is implemented to operate on different platforms (Windows 95, Windows NT, MacOS, and most flavours of Unix). However, due to the lack of screen-

capturing tools available for Unix and technical problems with the Mac version of TimeStore, this study was done with Windows 95 users only.

Users selected for this study range from a novice computer user (1 year experience on Windows 95) to advanced (more than 5 years on Windows). Five Windows 95 users were recruited based on their (1) diverse email systems; (2) open-mindedness with new software; (3) enthusiasm for the study; (4) diverse background; (5) Fast computer (Pentium class) and (6) multimedia support on their PC. The users were interviewed and the screen recordings collected weekly.

Initially, a short interview was used to gather basic user data and information on how they use email. Users were then given a short demo and training session in order to highlight the various features. After the user had a chance to try out TimeStore for a few minutes, the user's initial impressions were recorded.

Users were then asked to use TimeStore as their primary email system while keeping their old email system as a backup and for features not implemented in TimeStore. During a two to three week test period, users were asked to use Camcorder to record their TimeStore sessions. A microphone was provided to record the user's thoughts expressed aloud. The result is an AVI movie file that can be played back on any Windows PC using the standard MediaPlayer.

Originally program instrumentation was coded into TimeStore but the code was causing instability on the pilot user's PC. Consequently it was disabled and all data was collected solely with Camcorder.

## **Initial Interview Questions**

All interviews were informal. The initial interview took place after a short demonstration of TimeStore. The purpose of the interview was to gather basic information about the user, their current email system, and their email usage pattern. Some of the questions asked in this interview were the same as the ones in the previous TimeStore study [Silver, 96].

### ***General Information:***

1. What is your occupation?
2. How many years of computer experience do you have?
3. What platforms have you used in the past? Describe the level of expertise on the computer systems that you have used? (novice, competent, guru)
4. What is the platform you are using now?
5. What applications do you usually run on your computer?
6. How many years have you been using email?

### ***Information on User's Current Email Usage:***

1. What is your primary email system?
2. Why did you choose this program?
3. How many email accounts do you use with this system?
4. How many email messages do you have in your system?
5. How many messages in general do you receive in a day?
6. How many messages are from mailing lists, and how many are directed to you personally?
7. How often do you check for new messages?

8. How many locations from which you access your email? How do you access your mail remotely?
9. Do you use the status settings (Eudora Users only)?
10. Do you use the priority settings? Labels (Eudora Users only)?

***Reading and Filing Strategies:***

1. What is the percentage of messages that you read completely?
2. What is the percentage of messages that you keep?
3. How do you categorize your mail (a single inbox, multiple mailboxes, etc)?
4. How many messages are in the inbox (if mail system supports an inbox)?
5. What type of messages do you save? Delete?
6. Do you delete messages that were filed?
7. How often do you clean up your mailboxes?
8. Do you use the rule-based filtering feature to file messages (if applicable)?

***Retrieval Strategies:***

1. How often do you need to retrieve old messages? Usually from how long ago?
2. What is your usual strategy when you need to retrieve older messages?
3. How often do you have problems finding a message?
4. What problems have you experienced when you are trying to locate a message?
5. When you have problems locating a message, what information do you remember about the message but could not make use of in the search?

6. When you do have problems locating a message, where you able to find the message in the end? How long did it take you?

### ***Initial Impression of TimeStore***

1. What do you like about TimeStore?
2. What don't like about TimeStore?
3. Do you think TimeStore will be useful to you? For which tasks?
4. Do you think you will be using the Calendar feature?
5. Do you think you will be using the Tasks feature?
6. Do you think you will be using the Notes feature?
7. Do you think you will be using the Address Book feature?
8. Do you think you will be using the Mailbox View feature?
9. Other specific comments on the interface, interaction styles, etc.

### **Daily Usage Questions and Final Interview Questions**

As mentioned before, users were asked to think-aloud when they are using TimeStore. The result was then captured into a movie file. At the end of each day, the users were asked to answer a set of questions if relevant.

1. What did you do with TimeStore today?
2. What did you like about TimeStore?
3. What areas in TimeStore were superior to your usual email system?
4. What did you dislike about TimeStore?
5. What areas in TimeStore were inferior to your usual email system?
6. Describe any problems you had with TimeStore today. Were they severe?

7. Do you have any suggestions for improving the user interface?

A brief weekly interview would also be held to gather other comments that the questions may have missed. Users were encouraged to freely discuss what they think about TimeStore.

### ***Final Interview Questions***

The objective of the final interview was to determine if the user's impressions of TimeStore had changed. We also wanted to know what their opinion of the system was after four weeks of regular usage. In the final interview, the users were asked the following questions:

1. Did you like using TimeStore?
2. What did you like?
3. What did you dislike?
4. Do you like the time-based concept? Was it useful?
5. Did you find the calendar feature useful?
6. Did you find the task feature useful?
7. Did you find the association of messages with tasks useful?
8. Did you find the notes feature useful?
9. Did you keep persistent mailbox views? Why or Why not?
10. What would make TimeStore better?
11. If TimeStore was built with all the features that most email systems have (better SMTP/POP3 support, mailing list support, MIME / BinHex decoding, etc), would you keep using TimeStore?



## **Results of the Initial Interview**

The highlights of the results are discussed in this section. Please see Appendix A and B for individual responses to the questions.

### ***Results from the Questionnaire***

The size of the users' mailboxes ranged from under 500 to over 2000 messages. Users received between 10 and 40 messages per day. Only one user subscribes to many (5) mailing lists and he reported that he was overwhelmed by the number of messages. In general, users read all of their personal mail but only skimmed through mailing list messages.

All users have simple categorization in their mail folder hierarchy. Users mainly categorize messages by sender (friend, user X in department Y) or by origin (mailing list Z). This may be due to the fact that 3 (out of 5) users are administrators (2 network administrators, 1 Web administrator) and they mainly deal with people rather than with projects. One user did not need to categorize her mail because the mail client in Netscape Navigator provides basic conversation thread tracking. Semantic categorization used by the others are limited to categories such as meeting minutes, mailing list information, job search related, and jokes. The depth of the hierarchy for all users in the study was at most two.

All the users in this group needed to access their email away from their desktop machine. Their method for remote access is to use telnet to access their Internet service provider, and then use a text based program like *elm* and *pine*. Users were careful not to disturb their mailbox since they wanted to be able to keep a copy on their desktop computer.

When users needed to retrieve past messages, they could usually remember which mailbox the message belonged. Then, the user would search through the list using both name and subject fields. It is interesting to note that all users sorted their mailbox by time, and they had some idea as to the sequence of messages in the mailbox. When they scanned through the list past a certain point, they knew they had missed what they were looking for.

### ***Results from Discussions after the Initial Interview***

Users (3 out of 5) liked TimeStore from the start. They all liked the fact that TimeStore automatically categorized their messages by sender and they could see patterns in their correspondence with others. In addition, they all liked the ability to associate a message with a task and the fact that tasks appear in the timeline. The system administrators commented that TimeStore provided no easy way to delete a group of messages. A few users had trouble distinguishing between calendar and tasks, and none of them thought that the notes feature was useful. A majority of users thought the mailbox view was “cool” and one user commented that it can be used to track ongoing conversation with a group of people.

## **Results of the Daily Usage and Final Interview**

The purpose of the study was to determine if TimeStore was useful and to find ways to improve it. User responses were generally positive. The users’ comments are categorized into two parts: interface and functionality.

### ***Interface***

All but one liked the time-based visualization for email messages. All users commented that they could see trends of correspondence with their friends and associates. Two users who did not like the time-based visualization for

tasks preferred to see a regular calendar in the table format. There was also some confusion to the time used. Users were expecting the time sent was used in the display but the time received was used. In addition, some users complained that they have to scan up to the timeline to see the date. They preferred to see the date on the balloon itself, and a better indication of which date and name a dot represented when the mouse pointer enters a dot. A few users thought TimeStore took up too much screen space and there was no way to change the font size in the window.

A very important discovery was that users were not able to remember the exact date a message arrived as often as we had originally thought. They often had to click on a few dots in order to find the desired message. One user commented that TimeStore made it "more difficult" to read messages because he must locate the correct date, then double click on the correct message in order to read it. Another user commented that messages are displayed instantaneously when a message is selected (single clicked) in Netscape Navigator. TimeStore required the user to locate the desired message, possibly clicking on a few dots, then double click on the message. TimeStore's display of messages in separate windows also slowed the message reading process. One user commented TimeStore could open the message by double clicking on a dot. If there were multiple messages then TimeStore could open the first message.

The "Create Mailbox View" dialog box could be improved. The "logical operator" radio button was intended to select the logical operator to apply for different fields (from, subject, body, cc). However, one user who tried to search for two words in the message body thought the logical operator was to

be applied to the two words. As a result, TimeStore did not generate a useful mailbox view.

Users also did not like to search for outgoing messages by scrolling to their own name in the main display. They wanted to know who the recipient of the message was and preferred to see the outgoing message listed with any incoming messages from that person. The original intention for listing all outgoing messages in the user's name was so that if the user had to send mail from another location, the user can "cc" a copy to his email account. TimeStore will display it along with other outgoing messages.

A user commented that the colour coding scheme of the dots in the main display should be consistent throughout TimeStore. He thought the inbox as well as the bottom list should use red to highlight unread and marked messages.

TimeStore plotted dots using email addresses. If a person had multiple email addresses, the name would appear many times. The user can merge names together by dragging one name to another. All users appreciated this feature but we underestimated the number of people with many email addresses. Therefore, TimeStore was changed to automatically merge two names if their spelling were the same. The user has an option to undo a merge. Users also asked for the ability to select and merge multiple names rather than being forced to merge only two names at one time.

There were other minor usability bugs in TimeStore. A user opened a message but forgot she must select "reply" in the menu to reply to a message. She immediately started typing in the message window and TimeStore falsely allowed her to do so. Another problem was that the colour

chosen to represent “today” in the timeline was too bright and users had trouble reading it.

### ***Functionality***

All users had difficulty in differentiating between calendar and tasks. Users only used tasks. However, one user commented that if calendar events can be set to recur after a certain interval then it would be easier to distinguish between the two. Users did not find a use for notes because they rarely have information that is not related to a task that they have to do. One user commented that notes were more like “journal” or “diary” and he thought “notes” was not an appropriate description for the feature. He also mentioned that rather than having a separate notes feature, the user should be able to associate notes with a sender and a certain date.

Most users thought that the ability to associate a message with a task was useful. However, one user commented that it would be more useful if the content of the message was copied into the description area in the task dialog box. Users also demanded more support for cleaning up the mailstore. For example, users wanted the ability to delete all messages from a sender, or all messages in a mailbox view.

A minor annoyance in the task dialog box was that user must enter the day of month as an integer. Users complained that they prefer to select the date from a calendar in table form. Currently, users have to go back to the main window or use another calendar to find the correct day. Moreover, some users did not make use of the status (“in progress” or “complete”) but preferred to delete tasks when they had finished them. Another problem with tasks is that TimeStore requires tasks to be associated with a particular due date. However, users said that are certain tasks that “they just have to do” but

did not have any particular due date. It is “a hassle to have to figure out the due date”.

Only one user used the mailbox view consistently. He did so to help him find messages from a mailing list that was of interest to him. He did not subscribe to any mailing lists before because they brought too many messages and not all messages were interesting. However, he subscribed to mailing lists again because he thought TimeStore could help him by only showing messages that he was interested in. He also used mailbox views to help track an ongoing conversation with a few of his friends.

The inbox was confusing for one user. The user expected the inbox to only contain new messages. One user did not use the inbox because he did not read all of his incoming messages. This problem became more apparent when a user subscribed to mailing lists because TimeStore’s mailbox view was able to let him filter out messages of interest. He stopped using the inbox and found new messages by moving the main display horizontally so that “today” is on the left edge, then scrolled down to search for new messages.

“Cut and paste” functionality was omitted in the coding stage and was not discovered by the pilot user because he did not use TimeStore to compose or reply to messages. Other users who did not use TimeStore to send or reply to email used TimeStore much more after “cut and paste” was available.

An interesting insight occurred during a final interview session. The user commented that TimeStore has made him view his email more like a database than messages in folders. As a result, the user’s expectations of TimeStore grew and TimeStore’s inability to provide other statistics to

answer questions such as “Why can’t I view by how much time I spent on a person?” became apparent.

### **Lessons Learned about Methodology**

All users were somewhat reluctant to think-aloud, especially at the beginning of the study. One user “had to constantly remind himself that he is making a movie and someone else will be observing the movie, but some things comes so naturally it wasn’t worth mentioning”. This brings up a problem with the think-aloud protocol in a setting where no one is monitoring the process. In a usability lab, the experimenter can ask the user specific questions if the user is not talking or not being clear about his intentions and expectations. In this study, this was not possible since the movies are collected once a week and the user cannot be questioned until a week later. A partial solution is to play the movie back to the user at the next meeting and ask the question then, but the user may not be able to recall his exact thoughts.

Think-aloud evaluation depends on verbal data from subjects which implies accepting the subject’s interpretation of them [Ericsson and Simon, 93]. The effect of verbalization is dependent on whether the verbalization brings new information to the subject’s attention. Verbalization that involves explicit explanation can alter the structure of the subject’s thought process because of the subject’s focus on procedures. During a think-aloud session, subjects are asked to do something more than they normally do while thinking. When the task processes require a lot of working memory, subjects often stop verbalizing and remain silent as is observed in the study.

The explicit action to record the movies is both a problem and a solution. By allowing the user to control when to begin the recording and which movies

to keep, users feel less intimidated about this study. However, the explicit action has also “put pressure” on the user, thereby affecting the thought process.

Camcorder was never designed to be used to record uninterrupted in the background. It caused one user’s machine to crash often. The software is also a memory hog and machines with less than 32 Mb of memory took a large performance penalty. Moreover, it required a lot of disk space on the user’s boot drive when it is recording. When disk space ran out, the result was usually a corrupted movie file without warning. Therefore, users were always worried about running out of disk space. However, once the recording is complete, the movie files have a very high compression ratio (95%) and can be stored very efficiently.

### **Factors Affecting TimeStore’s Usage**

User motivation level is major factor in deciding how much TimeStore is used. Most users’ usage of TimeStore dropped significantly after the second week, which coincided with when they were familiar with most of TimeStore’s features (and quirks). One user felt that “he has an obligation to make interesting movies” and therefore, when he thought that he had done everything, TimeStore was used less and less. Most of the movies showed the user using TimeStore to retrieve and read email. Moreover, asking users to change their email habit temporarily is a burden and this may be the reason why users do not use TimeStore as much after the first week.

The second factor is that TimeStore is not a polished piece of software. There were many “rough spots” where conveniences were sacrificed for quick development. The initial version of TimeStore that the pilot user



received did not have “cut and paste”. As a result, his usage of TimeStore dropped significantly after the first few days. When “cut and paste” was added, his usage of TimeStore increased for a few days then dropped again because TimeStore did not support attachments. One user commented that “the missing features caused him to alter his normal way to use email.” Therefore, at times he would feel less motivated to use TimeStore.

The most important factor is that email is one of the users’ most sacred data. Users must be convinced that by the end of the study, they will not lose any data. Moreover, they must be able to transfer the data that they created with the new system back to their original system. This is especially true for TimeStore since the user would not want to create duplicate data simply for the sake of participating in the study.

There is no easy solution to the problem of synchronization of data. On one hand, TimeStore could rely on external applications as in the case for TimeStore 2.0. However, this limits the design flexibility and complicates the technical implementations to the point that the range of users we can study is limited to a small number of systems we can support.

There are two ways to deal with the synchronization problem. One solution is to build a system that integrates with existing systems seamlessly. The user can change data in one system and the change would propagate to the other system automatically. This solution would work well if the user’s system has an open architecture and additional functionality can be included with ease.

The other solution is to design the system to work independently. Support for the users’ existing systems can be implemented with one-time import and

export functions. The user can synchronize his data at discrete points during the usability study. This allows more flexibility in the design but the system that we implement must contain all the features users normally would use, including those that may not directly relate to the study.

Both solutions to the synchronization problem involve intimate knowledge of data organization in other applications. However, to export data back into the users' system would require full knowledge of the data organization and this may be impossible.

We incorrectly assumed that if users find a certain feature convenient then they would use it. This was the case with the calendar, tasks and notes features. All of the users thought they were useful but most had limited use for these features. Further usability studies should be based on a larger study group and the use of software based organizers should be taken into account.

In hindsight, the user selection should have considered the use of electronic organizers and contact managers in addition to general email usage. This work is a conceptual study of linking all personal information together into a single interface. We had hoped the tight integration into their email interface was enough to get people started using TimeStore to organize their tasks. This assumption was incorrect and users' usage of the task management features was not consistent. TimeStore's usefulness in the short term is dependent on its ability to import the user's existing data and the ability to allow the user to continue using his email without any interference. However, adding support for existing contact managers and organizers would be impractical, especially for a short study. Moreover, finding users who fit the criteria and who are willing to participate in the study is even more difficult.

---

## Chapter 6

# Contributions and Future Work

---

The TimeStore Project is a work in progress. We begin this chapter by listing the contribution of this thesis, and then discuss three areas for future work: user interface, technical enhancement to the software, and better data capture tools.

### **Contributions**

This thesis is the result of work done in three stages. First, TimeStore 2.0's interface was redesigned based on recommendations from Silver [Silver 96] and other researches. Support for additional data such as calendar, tasks, notes, and address book was included into the new interface. Then, TimeStore 3.0 was completely rewritten using Tcl/Tk to include features such as a continuous timeline, full text searching, support for addition data, basic Internet mail support, and cross platform capabilities. Finally, a qualitative usability study with a novel data collection and playback methodology was designed and carried out.

## **Improving the User Interface**

The usability study has revealed a number of shortcomings in the TimeStore user interface. The most severe is that users are not able to remember the exact date for many messages. Consequently, the user must click on a succession of dots in order to find the desired message. How does one improve upon an interface designed around the assumption that users can remember the exact date for a large number of their messages?

Users find TimeStore more useful for finding messages than their current email system despite the fact that they may not remember the exact date. Although users were not always able remember the exact date, they were able to estimate the date and use that as a starting point for further browsing. From this point, the users prefer to browse the subject field. Therefore, if the message was temporally isolated, the user can immediately locate the message. If not, the user will have to click on a succession of dots and search for the message using the subject field.

To improve the distinction between unread messages and read messages, the dots are replaced by small icons (see Figure 23). The icons are kept very simple since complex icons do not help the user to locate information [Byrne, 93]. The following is a list of the new icons and what they represent.

- An open yellow envelope represents read messages.
- A closed yellow envelope represents unread messages.
- A rectangle with a line at the top and a small dot near the bottom represents a calendar event.
- A task is represented by a slanted rectangle with a red dot in the upper left corner.
- A white rectangle with a folded corner represents a note.

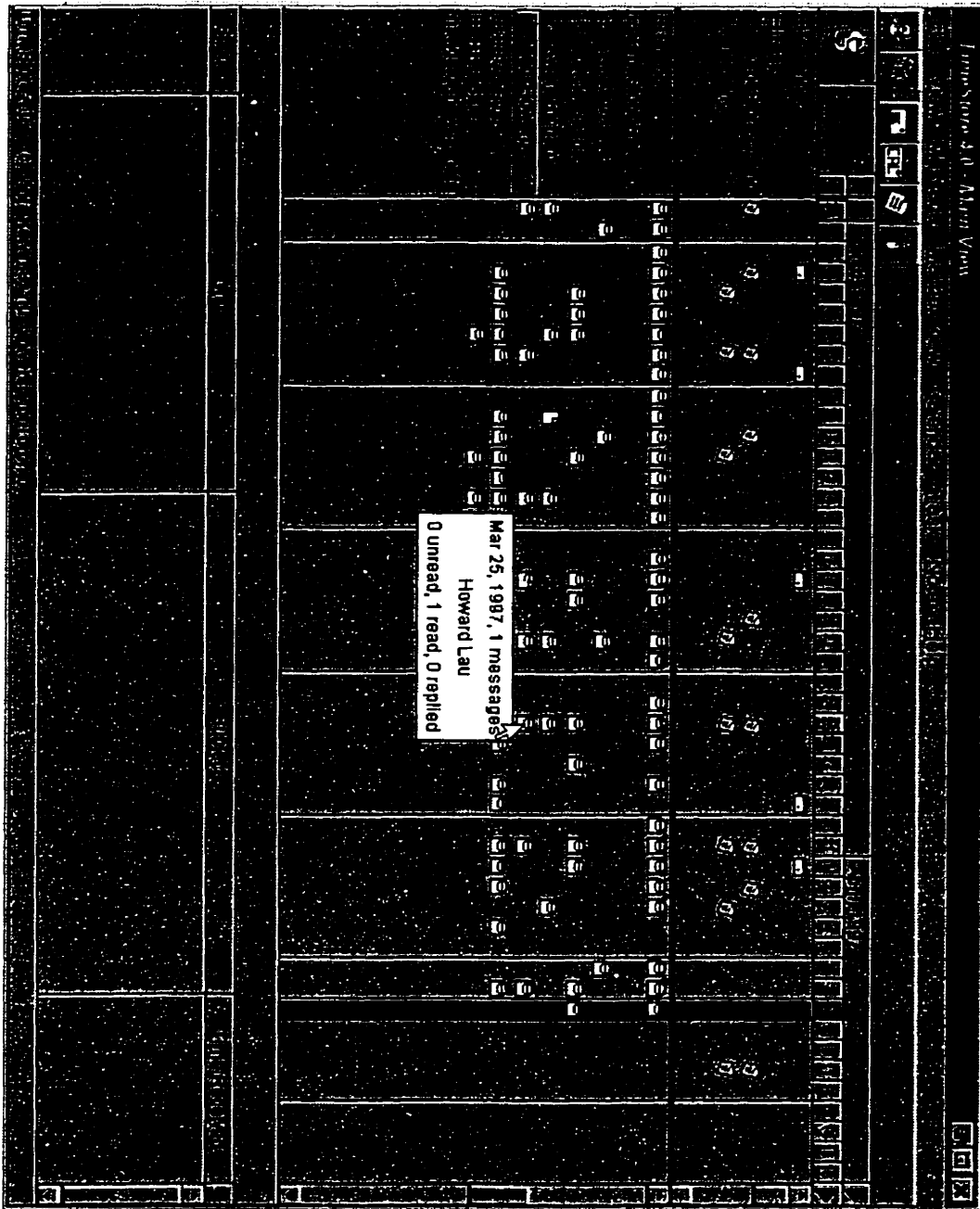


Figure 23 – Proposed TimeStore main window showing the hierarchical organization of sender names and new icons.

In the current version of TimeStore, the user must repeatedly browse the small list then click on an adjacent dot if the desired message is not found. This can be annoying when the user can remember that the general sequence of messages but not the exact date.

One solution is to display messages that range from  $x$  number of days before to  $y$  number of days after rather than just from a single day. The bottom list can be enlarged to display more messages at one time. Its size should also be adjustable. When the user click on an icon, TimeStore will show messages from that day at the centre of the window. Older messages are shown above and newer messages below. The range can be set depending on performance requirements or the user's needs.

The overwhelming number of names must be organized more effectively. Users should only see names of sender that they know, and TimeStore should be able to hide all individual names from sources such as mailing lists, while maintaining access to the messages. As a possible solution, TimeStore can provide the ability to organize names in a hierarchical structure. Names can be included into a group by drag and drop or built automatically by using keywords. These name group folders could be "collapsed" to hide the names in that group. Icons the represent a summary of the "collapsed" group can be drawn in gray to differentiate them from the regular icons.

The current version of TimeStore can only display messages, because it was designed to be a mail system. However, TimeStore's display can be extended to include information normally found in contact management software. For example, notes that are associated with a sender can be displayed with a different icon in the main display. Tasks can also be associated with a

sender. However, the top portion should still display all of the user's tasks, including those who may not be associated with a sender.

TimeStore lacks a to-do list. Originally, we had hoped that users would use the taskbox as a to-do list. However, those users who used tasks preferred to see dots on the main display rather than having to open the taskbox because they reminded them to click on the icon to find out what they have to do. Users did not use the tasks feature when they had a task that did not have an explicit due date. A possible workaround for such tasks is if the user do not wish to specify a due date, then such tasks will always be displayed as today's tasks. This will prevent the tasks from "disappearing" with the moving timeline.

The search engine in the current version of TimeStore is very primitive. Due to the cross platform nature of TimeStore, we could not use an off-the-shelf "component" from companies such as Open Text. A better search engine will improve both the performance and accuracy of the search result. In addition, when the user is viewing a message, the keywords in the message that caused the message to be included in the result should be highlighted. This will aid the user when he needs to locate the text around the keyword, especially in a long message.

TimeStore also needs better conversation thread tracking. The technical problem is caused by email software that does not include a "In-Reply-To" or similar field in the header of the message. By including the message ID of the previous message, TimeStore can determine whether the incoming message is a reply to a message that is in the mailstore. Users should also be able to go to the next message without having to go back to the main window. There are three possible dimensions in which the user may navigate

by: time of arrival, sender, and conversation thread. The message window must provide three sets of buttons to direct the window to display the “next” message for each of the three possible dimensions (see Figure 24).

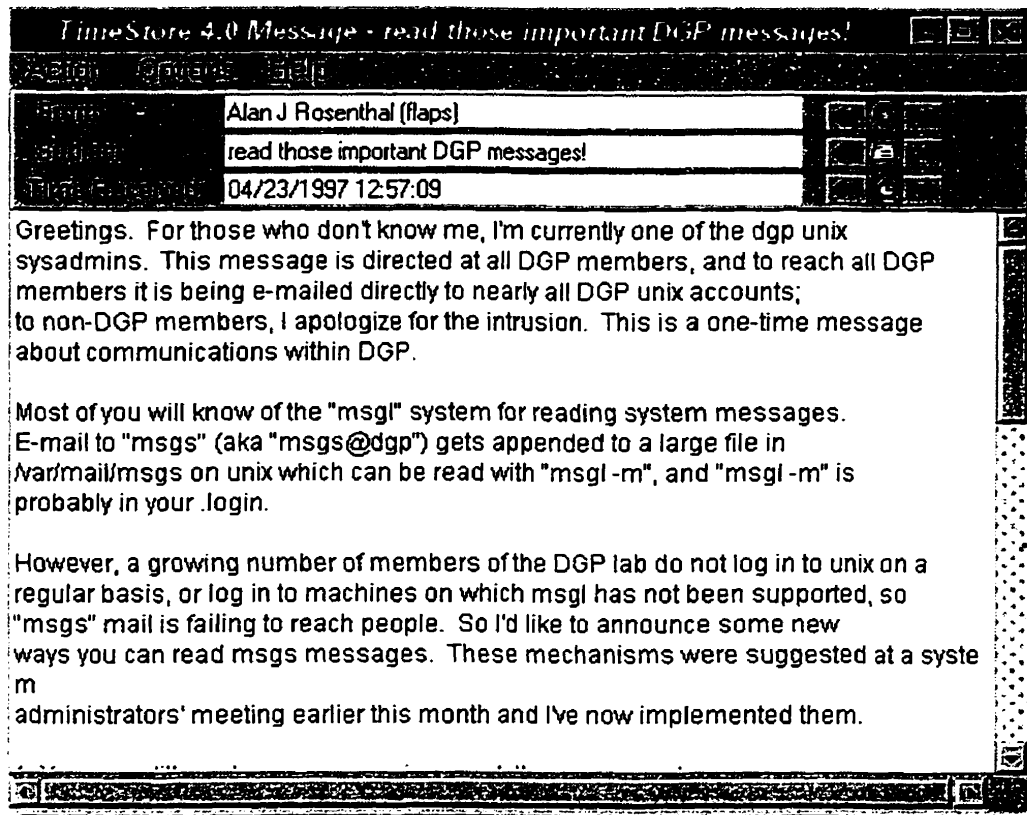


Figure 24 – Proposed message window with additional message navigation controls. There are three sets of buttons. The first row will move to messages from the same sender. The second row will move to messages in the same conversation. The last row will move to the next message in time (from any sender or in any conversation). When the user clicks on a button, the content of the window is replaced with the content of the next message.

The inbox algorithm was changed from including all unread and recent messages to just including recent messages after the second week of the user study because the assumption that the user would at least skim through all messages was incorrect. TimeStore should also provide alternative



algorithms for the inbox. For example, one algorithm would be to only list new messages. Another would be to list all new messages and unread from the past week. The user should be able to choose how the inbox operates.

### **Proposed Extensions to TimeStore**

A possible extension to TimeStore is support for PDAs such as the Apple Newton or the U.S. Robotics Pilot. The users do not sit in front of their PCs all the time. Consequently, they did not use the tasks or notes much because it was not convenient. PDAs are designed for dealing with such information and TimeStore can provide the interface that links tasks, notes, and messages together.

In addition to the support of PDAs, TimeStore can benefit from a number of smaller technical improvements. Users have complained TimeStore took a long time to complete certain simple operations. The reason was that TimeStore performed a full redraw of the window, which also included completely rebuilding all data structures and redrawing the entire timeline.

Redrawing the timeline is a very lengthy operation because in order to show a continuous timeline, the entire timeline must be drawn in memory. A possible method to speed up this process would be to draw the timeline in sections. The section that can be seen by the user is drawn immediately. Other sections could be drawn later when TimeStore is idle.

TimeStore requires a lot of memory at runtime because it keeps the indices for all messages in memory. As the number of messages increases over time, the amount of memory required will also increase. The user will rarely need to access old messages and loading all indices is a waste of memory. The

solution would be to only draw the main display with recent data. Earlier messages would be loaded and drawn only when the user navigates backwards in time. Statistical information, such as how many messages a sender has sent, can be summarized in another file to combine with the statistics from recent messages.

TimeStore also needs a better POP3 implementation that uses the "TOP" command to determine whether a message on the server has been downloaded. This feature was omitted because of the assumption that the user would use his regular email program to clear the messages on the server. This approach to the problem of maintaining identical data on both email programs is not always convenient.

Support for mail attachments is necessary for any future versions of TimeStore. Although this feature has nothing to do with research into time-based visualization and management of personal information, it is impractical to expect users to put up with such inconvenience over a long period in a usability study. Other "convenience" features such as selecting the display font and colours, name alias, and queuing outgoing messages should also be added especially if a longer duration usability study is used.

TimeStore's support for existing email systems should be improved. At the present, TimeStore can only import mailboxes from Eudora and Netscape Navigator. Additional support for Microsoft Exchange, Outlook and the standard Unix mailbox file used by *elm* should be added. Also export functionality should also be added for all systems supported.

The usability study has uncovered a number of problems that should be corrected in the next version. Generally, users find TimeStore useful.

However, some said they would replace their current email system with TimeStore if it has many of features in commercial email software such as attachments, and spell check. As mention before, providing such features is critical for enticing users to use TimeStore for long durations.

Another issue that must be resolved in the next version deals with the reference "time" that TimeStore uses to display messages. There are three different "times" that can be associated with a message: the message send time; the time that a message arrives at the mail server; and the time the user downloads the message. Each time stamp has its own advantages and disadvantages and users have different opinions on which time they prefer to see. Any further research with TimeStore must consider the three different possible time associations. Perhaps TimeStore would be able to track all three and give the user a choice as to which one to use for display.

One final note is that TimeStore was built using an early alpha version of Tcl/Tk 8.0. As with any software in the alpha stage, it has some instability problems. The reason we chose to use this version rather than an older, more stable version is because version 8 supports native widgets on both Windows and Macintosh. This support is especially important for the Mac version because of the location of the menu is different from that of Windows and Unix. More testing is needed when TimeStore is integrated with the final version of Tcl/Tk 8.0.

### **Building A Better Data Collection Tool**

Clearly, MS Camcorder has many flaws, especially in the context of this study. However, the use of such a tool has promise. The ability to analyze the user's interaction along with the user's intentions and expectations offers

very rich data, thereby allowing the usability test to run for a shorter duration.

Based on the findings in this study, the ideal recording tool should possess the following features. The tool should be able to write compressed video data directly to disk. Camcorder uses a compressing video codec but does not provide a way to write to a compressed file. The user should not have to start Camcorder manually. A single indicator is enough to inform the user that Camcorder is active. Also there should be a "Pause" button so that the user has control of what is recorded.

Camcorder does not record any keystroke or mouse clicks. This missing feature was not a big problem since most of the time we were able to tell what the user was typing or where the user clicked on the screen. However, if TimeStore had more keyboard shortcuts, such user actions would not be recorded. Another problem is that there was no way to tell the difference between a single click and a double click. Moreover, if TimeStore made use of the right mouse button, such data would not be recorded as well. Ideally, the tool should show a picture of a keyboard as well as a mouse during playback to show what keys and buttons are pressed.

The ideal monitoring tool should have the ability to remind the user to speak if the user did not talk as we have found in our study. However, such reminders may make the recording session even more intrusive and may affect adversely the user's actions with the software under study. Agent technology can also be included to monitor the user and ask intelligent questions at certain points in a session. Such recognition must be programmable by the experimenter because the interaction for each piece of software is different.

Ideally, the tool should only record the parts of the screen in which the experimenter is interested. However, what is applicable may depend on how the user uses other applications with the prototype that we are studying. For example, the user may be looking at a web browser and creating a task. In this case, we will want to record the web browser as well. In other cases, such as when the user is checking private online banking information, we would want to block out the web browser then. At other times, such as when the user is composing private email messages, the tool should be able to block out part of the application that is under study. Therefore, it may be impossible to have intelligent blocking of selected windows on the screen. However, if we record the entire screen, the user will feel some pressure and it may affect their usage of the prototype, especially if the duration of the study is long. In our study of TimeStore, all of the users expressed concerns about the privacy of their email messages.

## **Conclusion**

TimeStore version 3 is a big improvement over version 2. Many of the interaction problems in version 2 were corrected. The result of the usability study was mixed. Although TimeStore was helpful in many of their tasks, TimeStore in its current state is still not good enough for everyday use. Users disliked the fact that TimeStore provided no way to organize names. They thought the mailbox view feature was useful, but should be integrated with the main display better. They liked the ability to associate messages with tasks, but they also wanted to have tasks that do not have due dates.

On the positive side, the majority of the users believed the time-based concept is useful. They especially like the fact that messages are

automatically arranged by sender. They also like the ability to see patterns and trends, which can remind them to contact people. However, users cannot remember the exact date for a number of messages and therefore future versions TimeStore must compensate for such. Integrating the users' suggestions as well as the recommendations mentioned in this thesis would make TimeStore better than anything in the email software market today.

# Appendix A

## Initial Interview Results

---

### *General Information*

1. What is your occupation?

|        |   |
|--------|---|
| User 1 | Computer network admin. Currently unemployed. |
| User 2 | Web site / system administrator.              |
| User 3 | Computer network analyst.                     |
| User 4 | Ph.D. student, Computer Science.              |
| User 5 | Undergraduate student in Fine Arts.           |

2. How many years of computer experience do you have?

|        |               |
|--------|---------------|
| User 1 | 6 years.      |
| User 2 | 6 to 7 years. |
| User 3 | 6 years.      |
| User 4 | 10 years.     |
| User 5 | 8 years.      |

3. What platforms have you used in the past? Describe the level of expertise on the computer systems that you have used? (novice, competent, guru)

|        |  |
|--------|--|
| User 1 | Windows 3.1/95 (competent), Mac (novice), Novell NetWare (competent), Unix (novice). |
| User 2 | Mac (competent), Unix (competent), Windows 95 / NT (guru).                           |
| User 3 | Windows (guru), Unix (novice), OS/2 (novice), Novell NetWare (competent).            |
| User 4 | Windows 3.1 (competent), Unix (competent), Mac (novice).                             |
| User 5 | Mac (novice), Windows 3.1/95 (competent).  |

4. What applications do you usually run on your computer?

|        |   |
|--------|---|
| User 1 | Mainly Internet related. Netscape Navigator, SimpTerm Telnet, Eudora Pro, MS Word 95. |
| User 2 | Notepad, Netscape Navigator, MS Internet Explorer, Adobe PhotoShop, Adobe PageMaker.  |
| User 3 | Netscape Navigator, MS Outlook, MS Word 97.   |
| User 4 | Microsoft Office, ProComm, Elm (remote), Internet Explorer.                           |
| User 5 | Adobe PhotoShop, MS Word 95, Adobe Illustrator, Netscape Navigator.                   |

5. How many years have you been using email?

|        |          |
|--------|----------|
| User 1 | 5 years. |
| User 2 | 6 years. |
| User 3 | 6 years. |
| User 4 | 9 years. |
| User 5 | 1 year.  |



***Information on User's Current Email Usage:***

1. What is your primary email system?

|        |  |
|--------|--|
| User 1 | Eudora Pro 3.0.  |
| User 2 | MS Outlook.  |
| User 3 | MS Outlook (work), Netscape Navigator 3.01 (personal). |
| User 4 | Elm (on Unix).   |
| User 5 | Netscape Navigator 3.01.                               |

2. Why did you choose this program?

|        |  |
|--------|--|
| User 1 | Initially used the free version because it was simple to use. He did not want to switch to another email system. |
| User 2 | Compatible with MS Exchange (company uses Exchange).   |
| User 3 | Compatible with MS Exchange (company uses Exchange), Navigator email part of browser.                            |
| User 4 | She had to choose between Pine and Elm; Pine had a bug on the system.  |
| User 5 | Netscape Navigator is free for students, email system part of Navigator.   |

3. How many email accounts do you use with this system?

|        |  |
|--------|--|
| User 1 | 3 accounts, Eudora Pro can check multiple accounts.        |
| User 2 | 2 accounts.  |
| User 3 | 2 (business) with Outlook 97, 1 (personal) with Navigator. |
| User 4 | 1 account (Other accounts are forwarded to this address).  |
| User 5 | 1 account.   |

4. How many email messages do you have in your system?

|        |                            |
|--------|----------------------------|
| User 1 | 2071 after recent cleanup. |
| User 2 | Over 1000.                 |
| User 3 | Over 1200.                 |
| User 4 | 700 to 800 messages.       |
| User 5 | 444 messages.              |

5. How many messages in general do you receive in a day?

|        |                               |
|--------|-------------------------------|
| User 1 | 10 to 15 messages.            |
| User 2 | Varies from 0 to 30 messages. |
| User 3 | 30 to 40 messages.            |
| User 4 | 20 to 25 messages.            |
| User 5 | Around 10 messages.           |

6. How many messages are from mailing lists and how many are directed to you personally?

|        |   |
|--------|---|
| User 1 | He subscribes to 5 mailing lists. 40% were from mailing list while 60% are personal. He recently cut down to 1 list because he was overwhelmed by the number of messages. |
| User 2 | Over 60% are from mailing lists.  |
| User 3 | 100% of messages are directly to him personally.  |
| User 4 | She only has mailing lists that were subscribed automatically. Most messages are directed to her personally.  |
| User 5 | She did not subscribe to any mailing list, but is part of three school related lists. About 90% of messages are directed to her personally.                               |

7. How often do you check for new messages?

|        |  |
|--------|--|
| User 1 | At least 4 to 5 times a day.   |
| User 2 | Every 5 minutes at work. Otherwise whenever he has access.   |
| User 3 | At work, Outlook checks mail automatically. At home, he checks mail everyday roughly every 40 minutes. |
| User 4 | Whenever she has access.   |
| User 5 | Twice a day.   |

8. How many locations from which you access your email? How do you access your mail remotely?

|        |   |
|--------|---|
| User 1 | Many locations. He checks when telnet is available. He uses Pine when accessing remotely. He only skims the messages when accessing remotely, but reads them again in detail at home. |
| User 2 | He checks for personal mail at work using Netscape Navigator but he deletes them from his work machine. When he gets home, he downloads his personal mail again.                      |
| User 3 | At school, at home and where telnet is available. He uses Pine when accessing through telnet.   |
| User 4 | At school and at home.  |
| User 5 | At school and at home.  |

9. Do you use the status settings (Eudora Users Only)?

|        |  |
|--------|--|
| User 1 | Yes, he uses it to look for unread messages. He doesn't care for reply or forward. |
| User 2 | Yes. To find new messages.   |
| User 3 | N/A.   |
| User 4 | N/A.   |
| User 5 | N/A.   |

10. Do you use the priority settings? Labels (Eudora Users Only)?

|        |      |
|--------|------|
| User 1 | No.  |
| User 2 | No.  |
| User 3 | N/A. |
| User 4 | N/A. |
| User 5 | N/A. |

***Reading and Filing Strategies:***

1. What is the percentage of messages that you read completely?

|        |   |
|--------|---|
| User 1 | He reads about 80% of his messages completely. The other 20% are from mailing lists or jokes from his friends.                      |
| User 2 | He reads all messages completely unless they are from mailing lists, which he may delete if the subject line does not interest him. |
| User 3 | 80% completely. He skims the long messages.   |
| User 4 | About 90%.  |
| User 5 | 70%. Skip all jokes unless she has time.  |

2. What is the percentage of messages that you keep?

|        |   |
|--------|---|
| User 1 | 99%.  |
| User 2 | Keeps all messages directed to him personally.            |
| User 3 | About 90% for work related, 75% to 80% for personal mail. |
| User 4 | 50% to 60%.   |
| User 5 | 90%.  |

3. How do you categorize your mail (a single inbox, multiple mailboxes, etc)?

|        |   |
|--------|---|
| User 1 | Categorize by name of sender (each sender has a mailbox) for frequent correspondence; previous employer; information about how to unsubscribe his mailing lists; a mailbox related to a web site he is helping to set up; and a mailbox for each mailing list, etc. |
| User 2 | Categorize by name of sender (each sender has a mailbox) for frequent correspondence. Infrequent correspondences are kept in a <i>misc.</i> mailbox. In addition, he has a <i>Business</i> mailbox for stuff related to job search as well as a jokes mailbox.      |
| User 3 | Categorize personal mail by sender; a jokes folder; categorize work mail by meeting minutes; by departments.  |
| User 4 | Mailboxes for each course she teaches, messages from her research advisor, internship, personal, IRG (a mailing list), messages from the university she was at previously.  |
| User 5 | All messages are in the inbox.  |

4. How many messages are in the inbox (if mail system supports an inbox)?

|        |                                |
|--------|--------------------------------|
| User 1 | 234 messages.                  |
| User 2 | 39 messages.                   |
| User 3 | 418 messages.                  |
| User 4 | 49 messages.                   |
| User 5 | 444 messages (entire mailbox). |

5. What type of messages do you save? Delete?

|        |  |
|--------|--|
| User 1 | He never throws out messages from his friends. He usually cleans out the mailing list mailboxes every 3 months or when the size of the inbox gets out of hand. |
| User 2 | Saves all messages directed to him personally.   |
| User 3 | He deletes messages that have a short useful life (meeting location, etc).   |
| User 4 | She deletes messages that have a short useful life (meeting location, etc).  |
| User 5 | She deletes mail that have a very short useful life ("call me", "what are you doing tomorrow" type messages).  |

6. Do you delete messages that were filed?

|        |   |
|--------|---|
| User 1 | Yes, when cleaning up.  |
| User 2 | Yes, but he only cleans up the <i>Misc.</i> mailbox. He also cleans up any attachments from messages that are no longer needed. |
| User 3 | Yes.  |
| User 4 | Yes.  |
| User 5 | No.   |

7. How often do you clean up your mailboxes?

|        |  |
|--------|--|
| User 1 | Every 2 to 3 months or when it bothers him.  |
| User 2 | He has a certain size in mind and when the entire mailstore reaches that size, he cleans up. However the size is not fixed and he has increased the size a few times in the past year. |
| User 3 | At work, he cleans up whenever the mail server is running out of space. He never cleans up at home.  |
| User 4 | At the end of each term or when the inbox has too many messages.   |
| User 5 | Never.   |

8. Do you use the rule-based filtering feature to file messages (if applicable)?

|        |   |
|--------|---|
| User 1 | Yes, but he did not set it to filter automatically. He does not like to hunt around to find new messages. |
| User 2 | No. He did not try to figure out how to do it.  |
| User 3 | No.   |
| User 4 | N/A.  |
| User 5 | N/A.  |

### *Retrieval Strategies*

1. How often do you need to retrieve old messages? Usually from how long ago?

|        |  |
|--------|--|
| User 1 | He has to retrieve every 2 to 3 days. The message is no more than a week old.  |
| User 2 | Never. He only keeps messages around just in case he needs them.   |
| User 3 | Once a month at work to find out things he hasn't done yet. For personal mail, he sometimes needs to find messages that are part of an ongoing conversation. |
| User 4 | Daily, usually recent correspondence from within a week.   |
| User 5 | Twice a week (usually for correspondence), usually from a few days ago.  |

2. What is your usual strategy when you need to retrieve older messages?

|        |   |
|--------|---|
| User 1 | He can remember the mailbox that the message belongs. He then sorts the messages by date and scan through the subject field.  |
| User 2 | N/A.  |
| User 3 | He can remember the mailbox that the message is in. He then scans the subject field.  |
| User 4 | She can usually remember where to find the message. She also uses the time and the name of the sender to help her.  |
| User 5 | In the Navigator message window, there is a field to go back to messages that were part of the conversation. When this information is not available she scrolls back the list while scanning for the sender's name. |



3. How often do you have problems finding a message?

|        |   |
|--------|---|
| User 1 | Do not know exactly. He has trouble finding messages from mailing lists.                        |
| User 2 | N/A   |
| User 3 | Not usually but he felt that he was spending too much time than he should looking for messages. |
| User 4 | Not often, usually only have problems when the message is very old.                             |
| User 5 | Never.  |

4. What problems have you experienced when you are trying to locate a message?

|        |  |
|--------|--|
| User 1 | Usually there are too many messages in the mailing list mailbox. He does not use keyword searches because they take too long and do not always work. He also thinks keyword searches fail because he could not remember the exact keyword. |
| User 2 | N/A  |
| User 3 | He has trouble finding a message when the subject field does not reflect the content of the message. He has to read all of them to find the one he wants. Not knowing whether the message he was looking for was deleted or not.           |
| User 4 | Not being able to remember the folder.   |
| User 5 | N/A.   |

5. When you have problems locating a message, what information do you remember about the message but could not make use of in the search?

|        |   |
|--------|---|
| User 1 | He is able to estimate roughly when the message arrived, but Eudora does not help. In addition, he can remember certain keyword within the messages, but he cannot remember them exactly. |
| User 2 | N/A.  |
| User 3 | Words in the message.   |
| User 4 | None, but she prefers <i>grep</i> in Unix for keyword search rather than using Elm's built-in function.   |
| User 5 | N/A.  |

6. When you do have problems locating a message, where you able to find the message in the end? How long did it take you?

|        |  |
|--------|--|
| User 1 | No. He just gives up after a few minutes.  |
| User 2 | N/A.   |
| User 3 | Always able to find message in the end. However, once he had to load the mailbox text file into a word processor and use the find feature of the WP to find the message. |
| User 4 | Can find message most of the time. Longest search was about 5 minutes.   |
| User 5 | N/A.   |

# Appendix B

## Initial User Impressions

---

### 1. What do you like about TimeStore?

|        |  |
|--------|--|
| User 1 | Ability to associate a task with a message, ability to merge names, can see trends in correspondence.              |
| User 2 | Able to see trends in correspondence. Address book feature.  |
| User 3 | Able to see trends in correspondence.  |
| User 4 | N/A.   |
| User 5 | She liked the icons, and thinks the task feature is useful. She also likes the list of sender sorted by last name. |

2. What don't you like about TimeStore?

|        |  |
|--------|--|
| User 1 | Can't delete every message in a mailbox, cannot click on column labels to sort display.  |
| User 2 | Takes up too much screen space, Calendar should be in a table, colour scheme is not very interesting. He does not like the Time-based view and prefers to see a list of messages.          |
| User 3 | Can't delete every message from a person.  |
| User 4 | N/A.   |
| User 5 | She thinks the dots are confusing. She doesn't like to have to find the date, then scan down to find the dot. She prefers to be able to identify what date the dot falls on right the way. |

3. Do you think TimeStore will be useful to you? For which tasks?

|        |   |
|--------|---|
| User 1 | Yes. He thinks it will be useful for task management, finding old messages. |
| User 2 | He feels he does not need the task management features.                     |
| User 3 | Likes the mailbox view to track ongoing conversation.                       |
| User 4 | Retrieval of messages, task management.                                     |
| User 5 | She thinks the task feature is useful because it is simple to use.          |

4. Do you think you will be using the Calendar feature?

|        |   |
|--------|---|
| User 1 | No.   |
| User 2 | He will try to use it to see if it is useful. |
| User 3 | Yes.  |
| User 4 | Probably.                                     |
| User 5 | Yes.  |

5. Do you think you will be using the Tasks feature?

|        |   |
|--------|---|
| User 1 | Yes.  |
| User 2 | He will try to use it to see if it is useful. |
| User 3 | Yes.  |
| User 4 | Probably.                                     |
| User 5 | Yes.  |

6. Do you think you will be using the Notes feature?

|        |  |
|--------|--|
| User 1 | Not sure, because Eudora did not have this feature.                          |
| User 2 | Not really. He thinks it is easier to write things down on a piece of paper. |
| User 3 | No. He does not need it.   |
| User 4 | Probably not. Notes are usually task oriented.                               |
| User 5 | No. She thinks she does not need it.   |

7. Do you think you will be using the Address Book feature?

|        |   |
|--------|---|
| User 1 | Yes.  |
| User 2 | Yes. He thinks it is a good idea as oppose to Outlook where he has to explicitly create address book entries.                 |
| User 3 | Maybe.  |
| User 4 | Maybe. Thinks it would be more useful if she can type in the name instead of the actual email address when sending a message. |
| User 5 | She will use the address book, but she thinks it would be more useful if it was not restricted to just names of sender.       |

8. Do you think you will be using the Mailbox View feature?

|        |   |
|--------|---|
| User 1 | Not sure. Probably.                             |
| User 2 | Yes, for keyword searches.                      |
| User 3 | Yes, for tracking conversation with friends.    |
| User 4 | Yes.  |
| User 5 | May use mailbox view, she does not really know. |

9. Other specific comments on the interface, interaction styles, etc.

|        |  |
|--------|--|
| User 1 | Calendar and Tasks seems to be the same.   |
| User 2 | He likes the cute icons but he hates the colour scheme. He feels that TimeStore is a big gray thing on his screen. |
| User 3 | None.  |
| User 4 | Make more use of colour for the dots.  |
| User 5 | She likes the icons, especially the changing icons.  |

# Appendix C

## Individual User Daily Usage Data

---

### *User 1 (Pilot User)*

#### **Session 1**

Duration: 3 minutes 14 seconds.

He started by clicking on *Get New Mail* icon to check mail.

He then scrolled up and down the main display to try to look for the new messages. After a little while, he remembered about the inbox and proceeded to use it to find the new messages.

Note: an unreproducible bug in TimeStore was discovered. The new messages failed to appear on the main display. As a result, the user was not able to locate them without using the inbox.

He then changed the sorting order to sort by domain name for no reason other than to try it out. He looked at the names that did not have a domain name and commented that he did not know what the “()” represented.

He tried to change the name of a sender by double clicking on a name. TimeStore was not able to parse this sender’s name correctly into the last name and first name fields and both names was in the first name field while the initial was in the last name field. He re-typed the last name into the last name field but incorrectly deleted the first name portion in the first name field. He then clicked on the *Save* button and closed the address book dialog box.

He then tried to find the same name that he changed by scrolling vertically. He scrolled through the list a few times before he noticed that the first name was incorrect. He then double clicked on that name and corrected the first name.

### **Session 2**

Duration: 1 minute 34 seconds.

He clicked on the *Get New Mail* icon to get new mail.

He again scrolled vertically to look for new messages but decided to use the inbox instead since “it is probably faster if I used the inbox”.

He read 2 messages, then exited.

### **Session 3**

Duration: 54 seconds.

He checked for new messages.



He read 1 message, then exited.

A newer version of TimeStore was sent to this user. However, the user's preferences file was overwritten by mistake.

#### **Session 4**

Duration: 5 minutes 49 seconds.

He tried to check for new messages but could not. He gave up after 2 more tries.

He wanted to create a mailbox view for one of his mailing list. He scrolled vertically and found a sender whom he recognized as part of the list. He then scrolled back horizontally by clicking on the lower left arrow beside the timeline. Initially he did not release the button, hoping TimeStore would be able to automatically scroll back 1 day at a time. He discovered that TimeStore did not support such auto-repeat. He then clicked on the same button repeated and commented "This is kind of slow". He then remembered that "I can drag this" and drag the date ribbon backwards. He clicked on several dots belonging to the sender that he recognized and tried to find a common keyword. He then discovered that the subject lines to all those messages started with "Re: SRH:". He then noticed that TimeStore did not support "Cut and Paste" and commented "it would be a nice feature to have cut and paste".

After the mailbox view was created, he scrolled vertically and commented "now how do I know if all these people belong to this mailing list". He then singled clicked on a few names and checked the email address at the bottom. "I assume all these names belong to the mailing list. How do I merge all these names into one?" He scrolled a little more and said, "I guess I have to

manually merge all these names together". He merged a few and said, "I guess I'll save some for later since I don't have the patience to do all of them right now". He was "overwhelmed with all of these names" and thought it would be nice if there was a single command to merge all names in a mailbox view.

He then tried two more times to retrieve new mail. He then opened the *Preferences* dialog box and saw that his name has been replaced. He then fixed this problem but still could not connect to the mail server.

### **Session 5**

Duration: 1 minute 6 seconds.

He opened the inbox a number of messages containing jokes from one of his friend. He said "I ... get jokes but most of the time I don't read and I keep". He tried to retrieve new messages and was able to do so. He then commented "I ... have to start Camcorder before I start TimeStore or it might get buggy".

He opens the inbox and reads a few messages.

### **Session 6**

Duration: 2 minutes 7 seconds

He downloaded his new messages. He opened the inbox and read some mail.

### **Session 7**

Duration: 42 seconds

He experienced some problems with his DNS setup after a recent cleanup.

### **Session 8**

Duration: 2 minutes 12 seconds

The user found and used the IP address for the POP3 server and proceeded to use TimeStore again.

He created a task from within the message window. However, this task is for Friday and he commented he could not easily get the day of month from for use to set up the task.

### **Session 9**

Duration: 1 minute 10 seconds

He commented that he felt overwhelmed with the number of names, many of which are from mailing lists and are people he does not know or correspond. He felt that displaying the subject line instead of names might be more helpful.

### **Session 10**

Duration: 1 minute 43 seconds

The user tried to reorganize his TimeStore mailstore by reinitializing his TimeStore mailstore. He felt he had to because the number of names confused him.

### **Session 11**

Duration: 2 minutes 3 seconds

He created the same task as the one in session 8.

### **Session 12**

Duration: 1 minute 40 seconds

User opened the inbox to get new messages.

User discovered the feature where in a listbox he could drag left and right to see hidden text on either side.

He read a few messages from the inbox and exited.

### **Session 13**

Duration: 1 minute 32 seconds

User created a task by clicking on the *New Task* button on the toolbar. He then drag the dialog box down a little so that he can see the timeline in the main window underneath.

User then clicked on a few dots on the task line to check upcoming tasks.

### **Session 14**

Duration: 4 minutes 16 seconds

Bug with task line redraw after deleting the last task discovered. The user then cleaned up some tasks in the past by deleting them.

Movie file error encountered. Although it was possible to skip the error, the movie screen was not fully redrawn. User was creating a task but had to rely on a paper calendar to enter in the correct date. User also reported that it was inconvenient to have to hold a microphone with one hand. It affected his typing.

### **Session 15**

Duration: 1 minute 2 seconds

When the user was scrolling the time line by dragging the date ribbon, he commented that the dragging was not easy to control.

### **Session 16**

Duration: 1 minute 48 seconds

The status (in progress/complete) was not used by this user. He just deleted the task when he has finished it.

**Session 17**

Duration: 1 minute 56 seconds

TimeStore's tasks may move out of view and there are tasks when it may not be possible to attach a due date to.

**Session 18**

Duration: 2 minutes 16 seconds

User commented that TimeStore has not "outbox" and although he could find all of his outgoing messages, the information about who the recipient was is hidden in the message and he must open the message to find out. User attempted to create a mailbox view for the outbox but he still could not see the recipient of the message.

**Session 19**

Duration: 1 minute 24 seconds

User created a task from the message window after reading it.

**Session 20**

Duration: 1 minute 1 second

User had no new message but started to comment on a recent TimeStore crash. He said he had two new messages but the system crashed immediately and he was not able to save the movie.

**Session 21**

Duration: 2 minutes 47 seconds

User downloaded messages. He then opened a message from the main display because he recognized the red dot and forwarded the message to another person.

**Session 22**

Duration: 43 seconds

User downloaded 1 message. He then opened and read the message from the main display because he recognized the red dot.

**Session 23**

Duration: 1 minute 30 seconds

User checked messages. User then opened the inbox to read these messages.

**Session 24**

Duration: 2 minutes 39 seconds

User downloaded new messages. User then opened the inbox to read some of these messages.

***User 2***

*User's hard disk was full and was consistently causing Camcorder to corrupt movie files. User data was mainly gathered from weekly interviews.*

**Session 1**

Duration: 1 minute 4 seconds

User downloaded some messages by clicking on the "Get new mail" button on the toolbar. He then browsed through his new messages from the main display. The user then scroll forward by clicking on the top right arrow beside the timeline. He then checked his calendar about things he had to do.

**Session 2**

Duration: 3 minutes 32 seconds

User downloaded some messages by clicking on the “Get new mail” button. He then browsed through a few messages from the main display. Movie file was corrupted at this point.

### *User 3*

For some unknown reason, the audio capture was not clear and occasionally the user’s speech was not audible. Questions about the data were asked during the next data collection session.

#### **Session 1**

Duration: 6 minutes 33 seconds

TimeStore downloaded all of his 174 messages on the server (he only deletes the messages of the server once a week). On a slow dialup link (33.6kps), this took about 6 minutes. User decided to stop and save the movie file first before continuing because TimeStore caused a system crash in a previous session.

#### **Session 2**

Duration: 10 minutes 46 seconds

User spent this session merging names together. User spent about 5 minutes to merge names together but could not manage to merge all of the names he wanted. He “will do the rest later”. User then created a mailbox view using a keyword to track an ongoing discussion with 2 other people. He commented “I liked that”. He then created another mailbox view with a friend’s name in the “from” field. He commented that the TimeStore interface makes it more difficult to find all messages from a single sender.

He uses mailbox view to find the person that he was searching for rather than just scroll and find the name in the main view. [Possible need for improved name manipulation and filtering] TimeStore then crashed (part of the screen in the movie file was corrupted and it was not possible to determine the exact condition when TimeStore crashed.)

### **Session 3**

Duration: 4 minutes 56 seconds

User has subscribed to a mailing list and he tried to build a mailbox view using the keywords “ford probe”. However, the user clicked on the “Or logic operator” and thought that it Or operation was to be applied to the 2 words. User then changed the keyword to just “probe”.

### **Session 4**

Duration: 2 minutes 41 seconds

User commented that he would like to see his messages on a per day basis and be able to distinguish between a.m. and p.m. In addition, he commented that he would like to be able to go to the “next” message in a conversation instead of having to close the message window, search for the next message then open the message.

### **Session 5**

Duration: 6 minutes 11 seconds

User tried to create a mailbox view (as in Session 3) but was unsuccessful. He then went back to the mailbox view created in Session 3 and browsed through a number of messages. He then changed back to the main view and browsed through some more messages. He then created another mailbox view about “hockey pool” and browsed through the result. There were many messages in the view that was unrelated to the search.



**Session 6**

Duration: 11 minutes 42 seconds

User scrolled down the main display. User avoided the inbox because he does not read all of the messages from mailing list and they are overwhelming the inbox. He read a few messages. User commented that he wanted the ability to delete all messages from a mailbox view. User created a mailbox view and proceeded to delete all messages from that view.

User scrolled up and down the main display looking for a person who he could not recall the work [email] address. He needed to send this person a message. He gave up after about 2 minutes of scrolling. He then browsed through messages from a person.

**Session 7**

Duration: 5 minutes 7 seconds

User downloaded new messages. Again, user complains about TimeStore inability to keep POP3 states and download only new messages. Again, user scrolled down the main display to look for new messages. User browsed through some messages.

**Session 8**

Duration: 5 minutes 33 seconds

User downloaded new messages and scrolled down the main display to look for new messages. This time the user scrolled horizontally until "today" is at the left edge. He then browsed through the new messages. In this session, an unknown problem in either the server or TimeStore caused all messages on the server to appear as new even though most of them have already been downloaded. User continued to browse through new messages but TimeStore crashed.

**Session 9**

Duration: 1 minute 38 seconds

User created a mailbox view for a single user because he does not like to search for the name in the main display. User commented that it was okay and switched back to the main view. He then opened the inbox and proceeded to scroll down the list. However, TimeStore crashed unexpectedly and the session was terminated.

**Session 10**

Duration: 47 seconds

User started another session after the last crash. He opened the inbox and performed the same operation that caused the crash. TimeStore did not crash this time and the user was able to scroll all the way down the bottom of the inbox. However, TimeStore crashed when he scrolled up a little.

**Session 11**

Duration: 3 minutes 59 seconds

User commented that TimeStore crashed in two previous sessions when he was viewing the inbox. He then created a mailbox view about a topic that he was interested in a mailing list. User “thought [he] had more messages” in the mailbox view. He then opened a message but said, “I guess I wasn’t being specific enough in my query” because he could not find what he was looking for. At this point, another one bug appeared. TimeStore did not display the cascade menu that let the user choose mailbox views 11 to 20. However, user did not make a comment immediately. He move the mouse over a few dots, then tried to change the mailbox view but could not. He finally made the comment. He then shutdown TimeStore and restarted but the bug persisted. He then opened the inbox and browsed through the

messages. He then closed the inbox and tried changing the mailbox view again but could not. He gave up and shutdown TimeStore.

### **Session 12**

Duration: 6 minutes 28 seconds

User said he wanted to clean up his mailbox and also wanted to know if any other programs that he was using was causing TimeStore to crash. He scrolled down the main display first looking for new messages. User also noticed TimeStore was downloading duplicate messages and he showed a few cases. He opened and read another message, then scrolled down further. He then selected another dot but he seems hesitant in choosing which message to read. After about 15 seconds of inactivity, he finally chose to read a message.

He then noticed an email address of a person he emailed frequently but there were not any dots. He then remembered that was an old email address. He then scrolled up and opened another message. User continued to scroll up and browsed through recent messages.

### **Session 13**

Duration: 58 seconds

TimeStore crashed in a previous session when he was just scrolling up and down in the inbox. (The inbox had over 700 items in it). User wanted to show how it crashed in a previous session, which he did not record. User opened the inbox and scrolled up and down and again, TimeStore crashed.

### **Session 14**

Duration: 16 seconds

User wanted to try again to see if the inbox consistently crashed TimeStore. He opened the inbox and scrolled down a little. TimeStore then crashed.

### **Session 15**

Duration: 4 minutes 34 seconds

User commented TimeStore consistently crashes when he is in the inbox and he is going to avoid the inbox. User then wanted to clean up his mailbox. He selected a dot and proceeded to delete messages from the bottom list. At first, he used the menu function to delete. After three tries, he switched to using the keyboard shortcut (<Del>). He continued to clean up his mailbox, sometimes opening a message to look at the content. User commented that he needed a more efficient way to clean up his mailbox. He mentioned it would have been easier to clean up if the inbox was working properly.

### **Session 16**

Duration: 1 minute 41 seconds

User is unable to access mailbox views 11 to 20. He looked at the mailbox views that were accessible. He then opened the inbox again and TimeStore crashed. User stopped recording.

### **Session 17**

Duration: 5 minutes 32 seconds

User browsed through the main display and read new messages. Again, TimeStore was downloading duplicate messages and the user opened a few messages and confirmed that the messages were old by looking at the time the message arrived at the server. User then moved today to the left edge and continued to read new messages. He notices more duplicate messages. The session ended because the telephone rang.

## ***User 4***

*Usability test not complete. More data to follow...*

### **Session 1**

Duration: 6 minutes 1 second

User noticed that TimeStore displayed a dot but there were no messages in the system yet. She entered her personal information in the preferences dialog box. In the POP3 server name, she made a mistake typing in machine@domain.name instead of machine.domain.name. She then checked for new mail but could not. She then returned to the preferences dialog box and fixed the error. She had messages from a few days sitting on the server but TimeStore displays all to be on the same day (user forgot TimeStore uses time received rather than time sent). She questioned how the messages all appear on the same line.

She then scrolled up and down a bit. She could not determine which of the messages were new and which were not. She worried that old messages will get mixed up with messages that will arrive tomorrow and TimeStore will show them all as new messages the next day. She then gave up and exited.

### **Session 2**

Duration: 22 minutes 2 seconds

User started up TimeStore. She then immediately started to search for new messages before selecting "Get new mail". She then realized this and got new mail. User then browsed through her new messages by going down the main window and opening all her mail.

User wanted to reply to a message and commented it would be nice to have a reply button in the message window. She commented that she should open all her messages. She then questioned about would happen with a double click on a dot. She thought it might be a good idea that a double click will bring up a message especially for dots with only 1 message.

She then created a mailbox view with a few names in the “from” field. However, she forgot to give it a meaningful name and she tried to figure out how to change the name of a mailbox view. Also she typed in three names but only 2 names showed up in the result. She thought her spelling was incorrect and she was checking against the name in the main window. (She forgot TimeStore’s search is case sensitive.) She then tried to open a message window to look at the exact spelling in the message but could not since the mailbox view dialog box is application modal. She commented that TimeStore should somehow signal the user.

She then created another mailbox view. She commented that “I guess you can only have one view up at one time.” User then created a calendar event. She then tried switching between different mailbox views to determine what happens with the event when she changed views.

She then created another calendar event, typing in “06” rather than “6” for June. TimeStore was supposed to warn the user but it didn’t and the user was lost. She searched for the event but couldn’t find it. She also question the meaning of the blue dot because the calendar events appear in blue but she “hasn’t open them yet”.

### **Session 3**

Duration: 6 minutes 22 seconds

User downloaded new messages. She then opened a message which she remembered to contain an attachment. TimeStore took roughly 15 seconds to open that message. She commented that the windows always open on top of each other and she always had to pull it back.

A bug was discovered. The user created a calendar event but it caused other events to disappear. User also commented that it would be better to have a darker line to separate the top and middle sections of the window.

#### **Session 4**

Duration: 7 minutes and 12 seconds

User downloaded new mail. She then browsed through the new messages.

#### **Session 5**

Duration: 6 minutes 18 seconds

User downloaded new mail. She then browsed through her new mail from the main display. She commented that it would be better to show outgoing messages under the names of the sender, rather than in her name. User commented that the "Get new mail" and the inbox icons are too similar and confusing.

#### **Session 6**

Duration: 5 minutes 1 second

User downloaded new messages and proceeded to read her messages from the main display. User then reply to a few messages from the message window. User commented that the message that she "cc" to herself should be displayed in blue rather than as a unread message. User also commented that the icon representing the taskbox is not clear.

**Session 7**

Duration: 1 minute 47 seconds

User commented that most email systems she have used do not require her to explicitly check for new messages. She then downloaded and read the new messages from the main display.

**Session 8**

Duration: 17 minutes 34 seconds

User downloaded new messages. Again, she read them from the main display. User then replied to a few messages. User commented that there should be a reply option where TimeStore does not automatically copy the content of the original message into the reply message. User commented that she needed to lookup previous mail that was not available in TimeStore. User also sent a few new messages.

***User 5***

*User's machine had limited RAM. When she used Camcorder, her system was very slow. As a result, user avoided using Camcorder. Most of the data were from the first week of usage.*

**Session 1**

Duration: 5 minutes 57 seconds

User typed in a message and complained about the message window cutting off words in the middle at the end of each line. Also after the user sent the original message, she remember she forgot to say a few things. She successfully retrieved the incoming message that she replied to by scrolling down the main window and searching for the name. She double clicked on the message and immediately started typing on the message window.



TimeStore did not prevent her from changing the original message and she did not notice that she had to “reply” first before typing.

User then proceeded retype the message. She complained about the lack of cut and paste feature.

**Session 2**

Duration: 1 minute 42 seconds

User replied to another message. She did not like having to delete the original quoted message when replying.

**Session 3**

Duration: 5 minutes 43 seconds

User read and replied to a message. User stopped Camcorder in the middle of the composition.

**Session 4**

Duration: 45 seconds

User complained that outgoing messages were not displayed properly on some mailers. User did not perform any action in this session.

**Session 5**

Duration: 2 minutes 50 seconds

User did not have any new messages but she was confused since the inbox was not empty. She expected the inbox would only contain new messages. She read the message and closed the inbox. However, the icon on the main display failed to change and she checked the inbox again.

**Session 6**

Duration: 26 seconds

User complained about TimeStore was not able to download mail while her “backup” system had download some messages. Again, user did not perform any action in this session.

**Session 7**

Duration: 44 seconds

User wanted to delete a message from the inbox. She opened the inbox, then selected the first message and chose delete from the menu.

# Appendix D

## User Comments from Informal Discussions

---

### *User 1 (Pilot user)*

#### **After Initial Interview**

- How do you delete every message in a mailbox? TimeStore can not do that.
- Thinks the ability to associate a message with a task would be helpful. Task feature can remind him of things to do.
- Commented that the mailbox view would be helpful for mailing lists that have a subject line that is more or less the same with every message.
- The ability to merge names in the main display was useful since he has to set up multiple filters to deal with people with multiple email addresses.
- He felt he should have no problems using Camcorder to record his thoughts. However, he cautioned that some of his friends do not like to talk to a machine and may not talk about the program if they were in this study.
- Calendar and Tasks seems to be the same.

- The different colours in the timeline were not obvious to this user. He did not see that the red represents weekend and green represented today.
- Feels TimeStore is a novelty, but appreciates the overview. Can see trends in correspondence.
- Attachment is necessary for him to be able to give up Eudora.
- Not sure if he will use the notes feature because Eudora does not have that.
- Has trouble adjusting to the new icons in TimeStore. Also must adjust to the timeline view.
- Asked about merging multiple names together at once. (Not supported)
- He tried to change the email address field because his friend has informed him of an address change. Possible need for better handling of the database in the address book.
- He tried to use TimeStore to count the total number of messages. He forgot TimeStore can show the number of messages for each sender if he switch the sorting order.
- Asked about displaying by subjects rather than just names in the main display.
- He tried to click on column labels to change the sorting order in the inbox window. However, TimeStore do not allow the different sorting order.
- He questioned whether the inbox could be cleared. The automatic removal of messages that have been read was not obvious to him at the beginning.
- He commented that multiple selection of messages was not possible in TimeStore and would be useful.

#### **Interview 1 (end of week 1)**

- Missing features caused the user the to alter the normal way he uses email. User noted that the addition of Cut and Paste would aid him much more.
- User did not talk aloud about what he was doing. He said that he had to constantly remind himself that he is making a movie and someone else

will be observing the movie, but “some things comes so naturally it wasn’t worth mentioning”.

- User does not like to have to click on several dots to find the correct message. User is unable to recall the exact date the message arrived (User was close).
- User does not like to use TimeStore as much because it lacks attachments as well as cut and paste capabilities.
- User liked the task management feature.
- The novelty of seeing all his correspondence is beginning to wear off. Missing features are becoming more annoying
- User does not have to use the mailbox view feature all that much because his categorizations in Eudora were based on senders. Such categorization is automatic in TimeStore.
- Mailbox view tool is not intuitive.
- User wanted way to group names together.
- User did find the inbox which automatically cleared out useful. However, when the user do not have time to read mail, the inbox grows and it would be nice to remove these messages from the inbox without actually opening them. (Distinction between new mail and unread mail)
- User commented on the ability to track the amount of time the user spends on each sender would be useful.
- User thought that explicit action to record all screen interactions has put pressure on him when using TimeStore. He feels that he has an obligation to make interesting movies.
- User also thought the amount of disk space was limited and he was afraid that he would run out of space before completing his tasks.
- Needs more frequent interaction with experimenter to keep motivated.
- TimeStore’s lack of support for attachments (feature that the user has taken for granted) is taking its toll.

#### **Interview 2 (end of week 2)**

- User did not want to record him writing email. User commented that Camcorder is missing a pause button.

- User commented that he spent most of his in the inbox and he was starting to ignore the main window unless he has to find things.
- User commented that he did not like the outbox in his name in the main display. He also he wanted to be able to see who he sent the email to.
- TimeStore forces events to be associated with a particular date but certain tasks cannot be associated with a specific date and the user did not know when the task will be finished.
- User had problems with tasks because tasks do not always require a due date and it becomes a hassle to have to figure out the due date.
- User does not like to use notes because it moves off the display along with the time line. User would rather use tasks to store information.

## *User 2*

### **After Initial Interview**

- User did not feel comfortable with MS Camcorder because of privacy concerns.
- User feels that the Calendar should be displayed in a table rather than on a timeline.
- User feels that the colour scheme is not very interesting. He feels that in Outlook, the different background colour in different windows groups those functions together. TimeStore is a big gray thing.
- He does not feel that he'll use any of the scheduling functions. So he thinks it won't be that useful to him.

### **Interview 1 (end of week 1)**

- Hardware incompatibility. He has a TV window and Camcorder becomes very slow. User did not like Camcorder.
- User's boot drive had very limited space and is limiting the size of the movie he can make. In addition, it is causing corrupt movies.
- User likes the speed of TimeStore. He said Exchange was very slow.
- User feels TimeStore is taking up too much screen real estate.

- User is used to seeing a list of messages and cannot adjust to seeing dots.
- User prefers to see the colour code extended to the message list as well.

### **Interview 2 (end of week 2)**

- User liked the simplicity of TimeStore but insisted that he still preferred to use Exchange since all his previous messages are there.
- User felt TimeStore is taking up too much screen real estate.
- User is used to seeing a list of messages.
- User feels that the Calendar should be displayed in a table rather than on a timeline.
- He only complained about cosmetic things (like non-standard MS controls, etc)
- He did not like to have to scan down to the bottom to find a subject field. He much rather be able to see the subject beside the name so that he can skip certain messages. He knew the inbox provides this feature but did not like to have to open it separately.
- TimeStore is inferior to Outlook for a number of reasons. TimeStore does not support auto-preview. He also does not like seeing the “time received”. He much prefer to see time sent because he feels that knowing when a co-worker sent him mail (or a co-worker knowing when he sent mail) is more important than knowing the time received.

### ***User 3***

#### **After Initial Interview**

- Likes the fact that he does not have to sort email.
- He was able to see trends in his correspondences.
- Thinks the mailbox views are useful for tracking conversation threads with multiple senders. He just search for a keyword and all people in the conversation appears in the view .

### **Interview 1 (end of week 1)**

- Wanted the difference between “new” messages and unread.
- He preferred TimeStore would only download new messages.
- He subscribed to a mailing list again. He thinks the names from the mailing list are cluttering the system.
- He felt that when changing a view’s properties, the messages that were already part of the view should be kept.
- TimeStore still crashed once in a while.
- He did not send mail with TimeStore because he wanted to keep all his outgoing messages on his regular email program.
- If was not possible to cancel mail download. In addition, TimeStore sometimes did not redraw properly.
- He needed attachments!
- TimeStore could use better address book support. He wanted to be able to click on “cc” and list all people in the database.
- He thought the ability to propone sending messages would be useful.
- Likes the clean simple interface.

### **Interview 2 (end of week 2)**

- Used mailbox views to get mailing list message of interest because he did not read a majority of the incoming messages.
- Inbox was not useful because he did not read a lot of mail.
- Likes everything is already sorted for him.
- User can handle mailing list now since TimeStore is able to extract messages interesting to him.
- User did not want to have to scan up to see the date for a dot.
- TimeStore was downloading duplicate messages and stating it as new.
- If TimeStore has attachments and “more bells and whistles”, he will use it over his regular mailer.



- Able to change the width of the name list. Distinguish between new and unread messages.

#### ***User 4***

##### **After Initial Interview**

*User did not have additional comments after the initial interview.*

##### **Interview 1 (end of week 1)**

- User's email volume was below average because the courses that she teaches were finished.
- User suggested there should be a better visual separation between the top section and the middle (message) section.
- Thought colour was not consistent throughout because she thought red represented unread messages while blue represented read messages.
- User also commented TimeStore lacks shortcuts for common tasks. A suggestion was to allow the user open the first message of a dot by double clicking. Also keyboard shortcuts are needed.
- Liked the visualization, but preferred to see calendar and tasks merged into one.
- Some menu functions such as reply should have been available in the toolbar as buttons.

##### **Interview 2 (end of week 2)**

*This user only used TimeStore for 2 weeks due to delays in moving and setting up her PC in a more area in the computer lab. The interview for Week 2 was considered as the final interview, which are presented in Appendix E.*

## *User 5*

### **After Initial Interview**

- Thought using different colours to highlight different things (like event, messages) would help.
- Thought the dots were confusing. She preferred to see the name and subject fields.
- Thought the program is confusing because it is different from Netscape.
- Liked the list of sender sort by last name, thinks it is very neat and not messy.
- Fonts were very small. Should be able to change the colour.
- She had to locate the date, then look down in order to find the dot. She preferred if she could just “know” what date the dot fell on
- Liked the changing icons. Thought program should generate sounds to alert the user about new mail, etc.

### **Interview 1 (end of week 1)**

- TimeStore / Camcorder performance was too slow. (Problem due to having insufficient RAM). She also thought the entire procedure was too cumbersome. (had to remember to start Camcorder, then TimeStore although both programs appear on the desktop).
- Liked TimeStore because it could tell her immediately how many messages there are.
- She liked the integration of tasks with her email. However, she preferred to see a table form calendar rather than a timeline.
- The procedure for reading a message in TimeStore was too cumbersome. In Netscape Navigator, she could just click on a subject and see the message immediately (no double clicking needed).

## **Interview 2 (end of week 2)**

- User did not like to use TimeStore only because the performance is terrible on her machine. User did not record any movies because she was doing the same thing as last week. User kept using TimeStore though.
- User did not like to have to close a message window after she has replied to a message. She thought that window should have been closed automatically.
- User did not have any tasks to track because her university was on strike and she did not really have anything important to track. She also felt that it was not worth the effort to use TimeStore to track them.
- User thought TimeStore's placements of the various windows (main, inbox, message windows) were not very organized when compared to Netscape. In Netscape, everything she needs is displayed neatly in the window and nothing is hidden.
- User also complained that date field in a message window should be displayed in text (Apr. 12, 1997) rather than just number (04/12/97)
- User also missed that conversation tracking capabilities in Netscape. The last message in the conversation is just one click away.

## Appendix E

# Final Interview Results

---

1. Did you like using TimeStore?

|        |   |
|--------|---|
| User 1 | Yes, only after cut and paste was added.      |
| User 2 | No, he could not adjust to the timeline view. |
| User 3 | Yes.  |
| User 4 | Yes.  |
| User 5 | Yes.  |

## 2. What did you like?

|        |   |
|--------|---|
| User 1 | Task management.  |
| User 2 | User liked the simplicity of the TimeStore UI but insisted that he still prefers to use Outlook since all his previous messages are there and he is used to Exchange / Outlook.   |
| User 3 | He liked the automatic sorting by name. He also liked the mailbox view because it actually allowed him to subscribe to a mailing list. He likes the overall time-based visualization concept.   |
| User 4 | She liked the visualization and the ease of use.  |
| User 5 | She liked the task feature. User liked the fact that accessing past mail was a lot easier than Netscape Navigator. However, the user still hated the dot. She preferred to see icons. She also liked the consistent alphabetical order of the y-axis. |

### 3. What did you dislike?

|        |  |
|--------|--|
| User 1 | He felt the number of names on the display has overwhelmed him. He preferred to use the inbox to view messages. He also felt the separation into days slowed him down because he needed to see the subject field. Tasks were sometimes awkward because TimeStore forced him to assign a due date but his tasks did not always need a due date.   |
| User 2 | He did not like to have to scan down to the bottom to find a subject field. He much rather be able to see the subject beside the name so that he could skip certain messages. He knew the inbox provided this feature but did not like to have to open it separately.  |
| User 3 | No easy way to erase many messages. Did not like the inbox because it did not group by name. Lack of attachments. Icons could have been better. Sometimes TimeStore would download a message twice and calls it new.   |
| User 4 | Window management was a problem. She always had to move a window when it appears. Outgoing messages should be shown with the name of the recipient. She also thought that the mark feature should be expanded to use different colours and allow the user to choose what the colours represent.  |
| User 5 | User still hated the fact that she has to manually check mail. User's inbox was full because she gets a lot of junk mail (mail she did not want to read). Netscape Navigator is faster for browsing messages. User also wanted a better way to show the name and date of a dot without actually moving the mouse into the dot. She suggested TimeStore could use a grid, similar to a spreadsheet. |

4. Do you like the time-based concept? Was it useful?

|        |   |
|--------|---|
| User 1 | User had trouble remembering the exact week a message arrives but can recall the month. He especially liked that he can see upcoming tasks on the timeline. He also liked that TimeStore could also remind him to contact certain people. |
| User 2 | He did not like the horizontal timeline (he preferred to scroll vertically rather than horizontally.) Actually, he preferred to see the calendar in a table form.   |
| User 3 | Yes. He found it useful.  |
| User 4 | Yes. She found it useful. However she questioned her ability to assess the "usefulness" of time-based visualization given such a short study duration.  |
| User 5 | Yes. She found it useful.   |

5. Did you find the calendar feature useful?

|        |   |
|--------|---|
| User 1 | No, he only used tasks because he thought they were the same.   |
| User 2 | Did not like the calendar. He insisted on seeing the calendar in table form. He feels a table calendar could help him to remember about the things that he has to do. |
| User 3 | Yes.  |
| User 4 | Yes.  |
| User 5 | No.   |

6. Did you find the task feature useful?

|        |   |
|--------|---|
| User 1 | Tasks were very useful.   |
| User 2 | No. He did not normally use an electronic organizer and could not start to use one just for three weeks.  |
| User 3 | Yes.  |
| User 4 | Tasks could have been useful, but she did not have a use for it. She commented that if she had access to the computer all the time she will probably use it more. |
| User 5 | Kind of. She likes it but it was not that useful because she was away from her computer most of the time.   |

7. Did you find the association of messages with tasks useful?

|        |  |
|--------|--|
| User 1 | He liked the ability to associate messages with tasks. |
| User 2 | N/A.   |
| User 3 | Yes.   |
| User 4 | Never had to use it.                                   |
| User 5 | Never had to use it.                                   |

8. Did you find the notes feature useful?

|        |   |
|--------|---|
| User 1 | He did not find the notes feature useful because they move off the display with the timeline and preferred to see all notes in one place. |
| User 2 | No.   |
| User 3 | No. He did not have a use for it.   |
| User 4 | No.   |
| User 5 | No.   |



9. Did you keep persistent mailbox views? Why or Why not?

|        |  |
|--------|--|
| User 1 | He did not find the mailbox views useful mainly because he can usually find the messages he needs from the main display. |
| User 2 | User did not use mailbox views at all.   |
| User 3 | He used the mailbox view both as a search tool and to extract useful information from his messages.                      |
| User 4 | Yes. To filter names.  |
| User 5 | User did not use mailbox views.  |

10. What would make TimeStore better?

|        |   |
|--------|---|
| User 1 | Ability to check multiple email accounts, attachments. He thought notes would be more useful if they can be associated with a sender. In addition, he wanted TimeStore to track how much time he spends on each sender. |
| User 2 | N/A.  |
| User 3 | Fix the technical problems with TimeStore. Better support for deleting messages. Ability to queue messages and a spell checker.   |
| User 4 | Include attachments, fix window management function, etc.   |
| User 5 | Fix up the technical bugs. HTML formatted text.   |

11. If TimeStore was built with all the features that most email systems have (better SMTP/POP3 support, mailing list support, MIME / BinHex decoding, etc), would you keep using TimeStore?

|        |   |
|--------|---|
| User 1 | Yes. In fact, he will replace Eudora with TimeStore.                              |
| User 2 | No.   |
| User 3 | Yes. He would replace Outlook with TimeStore if his company did not use Exchange. |
| User 4 | Yes.  |
| User 5 | Yes. She would use it over Netscape.  |

# References

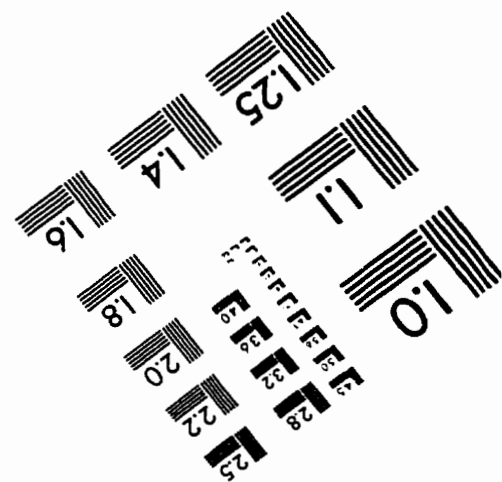
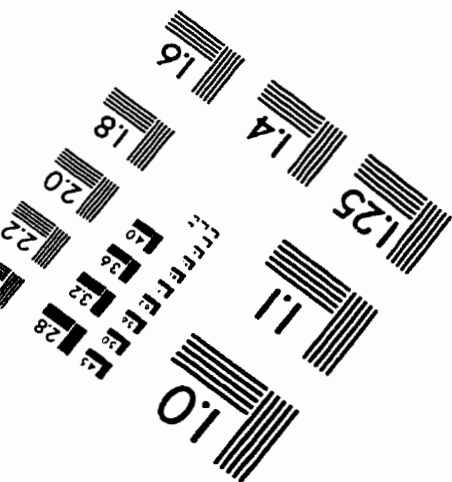
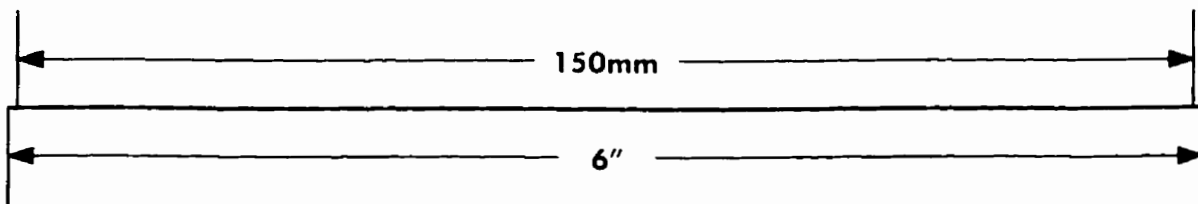
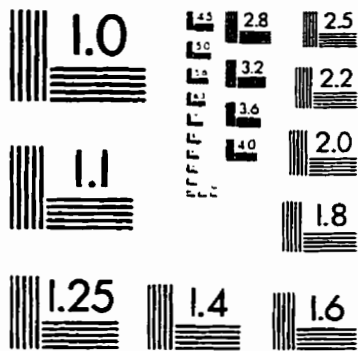
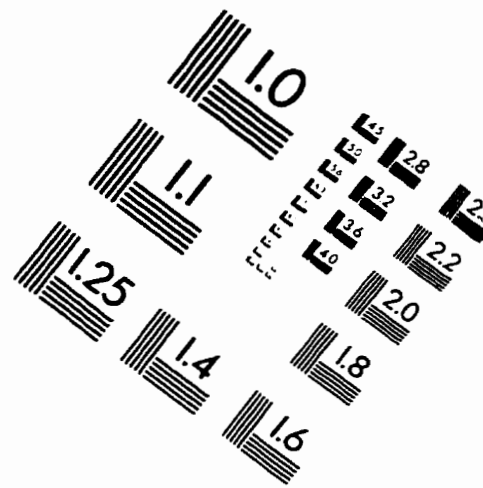
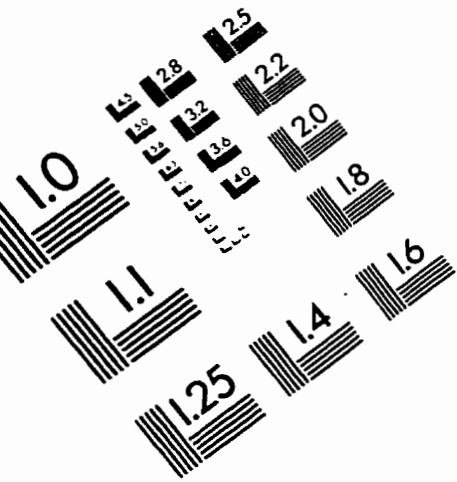
---

- Altman, E., Larkin, J. H., and John, B. E., "Display Navigation By An Expert Programmer: A Preliminary Model Of Memory." *Proceedings of CHI '95*, April, 1995. Online Document, URL: [http://www.acm.org/sigchi/chi95/Electronic/documnts/papers/ea\\_bdy.htm](http://www.acm.org/sigchi/chi95/Electronic/documnts/papers/ea_bdy.htm)
- Barreau, D. and Nardi, A., "Finding and Reminding: File Organization from the Desktop." *SIGCHI Bulletin*, July, 1995, pp. 39-43.
- Berlin, L. M., Jeffries, R., O'Day, V. L., Paepcke, A., and Wharton, C., "Where Did You Put It? Issues in the Design and Use of a Group Memory." *Proceedings of INTERCHI '93*, April 24-29, 1993, pp. 23-30.
- Brewer, W. F., "What is Autobiographical Memory?" In D. C. Rubin (Ed.) *Autobiographical Memory*, Cambridge University Press, Cambridge NY, 1988.
- Byrne, M., "Using Icons to Find Documents: Simplicity is Critical." *Proceedings of INTERCHI 93*, April, 1993, pp. 446-503.
- Carriero, N., Fertig, S., Freeman, E., and Gelemter, D., "The "Lifestreams" Approach to Reorganizing the Information World." Yale-CS TR 1070, submitted to *Technology Review*, April, 1995.
- Conway, M. A., *Autobiographical Memory: An Introduction*. Open University Press, Milton Keynes, 1990.
- Craik, F. I. M., and Watkins, M. J., "The role of rehearsal in short term memory." *Journal of Verbal Learning and Verbal Behavior*, 1973, pp. 599-607.
- Ericcson, K. A., and Simon, H. A., *Protocol Analysis Revised Edition*, The MIT Press, Cambridge MA, 1993.

- Erickson, T. and Saloman, G., "Designing a Desktop Information System: Observations and Issues." *Proceedings of CHI '91*, 1991, pp. 49-54.
- Fertig, S., Freeman, E., and Gelernter, D., "Finding and Reminding" Reconsidered." *SIGCHI Bulletin*, January, 1996, pp.66-69.
- Fischer, G. and Stevens, C., "Information Access in Complex, Poorly Structured Information Spaces." *Proceedings of CHI '91*, 1991, pp. 63-69
- Fitzmaurice, G., Baecker, R., and Moore G., "How Do People Organize their Computer Desktop?" Unpublished report, University of Toronto, 1994.
- Freeman, E. and Fertig, S., "Lifestreams: Organizing You Electronic Life." *AAAI Fall Symposium: AI Applications in Knowledge Navigation and Retrieval*, November, 1995.
- Friedman, W. J., "Memory for the time of past events." *Psychological Bulletin*, 113, 1993, pp.44-66.
- Gleitman, H. *Psychology Third Edition*, W. W. Norton & Company, Inc., New York, 1991, pp. 259-283.
- Kandogan, E., and Shneiderman, B., "Elastic Windows: Improved Spatial Layout and Rapid Multiple Window Operations." University of Maryland Technical Report CS-TR-3522, September, 1995.
- Larsen, S. F., Thompson, C. P., and Hansen, T., "Time in Autobiographical Memory." In D. C. Rubin (Ed.) *Remembering Our Past*. Cambridge University Press, Cambridge NY, 1996, pp. 129-156.
- Long, B., "TimeStore: Exploring Time-Based Filing." University of Toronto Unpublished Paper, May, 1994.
- Malone, T. "How Do People Organize Their Desks? Implications for the Design of Office Information Systems." *ACM Transactions of Office Information Systems, Vol. 1, No. 1*, January 1983: 99-112
- Microsoft Corp, Camcorder, 1997. URL: <http://www.microsoft.com/msoffice/office97/camcorder>.
- Microsoft Corp, Outlook, 1997. URL: <http://www.microsoft.com/outlook>.
- Nardi, B., Anderson, K., and Erickson, T., "Filing and Finding Computer Files." Apple Computer Inc. Advanced Technology Group Technical Report 118, July, 1995.

- Netscape Communications, Navigator, 1996. URL: <http://home.netscape.com>.
- Plaisant, C., Milash, B., Rose, A., Widoff, S., and Shneiderman, B.,  
“LifeLines: Visualizing Personal Histories.” *Proceedings of CHI'96*,  
1996, pp.221-227.
- Posner, I., Baecker, R., and Mitchell, A., “Evaluating REAL Users, using  
REAL Software, performing REAL Tasks, in REAL Contexts.” *To  
appear in proceedings of HCI International '97 – Conference on Human  
Computer Interaction*, 1997.
- Qualcomm Inc., Eudora Pro, 1997. URL: <http://www.eudora.com>.
- Rundus, D., “Maintenance rehearsal and single-level processing.” *Journal of  
Verbal Learning and Verbal Behavior*, 1977, pp. 665-682.
- Schooler, J. W. and Herrmann, D. J., “There is More to Episodic Memory  
than just Episodes.” In M. A. Conway, *et al* (Eds.), *Theoretical  
Perspectives on Autobiographical Memory*, Kluwer Academic,  
Dordrecht, The Netherlands, 1992, pp. 241-262.
- Shneiderman B., and Plaisant, C., “The Future of Graphic User Interfaces:  
Personal Role Managers.” University of Maryland Technical Report CS-  
TR-3285, May, 1994.
- Silver, N., “Time-Based Visualizations of Electronic Mail.” University of  
Toronto, Department of Computer Science M. Sc. Thesis, 1996.
- Steinberg, S. “Lifestreams.” *Wired*. Feb, 1997.
- Sun Microsystems Laboratories, Tcl /Tk, 1997. URL <http://www.sunlabs.com/tcl>.
- Thompson, C. P., Skowronski, J. J., Larsen, S. F., and Betz, A. C.,  
*Autobiographical Memory: Remembering What and Remembering When*.  
Lawrence Erlbaum Associates, Publishers, Mahwah New Jersey, 1996.
- Thompson, C. P., Skowronski, J. J., and Lee, D. J., “Reconstructing the date  
of a personal event.” In M. M. Grunberg *et al* (Ed.), *Practical aspects of  
memory: Current research and issues* (Vol. 1), Academic Press, New  
York, 1988, pp. 241-246.
- Whittaker, S. and Sidner, C. “Email Overload: Exploring Personal  
Information Management of Email.” *CHI'96 Conference Proceedings*,  
1996, pp. 276-283.

# IMAGE EVALUATION TEST TARGET (QA-3)



APPLIED IMAGE . Inc  
1653 East Main Street  
Rochester, NY 14609 USA  
Phone: 716/482-0300  
Fax: 716/288-5989

© 1993, Applied Image, Inc., All Rights Reserved