Understanding Sequence and Reply Relationships within Email Conversations: A Mixed-Model Visualization[1]

Gina Danielle Venolia
Microsoft Research
One Microsoft Way
Redmond, WA 98052 USA
+1 425 703 2891
ginav@microsoft.com

Carman Neustaedter
University of Calgary
Department of Computer Science
Calgary, AB, T2N 1N4 Canada
+1 403 210 9404
carman@cpsc.ucalgary.ca

INTRODUCTION

The mismatch between the user interfaces for email clients and user needs for handling email has been documented numerous times [1, 8, 14, 4]. This disparity has stimulated proposals for a plethora of client user interface design changes. One recurrent theme is that messages should be viewed as elements of a *conversation* rather than as independent elements [14, 10, 9]. A conversation, also known as a *thread*, is typically defined as the tree of messages that grows with the *reply* operation.

One of plausible benefits of viewing messages as conversations is that the related messages will provide *better local context*, which can help one better understand the meaning of the message. Although this context is preserved to a limited extent by current email programs when they automatically include the text of the original message when replying, this method breaks down when a message receives multiple replies, creating a complex, branching reply tree. Subsequent replies provide additional context but these are not captured by quoting.

TWO MODELS OF CONVERSATION

Implicit in the designs of email clients and related research projects are two models of conversation that appear to be in conflict. On one hand, a conversation is a simple sequence of turns; on the other, a conversation is a branching tree.

Sequential Model of Conversation

A visualization supporting the *sequential model* can answer certain questions about a conversation:

A. *Which of these two messages was sent first?*
B. *Which messages were sent before this one?*
C. *Which messages were sent after this one?*

An interface supports the sequential model to the extent that it can answer these questions at a glance. Note that displaying the message "sent date" on a non-chronological list of messages does not satisfy the "at a glance" requirement as reading and comparing dates is substantial cognitive act.

Typical email clients in a view sorted or grouped by conversation or subject, typical IM clients and Coterie [2] show a chronological list of messages clustered by conversation, thus strongly supporting the sequential model. Typical chat clients, typical email clients in normal operation, the interface proposed by Rohall et al. [10] and the Loom thread view [3] are chronological but messages are not clustered by conversation, undermining the sequential model because messages in the conversation are more likely to be scrolled out of view. Netscan [13] offers an interesting design point, being chronological by day but mixed within a day. Lotus Notes in its "Thread View", ConverSpace [9], Threaded Chat [12] and typical Usenet newsgroup browsers use a schematic tree view where replies to a message are sorted chronologically; such a tree view can answer some of the above questions some of the time but cannot be relied upon to answer all of the questions all of the time. Conversation Map [11] does not represent the chronological sequence of messages.

Tree Model of Conversation

A visualization supporting the *tree model* of conversation can answer these questions about a conversation:

D. *Which message is the root of the conversation tree?*
E. *Which message is this one a reply to?*
F. *Does this message have any replies?*
G. *Which messages are replies to this one?*

An interface supports the tree model to the extent that it can answer these questions at a glance. Lotus Notes in its "Thread View," ConverSpace and Threaded Chat show the messages in a tree view, and thus strongly support the tree model. Netscan, Conversation Map, the work of Rohall et al., the Loom thread view and typical Usenet newsgroup browsers each shows a schematic tree that reflects the selection of a single message to be viewed, but cannot answer questions E or G at a glance; all these interfaces would typically be used with quoting, so question E would be answered by the message content, leaving G unanswered. Typical email clients rely only on quoting, thus they answer only questions D and E and support the tree model weakly. Coterie and typical chat and IM clients do not support the tree model at all.

Models in Conflict?

The degree to which each model is supported can be taken as two orthogonal axes. The interfaces discussed here may then be placed approximately in the resulting space captured graphically in Figure 1. Note that the top-right corner of the space is empty.

Figure 1: Email systems can be placed approximately in the space defined by the degree to which they support the sequential and tree models of communication. Note that the top-right corner is empty – none of these systems fully supports both models simultaneously.

Is it possible for a visualization to fully support the sequential and tree models simultaneously? We set out to design one.

VISUALIZATION DESIGN

In preparation for designing a *mixed-model* conversation visualization that fully supports both the sequential and tree models, we gathered our design requirements. To begin with, it had to answer all seven questions above at a glance. The "at a glance" requirement ruled out dependence on interaction, e.g. selection or mouse-over highlighting (though we knew we might later add those to reinforce the visualization). That same requirement also ruled out a separate message viewing pane: the message content had to be present in the visualization. From examining many conversation trees, we were aware that they tend to be narrow rather than bushy – that is to say that a message is much more likely to get one reply than two or more – so chains of replies-to-replies should be visualized cleanly.

We chose to list the messages in a chronological, vertical list for three reasons. First, a chronological list of messages supports the sequential model trivially. Second, the messages could be reflowed to fit the available width. Third, scrolling (if necessary) would be in one dimension only. We chose to list the messages from old (top) to new (bottom) so it could be read like a script.

But what about supporting the tree model of conversations? The root is always the first chronologically, so Question D is answered, leaving Questions E, F and G open. Specifically we had to design a visualization that showed the reply relationships. We suspected that indenting would be a crucial part of the layout. Knowing the frequency of reply-to-reply chains, we knew that the first reply to a message would have to be at the same indentation level. It wasn't too difficult to surmise from there that the successive children would be indented successively.
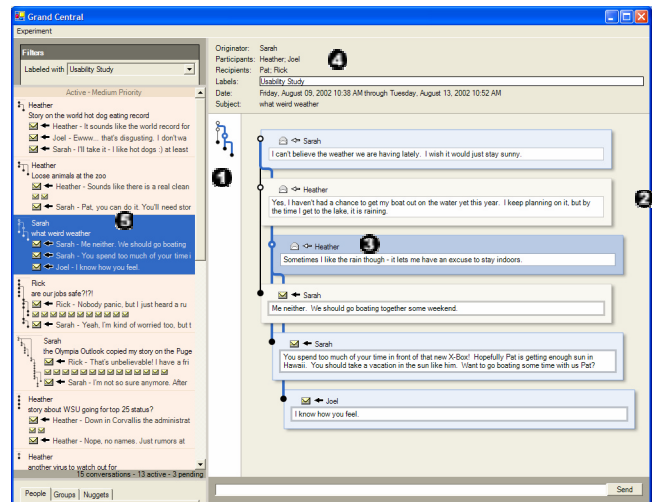
Figure 2: The conversation visualization incorporated into a conversation-based email client. (See text for callouts.)

Indentation alone didn't convey the message-to-reply relationship, so we knew that some mechanism needed to represent the branching of the conversation tree.

The visualization that resulted from our design process is shown in Figure 2, callout (2). The messages are in chronological order, root at the top and newest at the bottom. The root message is at the leftmost indentation level. Its first reply (the second message) is at the same indentation level, connected with a heavy vertical line. The first (and only) reply to the second message (the fourth message) is again at the outer-most indentation level, connected with a vertical line. The second reply to the root message is indented. A heavy horizontal line along the bottom of the root message curves into a heavy vertical line down to its second reply. This line passes beneath the intervening message. Heavy circles provide visual connections between the heavy lines and the message bodies.

We believed that this visualization effectively answered all seven questions at a glance, meaning that it supported both the sequential and tree models. This was verified during a usability study, described later.

Two things had to be done to make the message data presentable in the visualization. First, quoting actually worked against the visualization by introducing redundant content and inflating message length. Our implementation strips quoting when it is judged to not have changed. Second, messages sometimes appear to be younger than their children. This can happen when system clocks are out of sync. Our implementation repairs any out-of-order times.

PUTTING IT IN CONTEXT

The mixed-model conversation visualization is an important part of a user interface to support a conversation-oriented email client user interface (UI). Additional parts of the client UI (see Figure 2) are described in this section.

We designed the visualization to fit within a user-controlled window width, and to scroll vertically as needed. Scrolling

is in direct conflict with the requirement that all relationships be visible "at a glance". There are several established ways of helping users keep context while scrolling [7, 6]. We chose to use an "overview+detail" approach, using the schematic as an overview ((1) in Figure 2) to the detailed conversation visualization (2). The schematic is laid out the same as the detailed visualization, but without the message bodies. A gray band indicates the area of the overview that's currently visible in the detailed view (not shown in Figure 2 because this conversation doesn't need scrolling). It may be that the schematic is useful even when scrolling is not necessary.

(The schematic view of the conversation tree is similar to that shown by Rohall et al. [10]. As mentioned earlier, their visualization doesn't answer Questions E and G at a glance. By interleaving the messages with the schematic, the questions are answered.)

Another way that large conversations are accommodated is by automatically reducing the scale of the schematic and the indent of the detailed conversation visualization.

Because there are numerous menu commands that pertain to a particular message, we needed to introduce the ability to select messages in the conversation. This gave us the opportunity to identify the messages that relate to the selected message. The selected message is highlighted in blue, the selected message's parent and children are highlighted in a lighter blue and the ancestors and descendents are highlighted more lightly still. The lines connecting the message to its children and through its ancestors to the root are made heavier and blue, both in the schematic (1) and the detailed visualization (2). Selection can be then used to help focus on particular branches of a conversation tree.

We normally show a minimal message header (3) – little more than the sender's name and an "unread" flag. The header can be expanded to show all message fields.

An area of the screen (4) is devoted to summary information about the conversation. The first three fields show the name of the person who sent the first message in the conversation ("Originator"), other people who have sent messages in the conversation ("Participants") and people and groups who have received but not sent messages ("Recipients"). Other fields show the labels that have been applied to the conversation, the date range spanned by the conversation and the subject of the first message.

The three panes described above, (1), (2) and (4), provide different views of a particular conversation. The conversation that is being viewed is the selected member of the conversation list (5). The entries in the conversation list show a thumbnail-sized schematic, the name of the conversation originator, the subject of the first message in the conversation, and an indication of each unread message in the conversation.

TESTING THE CONVERSATION VISUALIZATION

We recruited 6 participants for our study, 3 females and 3 males. All participants had intermediate to advanced experience with Microsoft Outlook 2000 or XP, and had some experience working with threaded email conversations, e.g. newsgroups, work discussion lists, etc. Participants were also all considered to be knowledge workers from a variety of occupations, and had normal or corrected eyesight. Only one participant was not familiar with the concept of thread trees prior to the study.

During the study, participants used the conversation visualization within an email client designed to support conversations (Figure 2). The client was populated with a set of email conversations generated specifically for the user study. Participants did not have the ability to reply to or send emails within client.

Participants were first given the opportunity to explore the email client and build an initial conceptual model. This usually lasted between 5 and 10 minutes. Once the participant felt they had enough initial exploration, they performed a series of seven on-screen tasks with the client. Each task required the participant to find a particular conversation or email within a conversation and then answer questions about the found item. Questions included general usability questions as well as specific instances of Questions A-G. Following these tasks, participants completed a post-test questionnaire where they rated how easy they felt it was to answer each of the conversation model questions using the visualization.

Next, participants received a short training session where the conversation visualization was explained, and in most cases, this turned out to merely be validation of the participant's conceptual model. This segment normally lasted only a few minutes. Last, participants were shown two paper screenshots of the conversation visualization, each containing a different email thread. The message contents were replaced with nonsense text and selection highlighting was eliminated, so participants relied only the visualization itself. For each screenshot, participants were asked specific instances of Questions A-G twice, with the exception of D, which was only asked once.

User Study Results

We measured our design's ability to represent the sequential model by three means: observational, subjective and objective. By observing each participant and the answers given during each task, it was evident that by the third task all participants understood that emails were sorted chronologically within the visualization and thus they had a solid grasp on the sequential model. When asked in the post-test questionnaire to rate from 1 (most difficult) to 5 (easiest) how easy it was to answer Questions A, B and C, the median response was 4.5 (mean=4.4, s.d.=0.6), indicating that subjects' subjective perception was that the sequence was easily read. When reading the screenshots, the combined participant accuracy for Questions A, B and C was 90% (65 correct, 7 incorrect). When responding to

questions about the screenshots, all participants responded within one to two seconds, thus supporting the "at a glance" claim for the sequential model.

The same three measures were used to assess the effectiveness of the visualization's representation of the reply tree. We observed that by the seventh task, four of the six participants were able to understand the visual cues used to depict reply relationships between messages. (It should be noted that questions relating to the tree model were not asked until the fourth task.) The fifth participant was able to understand the tree model during the training session. The sixth participant still did not have a solid grasp of the tree model at the completion of the study; this participant is the one who had not been familiar with the concept of thread trees prior to the study. When asked in the post-test questionnaire to rate how easy it was to answer Questions D-G on the same scale as before, the median response was 5.0 (mean=4.2, s.d.=1.0), indicating that subjects' subjective perception was that the reply tree was very easily read. When reading the screenshots, the cumulative accuracy for these questions was 96% (81 correct, 3 incorrect). All participants except the one not previously familiar with thread trees were able to respond to the questions about the screenshots within one to two seconds, thus supporting the "at a glance" claim for the tree model.

Taken together, the three complementary measures of the mixed-model visualization's effectiveness show that participants found it easy to answer questions about *both* conversation models.

CONCLUSIONS AND FUTURE WORK

We have discussed two models of conversation that are embodied to varying extents in a variety of systems that support turn-based, tree-structured conversation. We have proposed a visualization that strongly supports both models, and have presented evidence that supports this assertion. We presented a user interface that puts the visualization in context as a part of an email client.

A robust conversation visualization like the one described here is an important building block of a modern email client. There are at least two ways that we would like to improve the mixed-model visualization. First, it could be more compact. When a conversation consists of a sequence of brief messages, the actual message content is small compared with the surrounding graphics. A more compact visualization would allow more relationships to be visible at a glance.

Second, the visualization is overkill for simple conversations. We know that the majority of conversations are one or two messages long [5]. Our casual observations indicate that simple, non-branching structures are common even among larger conversations. In these cases the "cost" of the visualization doesn't impart any immediate benefit. (On the other hand incurring the "cost" in all cases may

make it easier for the user to understand as more complex conversations are encountered and as a particular conversation transitions from simple to complex.)

Making conversations prominent in the user interface is not a panacea for all the ills of today's email clients. Much work remains to be done to make an email client that truly addresses user needs.

REFERENCES

1. Denning, P. (1982). Electronic Junk. In *CACM* 25(3), ACM Press, 163-165.

2. Donath, J. (2002). A semantic approach to visualizing online conversations. In *CACM* 45(4), ACM Press, 45-49.

3. Donath, J., K. Karahalios and F. Viegas (1999). Visualizing conversations. In *Proc. HICSS-32*, ACM Press and personal communication.

4. Ducheneaut, N. and Bellotti, V. (2001). Email as Habitat. In *Interactions* 8(5), ACM Press, 30-38.

5. Hewett, J. and C. Teplovs (1999). An analysis of growth patterns in computer conferencing threads. In *Proc. CSCL 1999*, Erlbaum, 232-241.

6. Hornbaek, K. and E. Frøkjær (2001). Reading of electronic documents: The usability of linear, fisheye and overview+detail interfaces. In *Proc. CHI 2001*, ACM Press, 293-300.

7. Leung, Y. and M. Apperley (1994). A review and taxonomy of distortion-oriented presentation techniques. In. *TOCHI* 1(2), ACM Press, 126-160.

8. Mackay, W. (1988). Diversity in the Use of Electronic Mail: A Preliminary Inquiry. In *Transactions on Office Information Systems* 6(4), ACM Press, 380-397.

9. Popolov, D., M. Callaghan and P. Luker (2000). Conversation space: Visualising multi-threaded conversation. In *Proc. AVI 2000*, ACM Press, 246-249.

10. Rohall, S., D. Gruen, P. Moody and S. Kellerman (2001), Email visualizations to aid communications. In *Late breaking hot topics – Proc. InfoVis 2001*, IEEE, 12-15.

11. Sack, W. (2000). Conversation map: A content-based Usenet newsgroup browser. In *Proc. IUI 2000*, ACM Press, 233-240.

12. Smith, M., J. Cadiz and B. Burkhalter (2000). Conversation trees and threaded chats. In *Proc. CSCW 2000*, ACM Press, 97-105.

13. Smith, M. and A. Fiore (2001). Visualization components for persistent conversations. In *Proc. CHI 2001*, ACM Press, 136-143.

14. Whittaker, S. and Sidner, C. (1996). Email overload: Exploring personal information management of email. In *Proc. CHI 1996*, ACM Press, 276-283.