PIM-Mail: Consolidating Task and Email Management

Jan-Peter Krämer

RWTH Aachen University

52056 Aachen, Germany

jpk@cs.rwth-aachen.de

Abstract

Traditional email clients are built with a "one-touch" model in mind that assumes an immediate action is performed once an email is read. However, some emails require a follow-up action or users decide to read them later, so they cannot be discharged immediately. We present a prototype to keep track of these email-associated tasks that works as a plug-in inside a traditional email client. Besides providing flexible task management features, such as linking more than one message to a task to follow conversations, our system also supports exchanging tasks for collaborative work.

Keywords

Email, PIM, Personal Information Management, Tasks, Organization, Collaboration, Conversations, Plug-in

ACM Classification Keywords

H.5.2 User Interfaces (D.2.2, H.1.2, I.3.6) - Graphical user interfaces (GUI), H.5.3 Group and Organization Interfaces - Organizational design, Computer-supported cooperative work, H.4.3 Communications Applications - Electronic mail

General Terms

Human Factors, Design

Introduction

Email is the most successful internet application. In 2009, the average corporate user spent roughly a quarter of his/her workday on email-related tasks [1]. Clearly, this time is not solely spent on communication. The data indicates that several other tasks are accomplished with emails as well. Today, email, in addition to being a highly flexible communications tool, is used to send out event invitations, share documents, or delegate tasks. These new uses of email have been identified in several studies [12], [3], [6], [4].

Current email clients lack adequate support to deal with these new aspects of email-based communication. Only event invitations are handled well by all major email clients since the introduction of the versatile and open iCal format [5]. Most other features mentioned above, like keeping track of multiple versions of the same document that is sent back and forth between colleagues, or managing tasks that are received via email, are not widely used, even if supported.

We want to investigate how email tools should be changed to support delegating tasks and sharing documents, and what the effect of these changes is on the user behavior. Therefore we implemented an extension to an existing email client that should increase user efficiency and sense of control. Further, we hope to learn from the users, which features are required to achieve these improvements.

In this paper we first introduce related work in the area of email and personal information management. Secondly, we present an early prototype and describe the findings of a user study with this prototype. Afterwards, we summarize design lessons learned from the first iteration and how they affected the refined version of our plug-in. Lastly, we give an overview about the future iterations we plan to further improve and analyze our prototype and our understanding of the field.

Related Work

The seminal work by Winograd and Flores [13] first analyzed communication between agents as "speech acts" and was highly influential for the work in automated support for email communication management.

In 1996, Whittaker et al. [12] performed an in-depth study to analyze how users work with their email inboxes. They describe a "one-touch" model to deal with emails. In this model users are expected to perform an action immediately after they read a message. Emails containing a task were often kept in the inbox, because users valued the "visual reminding function" of these messages in the inbox. Also, some users did not file open discussions, i.e., messages they expected further responses to. Unfortunately, since for these messages no action was executed, as expected in the one-touch model, they quickly cluttered the users' inboxes.

Bellotti et al. [3] use "thrasks" as new and unique top-level organizational category for emails. A thrask is a threaded, task-centric collection of messages, or even other content. An email client, "Thraskmaster", that uses thrasks for organization is suggested as well. Hence, the email client becomes the primary task management tool, which was greatly appreciated by users. However, the prototype suffered from technical

limitations that kept some users from using it for a prolonged period of time.

Gwizdka et al. [6] visualize emails as points on a 2-dimensional grid. Lines represent emails and time is represented on the x-axis. Future deadlines are marked as dots on the resulting grid. Hence spatial memory can be used to retrieve task information, less time is required for date-related retrieval tasks. However, the proposed design requires setting the expected deadline for each task, thus creating an additional overhead.

Many commercial email clients already have task-management facilities built in. However, they suffer from different drawbacks. In Apple Mail[1], tasks can be attached to specific parts of any email. These are collected in a special folder together with a link to the message. However, tasks are not visible from the inbox, sacrificing the "visual reminding function" mentioned above. Microsoft Outlook[2] offers a multitude of different utilities to handle pending tasks. A study conducted by Microsoft Research [4] found that still by far the most popular method to keep track of tasks is to leave them in the inbox. It remained unclear if the available advanced features are too limited, too difficult to use, or not visible enough in the interface. A more successful approach to task management in Outlook by Bellotti et al. [2] combined a simple task list that can be populated via drag and drop with automatically generated folder structures for more complex tasks.

Prototyping new interfaces to solve problems in the area of personal information management is generally

problematic if these prototypes require users to replace the application they currently use with a new one [8]. Users usually spent a lot of effort to form their current workflow, and are unwilling to adopt another one if this requires spending effort again. Jones et al. [7] suggest providing prototypes as plug-ins for existing applications to circumvent this problem.

First Prototype

We started exploring ways to ease the management of emails that require future action by implementing a very simple plug-in for Apple Mail to support this task. Although developing a prototype bottom-up as a new application allows more radical redesigns, we chose to extend an existing application. On the one hand, prototypes developed this way will less likely suffer from missing features, on the other hand, our users can decide for each message if using our plug-in is advantageous for them, or otherwise hold on to their current strategy for email management and continue leveraging their trained knowledge about the host application. We are confident that users are more willing to use such a prototype for a longer period of time during a user test. Additionally, they will not be tempted to switch back and forth between their old email client and our prototype.

Our plug-in adds a *to-do* drawer (a dynamic sidebar that slides out of a window to either side) to Mail that shows a list of tasks in an outline view. This way, tasks can be always visible, to preserve their visible reminding function, or hidden, depending on user preference.

To add a task, users can drag and drop any message (or note or RSS feed entry), to the to-do drawer. The

task is named after the subject of the email by default, but can be renamed to something more meaningful.

Creating a task from a message is different from filing it, because the first interaction only creates a link. The message is not moved by the plug-in and can be filed to the user's liking. Optionally, our plug-in can be configured to move all messages that belong to a task to a user-specified mail folder. This could be an "Action" or "Tasks" folder that many users wanted to use but failed in doing so due to lacking support [12].

Supporting collaboration can help users organizing their own as well as their co-workers' or friends' tasks more efficiently. If a message subject is prefixed with "Todo:", it is considered to be a task and automatically added to the user's task list. Hence, to dispatch a task to a colleague or friend, it is sufficient to prefix the subject accordingly. This addresses one crucial aspect for success of this kind of application extension: elegant backwards compatibility. If the recipient is not using PIM-Mail, the prefix will still look natural and fit in the context of the conversation. Once a task is marked done, a new message pops up that is addressed to the sender of the task and prepared with a short message.

Initial user feedback
To collect initial user feedback, we conducted a small, informal user study with 4 colleagues that use Apple Mail as their primary email client. After one week, we asked them for their experiences with PIM-Mail. Additionally, we encouraged them to tell us everything they noticed while using our plug-in immediately.

The feedback was very positive. All users appreciated the possibility to seamlessly mark messages as tasks

and get reminded of them every time they use their email client. Also, the message that pops up automatically once a task is marked done was considered useful most often.

All participants requested the ability to link more than one message to a task. This feature allows using the plug-in to follow conversations or to organize more complex tasks that require discussion with others to complete. Other requested features included the ability to reflect the coloring of the original messages and to allow revealing a message by selecting it in the drawer.

Although the feature set of the tested version was fairly limited, one test participant still used the plug-in after several weeks until it stopped working due to an incompatibility with an operating system update (which has since been addressed).

Design Lessons
From our first study and results collected from related work, we were able to distill a few guidelines that we intend to apply to the next iteration of our prototype.

The requests for features providing a better integration of our plug-in with the host application show a strong desire of users to keep their existing strategies for organizing their inboxes while integrating or transitioning to a new tool. We consider this a confirmation of our initial hypothesis regarding the benefits of implementing PIM-Mail as a plug-in for an existing, fully-featured email client.

To understand the value of linking multiple emails to a single task, we considered the classification of emails that cannot be discharged immediately by Whittaker et

The outline view (1) shows tasks and their associated messages.

Messages can be clicked to reveal them in list view (2) or double clicked to open them in a new window.

More messages can be associated with a task by dragging them on the task.

Replies are added to tasks automatically (3).

Tasks and messages are shown with colors set in Mail (4).

Dragging a message in the empty space below the other tasks (5) creates a new task.
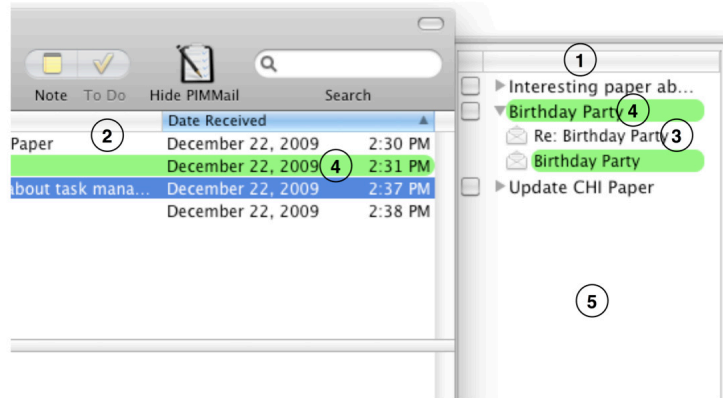
al. [12]. Messages can, among others, be "to dos", meaning they require further action, or "to reads", meaning they are very long and users prefer to read them later. "To reads" can be interpreted as "to dos", too, because the implied follow-up action required is reading the email. Both categories are supported quite well by our first prototype. Linking multiple messages to a task becomes necessary if the task is more complex and requires communication with others. Then the messages belonging to this conversation also fit into another category from [12]: "ongoing correspondence". Hence, by adding the requested support for more complex tasks, we implicitly also support organizing another class of messages.

Attaching multiple emails to a task also implies grouping them. This is one of several kinds of actions that are required for an "email scratch area", that is envisioned in design implications that Siu et al. derived from a study about handling deferred messages [11]. They point out that implementing natural interactions such as reordering and grouping of items is important, because this is what users naturally do to organize tasks on paper. We decided to pursue this idea during the design of a refined prototype.

Refined Prototype

In our refined prototype, each task can be associated with a list of messages. When a message is first dropped into the drawer, the message list for the task, which can be revealed by clicking the small triangle before the task, will contain only the dropped message. If an incoming message is detected to be a follow-up message (i.e. it has the same subject except for "Re:" and "Fwd:" prefixes), it will be attached to the task automatically. Messages can also be dragged onto a task to link the message to the task manually.

We also added the requested features to provide better integration with Apple Mail: Clicking on a message in the list now reveals the message in the folder hierarchy; double clicking the message opens it in a new window. Furthermore, the messages and tasks reflect colors that are chosen for the linked items in Mail. The plug-in was also updated to again work with the most current operating system version.

Future Work

The design of our tool as a plug-in now allows deploying it to a large user group. Especially for Apple Mail, there is an active online community around plug-ins. Thus, we hope to gather profound quantitative data by deploying the plug-in as a free download, in turn asking users to participate in our study. We plan to collect anonymous usage statistics accompanied by a weekly questionnaire for the duration of the study. Very basic usage statistics may be sufficient [9]. In the study, we want to explore how user strategies to deal with email flow change when using PIM-Mail in comparison to former studies (e.g. [12], [11]).

In a future version of the plug-in, we also plan to add the tasks that are dispatched to others to a separate section in the drawer. If the task is done, the response that is generated by PIM-Mail will be automatically registered. PIM-Mail will then mark the task as completed on the sender side. In this configuration the plug-in provides a very efficient way to pass tasks to a lot of colleagues, while still seamlessly blending into the current workflow of users not using PIM-Mail.

Additional future work will include developing our prototype further to support more tasks. While we do show ongoing conversations that are related to a task, currently our plug-in is not primarily designed to visualize conversations. Support for conversations could be improved by allowing groups of messages that are no tasks.

Another interesting problem to look at in future work is file management in emails. When collaboratively creating documents, different versions of these are frequently sent to other group members via email [10]. As a result, the email client becomes a tool to organize and browse different revisions of a document, similar to the feature set versioning systems provide. We will explore possibilities to support this task by grouping messages containing different versions of the same file.

Acknowledgements

References
[1]   The Radicati Group, Email Statistics Report 2009-2013 Executive Summary, The Radicati Group (2009).

[2]   Bellotti, V., Thornton, J., Chin, A., Schiano, D., Good, N. TV-ACTA: Embedding an Activity-Centered Interface for Task Management in Email, Proc CEAS '07

[3]   Bellotti, V., Ducheneaut, N., Howard, M., Smith, I. Taking email to task: the design and evaluation of a task management centered email tool, Proc CHI '03.

[4]   Cadiz, JJ, Dabbish, L., Gupta, A., Venolia, G.D. Supporting Email Workflow, Tech Report MSR-TR-2001-88, Microsoft Research (2001)

[5]   Desruisseaux, B. Internet Calendaring and Scheduling Core Object Specification (iCalendar), IETF RFC 5545 (2009).

[6]   Gwizdka, J., Chignell, M. Individual differences and task-based user interface evaluation: A case study of pending tasks in email, Interacting with computers, Vol. 16 (2004).

[7]   Jones, W., Munat, C.F., Bruce, H., Foxley, A. The universal labeler: Plan the project and let your information follow, Proc ASIST, Vol. 42 (2005)

[8]   Karger, D.R., Jones, W. Data unification in personal information management, Commun. ACM, Vol. 49 (2006).

[9]   Renaud, K., Gray, P. Making sense of low-level usage data to understand user activities, Proc. SAICSIT '04.

[10] Samiei, M., Dill, J., Kirkpatrick, A. EzMail: using information visualization techniques to help manage email, Proc. Information Visualisation '04.

[11] Siu, N., Iverson, L., Tang, A. Going with the flow: email awareness and task management, Proc. CSCW '06.

[12] Whittaker, S., Sidner, C. Email overload: exploring personal information management of email, Proc. CHI '96.

[13] Winograd, T., Flores, F. Understanding Computers and Cognition. Addison-Wesley, 1987.