

Mapmail: Restructuring an Email Client for Use in Distributed Teams

Les Nelson

PARC
3333 Coyote Hill Rd.
Palo Alto, CA 94304 USA
lesnelson@acm.org

Elizabeth F. Churchill

Yahoo! Research
2821 Mission College Blvd.
Santa Clara, CA 95054, USA
churchill@acm.org

Figure 1. The Mapmail is an example of an ‘application surrogate’, in which selected Web mail messages are displayed on a geographical map, using locations given from a contact manager. Yahoo! Mail and the Plaxo contact manager are shown here.



Abstract

In this paper we examine a solution for coping with information sharing across location, time zone, and organizational boundaries. Email, a principal tool for such collaboration, relies on textual and/or numeric sorting, filtering and presentation by subject, data, sender addresses and/or time. Typically, spatial properties of presentation are limited to filtering incoming messages into folders. We have been designing for a different form of social expressiveness – mapping email by geographic region. We describe an example of how an email client was extended, preserving its original familiarity and functions, but augmented with new features expressed in a spatial map arrangement. We describe how this approach exemplifies a general technique we call “application surrogates”. We discuss application surrogates in terms of emerging “mashup” approaches to application development.

Keywords

Information sharing, communication, collaboration, distributed teams, email, repurposing, application surrogacy

ACM Classification Keywords

H.5.2 User Interfaces.

Copyright is held by the author/owner(s).

CHI 2007, April 28–May 3, 2007, San Jose, California, USA

ACM 978-1-59593-642-4/07/0004.

Introduction

There is a tension between the comfortable familiarity of an application and the desire to move beyond that form of presentation as needs change. It is not unusual to hear complaints at the water cooler about the different merits of applications and why can we not combine the 'best bits' among them. There is difficulty in influencing an established application (even inside the providing organization, [7] in the face of allocation choices (which users, brand concerns, and so on to support now).

The email 'inbox' is a good example of this tension. The sorting and filtering by subject, data, and sender all support a consistent view that is largely textual and sometimes numeric (e.g., dates). Such a representation, while familiar, is often not the easiest way to glean an overall awareness or overview of communication through this channel. In this report, we use our recent work in supporting globally distributed teams to illustrate our approach to the application limitations noted above. Specifically, we wish to explore the advantages of viewing email flows spatially using a map representation. Following a brief description of our Mapmail application, we discuss the broader issue of "application surrogacy", our general technical approach to repurposing interactions.

Email Re-visualization

We are not alone in the desire to offer novel presentations of email. Recent work in visualization considers alternative representations for reading email content. Themail, for example [6], presents email content through a series of columns of keywords arranged along a timeline. Keywords are

shown in different colors and sizes depending on their frequency and distinctiveness as determined by text analysis of the message content. This abstracted representation of the messages gives a quick overview of the key topics being discussed over time, thereby aiding in reading and browsing.

However, the Themail visualization is a generated report of the email content that may be browsed, but is not integrated with the application view of an email client. In our work, we wish to overcome this dissociation, and also provide visual support for distributed groups working across time zones.

Mapmail: A Spatial Restructuring of Email

We have here repurposed Yahoo Mail so that an alternative mode of presentation is made available. However, we also require that the repurposed view not only shows the relevant information for browsing, but maintains a consistent and up-to-date interaction across all mail information shown. We refer to such active alternative interactive representations as *application surrogates*.

An *application surrogate* provides an alternative and parallel interaction with data accessed within an existing application, while maintaining selection and state consistency between views. Selection consistency requires that any selection in one view is reflected back in the other. State consistency ensures that changes due to events (e.g., new message arrival, message deleted or filed) are replicated in all views.

One such surrogate is the Mapmail shown in Figure 1. In this view, email messages that arrive in the



Figure 2. The sorting and filtering of email inbox content by subject, data, sender addresses all support a consistent view that is largely textual. Activity awareness is limited to a few textual cues such as boldface

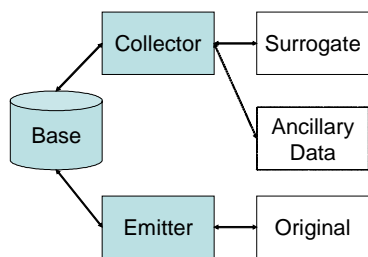


Figure 3. The Emitter-Base-Collector arrangement creates a loosely coupled independent surrogate process that is kept in synchrony with the original interface target.

inbox are displayed over a geographical map, using locations given from a contact manager. Mail can be arranged spatially with respect to locations the receiver either associates with the sender (e.g., home town, main work or satellite office, event location), or that the senders indicate themselves (as in the case of a shared contact manager such as Plaxo). Figure 1 shows mail messages ‘arriving from’ Hawaii, Connecticut, and Fargo, North Dakota, USA. This representation of incoming messages provides an expressive and quickly assessed context to the incoming communications, reducing cognitive overload by offering a scannable view of a broad email landscape.

In the case of the Mapmail shown here, Yahoo! Mail (mail.yahoo.com) is accessed (e.g., POP3 mail retrieval in the current instance, soon to be replaced by the upcoming Yahoo Mail Application Programming Interface [API]). The Plaxo contact manager is accessed through its Synchronization API (www.plaxo.com/api/sync). The final presentation is made over a mapping interface (e.g., Yahoo Map Services, developer.yahoo.com/maps). This implementation is only one of many possible ways that illustrate the more general capabilities of email and contact management, with graphical layout.

A Scenario of Use

In the following we have adapted an email task scenario from practice observed in our fieldwork. We have replaced the point-and-click folder search we observed in practice with use of our Mapmail.

Jennifer, a VP for a global consultancy is responsible for overseeing teams in the US, Germany, Russia, China and

Singapore. She likes to keep track of emails from the different divisions and be aware of the activity level in each region (Figure 2). Now and again she will respond to an email. To manage her email, she has folders for each region where mail in that area is filtered. To review activity levels in each area, she opens each folder in turn, scanning From and Subject lines to get a sense of the level of activity and latest “buzz”. This requires scanning upwards of 200 emails to get a sense of the activities and issues in these areas.

With Mapmail, she opens the map visualization, and easily gets an overview of activity that is occurring in each region. She can zoom into specific regions, or zoom out to get a global perspective. She can overlay the representation with local zones. She can filter what is represented on the map to show email sent directly to her, or by mails that she is cc’ed on, or to show all email. By mousing over the Mapmail representations, she can see a list of the senders of the emails, the title and time sent. Emails can be clustered by subject line, and color coding can show which emails have been responded to. Clicking on the email icon opens the email application where she can see more detail. This gives her a quick overview of global activity.

This scenario, drawn from fieldwork, was the motivating example for our initial design. Many other email re-presentation scenarios are being explored.

Implementing the Mapmail

To provide the interactivity and synchronization across email representations, we use the following architectural components, we call the Emitter-Base-Collector model (Figure 3). Each component may be run concurrently as separate agents/processes or grouped together as functions in a single service.

The Collector component gathers information and events (e.g., selections and state changes) from the original application (e.g., Yahoo Mail). Similarly, an Emitter component does the same job for the surrogate representation. A Base data repository (e.g., MySQL database for content caching and shared information pool for quickly communicating selection events) is queried by all connected processes (surrogates and original) for synchronization. Other supporting sources of information are managed by the collector to assist in the presentation (e.g., Plaxo contact manager access for geographic locations, www.plaxo.com).

The Map presentation currently supports the following capabilities, though many other arrangements are possible. We provide state consistency in the following ways. We display incoming messages at the location (street address or latitude-longitude if available) from the contact record. Senders may dynamically update their location through the contact manager. When messages are filed or deleted they are removed from the map. Color-coding is used for new arrivals and message importance. Multiple messages from recipients are stacked on top of each other. Successive clicks cycles through the messages at that location. All messages from sources not in the contact list are placed in a default location. In the case of Figure 1, we arbitrarily chose to stack all such contacts starting in Fargo, North Dakota, USA, and put successive message north of there. This illustrates opportunistic advantages such as having a central location not in the contact list.

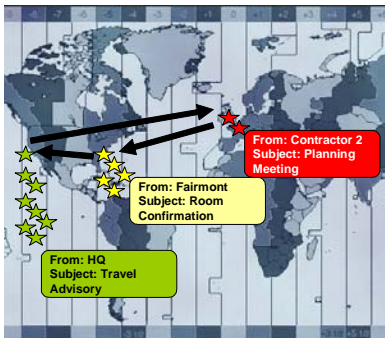


Figure 5. Map view for showing messages relating to a pre-arranged business trip.

Selection consistency involves supporting the following action: clicking on message icon shows detail and permits the original mail client to be opened to that message. We intend to support more such capabilities when a full Yahoo Mail Application Programming Interface becomes available, including selection of icons on the map causes selections on the Yahoo Mail folder items, and vice versa.

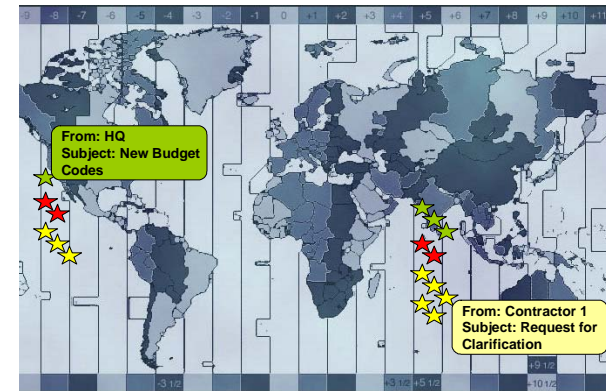


Figure 4. Map view for showing open communications during a shift handover.

Mapmail: Future Work

Many other arrangements for the map are possible in terms of layout (e.g., alignment patterns for multiple messages), icon (e.g., representations with photographs of senders, different maps), and detail shown on hovering or clicking a message indicator. Further, filters for different folders, message fields (e.g., subject keys), preferred senders, organizational role, and temporal filters are being considered. We are investigating spatial arrangements suggestive of organizational concerns (e.g., shift handover across time zones, Figure 4),

and related to business activities (e.g., communications relating to a business travel, Figure 5). Currently, we have iterated on the design for collaboration for personal use within our local team needs. Our next stage will be evaluating its effectiveness with users in different settings, addressing matters of scale and interaction design. In other work, people's contacts have been visually clustered for easy browsing [3]. Mapmail uses a geosocial location where people are identified to cluster their messages. With advances in geo-tagging, we anticipate being able to chart their actual location as well.

Discussion

Our Mapmail application is one illustration of a general issue we have observed in our fieldwork into the use of off-the-shelf applications in organizations. A recurring issue with general-purpose capabilities such as an email client is that needs of an individual or small group gets lost in the demands of a larger application market. Mapmail addresses the visualization needs of distributed communicators with large email loads – providing them with a simple, unified overview of activity.

In our studies of personal and organizational information sharing, we identify several use situations that aid us in our understanding of the needs for application restructuring.

Personal View Preferences

Highly personal ways of organizing and accessing information lead to unique arrangements of applications. Applications try to accommodate these

with pre-planned customizations that are often the result of many iterations of feedback from users. How responsive an application may be to such customization is influenced by the flexibility of application platform and how common the 'unique' requirements are among the user base. Platform-based applications such as Microsoft Outlook (built with an open API) and open source solutions (built on an open implementation), may afford considerable adjustment if one is willing to 're-program' the application.

Cross-Organizational, Close Collaboration

We have investigated a number of client-server alternatives supporting collaboration across organizational boundaries, including extending existing document management products (e.g., Microsoft Sharepoint, www.microsoft.com), research collaboration software (e.g., PARC CoLabSpace [1]), and open source collaboration software (e.g., PmWiki, www.pmwiki.org). There are problems with such a tightly coupled approach in current distributed settings, largely related to the system maintainability and feasibility concerns, including distributing expertise and maintenance demands across organizations and system versions and negotiating sensitive security concerns between organizations.

Consequently, corporations often rely on email communication, and are faced with many practical concerns. Targeting and distribution of information to subsets of project members is imprecise and causes extra reading, filtering, and maintenance. Much effort is spent in keeping current with ever-growing email threads.

In keeping with the familiarity of email in the work place (email as “habitat” for workers [2]) we are here seeking to restructure the email presentation to address such collaboration concerns.

Conclusion

Mapmail is an illustration of application surrogacy. In this way we create flexible “application mashups” that extend the presentation and interaction properties of existing work tools to provide better task support. We see this as an important direction for ‘post-Web 2.0’ interaction, where people do what they already do, using the tools familiar to them, but layering them with more useful interaction and visualization properties that address evolving needs. Further, personal and organizational infrastructures do not need to change to accommodate extended uses. Changes occur at an application’s interaction layer, within visible parts of the underlying support (e.g., proxy serving and routing), and through loosely couple integration with other applications and data sources.

References

- [1] Bobrow, D., Cheslow, B., Churchill, E., Nelson, L., Whalen, J., Conversation Support in a Collaboration Space for a Distributed Workgroup, Workshop Paper, Workshop on Beyond Threaded Conversation, CHI 2005, Portland, OR, April 2, 2005.
- [2] Ducheneaut, N., Bellotti, V., E-mail as Habitat: An Exploration of Embedded Personal Information Management, interactions, Sept. 2001.
- [3] Nardi, B.A., Whittaker, S., Isaacs, E., Creech, M., Johnson, J., Hainsworth, J., Integrating communication and information through ContactMap, CACM, vol. 45, no. 4, pp. 89-95, 2002.

Mapmail provides an opportunity for elaborating a new interface repurposing technique [4]. We take a “mashup”-like view [5] of a standard desktop information management tool: extract data from the desktop source, represent it in new ways, and integrate the interactions with the new representation to the original source. Unlike the Web 2.0 Mashup, which recombines and re-presents different internet data sources, we extend the range of data acquired to the platform and application level and make information presentations active with information flows multi-directional between sources. By running a parallel and cooperating process, with state and selection consistency enforced through APIs, we can provide customized interactions over existing systems, thereby extending their usefulness as interaction needs change.

- [4] Nelson, L. D.; Churchill, E. F., Approaches for interaction repurposing, Proc. IEEE Conference on Information Reuse and Integration, 2006.
- [5] Nelson, L.D., Churchill, E.F. Mash me up, Mash me down. CSCW 2006 Workshop on Mashups, Banff, 2006.
- [6] Viégas, F.B., Golder, S., Donath, J., Visualizing Email Content: Portraying Relationships from Conversational Histories, ACM Proc. CHI 2006.
- [7] Wattenberg, M., Rohall, S.L., Gruen, D., Kerr, B., Email research: Targeting the Enterprise, Human-Computer Interactions, v. 20., no. 1-2, 2005.