This Time It's Personal: from PIM to the Perfect Digital Assistant

m.c. schraefel[1], Paul André[1] and Max Van Kleek[2]

[1]Electronics and Computer Science
University of Southampton
SO17 1BJ, United Kingdom
{mc, pa2}@ecs.soton.ac.uk

[2]CSAIL, MIT
32 Vassar St.
Cambridge, MA, 02139, USA
emax@csail.mit.edu

INTRODUCTION

Interacting with digital PIM tools like calendars, to-do lists, address books, bookmarks and so on, is a highly manual, often repetitive and frequently tedious process. Despite increases in memory and processor power over the past two decades of personal computing, not much has changed in the way we engage with such applications. We must still manually decompose frequently performed tasks into multiple smaller, data specific processes if we want to be able to recall or reuse the information in some meaningful way. "Meeting with Yves at 5 in Stata about blah" breaks down into rigid, fixed semantics in separate applications: data to be recorded in calendar fields, address book fields and, as for the blah, something that does not necessarily exist as a PIM application data structure. We argue that a reason Personal Information Management tools may be so manual, and so effectively fragmented, is that they are not personal enough. If our information systems were more personal, that is, if they *knew* in a manner similar to the way a personal assistant would know us and support us, then our tools would be more helpful: an assistive PIM tool would gather together the necessary material in support of our meeting with Yves. We, therefore, have been investigating the possible paths towards PIM tools as tools that work for us, rather than tools that seemingly make us work for them. To that end, in the following sections we consider how we may develop a framework for PIM tools as "perfect digital assistants" (PDA). Our impetus has been to explore how, by considering the affordances of a Real World personal assistant, we can conceptualize a design framework, and from there a development program for a digital simulacrum of such an assistant that is not for some far off future, but for the much nearer term.

SCENARIO FOR A PDA

In our approach to imagining a perfect - or perfectible - digital assistant, we have currently constrained our imaginings to assistants that will handle repetitive tasks that could be carried out manually if the person had either the necessary time or inclination to spend as much time as necessary to support such a task. Another factor in our PDA PIM is that it must demonstrate an ability to blend relevant personal and public information. To that end, we present the following scenario of Ximo's frantic morning. Ximo is a researcher at a university lab.

Ximo is running late this morning. His table has anticipated his arrival, and pre-assembled a newspaper with a combination of the most important stories out of the 1,520 news and events feeds to which he regularly subscribes. On the front page, the leftmost column ("Your day"), reminds him that the majority of his day is going to be consumed by a workshop held at his lab, and that he has to give a talk at 11:30. As he sits down and starts to read, a breaking news event is received from London Transport relevant to his travel plans: an incident is causing delays along his usual route to work. It recommends an alternate route, and anticipates that his arrival will be delayed by around 20 minutes. "Oh great", he says and flips the page to the music and events section. The top story is a music show that will be happening that evening, two tube stops from his last meeting venue. Not recognising the band name, Ximo touches the "more information" button. The newspaper quickly unfolds a side bar revealing the reason it was recommended to him -- Zara, a friend with whom he shares similar music tastes, has been listening to this band a lot recently. It also shows Zara's public calendar is free that evening. After listening to a sampler of the band's music, Ximo drags the event to Zara's calendar, and attaches a brief message -- "The Sways are playing, wanna see them tonight? I'm gonna be in that part of town anyway." Later, while on the train to the City for his first meeting, thirty minutes before arriving at the station Ximo's laptop brings up the relevant context to review for it -- e-mails, documents, related to-do items, a note-taking space, and a Scrabble application; (Ximo has logged considerable time with Scrabble during these review meetings, so the laptop has simply opened the application for him). Ximo quickly runs through the set of documents, moving two to the top of the stack, and flicking several others to the bottom of the stack. He also removes e-mails from a particular colleague who has ended up having little to do with the project. He also pulls into the set of docs a cached web site that he found yesterday and thinks relevant, but that was not in the current list of related pages. Ximo quickly reviews the summary of material, and dashes for a cab to his venue.

While the similarity of our scenario to earlier conceptualizations of idealised data interaction, such as Knowledge Navigator [3], VISION [2], and Connections [5] may make our vision seem unfeasible, we believe (realistic avatars and natural dialog interaction aside[1]) that a number of closely related fields have already demonstrated systems that exhibit many of the capabilities that such a system would need. In particular, we believe that work in user, task [23], and context modeling [4], unified data models [1], mixed initiative interaction [15], and end-user programming/programming-by-demonstration [12] have shown how systems can learn from user activities and interactions. Additionally, the notion of activity-based computing [19, 17] have helped to start to bring data management tasks into human-terms, while work in experience capture and ubiquitous computing [14,20] have demonstrated techniques for sensing human activities and the environment in ways that are both tractable and useful today. Finally, the SRI CALO project [9] has recently provided a number of demonstrations of rich, model and learning-driven adaptive personal information assistants, and, most importantly have helped to identify key technical challenges towards making such systems "real".

STRATEGY

In beginning to design PDAs for such tasks, our approach is informed by two key concepts. First, we agree as Jones [16] argues, PIM should be about “outgoing” information and how to use this to shape the user’s world. Teevan’s work in personalizing search [24] is an example of taking a rich user model of existing information (previous queries, visited Web pages, documents read/created) into account in order to aid the user in a task. For instance, a user generates a significant amount of data on the Web and on the desktop, and much is (or could be) available in the form of RSS feeds. A user’s music tastes from last.fm; friends and interests from social networking sites; wi-fi connections (indicating location) in specific meetings whilst working on certain documents; and calendar information can all be informative of personal and social practice and thus useful for our PDA.

Second, we consider Dix's Deconstruction / Reconstruction [13]: the approach is to analyse a physical artefact for its functions and affective properties in order to consider how best to translate those properties to the digital realm. In the Real World, the kinds of repetitive, context-specific activities described above may be provided to some by personal assistants, where it is the assistant's job to carry out such activities that will provide support for a person in a given area of that person's life. There are many types of personal assistants: personal trainers, executive assistants, butlers and so on. If we analyze how the human personal assistant works, we may, following Dix's functional and affective approach, see that there are four key features:

1. *upfront explicit domain expertise* (a personal trainer knows about fitness plans and possibly nutrition; is constantly updating this knowledge) ;

2. *upfront explicit client model* (a PA knows when the client comes into work; how they like their coffee; who is at the top of their contact list: this information can be shared explicitly) ;

3. *implicit refinement of the client model based on iteration of practice over time* (the client doesn't get into work before 9.30am, and even later every Thursday - this exception has simply been observed rather than explicitly stated);

4. *explicit negotiation/refinement about new tasks as they are introduced* (the client has said that for a particular project, they only want to see emails from the main stakeholders; after just this info was provided, the client revised the requirement to ask explicitly for any documents that they and the stakeholders had touched in the past two weeks before a meeting).

By using our four-part human personal assistant model to inform our design practice, we can see opportunities for a staged development process that involves progressive increments rather than having to have than a perfect PDA from the outset. Our premise is that a human PA blends explicit domain knowledge with client preferences to deliver support, and such support is most often a blend of personal and public information. We explore this blend in related work (see below).

The above scenario demonstrates providing personal information management in a way that is distinct from the classic PIM tool application model in which the human is the actant creating passive data and feeding it into specific data silos via specific applications. In the above PDA paradigm, we therefore also implicitly suggest data not be locked to any particular application, but persistently available to all. Apple tried such a concept with the Newton's "Data Soup"; for our PDA that is highly integrated with the Web, dependent on the personal/public blend, we propose a "Data Sea" reflective of the scale and diversity of elements to be found in this space that our PDA may access. This approach of a data sea informs our Jourknow work, too [7,8].

RELATED FIELDS

While our simple show-finder provides a service so that manual search can be replaced by automated assistance, in our vision of a PDA, we want to enable richer kinds of interactions with personal and public data than this: we wish to emulate more of the implicit learning and explicit refinement/negotiation that a human PA develops to provide increasingly effective support. We see that such

[1] While avatars and dialogue-based interaction have also enjoyed significant successes [10,21] we believe that this problem is rather orthogonal to PIM and focus on the remaining challenges.

refinement will require engagement with multiple areas of research:

- user and task modelling -- activity-based computing, semantic desktop, automatic task modelling, learning preferences;
- end-user programming / programming by demonstration -- letting people 'teach' their PIM systems how to perform new tasks and proactive assistance and recommendation;
- information retrieval and visualization -- to combat information overload, provide 'visual cognition', and new ubiquitous UIs;
- interaction design for engaging with PDAs;
- policy management for privacy, access and sharing of information.

Indeed, with respect to this last point, we would go so far as to suggest in an ecosystem of increasingly available public and social data, software engineers will have to take into account privacy/policy needs as a fundamental part of core engineering practice.

CURRENT WORK

Since the challenges associated with realizing the earlier scenario are varied and extremely daunting, we have been working out a long-term, cross-institution open collaborative research agenda (as part of WSRI[2]) that both aims to distribute the investigation by isolating questions and components among researchers, and that establishes a timeline for incremental progress that produces integrated "actually functional" prototypes at periodic milestones. Developing and deploying functional prototypes remains a key goal for two reasons; first, we believe that giving functioning prototypes to people will be the easiest way to get feedback necessary to answer many of the yet unposed, unanswered design questions in this space; and second, to help us stay grounded to avoid the "pure demoware/concept prototype" syndrome.

Our first few parallel efforts have begun to build the following components focusing on several small but essential functionality that we believe is of primary need and most lacking in today's toolkits and software:

Jourknow - *liberating PIM from "desktop organizer"* - Jourknow is personal notepad application that returns to text to allow users to flexibly mix both structured data types and unstructured text, while simultaneously providing the affordances to find, organize, visualize and use information that more structured PIM tools provide. Goal: to let users capture more information in their PIM by reducing the effort required to do so, so they have a greater chance of finding it later. (See [7,8]).

[2] http://webscience.org/

PLUM - *relating PIM to user activities and context* - A key limitation of current systems is that they oblivious to the highly contextual nature of PIM -- how information items describe, pertain to, involve people, places, times, and things. PLUM attempts to re-contextualize personal info in two ways: by 1) creating activity logs of desktop activities and their users' environmental contexts (i.e., locations) and 2) mining these logs for people, places, and contexts in which information activities occur. (See [25,26]).

AtomSmasher - *activity-driven lightweight end-user automation* - As a first step towards PIM automation, we have a project that demonstrates how multiple sources of information about a person's activities (coming in as RSS feeds) can be combined to enable end-users to easily script reactive behaviours based on public (global-) and private (user-specific) temporal activity . Intended applications are for supporting context-sensitive reminding, alerting and proactive notification, and social awareness information delivery. (In submission to WWW2007).

Our prototypes employ flexible, relational graph-based data representations in RDF [18]. RDF facilitates representation of many types of data and relationships, integration of heterogeneous types of data into a single "Data Sea", and the use of standard tools for exploring and visualizing the data. It also facilitates integration and data schema evolution.

CONCLUSION

Current common PIM tools are largely passive data silos. In this paper we have asked what would be necessary to begin to evolve PIM tools from passive data collectors to more active information assistants - tools that could take advantage of contextual information to *seem* proactive about supporting a person's activities. We have proposed an approach that is grounded around an analysis of the practice of a human personal assistant, and have asked how we can implement these functions. We are starting small by considering how we can take advantage of public generic data news feeds, the increasing wealth of available personal and social data now regularly published on social networking sites, to blend that with personal data that can be mined locally by systems like PLUM to bring about the kind of support assistants in our scenario above.

Our key motivation in looking at the PDA as an approach for PIM is that it has the potential to improve quality of life. Our founding question is what would be useful/beneficial for a perfect digital assistant to do? Our initial scenario focuses on managing lightweight, repetitive, refinable tasks that add value by providing people with information they have identified as desirable, but do not want to spend the time to find themselves. By starting here, we can begin to make progress in automating PIM in a way that will improve the quality of life we can achieve by not having to manage our information, but in it helping to manage us. As we increase the robustness of, and interaction effectiveness with such assistants, and extend to pervasive, ubiquitous

interaction coupled with sensor data, we start to get at the question: when we no longer need to manually micro-manage our own information, what kind of time for other activities (including rest) becomes possible?

COLLABORATORS

Our previous and ongoing work in this area has been developed in conjunction with Michael Bernstein and David Karger at MIT, Daniel A. Smith and Max Wilson at the University of Southampton, and Ora Lassila, Mark Adler, and Deepali Khushraj at Nokia Research Center Cambridge.

REFERENCES

1. Adar, E., Kargar, D., and Stein, L. A. Haystack: per-user information environments. CIKM 1999.
2. Andersen Consulting, VISION: "http://paleo-future.blogspot.com/2007/08/vision-clip-1-1993.html"
3. Apple Knowledge Navigator, 1987.
4. Ashbrook, D. and Starner, T.E. Learning Significant Locations and Predicting User Movement with GPS. GVU Technical Report;GIT-GVU-02-27.
5. AT&T Labs, Connections: "http://paleo-future.blogspot.com/2007/04/connections-at-vision-of-future-1993.html"
6. Bardram, J., Bunde-Pedersen, J., and Soegaard, M. Support for activity-based computing in a personal computing operating system.CHI 2006.
7. Bernstein, M., Van Kleek, M., Karger, D.R., schraefel, mc. Information Scraps: How and Why Information Eludes our Personal Information Management Tools; in submission to ACM Transactions on Information Systems. 2007.
8. Bernstein, M., Van Kleek, M., schraefel, m.c. and Karger, D. Management of Personal Information Scraps. CHI 2007.
9. CALO Project: http://caloproject.sri.com/
10. Cassell, J., Bickmore, T., Vilhjálmsson, H., and Yan, H. More than just a pretty face: affordances of embodiment. IUI 2000.
11. Cutrell, E., Robbins, D.C., Dumais, S.T. & Sarin, R. Fast, flexible filtering with Phlat - Personal search and organization made easy. CHI 2006.
12. Cypher, A. Watch What I Do: Programming by Demonstration. MIT Press, Cambridge, MA. 1993.
13. Dix, A. Deconstructing Experience - pulling crackers apart In Funology: From Usability to Enjoyment. 2003.
14. Gemmell, J., Bell, G., Lueder, R., Drucker, S., and Wong, C. MyLifeBits: Fulfilling the Memex Vision. ACM Multimedia 2002.
15. Horvitz, E. Principles of mixed-initiative user interfaces. CHI 1999.
16. Jones, W. Personal Information for a World as We Want It to Be. PIM Workshop, SIGIR 2006.
17. Lamming, M.G. and Newman, W.M. Activity-based Information Retrieval: Technology in Support of Personal Memory. IFIP Congress, Vol. 3. 1992.
18. Lassila, O., and Swick, R. Resource Description Framework (RDF) model and syntax specification. W3C Working Draft.
19. MacIntyre, B., Mynatt, E. D., Voida, S., Hansen, K. M., Tullio, J., and Corso, G. M. Support for multitasking and background awareness using interactive peripheral displays. UIST 2001.
20. Newman, W. M., Eldridge, M. A., and Lamming, M. G. PEPYS: generating autobiographies by automatic tracking. ECSCW 1991.
21. Sidner, C. L., Lee, C., Morency, L., and Forlines, C. The effect of head-nod recognition in human-robot conversation. HRI 2006.
22. Stephenson, N. Snow Crash. 1992.
23. Stumpf, S., et al. The TaskTracer system. AAAI 2005.
24. Teevan, J. Dumais, S.T., and Horvitz, E. Personalizing Search via Automated Analysis of Interests and Activities, SIGIR 2005.
25. Van Kleek, M. and Shrobe, H. A Practical Activity Capture Framework for Personal Lifetime User Modeling (PLUM). 2007
26. Van Kleek, M., Berstein, M. Karger, D., schraefel, m.c. Gui Phooey: The Case for Text Input. UIST 2007.