A decentralized architecture for consolidating personal information ecosystems: The WebBox

Max Van Kleek, Daniel A. Smith, Nigel R. Shadbolt, mc schraefel
Electronics and Computer Science
University of Southampton
Southampton, UK
{emax, ds, nrs, mc}@ecs.soton.ac.uk

ABSTRACT

Most existing PIM tools either suffer from having data isolated by being siloed in an application or only interact with specific tools offered through specific services. The consequences are that what people can do with their own data becomes constrained by what services application developers afford. We propose a web-standards based architecture called WebBox to support easy maintenance and repurposing of one's own data for private, social or public publishing, collaboration and reuse.

Categories and Subject Descriptors

H.4 [**Information Systems Applications**]: Miscellaneous

Keywords

privacy, linked data, semantic web, personal data vault

1. INTRODUCTION

Many people today find themselves relying on an ever-expanding confederation of apps and web-based "cloud services" to manage their personal information, including e-mail apps, digital and online calendaring tools, project and task management tools, CRMs, and bookmarking and reference managers to name just a few [8]. Similarly, to share information with others, people manually transfer information from such PIM tools to one or more sharing platforms – such as microblogging services, Facebook-like social networking services, or groupware applications, depending how and with whom they wish to share it.

Despite the new capabilities these tools offer, studies of people's personal information ecosystems reveal that the use of this multitude of disparate apps and services may be causing greater frustration than help [5, 9]. In particular, the information fragmentation and disorganization resulting from the use of such services is contrary to observations revealing that simplicity and ease of use and access correlate with tool adoption, preference, and frequency of use (e.g., [12]). Such studies recommend eliminating extra steps that get in the way of the user's primary PIM paths. In fragmented personal information environments, such extra steps are pervasive; in particular, since the data models of today's apps and services are largely isolated, coordinating related information across applications and services (for instance, gathering URLs, posts or documents pertaining to a particular topic or project) requires manual, often time-consuming effort.

The primary reason that people end up using multiple tools in the first place is that most of these tools are narrowly focused, providing specific support for certain data types and not others. For example, online calendaring services exclusively afford the creation and management of information items that have a date and time of occurrence, but do not support other data types that may be related — addresses/contact information for participants, for example, or to-do entries related to a particular calendar event. Meanwhile, studies have revealed that the information forms in people's PIM environments are numerous and varied, the frequencies of which often follow a heavy-tailed distribution [7]. Since this means many types of information items that are each found rarely, but cumulatively constitute a significant fraction of the total items in a person's collection, any specific tool will inherently be unable to provide adequate coverage for all of a person's information items. Moreover, many items found in the wild exhibit characteristics of multiple information forms simultaneously; these hybrids/information "frankenforms" may serve multiple roles, and consist of several components of multiple types [18]. Such frankenforms are not well supported by any single-information-form tools.

It is partly for these reasons that simpler, versatile tools such as sticky notes and text editors become so widely used as fall-backs, or sometimes, as a person's primary PIM tool [7]. The flexibility of these tools also means that they are more adaptable as needs change over time. Unfortunately, however, general-purpose tools do not provide many of the capabilities that specific PIM tools support – capabilities such as reminding (e.g., calendar alarms), visualizing events on a map, flexible grouping and sorting capabilities, the ability to cross-reference event invitation lists with invitees' contact information, and so on.

A separate set of issues concerning the use of online PIM services and platforms pertain to access, control, ownership,

and privacy. Consenting to the terms of service for each platform means that people are relinquishing control over their information, how it can be accessed, stored, and guarantees such as long-term persistence and security from disclosure. In some cases, sharing platforms may simply not support sharing information securely or privately (none of the major sharing platforms support encrypted messaging), in which case, people may be compromising the security of their information merely in order to share it.

In this work, we address the aforementioned concerns by proposing a way for personal information environments to be kept unified on the devices that people own and control, and to be accessed in a single place anytime it is needed. Furthermore, instead of being forced to use separate tools to keep information items of different types, our approach blurs boundaries between PIM information forms and allows any information item to be created, managed, shared and collaboratively shaped with any person or entity regardless of its structure or significance. By supporting more natural connections and groupings among information items, individuals have more freedom to construct meaningful organizational schemes that fit their needs, and require less effort to do so.

We refer to our platform as *WebBox*, and it consists of a decentralized personal and social information platform that supports the aforementioned goals while meeting the needs of a wide range of potential collaborative and personal PIM applications. WebBox represents a realization of Socially-Aware Cloud Storage concept proposed by Berners-Lee [6] in which Web applications running on a user's devices gain privileged access a unified, user-controlled data space. By acting as a generic collaboration-centric data storage layer, WebBox handles the complexity of data synchronization, authentication and encryption, allowing PIM application developers to focus on interfaces and functionality rather than data management. This also allows the data model to be integrated, allowing seamless and secure personal information ecosystems to emerge around the individuals themselves, instead of at a handful of "data silo" cloud services.

In the following sections, we describe WebBox by first relating it to previous efforts and platforms, and, briefly describe how it works.

2. RELATED WORK

Two sets of related work are relevant to WebBox. The first consist of efforts focusing on unifying personal data spaces and eliminating fragmentation among (primarily desktop) apps. In this space, "Semantic desktop" efforts are the most notable, including the Nepomuk Project [10] and Haystack [13]. Nepomuk in particular is very much aligned with WebBox in that it provided integrated RDF-based storage services for KDE PIM applications. WebBox similarly uses RDF, but instead of being integrated into any particular desktop is designed to be accessed via any devices or applications that support standard HTTP-based SPARQL protocols. WebBox also makes core to its capabilities distributed collaborative data authoring, which is unique to this framework. Alternative approaches to tackling fragmentation, have included interface-level overlays to existing storage systems demonstrated in such systems as Planz [11] and ProjectFolders [5].

The second set of relevant work concerns "personal data lockers" that consolidate data across online "silos" – namely service providers, online apps, and sharing platforms – into unified data stores under the individual's control. The Mydex Project [4] has produced Higgins [2], a personal data store designed to facilitate Vendor-Relationship Management[1], that is, giving consumers control over and use of personal data collected by vendors. Open-source personal data storage containers include Singly/The Locker Project [15], data.fm [16], Owncloud [3], and OpenStack [1], which provide "personal data storage containers" with slightly different capabilities. Openstack, Higgins and data.fm are generic schema-agnostic data containers that provide simple storage and retrieval APIs for this data, typically via a REST-style API, the latter two which have native support for linked data (RDF). The Locker Project, Owncloud, and Diaspora, meanwhile are social-network inspired and centered around a fixed set of simple data types, such as hosting files, status messages, photos, files, and calendar events – types that, while a start, would likely prove overly restrictive for future PIM tools and services.

While we considered basing WebBox on one of these above, we found none supported the complete set of data handling capabilities derived from popular PIM apps and sharing platforms of our analysis. In particular, secure storage and collaborative data editing were two notable weaknesses of these platforms. Since we saw these emerging platforms as a starting point for the kinds of needs of future PIM tools, we wanted the architecture to make it easy for app writers to support the creation of such apps from the outset.

3. ARCHITECTURE AND CAPABILITIES

In this section, we provide a brief overview of the WebBox architecture, highlighting its components, capabilities, how WebBox applications interface with the platform, and how this design supports the design goals for a new generation of social PIM tools.

Due to space constraints, we refer developers to the WebBox specification [17] for details, and the project codebase[2] where a prototype implementation is available.

3.1 Design

As describe in detail in [17], we derived the requirements for WebBox from an analysis of 25 of the most popular PIM and sharing platforms currently on the Web. From each, we identified the data type(s) and capabilities offered. Based upon a clustering analysis, 8 common capabilities were identified: hosting/storage, support for comments/annotation, link sharing, link re-sharing, group/individual access control, support for surveys and polls, data publishing/sharing and concurrent multi-user editing. We then identified what a storage architecture would need to support all such capabilities. These were as follows: schema-agnostic semi-structured data hosting, granular access control, and sharing. We describe how the WebBox supports each of these.

[1]ProjectVRM — http://cyber.law.harvard.edu/projectvrm/Main_Page
[2]WebBox - http://webbox.ecs.soton.ac.uk

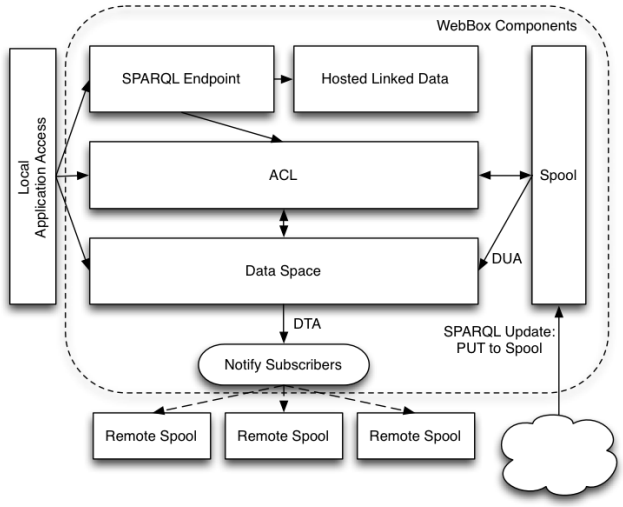

Figure 1: *WebBox Components and Data Flow* - The primary component is a *data container* where files and structured data (in RDF) are kept encrypted; access to this data space is governed by WebACL, which uses WebID for auth. WebBox supports sharing and collaboration through a distributed pub-sub data model in which updates cause notifications to be generated and delivered directly to others' WebBoxes in a peer-to-peer fashion. Local apps can talk to the WebBox via a variety of Web protocols (REST/JSON, RDF/XML, SPARQL) to provide flexibility and platform-agnostic for app writers.

3.2 Components

Unlike the current infrastructure of the Web, where we use browsers to access content served up by Web servers over which we have little control, WebBox makes the assumption every individual will have their own WebBox, a data store and HTTP server which hosts and securely maintains their data and which mediates interactions with others' WebBoxes to manage shared data artifacts. A WebBox can run on the user's own device(s), or on a virtual host such as an Amazon S3 instance. The main components of WebBox are illustrated in Figure 1 and described next.

3.2.1 Data space

The primary purpose of WebBox is to serve as a repository for data objects. Since many of the kinds of data people share consist of semi-structured *micro-data* – for example, Tweets, Foursquare "check-ins", social bookmarks, Facebook "Likes" and so on, we wanted WebBox to support storage and sharing spanning these small structured information forms. To support both such structured microdata and unstructured data (such as photos and videos), our reference implementation uses two stores: a query-optimized RDF triple-store and a filesystem-based hosted linked data store.

3.2.2 Access control

Access to this data space is governed by a "Sharing and ACL" layer that lets users and applications (on behalf of users) configure access and sharing policies. The design goal of the ACL was to make access control easy for users to understand and to minimize both the effort required to manage them and likelihood of error doing so. Towards this end, WebBox combines sharing and access control policies into its sharing policies; if people are granted access to a particular resource (access control), they can, in turn, specify to be notified when the resource is updated (sharing). Doing so ensures that access control and sharing policies never contradict one another, and that people can think of access control as *"who should have access to this?"* instead of having to consider multiple factors. These policies can be highly granular (per-resource) or broader (sets of resources, per directory/location/graph) as needed. An example of such an access policy might be a policy that automatically grants access to anything tagged "family" to one's family members.

3.2.3 Messaging

The Messaging architecture is responsible for the collaborative aspects of WebBox, in particular notifying remote WebBoxes when shared data artifacts change or are created anew, and receiving corresponding update notifications from remote WebBoxes. Messages are small RDF records that are directly PUT in to a user's **Spool** file by remote hosts. This causes the WebBox **Data User Agent (DUA)** to take messages from the spool, compare the sender against the ACL-Sharing policies, and, finally to determine whether or not to update the local user's data space with the full update from the remote WebBox. The **Data Transfer Agent (DTA)** watches for changes in the data space and notifies people with whom an entity is shared when it has changed, or when new data has been published.

3.3 Applications

The relationship between apps and their data in WebBox is fundamentally different that of current apps and online services; in the current model, apps completely internalize and control the data created therein using internal databases and logic, making integrated views of data created using multiple apps challenging. The situation for Socially-Aware Cloud platforms like WebBox [6] is similar to that of Semantic Desktops in that most of the data is externalized into a common, consolidated (WebBox) data model. (In WebBox, apps can optionally "make private" components of entities for internal use such as to store program state.) The overall result of this shared data layer is that applications essentially turn into interchangeable "lenses and pickaxes" of a common unified personal data space. In order to achieve true interchangeability, however, externalizing the data is not enough; rather, common reprsentations must be agreed upon so that multiple apps can unambiguously read and write data written by other apps. This is where the use of RDF and common ontologies (such as Dublin Core Metadata Terms[3], vCard[4], RDFCalendar[5] and FOAF[6] play an essential role.

Having an externalized "data space" dissociated from apps

[3]Dublin Core — `http://dublincore.org/specifications/`
[4]vCard — `http://www.w3.org/TR/vcard-rdf/`
[5]RDFCalendar — `http://www.w3.org/TR/rdfcal`
[6]Friend of a Friend - `http://xmlns.com/foaf/spec/`

themselves is also desirable from a user-experience standpoint because personal data usually out-lives applications used to manage them. Thus, by keeping personal data securely stored in a way that is application independent and separate means it can be maintained independently as well – ensuring its longevity and sustainability.

Unlike in Platform-as-a-Service environments [14] which mandate particular supported programming languages, frameworks, and APIs, WebBox applications can be written in any language and on any platform, provided that HTTP and HTTPs-calls can be made and standard data formats can be parsed. A number of different APIs are provided to facilitate application development, including a pure REST+JSON API that produces a "document store" illusion for app writers that do not wish to use RDF and SPARQL.

3.4 Ongoing and Future Work

Beyond our reference implementation of a WebBox server[7] we have been working on client libraries for various languages to enable developers to easily write apps that read and write to the shared data model using common ontologies, as well as collaborative updating of shared data objects in this model. Our current client libraries are written entirely in Javascript and are suitable for use in browser extensions, including Chrome and Firefox.

Our next work includes making version control intrinsic to the WebBox API so that collaborative apps can ensure consistency with minimal application-level complexity.

4. DISCUSSION AND CONCLUSION

In this position paper, we have discussed the pervasive problem of PIM fragmentation and the use of cloud PIM and sharing services that force users to relinquish control over their data.

Our proposed solution is to re-think web architectures fundamentally to eliminate the need for central silos of control, instead taking advantage of the computational resources at the edge – the fact that the personal devices will continue to get more powerful, connected, and have greater capacity implies there is plenty of opportunity to take advantage of it for people's direct benefit. Our architecture requires application writers to externalize personal information people create into a common personal data store, so that fragmentation can be avoided, and that such data can be re-purposed by multiple applications. This allows data to remain consolidated while permitting the construction of multiple views, input interfaces and manipulation affordances provided by independent software providers. At the same time, WebBox takes care of complexities of data secure, replication, and collaboration so that sharing and security become first class capabilities inherent in all PIM applications.

5. REFERENCES

[1] Open source software for building public and private clouds. *www.openstack.org*.

[2] Higgins Personal Data Services. *Eclipse wiki* *eclipse.org/higgins*, 2009.

[3] Web services under your control. *OwnCloud* *owncloud.org*, 2010.

[4] D. Alexander, A. Mitchell, and W. Heath. Mydex Community Prototype Launch Press Release. 2010.

[5] O. Bergman, R. Beyth-Marom, and R. Nachmias. The project fragmentation problem in personal information management. In *Proceedings of the SIGCHI conference on Human Factors in computing systems*, CHI '06, pages 271–274, New York, NY, USA, 2006. ACM.

[6] T. Berners-Lee. Socially Aware Cloud Storage. *W3C Design Note* *http://www.w3.org/DesignIssues/CloudStorage.html*, 2009.

[7] M. Bernstein, M. Van Kleek, D. Karger, and m. schraefel. Information scraps: How and why information eludes our personal information management tools. *ACM TOIS*, 26(4):1–46, 2008.

[8] R. Boardman. Improving tool support for personal information management, September 2004.

[9] R. Boardman, R. Spence, and M. A. Sasse. Too many hierarchies? the daily struggle for control of the workspace. In *Mathematical ModelsâĂİ, Proceedings of the ACM CHIâĂŹ96 Conference*, pages 406–412. ACM Press, 2003.

[10] S. Decker and M. Frank. The Social Semantic Web. *DERI Technical Report* *http://www.deri.ie/fileadmin/documents/DERI-TR-2004-05-02.pdf*, 2004.

[11] W. Jones, D. Hou, B. D. Sethanandha, S. Bi, and J. Gemmell. Planz to put our digital information in its place. In *Proceedings of the 28th of the international conference extended abstracts on Human factors in computing systems*, CHI EA '10, pages 2803–2812, New York, NY, USA, 2010. ACM.

[12] V. Kalnikaité and S. Whittaker. Software or wetware?: discovering when and why people use digital prosthetic memory. In *Proc. CHI '07*. ACM Press, 2007.

[13] D. R. Karger and D. Quan. Haystack: a user interface for creating, browsing, and organizing arbitrary semistructured information. In *CHI '04 extended abstracts on Human factors in computing systems*, CHI EA '04, pages 777–778, New York, NY, USA, 2004. ACM.

[14] P. Mell and T. Grance. The nist definition of cloud computing. *National Institute of Standards and Technology*, 53(6):50, 2009.

[15] J. Miller, S. Murtha-Smith, and Team. The Locker Project. *lockerproject.org*, 2010.

[16] J. Presbrey. Read Write Linked Data Space. *data.fm* *data.fm*, 2010.

[17] D. A. Smith, M. V. Kleek, O. Seneviratne, monica schraefel, A. Bertails, T. Berners-Lee, W. Hall, and N. Shadbolt. Webbox: Supporting decentralised and privacy-respecting micro-sharing with existing web standards. In *WWW2012: 21st International World Wide Web Conference*, 2012.

[18] M. G. Van Kleek, M. Bernstein, K. Panovich, G. G. Vargas, D. R. Karger, and m. schraefel. Note to self: examining personal information keeping in a lightweight note-taking tool. In *Proc. CHI '09*. ACM Press, 2009.

[7] Available under the MIT License at `http://webbox.ecs.soton.ac.uk`