Using Personal Information Management Infrastructures to Facilitate User-Generated Services for Personal Use

Olaf Grebner

SAP Research, Vincenz-Priessnitz-Str. 1, 76131 Karlsruhe, Germany
Olaf.Grebner@sap.com

Abstract. Ad-hoc and situational applications for personal use will gain more and more traction in the work support for knowledge workers (KWers). Personal information is a key element in these applications. Composition environments for situational applications like, e.g., Yahoo Pipes, enable end-users to compose services into an application targeting their individual problems. However, we analyze that these composition environments lack access for a KWer's personal information and require redundant development of services for common KWer activities. Addressing these issues, we present [1]an infrastructure that manages the KWer's personal information consistently and thus provides services that serve as basis for enabling end-user driven service composition for application for personal use. The infrastructure consists of two key components, a basic personal information management system to maintain a KWer's personal information cloud in a unified and integrated form and domain-specific services that offer business logic for frequently occurring activities in applications for a KWer's personal use.

Keywords: Service design, User-centric software development, Personal information management

1 Introduction

Ad-hoc and situational applications for personal use will gain more and more traction in the work support for knowledge workers (KWers). The work conducted by *knowledge workers* is characterized by variety rather than routine and there are few activities that can be automated. Each KWer possesses an individual working style and skill level making it hard to design applications that fit to a broad range of KWers. *Situational applications* enable a KWer to adapt services to create functionality and solve problems in their work. This allows the KWer to leverage an individually created application tailored to the needs of that individual KWer for the particular use case. The *advantages of situational applications* supporting a KWer are manifold. On the one hand, the application fits to the KWer's individual skill level, targeting both novices and experts appropriately. On the other hand, KWers being

[1] This work was has been partially funded by the European Commission as part of the NEPOMUK IP (Grant FP6-027705) and of the MATURE IP (Grant FP7-216346).

experts in their respective domain can come up with design ideas for solutions target their individual problems of which they have a deep understanding.

Personal information is a key element in applications for personal use. A KWer deals with personal information in applications for personal use. A KWer's *personal information* refers to the information that is owned by an individual KWer. This includes for example email messages in an email account or files on the computer's hard drive. A *personal information cloud* (PIC) is "the 'working set' of information that is relevant to the individual and his work" [Moran&Zhai, 2007, p. 338], i.e., the information an individual KWer manages for herself and deals with in the activities she executes. The KWer regards the personal information as "*one single body of information*" [Ravasio&Tscherter, 2007, p. 275], complementary to the personal information cloud. "Ravasio et al. (2004) indicate that users explicitly desire linking: 'most interviewees expressed the need to have their information linked together, e.g., article author and respective address book entry, or citation and cited article, etc.' (p. 169)" [Jones&Teevan, 2007, p. 143] [Ravasio et al., 2004].

Composition environments for situational applications *lack access for a KWer's personal information* and require redundant development of services for common KWer activities. Research fields such as end-user-development acknowledge the importance of situational applications where individuals drive the creation of an application to support their individual problems. On the web, mash-up services enable an individual KWer to perform a specific form of end-user development already today. Mash-up services like, e.g., Yahoo Pipes [Yahoo! Inc., 2009] allow end-users to create custom situational applications for a particular use case by visually composing web-based services and designing corresponding user-interfaces. However, these mentioned mash-up services lack two core elements when targeting applications for personal use:

- Consistent access to a KWer's personal information is not possible, as the KWer's personal information is scattered across multiple applications. This is a key requirement as the KWer needs to have her personal information available during work with these applications.
- High effort for developing applications in the personal domain due to a lack of services that offer business logic for common, recurring activities in the KWer's personal domain, especially when dealing with personal information.

Personal information management research started to integrate the KWer's personal information cloud into such composition environments, for example Konduit [Dragan et al., 2009]. However, Konduit focuses solely on integrating desktop data into the built applications, not tackling the problem of high development effort for domain-specific components dealing with a KWer's activities.

The next section discusses in detail the respectively underlying problems of these issues. These issues so far prevented the large-scale adoption of mash-up services for personal use. To address these issues, we present an infrastructure that manages the KWer's personal information consistently and provides services that serve as basis for enabling end-user driven service composition for application for personal use. The infrastructure consists of two key components:

- We leverage a *basic personal information management system* to maintain a KWer’s personal information cloud in a unified and integrated form by using a semantic desktop. Services can access all of the KWer’s personal information by querying the personal information cloud’s consistent and integrated set of information.
- We provide *domain-specific services* for composition that cover common business logic which frequently occurs in applications for a KWer’s personal use. These domain-specific services cover for example activities like task management or meeting management. They enable the end-user to access and act on the personal information and represent a basic infrastructure for the KWer’s personal domain in end-user driven composition environments like the mentioned mash-up services.

The here presented infrastructure offers service composition environments the ability to access a KWer’s personal information in a consistent manner and thereby offers re-usable services that match a KWer’s activities.

The remaining paper is structured as follows: We first discuss the core problems underlying for the identified issues of situational applications for personal use. This includes on the one hand the personal information fragmentation in workspaces which prevents efficient use of personal information as well as on the other hand that KWers conduct a common set of supporting activities in their work which leads currently to redundant implementations of supporting business logic. Then we propose a personal information handling infrastructure that facilitates creating user-generated services for personal use. It consists of the three elements of functionally-oriented applications, domain-specific services and a basic personal information management (PIM) system. We then discuss the benefits of this infrastructure and give a summary and an outlook.

2 Core Problems of Situational Applications for Personal Use

2.1 Personal Information Fragmentation in Workspaces Prevents Efficient Use of Personal Information

The KWer’s *personal information is fragmented* across applications in digital workspaces, i.e., the *KWer’s personal information is scattered across the desktop and its applications* [Jones, 2008]. Current desktop systems and applications don't represent the personal information cloud in a unified and integrated form as the KWer expects it, i.e., the personal information is fragmented across applications in the workspace. E.g., an email client folder containing emails has for the KWer no visible connection to a file system folder containing documents despite dealing with the same topic.

Workspace applications *managing only a particular type of personal information and locking it in* a proprietary storage causes one major type of personal information fragmentation. Karger mentions that these applications “often store their data in their own particular locations and representations, inaccessible to other applications" [Jones&Teevan, 2007, p. 127]. Karger outlines this type of information fragmentation as paradox as this “information is fragmented by the very tools that have been designed to help us manage it” [Jones&Teevan, 2007, p. 127]. Jones as well sees here "many examples of seemingly avoidable information fragmentation" [Jones, 2008, p. 392].

Another related type of *information fragmentation is produced by the KWer herself*. Due to application information silos, the KWer is forced to fragment the own personal information when keeping it. This is due as each application only manages particular own information types, i.e., information silos, regardless of the KWer's intent for keeping the information. Bergmann reports this type of personal information fragmentation problem in the context of organizing project-related information, called project fragmentation problem [Bergman et al., 2006]. Figure 1 shows an example of "information related to a chemistry course" which is "fragmented into separate collections" across three applications. The involved applications' ability of managing each only a particular information element type forces the KWer to organize the information in a fragmented way.

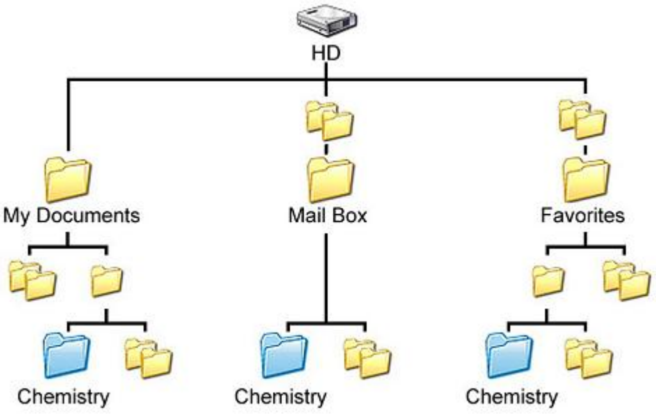

Fig. 1. Personal information fragmentation example – project-related [Bergman et al., 2006].

2.2 KWer Conduct a Common Set Of Supporting Activities in Their Work

The KWer performs activities in a personal value chain in analogy to an enterprise’s value chain [Porter, 1998], see Figure 2. The KWer’s activities can be categorized into the two types of personal primary activities and personal support activities.

Personal primary activities are the working activities where the KWer actually works towards achieving the individual goals. They depend largely on the KWer’s job role and the KWer’s individual skills and capabilities. By executing personal primary activities, the KWer produces the output that she gets rewarded for.

Personal supporting activities enable the KWer to manage herself and things that support her. They are independent of the KWer’s job role as they concern each individual KWer. These supporting activities are cross cutting primary in the execution through their support for primary activities. They as well vary in their concrete instantiation from KWer to KWer depending on, e.g., the KWer’s job role, gathered experience, felt importance and skill level. For example, KWers with a busy schedule and workload feel a more urgent need to perform efficient task and time management than KWers whose workday leaves sufficiently enough time to cope with the workload. The following list covers the major supporting activities without claiming to be exhaustive.

- Task Management (Task): Plan, structure and prioritize a set of tasks.
- Time Management (Time): Plan and control available time.
- Personal Social Network Management (People): Build personal social network, maintain the network and activate nodes within the network as needed [Fisher&Nardi, 2007, p. 171].
- Meeting Management (Meeting): Prepare, conduct and post-process meetings.
- Information Management (Information): Collect, organize, browse and retrieve information.
- Collaboration Management (People, Information): Communicate and interact with people.

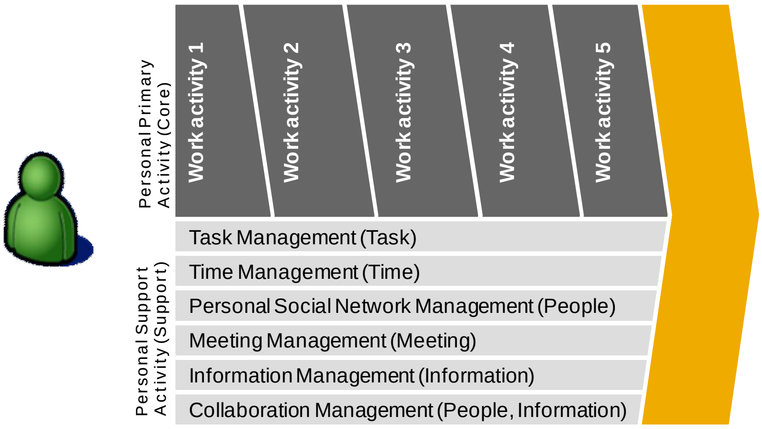
Personal Primary Activity (Core)
Work activity 1
Work activity 2
Work activity 3
Work activity 4
Work activity 5
Personal Support Activity (Support)
Task Management (Task)
Time Management (Time)
Personal Social Network Management (People)
Meeting Management (Meeting)
Information Management (Information)
Collaboration Management (People, Information)

Fig. 2. Personal primary and personal supporting activities conducted by a KWer.

The KWer's personal activities always include information processing activities. Information processing activities include the KWer's (re-)acquiring, processing, organizing, and (re-)distributing of information [Jones, 2008, p. 60]. The information processing activities represent an orthogonal, information-centered perspective on the activities that a KWer conducts, compared to this activity-centered perspective.

3 Personal Information Handling Infrastructure Facilitates Creating User-Generated Services for Personal Use

We present an *infrastructure* that tackles the identified issues when developing user-generated applications for personal use. The proposed infrastructure consists of three layers [Grebner, 2009]. Figure 3 shows on the left these layers from an end-user perspective and shows on the right an architectural view on the same layers from an application developer's perspective. In the following we explain each layer in detail.

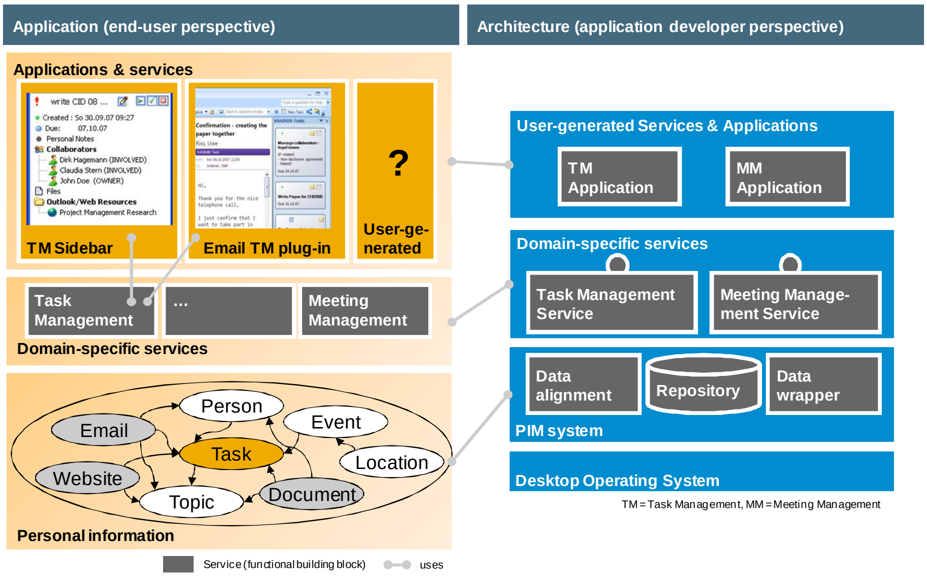

Fig. 3. Personal information cloud and domain-specific services serve as basis for user generated services and applications.

3.1 Pre-built and User-Generated Applications for Personal Use

First, facing the end-user, *functionally-oriented applications* each focus on supporting a particular type of a KWer's activity. Figure 3 shows two different user interfaces targeting a KWer's task management activities in different situations. On the one hand, a task management (TM) sidebar [Grebner et al., 2008] shows the KWer's tasks and related information in a dedicated application focusing on task management. On the other hand, an email TM plug-in enables the KWer to conduct task management in the email client Microsoft Outlook while focusing on the tasks related to emails.

These two vastly different applications base on the same information, i.e., a task shown in one application can appear in the other application. Each application represents a distinct perspective on the common underlying unified personal information set. These applications can be implemented using multiple programming languages and technologies. The two here presented applications have been built in Java Swing and C#, respectively. Further applications for both the task management and meeting management domain can be found in [Grebner, 2009].

The vision fueled by the results of the here mentioned applications is that the KWer can leverage composition environments for situational applications to create user-generated applications for personal use. This happens by composing an application that addresses the KWer's individual activities and problems using the offered domain-specific services, see the next section for details. For example, a

KWer composes a task management application following her own ideas on how to conduct task management. By using the proposed infrastructure with domain-specific services and the basic PIM system the task information produced by the composed application can be integrated with the KWer's personal information cloud and thus neatly works with other existing applications of the KWer.

3.2 Domain-Specific Services

Second, a set of *domain-specific services* handles the access to a particular perspective on a KWer's personal information from the application layer to the unified information model. The domain-specific service's functionality provides the needed parts of the personal information to domain-knowledgeable developers. This way, the user experience of each application can be designed solely relying on the domain and completely independent from the underlying (operating) system and associated metaphors and paradigms. Domain-specific services provide controlling business logic for applications on the application layer in the form of supporting functionality and commonly handling the access to a corresponding perspective of the KWer's unified information model. We implemented domain-specific services targeting the two personal supporting activities task management and meeting management, see Figure 3.

There are particular services for defined types of a KWer's activities. The services include functionality needed to support these activities. This includes, but is not limited to, the management of a corresponding perspective on a KWer's unified personal information. Typically focusing on a major entity of the information model, as for example tasks or people, the particular domain-specific service handles this perspective and the major entities and contains business logic needed to process these items. For example, for tasks there are service methods that create a new task, retrieve tasks by different criteria or methods to delegate a task.

The domain-specific service encapsulates the domain-specific business logic in a way that domain-knowledgeable KWers can apply these services without being experts in personal information management. As this domain-specific business logic is encapsulated in a service it can be re-used at every workspace or workspace-level application that needs to access an entity of the information model.

On the implementation side of the domain-specific services, their business logic interfaces core services provided by the basic PIM system such as, e.g., storage and communication. The domain-specific services' business logic interfaces the basic PIM system by invoking its services like, e.g., a PIM service to read and write the unified information model and a data wrapper for handling desktop information elements like files and emails. In our implementation with the Nepomuk semantic desktop as basic PIM system, they interface the core system services like for example PimoService [Sauermann&Klinkigt, 2009] to read and write the unified personal information model and the Aperture DataWrapper [Aduna B.V.&DFKI GmbH, 2005] as indexer for desktop information elements like files and emails.

3.3 Basic PIM System

Third, using a *basic PIM system* we can access numerous desktop services and thus re-use desktop functionality without replicating it in each workspace. Here we use the semantic desktop *Nepomuk* [Groza et al., 2007] as example for a service-oriented basic PIM system. The Nepomuk semantic desktop provides a service-oriented desktop architecture and services. The basic PIM system manages the unified personal information model representing the KWer's personal information cloud and offers a PIM Service providing generic read and write functionality for the unified personal information model. Nevertheless, it is user-owned but formally represented using a set of ontologies. We use a formalized representation of this model, the Personal Information Model Ontology (PIMO) [Sauermann et al., 2007] to enable machine processing on it by the domain-specific services. It features a unified representation through the RDF representation and it is integrated along the KWer's personal things. This represents the KWer's mental model, i.e., the personal, subjective view on the world. Furthermore, it encapsulates core operating functionality in services like, e.g., handling desktop information objects like for example, emails, websites and files.

4 Discussion of Benefits

Through using the proposed infrastructure, developers and end-users in a role to compose an application gain a number of benefits. First, it significantly reduces the complexity of implementing applications.

Offering a domain-specific interface reduces the complexity to build an application by opening the development of personal activity support applications to domain-knowledgeable developers while omitting the need for deep personal information management expertise. The domain-specific interface of the domain-specific services allows domain-knowledgeable developers to efficiently handle unified personal information. A domain-specific interface hides the underlying personal information management technology details. In comparison to a generic personal information management interface this saves domain-knowledgeable developers getting into the internals of personal information management and related technology.

A *common PIM system infrastructure* reduces the complexity to build a personal activity support applications by replacing the need for applications and the domain-specific services to each redundantly implement a proprietary stack for integrating desktop information objects. The common PIM system infrastructure underlies the applications and the domain-specific services. The PIM system, i.e., the semantic desktop, provides services and models to manage existing desktop information elements and to integrate them with the unified information model. This means that the PIM system infrastructure is re-usable instead of each application building an own management stack. This replaces the need for applications and the domain-specific services to each redundantly implement a proprietary stack for integrating desktop information objects, significantly reducing the complexity of implementing applications.

Second, using this reference architecture significantly improves the capabilities that developers can implement into personal activity support applications.

An *increased flexibility for building applications* can be realized as existing business logic for certain KWer activities can be re-used and only the user interface needs to be re-developed and designed, for example using the described composition environments for situational applications. The domain-specific services provide a number of services to realize the functionality needed for supporting a KWer's activities like, e.g., task delegation to send off a task to other people. As long as these services are available, developers can quicker develop new applications featuring an improved user interaction compared to when they need to re-develop the full personal information management stack. The service-oriented desktop architecture of the Nepomuk semantic desktop PIM system facilitates this re-use further as it keeps the available services in a service registry allowing other applications to discover and use these services. This enables multiple applications that are implemented in different programming languages to invoke the provided services. By invoking services, multiple applications can, e.g., query and write to the common unified personal information model.

5 Summary and Outlook

To summarize, the here presented infrastructure offers service composition environments the ability to access a KWer's personal information in a consistent manner and thereby offers re-usable services that match a KWer's activities on a domain-specific level. This enables KWers to compose applications and services for personal use. Alternatively, it as well eases the work of developers developing respective components. At hand of task and meeting management services and two example applications we showed how tailored application "hats" can re-use common business logic to offer services and applications in the KWer's personal support domain.

Due to the service-based nature of this infrastructure, we want to enable service composition environments to consume the domain-specific services and thus offer application support for the personal use domain of KWers. The here presented principle of a personal information cloud and domain-specific services has been implemented for personal information on the desktop. The KWer's personal information is thus securely stored in an information repository that is accessible only to the one individual KWer owning it. To enable these services for use by the mentioned service composition environments, the KWer's personal information cloud, the used basic PIM system and the domain-specific services will be ported into web-based services and a multi-user environment while maintaining the privacy of the KWer's personal information cloud. In addition, the next implementation version will incorporate as well personal information already managed by web-based services, such as for example social networks, online storage providers as well as online email providers.

References

Aduna B.V.; DFKI GmbH (2005). *Aperture.* http://aperture.sourceforge.net. Accessed 2009.10.31.

Bergman, O., Beyth-Marom, R. & Nachmias, R. (2006). The project fragmentation problem in personal information management. *CHI '06: Proceedings of the SIGCHI conference on Human Factors in computing systems (p.* 271-274). New York, NY, USA: ACM.

Dragan, L., Möller, K., Handschuh, S., Ambrus, O. & Trüg, S. (2009). Converging Web and Desktop Data with Konduit. *Proc. of Scripting and Development for the Semantic Web Workshop.* http://CEUR-WS.org/Vol-449/Paper4.pdf. Accessed 2009.10.31.

Fisher, D. & Nardi, B. (2007). Soylent and ContactMap: Tools for Constructing the Social Workscape. In V. Kaptelinin & M. Czerwinski (Eds.), *Beyond the desktop metaphor: designing integrated digital work environments (p.* 171-190). MIT Press, Cambridge, Mass.

Grebner, O. (2009). Using Unified Personal Information in Workspaces. PhD thesis. http://digbib.ubka.uni-karlsruhe.de/volltexte/1000012169. Accessed 2009.10.31.

Grebner, O., Ong, E. & Riss, U. (2008). KASIMIR - Work process embedded task management leveraging the Semantic Desktop. In M. Bichler, T. Hess, H. Krcmar, U. L. F. Matthes, A. Picot, B. Speitkamp & P. Wolf (Eds.), *Multikonferenz Wirtschaftsinformatik (p.* 715-726). GITO-Verlag, Berlin.

Groza, T., Handschuh, S., Moeller, K., Grimnes, G., Sauermann, L., Minack, E., Mesnage, C., Jazayeri, M., Reif, G. & Gudjonsdottir, R. (2007). The Nepomuk project-on the way to the social semantic desktop. *Proceedings of I-Semantics (*Vol.7, S. 201-211).

Jones, W. & Teevan, J. (2007). *Personal Information Management* University of Washington Press.

Jones, W. (2008). *Keeping Found Things Found* Elsevier Inc.

Moran, T. P. & Zhai, S. (2007). Beyond the Desktop Metaphor in Seven Dimensions. In V. Kaptelinin & M. Czerwinski (Eds.), *Beyond the desktop metaphor: designing integrated digital work environments (p.* 335-354). MIT Press, Cambridge, Mass.

Porter, M. E. (1998). *Competitive advantage: Creating and sustaining superior performance* Free Press.

Ravasio, P. & Tscherter, V. (2007). Users' Theories of the Desktop Metaphor, or Why We Should Seek Metaphor-Free Interfaces. In V. Kaptelinin & M. Czerwinski (Eds.), *Beyond the desktop metaphor: designing integrated digital work environments (p.* 265-294). MIT Press, Cambridge, Mass.

Ravasio, P., Schär, S. G. & Krueger, H. (2004). In pursuit of desktop evolution: User problems and practices with modern desktop systems. *ACM Trans. Comput.-Hum. Interact., 11*(2), 156-180.

Sauermann, L. & Klinkigt, M. (2009). *PIMO Service.* http://dev.nepomuk.semanticdesktop.org/wiki/PimoService. Accessed 2009.10.31.

Sauermann, L., Van Elst, L. & Dengel, A. (2007). Pimo-a framework for representing personal information models. *Proceedings of I-Semantics (p.* 270-277).

Yahoo! Inc. (2009). *Yahoo Pipes.* http://pipes.yahoo.com/pipes/. Accessed 2009.10.31.