Networked PIM using PDMS

Alexander Albrecht Felix Naumann Armin Roth

Hasso-Plattner-Institut at the University of Potsdam, Germany
{albrecht,naumann,roth}@hpi.uni-potsdam.de

Abstract

Personal information management (PIM) is a promising new type of application allowing not only to search a desktop, but to pose complex, structured queries against the data on ones computer. We propose to remove the confines of PIM and make selected data available to a network of peers using peer data management system (PDMS) technology. The result is an application for collaborative information management within workgroups.

To achieve this vision, several participating tools and technologies must be adapted: PIM systems must be augmented with privacy concepts to protect non-public data, which in turn must be interpreted by the PDMS query rewriting mechanisms. The entity resolution methods of individual PIM systems must be extended across multiple PIM systems with possibly heterogeneous schemata and must support ad-hoc queries. Finally, traditional PDMS are designed to find the complete and correct query result. In a networked PIM application it is not necessary to find all results to a query and it is acceptable to respond with inexact results.

1 Personal information management across peers

The range of applications to save and manage personal data is wide and ranges from email, address book, calendar, and to-do-lists up to presentations and publications. Additionally, there is a variety of platforms to perform these applications, ranging from the classical personal computer, to PDAs, and to webspaces with personal blogs.

This huge number of applications and platforms leads to a heterogeneity and fragmentation of data digitally available. There are several approaches to sum up and query this data in so called personal information management systems (PIMS). A first approach is to enable full text search within the entire amount of data (emails, files, bookmarks, etc.) as supplied for instance by Google Desktop Search or Apple Spotlight.

More complex queries are made possible by systems such as SEMEX or iMemMex: The SEMEX system organizes data of different sources as objects and connects these objects semantically. Personal information can be requested with meaningful semantic connections [3]. iMemMex also integrates information of different data sources into a common data model, but it does not perform semantic data integration; duplicates or correspondences between attributes of different schemata are not taken into account nor automatically detected. This approach allows rapid deployment and one can query all data across data sources and file boundaries any time [2]. In [5] the authors suggest a distributed version of iMeMex; we go a step further - networked PIM are not only distributed but also shared across users.

Apart from integrating personal data into a global schema, we propose that PIMS exchange information with other persons within a network. Through this process information can be published in the network of workgroups or companies. This approach allows several types of applications, which we present in Section 4. Even a single person may have the need to integrate data on different devices such her desktop computer, her laptop, her web-mail account, and her PDA.

The structure of this network, where information is exchanged, is characterized by its dynamics. New information, data sources, and PIMS are plugged in, while other PIMS simultaneously plug out or suddenly fail. It is also possible that PIMS are concurrently active within several networks.

Apart from the dynamics of the network one has to consider the aspect of specific data models of PIMS. Information can be modeled in object-oriented structures in one PIMS while the informa-

tion is organized hierarchically or relationally in a second PIMS. Even if PIMS organize information in the same data model, their schemata may differ. For example, if one PIMS extracts personal data from an email application as well as from an address book, then the person schema would also include phone and address in addition to name and email address. In a second PIMS that has only access to the email application, the person schema would include only name and email address.

Because of the specific schemata and data models of the involved PIMS and because of the dynamics of the network, already the task of setting up a network of a few PIMS is time-consuming and non-trivial. The management of a large-scale network through a single central instance is hardly feasible. In consequence, such a network must be organized decentrally.

This approach provides the following advantages to potential users.

Scalability. Only a small set of PIMS are affected during a network expansion.

Reflects reality. Networked PIMS corresponds to the familiar approach for private or personal communication.

Transparency. The user's data schema is used and no integrated and possibly changed global schema. Thus no enhancement of a user's PIMS schema is necessary and he can query all available external data sources from within.

Rapid deployment. A PIMS is characterized by its portability and can be set-up successfully in various networks, i.e., different project groups or social networks.

The decentral structure of the peer data management system (PDMS) architecture, introduced in Section 2.1, offers a perfect possibility to set up such a network. A PIMS provides its data as one peer. In a PDMS, peers are connected by mappings between the schemata describing the data the peers export. Because of the absence of a global schema, which should reflect the information of *all* peers, only local coordination between a small set of peers is needed to extend the network of peers.

In general, schema mapping is a difficult task, but usually a peer establishes mappings to peers it knows quite well, and generation of mappings can be supported by tools such as Clio [15].

There are different implementation approaches of how one PIMS can exchange information with other peers in a PDMS.

Re-Programming. If available, the (Java-) sources of the PIMS are directly extended with PDMS functionality.

Interfaces. The PDMS service installed on a peer interacts through existing interfaces, i.e., the WebDav-Interface of iMemMex, with the local PIMS. The service manages communication between PIMS and PDMS (s. Fig. 1).

Plugins. Build a PDMS plugin for the PIMS, besides the existing plugins such as PDF- or email plugins. I.e., the set of connected peers acts as just another data source for the local PIMS. This approach is similar to the indexing-plugins of Google Desktop Search. The PDMS-plugin uses PDMS to integrate other PIMS as external data sources.

An additional functionality that must be newly considered and implemented is privacy management. Every user should be able to specify at different levels of granularity those resources that are made accessible by other PIMS through the PDMS. One authorization method would be the explicit selection of individual data sources, i.e., "share only documents in a publication or presentation folder". Another authorization method would be the use of (preformulated) rules, i.e., "share all emails sent to the project group". A more detailed analysis of privacy aspects can be found in Section 3.3.

2 Peer data management for personal and workgroup data

To extend PIMS from one's personal desktop to external sources, so-called peer data management systems (PDMS) can serve as an infrastructure for query answering and keyword search. This section explains why this kind of systems is ideally suited to support integration of personal and workgroup data. Additionally, we describe some difficulties that arise when using PDMS in this context.

2.1 Characteristics of PDMS

Peer data management systems (PDMS) are a highly dynamic, completely decentralized infrastructure for large-scale data integration [4, 10, 17]. They consist of a dynamic set of autonomous peers. Each peer offers data to others, which in our context are described by the so-called PIMS interface (s. Fig. 2), which plays the role of a peer export schema. The peers in a PDMS are inter-connected with schema mappings forming a network. Queries submitted at a peer are answered with local data and with

data that is reached by repeated query reformulation along paths of mappings. PDMS follow a virtual approach to data integration, i.e., queries are transported to the data instead of carrying *all* the data to the locations the queries are submitted to as it is done in the materialized approach.

2.2 PDMS as an infrastructure for networked PIMS

PIMS has several requirements to the underlying infrastructure. First, users desire to have their customized view on their personal and other people's data. So each PIMS acts as an individual peer. The corresponding PIMS interface reflects the personal view on the data. The PIMS of the members of a workgroup are connected by mappings between their PIMS interfaces. This means that individuals can establish a relationship with a small number of other PIMS. In effect, in a PDMS each PIMS has indirect access to all the others PIMS their neighboring peers can reach in turn. Observe that this is achieved solely by local coordination rather than globally negotiating a common schema. So PDMS provide high flexibility with low effort to connect a set of other PIMS already organized in form of a PDMS.

Example. In Fig. 1, PIMS_4 connects to the already existing networked PIMS consisting of PIMS_1, PIMS_2, and PIMS_3 by establishing a single mapping to PIMS_1. As a consequence, PIMS_4 has (possibly limited) access to PIMS_2, and PIMS_3 as well.

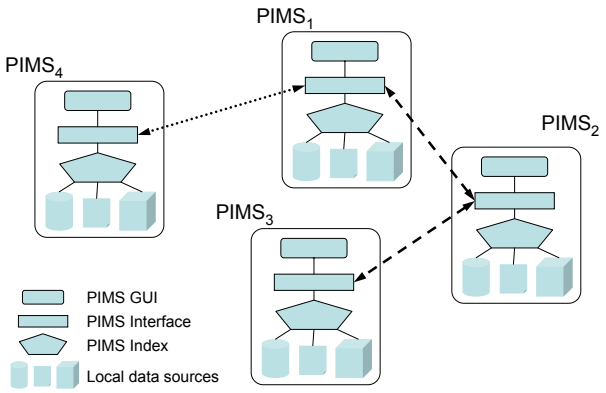

Figure 1: A PIMS connecting to an already existing networked PIMS.

Peers of a PDMS are required to route queries they received to their neighbors and return a query result obtained by merging these answers. We can assume that within workgroups or even in companies people are willing to pass queries received by their PIMS to neighboring peers, i.e., PIMS, and return result data, because participants aim to maximize the success of the organization. For the same reason, they may additionally return information about the origin of the data (lineage).

Due to high redundancy in the mapping network, query answering in PDMS turns out to be inefficient. Usually, result data is redundantly transported through the network of peers on different paths. Even for tens of peers query answering in PDMS tends to be intractable. However, we argue that in networked PIMS users are willing to make concessions to the completeness of query answers. In previous work, we have shown that compromising completeness can increase the efficiency of PDMS query answering considerably [17].

2.3 Extensions to current PDMS

A major challenge to use PDMS for PIMS is to extend the current approaches by a functionality for keyword search and possibly browsing, i.e., interactively proceeding from one object to others following links. In particular, it is not immediately clear how keyword search can interact with query answering. An even harder problem is to logically and physically distribute keyword search over the PDMS peers [9].

Data cleansing, in particular duplicate detection, is needed for merging information from disparate sources both for answering of structured queries and for keyword search. In individual data integration systems (DIS), data cleansing is highly developed. In PDMS, such techniques, e.g., the sorted neighborhood algorithm [13], have to be distributed across the peers, each of which acts as a DIS. The same holds for data fusion, i.e., applying resolution functions to data conflicts. Both of these issues are difficult research challenges. For instance, the sorted neighborhood algorithm requires to compare tuples in a sorted representation of *all* data in a networked PIMS, which contradicts the peer-to-peer paradigm.

Personal information management is an application, whose users usually accept inexact query answers. As it is usually the case in keyword search, the query result may include values that are slightly beyond a selection interval or correspond to concepts that are closely related to the concepts mentioned in the query. Therefore, current PDMS concepts have to be extended by techniques for approximate query answering, e.g., [8]. Again, this is especially difficult to achieve for distributed query answering.

3 Connecting personal data at schema and data level

A recent article on "dataspaces" observes a shift of focus of the integration community towards best ef-

fort integration as opposed to strictly correct and complete approaches [7]. In this section we point out enabling technologies and research areas that make the development of efficient and effective networked PIMS a promising opportunity. We omit the technologies for individual PIMS, which we reviewed in Section 1, and the technologies for connecting peers, which we reviewed in Section 2. Here we focus on the aspect of automatic or semi-automatic integration for networked PIMS and distinguish two broad areas: schema-level integration and data-level integration. Additionally we present different approaches how privacy may be protected when personal data is connected through a network.

3.1 Schema-level integration

Supporting automatic or semi-automatic integration at schema-level is mostly concerned with the discovery of schema mappings, through a process called schema matching [16]. By analyzing schema elements and data values stored therein, algorithms automatically suggest which elements of one schema correspond with which elements of the other schema.

Further aspects are the discovery of references within data sources, i.e., within a PIMS (e.g., [11]). While the former is not necessary if a PIMS already uses a well-defined schema, the latter is particularly interesting, as it allows easy specification of meaningful queries over multiple PIMS. Finally, recent advancements in the field of model-management are also helpful to integrate and manage multiple schemata in a networked PIMS [12].

3.2 Data-level integration

At data-level the most important task of a standalone PIMS is to identify different representations of same objects (i.e., duplicates). In the context of SEMEX this process is described in [6]; iMeMeX on the other hand performs no semantic integration, so recognizing multiple occurrences of, say, the same person is left to the user. Conceptually, the problem of duplicate within a PIMS is the same as among networked PIMS. However, due to autonomy and resulting access restrictions, duplicate detection cannot be performed offline for all objects stored in all peers. Rather, duplicate detection must be performed ad hoc as already mentioned in the previous section.

Ad hoc duplicate detection differs in that the set of objects typically is small, namely a query result and not an entire relation, and in that the result must be obtained quickly, i.e., within an acceptable query response time. Thus, a good solution must perform very fast comparisons but might be able to afford comparing all pairs of result objects. These requirements are in direct opposition to the assumptions of traditional duplicate detection, which is performed offline and on large data sets, so that not all pairs can be compared, but the comparison method (the similarity measure) can be quite complex.

Finally, after duplicates are discovered within a query result, the data stored in duplicates must be *fused* to present users with a concise and consistent result set. While there are some systems that enable data fusion ([14, 1]), fully automatic fusion, as would desirable in an ad-hoc setting is yet an open issue.

3.3 Privacy

Networked PIMS enable users to pose complex, structured queries against the private data of other persons. For this reason aspects of privacy must be seriously considered.

In networked PIMS, a user achieves privacy by defining the set of data that is accessible through the network. Access to the remaining data on one's desktop is prohibited. The public data set certainly contains individual files, such as presentations, publications or data sheets from private and shared folders. Additional sources could include blog entries or personal pages from social networks. In this approach the granularity of data access is at a coarse level.

Many applications, such as calendars, email clients or address books, are designed to allow categorization and classification of entries. This opens the possibility of a finer grained access control. With the assistance of PIMS, users are able to share only personal information that belongs to categories they have made accessible to the public. Possible categories might be: conferences, restaurants, or health.

An even finer grained access control is achieved by explicit tagging personal information, that is to be shared. A simple PIMS tagging tool could offer the opportunity to make currently used personal data available to the network with only a few mouseclicks. This approach is motivated by the fact that personal information is generally seen and modified as emails, contacts, dates, or documents.

A final approach for authorization is the use of (preformulated) rules, i.e., "make available all emails sent to a certain mailing list".

Fine grained access control leads to the necessity of implementing an access control management as an additional functionality of PIMS: Personal information is distributed over different applications, files, and folders at one's desktop and a user should be able to easily reverse given access grants within a PIMS.

Networked PIMS should also be able to grant role-based access to personal data, i.e., to give colleagues access to a larger set of data compared to external users or students. In this process a user may use role management of the underlying (company) network. Even the topology of networked PIMS may be taken into account to permit access only for peers at close range.

4 Applications

Given a set of efficient and effective PIMS, interconnected with PDMS technology, there are several types of queries and applications that can be offered to users. The main difference to traditional PIMS is the ability to query across multiple people's desktops and the resulting need to logically and visually identify the original source of a query result.

Users can access the PIMS data using any of the following query types.

Browsing. Starting with a set of classes (people, publications, venues, ...), users point and click to open a list of other related objects, grouped by class. This type of interaction is intuitive but not powerful enough to explore the full capabilities of a PIMS.

Search queries. Keyword search is also an easy to understand interface. Search results are ranked according to their relevance with respect to the keyword, but also according to the origin (own desktop or remote desktop) and their class.

Structured queries. With a set oriented query language, users are able to ask more complex request and obtain results structured and order to their needs. A result not easily obtained by merely browsing and searching is for instance *Who in my workgroup has an author of the paper "Fast Algorithms for Mining Association Rules" in their contact list?* or *Find an email-address of any of the authors of the paper "Fast Algorithms for Mining Association Rules".*

Canned queries. To allow complex queries are noted above but to avoid mastering a new query language and schema, pre-formulated (canned) queries can be developed to cover the set of relevant and interesting queries for a particular user or domain. Canned queries can take one or more parameters, which are inserted in the structured query appropriately. An interesting query for networked PIMS is for instance *Who in my workgroup is available for 'x' minutes at time 'yyyy-mm-dd'?*

The types of queries mentioned above can be embedded in different types of applications.

1. Enhancements of existing PIM application. In the proposed networked PIMS we assume existing PIMS already installed for the individuals of a workgroup. A simple way to extend these systems across the data of the workgroup is to simply make available the additional data to the PIMS by materializing the appropriate remote indices. The fact that some of the indexed data is in fact data from remote desktops remains hidden.

2. Standalone networked PIM application. A standalone application is best able to feature all the abilities of networked PIM. Query interfaces can range from simple browsing all the way to structured queries. It replaces the user interface of the existing PIMS to offer a richer interface, which is explicitly aware of the networked character of the underlying data. Data from other users can be visualized as such and for instance filtered.

3. Plugins for other applications. Plugins, that query networked PIMS, offer users related information within existing applications. Examples include email applications that popup a display of workgroup-colleagues that have also been in contact with the recipient of the currently edited email.

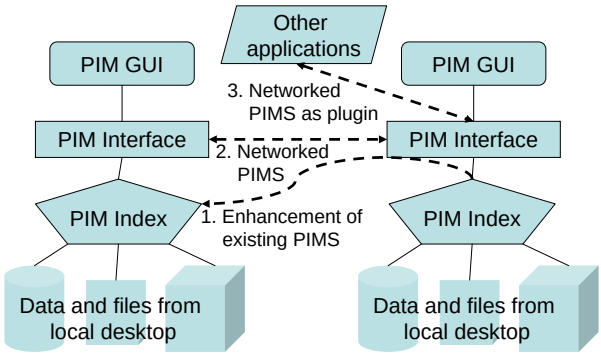

Figure 2: Three architectural alternatives

5 Outlook

In most organizations techniques for sharing personal information are often reduced to shared folders, email messaging, and passing along USB memory sticks. Networked personal information management systems (PIMS) offer complex querying against a large pool of permanently up-to-date information sources. This leads to a number of advantages, especially if groups closely collaborate on many projects and share many contacts.

In this paper we propose and illustrate an approach for setting up a network of PIMS using a peer data management system (PDMS). We described integration of PIMS at schema and data level and suggested different kinds of applications. The introduced kind of system provides ad-hoc integration

and querying capabilities of personal and workgroup data in a dynamic network environment.

We identified several challenges for future research regarding this novel marriage of technologies. Performing data cleansing in a distributed, ad-hoc and efficient manner is an open problem. The same holds for data fusion. Privacy of information is another issue not covered sufficiently in PDMS. In particular, it restricts query rewriting and distributed search. Also the unconsidered aspect of propagation of local updates at a peer leads to further interesting PIMS applications.

On the other hand, several of the underlying technologies are already well established, so that we are confident that networked PIMS as a crossover between networking and database research are both feasible and useful.

References

[1] Jens Bleiholder and Felix Naumann. Declarative data fusion – syntax, semantics, and implementation. In *Advances in Databases and Information Systems (ADBIS)*, Tallin, Estonia, 2005.

[2] Lukas Blunschi, Jens-Peter Dittrich, Olivier Ren Girard, Shant Kirakos Karakashian, and Marcos Antonio Vaz Salles. A Dataspace Odyssey: The iMeMex Personal Dataspace Management System. In *Proceedings of the Conference on Innovative Data Systems Research (CIDR)*, 2007. Demo.

[3] Yuhan Cai, Xin Dong, Alon Y. Halevy, Jing Liu, and Jayant Madhavan. Personal information management with SEMEX. In *Proceedings of the ACM International Conference on Management of Data (SIGMOD)*, 2005. (demo).

[4] Diego Calvanese, Giuseppe De Giacomo, Maurizio Lenzerini, and Riccardo Rosati. Logical foundations of peer-to-peer data integration. In *Proceedings of the Symposium on Principles of Database Systems (PODS)*, 2004.

[5] Jens-Peter Dittrich. iMeMex: A platform for personal dataspace management. In *SIGIR PIM*, 2006.

[6] Xin Dong, Alon Halevy, and Jayant Madhavan. Reference reconciliation in complex information spaces. In *Proceedings of the ACM International Conference on Management of Data (SIGMOD)*, pages 85–96, 2005.

[7] Michael J. Franklin, Alon Y. Halevy, and David Maier. From databases to dataspaces: a new abstraction for information management. *SIGMOD Record*, 34(4):27–33, 2005.

[8] Minos N. Garofalakis and Phillip B. Gibbon. Approximate query processing: Taming the terabytes. In *Proceedings of the International Conference on Very Large Databases (VLDB)*, 2001.

[9] Vijay Gopalakrishnan, Bobby Bhattacharjee, and Peter Keleher. Distributing Google. In *NetDB*, 2006.

[10] Alon Y. Halevy, Zachary Ives, Dan Suciu, and Igor Tatarinov. Schema mediation in peer data management systems. In *Proceedings of the International Conference on Data Engineering (ICDE)*, 2003.

[11] Ulf Leser and Felix Naumann. (Almost) hands-off information integration for the life sciences. In *Proceedings of the Conference on Innovative Data Systems Research (CIDR)*, pages 131–143, 2005.

[12] Sergey Melnik. *Generic Model Management: Concepts and Algorithms*. LNCS 2967. Springer Verlag, Berlin – Heidelberg – New York, 2004.

[13] Alvaro E. Monge and Charles P. Elkan. An efficient domain-independent algorithm for detecting approximately duplicate database records. In *SIGMOD Workshop on Research Issues on Data Mining and Knowledge Discovery (DMKD)*, pages 23–29, Tuscon, AZ, 1997.

[14] Amihai Motro and Philipp Anokhin. Fusionplex: resolution of data inconsistencies in the integration of heterogeneous information sources. *Information Fusion*, 2005.

[15] Lucian Popa, Yannis Velegrakis, Renée J. Miller, Mauricio A. Hernández, and Ronald Fagin. Translating web data. In *Proceedings of the International Conference on Very Large Databases (VLDB)*, Hong Kong, 2002.

[16] Erhard Rahm and Philip A. Bernstein. A survey of approaches to automatic schema matching. *VLDB Journal*, 10(4):334–350, 2001.

[17] Armin Roth and Felix Naumann. Benefit and cost of query answering in PDMS. In *Proceedings of the International Workshop on Databases, Information Systems and Peer-to-Peer Computing (DBISP2P)*, 2005.