

Research on Personal Dataspace Management

Yukun Li , Xiaofeng Meng
School of Information, Renmin University of China, Beijing, China
{liyukun,xfmeng}@ruc.edu.cn

ABSTRACT

Explosion of the amount of digital information has made Personal Information Management(PIM) become a hot topic. Personal data is always distributed, rough-and-tumble, personalized, heterogenous and evolutionary, which brings much challenge to to effective and efficient Personal Dataspace Management (PDSM). In the paper, by highlighting the importance of users in Personal Dataspace Management System(PDSMS), we proposed a user-centered framework. We first show research issues, related work, main research problems and challenges in this area. We then introduce the current research work and the preliminary results. Finally, the research plan of my PhD project is presented for discussion.

1. INTRODUCTION

Explosion of the amount of digital information made Personal Information Management (PIM) become a hot research topic. [1] presents the concepts, related disciplines, research issues and challenges of PIM. Compared with PIM, Personal Dataspace Management (PDSM) pays specific attention to management of personal digital information. A great number of new data are created on web every year, most of which are not structured and exist in various data styles, such as email, image, html, xml, audio, video, and so on. People can easily share them through Internet. So the amount of personal data is increasing actively. On the contrary, the time and capability of people for managing information are stable and limited, so how to improve the efficiency of PDSMS becomes an important problem.

Here is a motivation example. Mike has been working as a project manager of a company for many years. He has collected a great deal of personal data, including emails, images, web pages, and documents he has developed with various tools. The heterogenous data items are distributed in various devices, such as desktop, laptop, cell phone and so on. He sometimes meets the following troubles: Firstly, important data items are mixed with the data items out

of date together, which makes it hard for him to efficiently locate a specific data item; Secondly, the synchronization for his personal data items is highly desirable because of his frequent business trips. Every time before Mike trips for business, he has to backup some specific data items to keep the synchronization of his dataspace. Although he tried his best to do it, sometimes he still failed and fell into troubles.

To cope with the troubles of Mike, three problems must be studied: Firstly, how to efficiently integrate data items from various data sources. Secondly, how to efficiently organize the data items. Finally, how to efficiently operate the data items of PDS, such as query, update and backup. My PhD thesis focuses on building a personal dataspace management system and addressing several challenging issues in this area. By experiences of most people and our preliminary experiments, we observe that people play an important role in PDSM, It may become a key factor for improving performance of PDSMS. So I plan to take identifying user access pattern as the first step of my PhD project, and focus on studying user-centered algorithm to improve the efficiency of data integration, data query, data index, and so forth. In this paper, we will introduce our ideas, naive solutions and achieved results in details.

In session 2, a framework for PDSMS is presented. In Session 3, related work in this area is presented, and the open problems and challenges are discussed. In session 4, the research work we have done and the results we have achieved so far are introduced. In session 5, a prototype system for PDSMS developed by us will be demonstrated in detail.

2. OVERVIEW

In this section, we propose a framework for personal dataspace management, as shown in Figure 1. In this framework, we mainly focus on three aspects of personal dataspace management: data model, data integration and data output. Different from other works, user profile and data evolution are given specific consideration in this framework.

Data model is the center part of personal dataspace management. It includes logical model, physical model, evolution model and security model.

- Logical Model: Personal Dataspace(PDS) has some new characters, such as heterogenous data, pay-as-you-go integration, from-data-to-schema, data coexistence, and so on. How to model these characters is a promising and challenging topic.
- Storage: Storage strategy depends on two factors. data

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

Proceedings of the Second SIGMOD PhD Workshop on Innovative Database Research (IDAR 2008), June 13, 2008, Vancouver, Canada.
Copyright 2008 ACM 978-1-60558-211-5 / 08/ 06 ...\$5.00.

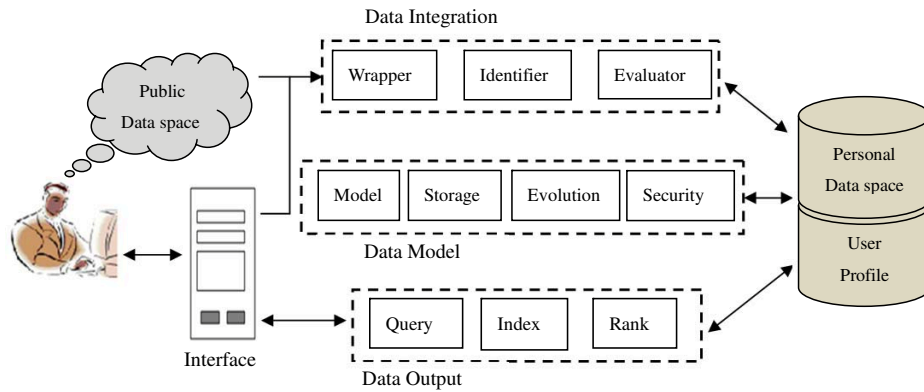


Figure 1: A framework for personal dataspace management

characters and operation characters. In personal dataspace, some data items are unstructured and distributed, and people need to access the data items anytime and anywhere. These factors may influence selection of storage strategy.

- **Evolution:** Evolution aims to improve data quality and operation efficiency by adaptive methods. For example, update operation may lead to optimization of query processing algorithm and index policy, and access behavior of user will result in the change of object importance. Evolution is always an automatic process, and there are few works on it in this field.
- **Security:** Security has always been one of the most important problems in data management systems. Heterogeneous and distributed storage make traditional approaches like disk mirroring and database backup inapplicable to dataspace systems, either costs too much or works inefficiently. These characteristics of Dataspace has made security a very challenging topic.

Personal data integration is the first step for personal dataspace management. According to the framework proposed, it includes three components:

- **Wrapper:** A wrapper is designed for a specific data source. The function of wrapper is to extract useful information from a specific data source. For example, a wrapper designed for email can identify the schema of email and extract the significant information related to the user. Wrapper selection and wrapper design are interesting topics.
- **Identifier:** Identifier is designed for checking the similarity of objects. When the user want to integrate an object into his dataspace, it must be checked by identifier firstly. If there has been a same object in PDS, the update operation will be executed, otherwise a new object will be inserted.
- **Evaluator:** Evaluator is designed for evaluating the correlation of a data object to the owner of PDS. The standard for evaluation is if it is useful or may be useful to the user. Just the useful data items are integrated.

Data Output: Query, index and rank are three main technologies to improve efficiency of data access.

- **Query:** Query in personal dataspace is to help the user to find specific data items in dataspace. There are mainly two kinds of query. Structured Query(SQ) and Keyword Query(KQ). Structured query performs well in RDBMS, keyword query fits the characters of web space, and is widely used in web search engine. But neither of them can dance well in personal dataspace. A new kind of querying paradigms is needed, which should combine structured and unstructured querying in a fundamental way.
- **Index:** Index is a well-known way to improve the efficiency of data query. Index selection depends on data model and storage strategy. Full-text index and invert list may play an important role in personal dataspace management.
- **Rank:** Rank method in PDS is affected by many factors, such as the user interest, the context of user behavior and so on. How to formulate these factors, and how to take advantage of them to improve efficiency of rank, are still problems.

Except for the three parts listed above, there are additional two components outlined in the framework: user profile and interface:

- **User profile:** Different from traditional DBMS, PDS is user-centered. This character will bring great influence to data integration, data model and data query. So we plan to take it as the focus of my research works.
- **Interface:** Interface design is also an important problem, which has close relation to the area of Computer Human Interface (CHI).

In fact, each component of the framework can be considered as a research issue itself, and it is not practical for me to study so many topics in my PhD issues. In my PhD project, I plan to focus on the following topics: a flexible and user-centered data model, an efficient and adaptive data integration strategy, and user-centered query and index algorithm. Meanwhile we will incrementally develop a prototype system to demonstrate my ideas and solutions.

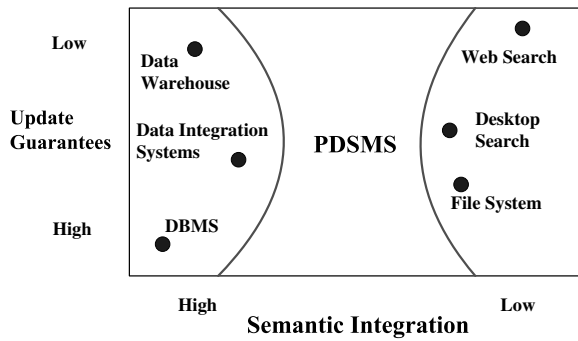


Figure 2: A space of data management solution

3. RELATED WORK

As shown in Figure 1, data model, data integration and data query are three primary parts of PDSMS. In my PhD project, I plan to focus on the three issues. The related work on these issues will be introduced. And the main research problems and challenges will be presented and discussed.

3.1 Data Model

There are many models for DBMS. Each of them is designed for a specific scenario, and has its own advantages and disadvantages.

Figure 2 [2, 3] illustrates the popular models for data management. For example, RDBMS is a schema-first data management technology, which can provide high update guarantees and high semantic integration. On the opposite, Desktop Search is a no-schema data management system, it just support simple keyword query. Neither of them can efficiently support personal dataspace management. The middle area of the picture presents a new data management technology, called Dataspace Management System (DSMS), [4] presents the principles of Dataspace Systems.

Dataspace incrementally becomes a data model for PDSMS. iMemex Data Model(iDM) [5] describes versatile data items with an universal form. Another contribution of iDM is that the model enables to represent the structural information available inside files. Resource Space Model (RSM) [6] is a model for specifying, sharing and managing versatile web resources with a universal resource view. A normal resource space is a semantic coordinate system with independent coordinates and mutual-orthogonal axes. Desktop Search Engine (DSE) is a no-schema style of data management. It does not provide any update guarantees and does not allow structural information to be exploited for queries. And it pays no consideration to data security, backup, semantic query, and so on.

Although there has been some works for modeling personal dataspace, because of the distinct characters of PDS, data model is still a basic and challenging research problem. The first problem is that it lacks the ability to express the complicated functions and requirements. When specific functions of PDSMS are considered, such as security, privacy, consistency, and so forth, the iDM model need to be extended to match the special requirements. Also it did not pay enough attention to the flexibility of the schema for from-data-to-schema data integration. The challenge of this field is from two factors of PDS: user centered and data evolution. How to model and formulate the two features is of

great challenge. In addition, PDS is generally regarded as a large graph, The versatile relations among objects will lead to a lot of join operations and low efficiency.

3.2 Personal Data Integration

Data integration aims to extract data items and store them with a suitable strategy, so that they can be efficiently accessed by users. Web data integration (WDI) and personal data integration (PDI) are two main fields in the area. WDI is domain-oriented, but PDI is user-oriented, which makes them distinct from each other. Some approaches and results on WDI can be introduced into PDI field, such as object identification, schema mapping, and so on. Here we mainly present related works on PDI. Semex [7] is proposed as a platform for personal information management and integration. [7] presents a model of uncertainty integration, [8] proposes an pay-as-you-go integration strategy: creating iTrails between objects to improve data quality. The LifeStreams Project [9] organizes documents in chronological order and allows the user to view the documents from different viewpoints in terms of time.

Although there have been many works on PDI, there still exist a lot of challenging problems unsolved. Data quality and reusing human attention are main challenges of personal data integration.

3.3 Query and Index

Traditional query technologies could be categorized into keyword query [10] and structured query. Distribute storage, disparate data source and lack of schema distinguish the query processing in dataspace from that of other data manage systems. Keyword query provides poor performance due to lack of semantic information. Structured query doesn't apply to the schema-later framework of PDS. So the two methods should be combined to support more flexible and complicated queries. How to make a tradeoff between them is still a problem. Index is an efficient way to improve query performance, [11] attacks the problem of indexing in dataspace. By summarizing the characters of personal dataspace, the query in the area should match the following characters: easy and flexible interface, best-effort and association-based query. In PDS, people tend to find some objects by cluttered clues remembered. Maybe it is a date, or a conference name, etc. Association-based query will lead to a lot of join operations and result in a low efficiency.

There are still some problems waiting for solutions. The first is query performance. The unstructured query makes query optimization more difficult. The full-text index style makes index maintaining a challenge. The second is query interface. There have been many query languages, Such as SQL, XQuery, iQL, and so on. Strict and complex grammar is their common character. It is not easy for average user of PDS to accept these languages. Keyword is very easy, but it can not efficiently support semantic query.

There are also some prototype systems on PDSMS, such as iMemex [12, 2], Semex [13], Haystack [14] and so on. iMemex frees the data contained in a dataspace from its formats and devices, by representing it using a logical graph model. Semex and Haystack extend data warehouse and information integration technology, and allow users to browse by objects' association. They are based on a high-level mediated global schema over the personal information sources. Desktop search engine (DSE), e.g. Google desktop search

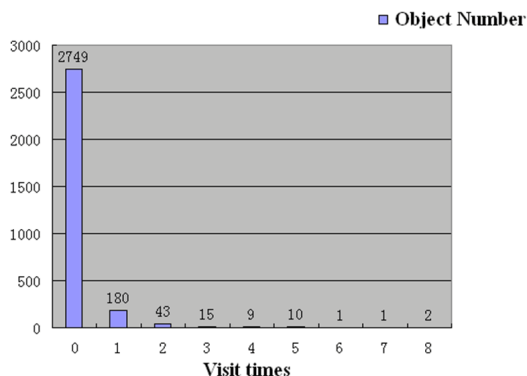


Figure 3: An experiment on personal data access

[15], is another attempt for PIM, and there are some approaches [16] presented by researchers to improve the efficiency of DSE. But DSE does not allow structural information to be exploited for queries [2]. As to interface design and user interest patten mining [17], there are some approaches presented in CHI area and IR area.

4. RESEARCH WORK

In my PhD project, I plan to focus on a few key problems in PDSMS. To make our work based on a strong foundation, I'm taking data integration as the first step. On one hand, data integration provides a large size of data pool for our comprehensive experiments. On the other hand, we can possibly find some rules on user access behavior by analyzing the large dataset integrated. This section will introduce our main work on data integration, data model and data query in PDSMS.

4.1 A User-centered and Flexible Model

Different from the existed models of PDSM, we focus on two characters: user-centered and from-data-to-schema. So we guess that the characters of personal access behavior may be the key factors to improve efficiency of data operation in PDS. So we create a user-centered data model for PDS, it includes two main concepts: CoreSpace framework and vertical data model.

4.1.1 CoreSpace Framework

According to the experiences of people, some objects are visited by the user in a high frequency in a certain period, meanwhile the other objects are visited in a relatively low frequency. It means that the importance of objects in a PDS are not equal. To test it, we did an experiment based on a real PDS of one author of the paper. In the PDS, there are 3100 objects kept by the owner. The size of it is 4.5G. We take the visit log in continuous twenty days as the dataset for our experiment. The Figure 3 shows the result of the experiment.

In Figure 3, we can see the number of objects that are visited in the twenty days is 261, just 8 percent of the total number of the objects in the PDS. The number of the objects that are visited more than 3 times is 38, which is just about 1 percent of the total number. According to the Pareto Principle and our primary experiments, we find a phenomena: in a PDS, different objects often show different importance to the owner; the importance of object is a

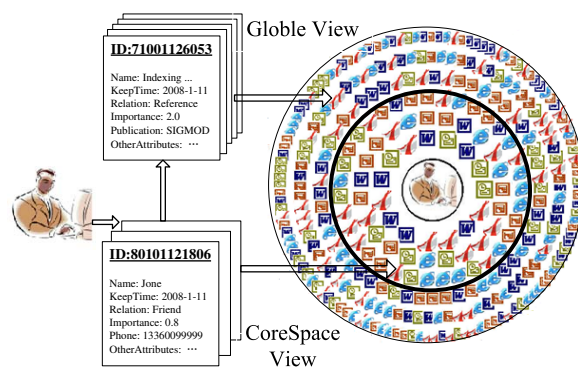


Figure 4: CoreSpace architecture

dynamic value; people tend to access the objects that are visited frequently in a recent period.

Based on the observations, we introduce two concepts: Object Weight and CoreSpace to model the correlation between the owner and objects. Object Weight represents the importance and correlation of an object to the owner. Its value is dynamic and evolves over time. CoreSpace is a sub-space of PDS which is composed of important objects to the owner whose Object Weight is larger than a predefined threshold during a period. Figure 4 illustrates the architecture of CoreSpace. When a person submits a query, the CoreSpace is scanned firstly, if the result is not satisfying, the full space will be scanned.

This concept presents a new view for improving efficiency of data operation, its efficiency still needs a large number of experiments to test, also it results in some challenging problems, for example, how to efficiently compute the Object Weight, and how to define the boundary of CoreSpace.

4.1.2 Vertical Data Model

From-data-to-schema and pay-as-you-go are two main characters of PDS, To fit the characters, we try to propose taking the vertical data model [18, 19] to map the characters of PDS. Although vertical data model is not a novel idea, there is still no work on introducing it to the area of PDS. In this model, it takes a 3-ary vector to describe each attribute of an object, and the 3-ary vector is defined as (ObjectID, AttributeName, AttributeValue). According to the model, each object is presented as a set of the 3-ary vectors. This model brings advantages to PDSMS:

- From-data-to-schema integration: Based on the vertical data model, we can easily implement from-data-to-schema integration. When integrating an object which can not match the schema, we can finish it easily.
- Pay-as-you-go integration: Before integration, a person doesn't need to create precise schema for it in advance. With increasing of PDS size and changing of the content, the schema will be increasingly summarized from the data.
- Easy interface: Because there is no complex schema, and the structure is universal and simple, it is easy for people to implement keyword-based query strategy.

So the model fits PDS characters and can make data integration and data query easier. Also there are some dis-

advantages for the vertical model. Efficiency may be the first question argued by persons. The complex graph may lead to a lot of join operation, which will result in lower efficiency. To attack the problems, we are planning to take the following measures: Simplify the graph structure and take advantage of memory resources. I'm also planning to design and develop a specific system to apply the vertical data model. Different from the traditional complex DBMS, it will be lightweight and easy for people to manage his or her PDS.

4.2 Efficient Data Integration Strategy

The process of personal data integration has been introduced in session 2. Efficiency and data quality are two main standards for evaluating integration strategies. I implemented a prototype system for PDSM: OrientSpace, which supports desktop data integration. Different from other works, the features of user access behaviors are paid special consideration to in OrientSpace system. By utilizing the pattern of user access, data integration will become more automatic and efficient. It is still an open challenge to effectively mine the pattern, which is attacked by many works in Information Retrieval(IR) area.

Wrapper is an important part of data integration. By various wrappers, more data can be integrated into PDS automatically. Now we have designed two wrappers in OrientSpace system: one is for formal papers of pdf type, and the other is for email data source. Also an initial data set has been constructed by OrientSpace system. In PDS, data integration always is automatically done. So a monitor is needed to check the changes of data items in desktop. It will take high cost of computer. We just implement a naive solution by full disk scanning. Next step, we will focus on finding efficient way to improve the efficiency of the monitoring algorithm. Object Weight is a new concept we proposed. By experiments, we proved the reasonability of the concept Object Weight, and it is a main parameter effecting operation efficiency. How to formulate it and compute it is still a problem. It depends on many factors, such as data type, created time, modified time, data size, and so on. As a naive solution, I am proposing a pattern for computing Object Weight.

By vertical data organization, we really implement from-data-to-schema integration. Instead of traditional DBMS, the schema is not predefined here, and it is summarized from the data integrated before. When keeping an encountering object, the user need not pay specific attention to its schema.

4.3 User-centered Query Strategy

To improve the efficiency of query in PDS, we paid specific attention to the following problems and have achieved some results.

Query Framework: CoreSpace is a new framework we proposed for PDSMS. Based on the framework, the user can efficiently perform a query by scanning the CoreSpace, instead of scanning the whole dataspace. Only when the user is not satisfied with the result, the full space is scanned. It is able to improve query efficiency. But there are still many challenging problems, such as how to define the boundary of CoreSpace, how to define and formulate the satisfaction of user, and so on.

Query Interface: Based on the two-level CoreSpace architecture and the characters of personal query, OrientSpace

system provides an universal search service by combining keyword search and structured query. The query interface can be the combination of several keywords as well as the following form: $\{\{attribute\} \setminus \{keyword\}^*, K\}$. By parameter K , user can specify the size of result set. If no parameter is given, it means all objects which match the keywords will be returned. For processing a query, the system first find result from the CoreSpace. When the user is not satisfied with the current results or the number of result is less than K specified by the user, the whole dataspace will be searched. The results of query are ranked according to the Object Weight. The user can find more objects related to the results by further navigation.

Index strategy: Index is an efficient way to improve query performance. Full text index always leads to a large size of index file, and results in low efficiency in maintaining index and performing query operation. According to the CoreSpace framework proposed, I plan to take a two-level index strategy, the first level is index of CoreSpace, and the second level is index of the full space. Different policies are taken in maintaining the two kinds of index. The effectiveness of this solution needs to be tested with experiments.

Rank algorithm: In PDSM, ranking mainly depends on the two factors: user character and operation context. Some rank approaches in web search can be introduced to attack the problem of PDSM. But the differences between web search and PDS query should be paid specific attention to. For example, the objects of query result of PDS may be of different styles, it may include emails, images, pdf documents, and so on. How to organize and rank them is an interesting topic.

By attacking the problems listed, we plan to build an user-centered strategy for query in PDS, which is based on CoreSpace framework and simple interface, and will have a better performance than the current desktop search engines.

5. A PROTOTYPE SYSTEM

OrientSpace is a prototype system developed by us for personal data integration and management. Based on CoreSpace framework and vertical data model, OrientSpace implements two functions: data integration and data query. By it we can initially integrate most data items in personal desktop computer. Different from DSE, it can provide a comprehensive solution for PDSM, It is not only support desktop query, but also support data integration, data update, data backup, and so on. Figure 5 is the main interface of OrientSpace system. It includes three parts: The left-top portion is schemas of PDS, which are summarized from data and are not completely synchronous with data; the left-bottom portion is a resource manager just like the resource manager of Windows operation system; the right part is for data operation, where user can integrate data, execute a query, navigate query result, explorer CoreSpace and analyze the data in PDS.

Data integration: According to the number of objects integrated one time. It is categorized into two classes: single integration and batch integration. By single integration, just one object is integrated into PDS one time, and it is often used to integrated scattered encountering data items. On the other hand, by batch integration, a lot of objects can be integrated one time, and it is mostly used to build initial dataspace for a person, or to keep multiple objects in specific conditions. Both the two methods are implemented

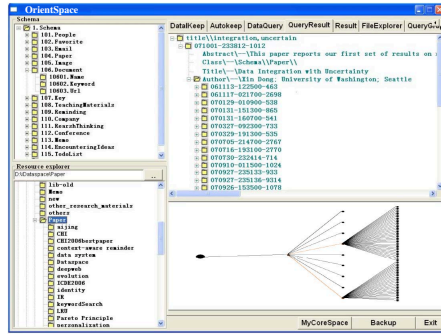


Figure 5: Main interface of OrientSpace system

in OrientSpace system.

According to the process of integration, it also can be categorized into two classes: automatic mode and manual mode. Automatic mode is based on wrapper, and it is specially used to integrate the data items from specific data sources with fixed structure or stable format. Such as email, pdf paper, etc. Manual mode is designed for the encountering of scatter data items, such as a person, an idea, and so forth. In OrientSpace system, we applied a wrapper for pdf paper and a wrapper for email. when user encounters an interesting pdf paper and want to keep it, he can easily do it by dragging it into dataspace. Meanwhile the useful data items of it will be automatically extracted, formulated, translated into unified data structure and stored.

Data query: OrientSpace provides two methods for user to find what he or she wants: CoreSpace Explorer(CSE) and Extended Keyword Query(EKQ). By CSE method, the user can easily explorer the objects in his or her CoreSpace by the specific order. For example, the objects can be ordered by Object Weight, visited date, size, subject, and so on. The user also can filter the result by selecting an attribute value of a data item. Different from file system, EKQ breaks the boundary of folders and data sources, which makes data query easier.

Figure 5 shows the result window of the example query "Title\\integration, uncertain", which means that "Please find me the objects whose title contain the words *Integration* and *uncertain*". In addition, by the further navigation, the user can see a chain of objects connected by associations. For example, the result in this example is a paper about integration and uncertainty, and we can further find the other papers written by the same authors. The red lines show navigated routes.

Distinct from other systems for PDSMS, OrientSpace implements wrapper-based personal data integration and make a try in research on query interface and query method. With the development of our research work, more and more results will be integrated into OrientSpace system.

6. CONCLUSIONS

This paper presents the sketch of my research plan on PhD project, Firstly, a user-centered framework for personal dataspace management is proposed. It includes three main parts: data integration, data model and data output. After summarizing the research status in this area, we introduce the issues we are focusing on and we will address in the future. In conclusion, the main works of my PhD project

are to build a user-centered personal dataspace management system and to address several key issues in this area.

7. ACKNOWLEDGMENTS

This research was supported by the National High-Tech Research and Development Plan of China under Grant No. 2007AA01Z155; Program for New Century Excellent Talents in University (NCET); China National Basic Research and Development Program's Semantic Grid Project (No. 2003CB317000).

8. REFERENCES

- [1] Jones W and Bruce H. A Report on the NSF-Sponsored Workshop on Personal Information Management, Seattle, WA, 2005.
- [2] Blunski L, Dittrich J-P, Girard OR, Karakashian S.K and Salles MAV. A Dataspace Odyssey: The iMeMex Personal Dataspace Management System. CIDR 2007: 114-119
- [3] Franklin M, Halevy A, and Maier D. From databases to dataspace: A New Abstraction for Information Management. SIGMOD Record, 34(4):27-33, 2005.
- [4] Halevy A , Franklin M. Maier D. Principles of dataspace systems, PODS 2006: 1-9.
- [5] Dittrich J-P and Salles MAV. iDM: A Unified and Versatile Data Model for Personal Dataspace Management. VLDB 2006: 367-378
- [6] Zhuge H. Resource space model, its design method and applications. The Journal of Systems and Software 72 (2004) 71-81
- [7] Dong X, Halevey A. Data Integration with Uncertainty. VLDB 2007 687-698 :
- [8] Salles MAV, Dittrich J-P, Karakashian S.K, Girard OR, Blunski L. iTrails: Pay-as-you-go Information Integration in Dataspace, VLDB 2007: 663-674
- [9] Freeman E and Gelernter D. Lifestreams: A Storage Model for Personal Data. In SIGMOD Record 25(1):80-86, 1996.
- [10] Hristidis V, Gravano L, Papakonstantinou Y. Efficient IR-style keyword search over relational databases. VLDB 2003: 850-861.
- [11] Dong X, Halevey A. Indexing Dataspace. SIGMOD 2007: 43-54
- [12] Dittrich J-P iMeMex: A Platform for Personal Dataspace Management. In SIGIR PIM Workshop, 2006.
- [13] Dong X and Halevy A. A Platform for Personal Information Management and Integration. CIDR 2005:119-130.
- [14] Karger DR. Haystack: A Customizable General-Purpose Information Management Tool for End Users of Semistructured Data. CIDR 2005: 13-26
- [15] <http://desktop.google.com>
- [16] Chirita P-A, Nejdl W : Analyzing User Behavior to Rank Desktop Items. SPIRE 2006: 86-97
- [17] Qiu F, Cho J. Automatic Identification of User Interest For Personalized Search. WWW 2006: 727-736
- [18] Copeland G. P and Khoshafian S. A decomposition storage model. SIGMOD 1985: 268-279
- [19] Agrawal. R, Somani. A, Xu. Y. Storage and Querying of E-Commerce Data. VLDB 2001: 149-158