Towards Realization of Dataspaces

Ibrahim Elsayed and Peter Brezany
Institute of Scientific Computing, University of Vienna
Nordbergstrasse 15/C/3, A-1090 Vienna, Austria
E-mail: {brezany,elsayed}@par.univie.ac.at

A Min Tjoa
Institute of Software Technology and Interactive Systems
Vienna University of Technology, Favoritenstrasse 9-11/E188, A-1040 Vienna, Austria
E-mail: tjoa@ifs.tuwien.ac.at

Abstract

Database management systems providing powerfull data management services have been developed in the recent decades to a great extent. All these systems are based on its own single data model, whereas the demand for managing multiple data sources with different data models is rapidly expanding. Therefore the need for intelligent management systems providing access to those heterogeneous and often distributed data sources and allowing to search and query them as a single information source, has never been greater. This reseach challenge is faced by the vision of Dataspaces. So far, no effort was devoted to realization of Dataspace concepts on the Grid. In this paper, we propose the architecture of a Dataspace Management System and discuss how some current components of the Grid technology can support the future implementations of such an architecture.

1 Introduction and Motivation

The term Dataspace is differently considered by different research communities. In the data mining and knowledge discovery community, a Dataspace is described as an infrastructure for remote data analysis and distributed data mining, where a Dataspace is an example of a data web - that is, a web-based infrastructure for working with data [15]. The utility of such a Dataspace is to reduce the time required to accomplish the data extraction, cleaning, and transforming step as well as the exploratory data analysis step of a data mining task. In the paper, Grossman and Mazucco describe a special data transport protocol for accessing data stored physically as files in distributed Dataspace servers on a data web. The data within their Dataspace is a distributed collection of columns. Another usage of the term Dataspace is given by Imielinski and Goel in [20], where a Dataspace is defined as a three dimensional physical space enhanced with connectivity to the network. Here, a Dataspace is populated by classes of mobile objects producing and storing their own data. These physical objects can be queried and monitored on the basis of their properties. Spatial coordinates are the basic points of reference to navigate and query the Dataspace. The idea is to provide digital information embedded in physical space for wide range applications such as efficient transportation, environmental protection, and rapid emergency response.

In December 2005, Halevy et al. [14] introduced a new abstraction for information management by describing a platform supporting Dataspaces, where a Dataspace contains a set of *participants* and a set of *relationships*. A participant can be any data element. The idea is to raise the abstraction level at which data is managed in order to provide a system managing different data sources, each with its own data model. The concepts are presented in a visionary way, however, their implementation in real application environments opens new research challenges.

In this paper, we deal with the architecture and functionality of a *Dataspace Management System*, which we understand as *a set of software programs that controls the organization, storage and retrieval of data in a Dataspace. It also handles the security and integrity of the Dataspace.*

The paper is organized as follows. Section 2 provides an analysis of requirements, the fulfillment of which can bring Dataspaces to real-world applications. Section 3 deals with the realization of a Dataspace Management System. In Subsection 3.1, we propose its architecture. Subsection 3.2 discusses the idea of Sub-Dataspaces and Subsection 3.3 outlines several dataspace management workflows. An application scenario is given in Section 4. Finally, in Subsection

Proceedings of the 17th International Conference on Database and Expert Systems Applications (DEXA'06)
0-7695-2641-1/06 $20.00

3.4, we show how existing Grid tools can support the implementation of Dataspace Management Systems. In Section 5 we provide a brief conclusion and outline our future plans.

2 Requirements

We discuss the requirements of applications on a Dataspace by the means of the Dataspace Environmet components depicted in Figure 1.

Information Retrieval - Querying and searching are two different information retrieval methods and represent one of the main services supported by a Dataspace Management System. In general querying and searching methods should be supported by all Dataspace participants independent of their data models as if applied to a single data base. A well known and simple search process is the keyword search. Most users are familiar with this search method as it is the main instrument for retrieving information from the world wide web. The support of such a search method on all participants in a Dataspace regardless which data model it contains is a challenging research topic. The development of keyword based search methods for relational and XML databases is faced by the data engineering community [1, 16, 19]. Supporting a global query functionality allowing to formulate queries on top of all participants in a Dataspace, needs intelligent methods for interpreting and translating queries into various languages. Query translation methods present a main research challenge, faced by a large body of research communities [9, 22].

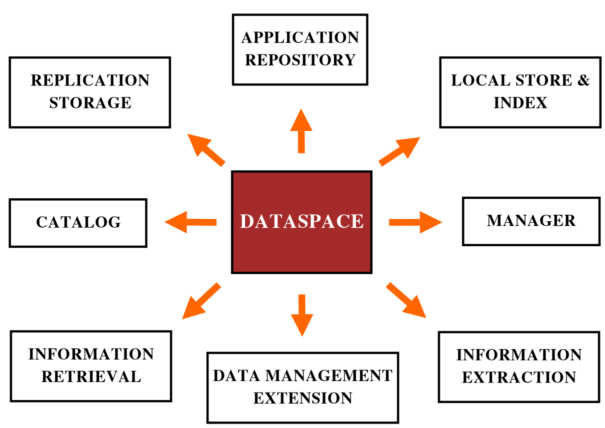

Figure 1. The Dataspace Environment

Information Extraction - Most frequently used data containers are relational databases, object-oriented systems, XML databases, and files. However, the world wide web has become a huge data container and thus a storehouse of knowledge. In order to allow postprocessing of data obtained from the web, web information extraction techniques, extracting relevant information from semi-structured web pages should be supported. Extracted contents have to be transformed into structural information and saved locally for further processing. Non-structured web documents need to be classified first using text mining techniques, which map the parsed documents into groups organized with the help of ontologies [18]. Based on those ontologies a keyword search is possible to retrieve such documents. The third and last kind of documents that can be found on the web are structured documents, which allow easier access and integration due to the rich semantically information included in the data representation.

Data Management Extension - The Data Management Extension component offers features for enhancing low level Dataspace components. A Dataspace component for example could be a simple fileserver or set of web documents. As these types of data elements have no or only limited data management functions it is a task of the Dataspace Management System to provide additional data management features such as backup, recovery, and replication.

Catalog - It contains a detailed descriptions of all participants included into the Dataspace. Besides the basic information about the participant, such as owner, creation date, etc., the description should also include semantic information about the data of the participant. The user should be able to browse the Catalog to get more information about specific data sources. The Catalog can refer to a metadata repository in order to separate basic information from more detailed data descriptions.

Manager - In order to support the above mentioned features, the system needs to be managed by a central component interacting with the user. Besides user authentication, right assignment, and other services, the Manager is responsible for communication with the participants thus serving as an interface between the users and the participants of the Dataspace.

Local Store and Index - This component manages the cache containing search and query results so that certain queries can be answered without accessing the actual data source and supports the creation of queryable association between participants.

Replication Storage - It allows to replicate the participant data in order to incease access performance, thus high availability and recovery is supported.

Application Repository - Here, the users can share data analysis tools, domain specific models, evaluations, etc. that can be applied to the data available in the Dataspace.

3 Dataspace Management System

3.1 Architecture

In general, a Dataspace consists of Dataspace components, the so called participants and relations between these

Proceedings of the 17th International Conference on Database and Expert Systems Applications (DEXA'06)
0-7695-2641-1/06 $20.00 © 2006 IEEE

components. Dataspace components are individual datasources, such as relational databases, text databases, XML databases, data stream systems, web services, or any other data repositories.

A Dataspace component contains a description describing what kind of data it contains, which format is used for storage and which querying mechanisms are allowed. It contains information about data location (where the data is stored) and if the data is replicated somewhere as well as about relations to other participants. The information and data configured corresponding to the participant is marked whether it is retrieved from the underlying component management system, added by the user or automatically generated by the system. Such information is important for reporting functions and could be usefull for possible later system optimizations.

Each participant is described in the above discussed manner and registered within the Catalog of the Dataspace Management System, thus they are available as data sources for a higher abstraction layer providing global queries and search features. This higher abstraction layer is covered by the Dataspace Management System. It assists the user to add new Dataspace components, to create relations with other participants, to register participants within the Catalog, to browse the Catalog for available participants, to share all or only selected participants within a community including assignment of access rights, and to search and query all or only selected participants of a Dataspace.

As each participant within a Dataspace has its own management system and thus is responsible for its data management such as data update, recovery, and replica respectively, it is not the duty of the Dataspace Management System to change the data represented by the participants. In other words the system has no administration rights for writing or changing content of data of Dataspace participants. But the Dataspace Management System offers the possibility to add probably missing data management features, such as data replication, transformation, download, upload, etc. Figure 2 shows the system architecture of the Dataspace Management System with three participants having different data models: a relational database, a XML repository, and a file repository. The UK core e-Science program's Open Grid Services Architecture - Database Access and Integration (OGSA-DAI) project [24] developed wrappers for relational and XML repositories and since recently also for files so that they offer Grid-enabled services conforming to the OGSA framework. OGSA-DAI is used to access those different data containers. More information about OGSA-DAI is provided in Section 3.4. Each of these participants are registered within the Catalog of the system. The Catalog stores basic information such as identifier, type, owner, creation date and so forth, of all participants and the relationships within the Dataspace. Furthermore it

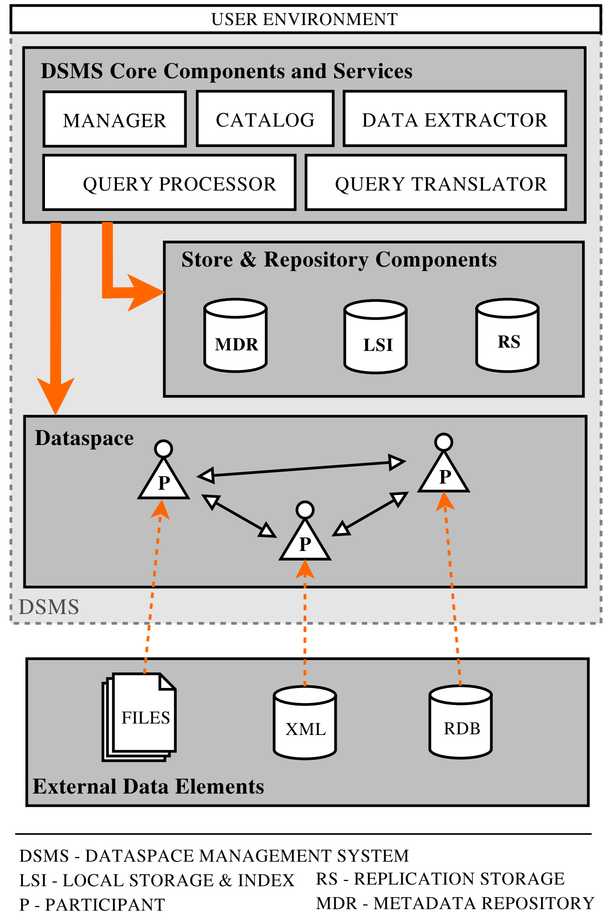

Figure 2. The System Architecture

contains information about supported query languages and data structures. It acts like a central information component that can be browsed by the user to retrieve information about a data source. Metadata corresponding the participants will be saved within the Metadata Repository in order to separate basic information from more detailed data descriptions. GridFTP [2], a high-performance, secure data transfer protocol will be supported by all internal data containers, namely the Metadata Repository, the Replication Storage, and the Local Storage and Index thus allowing the user to transfer data between these components and local machines whereas OGSA-DAI is used to access these data containers within the Dataspace Management System. Details on GridFTP are discussed in Section 3.4.

As on-line databases, such as those offered by the NCBI (National Center for Biotechnology Information) [1]

[1] Established in 1988 as a national resource for molecular biology information, NCBI creates public databases, conducts research in computational biology, develops software tools for analyzing genome data, and disseminates biomedical information - all for the better understanding of molecular processes affecting human health and disease.

and many other scientific databases increases in quantity and also in usage by scientists, it is of strong interest to provide access to such data sources within a Dataspace. This can be promoted by access to the background information hidden in the web. This requires the extraction of relevant data from structured and semi-structured query result web documents, which is faced by the web information extraction research [6, 11, 23]. The extracted information will be saved and indexed locally. Storing and indexing relevant data locally improves the access to specific data sources and enables that common queries can be answered without accessing the actual data source. Moreover it supports high availability and recovery of the data within the Dataspace components. The Manager generates relationships between the components and helps the user to refine these relationships. When a Dataspace component is created, the Manager registers it within the Catalog. User authentication and assignment of access rights are also handled by the Manager. The Query Processor faces the problem of global query on multiple databases at multiple sites. A Query Interpreter translates the queries into languages supported by the participants. Figure 3 illustrates a simple usage scenario, where the data within participants of the Dataspace are used as input source data for an application within the Application Repository.

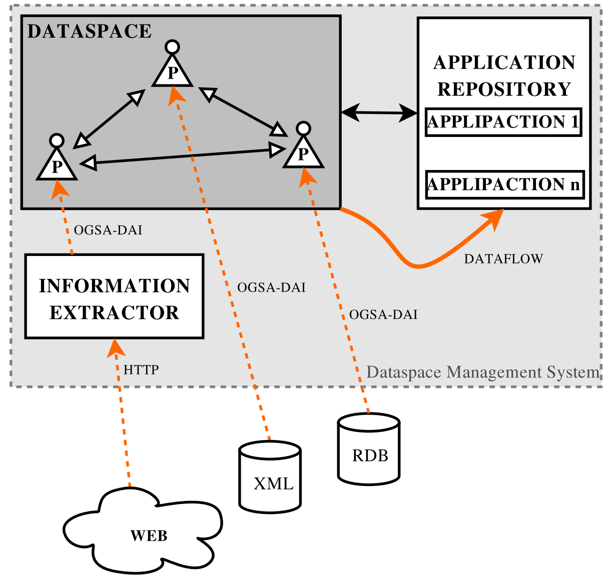

Figure 3. A Usage Scenario

3.2 Managing Sub-Dataspaces

The idea is to create a new Dataspace i as a subset of Dataspace j, where i, the Sub-Dataspace consits of a subset of participants and a subset of relationships of the origin Dataspace j. This functionality is needed in connection with access rights and security management - a Dataspace administrator can define user groups with different data access rights. At the top level, users can see all participants within the Dataspace, while other user groups with lower level access rights see only some participants involved in the targetted Sub-Dataspace.

3.3 Dataspace Management Workflows

In this Section most relevant workflows for managing Dataspaces are described. First of all, a Dataspace is defined and labeled. The next step is to add participants, by registering the new data sources. Therefore the user enters a reference to the participant. Then data descriptions can optionally be entered, either by hand or by entering a reference to a description file if available. Containing these information the participant will be registered within the Catalog and the Metadata Repository. All these steps, including the definition of relationships between two or more participants can be integrated into the processing step "Add Participant" within a workflow. Search and Query, Information Extraction and Data Replication are other possible workflow components that can be defined within a Dataspace Management Workflow. Depending on domain specific applications, more particular workflows can be defined.

3.4 Grid Technology Support for Dataspaces

Grid computing has been identified as an important new technology by a remarkable thread of scientific and engineering fields as well as by many commercial and industrial enterprises [7]. Its goal is to share and manage geographically distributed computer resources and data across enterprises, industry or workgroups independently of the operating characteristics of their computer systems. It can be used to temporarily increase computational power and storage needs on demand. So far, essentially all major Grid projects have been built on protocols and services of the Globus Toolkit [13], which is an open source software toolkit used for building Grids, developed by the Globus Alliance [3].

Many advanced Grid applications are data intensive, that is, significant processing is done on very large amounts of heterogeneous and geographically distributed data. Therefore, the management of data within Grids is a challenging research and development problem. Almost all Grid based applications are file-based, and until recently, there has been relatively little effort applied to integrating databases into the Grid [24]. In the following existing Grid tools concerning data management on the Grid are described. Further, the ways how these tools can support the management of Dataspaces is discussed.

GridFTP [2] is a high-performance, secure protocol supporting parallel, striped, partial and reliable data transfer, based upon the standard FTP protocol; so it can interoperate with conventional FTP clients and servers. It provides third-party control of transfers between storage servers and uses basic Grid security on both control (command) and data channels. The current GridFTP protocol specification is now a "proposed recommendation" document in the Global Grid Forum[2] and therefore the main candidate for a standard data transfer protocol on the grid. Once a standard is defined, existing storage and database systems will have to adopt support for it, as it will be advantageous for both users and providers. GridFTP could be used for downloading data for further processing on local machines. Users may need to use local tools, such as statistical and reporting tools, and domain specific applications for their research field. As scientists often deal with data and compute intensive applications, the data within a Dataspace could be transferred to a compute grid using GridFTP. Another scenario is to upload data into the Dataspace using GridFTP allowing users to share the data source which was created or modified. In this use case, the user creates a new participant, which stores and manages the data within the storage component of the Dataspace Management System. He defines the storage capacity and sets its properties, such as owner, access rights, etc. and uploads the data using GridFTP. As mentioned already, GridFTP will be supported by all internal data containers thus allowing the user to transfer data between these components and local machines.

Replica Management [2], a component within the Globus Toolkit [13] responsible for managing the replication of complete partial copies of datasets, defined as collections of files. The service allows users to create new copies of a complete or partial collection of files, to register these new copies in a replica catalog, and to query the catalog to find all existing copies of a particular file or collection of files. The data model assumed by the Replica Management architecture is based on files. A replica is a subset of a collection of files that is stored on a particular physical storage system. Combined with GridFTP, Replica Management can be used for the replication service within the Dataspace Management System. As already mentioned above the system can support participants with low data management features by providing additional functions, for example data replication.

The **Metadata Catalog Service (MCS)** [26], developed by the Globus Alliance provides a mechanism for storing and accessing descriptive metadata and allows users to query for data items based on desired attributes. MCS is implemented as a stand-alone metadata catalog service with an OGSA[3] service interface, which associates application-specific descriptions with data files, tables, or objects. These descriptions, which are encoded in structured ways defined by "schema" or community standards, make it easier for users and applications to locate data relevant to specific problems. MCS is integrated with OGSA-DAI, which provides metadata storage and basic Grid authentication mechanisms. When MCS is used with OGSA-DAI, Grid authentication is used to authorize accesses and manipulation of the metadata. MCS can be used in conjunction with the Resource Location Service (RLS) [10] component, which maps logical filenames to associated physical filenames in the underlying storage systems.

Storage Resource Broker (SRB) [25] is a middleware aiming to provide a unified view of the data files stored in different storage systems and locations by organizing the data into virtual collections independent of their physical location and organization. These collections own and manage all of the information required for describing the data independent of the underlying storage system, thus a transparent data access and transfer are granted. Therefore SRB is a potential candidate for handling the data of participants, which are dealing with files. But if there is a need to deal with multiple and different data containers within a dataspace, the SRB middleware alone is insufficient.

The variety of different data containers and corresponding data models available, makes it pretty complicated to support unique access to these different data sources. Following there are many incompatible data access protocols and client libraries leading to the need of multiple access methods for accessing the data within participants of a Dataspace. **OGSA-DAI** [24, 21] is a project to provide a component library for accessing and manipulating data in a Grid for use by the Grid community as it is integrated into the Globus Toolkit 4.0.1. OGSA-DAI can be seen as the standard for data access on the Grid thus it is used within the Dataspace Management System to access the data within a participant. Right now, Grid Database Services (GDSs) provided by the OGSA-DAI project support access to relational and XML data bases and since recently also to files.

Although its is not mentioned in [5] that OGSA-DAI will be extended for other common database and storage systems in the near future, we can be sure that the Grid community will continue to produce components for working with and managing data on the Grid, as the demand for Grid based data storage and management systems has never been greater.

OGSA-DQP [4], OGSA-Distributed Query Processor is a collection of OGSA compatible services supporting the compilation and evaluation of queries that combine data obtained from multiple data sources on the Grid, including the GDSs provided by the OGSA-DAI project. It handles

[2]The Global Grid Forum (GGF) is the community of users, developers, and vendors leading the global standardization effort for grid computing.

[3]http://www.globus.org/ogsa/

the construction and execution of distributed query plans in a declarative language and can be used in any Grid application that aims to integrate and analyze structured data collections. Within the Dataspace Management System, OGSA-DQP will be used to perform queries on all or only selected participants that can be accessed via the OGSA-GDSs. For dealing with data elements, which are not supported by OGSA, wrappers are needed to interact with the OGSA-DQP services.

The set of other possible candidates includes the Grid-based systems like GridDB [12], DataCutter [8], Mobius [17], etc.

4 Application Scenario

The following application scenario gives an overview on how the Dataspace Management System could be used for ecological data analysis.

The ecological research community is recently trying to integrate several ecological data collected and managed by 27 organizations spread across Europe. The goal is to share data on biodiversity, ecology and socio-economy in order to extend their understandings and co-operations [27]. Every owner of ecological data has his own data container with its own structure, semantic and metadata description. Most of the data containers are relational databases while some are object-oriented systems or XML databases. Some data are in EXCEL spreadsheets or in information systems like statistics packages or geographic information systems. The data sources include data from different interest domains; each is represented as a participant within the Eco-Dataspace. The participants can be dynamically included into the Dataspace or withdrawn from it (e.g. for privacy or security reasons). Within the global Dataspace, participants will be dynamically integrated into different problem-oriented virtual sub-Dataspaces performing different data exploration processes, for example: geostatistics, flow analysis, building prediction models. Detailed descriptions on all participants and also available sub-Dataspaces (which are in fact sets of participants and the relationships among them) can be browsed via the Catalog.

Within the Application Repository the eco-community will share domain specific applications, as there are many specialized models and evaluations distributed among multiple institutions and thus not easily accessible. Such applications are often expensive and difficult to develop and to install, as well as being quite resource consuming. Results of these specialized models and evaluation methods will be dynamically integrated into the Dataspace with a *result-of* relationship to the participants representing their input data. The relation itself keeps information about the model used to get the corresponding result set. Such a relationship enables the system to apply intelligent queries such as "How this participant is created and based on which model?" A special participant will be the location participant, which holds *located-in* relationships to other participants in order to provide a geographic search functionality. So the system will be able to answer queries like "Give me all terrestrial data for Lower Austria!".

Another relation that will be used within the Eco-Dataspace is the *related-to* relation, which describes that the participants connected by the relation represent the same data but collected at different times. Models will be applied on both, the participants containing the actual data and the ones containing the old data. The results will then be compared together in order to achieve a good view on ecological effects in the corresponding area.

5 Conclusions and Future Work

The vision of Dataspaces is a new research topic to be faced by the data management community. This paper responses to this challenging task by proposing the architecture and functionality of a management infrastructure for Dataspaces. We assume that Dataspaces will play a very important role in virtualization of Grid data resources. Therefore, we also analyze a possible Grid support, which can be considered in our plans to implement a first prototype of the proposed architecture on the Grid. We also plan to investigate how our data mining infrastructure GridMiner (www.gridminer.org) could efficiently operate on Dataspaces.

References

[1] S. Agrawal, S. Chaudhuri, and G. Das. Dbxplorer: A system for keyword-based search over relational databases. Proceedings of the International Conference on Data Engeneering, 2002.

[2] B. Allcock, J. Bester, J. Bresnahan, A. L. Chervenak, I. Foster, C. Kesselman, S. Meder, V. Nefedova, D. Quesnal, and S. Tuecke. Data management and transfer in high performance computational grid environments. Parallel Computing Journal, Vol. 28 (5), pp. 749-771, May 2002.

[3] Globus Alliance. http://www.globus.org.

[4] M. N. Alpdemir, A. Mukherjee, N. W. Paton, P. Watson, A. A. A. Fernandes, A. Gounaris, and J. Smith. Ogsa-dqp: A service-based distributed query processor for the grid. Proceedings of the UK e-Science All Hands Meeting, 2003.

[5] M. Antonioletti, M. P. Atkinson, R. Baxter, A. Borley, N. P. Chue Hong, B.Collins, J. Davies, D. Fitzgerald, N. Hardman, A.C. Hume, M. Jackson, A. Krause,

S. Laws, N. W. Paton, T. Sugden, P. Watson, M. Westhead, and D. Vyvyan. Status report and future directions. Proceedings of the UK e-Science All Hands Meeting, September 2004.

[6] R. Baumgartner, S. Flesca, and G. Gottlob. Visual web information extraction with lixto. Proceedings of the Very Large Data Base Conference, September 2001.

[7] F. Berman, A. J. G. Hey, and G. Fox. Grid computing: Making the global infrastructure a reality. John Wiley & Sons, 2003.

[8] M. Beynon, R. Ferreira, T. Kurc, A. Sussman, and J. Saltz. Datacutter: Middleware for filtering very large scientific datasets on archival storage systems. IEEE Symposium on Mass Storage Systems, 2000.

[9] M. Carey, J. Kiernan, J. Shanmugasundaram, E. Shekita, and S. Subramanian. Xperanto: A middleware for publishing object-relational data as xml documents. Proceedings of the Very Large Data Base Conference, September 2000.

[10] A. Chervenak, E. Deelman, I. Foster, L. Guy, W. Hoschek, A. Iamnitchi, C. Kesselman, P. Kunst, M. Ripeanu, Schwartzkopf B, Stockinger H, K. Stockinger, and B. Tierney. Giggle: A framework for constructing sclable replica location services. Proceedings of Supercomputing, November 2002.

[11] C. Chia-Hui, H. Chun-Nan, and L. Shao-Chen. Automatic information extraction from semi-structured web pages by pattern discovery. Decision Support Systems, 35(1):129-147, Special Issue on Web Retrieval and Mining, April 2003.

[12] M. J. Franklin D. T. Liu. Griddb: A data-centric overlay for scientific grids. Technical Report UCB//CSD-04-1311, 2004.

[13] I. Foster. Globus toolkit version 4: Software for service-oriented systems. IFIP International Conference on Network and Parallel Computing, Springer-Verlag LNCS 3779, pp 2-13, 2005.

[14] M. Franklin, A. Halevy, and D. Maier. From databases to dataspaces: A new abstraction for information management. ACM SIGMOD, December 2005.

[15] R. Grossman and M. Mazzucco. Dataspace: A data web for the exploratory analysis and mining of data. IEEE Computing in Science and Engineering, Vol. 4, No. 5, July 2002.

[16] L. Guo, F. Shao, C. Botev, and J. Shanmugasundaram. Xrank: Ranked keyword search over xml documents. ACM SIGMOD, June 2003.

[17] S. Hastings, S. Langella, S. Oster, and J. Saltz. Distributed data management and integration: The mobius project. GGF Semantic Grid Workshop, 2004.

[18] Q. He, L. Qiu, G. Zhao, and S. Wang. Text categorization based on domain ontology. Lecture Notes in Computer Science, Springer, 2004.

[19] V. Hristidis and Y. Papakonstantinou. Discover: Keyword search in relational databases. Proceedings of the Very Large Data Base Conference, 2002.

[20] T. Imielinski and S. Goel. Dataspace - querying and monitoring deeply networked collections in physical space. IEEE Personal Communications Magazine. Special Issue on Networking the Physical World, October 2000.

[21] K. Karasavvas, M. Antonioletti, M. P. Atkinson, N. P. Chue Hong, T. Sugden, A. C. Hume, M. Jackson, A. Krause, and C. Palansuriya. Introduction to ogsa-dai services. Lecture Notes in Computer Science, Volume 3458, Pages 1-12, 2005 May.

[22] R. Krishnamurthy, V. T. Chakaravarthy, R. Kaushik, and J. F. Naughton. Recursive xml schemes, recursive xml queries, and relational storage: Xml-to-sql query translation. Proceedings of International Conference on Data Engineering, 2004.

[23] N. K. Papadakis, D. Skoutas, K. Raftopoulos, and T. A. Varvarigou. Stavies: A system for information extraction from unknown web data sources through automatic web wrapper generation using clustering techniques. IEEE Transactions on Knowledge and Data Engineering, Vol. 17, No. 12, Decemberl 2005.

[24] N. Paton, M. Atkinson, V. Dialani, D. Pearson, T. Storey, and P. Watson. *Database Access and Integration Services on the Grid*, Feb 2002.

[25] A. Rajasekar, M. Wan, R. W. Moore, W. Schroeder, G. Kremenek, A. Jagatheesan, C. Cowart, B. Zhu, S. Y. Chen, and R. Olschanowsky. Storage resource broker - managing distributed data in a grid. Computer Society of India Jornal, special issue on SAN, October 2003.

[26] G. Singh, S. Bharathi, A. Chervenak, E. Deelman, C. Kesselman, M. Manohar, S. Patil, and L. Pearlman. A metadata catalog service for data intensive applications. Proceedings of Supercomputing, 2003.

[27] A. Woehrer, M. Bloech, P. Brezany, G. Gruber, A. Langegger, H: Schentz, and W. Woess. Towards semantic data integration for advanced data analysis of grid data repositories. Proceedings of the First Austrian Grid Symposium, December 2005.