Grid Enabled Data Integration Framework for Bioinformatics Research*

Jia LIU, Yongwei WU, Weimin Zheng

Department of Computer Science and Technology; Tsinghua National Laboratory for Information Science and Technology，Tsinghua University Beijing 100084, China
liu-jia04@mails.tsinghua.edu.cn, wuyw@tsinghua.edu.cn, zwm-dcs@tsinghua.edu.cn

Abstract

A framework is proposed to manage the distributed and heterogeneous databases in grid environment for understanding protein-protein interaction. Furthermore, the framework is used into bioinformatics division of Shanghai Research and Development Public Service Platform. In this paper, the need of data integration and the related works are discussed first, followed by a brief analysis of the requirement of grid-enabled data integration framework. In the third part, the grid enabled integration framework is put forward in detail. At last, one case for bioinformatics research based on the framework is given out to show the benefits of the framework for bioinformatics research.

Key words: ***Protein-Protein Interaction, OGSA-DAI, Activity, Metadata Catalog***

1. Introduction

Exponential data growth appears in all the fields of science, business and engineering today, data exist in all shapes and sizes, store as public resources as well as private databases, from relational and XML databases to excel spreadsheets and flat files. Combining information from these multiple, diverse and distributed data resources provides huge potential to make use of the wealth of data, promotes intra-enterprise and inter-enterprise collaboration.

Considerable effort and remarkable progress have been made in data integration system in database and AI communities, including the early multi-database systems (e.g., Multibase [11]) , mediator systems (e.g., Garlic [12]) and ontology-based integration approaches (e.g., OBSERVER [13]), projects like EnsEMBL Database project [14]. TAMBIS [15] and BioMediator [16] consider data integration in biomedical domain.

In this paper, we are going to discuss data integration in grid environment for bioinformatics research. Grid technology is an emerging technology for seamless and loose integration of diverse resources distributed on the Internet, it must provides the capability to manage data resources across distributed, heterogeneous environments globally. One of the main goals of grid is to encourage the publication of scientific data in a more open manner, for greatest advances would be made by combining data from separate, distributed sources to produce new results. Bioinformatics is application and development of computing and mathematics to the management, analysis and understanding of data to solve biological questions [16], Data integration systems are widely researched in bioinformatics applications for reasons: 1) a huge amount of biological databases and bioinformatics systems are accessible by internet, covering topics as RNA sequences, protein sequences, genomics and proteomics; 2) bioinformatics need to dealt information in different types, granularities and levels of the biological structure and heterogeneous semantics. To utilize the community resources and assemble all available information to do research on bioinformatics, building grid environment to share resources like professional processing software and various instrumentations, using grid middleware to locate, analyze and integrate information from multiple, heterogeneous data sources is feasible and affordable solution.

We propose the grid-enabled data integration framework that based on OGSA-DAI middleware, then the heterogeneous databases can be accessed by the uniform interface transparently, data integration could implement well interaction with the other services such as blast processing and enable users to provide new

** This Work is supported by ChinaGrid project of Ministry of Education of China, Natural Science Foundation of China (60373004, 60373005, 90412006, 90412011, 60573110, 90612016), and National Key Basic Research Project of China (2004CB318000, 2003CB316907)

Proceedings of the Fifth International Conference on
Grid and Cooperative Computing Workshops (GCCW'06)
0-7695-2695-0/06 $20.00 © 2006 IEEE

functionality easily. Distributed data management components like metadata manager, transaction manager are developed on the basis of extended OGSA-DAI activities, considering the benefit of existing basic service OGSA-DAI provided and the convenience of further development it would bring. It is easy to extend the architecture including new functionalities to take into account a wider range of issues related to grid data management. The case study of protein-protein interaction is introduced as an example to demonstrate the contribution of the framework (proteins carry out the majority of biological "function", understanding the behavior of proteins are most important), it approved that the framework can assist with the management of bioinformatics infrastructures and processes.

The rest of this paper is organized as follows. In Section 2, we review the previous works and existing data integration approaches; then In Section 3 we analyze the design requirement of the grid-enabled data integration architecture, while presenting the architecture, we concentrate on the observation of requirement and functionality of every component; In Section 4, we use a case study of protein-protein interaction to demonstrate the benefit of our framework, in the end, we give a conclusion and future plan of our work.

2. Previous work

In this section, we consider the main data integration approaches and discuss some other grid data access and integration solutions. The current methods of data integration generally can be classified to two basic types [2]:

- Schema-equivalence oriented(e.g., AutoMed):

This method focuses on providing mechanisms for defining the equivalence between different database schemas, one general approach is using ER data model as a common data model and then mapping the other data models to ER data models.

- Query-processing oriented:

This approach constructs a global schema on the top of distributed databases. Therefore, it focuses on providing mechanisms of translating the global schema queries into source specific queries using mapping rules, combining the data sources return results to final integrated results for users. Query-processing oriented approaches are prevalent to resolving semantic heterogeneity which is one of the great challenges confronting the data integration. It can be broadly classified as traditional global schema approach, Object-Oriented global schema approach and Ontology-based global schema approach [1].

Spitfire [7] and OGSA-DQP (Distributed Query Processing) [6] are projects trying to manage databases in grid. Spitfire is using Web Service technology to provide SOAP-based RPC (through Apache Axis) to a few user-definable database operations like single-row or few-row lookups and inserts, but it disable to handle large result set and complex operations. OGSA-DQP is a project extends OGSA-DAI to provide distributed query using OQL as the query language rather than SQL, another limitation of DQP is that the data query service is a non-standard OGSA-DAI service, it not allowed DQP functionality to be accessed via standard OGSA-DAI Perform documents and services, DQP is not equipped to deal with the semantic heterogeneity is another defect.

ISPIDER project is aim at providing an environment for constructing and executing analyses over proteomic data, and a library of proteomics-aware components that can act as building blocks for such analyses [17], ISPIDER wraps sources with OGSA-DAI, AutoMed [18] wrappers are used to extract sources' metadata and accomplish schema mapping and integration.

3. Framework

3.1. Requirement

The following items are the basic requirements to design the data integration architecture:

- Uniform and Transparency: Uniform refers to the access mechanism will be independent of the actual implementation of the data source. Transparency refers that the users need not to know the technical-level heterogeneities of underlying data sources and the physical location of the databases.
- Multi-database federation: it can combine information from multiple database, providing distributed query for heterogeneous database, it allows user to aggregate information from multiple data sources to gain a more complete picture.
- Efficiency: grid application always have especially performance requirement, it includes high-throughput, fault tolerance, etc. A high-performance mechanism for query execution and a best way to transfer the query results to the computation is significant.
- Security: data security is a fundamental requirement, it often refers access control and data protection, generally, security is ensured by existing database server products and the Grid-wide security system where grid components

exist within a single unified security framework.

3.2. Architecture

The architecture is described in Figure 1.

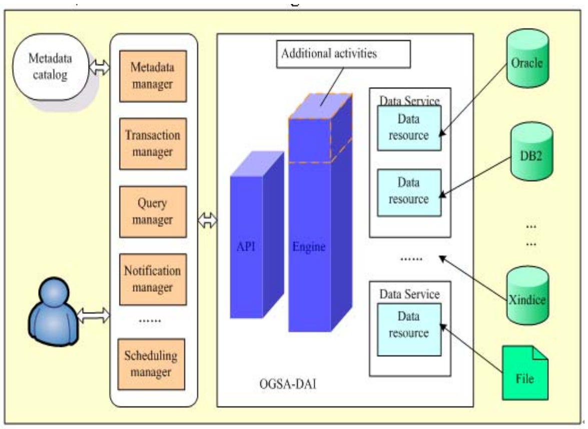


Fig. 1. Architecture

The detailed functions of the components are described in the following section.

3.3. Components

3.3.1. OGSA-DAI. Our system is constructed on the top of OGSA-DAI, OGSA-DAI [5, 3] is a project that develops middleware to assist with access and integration of data from separate sources via the grid. The project was funded by the UK e-science core programme and is working closely with the Global Grid Forum DAIS-WG 。 OGSA-DAI includes a collection of components for querying, transforming and delivering data in different ways, and a simple toolkit for developing client application, it focuses on data accession, rather than data integration.

We build our system on the top of OGSA-DAI, benefiting from the following reasons:

- The hiding of heterogeneity.
- The scalable framework, we can extend the functionalities by adding user-defined activities.
- Providing a uniform interface, accessing data resources in grid by the form of service.
- Including basic activities of statement (e.g. SQL query), transformation (e.g. compression) and transport (e.g. GridFTP).

Adopting OGSA-DAI as the foundation sharply reduced the complexity.

3.3.2. Additional activities. The additional activities we defined are foundation units for managing grid database, they are organized in catalogs, which can be described in figure 2, the rectangles on the left side of dashed line represent the basic activity catalogs that OGSA-DAI provided (e.g., statement), the rectangles on the right side represent the additional activity catalogs we developed (e.g., transaction), each catalog includes a series of activities around the catalog name. Practically, most of the products of DBMS support functions of metadata manipulation, transaction control etc, to them, the additional activities of metadata and transaction are the wrappers of these function templates of different data source.

- Metadata activities

Metadata of data resources includes information of name and location, context description and the structure of the data held within the resources, it is vital important for discover and automatically interpret data of autonomously managed databases.

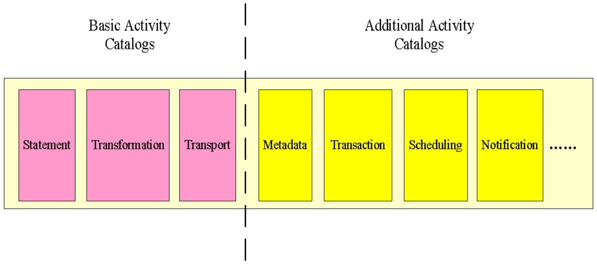


Fig. 2. OGSA-DAI activities

As the summary in section 2, using metadata to construct integrated schema is popular for data integration, metadata manager adopting this approach to revolve semantic heterogeneous, metadata activities provide build blocks for metadata manager, the functions of metadata activities include: 1), get metadata of the data resource; 2) create new metadata of the data resource according to metadata specification of the framework.

- Transaction activities

In grid environment, there are local transactions and global transaction need to be considered. Transaction activities primary need to wrap the operations include begin, commit, rollback on the local data resource.

- Scheduling activities

The scheduling activities could develop to support abundant functions, basic scheduling tasks could implement pre-allocate resource such as bandwidth of network to a request, provide sufficient resources as disks, memory to a particular job, advanced scheduling tasks could support data replication, data fragment, etc.

- Notification activities

Notification activities should implement the functions of specializing what data (data set) it is interested in and notifying the users the data changes in data resource.

3.3.3. Managers.

- Metadata manager

The metadata manager is a core component of the system, it is resposible for collecting and managing the data resources metadata, constructing integrated data schema and conducting schema mapping.

First, it could collect and manage the metadata of the data reosurce by the functionality offered by metadata activity.

Second, two schemes are included to construct integrated data schema. one scheme is constructing global schema on the top of all the data resources, we call it global scheme, global scheme is for users that hoping to use our framework as information integrated system (giving some keywords like "protein", querying for all useful information, then, the system returning the results in some specified order like relativity), which works as an extended search engine. The other scheme is part scheme, there is no global integrated schema but part integrated schema provided in this scheme, this is for users who would like to customize workflow (they know the structure and semantics of low-level data schemas of the data resources in the system, they define the part integrated schema for their specify motivation). The two different schemes satisfy different requirement. Global scheme need little workload for end users but enormous workload for administrator to construct global schema, importing automatic schema matching mechanism [8] is our next plan to reduce the overhead. Part scheme is much flexible, different users could design their own part integrated schema for their specified need, it is imposssible to design a global schema that could perfectly fit to a particlular user's information need and emphasize his individual domain of research.

Third, schema mapping rules are recorded and managed by metadata manager, it would conduct transformation between integrated schema query and respective data resources queries.

Metadata manager is the essence of constructing and managing data integration system, the assumption of data schema evolution is focus on this template.

- Query manager

Query for a single database could be accomplished by the basic activities of statement activities provided in OGSA-DAI. Distributed query processing need the coordinated work of metadata manager, query manager, transaction manager, OGSA-DAI data services and etc, query manager works like the coordinator during this process, the query processing is a two-step process, it firstly parses the query into a logical query plan by calling the metadata manager, and then execute the query plan. Different query algorithms could be develped in query manager, transaction manager is employed to ensure the ACID properties during the procedure.

- Transaction manager

In transaction manager, local transaction of the data resource can be implemented using transaction activities, global transaction implemented by coordinating the execution of various transactions on data resources. To adapt the dynamic grid, transaction mechanism with different strictness and granularity should implement, the transaction manager are indispensable for ensuring correctness of sharing information and cooperating work.

- Scheduling manager

Gird environment is dynamic, adaptively, dynamicly scheduling the grid resources (network bandwidth, storages, and etc) could improve performance, availability and efficiency. To make scheduling decision, scheduling managers should make interaction with grid monitor service to get static and dynamic information about computer nodes (e.g., CPU, memory, disk) and network (e.g., bandwidth and latency).

- Notification manager

This would allow users to register some interest in changes of a set of data, updates of integrated schema, switches of request state, it includes mechanisms for users to specify what it is interested in and a method for notifying the users for notifying the users the change.

3.3.4. Metadata catalog. Metadata catalog is a basic component in grid, It keeps the metadata and provides a mechanism for storing and accessing metadata. The metadta information related to this framework includes:

- Metadata of OGSA-DAI services and data resources.
- Integrated schema of the data resources in the system.

3.3.5. Data resource. Consists of a set of structure (e.g. Oracle, DB2), semi-structure and unstructured data resources.

4. A case study of bioinformatics research

The case of studying protein-protein interaction is described as an example integration scenario for this paper, the steps include:

1. Through BLAST search, find the homologous protein sequences;
2. Select an interested protein A from the result of step 1;
3. Query PIRPSD database for the basic information of the protein A;

4. Query PUBMED database for literature reference about protein A;
5. Find the proteins having interaction with protein A by querying DIPDB database, DIPDB is a database that contain the interactions information between proteins, choose one of them as protein B;
6. Query PIRPSD database for the basic information of the protein B;
7. Query PUBMED database for literature reference about protein B;

The basic information of the databases are described in table 1:

Table 1. database list

DB name	Type	version	Description
PIRPSD	Mysql	4.0	Database of Basic protein information
DIPDB	PostgreSQL	8.0	Database of protein interaction
PUBMED	Oracle	9.0	Literature database of life sciences

In the conventional method, the researchers need to query different databases separately and execute the sections of step 5-7 and step2-7 repetitiously, the enormous records in the database and reiterative steps made it an incredible job for both researchers and computational resource.

Using our proposed framework, it could integrate the protein-related resources easily, the process of the above steps can be accomplished using a single query, the workflow can be describe in figure 3:

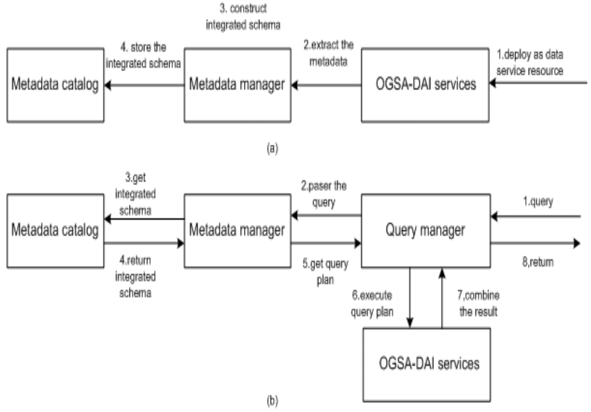
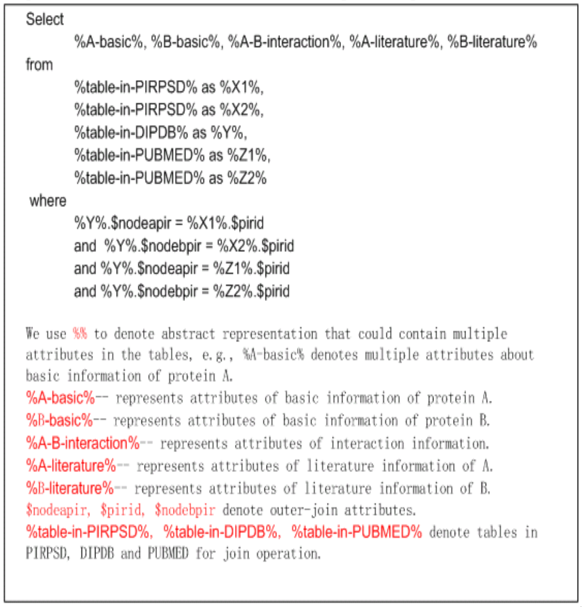


Fig. 3. (a) Deploying database, (b) Query processing

From Figure 3(a), we know that the deploying process firstly make the database be accessible from other machines as grid service, secondly update the integrated schema.

After deploying databases according to workflow of Figure 3(a), researches can query the heterogeneous databases PIRPSD, PUBMED and DIPDB as local database, join operation is needed as follows to accomplish all jobs:



```
Select
        %A-basic%, %B-basic%, %A-B-interaction%, %A-literature%, %B-literature%
from
        %table-in-PIRPSD% as %X1%,
        %table-in-PIRPSD% as %X2%,
        %table-in-DIPDB% as %Y%,
        %table-in-PUBMED% as %Z1%,
        %table-in-PUBMED% as %Z2%
 where
        %Y%.$nodeapir = %X1%.$pirid
        and  %Y%.$nodebpir = %X2%.$pirid
        and %Y%.$nodeapir = %Z1%.$pirid
        and %Y%.$nodebpir = %Z2%.$pirid

We use %% to denote abstract representation that could contain multiple
attributes in the tables, e.g., %A-basic% denotes multiple attributes about
basic information of protein A.
%A-basic%-- represents attributes of basic information of protein A.
%B-basic%-- represents attributes of basic information of protein B.
%A-B-interaction%-- represents attributes of interaction information.
%A-literature%-- represents attributes of literature information of A.
%B-literature%-- represents attributes of literature information of B.
$nodeapir, $pirid, $nodebpir denote outer-join attributes.
%table-in-PIRPSD%, %table-in-DIPDB%, %table-in-PUBMED% denote tables in
PIRPSD, DIPDB and PUBMED for join operation.
```

Fig. 4. SQL for distributed query

The query is submitted to query manager, metadata manager decompose the query to query plan according to the integrated schema stored in metadata catalog, query manager execute the query plan by coordinating the corresponding data services, transaction manager will be used in the process when concurrency occurs. The distributed query would take a long time to finish the outer join between tables in different database, then, employing notification manager to inform the end of query is a favorable manner.

In summary, the following advantages are obvious when employing our framework in this application:

- It provides well interoperability for sharing community resources, we can access the heterogeneous databases PIRPSD, PUBMED and DIPDB by standard interface of grid service.
- Workflow is simplified by distributed query processing. Considering the above case, the times of select statement needed in the application is depend on the total records in database if using the conventional method, but only one select statement is required in our framework, it improves the access and interactions with their data.
- Using metadata manager could eliminate the semantic heterogeneity between databases, for instance, the field of $nodeapir and $nodebpir in

DIPDB have similar semantic with the field $pirid in PIRPSD, the metadata manager resolve it by constructing integrated schema.

5. Conclusion

In this paper, we have presented an OGSA-DAI-based architecture, in which basic functions are implemented as extended activities of OGSA-DAI, it employs metadata manager to construct integrated schema and eliminate semantic heterogeneities, two integration schemes are proposed for different users and different motivations of using this framework, query manager is employed to accomplish heterogeneous distributed query processing, transaction manager is employed to ensure the ACID properties of databases, notification manager in the framework makes register interesting data, integrated schema and request state available, scheduling manager could be used to improve the system performance for dynamic grid environment. The case study demonstrated in the section 4 proved that the framework brought benefits for the application of protein-protein interactions research.

Future work will mainly focus on two areas: firstly, import the research fruit of automatic semantic matching to metadata manager, make the process of schema integration much intelligent; secondly, integrate other databases of protein structure and protein family to the framework, test much more bioinformatics applications under this framework.

6. References

[1] P. Ziegler, K.R Dittrich, “User-Specific Semantic Integration of Heterogeneous Data: The SIRUP Approach”, In First International IFIP Conference on Semantics of a Networked World (ICSNW 2004), volume 3226 of Lecture Notes in Computer Science, Springer, Paris, France, June 17-19, 2004, pp. 44-64.

[2] P.J. McBrien, A. Poulovassilis, “Automatic migration and wrapping of database applications - a schema transformation approach”, In Proceedings of ER99 Springer Verlag LNCS 1728, 96-113, 1999.

[3] M. Antonioletti, M.P. Atkinson, R. Baxter, A. Borley, N.P. Chue Hong, B. Collins, N. Hardman, A. Hume, A. Knox, M. Jackson, A. Krause, S. Laws, J. Magowan, N.W. Paton, D. Pearson, T. Sugden, P. Watson, and M. Westhead. “The Design and Implementation of Grid Database Services in OGSA-DAI”, Concurrency and Computation: Practice and Experience, February 2005, Volume 17, Issue 2-4, pp. 357-376.

[4] M. N. Alpdemir, A. Mukherjee, N.W. Paton, P.Watson, A. A. Fernandes, A. Gounaris, and J. Smith. “Service-based distributed querying on the grid”. In the Proceedings of the First International Conference on Service Oriented Computing, Springer, 15-18 December 2003, pp. 467-482.

[5] The OGSA-DAI Project, URL: [http://www.ogsadai.org.uk/].

[6] The OGSA-DQP Project, URL: [http://www.ogsadai.org.uk/about/ogsa-dqp/].

[7] The Spitfire Project, URL: [http://edg-wp2.web.cern.ch/edg-wp2/spitfire/index.html/].

[8] Rahm, E, and P. A. Bernstein, "A Survey of Approaches to Automatic Schema Matching", VLDB Journal 10, 4, Dec. 2001, pp. 334-350.

[9] L. Zamboulis, H. Fan, K. Belhajjame, J. Siepen, A. Jones, N. Martin, A. Poulovassilis, S. Hubbard, S. M. Embury, N. W. Paton, “Data Access and Integration in the ISPIDER Proteomics Grid”, In Proc Data Integration in the Life Sciences 2006, July 2006.

[10] The BioMediator Project, URL: [http://www.biomediator.org/].

[11] T. Landers and R. Rosenberg, "An Overview of Multibase ", in Distributed Databases, H.J. Schneider, Ed., North-Holland, The Netherlands, pp. 153-184.

[12] The Garlic Project, URL: [http://www.almaden.ibm.com/cs/garlic/].

[13] The OBSERVER Project, URL: [http://siul02.si.ehu.es/OBSERVER/].

[14] Y. Tohsato, T. Kosaka, S. Date, S. Shimojo and H. Matsuda, “Heterogeneous Database Federation Using Grid Technology for Drug Discovery Process”, Grid Computing in Life Science: First International Life Science Grid Workshop, LSGRID 2004, Kanazawa, Japan, May 31-June 1, 2004.

[15] R. Stevens, P. Baker, S. Bechhofer, G. Ng, A. Jacoby, N.W. Paton, C.A. Goble, “A. Brass: TAMBIS: Transparent Access to Multiple Bioinformatics Information Sources”, Bioinformatics, 2000; 16 (2): pp.184-185.

[16] Attwood TK and Parry-Smith DJ, Introduction to Bioinformatics. Prentice Hall. Harlow, 1999.

[17] The ISPIDER Project, URL: [http://www.ispider.man.ac.uk/].

[18] The AutoMed Project, URL: [http://www.doc.ic.ac.uk/automed/].