Yale University
Department of Computer Science

Integration of Heteregeneous Bio-Medical Databases: A Federated Approach using Semantic Schemas

Swathi Yadlapalli Gordon Shepherd Perry Miller
Abraham Silberschatz Luis Marenco

YALEU/DCS/TR-1348
April 2006

Integration of Heteregeneous Bio-Medical Databases: A Federated Approach using Semantic Schemas[*]

Swathi Yadlapalli[†] Gordon Shepherd[‡] Perry Miller[§]
Abraham Silberschatz[¶] Luis Marenco[‖]

Abstract

Biomedical experiments generate a vast amount of data that needs to be organized, integrated and analyzed. Important research challenges in the retrieval of these data include flexible, integrated analysis of data held in existing heterogeneous data sources. Indeed, the problems of supporting ad hoc queries across multiple data sources and correlating the data retrieved pose a host of challenging research problems. Our approach to integrate data from heterogeneous databases is to build a federated database system with a centralized mediator. We build a shared global schema and the mappings between the schemas are captured using rules. This paper focuses on issues concerning manipulation of large volumes of biomedical data in centralized, distributed or heterogeneous environments. We develop new computer science approaches to managing biomedical data, building on major biomedical informatics initiatives at Yale including over a decade of research performed as part of the national Human Brain Project. Both the functionality and performance of our system are being tested with data from the SenseLab database, CoCoDat database, and Cell Centered Database.

[*] This work is partially supported by NSF grant 0331548. This work is also supported by NIH grant P01 DC04732.

[†] Department of Computer Science, Yale University, CT

[‡] Department of NeuroBiology, Yale University, CT

[§] Yale Center for Medical Informatics, and Department of Anesthesiology, Yale University, CT

[¶] Department of Computer Science, Yale University, CT

[‖] Center for Medical Informatics, Yale University, CT

1 Introduction

The main thrust of our work is the integration of three neuroscience databases that contain data about neurons: SenseLab [6], a Human Brain Project neuroscience database at Yale, the Cell Centred Database [5], a database at University of California at San Diego and the CoCoDat database [4], built at the C. &. O. Vogt Brain Research Institute in Dusseldorf, Germany.

We have noticed significant theoretical barriers that complicate the integration of heterogeneous data sources. The most important of which is the representational heterogeneity of the data, that is, the differences in data models, schemas, naming conventions, and levels of granularity used to represent data that are conceptually similar. Additional challenges include the highly diverse nature of data, performance optimizations for translating queries and executing them across multiple databases, and methods to efficiently maintain mappings among databases that are autonomously managed and frequently changed. A new challenge presented by biological databases in particular is the frequent modification of the database schemas. Broadly speaking, the differences between databases can be classified according to four different conflicts:

1. ***Heterogeneity conflict*** Use of different data models and query languages (or software). For example, Relational vs. XML, Oracle vs. SQL Server.

2. ***Semantic conflict*** Use of same term to describe two semantically different concepts, or use of different terms to describe the same concept.

3. ***Descriptive conflict*** Naming conflicts, conflicts in the scope of attribute domain, scale, constraints, et cetera. It also includes the cases where the designers when covering the same domain opt to focus on different properties.

4. ***Structural conflict*** Use of different constructs to represent same real-world entities. For example, a concept can be represented as a separate object or as an attribute to an existing object.

There are known solutions to Conflict 1. We are using the Query Integrator System (QIS)[1], which is a database mediator framework addressing data integration from heterogeneous bio- sciences data sources to resolve this issue. This system, currently in the advanced prototype stage, has been developed by researchers at Yale Center for Medical Informatics. There are no good solutions to dealing with conflicts 2-3. Although some theoretical results, exists, there are many practical problems that need to be overcome. In this paper, we will concentrate on resolving these differences.

2 Related Work

The database community has long recognized the interoperability issues of heterogeneous data-sources. Past approaches can be categorized into two major groups: *Data Warehousing* [9] and *Database Federation* [2]. In the data warehousing approach, data from heterogeneous information sources is collected, mapped to a unified structure and stored in a central location. The data from the individual data sources is transformed to the warehouse format. It becomes necessary to periodically update the warehouse to reflect the changes in the individual data sources.

In the case of Database Federation, the idea is to create a large virtual database by combining the contents of several smaller ones, and introducing a central mediator that presents a uniform interface to the end user. The information required to answer the query posted is collected directly from the data sources, hence the results are up-to-date. Typically, the query should be decomposed into a set of sub queries and executed against the corresponding data sources and the result sets are returned to the mediator for integration. We are adopting the federated approach to integrate the heterogeneous bio-medical databases. Two major issues that come up in this context are how to generate a federated/global schema and the mappings between the global and the source-specific (local) schemas. A number of solutions have been presented for the former case; Manual Integration where the federated schema is the result of a domain expert driven schema editing process; and Automatic Integration where the responsibility for generating the federated global schema lies partially with the system. In our framework, we sought the help of domain experts to design a global schema manually based on the semantics of the domain. Two basic approaches have been proposed for dealing with the latter problem: Source-Centric Approach and the Query-Centric Approach. In the case of source-centric approach, each individual data source decides how to map the concepts in the local schema to those of in the global schema. This results in semantics being source-centric. In contrast, in the query-centric approach, concepts in the global schema are defined in terms of those in the local schemas. We are adopting the second approach as it is more suitable when the users need the ability to flexibly interpret and analyze information from autonomous sources.

Previous similar efforts in bioinformatics include *TAMBIS* and *DiscoveryLink*. TAMBIS (Transparent Access to Multiple Bioinformatics Information Sources) is a mediator-based integration system that uses a global ontology to form queries [10]. TAMBIS only addresses the horizontal dimension of data integration i.e., it mostly integrates sources that have complementary data. Our system takes into account the potential overlapping aspect or probable incompleteness of sources; thus presenting a more reliable and complete solution. IBMs DiscoveryLink is a wrapper-oriented integration system [11]. Users connect to DiscoveryLink and submit a query in SQL on the global schema. The essential idea is the same; we are trying to incorporate more features into our design so that it can accommodate frequent schema changes, fuzzy mappings, and incomplete data.

3 Methods

We will look at problems, such as mapping queries formulated over the global schema into queries against local database schemata, mapping actual data from the local databases to the global one, and integration of results from different data sources. In the past, Robbins and Karp have advocated the suitability of federated multi-database approach for integrating biological databases. We adopted a federated architecture with centralized mediator approach to integrate access to heterogeneous, distributed biological databases.

Architecture of the system The system is organized in four levels: Client Application, Mediator, Wrapper and the Local Databases. The user at the highest level interacts with the mediator, not with any of the component databases. The mediator is central to our federated database system. It acts as a bridge between the user applications and the actual data sources. It does the processing common to the component data sources. The source-specific transformations are done in the respective wrappers.

- **Mediator** It has several entities like rules, programs, and a global schema associated with it.
  i. ***Global schema:*** To build the desired federation of databases, we do believe we need a 'shared data model across data sources'. The

mediator's schema, Cm, describes the content of the data resources that are members of the federation. Ideally, we would like Cm, which we also refer to as the global schema, to be designed based on the semantics of the domain, rather than the organization of data in the external resources. We are building the global schema in an incremental fashion depending on the users needs. To make changes visible, we add rules suggesting mappings between the updated global schema and component database schemas.

ii. ***Rules:*** They are mapping functions between the mediator's global schema and the component database's local schemas. We use these mapping functions to carry on the query as well as data transformations. Our work is based on the assumption that relationship between schemas can be captured as a set of rules. As a first step, we are crafting them by hand in consultation with the domain experts. In the future, we will generate the mapping functions from models which are maintained by the domain experts. A survey of approaches to automatic schema matching has been done by Rahm, Bernstein [3].We have noticed three types of mapping functions so far:

a. *Term Mappings:* Terms/data items in the two databases are related to each other. The relationships between those terms can be one-one or one-many i.e., or, and, any combination of or and and.

Ex: NeuroSci.Axon Terminal corresponds to SenseLab.Axon. This means the term Axon Terminal in the global schema NeuroSci corresponds to the term Axon in the local schema SenseLab.

b. *Simple Structural Mappings:* Schema element in one database is related to schema element/elements in another database. This case is similar to the one described earlier, the only difference here being relationships between schema elements instead of data elements.

Ex: NeuroSci.Neuron corresponds to CocoDat.NeuronType AND CocoDat.Layer AND CocoDat.Brain Region.

c. *Conditional Structural Mappings:* The mapping is condition-dependent. This is the case where the schemas together with allowed data values are transformed in an integrated fashion to map a query to a target database in which both schema and the structure of the data values are different.

Ex: If NeuroSci.Neuron equals Neocortical Pyramidal Neuron:Superficial then Cocodat.NeuronType corresponds to Pyramidal Neuron AND CocoDat.Layer corresponds to 2—3 AND CocoDat.Brain Region corresponds to General Cortex.

iii. ***Programs:*** The mediator has several program modules to per-

form tasks such as query parsing, query and data transformation from global to local schemas, and result integration. These functions are common to all the data sources and so carried out in the mediator.
a. *Query Parser:* This parses the user queries expressed against Cm into intermediate code expressions. This parsing step is necessary for further processing, for instance when applying mapping functions.
b. *Query transformer:* The query is transformed from the global to local schemas of the component databases. This is done by applying the mapping functions that map the mediator's conceptual schema Cm to each of the data source's external schemas Es1, Es2,...Esn. These transformed queries are then sent to the wrappers associated with the corresponding databases.
c. *Result integrator:* This acts as a synchronization layer, combining results retrieved from component databases. The mediator receives the results sets expressed against the corresponding data source's local schemas from each of the wrappers. The mapping functions once again are applied to convert the data from local to global schemas. We remove the redundant information and merge the result sets.

- **Wrapper** QIS [1] takes the place of the wrapper for each of the component databases. They contain the source specific code. Wrappers take the transformed queries as input and produce queries that can be actually executed against the database. Transformations are done to suit the source's data model, and query language.

4 Implementation Details

In our system we are basing the mediator's schema, i.e, the global schema, on the relational data model. We are extensively using *Prologs* [7] powerful pattern matching and list processing capabilities to express the mapping functions between the schemas. Specifically, we are using SWI-Prolog which is an open source implementation of the programming language Prolog, licensed under the Lesser GNU Public License. It has a rich set of features, libraries, tools, and extensive documentation and it runs on Unix, Windows and Macintosh platforms. The mediator's program code is implemented in Java [8]. Currently the schemas are downloaded from the web and stored on the computer for mapping purposes.

5 Present Status

We have developed the initial design of the proposed federated system of neuroscience databases. Our initial focus has been on integrating *SenseLab, Cell Centered Database, and CoCoDat*, focusing initially on a relative modest set of data elements. We have designed the pilot version of the global schema, *NeuroSci*, based on the semantics of the neuroscience domain and covering the concepts of the three component databases. The schema is based on the relational data model and the user queries are expressed using SQL. We have also developed the pilot set of rules that map the global schema, NeuroSci, and the three component databases. We continue to explore the issues involved in mapping between schemas of neuroscience databases. We will ultimately extend the approach to other biomedical domains as well exposing the real world issues and problems in mapping biomedical database schemas. We have developed a generic query translator which takes a user query expressed against global schema as input, applies the mapping functions, and produces sub-queries that are expressed against component database schemas. We tested our system to map queries between the global schema, NeuroSci, and the local database schemata, SenseLab and CoCoDat. We developed a generic result integrator which does the reverse data translation and merge the result sets by removing redundancy.

6 Future Work

There are engineering as well as research challenges to the integration problem. Engineering challenges include coming up with global schemas shared by many, and creating schema mappings. The global schema we developed covers the information in the three component databases. We tried to design it based on the semantics of the domain. It is being incrementally built based on the users needs. We manually created the schema mappings. We realized the process is tedious; we intend to semi-automate the matching process in the future.

Research challenges include:

a. ***Maintaining mappings:*** when the schemas are updated, can we use the old mappings to infer the new mappings? This includes creating models upon which the mapping functions are based and inferring the new mappings from the existing models. There are many techniques like data mining and machine learning which help us in creating models from the data.

b. ***Fuzzy/Probabilistic mappings:*** it might be the case that the terms

belonging to two databases might not map perfectly or the domain experts may not be sure of the relationship. Ex: A in DB1 is similar to B in DB2, A in DB1 is 40% same as (B AND C) in DB2. This essentially means rating each mapping function on a level of 0-100 depending on the accurateness of the relationship. We think this information can be used in the generation of results, i.e., the user will be given results attached with the probability values giving him an idea about the accurateness of those results.
c. ***Changing meanings:*** the terms are interpreted differently as time goes by. There is no way of automatically detecting the changes, we think periodical checks should be done and changes should be made accordingly.

7 Conclusions

We developed a set of tools and approaches to integrate access to heterogeneous, distributed, and autonomous neurosciences databases. We explored the real world challenges in implementing the federated approach to integrating heterogeneous biomedical databases in the course of our project. We are working with our biological collaborators to have them use the tools we developed to help create, and iteratively refine our design. Although much of the work is being done in the context of neuroscience data, similar problems exist throughout biosciences as well as in many other domains that involve complexly inter-related, heterogeneous, evolving data.

8 Acknowledgements

This work is supported in part by NSF grant 0331548. This work is also supported by NIH grant P01 DC04732.

References

[1] Luis Marenco, Tzuu-Yi Wang, Gordon Sheperd, Perry Miller, and Prakash Nadkarni. QIS: A Framework for Biomedical Database Integration. In *J AM Med Inform Assoc*, pages 523-534, 2004.

[2] Sheth AP, and Larson JA. Federated Database Systems for Managing Distributed, Heterogeneous, and Autonomous Databases. In *CSACM*, 1990.

[3] Rahm E, Bernstein PA, and Griggs K. A survey of approaches to automatic schema matching. 2001.

[4] Dyhrfjeld-Johnsen J, Maeir J, Schubert D, Staiger J, Luhmann HJ, Stephan KE, and Kotter R. Cocodat: a Database System for Organizing and Selecting Quantitative Data on Single Neurons and Neuronal Microcircuitry. In JNM, 2005.

[5] Martone ME, Zhang S, Gupta A, Qian X, He H, Price DL, Wong M, Santini S, and Ellisman MH. The cell-centred database: a database for multiscale structural and protein localization data from light and electron microscopy. 2003.

[6] Miller PL, Nadkarni P, Singer M, Marenco L, Hines M, and Shepherd G. Integration of multidisciplinary sensory data: a pilot model of the human brain project approach. In JAMIA, 8, 2001.

[7] SWI Prolog. Available at *http://www.swi-prolog.org*, March 10, 2006.

[8] Kernel Prolog. Available at *http://www.kprolog.com/jipl/index_e.html*, March 10, 2006.

[9] Chaudari S and Dayal U. An overview of data warehousing and olap technology, 1997.

[10] Baker P, Brass A, Bechhofer S, Goble C, Paton N, and Stevens R. TAMBIS: Transparent Access to Multiple Bioinformatics Information Sources. In *ISMB98*, 1998.

[11] Haas L, Schwarz P, Kodali P, Kotlar E, Rice J, and W.Swope. DiscoveryLink: A System for Integrated Access to Life Sciences Data Sources. In *IBM Systems Journal*, pages 40(2), 2001.