MARGBench – Information Fusion for Modeling and Simulation of Metabolic Networks*

Uwe Scholz Andreas Freier Ralf Hofestädt
Matthias Lange Andreas Stephanik

Bioinformatics / Medical Informatics
Institute of Technical and Business Information Systems
Otto-von-Guericke-University Magdeburg
P. O. Box 41 20, D–39016 Magdeburg
E-mail: firstname.surname@iti.cs.uni-magdeburg.de
Tel.: ++49-391-67-18659 Fax: ++49-391-67-12020

1 Introduction

Biotechnology needs new methods, which allow retrieval, representation, modeling, visualization and simulation of biochemical networks. The related data is stored in various databases or data sources. Database systems for genes, proteins and metabolic reactions and integration approaches are available. In order to access this data an integration of molecular database systems and analysis methods [1] is still missing.

In this paper we present an architecture of an integrative molecular information system, which is called MARGBench. This system demonstrates new capabilities of the integration approach and provides essential design bases. The system combines the molecular information fusion and simulation of metabolic networks [3]. One application of our system is the detection of inborn errors [5, 4].

The idea to integrate molecular data is not a new one. The EcoCyc system of Karp [7] was one of the first integrative molecular database systems, although it was restricted to data about the Bacteria E. coli.

Another approach for an integrated access to different molecular databases via WWW is described in [2]. This system, called SRS (Sequence Retrieval System), is based on local copies of each component database. For that reason the owner of a component system has to implement an export interface, which converts the data into a simple text or hypertext format. Because of that, one has to deal with semantic as well as functional loss. During a query the user can specify in which systems to search. The results of the query are sets of

*This work is supported by the German Research Council (DFG).

WWW-links which the user can navigate through and retrieve the required the information. SRS is a data retrieval system, which does not include complex analysis programs. We also did not find real data-integration; i.e. data for one real world object (e.g. an enzyme) coming from two different databases (e.g. KEGG [6] and BRENDA [8]), which is represented twice by different WWW page objects. The information fusion is still task of the user. In order to overcome these deficits we developed a new architecture.

2 Architecture

An overview of the system's architecture is available on our web server under the address: `http://wwwiti.cs.uni-magdeburg.de/iti_bm/marg/` or in Figure 1.

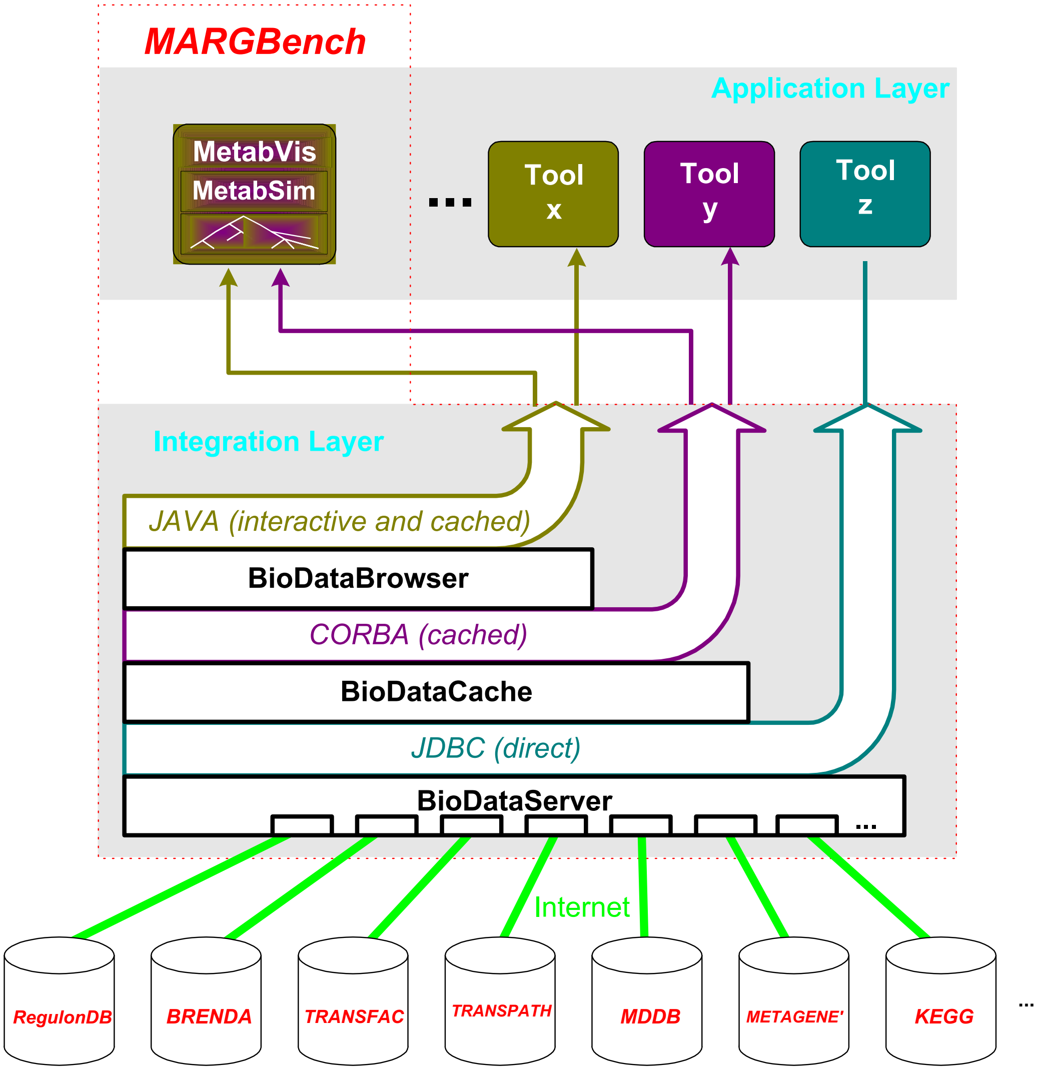

Figure 1: Architecture of the MARGBench

On first sight the system is divided into two parts, the integration layer and the application layer. But the integration layer consists of three different modules, BioDataServer, BioDataCache and BioDataBrowser.

Our BioDataServer realizes the access to different distributed and heterogeneous data sources (e.g. KEGG or BRENDA). For this reading-only-access special software modules, so called adapters, are responsible. Using special integrated user schemes, the BioDataServer combines the outcomes of adapter queries into integrated results, which is called information fusion. The BioDataServer is available as an Internet server and is acceswsible via a JDBC interface.

Next level in the integration layer is the BioDataCache. It handles the storage of the fusioned data in a data warehouse. Its access interface is based on the CORBA technology.

The third part in the integration layer is represented by the BioDataBrowser. This module allows the user to browse through the fusioned data, similar to a windows file explorer. A JAVA interface is offered to access this component.

Generally, the three described different opportunities (JDBC, CORBA or JAVA) are offered to access the integration layer.

As mentioned already above, the MARGBench does not only consist of the integration layer but also of the application layer. The reference application of the MARGBench is the visualization tool MetabVis, including the simulation environment MetabSim. This reference application uses the CORBA interface and the JAVA opportunities to access the fusioned data of the integration layer.

3 Components of the System

The components of our system, the BioDataServer, the BioDataCache, the BioDataBrowser and the visualization tool MetabVis including simulation environment MetabSim are described in more detail below.

In general, the BioDataServer realizes a logical database integration based on the concept of federated databases. A workable Internet access to the molecular-biological databases is the main prerequisite for a database integration. Thereby several problems must be solved:

- different interfaces (e.g. CGI, JDBC, ...)
- different query languages (SQL, OQL, non-standardized, ...)
- different data presentations (HTML, flat files, database objects, ...)
- different data structures (static, dynamic, ...)
- different data models (ERM, OO, ...)

To hide this heterogeneity, the BioDataServer uses adapters for physical data access. In the case of a HTML data source the adapter accesses the specific URL and parses the resulting HTML-page. These adapters are special software modules, which can be generated semi-automatically. For this generation a description file is necessary. A description file contains semantic structure and syntax information about the data source, which should be integrated into our system. This information enables a mapping between the data fields of the docked data sources and the attributes of an integrated user scheme. The integrated user scheme is the basis for the information fusion. It describes the accessible attributes of integrated data sources in transparent form.

In order to obtain a complete and wide spectrum of data, it is recommended to access as many databases as possible. Therefore the queries will be executed in each relevant data source. Information about the distribution of the queries are stored in an integrated user scheme. This scheme is relational and defines the source for each attribute. On the basis of these schemes it accesses the related attributes in the specific databases.

As can be seen, an automatic mechanism is necessary to merge the data values from the various databases. This is one task of the BioDataServer and can be solved using mathematical set operations.

The premise to access the database integration server by computer programs is the definition of an interface. Because the server should be accessible via the Internet, a communication protocol and a query language must be specified. Nowadays lots of database systems exist, which support a subset of SQL as query language, which in turn is based on the relation model and is standardized. This was the reason to support SQL elements by the BioDataServer. Different techniques in the field of interfaces for remote database access have been established e.g. JDBC and ODBC. ODBC is only supported by Microsoft platforms. Therefore the BioDataServer currently offers a JDBC driver, which provides a standardized database access to JAVA applications. Consequently any JAVA platform can simply access the BioDataServer by related JAVA programs.

The main advantages of this BioDataServer are the transparent physical database access, the dynamic building of a new virtual, logical integrated database, a standardized access interface, a client-server capability and the platform independence. With the offered JDBC driver the integration service of the BioDataServer is also stand alone useful for other external Java applications.

Next component of our architecture is the BioDataCache. It provides an interface for the BioDataServer, based on the Common Object Request Broker Architecture (CORBA). The access to the BioDataServer is realized by the JBDC driver of the integration module. Furthermore the BioDataCache uses an integrated user scheme for the selection of attributes, which should be integrated.

Once data from the integration service is read, it will be stored in the underlying object-oriented database-system (POET). Using a self implemented thin client-interface, any client can access objects registered in the BioDataCache and work with them in their own scope (three-tier-architecture).

By storing the fusioned information in the cache, a new data source will be created. This new database represents a quality of meta database and is comparable to a data warehouse. The offered CORBA interface, similar to the BioDataServer, enables other software tools the access to the BioDataCache.

Last component of the integration layer is the BioDataBrowser. Developing DBMS-supported applications forces the programming of database-related components to establish database-connection, query the data, transmit the results, store the data and so forth. The BioDataBrowser provides this functionality and can be included as a component in different Java-applications.

The visualization tool MetabVis, including the simulation environment MetabSim, as the reference application of the MARGBench, has been designed to animate metabolic networks and belongs to the group of discrete simulations. Because of the automatic access to BioDataCache and BioDataBrowser the simulation of complex networks is possible.

The formalism of the MetabSim-grammar bases on the approaches of [3]. The necessity to choose a discrete method arose from the generally high complexity of biological regulatory procedures.

In MetabSim the discrete states of cellular compartments are expressed by mixtures of substrates and are called configurations. The step from one configuration to a following (derivation) is defined by a rule-set, where each rule abstractly describes a biochemical process. A metabolic rule is a tuple $r = (V, N, k, E, I)$ with the elements:

- V Substrates
- N Products
- k Kinetics,
- E Influences,
- I Limits.

Rules do not only describe simple chemical reactions. They also allow to model chains of reactions, where data is only marginally available, as black box. Finally, the regulatory effect benefits from cascading influencial functions. The data to extract rules and the values to describe the substance-behavior is directly taken from the MARGBench system.

With the help of these reference applications the user has the opportunity to analyze metabolic networks. The necessary parameters are extracted automatically with the help of the MARGBench. This prototype simplifies the access to different heterogeneous data sources and the following analyses of the fusioned information in a new way.

4 Discussion

Goal of our MARGBench project is to offer a powerful information system for the integration of data and simulation of metabolic processes. The system provides services to integrate the data of different databases and to store a part of the data locally. Three interfaces, JDBC (uncached), CORBA (cached) and a browser (interactive and cached) for the integration layer are available. Its interfaces use standardized query languages or interfaces like JDBC, SQL and OQL to query the data. Furthermore the integration layer is divided into three modules: the BioDataServer, the BioDataCache and the BioDataBrowser. These modules provide the user with powerful access opportunities at different levels.

The concept of information fusion in the system, which is realized in the integration layer, enables and supports an integrated view on worldwide distributed information. With the opportunity to define integrated user schemes, the integrated data set is easily adaptable.

As a reference analysis software in the application layer, the visualization tool MetabVis including MetabSim, demonstrates the capabilities of MARGBench to view metabolic networks. MetabSim is able to animate regulatory effects in discrete states of cellular compartments. With the help of the different access interfaces the coupling of further analysis tools is simply realized.

Most components of our system are available as prototypes, e.g. the BioDataServer. Whereas other modules are still objects of our current work. Our roadmap points to extend the number of accessed databases, to automate the adaptation to special cases of application and to improve the simulation model step by step. One application of our system is the detection of inborn errors (supported by the German BMBF).

References

[1] J. Collado-Vides, R. Hofestädt, M. Mavrovouniotis, and G. Michal. Modeling and simulation of gene regulation and metabolic pathways. *Bio Systems*, 49(1):79–82, 1999.

[2] T. Etzold, A. Ulyanow, and P. Argos. SRS: Information Retrieval System for Molecular Biology Data Banks. *Methods in Enzymology*, 266:114–128, 1996.

[3] R. Hofestädt and F. Meinecke. Interactive Modelling and Simulation of Biochemical Networks. *Computers in Biology and Medicine*, 25(3):321–334, 1995.

[4] R. Hofestädt, U. Mischke, M. Prüß, and U. Scholz. Metabolic Drug Pointing and Information Processing. *Medical Informatics Europe*, 68:12–15, 1999.

[5] R. Hofestädt, M. Prüß, U. Scholz, and H. Urban. Molekulare Bioinformatik – Molekulare Informationssysteme zur Erkennung von angeborenen Stoffwechselerkrankungen. *Magdeburger Wissenschaftsjournal*, 3(1):29–40, 1998.

[6] M. Kanehisa and S. Goto. KEGG: Kyoto Encyclopedia of Genes and Genomes. *Nucleid Acids Research*, 28(1):27–30, 2000.

[7] P. D. Karp. A knowledge base of chemical compounds of intermediary metabolism. *CABIOS*, 8(4):347–357, 1992.

[8] D. Schomburg, I. Schomburg, A. Chang, and C. Bänsch. BRENDA the Information System for Enzymes and metabolic Information. In R. Giegerich, R. Hofestädt, T. Lengauer, W. Mewes, D. Schomburg, M. Vingron, and E. Wingender, editors, *Proceedings of the German Conference on Bioinformatics (GCB '99), Hannover, October 4-6*, pages 226–227, 1999.