Metabolic Information Control System

Andreas Stephanik, Ralf Hofestädt, Matthias Lange, Andreas Freier

Otto-von-Guericke-University of Magdeburg
Department of Computer Science
Institute of Technical and Business Information Systems
Bioinformatics Research Group
Universitätsplatz 2, D-39106 Magdeburg, Germany

ABSTRACT

Systems for the integration of data in molecular biology are becoming more and more important because scientists as well as applications can not always find all relevant data in one database. Another advantage of data integration is the possibility to derive information of a new quality using semantic relations between the integrated data of various databases. With the requirement to feed an application for the simulation of metabolic pathways with necessary data we are developing a system for the integration which is based on a hybrid approach. As a first possibility a datawarehouse is used for an easy and fast access. The storage system of this datawarehouse is an object oriented database system. The second possibility of our hybrid approach of integration is the capability of a homogeneous online access to various data sources such as database systems and flat file based systems via the internet. The components for the data access are modular. Thus they can be created and modified easily using a semi-automatic process. Therefore a mediator based system is available for the integration of data stored in databases and flat files. The applications can access the integrated data via various interfaces such as CORBA, JDBC or TCP/IP.

Keywords: Integration, Data Retrieval, Databases, Flat Files, Distributed Data Sources, Simulation of Metabolic Pathways

MOTIVATION

Scientists in molecular biology use application to analyze, compute or simulate complex scenarios. Those applications need data from various databases, because mostly not all relevant data can be found in one data source. An investigation distinguishes molecular data into 17 categories [1]. Accordingly, about 300 WWW based data sources are listed. The WWW is developing into the most powerful medium for information retrieval. This fact is consequently reflected in molecular biology, so that the majority of databases are accessible using the internet. With regard to persistent data storage two general techniques are used: flat files and database systems (DBS) [2]. The public access is mostly done by a WWW server, which acts as middleware between the user interface and the database.

In order to take advantage of the potential of these valuable databases it has to be considered that Bioinformatics is an inherently integrative discipline [3], requiring access to data from a wide range of sources. Without the ability to combine these data in new and interesting ways, the field of Bioinformatics would be severely limited in scope. Consequently, the integration of databases can help to derive new information. With these requirements some systems for the integration of biological data have been or will be developed.

We have begun to develop a flexible integration system in order to integrate data for several problems and applications. The first application is a metabolic information application applying the integration system. This system for data integration together with the metabolic information application are explained in the following paper. At first a short overview about the topic system for data integration is given.

SYSTEMS FOR DATA INTEGRATION

Systems for an automated acquisition of information from heterogeneous molecular biology databases for analyzing or simulation of biological processes are primarily based on four technical approaches which are closely related to distributed database management systems (DDBS) [4]. These are: hypertext navigation (e.g. KEGG [5]), data warehouse (e.g. SRS [6], PEDANT [7], HUSAR [8]), multi database query languages (e.g. BioKleisli [9], OPM [10]), agent based techniques (e.g. Multiagents [11]). Those systems enable the access to various databases and the scientists do not have to search for desired data in the forest of the internet.

Systems for the integration of data should enable a homogeneous access to dispersed and heterogeneous data sources. The diversities of data sources regarding the data formats and interfaces have to be hidden using

transformations so that the integrated data can be accessed uniformly.

The SRS System

A sample for a data integration system is the SRS (Sequence Retrieval System) from Lion Bioscience (http://www.lion-bioscience.com). The SRS system is a datawarehouse with over 100 internal and public domain databases. The meta-level approach is the base of that system, containing all of the relevant information about the structure, format, and syntax of the underlying databases. The meta information also includes information about database cross references, enabling cross database queries and cross database views. If a new database were integrated, all meta information would have to be created and the data have to be copied from the owner into the SRS system. Thence only databases are available which were imported. For an update of a partial database the according data have to be copied into the SRS system.

OUR HYBRID APPROACH

Because of the requirement to feed an application for the simulation of metabolic pathways with necessary data we are developing a system for the data integration which bases on a hybrid approach. In addition to the metabolic information application other applications for different problems shall be applied with the data integration system. The first possibility of our hybrid approach of data integration is the capability of a homogeneous online access to various data sources such as database systems and flat file based systems via the internet. The components of the online integration system are explained in the next section Mediator-Wrapper. As a second opportunity a datawarehouse is used for an easy and fast access, discussed in the further section BioDataCache.

Mediator-Wrapper

From our perspective, the mediator-wrapper concept [4] should be the basis for a homogenized, integrative and efficient retrieval of molecular biology data. A wrapper exports some information about its source schema, data and query processing capabilities for each data source. A mediator centralizes the information provided by the wrappers in a unified view of all available data (stored in a global data schema), decomposes the user query into smaller sub-queries (executable by the wrappers), gathers the partial results and computes the answer to the user query. Henceforward we will use the term adapter, because it will be demonstrated that not all properties of a mediator are necessary in our approach.

The origin of the data are heterogeneous „read only“ data sources. Moreover, interfaces for high number of molecular biology applications to further process the queried data have to be provided. Consequently, it is therefore necessary to choose an approach based on the mediator-wrapper concept, which respects the specific characteristics of molecular biology data or database systems, by meeting specific requirements. The scenario and the problem specific integration of heterogeneous data sources respectively has to tolerate differences in data models, data structures, query possibilities and data storage techniques. Furthermore, the opportunity for a homogeneous data source access, the implementation of data caching for an efficient data retrieval, the provision of adjustable global schemas based on local schemas, a high level interface, and the possibility to simply attach applications for further processing of the queried data must be considered sufficiently.

Finally one can conclude that the described requirements have to meet a general architecture for the integration of molecular biology data sources [12]. The idea of a mediator based database integration approach has resulted in the system architecture of the BioDataServer (BDS), which is shown in figure (1).

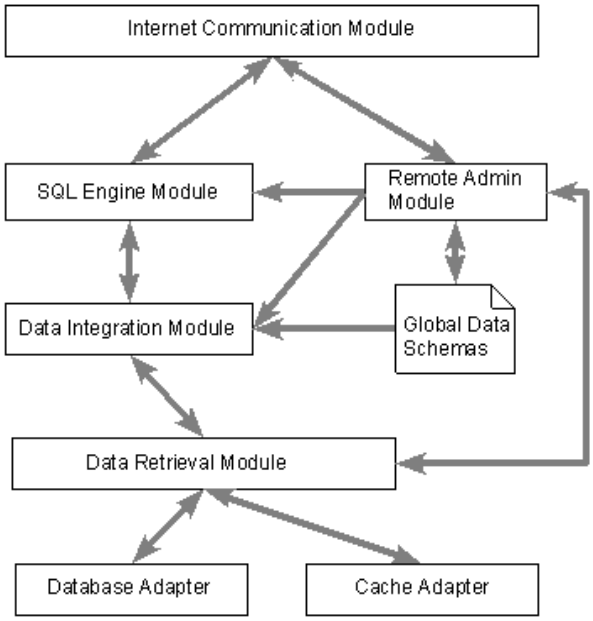

Figure 1 Architecture for our mediator based database integration

The shown architecture is realized as a client-server system where the BDS is the server and any molecular biology application can act as a client. In this way several users and related global data schemas can be managed.

The Internet Communication Module enables access of client applications using the Transmission Control Protocol/ Internet Protocol (TCP/IP) for data transmission to and from the BDS and it should be possible to remote control the BDS. Mechanisms like user and process management, editing of global schemas, data source wrapper control and integration progress information are provided by the Remote Admin Module. A high level data retrieval mechanism for a read-only access to the integrated data, with a high degree of system independence and acceptance, is offered by the SQL

Engine Module. Furthermore, the declarative character (specifying the properties of retrieved data and not how to obtain it) of the used mediator technique was taken into account. For this a data retrieval subset of the well standardized SQL (Structured Query Language) has been implemented, which is one of the most popular query languages for relational DBS's. The Data Integration Module is the core of the BDS, which includes a query and operator processor. In addition, the several global data schemas are managed by this module, which are again the basis for the integration process. This integration process implements the query decomposition into sub queries, the transformation into integration operators, generation of execution hierarchy and finally the merging of sub query results in exactly that order. The Retrieval Module organizes all adapters in a similar way to an operating system, which manages and controls the adapters. It loads each single adapter, manages an adapter list, propagates the exported data source schemas, dispatches data source operations and ensures the robust adapter operation (exception handling). The functions of a Database Adapter realizes homogeneous access to the data sources. For each wrapped data source the following tasks are performed: reproduce a relational view at the specific data model, export a view to the data source schema and provide data source independent data source operations.

The integration service BDS accesses the data sources using adapters. We have to distinguish between the access to database management systems (DBMS) and to file based systems (data stream parsing). Due to differences in interface, schemes and data structure of the accessed data sources, the corresponding adapters have to be programmed depending on those specific properties.

All adapters have to implement a defined interface for the communication with the BDS. The functions of this interface are based on data source operation and contain all the methods for the specific data access. Thus, the specific methods of data access are hidden from the BDS, which only calls the defined functions of the adapters to query the data.

If a data source were to be connected to the BDS an adapter would first have to be created. For the creation of an adapter its specific methods for data access and the relational scheme of the data have to be constructed.

The manual programming of the adapters includes an unnecessary effort in writing code segments, which are identical in every adapter (e.g. headers of functions) or depends on some information and entails trivial typing of code (e.g. data scheme declaration). Therefore we use the approach for generating the adapters. Another advantage of adapter generation is the simple facility for readjusting the adapters because of changes of the structure of a data source. In this case the adapter will be generated again including the modified information.

The tool, supporting the adapter generation, needs specific information about the data access, a data schema and in the case of file access a description of the files' structure for its work. This specific information of a data source is saved in a text file, called description file. This description file will be read from the generator. The user, who wants to create an adapter, has to analyze the possibilities of data access, model the data schema and writes this information with a defined structure into the description file. The first part of the description file contains information about data access, the second part contains schema information. The schema has a similar syntax as a table definition in SQL. As already mentioned, a description of the file structure of flat file data sources also has to be provided. Finally, the generated adapters are connected to the BDS, which can map the data source operations to the data sources using the functions of the corresponding adapters.

By the use of this approach we can enable an online access for the extraction of necessary distributed data for applications in a fast way. The largest disadvantage of the integration using an online access is a slow connection via the internet. Therefore we have developed a data-warehouse as a second capability in order to provide necessary data for an application.

BioDataCache

The second opportunity of our hybrid approach for data integration is a datawarehouse named BioDataCache (BDC). The storage system of this datawarehouse is an object oriented database system. Data are mainly imported by using the BioDataServer (BDS). The data import applying other tools is also possible.

The advantage in opposite to the online integration is the faster access resulting from the access via the local network. Another reason for the faster access is that no data have to be extracted from text or HTML files. All data can be queried from the database system of the BDC.

With the BDC integrated data can be stored persistently and queried as database objects. With BDC it is possible to create individual integration databases easily by the use of the BDC Configuration Tool. Thus it becomes possible to perform data analysis, consistency evaluation, enrichment and process modeling, without getting in conflict with write permissions, availability and database modification of component databases.

The process of using a BDC: First, the database content is described using the Interface Definition Language (IDL) [13]. This can be done by using a standardized CASE-tool such as Rational Rose (http://www.rational.com/products/rose), other tools or just manually. Then the BDC Configuration Tool will process the designed IDL document and automatically create the related integration database. At last the Import Module will retrieve integrated data by accessing the SQL interface of the BDS.

There are several interfaces provided by the system, e.g. Common Object Request Broker Architecture (CORBA) access [13, 14] and XML access. Additionally, there are interactive methods to query the object networks stored in the BDC. They can be embedded as components into other software. Firstly, there is the Network Browser

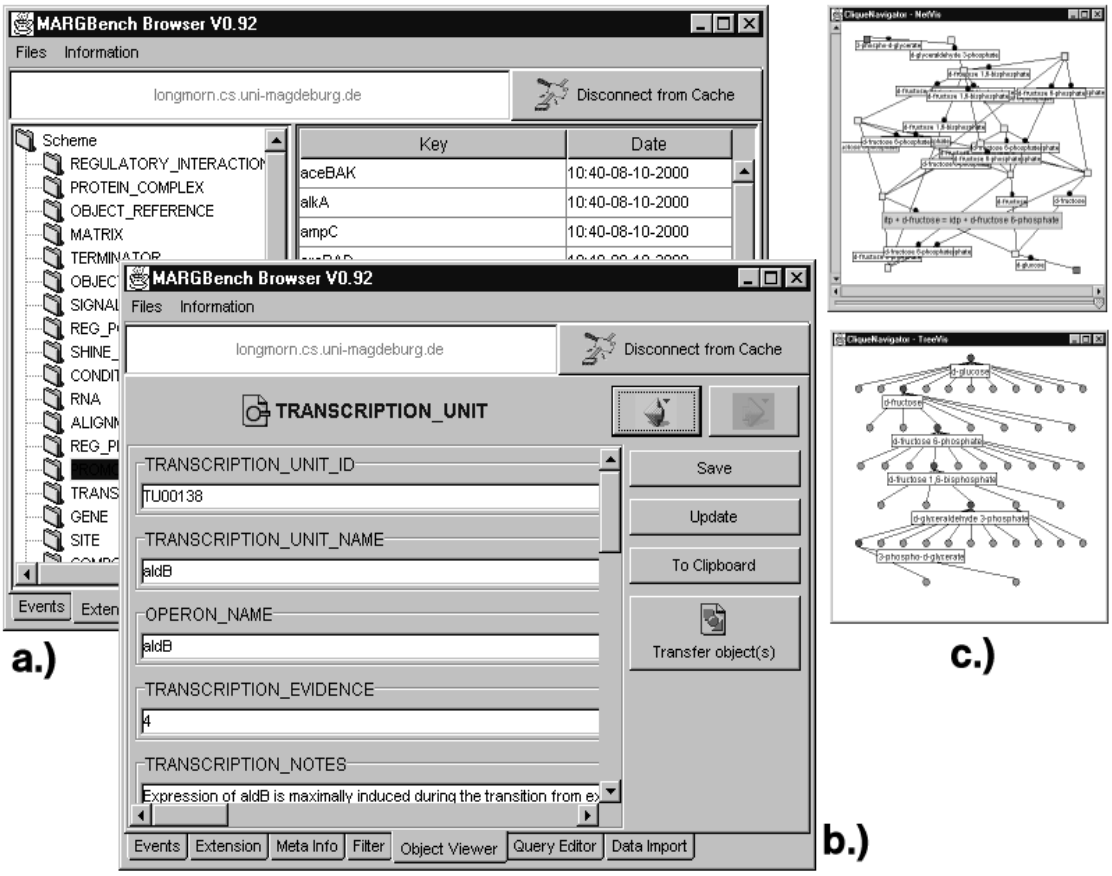

Figure 2 Screenshot of the BioDataCache: a.) Browsing object networks b.) Viewing and editing of network objects c.) Visualization of network graphs and paths

Access, where the user interactively navigate through the networks, as illustrated in figure (2).
Secondly, the user can search for paths and cliques by using the Network Navigation Access. Because of data security the BDC server should be installed at the Intranet. Multiple users can get access to the system as clients by using the provided applications and API's.
To summarize, the add-on BioDataCache is an integrative application example for the BioDataServer that provides the capabilities to support the database modeling and implementation of individual user-defined biological and biochemical processes as object networks.

APPLICATION

The BDS was designed as a universal applicable component for a homogeneous data acquisition in close context to molecular biology tools. The attachable software ranges from simple analysis tools (e.g. structural metabolism analysis: http://www-bm.cs.uni-magdeburg.de/phpMetatool) via various molecular information systems [15] up to complete frameworks for complex problems like simulations (e.g. our project MARGBench [12]). In the MARGBench (MARG – Modeling and Animation of Regulative Gene Networks) project (http://www-bm.cs.uni-magdeburg.de/iti_bm/marg) the BDS is responsible for the data acquisition.
Based on the rule based modeling of metabolic processes, we implemented the simulation environment MetabSim within the MARGBench project for the analysis and visualization of gene controlled metabolic processes. The simulation system MetabSim is the implementation of our rule based model. A rule currently contains the stoichiometry of substrates and products, enhancers, inhibitors, factors and the elasticity coefficients of one complex reaction. A second data type of our rule model is the metabolic state representing stepwise the configuration of compounds and enzymes in the metabolic network. The whole data structure is mapped into a database so that all rules and states are stored herein. In addition, the MetabSim contains a derivation logic. Because the system has been designed modular, several derivation modules can be implemented and applied independently. After defining the rule set and the root configuration (default cell states) the derivation logic can be applied to the data. In the first step, the „Rule Selection“ module accesses the current state and calculates the rules which can be applied, because their premise is becoming true related to the current state. The „Rule Application“ module calculates the following configuration(s). Optionally, the reaction time is applied by a „Rule Kinetics“ module. The new

configurations (states) are the input for the next derivation step.

Another application of the MARGBench project is the BioDataBrowser. The BioDataBrowser enable a graphical user interface to all data stored in the BioDataCache. For example it can be used for the interactive selection of metabolic pathways and single reactions for the export to the described application MetabSim. For the data exchange between BioDataBrowser and applications an active interface is available. The application program (MetabSim) implements this data exchange interface. When the user selects one or more objects from the cache the data exchange interface is called and the algorithm in MetabSim processes the transformation into MetabSim rules.

The advantage of our concept compared to a static data access is the integration of relevant molecular database systems using the BDS. The necessary data for the MARGBench applications are currently read from the data sources KEGG and BRENDA. In addition, our generic approach can be used in order to create adapters easily for an access to supplementary data sources. Thus we are developing a flexible system for the data integration and offer a gateway to the masses of data for various applications and problems.

CONCLUSION

Using methods of Biotechnology metabolic processes can be analyzed. For example the sequencing of genes, the detection of proteins, the recognition of the 3-D structure of proteins, the DNA/Protein interactions and the analysis of gene controlled biochemical networks is create an exponential growth of molecular data. The resulting databases will be the backbone of the computational analysis of metabolic processes. Today nearly 300 database systems of molecular data are available via the internet. The automatic homogeneous access and the integration of molecular database systems is one current topic of Bioinformatics. Thus, the key idea of our approach is the federated database integration based on the mediator approach in conjunction with specific adapters. Therefore, we developed and implemented tools which allow the semi automatic generation of adapters and the automatic generation of the user specific meta-database systems, which represent the user specific integrated molecular data. The applicability of our approach is demonstrated in the MARGBench project.

REFERENCES

[1] Baxevanis, A. D., Nucleic Acids Research, 2001, vol. 29, no. 1, pp. 1-10.

[2] Frishman, D. et al., Comprehensive, comprehensible, distributed and intelligent databases: current status, Bioinformatics, 1998, vol. 14, no. 7, pp. 551-561.

[3] Roos, D. S., Bioinformatics--Trying to Swim in a Sea of Data, Science, 2001, vol. 291, no. 5507, pp. 1260-1261.

[4] Özsu, M. T. and Valduriez, P., Principles of Distributed Database Systems, London et al.: Prentice-Hall, 2nd international edition, 1999.

[5] Ogata, H. et. al, KEGG: Kyoto Enzyclopedia of Genes and Genomes, Nucleid Acids Research, *1999*, vol. 27, no. 1, pp. 29-34.

[6] Etzold, T. et al., SRS: Information Retrieval System for Molecular Biology Data Banks, Methods in Enzymolgy, 1996, vol. 266, pp. 114-128.

[7] Frishman, D. et al, Functional and structural genomics using PEDANT, Bioinformatics, 2001, vol. 17, nr. 1, pp. 44-57.

[8] Senger, M. et al., X-HUSAR, an X-based graphical interface for the analysis of genomic sequences, Computer Methods and Programs in Biomedicine, 1995, vol. 46, no. 2, pp. 131-142.

[9] Davidson, S. B. et al., BioKleisli: a digital library for biomedical researchers, International Journal on Digital Libraries, 1997, vol. 1, pp. 36-53.

[10] Topaloglou, T. et al., Seamless Integration of Biological Applications within a Database Framework, Proceedings of the Seventh International Conference on Intelligent Systems for Molecular Biology (ISMB'99), Heidelberg, Germany, 1999, pp. 272-281.

[11] Matsuda, H. et al., Querying Molecular Biology Databases by Integration Using Multiagents, IEICE TRANS. INF. & SYST., 1999, vol. E82-D(1), pp. 199-207.

[12] Hofestädt, R. et al, MARGBench - An Approach for Integration, Modeling and Animation of Metabolic Networks, in Proceedings of the German Conference on Bioinformatics (GCB '99), Hannover, Germany, 1999, pp. 190-194.

[13] OMG (Object Management Group), The Common Object Request Broker: Architecture and Specification, OMG Document Number 91.12.1, 1991.

[14] Cattell, R. G. G. (Editor), The Object Database Standard: ODMG-93, Release 1.1, San Mateo, CA, U.S.A.: Morgan Kaufmann Publishers, 1994.

[15] Hofestaedt, R. et al., Information Processing for the Analysis of Metabolic Pathways and Inborn Errors, BioSystems, 1998, vol. 47, pp. 91-102.