CardioVINEdb: a data warehouse approach for integration of life science data in cardiovascular diseases.

Benjamin Kormeier[1,2], Klaus Hippe[1], Thoralf Töpel[1] and Ralf Hofestädt[1]

[1]Bielefeld University, Bioinformatics Department, PO Box 100131, D-33501 Bielefeld, Germany

Summary

One of the major challenges in bioinfomatics is to integrate and manage data from different sources as well as experimental microarray data and present them in a user-friendly format. Therefore, we present CardioVINEdb, a data warehouse approach developed to interact with and explore life science data. The data warehouse architecture provides a platform independent web interface that can be used with any common web browser. A monitor component controls and updates the data from the different sources to guarantee up-to-dateness. In addition, the system provides a ”static” and ”dynamic” visualization component for interactive graphical exploration of the data.

1 Introduction

Large amounts of high dimensional biological data are generated from different high-throughput experiments and from literature. The rapidly growing number of databases and data types poses the challenge of integrating the heterogeneous data, especially in biology. Currently there are about 1170 important molecular biology databases [1].

Thus, the challenge is to capture, model, integrate and analyze the data in a consistent way to provide a new and deeper insight into complex biological systems.

The huge quantity of information generated in life sciences is dispersed in many databases and repositories. Diverse integration approaches for molecular biological data sources have been developed. These systems are based on different data integration techniques. Several systems like Atlas [2], BioWarehouse [3], Columba [4], Systomonas [5] and Reactome [6] have been developed to integrate and present heterogeneous biological data. Some of these systems are not platform-independent and are implemented with different programming languages, and, thus, take more time to install and make it difficult to determine whether data is up-to-date due to lack of logging information. Therefore, the key task is not only to integrate and manage the data scattered in different sources by creating a data warehouse, but also to create a method for visually representing the integrated data in a simple way to understand the underlying biological complexity with ease.

2 Design and Implementation

In cooperation with our partners from medicine and biology we designed a platform-independent data warehouse system that integrates multiple heterogeneous data sources into a local database

[2]To whom correspondence should be addressed. E-mail: bkormeie@techfak.uni-bielefeld.de

doi:10.2390/biecoll-jib-2010-142

enriched with protein microarrays from human smooth muscle cells that related to cardiovascular disease. Based on our VINEdb [8] information system we extended CardioVINEdb with more data sources, better data warehouse infrastructure including monitoring and microarray data. In addition, we upgrade the visualization components and web-pages for better navigation and exploration. To ensure maximum up-to-dateness of the integrated data we developed a data warehouse infrastructure including a monitor component. Furthermore, the common web-based user interface provides a visualization component that allows interactive exploration of the integrated data.

The CardioVINEdb system architecture consists of a 4-layer architecture that is illustrated in Figure 1. The source layer contains the multiple data sources BRENDA, EMBL, GO, IntAct, KEGG, MINT, OMIM, PubChem, SCOP, Transfac, Transpath and UniProt. In addition to the public available databases we integrate experimental microarray data of human smooth muscle cells that are associated to cardiovascular disease. Most of these databases provide parseable flat files that can be processed by our data warehouse infrastructure. A monitor component that is part of the integration layer controls the different data sources. It recognizes changes in the original sources and starts download if the changed file. In a defined cycle the parser will be activated to start the ETL (Extraction-Transform-Load) process. ETL means that the data is extracted form the source data, transformed into the data warehouse schema and loaded into the data warehouse. Data marts for specific analysis applications can easily be constructed by the database layer, i.e. the data warehouse.

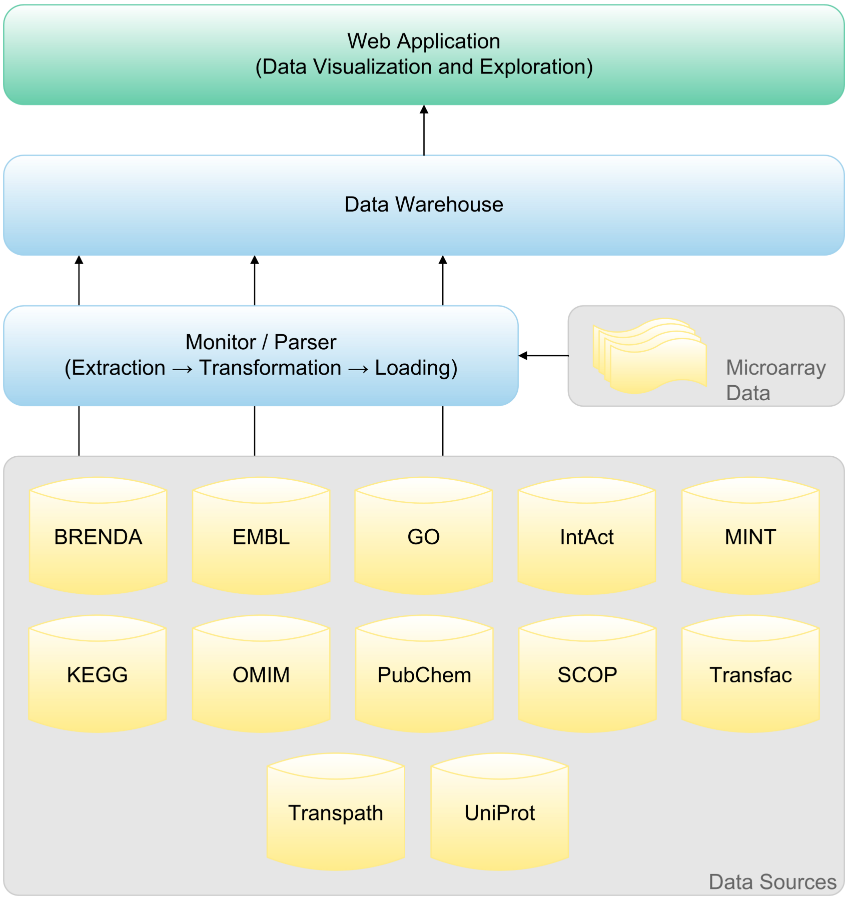
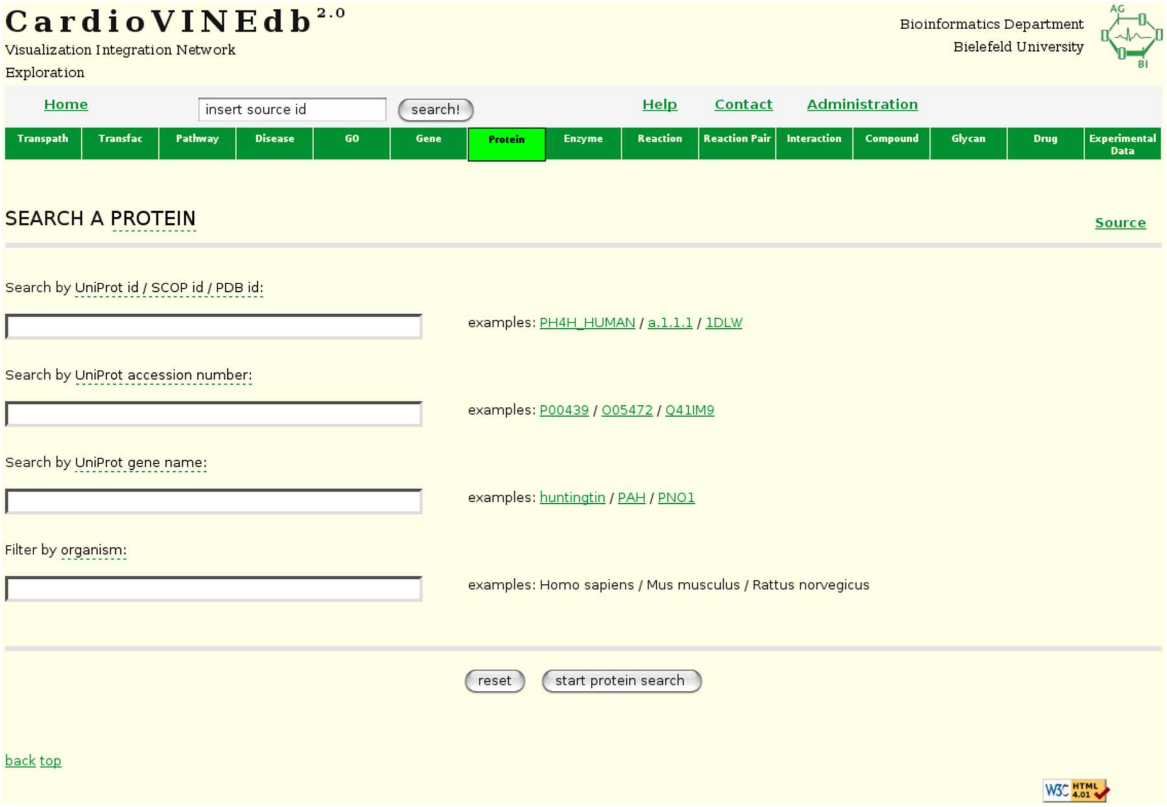


Figure 1: Schematic representation of the CardioVINEdb 4-layer system architecture from the original heterogeneous data sources to the web application layer.

The web-based graphical user interface of CardioVINEdb is implemented with JavaServer Pages (JSP) and runs on an Apache Tomcat web server. A general search engine allows the user to find information of interest spanning multiple domains, such as proteins, enzymes, genes, compounds etc. Additionally, each domain has its own specific search engine to find required information for research. An example of a search form for the *Protein* domain is shown in figure 2. Each data entry has detailed information and a further link to the original data source. After selection of an entry, all information of this entry will be displayed on a new web page as shown in figure 3.



SEARCH A PROTEIN

Search by UniProt id / SCOP id / PDB id:

examples: PH4H_HUMAN / a.1.1.1 / 1DLW

Search by UniProt accession number:

examples: P00439 / O05472 / Q41IM9

Search by UniProt gene name:

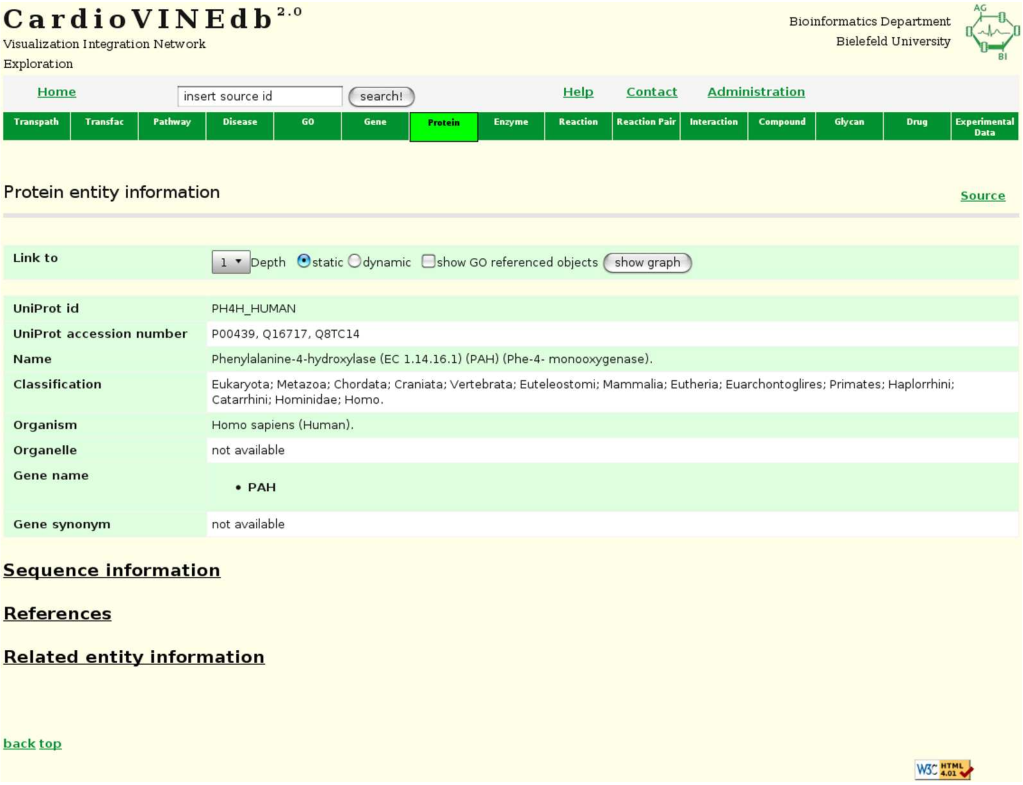
examples: huntingtin / PAH / PNO1

Filter by organism:

examples: Homo sapiens / Mus musculus / Rattus norvegicus

Figure 2: Search form for the *Protein* domain in CardioVINEdb.

The images or the graphical representations of the relationships between the entities of the data warehouse created using JUNG at runtime are dynamic and interactive allowing further exploration. JUNG (`http://jung.sourceforge.net/`) is a Java-based library that provides classes to describe graphs, nodes and edges with additional layout preferences. Thus, a graph is generated by JUNG according to the entities selected by the user. The system produces a PNG image file with the graphical visualization of biological objects in different domains and their linkage. Finally, this image is embedded in the HTML pages and displayed by the web browser. The dynamic component using a Java Applet works in the same manner, but in this case the graph is directly generated and displayed in the applet. For more interactive navigation and exploration the applet has a zoom function, different graph layouts and a picking function to move and select nodes within the graph. Therefore the applet is embedded in the HTML pages and can be displayed by the web browser if Java Runtime Environment is installed on the computer. For instance a search for the protein *PH4H_HUMAN* in depth 1 shows all direct links to enzymes, compounds, drugs etc. in a graph as illustrated in figure 4. Now, if the search goes for depth 2 all links of the enzyme, drug will show up. Hence, it is easy to generate a complex graph with links between the different domains.



Protein entity information

Link to: 1 Depth ● static ○ dynamic □ show GO referenced objects

UniProt id	PH4H_HUMAN
UniProt accession number	P00439, Q16717, Q8TC14
Name	Phenylalanine-4-hydroxylase (EC 1.14.16.1) (PAH) (Phe-4- monooxygenase).
Classification	Eukaryota; Metazoa; Chordata; Craniata; Vertebrata; Euteleostomi; Mammalia; Eutheria; Euarchontoglires; Primates; Haplorrhini; Catarrhini; Hominidae; Homo.
Organism	Homo sapiens (Human).
Organelle	not available
Gene name	• PAH
Gene synonym	not available

Sequence information

References

Related entity information

Figure 3: Detailed result page for the protein *PH4H_HUMAN* in CardioVINEdb merging the data from different sources for that entry.

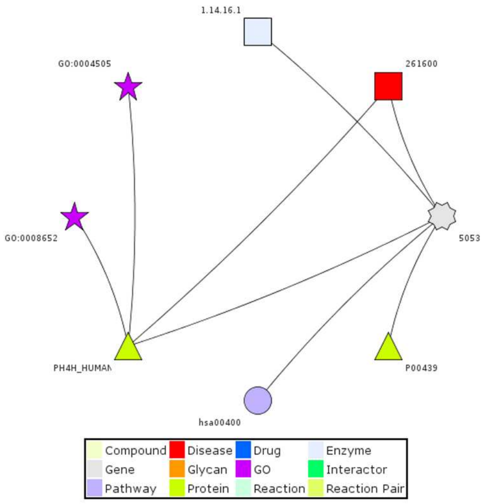


Figure 4: Graph representation of the *PH4H_HUMAN* protein search result with depth 2 (where star symbol represents GO annotations; rectangles are proteins; circles are pahways; red squares are genes; black squares are enzymes).

The database management for integrated data is realized in MySQL using Java Database Connectivity (JDBC). The core of the data warehouse infrastructure, with the name BioDWH [7], is completely implemented in Java, which ensures platform independence of the operating system. Therefore, it could be used separately from the web interface. BioDWH is a bioinformatics data warehouse software kit that integrates biological information from multiple public life science data sources into a local RDBMS. It provides up-to-date integrated knowledge, platform and database independence. This data warehouse infrastructure is available for interested scientific users as a SourceForge project (`http://sourceforge.net/projects/biodwh/`).

3 Summary

A major challenge in life sciences is the integration of heterogeneous data. Apart from the difficulty to facilitate the study of such data within the biological context, a fundamental problem is to represent and make the available knowledge accessible. CardioVINEdb provides integrated data from different popular life science databases and microarray data related to cardiovascular diseases from an EU project in a homogeneous web-based system. The system enables intuitive search of integrated life science data, simple navigation to related information as well as visualization of biological domains and their relationships. Equipped with a monitor component that updates the integrated data in defined update circles, it enables a way for presenting complex biological data in a very user-comprehensible manner. CardioVINEdb is available at `http://agbi.techfak.uni-bielefeld.de/CardioVINEdb/`.

Acknowledgements

This work is supported by Sixth Framework Programme Priority 1 Life Sciences, Genomics and Biotechnology for Health: ”CardioWorkBench - Drug Design for Cardiovascular Diseases: Integration of in Silico and in Vitro Analyses” (Proposal/Contract no.: PL 018671).

References

[1] M. Y. Galperin and G. R. Cochrane. Nucleic Acids Research annual Database Issue and the NAR online Molecular Biology Database Collection in 2009.
Nucleic Acids Research, 37(Database issue):D1-D4, 2009.

[2] S. P. Shah, Y. Huang, T. Xu, M. M. S. Yuen, J. Ling, and B. F. F. Ouellette. Atlas - a data warehouse for integrative bioinformatics.
BMC Bioinformatics, 6:34, 2005.

[3] T. J. Lee, Y. Pouliot, V. Wagner, P. Gupta, D.W. J. Stringer-Calvert, J. D. Tenenbaum, and P. D. Karp. BioWarehouse: a bioinformatics database warehouse toolkit.
BMC Bioinformatics, 7:170, 2006.

[4] S. Tril, K. Rother, H. Müller, T. Steinke, I. Koch, R. Preissner, C. Frömmel, and U. Leser. Columba: an integrated database of proteins, structures, and annotations.
BMC Bioinformatics, 6:81, 2005.

[5] C. C. Choi, R. Münch, S. Leupold, J. Klein, I. Siegel, B. Thielen, B. Benkert, M. Kucklick, M. Schobert, J. Barthelmes, C. Ebeling, I. Haddad, M. Scheer, A. Grote, K. Hiller, B. Bunk, K. Schreiber, I. Retter, D. Schomburg, and D. Jahn. SYSTOMONAS - an integrated database for systems biology analysis of Pseudomonas.
Nucleic Acids Research, 35(Database issue):D533-D537, 2007.

[6] G. Joshi-Tope, M. Gillespie, I. Vastrik, P. D'Eustachio, E. Schmidt, B. de Bono, B. Jassal, G.R. Gopinath, G.R. Wu, L. Matthews, S. Lewis, E. Birney and L. Stein. Reactome: a knowledgebase of biological pathways.
Nucleic Acids Research, 33(Database issue):D428-D432, 2005.

[7] T. Töpel, B. Kormeier, A. Klassen and R. Hofestädt. BioDWH: A Data Warehouse Kit for Life Science Data Integration.
Journal of Integrative Bioinformatics, 5(2):93, 2008.
Online Journal: `http://journal.imbio.de/index.php?paper_id=93`

[8] S. Hariharaputran, T. Töpel, B. Brockschmidt and R. Hofestädt. VINEdb: a data warehouse for integration and interactive exploration of life science data.
Journal of Integrative Bioinformatics, 4(3):63, 2007.
Online Journal: `http://journal.imbio.de/index.php?paper_id=63`