A Unified Integration Model and Database Management System for Genomic Data

Pervaiz Iqbal Khan[2], Muhammad Usman Ghani Khan[1, 2]
Abdul Nasir[2]
[1]Bioinformatics Lab, Al-Khwarizmi Institute of Computer Science
[2]Department of Computer Science and Engineering
University of Engineering and Technology, Lahore

Abstract

Genomic databases are heterogeneous in nature. This means, they have different notations, formats and terminologies for the same kind of data. To overcome the problems due to the heterogeneous and disperse nature of these databases, two main data integration approaches are used i.e. Mediator-based approach and Data warehouse approach. Systems developed using these approaches allow biologists read-only access of data. At times, they need to submit their own generated data during experiments to databases which is not possible with existing systems. In this paper, we design and develop data warehouse based system which not only allows biologists and researchers to access data but also to submit data they generated during lab experiments.

1. Introduction

Database integration has been considered an important area in bioinformatics for many years.Biological databases are heterogeneous in nature i.e. Different databases use different formats and terminologies for the same data which makes understanding of data difficult. Sometimes, there is redundancy and duplication of data across the databases. To solve a particular problem, sometimes, biologistsneed to get data from multiple databasesfor which they need to understand the structure and schema of many databases. This is not easy or sometimes they are not willing to learn schemas. Data integration is a way to somehowovercome these problems.

Mainly, there are two approaches followed in integration for genomic data i.e. Mediator-based approach and data warehouse approach. In mediator based approaches, user submits a single query which is internally divided into multiple sub-queries by the system, and then these sub-queries are transferred to different databases that can fulfill the requests. The result of these sub-queries is integrated and returned to the user as a single result set. In contrast to mediator-based approach, data warehouse approach uses a single physical database for the storage and retrieval of data. First, data is downloaded from the publicly available databases, parsed by different

parsers and then loaded into unified database by loaders. This unified database uses the same name for database attributes which are actually same but different database vendors used different names for them.Both approaches have their own advantages and disadvantages.Advantages of mediator-based approach are: no need to invest in hardware for local data storage, fresh and up to date copy of data is retrieved because query is always made to actual databases. Disadvantages are:overall performance of the system depends upon the speed of internet as well as the number of databases involved in a single query. Not all the databases are query-able in this manner. As data is in read-only format, it cannot be cleanse in case of errors. Advantages of warehouse approach are:single physical database eliminates the need of understanding structure and schema of different databases. Overall performance of the system is improved as data is available in single place. Warehouse systems can be designed in a way that data can be cleaned in case of errors. They can also allow users to query multiple versions of the data. Disadvantages of warehouse approach are: investment is required for the storage infrastructure. Developers of warehouse need to ensure the freshness of data.Data from the publicly available databases first needs to be downloaded locally, which is time consuming and cumbersome.

Data integration systems based on these approaches are discussed in the Literature Review section. These systems only allow biologists to query data. There are times when biologists need to store the data generated during experiments. In this paper, we use warehouse approach to develop a system which not only uniformly stores data from different available genomic databases but also allows biologists to submit their own data. Biologists can query data submitted by other biologists as well but they can modify their own submitted data.

2. Literature Review

Thomas J Lee et al. [1]designed and developedan open source toolkit “BioWarehouse” based on warehouse approach for the integration of heterogeneous biological databases. In BioWarehose, they use single Database Management System (DBMS) for the storage of data from different data sources. They wrote loaders in C and Java languages which convert data from different data sources into format suitable for DBMS schema. Then this formatted data is loaded into DBMS. They stored data of all the versions of dataset which gives users more control over which version they want to query. They used relational DBMS technology for data storage purpose. They storedatabases including ENZYME, KEG, BioCuyc, UniProt, GenBank, NCBI Taxonomy, CMR databases and Gene Ontology. As they used the same schema for all the databases, identifiers of these databases could match which could cause the problems. To avoid these problems, they introduced internal identifier named as warehouse identifiers (WIDs). The usage of single schema eliminated the requirement of learning different database schemas.

Sridhar et al. [2] developed an integration system “VINEdb”for life science data based onwarehouse approach. They integrated KEGG, OMIM, IntAct, GO and UniProt databases.

Their system also provides its users with visualization of integrated data.VINEdb architecture is based on 4-layers. The source layer contains the data sources such as KEGG, OMIM, IntAct, GO and UniProt. Next layer consists of monitor and parser components.Monitor recognizes changes in data sources and downloads them on the local server. Once downloaded, parser parses the files to the format suited for warehouse. After this, data is stored in the warehouse. End-user interacts with data through web application. Web application also allows users visualize the relationship between different data entities.

Atlas, a data warehouse for integrative bioinformatics was developed by Sohrab P Shah et al. [3]. They used relational data model to store heterogeneous sequence data in unified way and provided access to this data through Structured Query Language (SQL) calls implemented in a set of Application Programming Interfaces (APIs). They used Perl, Java and C++ languages to develop loader applications to parse and load data into their relation data model. They built a set of toolbox applications for the retrieval of stored data. They successfully integrated GenBank, RefSeq, Uniprot, Human Protein Reference Database (HPRD), Bimolecular Interaction Network Database (BIND), Database of Interacting Proteins (DIP, IntAct, LocusLink, Molecular Interactions Database (MINT), NCBI Taxonomy, Gene Ontology (GO), OnlineMendelian Inheritance in Man (OMIM), Entrez Gene and HomoloGene.Töpel et al. [4] developed BioDWH, a data warehouse kit for life science data integration. They integrated OMIM, KEGG, UniProt, Brendaand GO databases. [10] According to them, Oracle or MySQL can be used as data model to store integrated data. They used Java for parsers writing which automatically parse and load data into data model. They provided web based interface for the retrieval of data. XML configuration can be used to configure downloading time and other configurations related to the toolkit [11].

Markus Fischer et al. [5] developed an integrated system “DWARF” for proteins, protein sequence and protein structure. They wrote parsers to extractdata from publicly available databases like GenBank, ExPDB and DSSP. The extracted data is stored locally which can be accessed through web interface and direct SQL queries. There are separate interfaces for public access and in-house access.

Trißl et al. [6] developed an integrated database named as “Columba” for proteins, their structure and annotations.Columba integrated twelve different databases of protein including KEGG, PDB, Swiss-Prot, SCOP, Gene Ontology and Enzyme. Open sourced relational data model, PostgreSQL is used as adata model for the storage of data from heterogeneous data sources. There used different sub-schema in Columba for each data source like KEGG, PDB etc. Parsers were written in Perl and Python which convert data from heterogeneous data sources into format suitable for Columba database.They provided web interface for the retrieval of integrated data.

Vera et al. [7] developed open-source Java framework for bioinformatics data integration named as JBioWH. It integratedtwenty public databases including NCBI Taxonomy, GO, Gene, GenBank, KEGG Gene, RefSeq, IntAct, Mint DIP, BioGrid etc.MySQL was used as a data

model for local data warehouse.Java API parser functions were used to retrieve and store data into local data warehouse. It also included client desktop application for non-programmer users to query data. JBioWH also provided command line programs to execute quick queries. According to developers of this framework, it could also be used for high-throughput analyses or CPU intensive calculations. JBioWH also included a global class "JBioWHGraph" which could be used to create graphs from queries. The graph created included set of vertices (containing biological objects) and set of edges describing the relationships between these vertices.

BioDBnet: the biological database network is an application developed by Mudunuri et al. [8] which providedinterconnected access to over twenty biological databases. Integrated databases include Ensembl, Gene, UniProt, RefSeq, Affy, Go etc. Its web interface providedfunctionalities like db2db to convert identifier of one database to other database [9], dbReport to get all identifiers and annotations related to particular input etc.

3. Proposed Work

This section provides detailed discussion related to our proposed work. Figure 1 presents detailed architecture of the methodology. Initially heterogeneous data is downloaded from the publicly available data sources which is stored at local server. Different parsers are written for different data banks to parse and understand the provided data. Based on these parsers output, a unified format is generated which is stored in local database. Finally, a user is provided with a graphical interface to interact with the developed system.

As discussed earlier, we design and develop an integrated database management system to overcome the problems associated with heterogeneous nature of genomic data. We also allow biologists / researchers to store and retrieve the data they generated during experiments. Main components of our system are: i) single physical database; to contain data from diverse databases. ii) Download Service; to automatically download heterogeneous biological databases from the internet. iii) Parsing Service; to parse and load data into unified database. iv) User interface; to retrieve and save biological data. Following is the detail of each component:

i) Unified Database:

We designed a single unified database to store data from different genomic databases. We analyzed the schema and format of different publicly available databases like GenBank, Uniprot and Data Bank of Japan (DDBJ) and then created tables in our unified database to store information of these publicly available databases. We mainly store raw sequences, their authors, total number of Adenine, Cytosine, Guanine and Thymine in a given sequence in case of DNA sequence and Uracil instead of Thymine in case of

RNA sequence. We used a field Global Identifier (GID) which consists of original database name from where data is downloaded, download date and time and auto-generated integer

number to uniquely identify each DNA and RNA sequence in case of duplicate records. We also store the name of original data source of sequences separately.

ii) Download Service:

In order to store data in our local unified database, we need to download it from publicly available databases. One way is to download it manually which requires a lot of human effort and it is also error-prone. We have written a download service which automatically downloads data files from available databases to our local file system without requiring any human effort. This download service usually starts automatically once in two weeks to download data. Once done with all data download, it shuts down. So resources of the server are not wasted. The time duration after which service runs, is configurable so no need to redeploy system if we need to change this duration.

iii) Parsing and Loading Services:

After download of data by download services, we need to store it in our local unified database. As downloaded data is in different file formats and we need to store it in relational database, there needs to be some mechanism to convert those textual files in format suitable for relational database. For this purpose; we have created a parsing and loading service which automatically checks file system after every two weeks and if found any, parses those files and loads them into relational database. These parsers look for the specific tags in data files in order to retrieve data from them. Once parsing is done, all the data files on disk are deleted in order to free unnecessary storage. Like download service, the time duration after which service runs is also configurable. Currently, we have written parsers for .txt, .fasta and .dat file formats. We target parsers for other file formats in future work.

iv) User Interface:

Storing the biological data has no use without any retrieval mechanism. We provide a web based user interface to retrieve the stored data. Biologists and researchers can search data by looking for specific DNA or RNA sequences, by giving a part of the sequence as a search parameter. They can also search sequences by sequence authors, original data source of the sequence from where it is downloaded. We also allow them to download a particular sequence locally for further analysis. As discussed earlier, existing integration systems allow only the retrieval of biological sequence. In this system, we allow biologists and researchers to submit their own generated data during experiments. In order to do so, first they need to create a free of cost user account. Once created, they can submit their generated sequenced data. After submission, they can also modify and delete the data they have submitted. Data submitted by other users is in read only format to them. There is no need for account creation if biologists and researchers need to only read data.

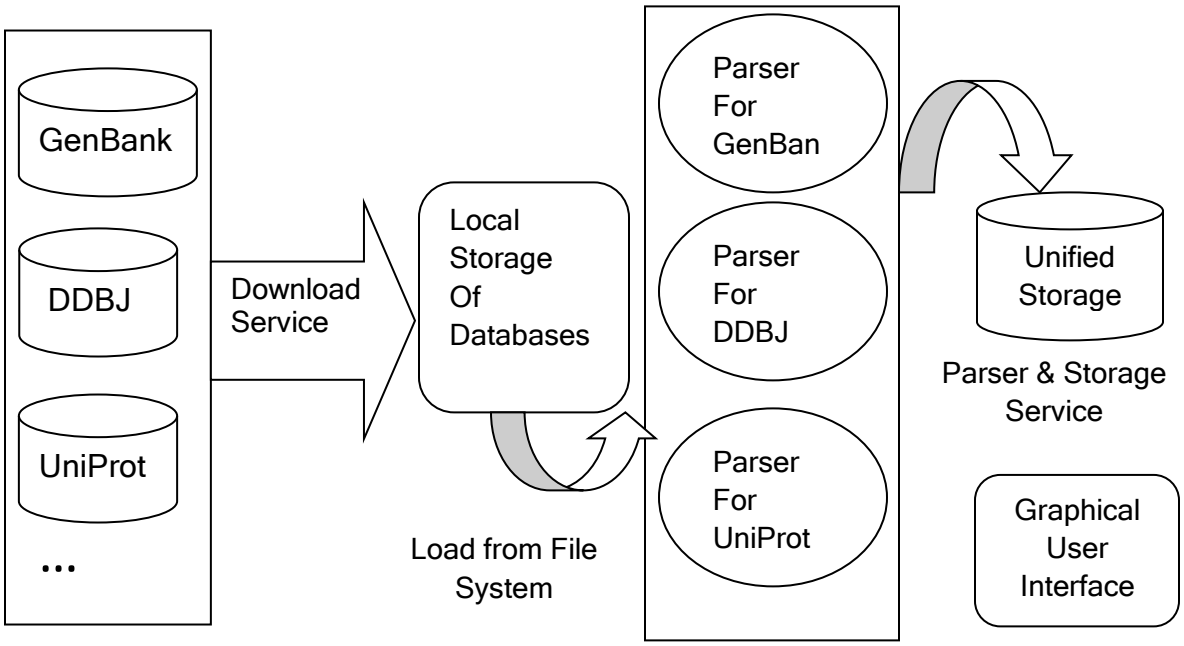

Figure 1: Architecture of the proposed system

4. Conclusion and Future Work

In this paper, we designed and developed a system to provided unified and integrated access of genomic data extracted from heterogeneous. Biologists and researchers can specify different search criteria to retrieve data they are interested in. We also allow them to submit data they created during lab experiments. They can see data from other researcher but can only modify their own generated data. This system bridges the gaps between different researchers by allowing them data sharing at central location. Currently; our system integrates three publicly available databasesi.e.GenBank, DDBJ and Uniprot. We are targeting to add other databases and their parsers to our system as future work.

References

1. Lee, Thomas J., YannickPouliot, Valerie Wagner, Priyanka Gupta, David WJ Stringer-Calvert, Jessica D. Tenenbaum, and Peter D. Karp. "BioWarehouse: a bioinformatics database warehouse toolkit." BMC bioinformatics 7, no. 1 (2006): 170.

2. Hariharaputran, Sridhar, Thoralf Töpel, BjörnBrockschmidt, and Ralf Hofestädt. "VINEdb: a data warehouse for integration and interactive exploration of life science data." Journal of Integrative Bioinformatics 4, no. 3 (2007): 63.

3. Shah, Sohrab P., Yong Huang, Tao Xu, Macaire MS Yuen, John Ling, and BF Francis Ouellette. "Atlas–a data warehouse for integrative bioinformatics." BMC bioinformatics 6, no. 1 (2005): 34.

4. Töpel, Thoralf, Benjamin Kormeier, Andreas Klassen, and Ralf Hofestädt. "BioDWH: a data warehouse kit for life science data integration." Journal of integrative bioinformatics 5, no. 2 (2008): 93.

5. Fischer, Markus, Quan K. Thai, Melanie Grieb, and Jürgen Pleiss. "DWARF–a data warehouse system for analyzing protein families." BMC bioinformatics 7, no. 1 (2006): 495.

6. Trißl, Silke, Kristian Rother, Heiko Müller, Thomas Steinke, Ina Koch, Robert Preissner, Cornelius Frömmel, and Ulf Leser. "Columba: an integrated database of proteins, structures, and annotations." BMC bioinformatics 6, no. 1 (2005): 81.

7. Vera, Roberto, Yasset Perez-Riverol, Sonia Perez, BalázsLigeti, Attila Kertész-Farkas, and SándorPongor. "JBioWH: an open-source Java framework for bioinformatics data integration." Database 2013 (2013): bat051.

8. Mudunuri, Uma, AnneyChe, Ming Yi, and Robert M. Stephens. "BioDBnet: the biological database network." Bioinformatics 25, no. 4 (2009): 555-556.

9. Idrees,M. & Khan, M. SMGCD: Metrics for biological sequence data. Nucleus 51(1): 125-131(2014).

10. Idrees,M.., Khan,M, & Shah. A. Unified Data Model for Biological Data. Mehran University Research Journal of Engineering & Technology 3(3):261-277(2014).

11. Ghani, M. U., Khan, P. I., Asif, K. H., Nasir, A., Arshad, M. J, & Amanat. S. An Agent-based CBIR system for medical images. Journal of faculty of Engineering &Technology 21(2):39-45(2014).