ANNODA: Tool for integrating Molecular-biological Annotation Data

Supawan Prompramote Yi-Ping Phoebe Chen
School of Information Technology
Faculty of Science and Technology
Deakin Univeristy, Melbourne, Australia
{sprom,phoebe}@deakin.edu.au

Abstract

Collecting, analyzing, and making Molecular-biological annotation data accessible in different public data sources is still an ongoing project. Integration of such data from these data sources might lead to valuable biological knowledge. There are numerous annotation data and only some of those are structured. The number and contents of related sources are continuously increasing. In addition, the existing data sources have their own storage structure and implementation. As a result, these could lead to a limitation in the combining of annotation. Here, we proposed a tool, called ANNODA, for integrating Molecular-biological annotation data. Unlike the past work on database interoperation in the bioinformatics community, this database design uses web-links which are very useful for interactive navigation and meanwhile it also supports automated large-scale analysis tasks.

1. Introduction

Typically, the knowledge of molecular and biological objects is defined by various data. An object from one data source is encoded by information in other sources. This is commonly called annotation. There are many public data sources that collect, analyze and make molecular and biological annotation data accessible [13]. Establishing such an integrated data source for the published annotation data would give a number of benefits:

- An integrated data source will create the ability to build up progressively detailed annotation data and will give access to this information to third parties.
- It will facilitate the cross-validation of data obtained by different data sources, to characterize various techniques and to establish benchmarks and standards.
- It will enable bioinformatics groups possibly not directly related to annotations, to participate in the data analysis and to develop new methods and tools for such analysis.
- It will promote a public sharing culture for these crucial data.
- It might lead to valuable biological knowledge

As a result, we have proposed the Tool for integrating Molecular and biological annotation data called ANNODA. Unlike the past work on database interoperation in the bioinformatics community, this database designs to meet the following requirements.

- A consistent view of annotation data to users
- A new annotation data source should be plugged in as it comes into existence
- Serving a query optimisation across multi-system
- Provide an interactive navigation
- A system should support automated large-scale analysis tasks
- Resolve the semantic conflicts and contradictions

The structure of this paper is organized as follows. Related works is described in Section 2. Section 3 describes overview of ANNODA architecture. Section 4 explains ANNODA query processing. Section 5 describes comparative discussion. Finally, Section 6 is the conclusion

2. Related works

The past work on database interoperation in the bioinformatics community has taken one of four technical approaches: 1) Hypertext navigation, 2) Data warehousing, 3) Unmediated multidatabase queries, and 4) Federated databases.

The first method, Hypertext navigation, also called the indexed data sources approach, allows the users to interactively navigate from a result of one query in one member database to a result in another database, by

Proceedings of the 21st International Conference on Data Engineering (ICDE '05)
1084-4627/05 $20.00 © 2005 IEEE

using indexes and links between those two databases. This approach achieves a basic level of integration with minimal effort; however, it neither provides a mechanism to directly integrate data from relational databases nor to perform data cleansing and transformation for complex data mining. The representative systems of this method include Entrez by NCBI [8], which is a search and retrieval system that integrates information from databases at NCBI (National Center for Biotechnology Information) through PubMed [6] and SRS [14], which provide an integrated browser interface and a basic query language for a range of important information sources.

Secondly is the Data warehousing approach, in which the data from a set of heterogeneous databases are exported into a single database, called the data warehouse. Translators are needed to transform this exported data, which in different in format and conceptualisation, into the format and conceptualisation of the warehouse database. This method simplifies the access to and query of data, allows for automated data mining, and quickly extends as new data sources are added, without effecting the original data source applications. On the other hand, the extraction, cleaning, transformation, and loading process can take considerable time and effort, which is a major drawback of Data warehousing. TSIMMIS [14] and DataFoundry [11] are examples of data warehousing system.

Next approach is Unmediated multidatabase queries. In this, the users are allowed to construct complex queries that are evaluated against multiple heterogeneous databases. Generally, a query is comprised of both a set of databases that it applies to and the tables as well as attributes (or classes and entities) that are to be queried within each database. This approach provides the format and access transparency, while it lacks the schema transparency and reconciliation. CPL/Kleisli [18] project is a representative system, which provides integrated access to multiple data sources, but does not present an integrated schema across the source databases. Hence, the users are required to directly specify the rules and constraints involved in queries of integrated access to the databases. That might imply that only users who are familiar with the details of the individual data sources can fully utilize the resource.

The last approach, Federated databases, resembles the two previous mentioned approaches. It is similar to data warehousing method in that it requires mapping between a single federated schema and the schemas of the member databases. Like the Unmediated multidatabase Queries approach, a set of member databases are not physically integrated within one single database management system. The federated approach is the traditional approach within the computer-science community owing to many reasons, such as an easily understandable architecture, and a basic level of integration with minimal effort. However, surprisingly, it has received little attention in the bioinformatics community. One explanation is that the federated approach is too complex to implement. The EasyQuery program [18] from CyberConnect, and P/FDM [15] are representative systems that have applied the federated database method

There are many other data integration systems that have applied one of the four technical approaches to interacting with scientific data, for example, the Object Protocol Model (OPM) [10], and TAMBIS [21]. However, none of these systems have integrated information with a strong consideration of taking into account semantic conflict, interactive navigation, and automated data analysis.

3. Overview of ANNODA

ANNODA architecture is mainly based on our proposed information management for microarray experimental data [22][24] More details are discussed on the following subsections.

3.1. System architecture design

In ANNODA, we desire a system that will meet the following requirements. Firstly, the system should provide a consistent view of annotation data to the user. It will allow a user to pose a single query, and to receive a single unified answer. Secondly, a new relevant data source should be wrapped and plugged in as it comes into existence. Thirdly, the system should serve a query optimization across multiple systems. Fourthly, the system should support automated large-scale analysis tasks. Lastly, the system should resolve the semantic conflicts and contradictions caused due to the unstructured of annotation data.

To serve these requirements, the system architecture will be based on a federated database approach. The ANNODA global schema is obtained to represent a virtual database, combining annotation data from each participating annotation source to form a single, consistent representation. Queries posed against the ANNODA global schema will be translated into individual queries against the relevant annotation databases, and their results combined before being returned to the user. To address semantic conflicts and contradictions, we modified our proposed matching method called MDSM: Microarray Database Schema Matching by using Hungarian Method [23] to map the object correspondences.

The capability of adding the new annotation data sources can be performed via two main steps: 1) mapping new annotation data source to the ANNODA global schema by using the mapping rules, transformation, and database descriptions, 2) creating the mediator interface to a new annotation data source.

3.2. System components

The components of ANNODA can be seen in Figure 1. Mainly, it consists of three parts: Wrappers, Mediator, Mapping module. Details of these components can be found in prior works [22][23][24].

We describe here two important components: ANNODA-OML local model and ANNODA-GML global model. Both ANNODA-OML and ANNODA-GML are expressed by the Object Exchange Model [20].

3.2.1. The Object Exchange Model (OEM). OEM is very simple, while providing the expressive power and flexibility needed for integrating information from disparate sources. OEM is simpler than conventional object models, but it does support the two key features required by object models: *object nesting* and *object identity*. Our primary reason for choosing a very simple model is to facilitate integration. The simple data models have an advantage over complex models when used for integration, since the operations to transform and merge data will be correspondingly simpler. In addition, [7][9][16][20], indicated that Semi-structured data model is the most effective and appropriate to be used to create global data model. OEM is a data model particularly useful for representing semi-structured data. The above discussion is a major support for choosing OEM to express our model.

Data represented in OEM can be thought of as a graph, with objects as the vertices and labels or attributes as the edges. For simplicity, when comparing the object's value, we extended the data type of the object's value into OEM. In the OEM data model all entities are objects. Each object has a unique *object identifier* (*oid*). Some objects are atomic and contain a value from one of the disjoint basic atomic types (e.g. integer, real, string, gif, etc). All other objects are complex; their value is a set of *object references*, denoted as a set of *(label, oid, type)* pairs.

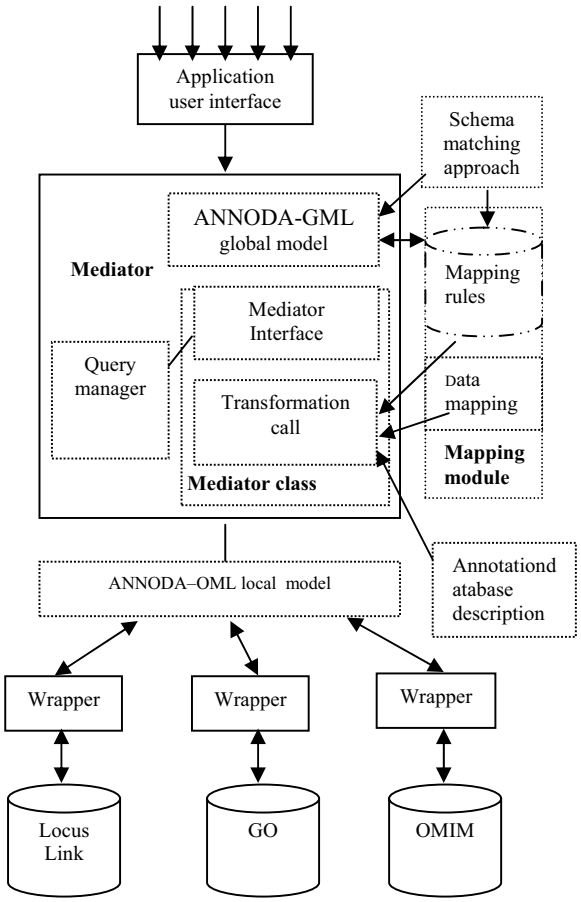

Figure 1. Architecture of ANNODA: Integrated tool for annotation data

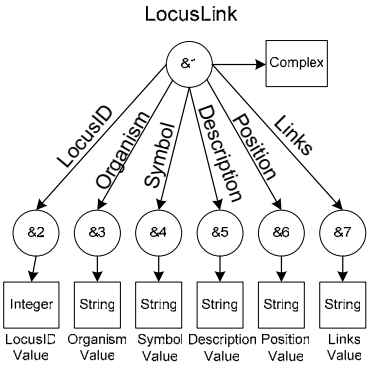

Figure 2. ANNODA-OML represents a fragment of LocusLink data model

3.2.2. ANNODA-OML. To match relevant data sources, they need to be expressed in the same model. As a result, we import these participating data sources into a common model called ANNODA-OML. Figure 2 shows an example of using ANNODA-OML to express the structure and contents of LocusLink. The oval shapes correspond to the objects, edges correspond to model attributes, and rectangular shapes correspond to object's types.

Really, we describe the structure and contents of LocusLink as shown in Figure 3.

```
LocusLink, &1, Complex, {LocusID, Organism,
                         Symbol, Description,
                         Position, Links}
  LocusID, &2, Integer, "LocusID Value"
  Organism, &3, String, "Organism Value"
  Symbol, &4, String, "Symbol Value"
  Description, &5, String, "Description Value"
  Position, &6, String, "Position Value"
  Links, &7, String, "Links Value"
```

Figure 3. ANNODA-OML representation of the structure and contents of LocusLink

Each line shows label, object's oid, object type, and object value. If the object is atomic, its value is given on that line. If the object is complex, and has not been described earlier, subsequent indented lines describe its object references. For example, LocusLink is a Complex object with oid &1. It consists six object references: {LocusID, Organism, Symbol, Description, Position, and Links}. Whereas, LocusID is an atomic object of type Integer with oid &2 whose value is 'LocusID Value'.

3.2.3. ANNODA-GML. ANNODA aims at providing visual interface for complex query. Users access the underlying annotation data sources indirectly through a global model (view), which has been constructed either from the local relevant models or from general knowledge of the domain. Our ANNODA global model is called ANNODA-GML, described by OEM.

Based on federated database approach, ANNODA-GML does not require a number of participating data sources to be physically integrated into a single database. ANNODA-GML acts as a navigator, which evaluate complex queries against multiple local data sources. As a result, ANNODA-GML requires mapping between the members of local data sources to a global model. We described in [23] how the mapping task was accomplished.

Figure 4 shows the key elements of ANNODA-GML global data model.

4. Query Processing

4.1. Query language

To answer the user questions, complex queries were expressed in Lorel language [7]. Lorel is a language, which were designed for querying semi-structured data. Semi-structured data is becoming more and more prevalent when performing simple integration of data from multiple sources. Traditional data models and query languages are inappropriate, since semi-structured data often is irregular. For example, some data is missing, similar concepts are represented using different types, heterogeneous sets are present, and object structure is not fully known. Lorel is user-friendly language in the SQL and OQL style for effectively querying such data. A *select-from-where* query in Lorel has the same semantics as a *select-from-where* query in SQL or OQL. In Lorel, the result is always a collection of OEM objects, and duplicate elimination is by *oid*.

Similar to SQL, each assignment of the variables in the *from* clause that passes the condition of the *where* clause, a value is generated according to the expressions in the *select* clause. Each of these values is then coerced into an OEM object. The coercion may result in the creation of new objects and edges in the OEM graph. Thus the query results may refer to either original database objects or new objects created by the coercion. For instance, the query:

Select X
From ANNODA-GML
Where Source.Name = "LocusLink"? X

would generate the following answer object:

answer &442
  SourceID &103
  Name &104
  Content &105
  Structure &106

The object &442 is a new object, which can be reused in later queries. Note that renaming is necessary so that answer is not overwritten.

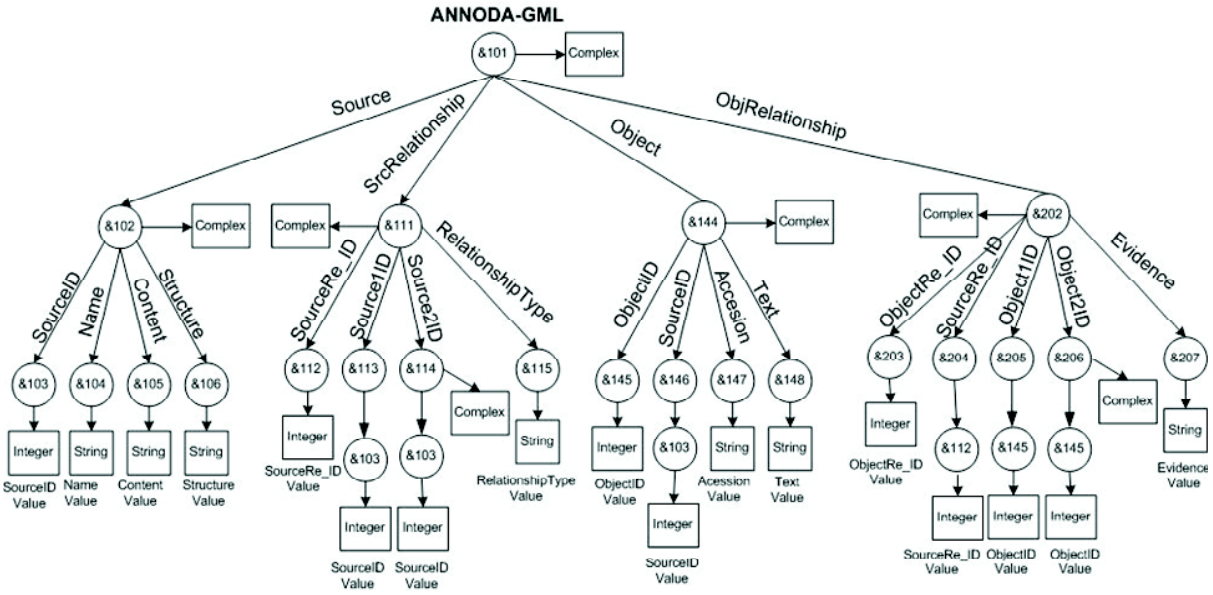

Figure 4. The ANNODA-GML data model

4.2. Interactive Query Interface

ANNODA provided a single access point for users to pose queries and retrieve annotations for a set of given objects from the particular sources. To use the system, users do not need detailed knowledge of computing and data management. Users can describe a query in biological question, not in SQL. The result of a query against multiple local annotation data sources is re-organized and can be used for further computation. Currently, we experimented with 3 annotation data sources, LocusLink [4], Go [3], and OMIM [5].

First, the user can specify either inclusion or exclusion of the target of interest from available sources. Second, the method for combining the selected mapping is specified. Third, some specific search conditions are provided to narrow the search results. The ANNODA query interface can be seen in Figure 5(a).

ANNODA is capable of addressing complex queries explained in biological questions. For example, figure 5(b) shows the results of the query in the form "*Find a set of LocusLink genes, which are annotated with some Go functions, but not associated with some OMIM disease*". In addition, the user can retrieve information of the particular object by following the provided web-links (as shown in Figure 5(c)).

5. Comparative discussion

In the biological domain, most integration systems are currently based on the query-driven approach. SRS, BioNavigator [1], K2/Kleisli [18], and DiscoveryLink [2] are representatives of this class. Although they differ in the capabilities they offer, they can be considered *middleware systems*, in which the bulk of the query and result processing takes place in a different location from where the data is stored. For example, K2/Kleisli, and DiscoveryLink use source-specific data drivers (or called wrapper) for extracting the data from the underlying data sources including application programs. The extracted data is then shipped to the integration system, where it is represented and processed using the data model and query language of the integration system (e.g. the object-oriented model and OQL in K2/Kleisli). Biologist access the integration system through a client interface, which hides many of the source-specific details and heterogeneities.

The data warehousing architecture looks similar to the query-driven integration architecture, except for the addition of a repository. This repository is used by the integration system to store (materialize) the integrated views over the underlying data sources. Instead of answering queries at the source, the data in the warehouse is used.

Despite the advancements in biological database systems research, we argue that current systems present biologists with only an incomplete solution to the growing data management problem they are facing [HS03, technical report].

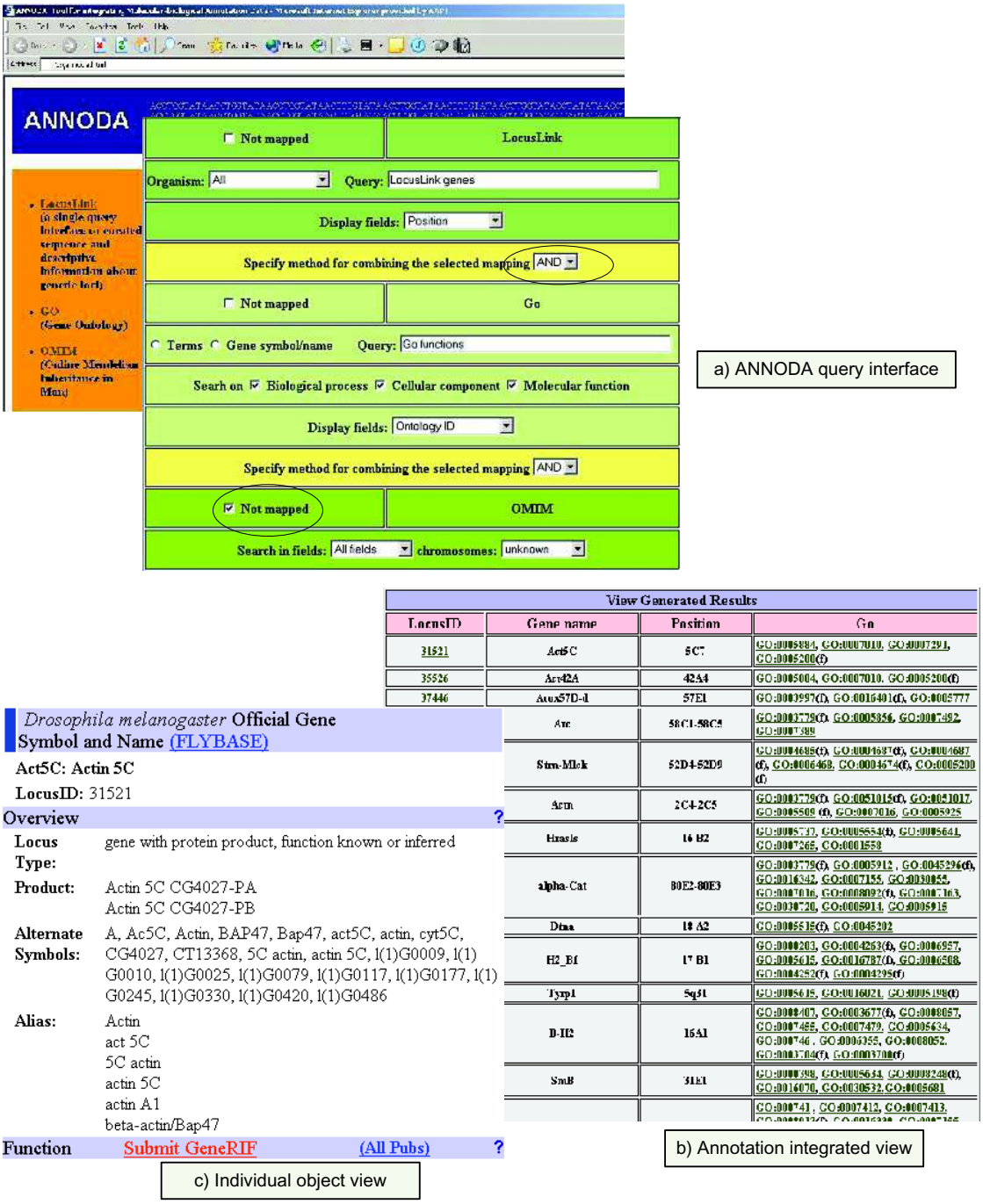

Figure 5. (a) ANNODA query interface, (b) Annotation integrated view for a set of LocusLink genes, which are not annotated with GO functions, but not associated with some given OMIM diseases, and (c) Individual object view.

Proceedings of the 21st International Conference on Data Engineering (ICDE '05)
1084-4627/05 $20.00 © 2005 IEEE

Table 1. The comparison of ANNODA with other existing integration systems

	K2/Kleisli [18]	**DiscoveryLink [2]**	**GUS [12]**	**ANNODA**
The heterogeneity of available data repositories	User shielded from source details	User shielded from source details	User shielded from source details	User shielded from source details
Missing standards for data representation	Global schema using object-oriented model	Global schema using object-oriented model	GUS schema based on relational model; OO views	Global schema using semistructured model (translated to OO model)
Multitude of user interfaces	Single-access point	Single-access point	Single-access point	Single-access point
Quality of user interfaces	Not a use level interface	Require knowledge of SQL	Require knowledge of SQL	Require Biological terms and knowledge; No require knowledge of SQL
Quality of query languages	Comprehensive query capability	Comprehensive query capability	Comprehensive query capability	Comprehensive query capability
Limited functionality of microarray repositories	New operations on integrated view data	New operations on integrated view data	New operations on warehouse data	New operations on integrated view data
Format of query results	Re-organization of result possible	Re-organization of result possible	Re-organization of result possible	Re-organization of result possible
Incorrectness due to inconsistent and incompatible data	No reconciliation of results	No reconciliation of results	Data in warehouse is reconciled and cleansed	Reconciliation of results
Uncertainty of data	No provision for dealing with uncertainity in data	No provision for dealing with uncertainity in data	No provision for dealing with uncertainity in data	No provision for dealing with uncertainity in data
Combination of data from different microarray repositories	Results integrated using global schema; source wrapper needed	Results integrated using global schema; source wrapper needed	Query results are integrated	Results integrated using global schema; source wrapper needed
Extraction of hidden and creation of new knowledge	Not supported	Not supported	Annotations supported	Annotations supported
Low-level treatment of data	Not supported	Not supported	Not supported	Supported (Self-describing model)
Integration of self-generated data and extensibility	Not supported	Not supported	Supported	Supported
Integration of new specialty evaluation functions	Not supported	Not supported	Not supported	Supported
Loss of existing repositories	No archival functionality	No archival functionality	Archiving of data supported	Not supported

We briefly compare ANNODA with three other integrated systems, namely, K2/Kleisli, DiscoveryLink and GUS, which were somewhat comparable to what has been done in our work. That is, those three systems focus on using global schema for obtaining the query results. In addition, two of those (K2/Kleisli and DiscoveryLink) provide an interactive navigation. Table 1 shows a summary of key aspects and evaluations of K2/Kleisli, DiscoveryLink, GUS, and ANNODA.

6. Conclusion

We presented the ANNODA tool for integrating Molecular-biological Annotation Data. The system architecture is based on Federated Information Systems (FIS). The main components of this Information Systems (IS) are: *Wrappers*, *Mediators*, and *Mapping module*. We use a common data model called ANNODA-OML to uniformly represent models from different annotation data sources. A global model (view), called ANNODA-GML is then constructed both from the local relevant models and from general knowledge of the domain.

The system allows scientists to easily retrieve and query data derived from different public annotation data sources. This will facilitate the exchange of information amongst researchers. Without knowledge of computing and data management, the users can construct the complex queries in biological questions. ANNODA system met the design requirements, especially using web-links to provide an interactive navigation and supporting automated large-scale analysis tasks.

As regards the future research, we will focus on the following issues:

- The larger and more variety of molecular and biological data models will be integrated to evaluate our proposed ANNODA.
- The query specification and interface will be taken into account.
- The new approaches of query optimization across multi-systems will be investigated to improve ANNODA query performance.
- Re-Organization of the retrieved results will be mainly focused on to facilitate the further analysis.

7. Acknowledgments

The work in this paper was partially supported by Grant DP0559251 and CE0348221 ARC Centre in Bioinformatics from the Australian Research Council.

8. References

[1] Bionavigator: http://www.bionavigator.com
[2] DiscoveryLink: http://www.ibm.com/discoverylink
[3] GeneOntology: http://www.geneontology.org/
[4] LocusLink: http://www.ncbi.nlm.nih.gov/Locuslink
[5] OMIM: http://www.ncbi.nlm.nih.gov/omim
[6] Pubmed: http://www.ncbi.nlm.nih.gov/pubmed
[7] Abiteboul, S., D. Quass, J. McHugh, J. Widom, and J.L. Wiener. The Lorel Query Language for Semistructured Data. *International Journal on Digital Libraries*, 1(1):68--88, 1997.
[8] Benson, D. A., M. Boguski, D. J. Lipman, and J. Ostell, Genbank. *Nucleic Acids Research,* 22:3441-3444, 1994.
[9] Chawathe, S., H. Garcia-Molina, J. Hammer, K. Ireland, Y. Papkoastantinou, J. Ullman, and J. Widom. The TSIMMIS project: Integration of Heterogeneous Information Sources. *Proceedings of the ISPJ Conference,* 1994.
[10] Chen, A., and V. Markowitz. An overview of the object protocol model (OPM) and the OPM data management tools. *Inform. Syst.,* 20(5), 1995.
[11] Critchlow, T., K. Fidelis, M. Ganesh, R. Musick, and T. Slezak, *IEEE Transactions on Information Technology in Biomedicine*, 4(1): 52-57, 2000.
[12] Davidson, S., J. Crabtree, B. Brunk, J. Schug, V. Tannen, C. Overton, and C. Stoeckert. K2/Kleisli and GUS: Experiments in integrated access to genomic data sources. *IBM Systems Journal*, vol. 40, pp 512-531, 2001.
[13] Do, Hong-Hai; Rahm, Erhard. Flexible Integration of Molecular-biological Annotation Data: The GenMapper Approach, Proc. EDBT 2004, Heraklion, Greece, Springer LNCS, March 2004
[14] Etzold, T. and P. Argos, SRS – an indexing and retrieval tool for flat file data libraries. *Computer Applications in the Biosciences,* 9(1): 49-57, 1993.
[15] Garcia-Molina, H., J. Hammer, K. Ireland, Y. Papakonstantinou, J. Ullman, and J. Widom, Integrating accessing heterogeneous information sources in TSIMMIS. *Proc. AAAI Symp. Information Gathering*, Stanford, CA, pp. 61-64, 1995.
[16] Goodman, N., S. Rozen, and L. Stein. Requirements for a deductive query language in the MapBase genome-mapping database. *Proceedings of Workshop on Programming with Logic Databases,* Vancouver, BC, October, 1993.
[17] Kemp, G., and P. Gray, Using the Functional Data Model to Integrate Distributed Biological Data Sources. In *P. Svensson and J. French, editors, Proc. SSDBM:* 176-185. *IEEE Press,* 1996.
[18] Overton, G. C., S. B. Davidson, and P. Buneman, Database transformations for biological applications. In *DOE HGP Contractor-Grantee Workshop VI* Santa Fe, NM, 1997.
[19] Shin, D. G., *et al*., Graphical *ad hoc* query interfaces for Federated Genome database, Computer Sc. & Eng. U of Connecticut. In *Storrs CT DOE HGP Contractor-Grantee Workshop VI,* Santa Fe, NM, 1997.
[20] Papakonstantinou, Y., H. Garcia-Molina, and J. Widom. Object exchange across heterogeneous information sources. *Proceedings of the Eleventh International Conference on Data Engineering,* pp. 251-260, Taipei, Taiwan, March, 1995.
[21] Paton, N.W., R. Stevens, P. Baker, C. A. Goble, S. Bechhofer, and A. Brass, Query Processing in the TAMBIS Bioinformatics Source Integration System. In *Proc. SSDBM*: 138-147. *IEEE Press,* 1999.
[22] PROMPRAMOTE S., CHEN, Y.-P. P. and MAIRE F. , Information Management for Microarray Experimental Data, 5th IFAC Symposium on Modelling and Control in Biomedical Systems, IFAC2003, Elsevier Science, pp377-382, 2003.
[23] PROMPRAMOTE S., CHEN, Y.-P. P., MDSM: Microarray Database Schema Matching Using Hungarian Method, Submitted to Information Science, 2004.
[24] PROMPRAMOTE S., CHEN, Y.-P. P., Managing Microarray Data in Bioinformatics, Technical Report, 2004.