GlycoVault: A Bioinformatics Infrastructure for Glycan Pathway Visualization, Analysis and Modeling

S. Nimmagadda[1], A. Basu[1], M. Evenson[1], J. Han[1],
M. Janik[1], R. Narra[1], K. Nimmagadda[1], A. Sharma[1],
Krys J. Kochut[1], John A. Miller[1] and William S. York[2]
[1]Computer Science Department and
[2]Complex Carbohydrates Research Center
University of Georgia
Athens, GA 30602

November 9, 2007

Abstract

Glycan biosynthesis is the process by which glycans are built in the cells. Information about the process is available from several Web accessible databases, published papers, experimental data produced at the Complex Carbohydrates Research Center (CCRC) as well as the knowledge accumulated by the biologists working at CCRC. The principal purpose of Glyco-Vault is to support the research of glycobiologists in collecting and analyzing data about glycans, such as the changes in their abundance level over a cell's life cycle as well as their correlation with gene expression levels for proteins that serve as enzymes in the biosynthetic process.

In order to support glycan biosynthesis pathway visualization, analysis and modeling, GlycoVault has been collecting data and knowledge from a variety of sources. Traditional, bioinformatics resources have been provided by relational databases. Some new research efforts have proposed replacing relational databases with ontologies. In GlycoVault, we are utilizing both technologies to supply an integrated bioinformatics infrastructure that can serve as a resource to both end users as well as other programs and applications.

1   Introduction

Rapid growth in the use and management of scientific data and knowledge is becoming more important as both are experiencing rapid growth along with greater importance in sharing this information particularly through the Web. The management of experimental scientific data has been greatly facilitated by database management and to a lesser degree, by spreadsheet technology. Knowledge management, that was largely developed by the AI Community, is becoming more mainstream due to the Semantic Web initiative along with advances in text mining. Modern scientific laboratories are highly dependent on the management of both data and the knowledge it infers.

There are three obvious ways to meet this challenge: One could rely on object-relational databases and encode the knowledge into relational tables, as is done in the ApiDB federated database [1]. This database has a subschema called SRES that maintains ontological information. Unfortunately, some of the richness of the relationships and constraints are lost by doing this. Alternatively, one could develop an ontology and populate it with all of the data that would be conventionally stored in relational tables, as is the case in [2]. Since experimental data is often tabular, relational databases are a natural fit, not to mention their obvious performance advances when the amount of data is large. The middle ground tries to utilize the best of both technologies in an integrated fashion.

In this paper, we briefly discuss these options and indicate why we chose the middle ground. We then describe the design and implementation of our proto-

type called GlycoVault. Two important applications of GlycoVault are also outlined.

The primary goal of GlycoVault is to provide a bioinformatics infrastructure to support research in Glycobiology. In particular, it provides a foundation for the integration and visualization of knowledge and data. An appropriate visual juxtaposition of knowledge with relevant experimental data can assist in understanding of complex biochemical processes and in hypothesis formulation. Visualization is also essential for assisting in the human curation needed in ontology population. A secondary goal of GlycoVault is to provide a foundation for the development of predictive models of biochemical processes. The knowledge provided by GlycoVault can be used by applications that assist the human designers in creating or customizing computer models/simulations. For example, the data provided by GlycoVault as well as other data sources can be used to parameterize physical models. Appropriate use of data mining and text mining for model calibration, although far from easy, can be very useful. Both scientific knowledge and experimental data will be needed for model validation and to resolve conflicts between existing knowledge and model predictions. Both data and knowledge are essential, but their interplay is really where the productivity as well as the research challenges are found.

GlycoVault provides a means of storing and retrieving data to support glycomics research at the Complex Carbohydrates Research Center (CCRC) at the University of Georgia. These data include quantative Real-Time Polymerase Chain Reaction (qRT-PCR) data as well as basic glycomics data, such as biologically relevant parameters and various types of data collected by the analytical services component, along with the explicit and implicit knowledge required to analyze and interpret these data. GlycoVault consists of databases, ontologies, and data files in various formats that are integrated by a sophisticated organizational structure and accessed by a comprehensive, yet easy to use Application Programming Interface (API). The API facilitates the development of methods for querying the knowledge and exporting the results in formats (such as XML) that can be readily digested by external applications. Thus, GlycoVault not only provides scientists within the Center with a robust means of retrieving and analyzing their results, it provides an Internet gateway for pub-

lication of the knowledge and data collected by the Center, including results obtained for diverse samples provided by scientists who use the analytical services.

The rest of this paper is organized as follows: In section 2, we highlight the types of data and knowledge that are stored in GlycoVault as well as the integration challenges they present. Part of the solution to providing integration or interoperability, is to support effective and extensible translations as outlined in section 3. Section 4 gives the system architecture of GlycoVault including how it is populated and accessed, both locally and remotely. Storage and query processing efficiencies are also discussed. Section 5 highlights two applications we are developing that utilize GlycoVault as the principal part of their infrastructure. Finally, conclusions and future work are given in section 6.

2 Diverse Data and Knowledge

A novel feature and research challenge of GlycoVault is the co-existence, unified management and interoperation of the diverse data and knowledge in a secure, yet accessible manner. In GlycoVault, much of the data and knowledge is stored in one of the following forms: Spreadsheets, Relational Tables, XML, RDF [3] or OWL. Three query languages (SQL, SPARQL and XQuery) are used to access the information, but a more basic file interface is also be provided. Using software that we developed, along with a large amount of open source software (e.g., Jena, ARQ, D2RQ, Apache POI), we provide multifaceted access to the diverse types of data and knowledge in GlycoVault. For example, SPARQL [4] may be used to access OWL [5], RDF and relational tables, while spreadsheets and relational tables can be accessed using SQL. We plan to identify the languages that are most suitable for our domain and extend them to provide additional access capability across various data protocols. These extensions will allow GlycoVault to provide rich support for Web services as well as custom interfaces for data retrieval, browsing and processing, thereby providing the basis for applications such as visualization via our Glycomics browser (GlyB) as well as predictive modeling and simulation, see section 5.

Spreadsheet files provide a convenient way to record and view experimental data. GlycoVault includes

several of the currently popular data models for persistent storage. Object-relational storage and the SQL query language are provided by Oracle 10g. Object-oriented storage and retrieval are provided by Java objects and the Java Persistence Architecture (JPA). Semistructured data in the form of XML documents provides a means for storing information about chemical structures. As data and information accumulates, is refined and generalized, it should be combined with existing knowledge. In the past, this usually meant publishing papers. More and more ontologies are being used to maintain this knowledge for both humans and machines to use. These five types of stores are used in conjunction to drive the production of more refined and useful information.

- *Experimental Data in Spreadsheets.* As raw data is processed into more refined data, it is recorded in spreadsheet files. Due to the size and number of such files, GlycoVault is used to keep track of these files as well as any backups of them. It is important to know the content and format of the data files for both humans and computer applications to use them effectively. Therefore, we store basic meta-data about these spreadsheets. This allows for rapid programmatic access as well as the maintenance of provenance information. This also opens up the possibility of semantic annotation of experimental data [6].

- *Processed Data in Object-Relational Databases.* Some of the more important data in spreadsheets will be transformed, integrated and stored in a relational database. The principal purpose of storing this type of information is to make refined experimental data readily available to biologists. It can be conveniently accessed using the SQL query language. For less sophisticated users, Web forms can be used to access this data.

- *Multifaceted Data in Object-Oriented Stores.* Persistent object storage can be used as a hub for the other stores. One such store, called the Glyde Object Model (GlyOM), is particularly useful. It is used for creating and storing complete information on chemical structures to facilitate the translation between XML representations and ontological representations.

- *Structural Information in XML Documents.* As XML has become the de facto standard for data interchange, will need to store XML documents such as Glyde-II structure specifications. These documents may be read by XML parsers or queried using XQuery [7].

- *Domain Knowledge in RDF/OWL Stores.* A major aim of our project is to create populated ontologies to capture domain knowledge relevant to the project. For example, the GlycO ontology [8] classifies and maintains glycan structures as well as structural constraints via GlycoTree and basic knowledge about biosynthetic pathways. We store OWL ontologies in Oracle using Jena [9] and for performance enhancement cache relevant portions in BRAHMS [10].

3 Interoperability Support via XML Translations

One purpose of GlycoVault is to serve as a clearinghouse for Web artifacts. As such, it provides translations between the following standards for representing chemical structures: Glyde-II, GlydeCT and Linucs. It also supports the translation between the GlycO ontology and Glyde-II via the Glyde Object Model. Some of the translations are provided by our colleagues at the German Cancer Research Center who are maintaining GlycomeDB (*www.glycome-db.org/About.action*). In particular, they provide translations between Glyde-II and Linucs as well as between Glyde-II and GlydeCT.

Given a glycan structure specified in Glyde-II, there is a two step process to add this information to the GlycO ontology. The first step of the process involves parsing the Glyde-II XML document to create a DOM tree. The DOM tree is traversed to create a Glycan Object Model (GlyOM) in Java in which the molecule and all of its residues are represented as Java objects that are linked together (see *glycomics.ccrc.uga.edu* for a UML diagram of GlyOM). The second step of the process takes the composite GlyOM object and makes a new instance in the GlycO ontology. Since all of the residues have been pre-loaded into GlycO as part of a canonical representation [8], populating the ontology amounts to establishing the linkage for the glycan molecule. Arbi-

trary linkage is not permitted, since the linkage pattern must comply with the GlycoTree. (Given a root residue, all possible linkages to the next residue are given in the tree). The new structure must match a subtree of GlycoTree, else it is likely to be invalid. It is assumed to be invalid unless a human expert says otherwise (which may indicate that GlycoTree may need to be extended). This semi-automatic curation of glycans in the ontology is facilitated by the GlyB application which visualizes the glycans as schematic structures easily readable by biologists.

4   System Architecture

The foundation of GlycoVault is an Oracle 10g database. In this section, we give a high-level view of all the components in GlycoVault and how they fit together. GlycoVault is a collection of open source systems and tools as well as open source software developed by our group to provide the user with easy access to the system. Figure 1 gives the architecture for the entire GlycoVault system, which is conceptually divided into three layers: a data layer, a knowledge layer and an access layer.
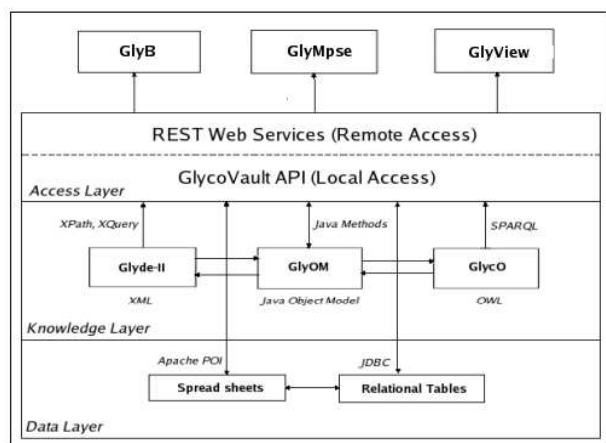

Figure 1: GlycoVault Architecture

When viewed from the bottom, the data layer stores experimental data in the form of relational tables or even spreadsheet files maintained by the database (as either CLOB's or BFILE's). The spreadsheets are loaded using the Apache-POI framework

and the relational tables using Java DataBase Connectivity (JDBC).

The knowledge layer shows the different types of knowledge that go into GlycoVault. This layer represents a much refined version of information/knowledge in the forms of ontologies, XML files and Java classes in GlyOM.

To be useful as a bioinformatics resource, GlycoVault needs to have clear and easy to use Application Programming Interfaces (API's) as well as Web based user interfaces built on these API's. It provides both a local API and a remote API that is a logical subset of the local API. Local access is provided by the GlycoVault API described in the next paragraph. Remote access to GlycoVault is provided using REST Web services. Alternatively, access is provided by a servlet component, which provides Web interfaces for easy access to GlycoVault.

The entire API is divided into several components or packages. For outside users, the important package is the interfaces component. It provides Java interfaces for storing and retrieving information/experimental data into the system. It provides methods to directly store experimental data, such as qRT-PCR and Masspec (Mass spectrometry) data, using one of the Oracle types: BLOB, CLOB or BFILE. It also enables batch-uploading and downloading, which operate on several files at once. We also provide interface to support loading large knowledge bases such as ontologies using the Jena-API.

GlycoVault is automatically populated with data from the performed experiments. Workflows that handle experiments using a mass spectrometer or a PCR machine are currently the main input sources. The qRT-PCR workflow, apart from raw experiment results, produces refined data in the form of spreadsheets which are loaded into GlycoVault with a unique identifier for each experiment. Using information about experimental setup during the upload process, GlycoVault links experimental results to entities in the GlycO ontology. Additionally, a qRT-PCR ontology (presently under development) will be used to annotate the data from the workflows. This ontology (which will be loaded into GlycoVault) along with GlycoVault's meta-structure will provide provenance information about the experiments that have been conducted.

The interface package includes a query interface

to view and retrieve the existing data. We provide interface support for querying the existing data in different query languages such as SPARQL for RDF and OWL stores and XQuery for XML stores. We also provide interfaces that support D2RQ [11] which allows running SPARQL queries on relational tables by initially creating a mapping of the relational table in the form of RDF triples. The query results can be viewed in various formats such as simple text, XML and JSON (JavaScript Object Notation). The implementation package consists of classes that implement the above interfaces.

The topmost part of Figure 1 shows three applications that use the GlycoVault API. Two (GlyB and GlyMpse) of the three use the API to access both experimental data and knowledge about biochemical pathways, while the third (GlyView) provides more general access to the contents of GlycoVault.

Information retrieval by various focused approaches, such as a pathway-centric approach, may be most efficiently implemented using a graph-oriented language such as SPARQL. To further its effectiveness, we are exploring extensions to SPARQL (along the lines of SPARQLeR [12]). SPARQLeR affords the capability to write filters on paths (interlinked triples) using regular expressions. Future extensions to the SPARQL query language will include a context mechanism to create view-dependent subgraphs. The most obvious example would be to create pathway-centric views of the GlycO ontology. This approach has the potential to simplify visualization, browsing and querying of the ontologies by pruning and ranking the relevant information. Finally, interacting with very large ontologies may be intolerably slow when implemented using relational databases such as Oracle. Therefore, we plan to export subsets of the ontologies to BRAHMS [10] for rapid query processing. Through the use of space efficient data structures and time efficient algorithms, BRAHMS is able to query very large ontologies more quickly than other main memory storage and query processing systems. We also plan to make a distributed version of BRAHMS that can handle ontologies in the tens of gigabyte size, with fast access implementations that utilize parallel processing on a high-speed cluster.

5   Application Support

GlycoVault serves as a bioinformatics infrastructure to support the development of applications for glycobiology. In the subsections below, we overview two of the primary applications that are under development (each are further discussed in their own papers that are in preparation).

5.1   GlyB: A Web-Based Browser and Visualization Tool

The centerpiece application for GlycoVault is our Glycomics Browser (GlyB) which supports visualization and analysis of glycomics data and knowledge via the Web. It is particularly useful in visualizing biochemical pathways that include reactions and biochemical structures along with relevant experimental data.

The Glycomics Browser accesses knowledge from the ontologies stored in GlycoVault using the SPARQL query language. It further utilizes Web 2.0 client-side technologies (including AJAX and JavaScript) to represent structural information in a way that is intuitive for glycobiologists and to link this information to experimental data (glycomics, proteomics, and transcriptomics analyses). GlyB provides a graphical display of glycan biosynthetic pathways and associated experimental data. The expermental data is produced by the the qRT-PCR workflow in the form of spreadsheets. GlycoVault converts and stores these data in relational tables that GlyB can then access via SPARQL using D2RQ (as discussed in the architecture section). Glycans are rendered "on the fly" using the standard representation endorsed by the Consortium for Functional Glycomics (CFG), extended to include partonomy relationships (e.g., a carbohydrate residue is drawn in a larger box that represents a glycan moiety). Relevant experimental data (such as the abundance of the Man9GlcNAc2 glycan) can be shown in a separate panel. This is a prototype application that is being extended to include a more powerful graphical interface that can be used for knowledge curation as well as knowledge browsing.

5.2 GlyMpse: Ontology Driven Simulation of Biochemical Pathways

A prototype simulation tool, the Glycomics Modeling pathway simulation environment (GlyMpse), is under development that will simulate biochemical pathways using Hybrid Petri Nets [13]. When applying Petri nets to biological pathways, place nodes are usually used to represent biochemical entities (compounds, enzymes, etc.) and transition nodes are used to represent reactions. In this process, we can simulate pathways based on firing delays and enzyme concentrations. Simulation datasets consist of metabolic pathways, enzyme kinetics, e.g., time courses of metabolite concentrations and structures of metabolites. A major issue in metabolic network simulation is collecting (or determining) enzyme kinetics. Following the ontology driven simulation methodology [14] these simulations are generated from information stored in ontologies and accessed using the SPARQL interface to GlycoVault.

6 Conclusions and Future Work

As discussed in the section on GlyB, the multifaceted and flexible approach used by GlycoVault to store data and knowledge makes it easier to write applications that utilize its content. The fact that relational tables can be accessed via SPARQL eliminates the messiness of making GlyB use both SPARQL and SQL, and then transforming/merging their results. Furthermore, the workflows that populate GlycoVault with data from experiments can store their final as well as intermediate results in relational tables that can easily be bidirectionally mapped from/to spreadsheets. The design philosophy behind GlycoVault was to avoid rigidity and a one solution fits all mentality. Both object-relational databases and ontologies are vital technologies. Although it is tempting to choose one over the other, the flexible, loose integration approach taken by GlycoVault illustrates some of the benefits of utilizing both technologies.

At this point, most of the GlycoVault software is complete and we are working to finish the Web service based remote access, completing the population of GlycoVault, finishing our prototype applications and providing a more general user interface (GlyView) that includes a meta-viewer to help users navigate the full content in GlycoVault. We plan to create a meta-level ontology describing a variety of data components stored in GlycoVault. The ontology will be used to provide the needed meta-descriptions to everything in GlycoVault. The ontology will be used by GlyView to search the GlycoVault contents. In the next few months, we plan to make Glyco-Vault available on the Web (*glycomics.ccrc.uga.edu*) as a generally accessible bioinformatics resource.

References

[1] Z. Wang, X. Gao, C. He, J.A. Miller, J.C. Kissinger, M. Heiges, C. Aurrecoechea, E.T. Kraemer and C. Pennington, "A Comparison of Federated Databases with Web Services for the Integration of Bioinformatics Data," Proc. of the 2007 Int. Conference on Bioinformatics & Computational Biology (BIOCOMP), Las Vegas, NV (June 2007) pp. 334-338.

[2] S.P. Gardner, "Ontologies and Semantic Data Integration," Drug Discovery Today, Vol 10, (2005) pp. 1001-1007.

[3] Graham Klyne, Jeremy Carroll, "Resource Description Framework (RDF)," W3C recommendation (February 2004) .

[4] Eric Prud'hommeaux, Andy Seaborne, "SPARQL: A Query Language for RDF," W3C recommendation (June 2007) .

[5] Deborah L. McGuinness, Frank van Harmelen eds, "Web Ontology Language (OWL)," W3C recommendation (February 2004) .

[6] S.S. Sahoo, A.P. Sheth, W.S. York, J.A. Miller, "Semantic Web Services for N-glycosylation Process," Proc. of the Int. Symposium on Web Services for Computational Biology and Bioinformatics, VBI, Blacksburg, VA (May 2005).

[7] Scott Boag, Don Chamberlin, Mary F.Fernandez, Daniela Florescu, Jonathan Robie, Jerome Simeon, "XQuery: An XML Query Language," W3C recommendation (January 2007) .

[8] C. Thomas, A.P. Sheth and W.S. York, "Modular Ontology Design Using Canonical Building Blocks in the Biochemistry Domain," Proc. of the 4th

Int. Conference on Formal Ontology in Information Systems (FOIS), Baltimore, MD (Nov 2006).

[9] J.J. Carroll, I. Dickinson, C. Dollin, D. Reynolds, A. Seaborne and K. Wilkinson, "Jena: Implementing the Semantic Web Recommendations," Proc. of the 13th Int. World Wide Web Conference (WWW), New York, NY (May 2004) pp. 74-83.

[10] M. Janik and K. Kochut, "BRAHMS: A Work-Bench RDF Store And High Performance Memory System for Semantic Association Discovery", Proc. of the 4th Int. Semantic Web Conference (ISWC), Galway, Ireland (Nov 2005).

[11] C. Bizer and A. Seaborne, "D2RQ: Treating Non-RDF Databases as Virtual RDF Graphs," Proc. of the 3rd Int. Semantic Web Conference (ISWC), Hiroshima, Japan (Nov 2004).

[12] K. Kochut and M. Janik, "SPARQLeR: Extended SPARQL for Semantic Association Discovery," Proc. of the 4th European Semantic Web Conference (ESWC), Innsbruck, Austria (Jun 2007) pp. 145-159.

[13] Hiroshi Matsuno, Yukoki Tanaka, Hitoshi Aoshima, Atsushi Doi, Mika Matsui, Satoru Miyano, "Biopathways Representation and Simulation on Hybrid Functional Petri Net," In Silico Biology, Vol 3, No.3(2003), pp.389-404 .

[14] G.A. Silver, O.A. Hassan and J.A. Miller, "From Domain Ontologies to Modeling Ontologies to Executable Simulation Models," Proc. of the 2007 Winter Simulation Conference (WSC), Washington, DC (Dec 2007).