An XML-Based Approach to Integrating Heterogeneous Yeast Genome Data

Kei-Hoi Cheung[1,4], Deyun Pan[1], Andrew Smith[2], Michael Seringhaus[3]
Shawn M. Douglas[3], Mark Gerstein[3]

[1]Center for Medical Informatics, [2]Computer Science Department, [3]Molecular Biophysics and Biochemistry Department, [4]Genetics Department, Yale University, New Haven, Connecticut, USA
kei.cheung@yale.edu, deyun.pan@yale.edu, andrew.smith@yale.edu, michael.seringhaus@yale.edu
shawn.douglas@yale.edu, mark.gerstein@yale.edu

Abstract. While there are an increasing number of genomes (including the human genome) whose sequences have been fully or nearly completed, the budding yeast *Saccharomyces cerevisiae* was the first fully sequenced eukaryotic genome. Given its ease of genetic manipulation and the fact that many of its genes are strikingly similar to human genes, the yeast genome has been studied extensively through a wide range of biological experiments (e.g., microarray experiments). As a result, a large variety of types of yeast genome data have been generated and made accessible through many resources (e.g., SGD, MIPS, and YPD). While these resources serve many specific needs of individual researchers, we can reap more benefits by integrating these disparate datasets to facilitate larger-context data mining. However, such integrated analysis is hampered by the heterogeneous formats that are used for data distribution. With the increasing use of eXtensible Mark Language (XML) in the bioinformatics domain, we demonstrate how to use XML to standardize the exchange of a variety of types of yeast data between different resources. In particular, we propose a standard XML format called "Yeast Hub XML" (YHX). This format consists of: i) metadata and ii) data. While the former describes the resource and data structure, the latter is used to represent the data. In addition, we apply various XML-related technologies including XPath and XSLT to query, integrate, and transform multiple XML datasets. We have implemented a prototype yeast hub server that allows sharing, querying, and integration of different types and formats of yeast genome data that are located in disparate sources.

1. Introduction

With the advent of sequencing technology, an increasing number of genomes including the human genome have been sequenced. The next step is to determine the biological functions encoded in such DNA sequences. The budding yeast *Saccharomyces cerevisiae* was the first fully sequenced eukaryotic genome [1]. Given its ease of genetic manipulation and the fact that many of its genes are strikingly similar to human genes, the yeast genome has been characterized extensively by a wide range of biological experiments such as transposon insertions [2], DNA microarrays [3], SAGE [4], etc. A large quantity of such experimental data have been published and distributed in disparate formats through numerous web-accessible databases including YPD [5], TRIPLES [6], MIPS [7], SGD [8], etc. While each of these databases serves as a valuable and unique resource that meets some specific research needs, a broader need can be served if the data provided by these resources can be mined or analyzed in an integrated way. For example, the reason that a particular group of yeast genes are found over-expressed (or under-expressed) in a microarray experiment may be explained by integrating such gene expression data with related categories of data such as protein-protein interactions and subcellular localizations. Bioinformatics efforts have been underway to perform large-scale integrative analysis on diverse genomic databases [9]. However, such integrated data analysis has been hampered by a number of factors including the following.

1. *Different identifier schemes.* It is not uncommon that the same genome object (e.g., gene) may be identified using different schemes in different databases. For example, the identifiers used in NCBI's Unigene database (Gene Cluster IDs) [10] are different from those used in TIGR's Gene Index database (Gene Index ID) [11]. In addition, each type of organism may be associated with its own set of gene identifiers (e.g., Flybase vs. Wormbase accession numbers). In the yeast community, this situation is better since the community has reached a consensus on the format of the yeast gene identifiers. Even

when the same identification scheme is used, lexical variations may still occur (e.g., uppercase vs. lowercase identifier strings).

2. *Different data access methods*. Although most biological data can now be accessible through a web interface, different data sources may make their data available in different ways, requiring different programmatic interfaces to be used to implement data access methods. For example, while some data are available in HTML format through web query forms, others are accessible as flat files through the FTP mechanism. Some databases may allow their data to be accessed programmatically through a set of Application Programming Interfaces (API).

3. *Different data representation formats*. To use or interpret the data that have been retrieved, the users need to be aware of the different formats that are used in representing the data. A wide variety of formats ranging from unstructured to structured text files have been used for data representation.

4. *Different data models and schemas.* Different data models such as the relational model and object oriented model can be used to describe the data. Even if the same model is used, different model constructs can be used to describe the same object. For example, while the concept chromosome can be modeled as a column in one relational database, it can be modeled as a table in another.

5. *Lack of standards.* Different databases may use different terms to code the same concept (e.g., different symbols such as *DRD2* and *D2* have been used to refer to the same gene such as *Dopamine Receptor D2*) or use the same term to represent different concepts (e.g., the term *insulin* may mean a gene, protein, or therapeutic agent). This inconsistent use of nomenclature makes cross-database comparison and validation challenging.

6. *Other problems.* When collecting data from multiple resources, we should also consider the following issues: (i) how up-to-date the data are, (ii) whether the data are curated or not, and (iii) how stable or reliable the resources are, and (iv) how evolvable the resources are.

It is noteworthy that the data integration described by this paper through the use of XML is non-trivial. There are many large worldwide commercial and semi-commercial efforts to present assistance for interconnecting web services, such as WSCI (http://www.w3.org/TR/wsci/) and UDDI (http://www.uddi.org/). These systems are somewhat more sophisticated than that proposed here; however, they have one principal shortcoming: the information that they are trying to integrate is so much more general than the specific application here that they have to cope with many, more disparate types of information and they are much less straightforward to use and often less powerful for a particular application. The key insight in this paper is realizing that many, but not all, biological resources can be expressed to some degree in a gene and features viewpoint. Note that this viewpoint does not completely describe the information in a resource but provides a simple view for gathering together lots of resources. Moreover, because of this simple view that is possible in terms of genes and feature, it is possible to construct a very simple web service integration platform such as that described here.

The remainder of the paper is organized as follows. Section 2 introduces the XML format that we use to represent various types of yeast genome data including metadata that describe the individual resources that provide such yeast datasets. Section 3 describes the implementation of the Yeast Hub Server based on the XML approach. Section 4 gives conclusions and outlines future research directions.

2. Use of XML

The first step towards addressing the problem of integrating heterogeneous genome data is the use of a common language to describe the diverse types and formats of data involved. There has been a growing use of the eXtensible Markup Language (XML) as a standard format for exchanging biological data between different resources. Examples include MAGE-ML [12] for gene expression data, BioML [13] for biopolymer data, BSML (http://www.bsml.org) and AGAVE (http://www.lifecde.com) for sequence annotation, SBML [14] for representation and exchange of biochemical network models, ProML [15] for the protein markup language for specification of protein sequences, structures and families, etc.

The advantages of using XML include: machine readability, validatability and a wide base of open software support. In addition, there are advanced XML-based technologies including Resource Description Framework (RDF) (http://www.w3.org/RDF/), Web Services (http://www.w3.org/2002/ws/) that allow the use of standardized metadata to describe resources in a machine readable way, and the automated composition of Web services [16]. As pointed out by Stein [17], these XML technologies can be used to unify the “bioinformatics nation”. However, Stein also mentioned that an incremental approach could be taken so that no significant changes or burdens are imposed on existing (non-XML-based) systems. It is this philosophy that we have adopted for our approach.

This paper presents a standard XML format called “Yeast Hub XML” (YHX) that provides a flexible way of representing and integrating a variety of types of yeast genome data that have been made available over the Internet.

2.1. Yeast Hub XML

One popular format that has been used to distribute genome data including yeast data in bulk is the tabular or grid format where each row represents a gene and each column represents a feature. Although this kind of format is easy for human users to view and can be processed easily by programs such as Excel Spreadsheet, it is not optimal because of the following.

1. *Nonstandard format.* This includes the use of different column delimiters (e.g., tabs vs. commas), different column headers (e.g., the gene id column header may be labeled differently) or no column headers at all, and different ways of describing the meanings of the columns (e.g., they can be described in a separate README file or embedded at the beginning of the file). All these different formats create a problem for parsing the data.
2. *Scalability*. Some of the datasets can involve a large number of columns or rows that can go beyond the capability of such programs as Excel Spreadsheet to handle.
3. *Sparse datasets.* For those datasets (whether they are integrated or not) involving a large number of features, not all genes have values for all of these features. Sometimes, these datasets are very sparsely populated, causing wasted space.

We present an XML format called Yeast Hub XML (YHX), which provides: (i) standardized metadata that describe the structure of the dataset (feature list) as well as the source from which the dataset is obtained, and (ii) a standardized representation of individual datasets. It is a gene-centric format and is based on the assumption that a standardized gene identification scheme is used. This format is based on the extension of an XML-based data exchange protocol (EDSP) described in [18], which is based on entity-attribute-value (EAV) data modeling. The EDSP format is designed to serve as a simple but flexible format for unifying diverse types of data represented in different formats (possibly in different XML formats).

2.1.1. Metadata

Fig.1 shows an example of metadata describing the source of a *data category* (protein descriptions based on Gene Ontology [19]) as well as the columns (features) included in the dataset. These features include specific descriptions of the following three categories: biological process, molecular function, and cellular component. This example dataset is available through the SGD website (http://www.yeastgenome.org/). The metadata include basic information such as name, description, contact person (who downloaded the dataset and made it available through yeast hub), and contact email. In addition, they include the following.

1. *Source_url.* This element indicates the URL through which the dataset can be fetched. It also has an attribute that indicates the format of the data file (e.g., YHX format vs. Tab-delimited format).
2. *Access_url.* This element gives the URL through which the detailed information is provided for a single yeast gene (i.e. this is a template URL for a direct link to the specific gene information at the source database). Notice that the string “~orf” represents a placeholder for a yeast ORF identifier.
3. *Columns.* This element describes the list of features included in the dataset. It has an attribute “key” that indicates which column in the dataset stores the yeast ORF identifiers.

To avoid name conflicts between resources, we use XML namespaces as a means to disambiguate the names (e.g., feature/column names).

2.1.2. Data

Fig. 2 illustrates how the actual data are represented in YHX format. As shown in the figure, each yeast gene is represented by the *orf* element with the *id* attribute holding the ORF identifier. Each feature (col) is represented as a child of the *orf* element with the column (feature) identifier indicated by the *idref* attribute and the feature value indicated by the *value* attribute. The link between the metadata and data documents is through the *category id.*

2.2. Data conversion, integration, and display

Fig. 3 depicts the process of converting and integrating individual datasets obtained from different sources. The figure also shows how the integrated (XML) output can be transformed using the eXtensible Stylesheet Language for Transformation (XSLT) into various display output formats (e.g., HTML format and tabular format). The conversion step takes the individual datasets in grid (tabular) format and translates each of them into the corresponding YHX format. The integration step is performed based on combining (keyed by *orf ID*) the individual YHX formatted datasets into a single integrated YHX dataset. Fig. 4 illustrates how this integration is done by merging two datasets: GO-based protein descriptions (as shown in Fig. 2) and protein properties (obtained also from SGD). The latter describes protein properties such as molecular weight, protein sequence length, and frequency (number of occurrences) of each amino acid (e.g., MET and LEU) within the protein sequence. As shown in this example, the individual feature lists (protein properties and GO descriptions) for each gene are combined into a single feature list. By integrating these two types of data (protein descriptions and protein properties), we can perform integrated analysis to find out whether there is any significant correlation between a certain group of protein properties and a certain group of protein functions. The flexibility of having the integrated output in XML format is that we can easily transform it into different formats (e.g., HTML, XML, and tab-delimited formats) for display purposes, and in general take advantage of the wide variety of standard XML processing technologies. For each YHX data document, our system generates the XSLT code automatically based on the corresponding metadata.

3. Web Server Application

We have implemented a prototype web server application to demonstrate how a Yeast Hub Server (YHS) can be built to integrate diverse types of yeast genome data that are represented in different tabular formats and scattered across different sources. YHS was implemented using the Apache Web server running on a Linux operating system (Redhat). To implement the conversion of a tabular file into our XML format, we have written a Perl program that uses the "XML::writer" module, which is a SAX-based (Simple Application Interface for XML) parser. To integrate multiple YHX documents, we have written a Perl program that uses associative arrays (i.e., hashes). As described previously, for each YHX data document, our system generates the XSLT code automatically based on the corresponding metadata document. This XSLT code generation was done by a Perl-based SAX module. The feature that allows searching through YHX data documents based on single yeast ORF identifier is done using Xpath (available as a Perl module). The XSLT transformation of the integrated output was implemented using the Java version of Xalan (http://xml.apache.org/xalan-j/). It provides the following functionalities.

1. *Registration.* A dataset provided by a particular resource can be registered as a category of information by uploading a metadata file (in YHX format) to the YHS. Once it is registered, the system will download the data file from the specified resource and convert the file (if it is in tabular format) into the YHX format. The generated XML document is stored on the YHS.
2. *Metadata generation and data conversion*. A YHX metadata file can be generated automatically based on the structure of a tabular dataset and the corresponding resource descriptions provided by the user. Based on the generated metadata, the tabular dataset will be converted into YHX format and stored on the server.
3. *Data integration.* The user can integrate two or more categories of data of his/her choice. The chosen datasets will be integrated or joined based on common ORF identifiers. The user can select which columns to be included in the integrated output that can be presented in either tabular or YHX format.

4. *Single ORF search.* The user can search across multiple categories of data based on a single ORF identifier. A composite report for the matched gene will be returned.

YHS represents a light-weight data warehouse (without using a database engine) that integrates and transforms different categories of yeast data into the YHX format.

4. Conclusions and Future Directions

We have described an XML-based framework that allows diverse types of yeast genome data, which are represented in different formats and located in disparate sources, to be integrated. A prototype yeast hub system was built to demonstrate that such a system could serve as a central resource to facilitate data sharing and large-scale integrated data analysis or data mining. The overhead of using this system is minimal. In many cases, the user only needs to publish the source data in our standard YHX metadata format (which requires a minimal amount of information). Also, we provide tools that allow the user to convert a non-standard tabular (grid) format into our YHX format (metadata are generated at the same time). While this paper represents a proof of concept, we consider the following areas as our future research directions.

1. *Other XML technologies.* While our approach demonstrates the use of XML in inter-linking different types and formats of yeast data, other XML-based technologies should be explored and incorporated to empower our system. These include Web Services, Web Service Choreography (http://www.w3.org/TR/2002/NOTE-wsci-20020808/), and Universal Description, Discovery and Integration of Web Services (UDDI) (http://www.uddi.org/), and Life Sciences Identifiers (LSID) (http://www.i3c.org/wgr/ta/resources/lsid/docs/).

2. *Semantic Web.* We have proposed a minimal standard metadata scheme to facilitate data sharing and publishing. However, this approach does not provide a more sophisticated way of describing and querying different resources. To provide a more intelligent reasoning capability, we can explore the use of semantically rich ontologies (including both generic and domain specific ontologies) in classifying, describing, querying and constraining the resources. This will allow us to make queries without knowing details of data, facilitating scientific discovery.

3. *Beyond the yeast genome.* Our current framework is designed to interoperate yeast genome data, but it can be extended to handle other genomes (e.g., the human genome). We will explore interoperation issues that are unique to some organisms.

4. *Gene-centric integration.* While our approach features a gene-centric way of data integration, not all genomic databases are organized by gene. For example, a database that stores synteny information between the mouse and human genomes will have mappings between sequence regions that contain many genes, contain only parts of genes, or do not contain genes (intergenic regions). Nevertheless, organization by gene is a natural way genomics databases can be organized, and the gene-centric view would probably be the best compromise representation if one were forced to pick a single representation for all genomics data, allowing the best retention of key data and minimizing information loss for transformations to it . Thus, importantly, genomics databases that are not organized by gene can likely be transformed into a gene-centric view, probably with some summarization of the encoded information necessary. For example, the mouse-human synteny database could be transformed to a gene-centric view with summary features for genes such as chromosome the gene is on, corresponding chromosome in the other genome, length of and number of other genes in the syntenic region the gene is in, etc. While the richness and complexity of the original database structure is lost in such transformations, much useful summary information can still be retained and, importantly, the new gene-centric view of the data allows a simple but powerful integration with the many other databases that are (or can be made, like the example) organized around the gene. We will thus explore the extent to which non-gene-centric yeast data can be converted into a gene-centric representation, and we will develop and integrate support for such transformations into our yeast hub system. We want to emphasize that such transformations are not trivial, but will be an interesting and key direction of our future research.

5. *Performance*. While XML has been used as a standard approach for interchanging data between heterogeneous sources, it is not as powerful as a database in terms of retrieving and querying data. Research has been done on the next generation XML-native databases [20, 21]. In addition, parallel and distributed technologies like parallel main memory databases [22] or parallel/distributed database servers [23] can potentially be used to increase performance to a significant extent.

Acknowledgement

This work is supported in part by NSF grant DBI-0135442 and NIH grant K25 HG02378.

References

1. Goffeau A, Barrell B, Bussey H, Davis R, Dujon B, Feldmann H, Galibert F, Hoheisel J, Jacq C, Johnston M, Louis E, Mewes H, Murakami Y, Philippsen P, H HT, SG SO, *Life with 6000 genes.* Science, 1996. **274**(5287): 546, 563-7.
2. Ross-Macdonald P, Sheehan A, Roeder GS, Snyder M, *A multipurpose transposon system for analyzing protein production, localization, and function in Saccharomyces cerevisiae.* PNAS, 1997. **94**(1): 190-5.
3. Gasch A, *Yeast genomic expression studies using DNA microarrays.* Methods Enzymol, 2002. **350**: 393-414.
4. Kal AJ, van Zonneveld AJ, Benes V, van den Berg M, Koerkamp MG, Albermann K, Strack N, Ruijter JM, Richter A, Dujon B, Ansorge W, Tabak HF, *Dynamics of Gene Expression Revealed by Comparison of Serial Analysis of Gene Expression Transcript Profiles from Yeast Grown on Two Different Carbon Sources.* Mol. Biol. Cell, 1999. **10**(6): 1859-72.
5. Csank C, Costanzo M, Hirschman J, Hodges P, Kranz J, Mangan M, O'Neill K, Robertson L, Krzypek M, Brooks J, Garrels J, *Three yeast proteome databases: YPD, PombePD, and CalPD (MycoPathPD).* Methods Enzymol, 2002. **350**: 347-73.
6. Kumar A, Cheung K-H, Tosches N, Masiar P, Liu Y, Miller P, Snyder M, *The TRIPLES database: a community resource for yeast molecular biology.* Nucl. Acids. Res., 2002. **30**(1): 73-5.
7. Mewes HW, Frishman D, Guldener U, Mannhaupt G, Mayer K, Mokrejs M, Morgenstern B, Munsterkotter M, Rudd S, Weil B, *MIPS: a database for genomes and protein sequences.* Nucl. Acids. Res., 2002. **30**(1): 31-4.
8. Dwight SS, Harris MA, Dolinski K, Ball CA, Binkley G, Christie KR, Fisk DG, Issel-Tarver L, Schroeder M, Sherlock G, Sethuraman A, Weng S, Botstein D, Cherry JM, *Saccharomyces Genome Database (SGD) provides secondary gene annotation using the Gene Ontology (GO).*Nucl. Acids. Res., 2002. **30**(1): 69-72.
9. Gerstein M, *Integrative database analysis in structural genomics.* Nat Struct Biol, 2000. **7**(Suppl): 960-3.
10. Wheeler DL, Church DM, Federhen S, Lash AE, Madden TL, Pontius JU, Schuler GD, Schriml LM, Sequeira E, Tatusova TA, Wagner L, *Database resources of the National Center for Biotechnology.* Nucl. Acids. Res., 2003. **31**(1): 28-33.
11. Quackenbush J, Cho J, Lee D, Liang F, Holt I, Karamycheva S, Parvizi B, Pertea G, Sultana R, White J, *The TIGR Gene Indices: analysis of gene transcript sequences in highly sampled eukaryotic species.* Nucl. Acids. Res., 2001. **29**(1): 159-164.
12. Spellman P, Miller M, Stewart J, Troup C, Sarkans U, Chervitz S, Bernhart D, Sherlock G, Ball C, Lepage M, Swiatek M, Marks W, Goncalves J, Markel S, Iordan D, Shojatalab M, Pizarro A, White J, Hubley R, Deutsch E, Senger M, Aronow B, Robinson A, Bassett D, Stoeckert C, Brazman A, *Design and implementation of microarray gene expression markup language (MAGE-ML).* Genome Biology, 2002. **3**(9): 1-9.
13. Fenyo D, *The Biopolymer Markup Language.* Bioinformatics, 1999. **15**(4): 339-40.
14. Hucka M, Finney A, Sauro H, Bolouri H, Doyle J, Kitano H, Arkin A, Bornstein B, Bray D, Cornish-Bowden A, Cuellar A, Dronov E, Gilles E, Ginkel M, Gor V, Goryanin I, Hedley W, Hodgman T, Hofmeyr J, Hunter P, Juty N, Kasberger J, Kremling A, Kummer U, Novere NL, Loew L, Lucio D, Mendes P, Minch E, Mjolsness E, Nakayama Y, Nelson M, Nielsen P, Sakurada T, Schaff J, Shapiro B,

Shimizu T, Spence H, Stelling J, Takahashi K, Tomita M, Wagner J, Wang J, *The systems biology markup language (SBML): a medium for representation and exchange of biochemical network models.* Bioinformatics, 2003. **19**(4): 524-31.

15. Hanisch D, R RZ, Lengauer T, *the protein markup language for specification of protein sequences, structures and families.* In Silico Biol., 2002. **2**(3): 313-24.
16. Chandrasekaran S, Miller J, Silver G, Arpinar B, Sheth A, *Performance Analysis and Simulation of Composite Web Services.* International Journal of Electronic Commerce and Business Media, 2003. **13** (2): 18-30.
17. Stein L, *Creating a bioinformatics nation.* Nature, 2002. **417**: 119-20.
18. Marenco L, Tosches N, Crasto C, Shepherd G, Miller PL, Nadkarni PM, *Achieving Evolvable Web-Database Bioscience Applications Using the EAV/CR Framework: Recent Advances.* J Am Med Inform Assoc, 2003. **10**(5): 444-453.
19. Ashburner M, Ball C, Blake J, Botstein D, Butler H, Cherry M, Davis A, Dolinski K, Dwight S, Eppig J, Harris M, Hill D, Issel-Tarver L, Kasarskis A, Lewis S, Matese J, Richardson J, Ringwald M, Rubin G, Sherlock G, *Gene ontology: tool for the unification of biology.* Nature Genetics, 2000. **25**: 25-29.
20. Jagadish H, Al-Khalifa S, Chapman A, Lakshmanan L, Nierman A, Paparizos S, Patel J, rivastava D, Wiwatwattana N, Wu Y, Yu C, *TIMBER: A native XML database.* VLDB, 2002. **11**(4): 274-291.
21. Sipani S, Miller J, KVerma, Aleman-Meza B, *Designing a High Performance Database Engine for the `Db4XML' Native XML Database System.* Journal of Systems and Software, 2004. **69**(1): 87-104.
22. Eich M, *The design of a main memory database machine.* IWDM, 1987: 325-338.
23. Boral H, Alexander W, Clay L, Copeland G, Danforth S, Franklin M, Hart B, Smith M, Valduriez P, *Prototyping Bubba, a highly parallel database system.* IEEE Trans Knowledge and Data Eng, 1990. **2** (1): 4-23.

```
<?xml version="1.0" encoding="UTF-8" ?>
<yeast-hub-meta>
    <category id=1>
        <name>GO-protein</name>
        <description>GO-based protein descriptions</description>
        <xml_URL>http://128.36.123.97/dpan_xml/orf_geneontology.xml
            </xml_URL>
                                            <source_URL>ftp://genome-
            ftp.stanford.edu/pub/yeast/data_download/literature_curation/
            orf_geneontology.tab</source_URL>
                            <access_URL>http://db.yeastgenome.org/cgi-
            bin/SGD/locus.pl?locus=~orf</access_URL>
        <contact-person>Kei Cheung</contact-person>
        <contact-email>kei.cheung@yale.edu</contact-email>
        <date>Tue Dec 2 10:55:52 EST 2003</date>
        <data-file name="orf_geneontology.tab" />
        <columns key="ORF">
            <column id="Gene" name="Gene" description="" />
            <column id="Length" name="Length" description="" />
            <column id="Process" name="Process" description="" />
            <column id="Function" name="Function" description="" />
                <column    id="Component"    name="Component"
              description="" />
            <column id="SGDID" name="SGDID" description="" />
        </columns>
    </category>
</yeast-hub-meta>
```

Fig. 1. YHX metadata representation.

```
<yeast-hub-data>
     <category id=1>
                . .
          <orf id="YBR105C">
             <col idref="Gene" value="VID24"/>
             <col idref="Process" value="vesicle-mediated transport*"/>
             <col idref="Function" value="molecular_function unknown"/>
             <col idref="Component" value="extrinsic to membrane*"/>
             <col idref="SGDID" value="S0000309"/>
           </orf>
                . .
       </category>
</yeast-hub-data>
```

Fig. 2. YHX data representation.

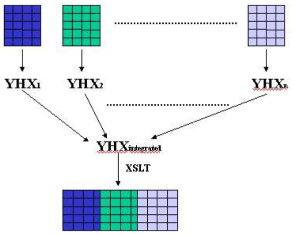

Fig. 3. The process of data conversion and integration.

```
[protein functional descriptions]
<yeast-hub-data>
                ...
     <orf id="YBR105C">
          <col idref="Gene" value="VID24"/>
          <col idref="Process" value="vesicle-mediated transport*"/>
          <col idref="Function" value="molecular_function unknown"/>
          <col idref="Component" value="extrinsic to membrane*"/>
          <col idref="SGDID" value="S0000309"/>
       </orf >
                ...
</yeast-hub-data>

[protein properties]
<yeast-hub-data>
                ...
      <orf id="YBR105C">
          <col idref="SGDID" value="S0000309"/>
          <col idref="MOLECULAR WEIGHT" value="41245"/>
          <col idref="PI" value="6.74"/>
          <col idref="CAI" value=".125"/>
          <col idref="PROTEIN LENGTH" value="362"/>
          <col idref="N TERM SEQ" value="MINNPKV"/>
          <col idref="C TERM SEQ" value="DCSFEFA"/>
          <col idref="CODON BIAS" value=".048"/>
          <col idref="ALA" value="19"/>
                …
          <col idref="VAL" value="19"/>
          <col idref="FOP SCORE" value=".438"/>
          <col idref="GRAVY SCORE" value="-.656077"/>
          <col idref="AROMATICITY SCORE" value=".11326"/>
```

```
        <col idref="Feature type" value="ORF|Verified"/>
      </orf >
            ...
</yeast-hub-data>
```

[integrated results]

```
<yeast-hub-data>
            ...
      <orf id="YBR105C">
        <col idref="Gene" value="VID24"/>
        <col idref="Length" value="1089"/>
        <col idref="Process" value="vesicle-mediated transport*"/>
        <col idref="Function" value="molecular_function unknown"/>
        <col idref="Component" value="extrinsic to membrane*"/>
        <col idref="SGDID" value="S0000309"/>
        <col idref="MOLECULAR WEIGHT" value="41245"/>
        <col idref="PI" value="6.74"/>
        <col idref="CAI" value=".125"/>
        <col idref="PROTEIN LENGTH" value="362"/>
        <col idref="N TERM SEQ" value="MINNPKV"/>
        <col idref="C TERM SEQ" value="DCSFEFA"/>
        <col idref="CODON BIAS" value=".048"/>
        <col idref="ALA" value="19"/>
            …
        <col idref="VAL" value="19"/>
        <col idref="FOP SCORE" value=".438"/>
        <col idref="GRAVY SCORE" value="-.656077"/>
        <col idref="AROMATICITY SCORE" value=".11326"/>
        <col idref="Feature type" value="ORF|Verified"/>
      </orf >
            ...
</yeast-hub-data>
```

Fig. 4. Integrated data in YHX format.