Biological Database Modeling

Artech House Bioinformatics & Biomedical Imaging
Stephen T. C. Wong and Guang-Zhong Yang, Series Editors

Advanced Methods and Tools for ECG Data Analysis, Gari D. Clifford, Francisco Azuaje, and Patrick E. McSharry, editors

Biomolecular Computation for Bionanotechnology, Jian-Qin Liu and Katsunori Shimohara

Electrotherapeutic Devices: Principles, Design, and Applications, George D. O'Clock

Intelligent Systems Modeling and Decision Support in Bioengineering, Mahdi Mahfouf

Life Science Automation Fundamentals and Applications, Mingjun Zhang, Bradley Nelson, and Robin Felder, editors

Matching Pursuit and Unification in EEG Analysis, Piotr Durka

Microfluidics for Biotechnology, Jean Berthier and Pascal Silberzan

Systems Bioinformatics: An Engineering Case-Based Approach, Gil Alterovitz and Marco F. Ramoni, editors

Text Mining for Biology and Biomedicine, Sophia Ananiadou and John McNaught

Biological Database Modeling

Jake Chen
Amandeep S. Sidhu

Editors

ARTECH
HOUSE

BOSTON | LONDON
artechhouse.com

Library of Congress Cataloging-in-Publication Data
A catalog record for this book is available from the U.S. Library of Congress.

British Library Cataloguing in Publication Data
A catalogue record for this book is available from the British Library.

ISBN-13: 978-1-59693-258-6

Cover design by Igor Valdman

685 Canton Street
Norwood, MA 02062

10 9 8 7 6 5 4 3 2 1

List of Contributors

Chapter 1 Amandeep S. Sidhu,
Curtin University of Technology, Perth, Australia
Jake Chen,
Indiana University School of Informatics, Indianapolis, Indiana, United States

Chapter 2 Viroj Wiwanitkit,
Chulalongkorn University, Bankok, Thailand

Chapter 3 Ramez Elmasri, Feng Ji, and Jack Fu,
University of Texas at Arlington, Arlington, Texas, United States

Chapter 4 Viroj Wiwanitkit,
Chulalongkorn University, Bankok, Thailand

Chapter 5 Amandeep S. Sidhu, Tharam S. Dillon, and Elizabeth Chang,
Curtin University of Technology, Perth, Australia

Chapter 6 Cornelia Hedeler and Paolo Missier,
University of Manchester, Manchester, United Kingdom

Chapter 7 Greg Butler, Wendy Ding, John Longo, Jack Min, Nick O'Toole, Sindhu Pillai, Ronghua Shu, Jian Sun, Yan Yang, Qing Xie, Regis-Olivier Benech, Aleks Spurmanis, Peter Ulycznyj, Justin Powlowski, Reg Storms, and Adrian Tsang,
Concordia University, Canada

Chapter 8 Willy A. Valdivia-Granda,
Orion Integrated Biosciences, Inc., New York, United States
Christopher Dwan,
BioTeam, Inc., Cambridge, Massachusetts, United States

Chapter 9 Zhong Yan and Jake Chen,
Indiana University School of Informatics, Indianapolis, Indiana, United States
Josh Heyen, Lee W. Ott, Maureen A. Harrington, and Mark G. Goebl,
Indiana University School of Medicine and the Walther Cancer Institute, Indianapolis, Indiana, United States
Cary Woods,
Ball State University, Muncie, Indiana, United States

Chapter 10 Karthik Raman, Yeturu Kalidas, and Nagasuma Chandra,
Supercomputer Education and Research Centre and Bioinformatics Centre, Indian Institute of Science, India

Chapter 11 Preeti Malik, Tammy Chan, Jody Vandergriff, Jennifer Weisman, Joseph DeRisi, and Rahul Singh,
San Francisco State University, San Francisco, California, United States

Contents

CHAPTER 11

Preface

Database management systems (DBMS) are designed to manage large and complex data sets. In the past several decades, advances in computing hardware and software and the need to handle rapidly accumulating data archived in digital media have led to significant progress in DBMS research and development. DBMS have grown from simple software programs that handled flat files on mainframe computers, which were prohibitively expensive to all but a few prestigious institutions, into today's popular form of specialized software platforms underpinning wide ranges of tasks, which include business transactions, Web searches, inventory management, financial forecasts, multimedia development, mobile networks, pervasive computing, and scientific knowledge discovery. Technologies of DBMS have also become increasingly sophisticated, diverging from generic relational DBMS into object-relational DBMS, object-oriented DBMS, in-memory DBMS, semantic Webs data store, and specialized scientific DBMS. Given the sustained exponential data growth rate brought forth by continued adoption of computing in major industries and new inventions of personal digital devices, one can safely predict that DBMS development will continue to thrive in the next millennium.

In this book, we want to share with our readers some fresh research perspectives of post-genome biology data management, a fast-growing area at the intersection of life sciences and scientific DBMS domains. Efficient experimental techniques, primarily DNA sequencing, microarrays, protein mass spectrometers, and nanotechnology instruments, have been riding the wave of the digital revolution in the recent 20 years, leading to an influx of high-throughput biological data. This information overload in biology has created new post-genome biology studies such as genomics, functional genomics, proteomics, and metabolomics—collectively known as "omics" sciences in biology. While most experimental biologists are still making the transition from one-gene-at-a-time type of studies to the high-throughput data analysis mindset, many leaders of the field have already begun exploring new research and industrial application opportunities. For example, managing and interpreting massive omics data prelude ultimate systems biology studies, in which one may analyze disparate forms of biological data and uncover coordinated functions of the underlying biological systems at the molecular and cellular signalling network level. On the practical side, understanding diverse intricate interplays between environmental stimuli and genetic predisposition through omics evidence can help pharmaceutical scientists design drugs that target human proteins with high therapeutic values and low toxicological profiles. With data management tools to handle terabytes of omics data already released in the public domain, the promise of post-genome biology looms large.

Compared with data from general business application domains, omics data has many unique characteristics that make them challenging to manage. Examples of these data management challenges are:

1. Omics data tends to have more complex and more fast-evolving data structures than business data. Biological data representation often depends on scientific application scenarios. For example, biological sequences such as DNA and proteins can be either represented as simple character strings or connected nodes in three-dimensional spatial vectors. Data representation is an essential first step.
2. Omics data is more likely to come from more heterogeneously distributed locations than business data. To study systems biology, a bioinformatics researcher may routinely download genome data from the Genome Database Center at the University of California, Santa Cruz, collect literature abstracts from the PubMed database at the National Library of Medicine in Maryland, collect proteome information from the Swiss-Prot database in Switzerland, and collect pathway data from the KEGG database in Japan. Data integration has to be carefully planned and executed.
3. Omics data tends to reflect the general features of scientific experimental data: high-volume, noisy, formatted inconsistently, incomplete, and often semantically incompatible with one another. In contrast, data collected from business transactions tends to contain far fewer errors, is often more accurate, and shows more consistencies in data formats/coverage. Meticulous data preprocessing before knowledge discovery are required.
4. Omics data also lags behind business data in standard development. For example, Gene Ontology (GO) as a standard to control vocabularies for genes was not around until a decade ago, whereas standards such as industrial product categories have been around for decades. The ontology standards and naming standards for pathway biology are still under development. This makes it difficult to perform mega collaboration, in which cross-validation of results and knowledge sharing are both essential.

Despite all the challenges, modeling and managing biological data represent significant discovery opportunities in the next several decades. The human genome data bears the ultimate solutions of expanding the several thousand traditional molecular drug targets into tens of thousands genome drug targets; molecular profiling information, based on individuals using either the microarrays or the proteomics platform, promises new types of molecular diagnostics and personalized medicine. As new applications of massive biological data emerge, there will be an increasing need to address data management research issues in biology.

In this compiled volume, we present to our readers a comprehensive view of how to model the structure and semantics of biological data from public literature databases, high-throughput genomics, gene expression profiling, proteomics, and chemical compound screening projects. The idea of compiling this book, which we found to be unique, stems from the editors' past independent work in bioinformatics and biological data management. While topics in this area are diverse and interdisciplinary, we focused on a theme for this book—that is, how to model and manage

omics biological data in databases. By promoting this theme for the past decade among ourselves and the contributing authors of this book, we have contributed to solving complex biological problems and taking biological database management problems to the next level. We hope our readers can extract similar insights by using this book as a reference for future related activities.

There are 11 chapters presented in this book. Individual chapters have been written by selected accomplished research teams active in the research of respective topics. Each chapter covers an important aspect of the fast-growing topic of biological database modeling concepts. Each chapter also addresses its topic with varying degrees of balance between computational data modeling theories and real-world applications.

In Chapters 1 through 5, we introduce basic biological database concepts and general data representation practices essential to post-genome biology. First, biological data management concepts are introduced (Chapter 1) and major public database efforts in omics and systems biology studies are summarized (Chapter 2). Then, biomedical data modeling techniques are introduced (Chapter 3). Next, Gene Ontology as an established basic set of controlled vocabulary in genome database annotations is described (Chapter 4). Finally, the latest research on protein ontology and the use of related semantic webs technologies are presented to enable readers to make the connection between emerging biological data collection and integration trends (Chapter 5).

In Chapters 6 through 9, we examine in detail how to develop data management techniques to process and analyze high-throughput biological data through case studies. First, quality control techniques to reduce variations during experimental data collection steps are described (Chapter 6). Then, biological sequence management experience for a fungi genomics project is discussed (Chapter 7). Next, data management and data integration methods for microarray-based functional genomics studies are investigated (Chapter 8). Finally, data management challenges and opportunities for mass spectrometry based expression proteomics are presented (Chapter 9).

In Chapters 10 and 11, we delve into the practical aspect, demonstrating how to apply biological data management for drug discoveries. First, fundamental drug discovery concepts based on macromolecular structural modeling are introduced (Chapter 10); then, a data management software system that implements high-throughput drug compound screenings is discussed (Chapter 11) to conclude the book.

We hope this book will become a useful resource for bioinformatics graduate students, researchers, and practitioners interested in managing post-genome biological data. By studying the techniques and software applications described in this book, we hope that bioinformatics students will use the book material as a guide to acquire basic concepts and theories of post-genome biological data management, bioinformatics practitioners will find valuable lessons for building future similar biological data management systems, and researchers will find rewarding research data management questions to address in the years to come.

Acknowledgments

We wish to thank all of the authors for sharing their insightful knowledge and making excellent contributions to this book based on their active research in biological data management. This book would not have been completed without the tremendous efforts, held to the highest standards, of all the authors, each of whom spent numerous hours over many drafts in preparing, collating, and revising their writings over the past 2 years. During the publishing process, many colleagues also helped and they deserve our whole-hearted appreciation. They are: David Wong from Indiana University, who provided legal advice for the contract agreement; Dr. Zongmin Ma from Northwestern University of China, who provided assistance in the initial conceptualization and execution of the book publishing process; Susan Lagerstrom-Fife from Springer Science+Business Media, Inc., who guided us from a publisher's perspective while we explored various publication options; Wayne Yuhasz from Artech House Publishers, whose persistence and dedication to timely assistance finally won us over in making our book part of the Artech House Bioinformatics & Biomedical Imaging Series; and Barbara Lovenvirth from Artech House Publishers, who assisted us throughout the final publication process.

We also want to thank the general support from our home institutions: Indiana University School of Informatics at Indianapolis, Indiana; Purdue University School of Science, Department of Computer and Information Sciences at Indianapolis, Indiana; and Curtin University of Technology, Perth, Australia.

Last, but not least, Jake Chen wishes to thank his family, including his wife, Mabel Liu, for her unbounded love and support in assuming additional responsibilities so that the book project could be completed.

Jake Chen
Indianapolis, Indiana
Amandeep S. Sidhu
Perth, Australia
Editors
October 2007

CHAPTER 1

Introduction to Data Modeling

Amandeep S. Sidhu and Jake Chen

Scientific data is often scattered among heterogeneous data repositories. Exploring data across multiple data repositories requires the ability to understand and correlate their structures (schemas). Such correlations need to address the diversity of views of the scientific domain represented by different data repositories as well as the diversity of data modeling languages used for expressing these views. In this chapter, we introduce the concepts of *data modeling* and discuss its application to *biological databases*.

1.1 Generic Modern Markup Languages

Modern markup languages, such as Standard Generalized Markup Language (SGML) [1] and eXtensible Markup Language (XML) [2], which were initially conceived for modeling texts, are now receiving increasing attention as formalisms for data and knowledge modeling. XML is currently establishing itself as a successor of HyperText Markup Language (HTML) for a better modeling of texts as well as of other kinds of data. There are several reasons for this evolution. Even though multiple databases may cover the same data, their focus might be different. Modern markup languages such as SGML and XML are generic in that:

- They serve to specify the semantic structure, not the layout, of documents or data items.
- They make it possible to freely specify application-dependent document or data structures.

In the following, the term "data" refers also, but not exclusively, to text data. Thus, a data item may consist of: (1) text only (such data items are also known as human-readable documents); (2) nontext only (such data items are also known as data-oriented documents); or (3) both (such data items are also known as mixed-model documents). In the terminology of generic markup languages, data items are called documents. In the following, the term "data item" is used in lieu of "document" for stressing that not only (structured) texts are meant, but more generally (structured) data of any kind.

Widespread specific markup languages such as PostScript or Rich Text Format (RTF), whose conceptual roots go back to the 1970s, serve to specify the layout of data items. Here, layout is not exclusively meant as the appearance of a data item when printed on paper, but more generally as any kind of presentation of a data item to human perception. Examples of such an extended notion of layout include the formats of data items as they are displayed on a terminal screen, rendered in the script on an output device, or presented by any other means on any device.

The family of generic markup languages started in the late 1980s with the conception of its first specimen, SGML. The purpose of a generic markup language is to specify the semantic—or logical—structure of data items, not their layout. In the following, the term "presentation" is reserved to refer to the layout of a data item in the extended sense above, while the term "representation" refers to how semantics is conveyed through structural elements of the underlying data modeling formalism.

The distinction between layout and structure is important, for a layout format is device or system dependent, whereas a semantic structure should not be. It is desirable that the semantic structure of data items be specified independently of any layout. This ensures both:

- Independence of data modeling from data usage;
- Independence of data modeling from presentation devices.

The first property, data independence from usage, is important because data is rarely used in a single manner only. The second property, data independence from presentation devices, is important for several reasons. To begin with, different kinds of presentation devices require different layouts. For example, a structurally complex data item is likely not to be displayed using identical layouts on standard size screens and on small screens like those of cellular phones. Also, such devices are likely to become technically obsolete sooner than data. Moreover, a presentation format does not necessarily fully convey data semantics. For instance, it is common practice to rely on printed text layout for conveying semantic structure when using text processing systems or the markup language HTML. This practice often leads to semantic losses, especially when files are transferred from one text processing system to another, because the layout of the one system cannot always be faithfully mapped into that of the other system.

In order to specify layouts for classes of documents specified in a generic markup language, so-called style-sheet languages are used in addition. These languages basically allow the definition of layouts for those structural elements specified with the markup language. Such definitions do not have to be unique, thus ensuring the desired independence of the data from their presentations in various contexts.

Generic markup languages (like the XML family of languages) do not impose any predefined structure, nor any predefined names for the structural elements occurring in data items. Structure and names can be freely chosen, hence the denomination of generic markup language. Thus, using generic markup languages it is possible to faithfully model the structure of data items needed in applications and to name the structural elements of a chosen structure in a way that is natural in the application context.

1.2 Modeling Complex Data Structures

Complex structures are essential for they are ubiquitous in data modeling—from records in programming languages to objects in programming languages, artificial intelligence, software engineering, databases, and logics for knowledge representation. The formalisms provided by generic markup languages make it possible to specify complex structures. Therefore, they are much richer than the data model of relational databases. They are also richer than the data models of current object database systems, because they allow optional elements. Optional elements are very appealing in databases, for they make it possible to express exceptions, which often occur in practical database applications.

The term "semistructured data" has been coined for emphasizing the possibility of such exceptions in the framework of structure-conveying data. It is under the denomination of semistructured data that most database research on using markup languages for data modeling is currently pursued.

1.3 Data Modeling with General Markup Languages

Data modeling with generic markup languages is an interdisciplinary area of research at the crossing of four traditionally distinct fields of research:

- Databases;
- Artificial intelligence;
- Information retrieval;
- Document processing.

This convergence is interesting, because each field brings its own focus, methods, and philosophy.

From databases, the research area of data modeling with generic markup languages gains an interest for declarative query languages, of which SQL is the most well-known example. Declarativeness is a loosely defined notion to be understood here as meaning that the users of such query languages do not have to be aware of the computation strategy, of the internal organization of the data in memory, nor—or as little as possible—of termination issues and of efficiency. Indeed, queries expressed in query languages are automatically optimized. This makes query optimization possible. Query optimization guarantees a predictable "average efficiency," which is one of the appreciated features of database systems. Also from databases, the area of data modeling with generic markup languages inherits its interest for data structures, ensuring an efficient storage, retrieval, and updating of very large data sets. Conversely, database research itself is enriched by the attention to text data, to data accessible from the Web, and to richer data models allowing for exceptions.

From artificial intelligence, the research area gains data and knowledge modeling methods that go far beyond the relational or object database models. Artificial intelligence approaches to knowledge representation have always been driven by natural language applications, where extremely rich and complex semantics are

encountered. The need for software interoperability and data interchange in Web-based applications such as electronic commerce, health care management, and computational biology (also called bioinformatics) nowadays has led researchers, as well as practitioners, to express advanced artificial intelligence knowledge representation formalisms such as description logics by relying upon the generic markup language XML.

From information retrieval, the research area can learn how to automatically "grasp" knowledge from the content of—in general, large—texts. The field of information retrieval itself gains from the consideration of structured texts, which up until recently have almost never been considered by the information retrieval community.

The contributions of the document processing field to the research area of data modeling with generic markup languages are indeed the generic markup languages and hyperlink models. Document processing itself might benefit from the interdisciplinary approach to data modeling with methods for declarative query answering and for an efficient storage.

The focus of the research activities described here is on data modeling as it has emerged from combining techniques and ideas from both databases and artificial intelligence. As a consequence, issues more specific to one of the fields of databases, information retrieval, and document processing are likely to receive less attention in this chapter.

1.4 Ontologies: Enriching Data with Text

Enriching standard data—like the numerical and string data of classical managerial databases—with more informative texts is an old issue in database research and in applications known as data dictionaries. Data dictionaries are basically agreed-upon vocabularies for an application or a class of applications and often taxonomies (i.e., classifications of terms).

Recently, the issue has gained new attention and has been enhanced with artificial intelligence knowledge modeling techniques, leading to so-called ontologies. An ontology provides a vocabulary whose terms are precisely defined by texts such as dictionary entries or encyclopedia entries. Further, ontology also defines semantic relationships between terms using formal modeling techniques, in general taken from logic-based specification formalisms such as description logics. Thus, ontology starts from precisely defined basic concepts, building up complex concepts by relying on relationships that are precisely defined as well. These relationships permit the construction of taxonomies, but also of richer structures.

With the advent of the World Wide Web and data interchange intensive applications, ontologies are becoming a key issue. They are used for ensuring software interoperability and data sharing [3]. In spite of their practical relevance, data dictionaries and ontologies have up until recently not received as much attention from the database research community as database practitioners might have wished. This discrepancy clearly reflects the fact that, up until now, textual data was not a central interest in database research and database system development, nor was the modeling of complex semantic relationships as in artificial intelligence knowledge model-

ing approaches such as description logics. This seems to be changing now. Arguably, textual data and the modeling of complex semantic relationships are gaining in importance within database research.

1.5 Hyperlinks for Semantic Modeling

A hyperlink model defines hyperlink types based on semantic relationships between data items. A browser model defines the behavior of such hyperlinks. The proposed distinction is analogous to the distinction between a generic markup language, whose purpose is to model the logical structure of data items, and a style-sheet language, whose purpose is to specify the layout for a given logical structure.

A hyperlink model might define the following, semantically characterized, hyperlink types: (1) hyperlink to new information; (2) hyperlink to alternative descriptions of the current information; and (3) hyperlink to supplementary information. A browser model could then specify that these three types of hyperlinks behave in the three ways described above. Different browser models may specify the behavior differently.

It is interesting to investigate different browser models for the same hyperlink model. There are two reasons. First, in the future many data items are likely to be browsed using not only standard size screens, but also mini-screens like those of cellular phones or new paper-like electronic output devices like electronic paper (e-paper), with which browsing will most likely take new forms. Second, if hyperlinks are to be used for expressing semantic dependencies, which is taken here as a working assumption, then necessarily browsing along such hyperlinks will not be uniquely definable. Thus, the distinction between hyperlink model and browser model proposed here contributes to both independence of data modeling from data usage and independence of data modeling from presentation devices, which, as pointed out in Section 1.1, are key issues in data modeling.

A wide class of modeling problems investigated in the areas of artificial intelligence and knowledge representation boils down to defining so-called ontologies. An ontology is a set of concepts with a set of relationships between concepts. The concepts represent the basic terms of an application domain, and the meaning of these basic terms must be precisely specified.

The relationships between concepts are domain independent. A very common example is the generalization relationship: it relates a more specific concept with a more general concept, such as rabbit with rodent, rodent with mammal, mammal with animal, car with vehicle, or house with building. Generalization hierarchies, so-called taxonomies, can be found in virtually every application domain. Somewhat less obvious is the fact that there are many other domain-independent relationships, some of them hierarchy-forming. Examples include part-whole relationships such as component-object (finger/hand, hand/arm), member-collection (person/family, family/clan), and agent-result (programmer/program, program/output) relationships, as well as order relationships and similarity relationships. A set of such relationships can represent much of the semantics of a specific domain in a completely domain-independent way.

It is suggestive to map ontologies to hypertext systems, with data items representing concepts and hyperlinks representing relationships between concepts. In this way the hyperlinks are no longer considered as mere contributors to the navigation infrastructure for a browser, but as constructs of formalism for modeling the semantics of data.

1.6 Evolving Subject Indexes

Evolution was first identified in biology, but it is one of the fundamental principles pervading reality. Biology studies a continuously replenished pool of organisms, classified into species according to certain characteristics which some organisms have in common. These characteristics may change over time, requiring the definition of what constitutes a given species to be time dependent and making possible the emergence of subgroups of organisms of the same species, which may start out as races or subspecies and eventually split off to become species of their own.

At this level of generality the issue is hard to grasp from the point of view of knowledge modeling. In most cases it is not even clear which features adequately describe the members of a "pool" and how such features contribute to the classification inside the pool. So let us focus the issue to cases where there is a pool of semi-structured data items that model members of some pool in reality, and some parts of these data items correspond to classification features. For example, such a data item might be a research article with classification features including an author and a date, a list of keywords, and a list of cited articles. It does not matter whether the keywords are provided by the author or automatically extracted from the article using information retrieval methods. Given the classification features, it is possible to establish relationships between data items. In the example domain of research articles, straightforward relationships would be *written by the same author*, *written before*, and *cited by*; and somewhat less straightforward relationships would be *similar subject* and *taken up and advanced by*.

Appropriate numeric measures of the density of such relationships then allow the identification of data items that are *condensation kernels* for classes, and measures of the distance from those condensation kernels might define the boundaries of the classes. Taking into account the relationships along the temporal dimension, one can distinguish condensation kernels at different times and identify metamorphoses of classes [4]. This sketched approach differs from ontologies in several important ways: the condensations kernels and classes cannot be predefined, and membership of data items in classes is fuzzy and time dependent. The research issue would be to integrate numerical and fuzzy notions into the framework of semi-structured data.

1.7 Languages

There is intense activity on query languages for XML-inspired database query languages [5, 6]. Recently, XQuery [7] emerged as the query language of choice for XML. Because XQuery relies upon XPath [8], XQuery is navigational. It is possible to express in XPath/XQuery queries that express redundant (e.g., back and forth)

traversals through an XML document tree. Approaches inspired by or related to logic programming have been proposed, which aim at a more declarative—in the sense of less navigational—query-answering for XML and semistructured data. One might call such approaches *positional* as opposed to *navigational*. This seems to be a very promising direction of research.

1.8 Views

Views are an essential feature of databases, for they ensure the conceptual independence needed in most applications between the primary data stored in the database and various interpretations, such as partial look-ups, of the stored data. Most applications based on databases rely on views. Views also make sense in texts, especially in texts with complex structures and/or contents.

Having different texts for different purposes would present several drawbacks. First, this would induce redundancies. Second, because of these redundancies, the approach would be error prone. Third, the consistency between the different texts giving complementary views of the same content would be difficult to maintain, which is yet another possible source of errors. For these reasons, it is desirable to model a notion of view while specifying the semantics of the considered documents.

1.9 Modeling Biological Data

Modern biology, particularly genomic research, is data and computation intensive. In biology in general and in genomic research in particular, it is a common practice nowadays to build databases of biological data. Most biological databases are—freely or not—accessible through the Web.

From the viewpoint of data modeling, especially of data modeling with markup languages, biological data and biological databases are interesting for several reasons:

- Biological data is subject to both general building laws, the discovery of which is a primary objective of biology, and exceptions. (The admittance of exceptions distinguishes modern markup languages from traditional data modeling formalisms.)
- Biological databases are based upon a multitude of data schemes. For most types of biological data there are no generally accepted data models or ontologies. (The resulting irregularities in structure are another form of exceptions and thus a case for modern markup languages.)
- Most biological databases contain data items that are enriched with texts. Typically such texts explain assumptions made in building up a data item. (Modern markup languages were designed for text in the first place.)
- Sophisticated querying of biological databases is an essential task in searching for laws governing biological data and processes. The querying has to take into account the irregularities because of exceptions, different data models, and enrichments with text.

- Generic markup languages, especially XML, are increasingly being used for modeling biological data.

Note also that most biological databases are, at least potentially, very large. For this reason, biological databases are also an interesting application from the viewpoint of (conventional) database system research and development.

Unfortunately, biological data modeling is rather difficult to understand for most computer scientists. These databases and the research issues they raise are not widely known outside computational biology. This is unfortunate because it prevents a fruitful cross-fertilization between application-driven computer science research, as mostly practiced by biologists and computational biologists, and method-driven computer science research, as practiced by computer science generalists.

An interesting research issue is to investigate, from the general computer science viewpoint, which are the essential aspects of biological data modeling. In this respect, approaches based on description logics seem especially promising. A further interesting research issue is to investigate whether specific querying methods are needed for biological databases.

References

[1] van Herwijnen, E., *Practical SGML*, Boston, MA: Kluwer, 1994.

[2] W3C-XML, "Extensible Markup Language (XML) 1.0," in *W3C Recommendation 16 August 2006, edited in place 29 September 2006*, T. Bray et al., (eds.), 4th ed., World Wide Web Consortium, 2006.

[3] Smith, B., "Ontology," in *Blackwell Guide to the Philosophy of Computing and Information*, L. Floridi, (ed.), Oxford, U.K.: Blackwell, 2003, pp. 155–166.

[4] Conrad, S., et al., "Evolving Logical Specification in Information Systems," in *Logics for Databases and Information Systems*, J. Chomicki and G. Saake, (eds.), Boston, MA: Kluwer, 1998.

[5] Abiteboul, S., P. Buneman, and D. Suciu, *Data on the Web: From Relations to Semistructured Data and XML,* San Mateo, CA: Morgan Kaufmann, 2000.

[6] Fernandez, M., et al., "A Query Language for a Web-Site Management System," *SIGMOD Record*, Vol. 26, 1997, pp. 4–11.

[7] W3C-XQuery, "XQuery 1.0: An XML Query Language," in *W3C Proposed Recommendation 21 November 2006*, S. Boag et al., (eds.), World Wide Web Consortium, 2006.

[8] W3C-XPath, "XML Path Language (XPath) Version 1.0," in *W3C Recommendation 16 November 1999*, J. Clark and S. DeRose, (eds.), World Wide Web Consortium, 1999.

CHAPTER 2

Public Biological Databases for -Omics Studies in Medicine

Viroj Wiwanitkit

The amount of biomedical information grows day by day. At present, the collection and systematization of medical information is developed from collections of hardcopy data in libraries and the collections of electronic data in computers. Several public bioinformatics databases have been developed in recent years. This chapter summarizes the application of public databases in genomics, proteomics, metabolomics, pharmacogenomics, and systomics in medicine.

2.1 Introduction

Computer applications in medicine start from the point of view of the levels of information processing, as suggested by van Herwijnen, and follow current research directions [1]. Current applications cover the subjects of communication, databases, complex calculations (including signal analysis and image analysis techniques), pattern recognition, and expert systems [1]. It is accepted that the use of computer technology in medicine is no longer the domain of only a few gadget-happy high tech aficionados, but rather reflects the rapid pace of medical progress [2]. With this in mind, the Council on Long-Range Planning and Development and the Council on Scientific Affairs of the American Medical Association have developed an informational report on medical informatics [2]. The report states that the technology for producing information about medicine and patients was current, but that the technology for managing this information had not kept up, at least to the extent of it being available in medical facilities where it was needed [2]. The continuing development of the physician as computer user will create a more efficient work environment for the physician while improving patient care at the same time [2]. In addition, a system design that addresses this concern by encouraging information sharing, reducing data duplication, and creating a database to produce needed statistical and management reports is important [3]. In this chapter, application of public databases in comparative genomics, proteomics, metabolomics, pharmacogenomics, and systomics in medicine is reviewed and presented.

2.2 Public Databases in Medicine

Presently, there are a number of generated databases in the field of medicine. Some are accessible to the public, and others are private. The use of public databases in medicine can enhance quality and effectiveness of patient care and the health of the public in general. Renschler [4] said that publications could be retrieved and downloaded anywhere and any time with the introduction of electronic publishing and full text databases becoming available. In addition, study groups for practice-based learning can, by using information technology, prepare themselves for discussions of their problems or of simulated cases which can be systematically provided by central organizations [4]. Renschler also mentioned that a pilot study showed great interest in the application of information technology: 80% of the responding colleagues showed interest in occasional or regular use of medical or nonmedical full text databases, preferably using their own computers. However, Keller [5] noted that the variations of medical practice by private doctors could be due to differences in data they received from public databases. Hence, it is necessary to establish a high-quality database that will allow for statewide peer review, exchange of practice guidelines, and promotion of standardization, all of which can improve outcomes and reduce costs [6]. To solve this problem, what is needed is the development and validation of a data collection system which could be used to establish a database from general practice as a means of health needs assessment as well as for performance review [7]. In addition, Dreyer [8] proposed five guiding principles to help outside contractors facilitate access to third-party data and avoid pitfalls: (1) understand the sponsor's objectives by understanding the purpose of the research; (2) identify and approach data resources that have appropriate information; (3) consider special issues relating to accessing confidential information; (4) establish terms of the research engagement with the sponsor; and (5) establish ground rules with the data provider.

Many open access public databases were launched in the past decade. In the early period, most of public medical databases were designed for literature searching purposes. Health sciences libraries in the United States use the National Library of Medicine (NLM) DOCLINE system to request more than 2 million items annually through interlibrary loan (ILL), and 97% of all ILL requests are for journal articles [9]. The most well-known medical database provided by NLM is Entrez PubMed or Pubmed (www.pubmed.com). This database was developed by the National Center for Biotechnology Information (NCBI) at NLM, located at the National Institutes of Health (NIH). In addition to PubMed, NCBI presently provides other public medical databases for several proposes, including nucleotide and protein sequences, protein structures, complete genomes, taxonomy, Online Mendalian Inheritance in Man (OMIM), and many others (Table 2.1).

Due to a dramatic increase in genomic and proteomic data in public archives in recent years, many public databases in bioinformatics have been launched. Databases and expert systems for several diseases, programs for segregation and linkage analysis, certain DNA and protein sequence databases, and information resources in general for molecular biology are addressed [10]. These systems can be effectively used with newly developed techniques of information exchange based on international computer networks [10]. A summary of some important applications of the bioinformatics databases in medicine will be discussed in the following section.

Table 2.1 Some Public Medical Databases Provided by NCBI

Database	*Usefulness*
Pubmed	Collection of journal literatures
Entrez Genome	Collection of genome of organisms
Entrez Protein	Collection of protein sequences compiled from several sources including SwissProt, PIR, PRF, PDB
Entrez Nucleotide	Collection of sequences from several sources, including GenBank, RefSeq, and PDB
Entrez Structure	Collection of 3D macromolecular structures, including proteins and polynucleotides
OMIM	Collection of human genes and genetic disorders
GENSAT	Collection of expression of genes in the central nervous system of the mouse, using both in situ hybridization and transgenic mouse techniques

2.3 Application of Public Bioinformatics Database in Medicine

2.3.1 Application of Genomic Database

Now with the complete genome sequences of human and other species in hand, detailed analyses of the genome sequences will undoubtedly improve our understanding of biological systems, and public databases become very useful tools [11]. However, because genomic sequences are potential sources of profit for the biotechnology and pharmaceutical industries, many private companies seek to limit access to this information [12]. Marks and Steinberg [12] state that some have argued that this would impede scientific progress and increase the cost of basic research, while others have argued that the privatization of genetic information was needed to assure profits and generate the considerable funding necessary to bring therapeutic products to the market. Controversy over intellectual property rights of the results of large-scale cDNA sequencing raises intriguing questions about the roles of the public and private sectors in genomics research, and about who stands to benefit and who stands to lose from the private appropriation of genomic information [13]. Eisenhaber et al. [13] said that while the U.S. Patent and Trademark Office had rejected patent applications on cDNA fragments of unknown function from the NIH, private firms had pursued three distinct strategies for exploiting unpatented cDNA sequence information: exclusive licensing, nonexclusive licensing, and dedication to public domain. Bentley [14] stated that genomic sequence information should be released and made freely available in the public domain. Marks and Steinberg [12] concluded that both private funding and public access to information are important in genetic research. In addition, Marks and Steinberg noted that precedents for compromise are necessary, as is increased dialog between private and public interests in order to ensure continued advancements in genetic science and medicine.

Many public tools based on genomic databases have been developed (Table 2.2 [15–19]). As we enter the post-genomic era, with the accelerating availability of complete genome sequences, new theoretical approaches, and new experimental techniques, our ability to dissect cellular processes at the molecular level continues to expand [15]. With newly developed tools, many advances have been realized. Recent advances include: (1) the application of RNA interference methods to

Table 2.2 Some Bioinformatics Tools on Genomics and Their Application

Tool	*Application*
1. MICheck [16]	This tool enables rapid verification of sets of annotated genes and frame shifts in previously published bacterial genomes. The Web interface allows one easily to investigate the MICheck results—that is, inaccurate or missed gene annotations: a graphical representation is drawn, in which the genomic context of a unique coding DNA sequence annotation or a predicted frame shift is given, using information on the coding potential (curves) and annotation of the neighboring genes.
2. Pegasys [17]	This tool includes numerous tools for pair-wise and multiple sequence alignment, ab initio gene prediction, RNA gene detection, masking repetitive sequences in genomic DNA as well as filters for database formatting and processing raw output from various analysis tools. It enables biologists and bioinformaticians to create and manage sequence analysis workflows.
3. PartiGene [18]	This tool is an integrated sequence analysis suite that uses freely available public domain software to: (1) process raw trace chromatograms into sequence objects suitable for submission to dbEST; (2) place these sequences within a genomic context; (3) perform customizable first-pass annotation of the data; and (4) present the data as HTML tables and an SQL database resource. It has been used to create a number of nonmodel organism database resources including NEMBASE (http://www.nematodes.org) and LumbriBase (http://www.earthworms.org/). This tool is a Java-based computer application that serves as a workbench for genome-wide analysis through visual interaction. The application deals with various experimental information concerning both DNA and protein sequences (derived from public sequence databases or proprietary data sources) and metadata obtained by various prediction algorithms, classification schemes, or user-defined features. Interaction with a graphical user interface (GUI) allows easy extraction of genomic and proteomic data referring to the sequence itself, sequence features, or general structural and functional features.

characterize loss-of-function phenotype genes in higher eukaryotes; (2) comparative analysis of human and other organism genome sequences; and (3) methods for reconciling contradictory phylogenetic reconstruction [15].

Since sequence databases represent an enormous resource of phylogenetic information, tools are necessary for accessing that information in order to: (1) assess the amount of evolutionary information in these databases that may be suitable for phylogenetic reconstruction and (2) identify areas of taxonomy that are underrepresented for specific gene sequences [20]. In phylogenetic studies, multiple alignments have been used [21].

With the explosion of sequence databases and with the establishment of numerous specialized biological databases, multiple alignment programs must evolve to successfully rise to the new challenges of the post-genomic era [21]. The exploitation of multiple alignments in genome annotation projects represents a qualitative leap in the functional analysis process, bringing the way to reliable phylogenetic analysis [21]. In 2005, MaM (http://compbio.cs.sfu.ca/MAM.htm), a new software tool that processes and manipulates multiple alignments of genomic sequence, was launched by Alkan et al. MaM computes the exact location of common repeat elements, exons, and unique regions within aligned genomics sequences using a variety of user identified programs, databases, and/or tables [22]. The program can extract subalignments, corresponding to these various regions of DNA, to be analyzed independently or in conjunction with other elements of genomic DNA [22]. In addition, the program could facilitate the phylogenetic

analysis and processing of different portions of genomic sequence as part of large-scale sequencing efforts [22].

Studies on the phylogenetic of the pathogen virus are a good application in medicine. For example, genetic analysis of parvovirus B19 has been carried out mainly to establish a framework to track molecular epidemiology of the virus and to correlate sequence variability with different pathological and clinical manifestations of the virus [23]. Gallinella et al. [23] said that most studies showed that the genetic variability of B19 virus was low, and that molecular epidemiology was possible only on a limited geographical and temporal setting. They also said that no clear correlations were present between genome sequence and distinctive pathological and clinical manifestations, but that, more recently, several viral isolates had been identified, showing remarkable sequence diversity with respect to reference sequences [23]. They mentioned that identification of variant isolates added to the knowledge of genetic diversity in this virus group and allowed the identification of three divergent genetic clusters [23]. They proposed that these variant isolates posed interesting questions regarding the real extent of genetic variability in the human erythroviruses, the relevance of these viruses in terms of epidemiology, and their possible implication in the pathogenesis of erythrovirus-related diseases [23]. Many other similar applications are performed in advance research in infectious medicine. Many reports on genetic epidemiology of pathogens are based on the application of genomic databases. In 2005, Lee et al. [24] performed a study to document the prevalence of *Ehrlichia chaffeensis* infection in *Haemaphysalis longicornis* ticks from Korea by PCR sequencing and performed additional phylogenetic analysis based on 16S rRNA gene. In this study, genomic DNAs extracted from 1,288 ticks collected from grass vegetation and various animals from nine provinces of Korea were subjected to screening by nested-PCR based on amplification of 16S rRNA gene fragments [24]. They found that *E. chaffeensis*-specific fragment of 16S rRNA was amplified from 4.2% (26/611) tick samples, and the comparison of the nucleotide sequence of 16S rRNA gene from one tick (EC-PGHL, GenBank accession number AY35042) with the sequences of 20 *E. chaffeensis* strains available in the database showed that EC-PGHL was 100% identical or similar to the Arkansas (AF416764), the Sapulpa (U60476), and the 91HE17 (U23503) strains [24]. Fadiel et al. [25] performed another study which aimed at exploring genomic changes mandated by organismal adaptation to its ecological niches. In this study, coding sequences from three phylogenetically related bacterial species (*Mycoplasma genitalium*, *M. pneumoniae*, and *Ureaplasma urealyticum*) were subject to in-depth sequence analyses [25]. They found that clear similarities in transcriptome structure were identified among the functionally similar species *M. genitalium* and *U. urealyticum*, while no such relationship was identified among the phylogenetically related species *M. genitalium* and *M. pneumoniae* [25]. They concluded that, in these bacterial species, environmental stimuli might be more influential in shaping sequence signatures than phylogenetic relationships and suggested that molecular signatures within the transcriptomes of the species examined were likely to be a product of evolutionary adaptation to diverse environmental ecological stimuli, and not a result of common phylogeny [25].

Indeed, within-species sequence variation data are of special interest since they contain information about recent population/species history, as well as the molecular evolutionary forces currently in action in natural populations [26]. This data, however, is presently dispersed within generalist databases, and is difficult to access [26]. The variation data of human DNA sequence has become more and more useful not only for studying the origin, evolution, and the mechanisms of maintenance of genetic variability in human populations, but also for detection of genetic association in complex diseases such as diabetes, obesity, and hypertension [27]. Phylogenetic foot printing, in which cross-species sequence alignment among orthologous genes is applied to locate conserved sequence blocks, is an effective strategy to attack this problem [28]. Single nucleotide polymorphisms (SNPs) in sequence contribute to the heterogeneity and might disrupt or enhance their regulatory activity. The study of SNPs will help in functional evaluation of SNPs. Many genomic databases have been launched for study of SNPs. For example, Bazin et al. [26] recently proposed Polymorphix (http://pbil.univlyon1.fr/polymorphix/query.php), a database dedicated to sequence polymorphism, containing sequences from the nuclear, mitochondrial, and chloroplastic genomes of every eukaryote species represented in EMBL. This tool contains within-species homologous sequence families built using EMBL/GenBank under suitable similarity and bibliographic criteria [26]. It is a structured database allowing both simple and complex queries for population genomic studies: alignments within families as well as phylogenetic trees can be downloaded [26]. Zhao et al. [28] proposed another tool, PromoLign (http://polly.wustl.edu/promolign/main.html), an online database application that presents SNPs and transcriptional binding profiles in the context of human-mouse orthologous sequence alignment with a hyperlink graphical interface. This tool could be applied to a variety of SNPs and transcription-related studies, including association genetics, population genetics, and pharmacogenomics [28]. In addition, other databases such as dbSNP, CGAP, HGBASE, JST, and Go!Poly have been developed to collect and exploit data of SNPs in the United States, Europe, Japan, and China [27].

The massive amount of SNP data stored at public Internet sites provides unprecedented access to human genetic variation [29]. Since decision rules for the selection of functionally relevant SNPs are not available, selecting target SNPs for disease-gene association studies is, at present, usually done randomly. Luckily, Wjst [29] recently implemented a computational pipeline that retrieves the genomic sequence of target genes, collects information about sequence variation, and selects functional motifs containing SNPs. There are some recent studies on SNPs based on application of genomic database. Ingersoll et al. [30] studied fibroblast growth factor receptors (FGFRs), of which mutations in FGFR2 cause more than five craniosynostosis syndromes. They refined and extended the genomic organization of the FGFR2 gene by sequencing more than 119 kb of PACs, cosmids, and PCR products and assembling a region of approximately 175 kb [30], and then compared between their derived sequence and those in the NCBI database. According to this study, they detected more than 300 potential SNPs [31]. In 2005, Levran et al. [31] summarized all sequence variations (mutations and polymorphisms) in FANCA described in the literature and listed in the Fanconi Anemia Mutation Database, and reported 61 novel FANCA mutations identified in Fanconi anemia patients

registered in the International Fanconi Anemia Registry (IFAR). In this study, 38 novel SNPs, previously unreported in the literature or in dbSNP, were also identified [31]. Twenty-two large genomic deletions were identified by detection of apparent homozygosity for rare SNPs, and a conserved SNP haplotype block spanning at least 60 kb of the FANCA gene was identified in individuals from various ethnic groups [31]. Levran et al. [31] mentioned that FANCA SNP data was highly useful for carrier testing, prenatal diagnosis, and preimplantation genetic diagnosis, particularly when the disease-causing mutations were unknown.

Wjst [29] found that most of the SNPs were disrupting transcription factor binding sites but that only those introducing new sites had a significant depressing effect on SNP allele frequency. Wjst [29] noted that only 10% of all gene-based SNPs have sequence-predicted functional relevance, making them a primary target for genotyping in association studies. Genetic fine mapping technique is also applied for studies of some specific diseases. This would allow scientists to focus subsequent laboratory studies on genomic regions already related to diseases by other scientific methods [32]. Indeed, microarray gene expression studies and associated genetic mapping studies in many diseases would benefit from a generalized understanding of the prior work associated with those diseases [28]. Ferraro et al. [33] mapped a gene for severe pediatric gastroesophageal reflux disease (GERD1) to a 9-cM interval on chromosome 13q14 and presented the results of DNA sequencing and allelic association analyses that were done in an attempt to clone the GERD1 gene. Using a candidate transcript approach, Ferraro et al. screened affected individuals for mutations in all transcribed regions of all genes, putative genes, and ESTs identified within the 6.2-Mb GERD1 locus based on alignments with the GenBank cDNA databases. In this work, from a total of 50 identifiable genes and 99 EST clusters in the GERD1 locus, we identified 163 polymorphisms (143 SNPs and 20 INDELs) in 21 genes and 37 ESTs [32]. Ferraro et al. [33] concluded that the patterns of inheritance and/or the high population frequencies of all polymorphic alleles identified in this study argued against causative relationships between any of the alleles and the GERD phenotype. Hu et al. [34] confirmed a seizure-related QTL of large effect on mouse Chr 1 and mapped it to a finely delimited region. In their study, they compared the coding region sequences of candidate genes between B6 and D2 mice using RT-PCR, amplification from genomic DNA, and database searching, and discovered 12 brain-expressed genes with SNPs that predicted a protein amino acid variation [34]. They found that the most compelling seizure susceptibility candidate was Kcnj10 [34]. They concluded that the critical interval contained several candidate genes, one of which, Kcnj10, exhibited a potentially important polymorphism with regard to fundamental aspects of seizure susceptibility [34].

In order to take full advantage of the newly available public human genome sequence data and associated annotations, visualization tools that can accommodate the high frequency of alternative splicing in human genes and other complexities are required [35]. The identification and study of evolutionarily conserved genomic sequences that surround disease-related genes is a valuable tool to gain insight into the functional role of these genes and to better elucidate the pathogenetic mechanisms of disease [36]. The visualization techniques for presenting human genomic sequence data and annotations in an interactive, graphical for-

mat include: (1) one-dimensional semantic zooming to show sequence data alongside gene structures; (2) color-coding exons to indicate frame of translation; (3) adjustable, moveable tiers to permit easier inspection of a genomic scene; and (4) display of protein annotations alongside gene structures to show how alternative splicing impacts protein structure and function [35]. These techniques are illustrated using examples from two genome browser applications: the Neomorphic GeneViewer annotation tool and ProtAnnot, a prototype viewer which shows protein annotations in the context of genomic sequence [35]. Etim et al. [32] developed a database of prostate cancer–related chromosomal information from the existing biomedical literature, named ChromSorter PC. In this work, the input material was based on a broad literature search with subsequent hand annotation of information relevant to prostate cancer [32]. Etim et al. [32] used this database to present graphical summaries of chromosomal regions associated with prostate cancer broken down by age, ethnicity, and experimental method. In addition, Etim et al. [32] placed the database information on the human genome using the Generic Genome Browser tool (http://www.prostategenomics.org/datamining/chromsorter_pc.html), which allowed the visualization of the data with respect to user generated datasets. Boccia et al. [36] recently created the Disease Gene Conserved Sequence Tags (DG-CST) database (http://dgcst.ceinge.unina.it/) for the identification and detailed annotation of human-mouse conserved genomic sequences that were localized within or in the vicinity of human disease-related genes. The database contains CST data, defined as sequences that show at least 70% identity between human and mouse over a length of at least 100 bp, relative to more than 1,088 genes responsible for monogenetic human genetic diseases or involved in the susceptibility to multifactorial/polygenic diseases; and this data might be searched using both simple and complex queries [36]. In addition, this tool includes a graphic browser that allows direct visualization of the CSTs and related annotations within the context of the relative gene and its transcripts [36].

2.3.2 Application of Proteomic Database

Proteomics has rapidly become an important tool for life science research, allowing the integrated analysis of global protein expression from a single experiment [37]. Previously, to accommodate the complexity and dynamic nature of any proteome, researchers had to use a combination of disparate protein biochemistry techniques—often a highly involved and time-consuming process [37]. Now, however, there are many attempts to develop new bioinformatics tools to solve this problem. The development of high-throughput proteomic platforms, which encompass all aspects of proteome analysis and are integrated with genomics and bioinformatics technology, therefore represents a crucial step for the advancement of proteomics research [37]. One of the most well-known proteomic databases is the Swiss-Prot protein knowledge base [38]. This database provides manually annotated entries for all species, but concentrates on the annotation of entries from model organisms to ensure the presence of high quality annotation of representative members of all protein families [38]. The numerous software tools provided on the Expert Protein Analysis System (ExPASy) Web site might help to identify and reveal the function of proteins [39]. In addition to bibliographic references, experimental

results, computed features, and sometimes even contradictory conclusions, direct links to specialized databases connect amino acid sequences with the current knowledge in sciences [38]. Recently, a single, centralized, authoritative resource for protein sequences and functional information, UniProt, was created by joining the information contained in Swiss-Prot, Translation of the EMBL nucleotide sequence (TrEMBL), and the Protein Information Resource–Protein Sequence Database (PIR-PSD) [38]. A rising problem, however, is that an increasing number of nucleotide sequences are not being submitted to the public databases, and thus the proteins inferred from such sequences will have difficulties finding their way to the Swiss-Prot or TrEMBL databases [38]. Many other public tools based on proteomic databases have been developed (Table 2.3) [40, 41]. With these newly developed tools, many advances are being generated.

Proteomics, which identifies proteins and analyzes their function in cells, is foreseen as the next challenge in biomedicine, as diseases in the body are most easily recognized through the function of their proteins [42]. Achieving this recognition, however, is more difficult than pure gene analysis since it is estimated that 35,000 genes are present in human DNA, encoding more than 1 million proteins [42]. In medicine, many proteomic databases have been generated. The integration of genomic and proteomic data will help to elucidate the functions of proteins in the pathogenesis of diseases and the aging process, and they could lead to the discovery of novel drug target proteins and biomarkers of diseases [43, 44]. Of the proteomics for several diseases, cancer proteomics are very important; because of these proteomics, medical scientists can understand the protein profiles during the different stages of the tumorigenesis, and this has brought a new hope for discovery of the tumor-specific biomarkers [44]. Techniques that allow data mining from a large input database overcome the slow advances of one protein–one gene investigation and further address the multifaceted carcinogenesis process occurring even in cell line mutation-associated malignancy [45]. Proteomics, the study of the cellular proteins and their activation states, has led the progress in biomarker development for cancers and is being applied to management assessment [45]. Kohn et al. [46] said that more achievements have been made and many promising

Table 2.3 Some Bioinformatics Tools on Proteomics and Their Application

Tool	*Application*
1. ProteomIQ [37]	This tool is the combination into a single suite of integrated hardware and software of sample preparation and tracking, centralized data acquisition and instrument control, and direct interfacing with genomics and bioinformatics databases tools. It is useful for the analysis of proteins separated by 2D polyacrylamide gel electrophoresis.
2. BioBuilder [40]	This tool is a Zope-based software tool that was developed to facilitate intuitive creation of protein databases. Protein data can be entered and annotated through web forms along with the flexibility to add customized annotation features to protein entries. A built-in review system permits a global team of scientists to coordinate their annotation efforts.
3. DBParser [41]	This tool is for rapidly culling, merging, and comparing sequence search engine results from multiple LC-MS/MS peptide analyses. It employs the principle of parsimony to consolidate redundant protein assignments and derive the most concise set of proteins consistent with all of the assigned peptide sequences observed in an experiment or series of experiments.

candidates of tumor-markers have been identified, even though most of them have not yet been affirmed with the recent use of proteomic tools with the laser capture microdissection (LCM) technique. Sorace and Zhan [47] recently performed a data review and reassessment of ovarian cancer serum proteomic profiling. They analyzed the Ovarian Dataset 8-7-02 downloaded from the Clinical Proteomics Program Databank Web site, using nonparametric statistics and stepwise discriminant analysis to develop rules to diagnose patients and to understand general patterns in the data that might guide future research [47]. Le Naour [48] said that proteomics could allow serological screening of tumor antigens. Le Naour [48] noted that proteins eliciting humoral response in cancer could be identified by 2D Western blot using cancer patient sera, followed by mass spectrometry analysis and database search. Le Naour [48] mentioned that application of this principle to different types of cancer could allow us to define several tumor antigens and that the common occurrence of auto-antibodies to certain of these proteins in different cancers might be useful in cancer screening and diagnosis, as well as for immunotherapy.

For infectious diseases, the combined technologies of genomics, proteomics, and bioinformatics has provided valuable tools for the study of complex phenomena determined by the action of multiple gene sets in the study of pathogenic bacteria [49]. There are some recent developments in the establishment of proteomic databases as well as attempts to define pathogenic determinants at the level of the proteome for some of the major human pathogens [49]. Proteomics can also provide practical applications through the identification of immunogenic proteins that may be potential vaccine targets, as well as in extending our understanding of antibiotic action [49]. Cash [49] noted that there was little doubt that proteomics has provided us with new and valuable information on bacterial pathogens and would continue to be an important source of information in the coming years. In additional to bacteria, other problematic organisms such as fungus have also been studied based on proteomic database tools [50].

Another interesting application of the public proteomic database in medicine is its use as a tool for molecular structure studies. Modeling of secondary, tertiary, and quaternary structures of proteins can be performed based on the data on their amino acid sequences in the proteomic database. A good example is the structural modeling for the uncommon hemoglobin disorders. Wiwanitkit [51, 52] recently reported the secondary and tertiary structures of hemoglobin Suandok from the structural analysis based on the data from the public proteomic database. According to these studies, the significant aberration in the secondary [51] and tertiary structures [52] of hemoglobin Suandok could not be demonstrated.

2.3.3 Application of the Metabolomics Database

Generally, a large proportion of the genes in any genome encode enzymes of primary and specialized (secondary) metabolism [53]. Not all primary metabolites—those that are found in all or most species—have been identified, and only a small portion of the estimated hundreds of thousand specialized metabolites—those found only in restricted lineages—have been studied in any species [54]. Fridman and Pichersky [53] noted that the correlative analysis of extensive metabolic profiling and gene

expression profiling have proven to be a powerful approach for the identification of candidate genes and enzymes, particularly those in secondary metabolism [55]. It is rapidly becoming possible to measure hundreds or thousands of metabolites in small samples of biological fluids or tissues. Arita [55] said that metabolomics, a comprehensive extension of traditional targeted metabolite analysis, has recently attracted much attention as the biological jigsaw puzzle's missing piece because it can complement transcriptome and proteome analysis. Metabolic profiling applied to functional genomics (metabolomics) is in an early stage of development [56]. Fridman and Pichersky [53] said that the final characterization of substrates, enzymatic activities, and products requires biochemical analysis, which have been most successful when candidate proteins have homology to other enzymes of known function. To facilitate the analysis of experiments using post-genomic technologies, new concepts for linking the vast amount of raw data to a biological context have to be developed [57]. Visual representations of pathways help biologists to understand the complex relationships between components of metabolic network [57].

Since metabolomics is new, the associated database tool (Table 2.4) [58], as well as the application of the metabolomics database in medicine, is still limited. German et al. [54] noted that the metabolomics made it possible to assess the metabolic component of nutritional phenotypes and would allow individualized dietary recommendations. German et al. [54] proposed that the American Society for Nutritional Science (ASNS) had to take action to ensure that appropriate technologies were developed and that metabolic databases were constructed with the right inputs and organization. German et al. [54] also mentioned that the relations between diet and metabolomic profiles and between those profiles and health and disease should be established.

2.3.4 Application of Pharmacogenomics Database

Pharmacogenomics is defined to identify the genes that are involved in determining the responsiveness—and to distinguish responders and nonresponders—to a given drug [59]. Genome sequencing, transcriptome, and proteome analysis are of particular significance in pharmacogenomics [59]. Sequencing is used to locate polymorphisms, and monitoring of gene expression can provide clues about the genomic response to disease and treatment [59]. Thallinger et al. [60] suggested that the development of a pharmacogenomics data management system that integrates public and proprietary databases, clinical datasets, and data mining tools embedded in a high-performance computing environment should include the following components: parallel processing systems, storage technologies, network technologies, databases and database management systems (DBMS), and application services. In pharmacogenomics, several primary databases are being generated to understand fundamental biology, identify new drug targets, and look

Table 2.4 A Bioinformatics Tool on Metabolomics and Its Application

Tool	*Application*
MSFACTs [58]	This tool is for metabolomics spectral formatting, alignment, and conversion.

at compound profiling in a new light [61]. Many public tools based on pharmacogenomics databases have been developed [62] (Table 2.5). Although great strides have been made in understanding the diversity of the human genome—such as the frequency, distribution, and type of genetic variation that exists—the feasibility of applying this information to uncover useful pharmacogenomics markers is uncertain [63]. The health care industry is clamoring for access to SNP databases for use in research in the hope of revolutionizing the drug development process which will be important for determining the applicability of pharmacogenomics information to medical practice [63–65].

For a few years, molecular pharmacology has focused on the relationship between patterns of gene expression and patterns of drug activity. Using a systematic substructure analysis coupled with statistical correlations of compound activity with differential gene expression, Blower et al. [66] identified two subclasses of quinones whose patterns of activity in the National Cancer Institute's 60-cell line screening panel (NCI-60) correlate strongly with the expression patterns of particular genes: (1) the growth inhibitory patterns of an electron-withdrawing subclass of benzodithiophenedione-containing compounds over the NCI-60 are highly correlated with the expression patterns of Rab7 and other melanoma-specific genes; and (2) the inhibitory patterns of indolonaphthoquinone-containing compounds are highly correlated with the expression patterns of the hematopoietic lineage-specific gene HS1 and other leukemia genes. Blower et al. [66] also noted that the approach (SAT, for Structure-Activity-Target) provided a systematic way to mine databases for the design of further structure-activity studies, particularly to aid in target and lead identification. The application of pharmacogenomics database in oncology is of interest at present. Indeed, the association of transporter proteins and cancer drug resistance has been known for approximately 25 years, with recent discoveries pointing to an ever-increasing number of ATP binding cassette (ABC) transporter proteins involved with the response of cancer cells to pharmacotherapy [67]. Microarray-based expression profiling studies in the field of oncology have demonstrated encouraging correlations between tumor transcriptional profiles and eventual patient outcomes [68]. These findings bring great interest in the application

Table 2.5 Some Bioinformatics Tools on Pharmacogenomics and Their Application

Tool	*Application*
VizStruct [64]	This tool uses the first harmonic of the discrete Fourier transform to map multidimensional data to two dimensions for visualization. The mapping is used to visualize several published pharmacokinetic, pharmacodynamic, and pharmacogenomics data sets. It is a computationally efficient and effective approach for visualizing complex, multidimensional data sets. It could have many useful applications in the pharmaceutical sciences.
Exploratory Visual Analysis (EVA) [65]	This tool demonstrates its utility in replicating the findings of an earlier pharmacogenomics study as well as elucidating novel biologically plausible hypotheses. It brings all of the often disparate pieces of analysis together in an infinitely flexible visual display that is amenable to any type of statistical result and biological question.

of transcriptional profiling to samples available from real-time clinical trials, and clinical pharmacogenomics objectives utilizing transcriptional profiling strategies are becoming increasingly incorporated into clinical trial study designs [68]. Burczynski et al. [68] said that strategic implementation of transcriptional profiling in early oncology clinical trials could provide an opportunity to identify predictive markers of clinical response and eventually provide a substantial step forward towards the era of personalized medicine.

2.3.5 Application of Systomics Database

Systemic biology is a complexion in advance medicine [69]. Systomics is the new concept in "omic" science. The collections of metabolomics or systomics lead the next revolution in human biology. The physionomics or combination of molecular (functional genomics, transcriptomics), biochemical (proteomics), and analytical (metabolomics) approaches are still particularly discussed at present [70]. The application of systomics database in medicine is still limited.

References

[1] Hasman, A., "Medical Applications of Computers: An Overview," *Int. J. Biomed. Comput.,* Vol. 20, No. 4, 1987, pp. 239–251.

[2] American Medical Association, "Medical Informatics: An Emerging Medical Discipline," Council on Scientific Affairs and Council on Long Range Planning and Development of the American Medical Association, *J. Med. Syst.,* Vol. 14, No. 4, 1990, pp. 161–179.

[3] Sulton, L. D., et al., "Computerized Data Bases: An Integrated Approach to Monitoring Quality of Patient Care," *Arch. Phys. Med. Rehabil.,* Vol. 68, No. 12, 1987, pp. 850–853.

[4] Renschler, H. E., "Rational Continuing Medical Education," *Schweiz. Rundsch. Med. Prax.*, Vol. 80, No. 19, 1991, pp. 515–523.

[5] Keller, R. B., "Public Data and Private Doctors: Maine Tackles Treatment Variations," *J. State. Gov.*, Vol. 64, No. 3, 1991, pp. 83–86.

[6] Arom, K. V., et al., "Establishing and Using a Local/Regional Cardiac Surgery Database," *Ann. Thorac. Surg.*, Vol. 64, No. 5, 1997, pp. 1245–1249.

[7] Paris, J. A., A. P. Wakeman, and R. K. Griffiths, "General Practitioners and Public Health," *Public Health,* Vol. 106, No. 5, 1992, pp. 357–366.

[8] Dreyer, N. A., "Accessing Third-Party Data for Research: Trust Me? Trust Me Not?" *Pharmacoepidemiol. Drug Saf.,* Vol. 10, No. 5, 2001, pp. 385–388.

[9] Lacroix, E. M., "Interlibrary Loan in U.S. Health Sciences Libraries: Journal Article Use," *Bull. Med. Libr. Assoc.,* Vol. 82, No. 4, 1994, pp. 363–368.

[10] Fischer, C., et al., "Programs, Databases, and Expert Systems for Human Geneticists—A Survey," *Hum. Genet.*, Vol. 97, No. 2, 1996, pp. 129–137.

[11] Yu, U., et al., "Bioinformatics in the Post-Genome Era," *J. Biochem. Mol. Biol.,* Vol. 37, No. 1, 2004, pp. 75–82.

[12] Marks, A. D., and K. K. Steinberg, "The Ethics of Access to Online Genetic Databases: Private or Public?" *Am. J. Pharmacogenomics,* Vol. 2, No. 3, 2002, pp. 207–212.

[13] Eisenhaber, F., B. Persson, and P. Argos, "Protein Structure Prediction: Recognition of Primary, Secondary, and Tertiary Structural Features from Amino Acid Sequence," *Crit. Rev. Biochem. Mol. Biol.,* Vol. 30, 1995, pp. 1–94.

[14] Bentley, D. R., "Genomic Sequence Information Should Be Released Immediately and Freely in the Public Domain," *Science,* Vol. 274, No. 5287, 1996, pp. 533–534.

[15] Karlin, S., J. Mrazek, and A. J. Gentles, "Genome Comparisons and Analysis," *Curr. Opin. Struct. Biol.*, Vol. 13, No. 3, 2003, pp. 344–352.

[16] Cruveiller, S., et al., "MICheck: A Web Tool for Fast Checking of Syntactic Annotations of Bacterial Genomes," *Nucleic Acids Res.,* Vol. 33, 2005, pp. W471–W479.

[17] Shah, S. P., et al., "Pegasys: Software for Executing and Integrating Analyses of Biological Sequences," *BMC Bioinformatics,* Vol. 5, No. 1, 2004, p. 40.

[18] Parkinson, J., et al., "PartiGene—Constructing Partial Genomes," *Bioinformatics,* Vol. 20, No. 9, 2004, pp. 1398–1404.

[19] Vernikos, G. S., et al., "GeneViTo: Visualizing Gene-Product Functional and Structural Features in Genomic Datasets," *BMC Bioinformatics,* Vol. 4, No. 1, 2003, p. 53.

[20] Jakobsen, I. B., et al., "TreeGeneBrowser: Phylogenetic Data Mining of Gene Sequences from Public Databases," *Bioinformatics,* Vol. 17, No. 6, 2001, pp. 535–540.

[21] Lecompte, O., et al., "Multiple Alignment of Complete Sequences (MACS) in the Post-Genomic Era," *Gene,* Vol. 30, No. 270, Iss. 1-2, 2001, pp. 17–30.

[22] Alkan, C., et al., "Manipulating Multiple Sequence Alignments Via MaM and WebMaM," *Nucleic Acids Res.*, Vol. 33, 2005, pp. W295–W298.

[23] Gallinella, G., et al., "B19 Virus Genome Diversity: Epidemiological and Clinical Correlations," *J. Clin. Virol.,* Vol. 28, No. 1, 2003, pp. 1–13.

[24] Lee, S. O., et al., "Identification and Prevalence of Ehrlichia Chaffeensis Infection in Haemaphysalis Longicornis Ticks from Korea by PCR, Sequencing and Phylogenetic Analysis Based on 16S rRNA Gene," *J. Vet. Sci.,* Vol. 6, No. 2, 2005, pp. 151–155.

[25] Fadiel, A., S. Lithwick, and F. Naftolin, "The Influence of Environmental Adaptation on Bacterial Genome Structure," *Lett. Appl. Microbiol.*, Vol. 40, No. 1, 2005, pp. 12–18.

[26] Bazin, E., et al., "Polymorphix: A Sequence Polymorphism Database," *Nucleic Acids Res.,* Vol. 33, 2005, pp. D481–484.

[27] Gu, H. F., "Single Nucleotide Polymorphisms (SNPs) and SNP Databases," *Zhonghua. Yi. Xue. Yi. Chuan. Xue. Za. Zhi.,* Vol. 18, No. 6, 2001, pp. 479–481.

[28] Zhao, T., et al., "PromoLign: A Database for Upstream Region Analysis and SNPs," *Hum. Mutat.*, Vol. 23, No. 6, 2004, pp. 534–539.

[29] Wjst, M., "Target SNP Selection in Complex Disease Association Studies," *BMC Bioinformatics,* Vol. 5, No. 1, 2004, p. 92.

[30] Ingersoll, R. G., et al., "Fibroblast Growth Factor Receptor 2 (FGFR2): Genomic Sequence and Variations," *Cytogenet. Cell. Genet.*, Vol. 94, No. 3-4, 2001, pp. 121–126.

[31] Levran, O., et al., "Spectrum of Sequence Variations in the FANCA Gene: An International Fanconi Anemia Registry (IFAR) Study," *Hum. Mutat.*, Vol. 25, No. 2, 2005, pp. 142–149.

[32] Etim, A., et al., "ChromSorter PC: A Database of Chromosomal Regions Associated with Human Prostate Cancer," *BMC Genomics,* Vol. 28, No. 5, 2004, p. 27.

[33] Ferraro, T. N., et al., "Fine Mapping of a Seizure Susceptibility Locus on Mouse Chromosome 1: Nomination of Kcnj10 as a Causative Gene," *Mamm. Genome,* Vol. 15, No. 4, 2004, pp. 239–251.

[34] Hu, F. Z., et al., "Fine Mapping a Gene for Pediatric Gastroesophageal Reflux on Human Chromosome 13q14," *Hum. Genet.,* Vol. 114, No. 6, 2004, pp. 562–572.

[35] Loraine, A. E., and G. A. Helt, "Visualizing the Genome: Techniques for Presenting Human Genome Data and Annotations," *BMC Bioinformatics,* Vol. 30, No. 1, 2002, p. 19.

[36] Boccia, A., et al., "DG-CST (Disease Gene Conserved Sequence Tags), a Database of Human-Mouse Conserved Elements Associated to Disease Genes," *Nucleic. Acids. Res.,* Vol. 33, 2005, pp. D505–D510.

[37] Stephens, A. N., P. Quach, and E. J. Harry, "A Streamlined Approach to High-Throughput Proteomics," *Expert. Rev. Proteomics,* Vol. 2, No. 2, 2005, pp. 173–185.

[38] Schneider, M., M. Tognolli, and A. Bairoch, "The Swiss-Prot Protein Knowledgebase and ExPASy: Providing the Plant Community with High Quality Proteomic Data and Tools," *Plant. Physiol. Biochem.,* Vol. 42, No. 12, 2004, pp. 1013–1021.

[39] Eisenberg, R. S., "Intellectual Property Issues in Genomics," *Trends Biotechnol.,* Vol. 14, No. 8, 1996, pp. 302–307.

[40] Navarro, J. D., et al., "BioBuilder as a Database Development and Functional Annotation Platform for Proteins," *BMC Bioinformatics,* Vol. 5, No. 1, 2004, p. 43.

[41] Yang, X., et al., "DBParser: Web-Based Software for Shotgun Proteomic Data Analyses," *J. Proteome. Res*., Vol. 3, No. 5, 2004, pp. 1002–1008.

[42] Haley-Vicente, D., and D. J. Edwards, "Proteomic Informatics: In Silico Methods Lead to Data Management Challenges," *Curr. Opin. Drug. Discov. Devel.,* Vol. 6, No. 3, 2003, pp. 322–332.

[43] Lau, A. T., Q. Y. He, and J. F. Chiu, "Proteomic Technology and Its Biomedical Applications," *Sheng. Wu. Hua. Xue. Yu. Sheng. Wu. Wu. Li. Xue. Bao. (Shanghai),* Vol. 35, No. 11, 2003, pp. 965–975.

[44] Lopez, M. F., "Proteomic Databases: Roadmaps for Drug Discovery," *Am. Clin. Lab.,* Vol. 17, No. 10, 1998, pp. 16–18.

[45] Ni, X. G., et al., "Application of Proteomic Approach for Solid Tumor Marker Discovery," *Ai. Zheng.,* Vol. 22, No. 6, 2003, pp. 664–667.

[46] Kohn, E. C., G. B. Mills, and L. Liotta, "Promising Directions for the Diagnosis and Management of Gynecological Cancers," *Int. J. Gynaecol. Obstet*., Vol. 83, Suppl. 1, 2003, pp. 203–209.

[47] Sorace, J. M., and M. A. Zhan, "Data Review and Re-Assessment of Ovarian Cancer Serum Proteomic Profiling," *BMC Bioinformatics,* Vol. 4, No. 1, 2003, p. 24.

[48] Le Naour, F., "Contribution of Proteomics to Tumor Immunology," *Proteomics,* Vol. 1, No. 10, 2001, pp. 1295–1302.

[49] Cash, P., "Proteomics of Bacterial Pathogens," *Adv. Biochem. Eng. Biotechnol.,* Vol. 83, 2003, pp. 93–115.

[50] Pitarch, A., et al., "Analysis of the Candida Albicans Proteome. II. Protein Information Technology on the Net (Update 2002)," *J. Chromatogr. B. Analyt. Technol. Biomed. Life. Sci.,* Vol. 787, No. 1, 2003, pp. 129–148.

[51] Wiwanitkit, V., "Is There Aberration in the Secondary Structure of Globin Chain in Haemoglobin Suan-Dok Disorder?" *Haema.,* Vol. 8, No. 3, 2005, pp. 427–429.

[52] Wiwanitkit, V., "Modeling for Tertiary Structure of Globin Chain in Hemoglobin Suandok," *Hematology,* Vol. 10, 2005, pp. 163–165.

[53] Fridman, E., and E. Pichersky, "Metabolomics, Genomics, Proteomics, and the Identification of Enzymes and Their Substrates and Products," *Curr. Opin. Plant. Biol.,* Vol. 8, No. 3, 2005, pp. 242–248.

[54] German, J. B., et al., "Metabolomics in the Opening Decade of the 21st Century: Building the Roads to Individualized Health," *J. Nutr.,* Vol. 134, No. 10, 2004, pp. 2729–2732.

[55] Arita, M., "Additional Paper: Computational Resources for Metabolomics," *Brief. Funct. Genomic. Proteomic,* Vol. 3, No. 1, 2004, pp. 84–93.

[56] Mendes, P., "Emerging Bioinformatics for the Metabolome," *Brief. Bioinform.,* Vol. 3, No. 2, 2002, pp. 134–145.

[57] Lange, B. M., and M. Ghassemian, "Comprehensive Post-Genomic Data Analysis Approaches Integrating Biochemical Pathway Maps," *Phytochemistry,* Vol. 66, No. 4, 2005, pp. 413–451.

[58] Duran, A. L., et al., "Metabolomics Spectral Formatting, Alignment and Conversion Tools (MSFACTs)," *Bioinformatics,* Vol. 19, No. 17, 2003, pp. 2283–2293.

[59] Tanaka, T., et al., "Pharmacogenomics and Pharmainformatics," *Nippon. Rinsho.,* Vol. 60, No. 1, 2002, pp. 39–50.

[60] Thallinger, G. G., et al., "Information Management Systems for Pharmacogenomics," *Pharmacogenomics,* Vol. 3, No. 5, 2002, pp. 651–667.

[61] Furness, L. M., S. Henrichwark, and M. Egerton, "Expression Databases—Resources for Pharmacogenomics R&D," *Pharmacogenomics,* Vol. 1, No. 3, 2000, pp. 281–288.

[62] De La Vega, F. M., et al., "New Generation Pharmacogenomics Tools: A SNP Linkage Disequilibrium Map, Validated SNP Assay Resource, and High-Throughput Instrumentation System for Large-Scale Genetic Studies," *Biotechniques,* Suppl., 2002, pp. 48–50, 52, 54.

[63] McCarthy, J. J., and R. Hilfiker, "The Use of Single-Nucleotide Polymorphism Maps in Pharmacogenomics," *Nat. Biotechnol.,* Vol. 18, No. 5, 2000, pp. 505–508.

[64] Bhasi, K., et al., "Analysis of Pharmacokinetics, Pharmacodynamics, and Pharmacogenomics Data Sets Using VizStruct, a Novel Multidimensional Visualization Technique," *Pharm. Res.,* Vol. 21, No. 5, 2004, pp. 777–780.

[65] Reif, D. M., et al., "Exploratory Visual Analysis of Pharmacogenomics Results," *Pac. Symp. Biocomput.,* Vol. 1, 2005, pp. 296–307.

[66] Blower, P. E., et al., "Pharmacogenomics Analysis: Correlating Molecular Substructure Classes with Microarray Gene Expression Data," *Pharmacogenomics J.,* Vol. 2, No. 4, 2002, pp. 259–271.

[67] Ross, D. D., and L. A. Doyle, "Mining Our ABCs: Pharmacogenomics Approach for Evaluating Transporter Function in Cancer Drug Resistance," *Cancer. Cell.,* Vol. 6, No. 2, 2004, pp. 105–107.

[68] Burczynski, M. E., et al., "Clinical Pharmacogenomics and Transcriptional Profiling in Early Phase Oncology Clinical Trials," *Curr. Mol. Med.,* Vol. 5, No. 1, 2005, pp. 83–102.

[69] Witkamp, R. F., "Genomics and Systems Biology—How Relevant Are the Developments to Veterinary Pharmacology, Toxicology and Therapeutics?" *J. Vet. Pharmacol. Ther.,* Vol. 28, No. 3, 2005, pp. 235–245.

[70] Grossmann, K., "What It Takes to Get a Herbicide's Mode of Action. Physionomics: A Classical Approach in a New Complexion," *Pest. Manag. Sci.,* Vol. 61, No. 5, 2005, pp. 423–431.

CHAPTER 3

Modeling Biomedical Data

Ramez Elmasri, Feng Ji, and Jack Fu

Biological data such as protein structure and function, DNA sequences, and metabolic pathways require conceptual modeling characteristics that are not available in traditional conceptual modeling, such as in the widely used entity-relationship (ER) model and its variant, the enhanced-ER (EER) model. In particular, there are three constructs that occur frequently in bioinformatics data: ordered relationships, functional processes, and three-dimensional structures. In addition, biological data modeling requires many levels of abstraction, from the DNA/RNA level to higher abstraction levels such as cells, tissues, organs, and biological systems. In this chapter, we discuss some of the concepts that are needed for accurate modeling of biological data. We suggest changes to the EER model to extend it for modeling some of these concepts by introducing specialized formal relationships for ordering, processes, and molecular spatial structure. These changes would facilitate more accurate modeling of biological structures and ontologies, and they can be used in mediator and integrative systems for biological data sources. We propose new EER schema diagram notation to represent the ordering of DNA sequences, the three-dimensional structure of proteins, and the processes of metabolic pathways. We also show how these new concepts can be implemented in relational databases. Finally, we discuss why multilevel modeling would enhance the integration of biological models at different levels of abstraction, and we discuss some preliminary ideas to realize a multilevel model.

3.1 Introduction

Large quantities of biological data are being produced at a phenomenal rate. For example, the GenBank repository of nucleotide sequences, and their encoded proteins, has an exponential rate of increase for new entries from 1 million sequences in 1996 to 46 million in 2005 [1]. To our best knowledge, most molecular biological databases have evolved from legacy systems and they lack many good practices of modern database systems [2]. Also, biological data is inherently complicated in its content, format, meaning, and sources [3, 4]. Storing, retrieving, and analyzing this data requires a suitable data model. However, many traditional

data models do not contain sufficient concepts to model frequently occurring phenomena in biological data. Some work has been done on the conceptual enhancement of data models to accommodate accurate modeling of biological data. These enhancements are the main focus of this chapter.

In modeling biological data we notice that there are at least three frequently occurring concepts: sequence ordering, input/output processes, and molecular spatial structure. Sequence data, such as nucleotides in DNA/RNA and amino acids in proteins, has this order property in their physical constructs. Important biological processes such as gene expression, metabolism, cell signaling, and biochemical pathway regulation all involve ordered events and input/output processes. The biological functionality of these entities is totally determined by their internal spatial molecular structures and various external interactions.

Because of the importance of these types of relationships, there is a need to model them. Database conceptual models, such as the widely used ER and EER models, do not easily represent these commonly occurring concepts from bioinformatics. This is because many traditional database applications do not require these concepts. Although ordering can be incorporated into relationships by adding one or more relationship attributes, this would complicate the conceptual schema and would make it difficult to identify the ordered relationships by looking at the conceptual schema diagram. It is preferable to have explicit and clear representations of such important and frequently occurring concepts.

In order to accommodate these features, we suggest significant yet minimal changes to the EER model by introducing three special types of relationships: the *ordered relationship*, the *process relationship*, and the *molecular spatial relationship*. In addition, many relationships in bioinformatics require duplication of instances in ordered relationships, so we also propose extensions to allow *multisets*, or *bags*, of relationship instances, where needed. Although the notational changes are minimal, they enhance the modeling power to directly capture these concepts. We also formally specify these constructs. The proposed extensions to the EER model that incorporate the above-mentioned omnipresent concepts can be used to direct subsequent implementation activities. These extensions should facilitate future work on the development of ontology and mediator systems [5, 6] as well as data mining and processing tools.

There is already some related work in the literature. For example, Keet [7] discusses the characteristics of biological data and its effect on Entity Relationship (ER), Object Oriented (OO), and Object Role Modeling (ORM) methodologies. Chen and Carlis [8] present a genomic schema element data model to capture this basic biological sequence notion, but there is no special notation for sequence order. Ram and Wei. [9] also propose a semantic model for 3D protein structure and DNA sequences, but their enhancements require many additional constructs and notations. The ONION framework [10] incorporates sequences in Resource Description Framework (RDF) methodology. Our extensions differ from these previous works because we achieve the power to represent all the discussed concepts with minimal changes to the EER model and its diagrammatic representation, thus adding considerable representation power with a minimum of additional concepts.

Another important characteristic of biological data modeling is that there are many different abstraction levels. Most biological data sources focus on gene,

protein, and pathways data. But biological modeling requires many levels of abstraction, from the DNA/RNA level to higher abstraction levels such as cells, tissues, organs, and biological systems. In addition, cross-referencing between the biological data, the references that discuss how this data was discovered and interpreted, and the experiments that led to these discoveries, are very important for biological researchers. Hence, it is important to represent and integrate and cross-reference data at different levels of abstraction. We introduce some preliminary ideas of multilevel modeling at the end of the chapter.

This chapter is organized as follows. In Section 3.2, we provide several examples of sequence ordering, multisets, input/output processes, and molecular spatial structure, and we establish the data modeling needs for these new concepts. In Section 3.3, we give formal definitions for ordered, process, and molecular spatial relationships to enhance the modeling features of the ER/EER models. Section 3.4 summarizes the new EER notations for these relationships. Section 3.5 gives the details of semantic models for the DNA/gene sequence, the protein 3D structure, and the molecular pathway. Section 3.6 describes mapping techniques that can be used for the implementations of our new EER constructs using relational databases.

Sections 3.7 and 3.8 introduce the concepts of multilevel modeling and discuss how they can be used in biological data applications. We conclude the chapter with some observations about our methods and directions for future work, in Section 3.9.

3.2 Biological Concepts and EER Modeling

This section focuses on the biomolecular subset of biological data. Closely related to these are the familiar concepts of sequence ordering, multisets, input/output processes, and molecular spatial structure. We illustrate these concepts using biological examples and present the associated ER conceptual modeling notation.

3.2.1 Sequence Ordering Concept

Molecular structural data includes linear nucleotide sequences of DNA (genes, intergenic and regulatory regions), as well as the linear amino acid sequences (proteins) resulting from gene expression. They are internal properties of biological entities (in contrast to external properties such as environment), and although both genetic and protein sequences can change slightly (the basis of evolution), for modeling purposes it is reasonable to treat them as static.

Example 1 Figure 3.1 shows the biological data of DNA sequence, genes, and their EER conceptual modeling. A DNA is an ordered sequence of bases A, T, C, and G. A gene is one segment of a DNA sequence. Different genes may or may not concatenate to each other. Some genes can be repeated in a DNA sequence; hence, both order and repetition are needed in the model. We model DNA-Base and DNA-Gene as an ordered bag (multiset) relationship.

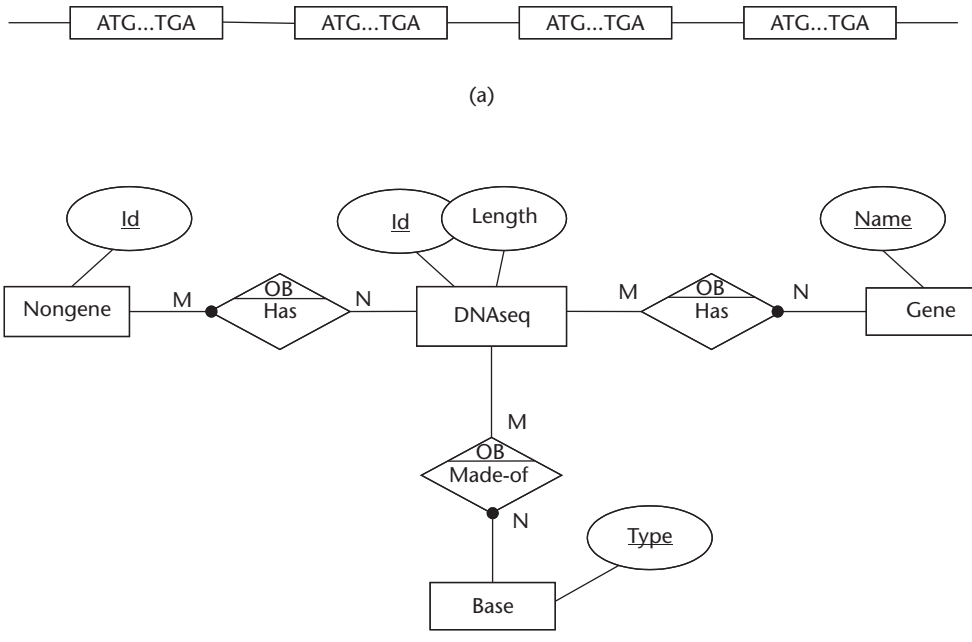


Figure 3.1 (a) Genes and nongenes on a DNA sequence. (b) EER model of DNA, gene, and base.

First, the ordering of elements in these sequences is their most important feature because changes to that order are likely to impact higher levels of structure and therefore also function. In Example 1, protein-coding genes are segments in a DNA sequence, as shown in Figure 3.1(a). Boxes and lines denote genes and intergenic regions, respectively. Each triplet of the bases A, T, C, and G in these genes determines one amino acid in the proteins they encode; a single change to one base can dramatically impact protein function (the classic example of this is sickle cell anemia). Obviously, in order to capture the ordering relation between DNA and genes, we need a special relationship for ordering features (symbol *O* denotes ordering) in EER models. In addition to the ordering of base pairs, ordering of sequence subsets (genes and intergenic regions) relative to one another is also important to model. Figure 3.1(b) is the EER schema for DNA, gene, and intergenic entities in our extended notation. We represent their relationships as binary relationship type.

A second important characteristic of modeling molecular data is that sequences may be a bag (or multiset) rather than a set of relationship instances, since the same gene (or intergenic sequence) may appear multiple times within the same DNA sequence, such as in gene homologs or tandem repeated DNA sequence blocks. We use the letter *B* in *OB* to denote that the relationship is an ordered bag that allows repetition.[1] Third, we have to specify the direction of ordering, which applies to all entities of one type related to a single entity (out of many) of another type. The solid

1. In the traditional ER model, repetition is not allowed for relationship instances.

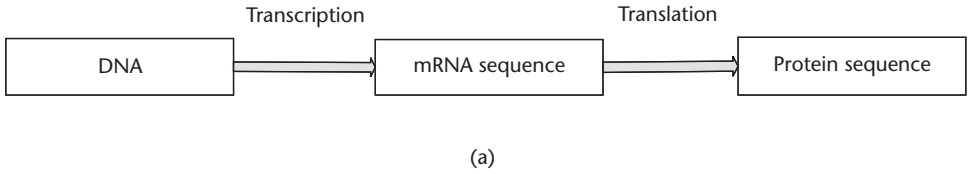
dot at one end of the relationship in Figure 3.1(b) indicates that related entities on this side are ordered.[2]

3.2.2 Input/Output Concept

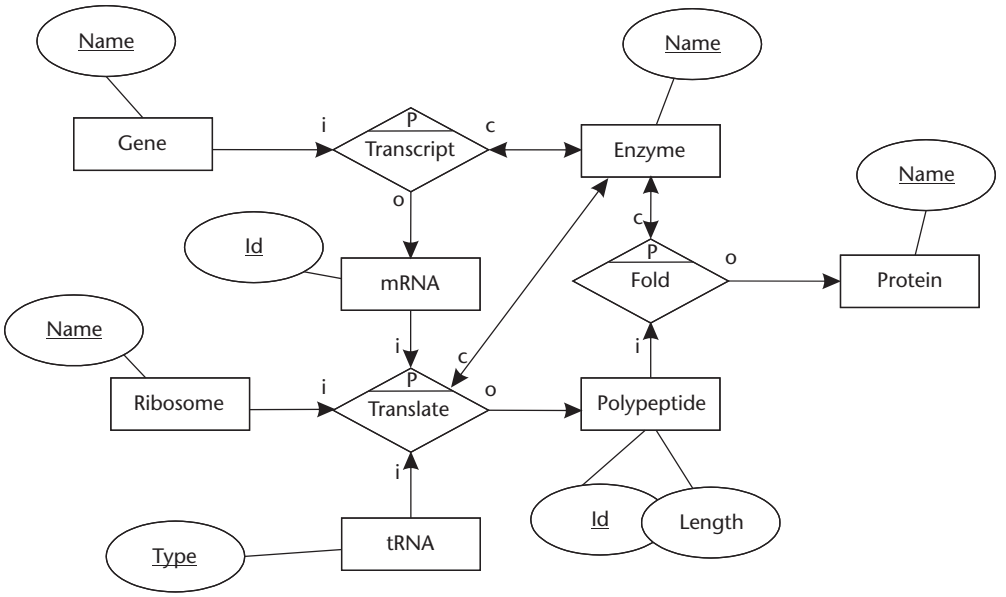
Figure 3.2 shows the biological concept of a gene expression process and its EER conceptual modeling. A process such as transcription or translation in Figure 3.2(a) relates the data of genes, mRNA sequence, or protein sequence in a directed way. The entities in a process can have three main participating roles: input, output, or catalyst [i, o, and c, respectively, in Figure 3.2(b)]. We model transcription or translation as a process relationship.

Molecular interaction is the key to the dynamics of biological processes such as gene expression, protein folding, metabolic pathways, and cell signaling.

Example 2 A protein is created from its gene through a series of interactions known as transcription and translation, as shown in Figure 3.2(a). Some entities act



(a)



(b)

Figure 3.2 (a) Gene expression process. (b) EER model of gene expression.

2. We note that other data models also have ordered and multiset relationships, such as the list and bag constructors in the ODMG object model.

as inputs, some as outputs (products), and others (typically enzymes) as catalysts to steer the process in a certain direction. These three roles in the system of a biochemical interaction are fundamental to molecular biology and important to any modeling scheme. It is also important to reflect hierarchy and subsets in such reactions since a complex process is made up of a sequence of unit processes resembling the workflows of assembly lines.

Pathway data has these kinds of attributes. A pathway is a linked set of biochemical reactions. The product of one reaction is a reactant of, or an enzyme that catalyzes, a subsequent reaction [11]. Figure 3.2(b) is the EER schema for the gene expression process. In this modified EER model we can represent the dynamic behavior of different agents. For example, mRNA is the output of the transcription process and an input of the translation process as well. Our extension of the EER model enables us to incorporate this important input/output process concept. In the notation shown in Figure 3.2(b), the P in the relationship indicates a process relationship; the edges marked i represent input entities to the process relationship. Edges marked o and c represent output entities and catalyst entities, respectively. The arrow directions also indicate the type of the role of each entity in the process relationship.

3.2.3 Molecular Spatial Relationship Concept

Example 3 Figure 3.3 shows the 3D chemical structure of amino acid alanine and its EER conceptual modeling. Each amino acid (residue) is composed of various types of atoms. Some atoms form chemical bonds in 3D space between each other. We model atoms and residues as molecular spatial relationships.

The function of a molecule is partly determined by its three-dimensional spatial structure; for example, the structure of DNA affects which regions can be read to make proteins, and the function of enzymes is often altered by minor influences on their structure due to changes in temperature, or salt concentration. These spatial structures are experimentally determined by X-ray crystallography or NMR [12], which generates topographical measurement data such as bond angles or distances as well as image data. As previously mentioned, a protein is a polypeptide chain built from 20 possible amino acids, or residues. Each residue is itself structurally composed of various types of atoms such as C, H, O, and N, shown in Figure 3.3(a). Each atom can be treated as a point and its position is thus represented by x, y, z coordinates in space. How those atoms are positioned can affect their fundamental chemical interactions, because of charge repulsion and proximity needed for break-

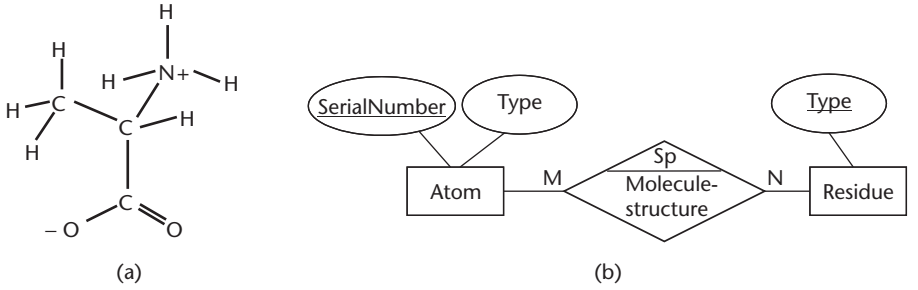


Figure 3.3 (a) 3D chemical structure of alanine. (b) EER model for atoms and residues.

ing and forming new chemical bonds between the atoms. This type of information is particularly important to biochemistry research, and is applied to pharmaceutical design since drug interactions are often adjusted at this level of structure and shape modifying function. Figure 3.3(b) is the EER schema for the 3D structure of residues. We use the letters Sp to represent the molecular spatial relationship between residues and atoms.

3.3 Formal Definitions for EER Extensions

We now give a formal definition for our extensions to the relationship concept in the ER/EER models. The main concepts in these models are entities and entity types, relationships and relationship types, attributes, and class/subclass inheritance [13, Chapters 3 and 4]. Entities represent objects, and relationships represent interactions or associations among objects. Attributes describe properties of entities or relationships.

A relation type R among n entity types $E_1, E_2, ..., E_n$ defines a set of associations among the participating entities. Mathematically, R is a set of relationship instances r_i, where each r_i associates n entities $(e_1, e_2, ..., e_n)$ and each entity e_j in r_i is a member of entity type E_j, $1 \leq j \leq n$. A relationship type can be defined as a subset of the Cartesian product $E_1 \times E_2 \times ... \times E_n$. A basic binary relationship in the EER model is a set of the relationship instances, where each pair (e_i, e_j) has the properties that $e_i \in E_1$, $e_j \in E_2$, and $R \subset E_1 \times E_2$. The next three sections describe formally the proposed new relationships for enhancing the existing EER model to represent biological data.

3.3.1 Ordered Relationships

To model ordering, we must extend the relationship concept in two directions: (1) allow related entities to be ordered; and (2) allow repetitions of the relationship instances. This means that the relationship set must be extended to allow duplicates. Before giving formal definitions of the extensions to the relationship concept, we show how we propose to extend the revised diagrammatic notation in order to minimize the changes to the ER/EER. We propose four types of relationships:

1. The original ER model relationship, which is an unordered set of relationship instances;
2. An ordered set relationship, where each relationship instance is unique (no duplicate instances are allowed);
3. An unordered bag relationship, which allows duplicate relationship instances;
4. An ordered bag relationship, which allows duplicates with ordering and can be used to model the situations discussed earlier.

The notation for these four relationships is shown in Figure 3.4. The letters O, B, and OB stand for ordered, bag (or multiset), and ordered bag, respectively. An edge with the filled circle (or solid dot) indicates that the attached entity type is the

Figure 3.4 EER notations for unordered set, ordered set, unordered bag, and ordered bag relationships (from left to right).

one whose elements are ordered by the relationship instances that are related to a specific entity type from the other entity type.

We now formalize these four types of relationships. We first define the concepts of unordered set, ordered set, unordered bag, and ordered bag, and then give the formal definitions of the relationships.

Let E be a set.

Let $\underbrace{E \times E \times \ldots E}_{n}$ be the set of all ordered n-tuples $(e_1, e_2, ..., e_n)$ where $e_1, e_2, ..., e_n \in E$.

Let $\underbrace{E \otimes E \otimes \ldots \otimes E}_{n}$ by $\otimes E^n$ be the set of all unordered n-tuples $[e_1, e_2, ..., e_n]$, where $e_1, e_2, ..., e_n \in E$. For convenience, we denote by E^n and $\underbrace{E \otimes E \otimes \ldots \otimes E}_{n}$ by $\otimes E^n$.

Definition 1. Unordered Set.
We say that F is an unordered set on a set E if[3]

$$F \subseteq \bigcup_{n \in N} F_n$$

where

$$F_n = \left\{ [e_1, e_2, \ldots, e_n] \in \otimes E^n \middle| e_i \neq e_j, \forall i \neq j \right\}$$

Definition 2. Ordered Set.
We say that F is an ordered set on a set E if

$$F \subseteq \bigcup_{n \in N} F_n$$

where

$$F_n = \left\{ (e_1, e_2, \ldots, e_n) \in E^n \middle| e_i \neq e_j, \forall i \neq j \right\}$$

Definition 3. Unordered Bag.
We say that F is an unordered bag on a set E if

$$F \subseteq \bigcup_{n \in N} \otimes E^n$$

Definition 4. Ordered Bag.
We say that F is an ordered bag on a set E if

3. N is the set of natural numbers.

$$F \subseteq \bigcup_{n \in N} E^n$$

Definition 5. Unordered Set Relationship.
We say that R is an unordered set relationship between E_1 and E_2 if $R \subseteq E_1 \times E_2$.

Definition 6. Ordered Set Relationship.
We say that R is an ordered set relationship between E_1 and E_2 if $R \subseteq E_1 \times E_2$, and for each $e \in E_1$, $\{e_j \mid (e, e_j) \in R$, is an ordered set on E_2.

Definition 7. Unordered Bag Relationship.
We say that R is an unordered bag relationship between E_1 and E_2 if R is a multiset of elements, and for each $e \in E_1$, $\{e_j \mid (e, e_j) \in R$, is an unordered bag on E_2.

Definition 8. Ordered Bag Relationship.
We say that R is an ordered bag relationship between E_1 and E_2 if R is a multiset of (e_i, e_j) elements, and for each $e \in E_1$, $\{e_j \mid (e, e_j) \in R$, is an ordered bag on E_2.

When the ordered relationship of Definitions 6 and 8 are represented diagrammatically, the dot is on the side of E_2. For example, suppose that a DNA sequence entity with identifier X is as follows:

$$\ldots \underbrace{A\ldots A}_{gene} \underbrace{C\ldots C}_{nongene} \underbrace{A\ldots A}_{gene} \underbrace{G\ldots G}_{nongene} \underbrace{T\ldots T}_{gene} \ldots$$

Suppose that $A\ldots A$, $A\ldots A$, $T\ldots T$ are genes in the sequence, whereas $C\ldots C$, and $G\ldots G$ are non-gene sequences. Then, the relationship instances including genes in sequence X will be the following ordered list:

$$(\ldots, (X, A\ldots A), (X, A\ldots A), (X, T\ldots T), \ldots)$$

3.3.2 Process Relationships

There are three basic roles in a process relationship:

- *Input(s):* entities consumed by the process, for example, by being transformed to some other entities;
- *Output(s):* entities produced by the process;
- *Catalyst(s):* entities that are needed for the process to work.

In Figure 3.5(a), E_1 represents the input entity, E_2 represents the output entity, and E_3 represents the catalyst entity. Symbol i stands for input, o stands for output, and c stands for catalyst. We use e_1 to represent entities in E_1, e_2 to represent entities in E_2, and e_3 to represent entities in E_3.

Definition 9. Process Relationship (Basic Type).
A basic process relationship is defined as a set of relationship instances (e_1, e_2, e_3), where $e_1 \in E_1$ represents the input entity, $e_2 \in E_2$ represents the output entity, and $e_2 \in E_3$ represents the catalyst entity. The relationship instance can also be represented

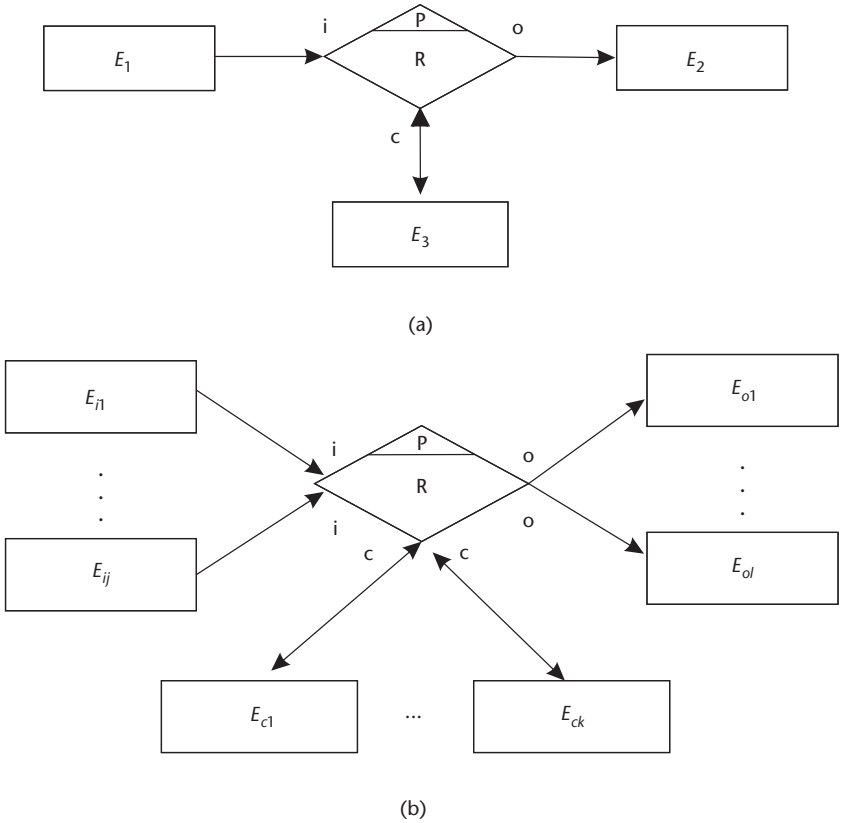


Figure 3.5 EER notations for process relationships: (a) basic type; and (b) general type.

as $\{e_1 \xrightarrow{\{e_3\}} e_2\}$, where the catalyst is optional and the input, output, and catalyst entity types do not have to be distinct.

Definition 10. Process Relationship (General Type).
In general, a process can have multiple inputs, outputs, and catalysts. In Figure 3.5(b), E_{i1} ... $E_{i,j}$ represent the input entities. E_{o1} ... E_{ol} represent the output entities. E_{c1} ... E_{ck} represent the catalyst entities. The process relationship is a set of relationship instances:

$$\left(e_{i1},\ldots,e_{ij},e_{o1},\ldots,e_{ol},e_{c1},\ldots e_{ck}\right)$$

where

$$e_{im} \in E_{im}\left(1 \le m \le j\right), e_{om} \in E_{om}\left(1 \le m \le l\right), e_{cm} \in E_{cm}\left(1 \le m \le k\right)$$

3.3.3 Molecular Spatial Relationships

Definition 11. Molecular Spatial Relationship.
A molecular spatial relationship is defined to describe the spatial relationships among a set of atoms in 3D space. Let A be a set of atoms, and let M be a set of mole-

cules, as shown in Figure 3.6. The molecular spatial relationship instance is a 3-tuple:

$$\left\langle \left(\langle a_1, x_1, y_1, z_1 \rangle, \langle a_2, x_2, y_2, z_2 \rangle, \ldots, \langle a_{ni}, x_{ni}, y_{ni}, z_{ni} \rangle \right), formula, m_i \right\rangle$$

where (a_1, a_2, ..., a_{ni}) is a group of associated atoms forming a molecule, *formula* denotes the chemical composition of the molecule, and $m_i \in M$. The x_{ni}, y_{ni}, z_{ni} associated with each atom a_{ni} describe the 3D atom location in the spatial structure. The molecular spatial relationship bares some characteristics of aggregation (the reverse is part-of) relationship in which atom entities are part-of a molecule entity. But it has its own property, which is that these atom entities are connected by some forces (bonding) that need explicit modeling.

3.4 Summary of New EER Notation

Table 3.1 summarizes the notation of the proposed new relationships. Notice that we have added considerable modeling and representation power to the basic relationship concept in ER models. However, the notation to display all these new complex concepts is not overly complex and hence should be easy to utilize.

3.5 Semantic Data Models of the Molecular Biological System

In this section, we provide the details of the EER conceptual schema of our molecular biological system that utilize the new EER constructs for the new types of relationships we defined in Section 3.3. The conceptual schema is roughly divided into three parts: the DNA/gene sequence, the protein structure, and the pathway.

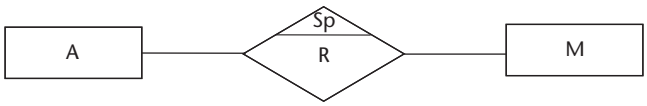


Figure 3.6 EER notation for molecular spatial relationship.

Table 3.1 New EER Relationships and Their Usage

EER Relationship	*Comments*
Unordered set	General relationship, unique instance.
Ordered set	Associates entities with ordering features. The relationship instances are unique.
Unordered bag	Associates entities without ordering features. The relationship instances can be duplicated.
Ordered bag	Associates entities with ordering features. The relationship instances can be duplicated.
Process	Associates different entities by the roles they have in a process. The roles are input, output, and catalyst.
Molecular spatial	Associates atom entities with molecule entities in 3D space.

3.5.1 The DNA/Gene Model

As we know, a DNA sequence is made up of four nucleotide bases in a specific order. A gene is one segment of a DNA sequence with a specific function. Usually, DNA sequences come from different sources of organisms, which have well-established phylogenetic classification schema, such as common name, genus, and species. Figure 3.7 shows the details of an EER conceptual schema for DNA/gene sequence that utilizes the order and bag (multiset) relationship. Note that we use the binary relationship type to represent the order relationship between the DNA, gene, and so on. Practically, we can have several options to model this relationship. Figure 3.8(a) shows their relations using binary relationship type. Gene and Nongene each have a m:n binary relation with DNAseq. Figure 3.8(b) shows the EER modeling option that DNA sequence, gene, and nongene form a ternary relation whenever a relationship instance (DNA, gene, and nongene) exists. Figure 3.8(c) shows a new entity type Segment that is created to include all entities of both Gene and Nongene, and this Segment has an order relationship with DNAseq. In Figure 3.8(d) we have a general approach to represent the relationship between biological sequence entities. This model is easy to modify and extend depending on various situations. Because some instances in DNAseq are Gene type, some are Nongene type, and some are other type, we could create a union of these types and thus make DNAseq to be a subclass of it. A recursive ordered relationship Has exists between DNA sequences themselves. One (long) sequence participates in the supersequence role, and the other (shorter) sequence in the subsequence role. In Section 3.6 we will discuss the different models in the ER-to-relational mapping process.

3.5.2 The Protein 3D Structure Model

Usually a structure-determined protein contains one or more chains of residues. These originally spatial-free linear chains are constrained by various physical or

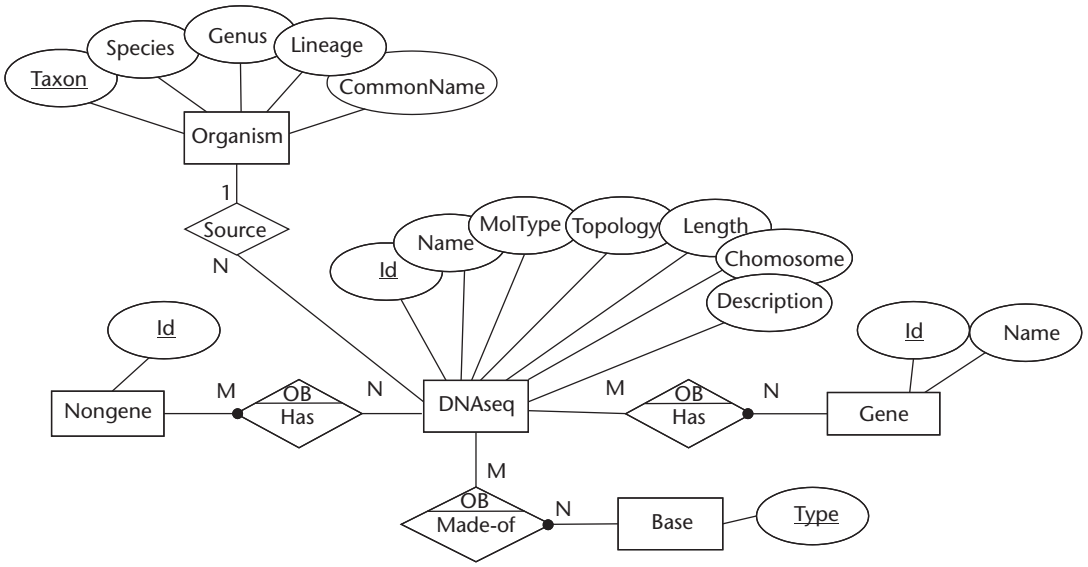
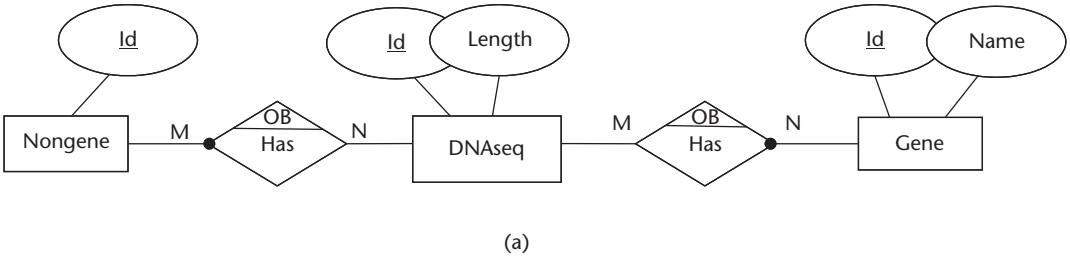
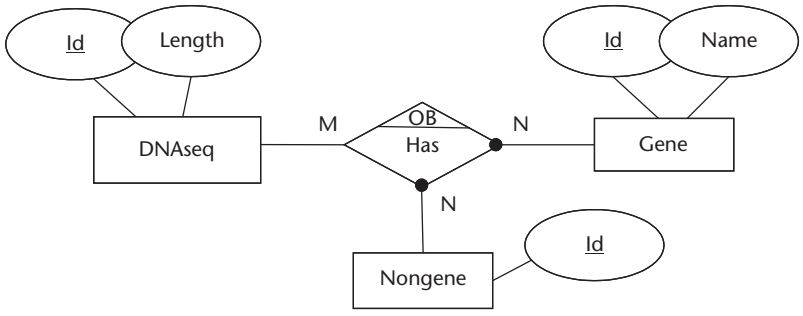


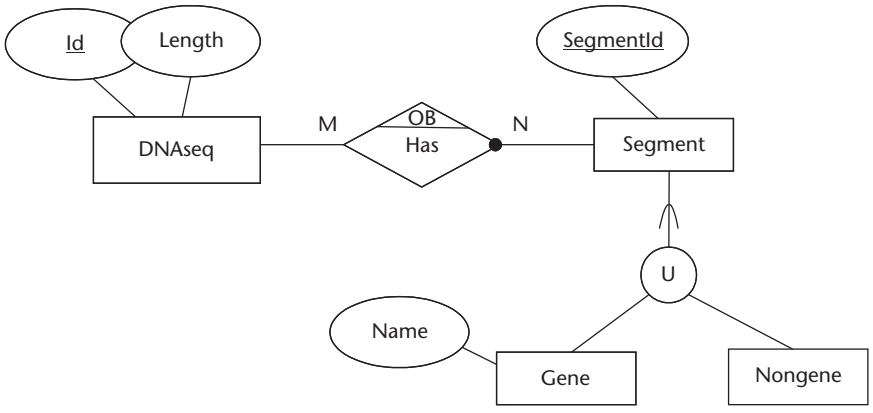
Figure 3.7 EER schema of the DNA/gene sequence.



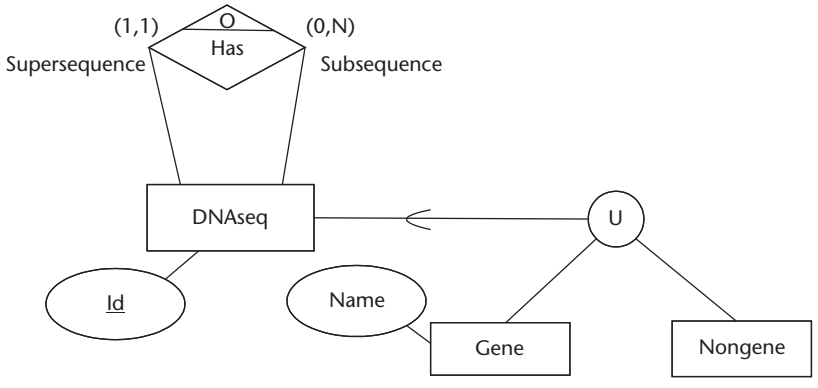
(a)



(b)



(c)



(d)

Figure 3.8 EER model options for DNA/gene sequence: (a) binary type; (b) ternary type; (c) union type; and (d) general type.

chemical forces to form higher levels of 3D structure. They are secondary, tertiary, and quaternary structure of the protein.

The primary structure is a linear polypeptide chain of residues with specific order. Secondary structure refers to the general three-dimensional form of local regions (segments of chains) or overall shape of polypeptide chain. Helix, sheet, and turn are characteristic structural components. An α-helix is a tight helix formed out of the polypeptide chain. The polypeptide main chain makes up the central structure, and the side chains extend out and away from the helix. The CO group of one amino acid (n) is hydrogen bonded to the NH group of the amino acid four residues away (n +4). In this way every CO and NH group of the backbone is hydrogen bonded. They are formed by hydrogen bonding. Multiple hydrogen bonds make a segment of amino acid chains fold in specific ways.

Tertiary structure is the full three-dimensional folded structure of the polypeptide chain. It assembles the different secondary structure elements in a particular arrangement. As helices and sheets are units of secondary structure, so the domain is the unit of tertiary structure. In multidomain proteins, tertiary structure includes the arrangement of domains relative to each other as well as that of the chain within each domain [14].

Quaternary structure is a protein complex assembled by multiple-subunit proteins. Examples of proteins with quaternary structure include hemoglobin, DNA polymerase, and ion channels. Quaternary structures are stabilized mainly by noncovalent interactions; all types of noncovalent interactions (hydrogen bonding, van der Walls interactions, and ionic bonding) are involved in the interactions between subunits. In rare instances, disulfide bonds between cysteine residues in different polypeptide chains are involved in stabilizing this level of structure.

Figure 3.9 shows the EER conceptual schema of protein 3D structure which utilizes the new types of relationships. We go though these entities and relationships from the bottom to the top level.

- *Atom:* This entity type represents the chemical atoms such as C, H, O, N, and S in the molecular structure. They can be identified uniquely by their atom serial number and spatial position. Cartesian coordinates (x, y, z) are one such coordinate model.
- *SSBond and HBond:* These are typical examples of molecular spatial relationship types denoting the chemical bonding formed among atoms. The spatial bond relationship can be identified uniquely by its bond type, bond length, and atoms that participate.
- *Residue:* This entity type represents the amino acids connecting to each other in the chains of protein primary sequence. Each residue is a molecule (exists dependently) composed of atoms via the Molecule-Structure spatial relationship.
- *Molecule-Structure:* This type of molecular spatial relationship is defined to describe the spatial relationship between a set of atoms within a molecule.
- *Made-of:* This is the ordered bag relationship type. It denotes that a sequence of residues (some residues can be duplicated) forms a specific chain of protein primary sequence. The solid dot at one end of the relationship indicates that related entities of Residue are ordered with respect to a single entity (out of

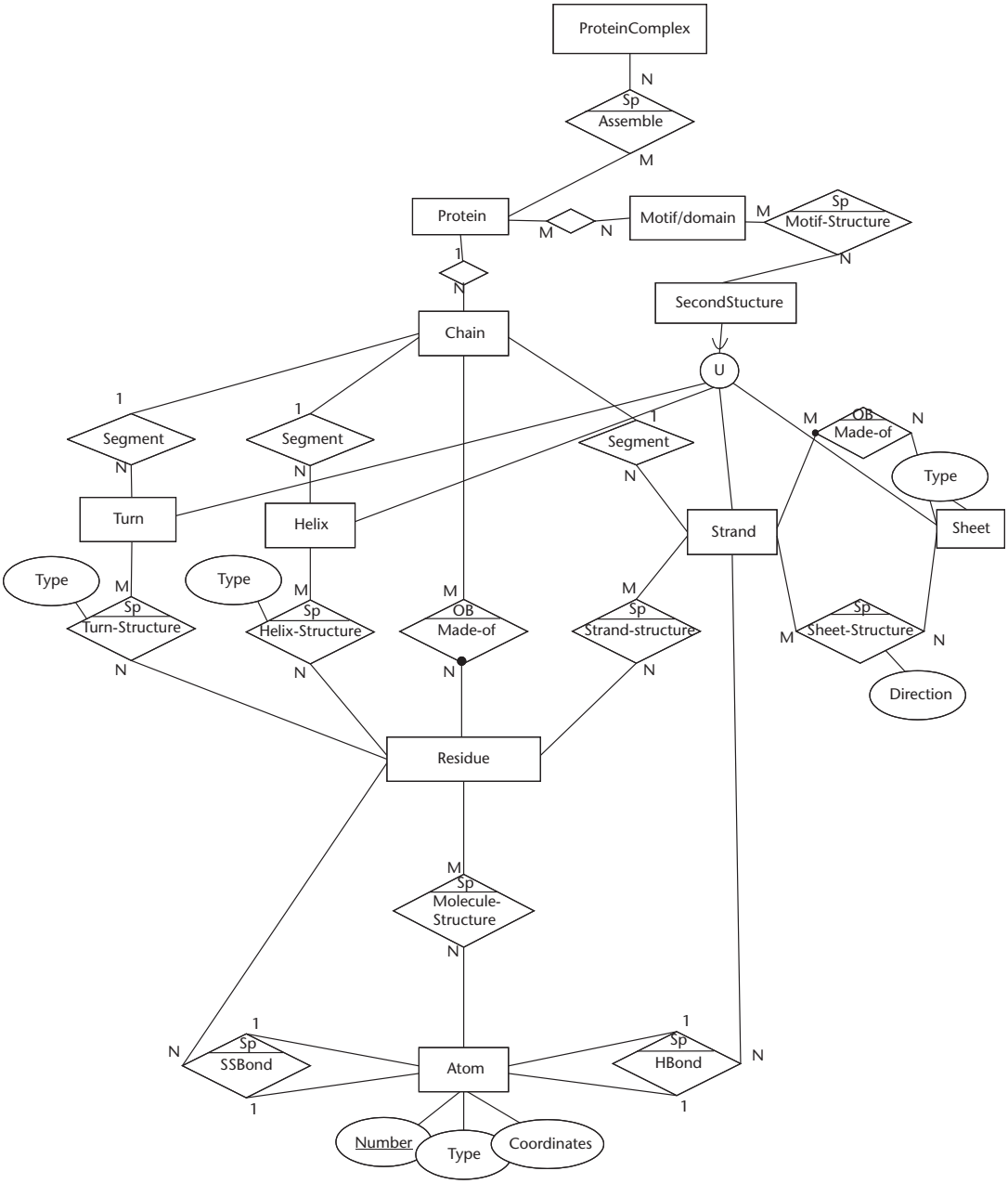


Figure 3.9 EER schema of the protein 3D structure.

many) of Chain. Thus, there exist inherent attributes of Made-of ordered relationships, such as the length of chain in terms of residue count and the order number of each residue in this chain.

- *Chain:* This entity type models the one-dimensional structure of protein sequence. It is a line of residues without constraint. A single chain can be partitioned into one or more segments. These segments make up the central structure of secondary components. They can form α-helix, α-pleated sheet, or turn.

- *Helix:* This entity type models one type of the secondary structural component, α-helix. It is formed by every five consecutive residues via the Helix-Structure spatial relationship. Its cardinality constrain is 1:5 between entity type Helix and Residue.
- *Sheet:* This entity type models another type of the secondary structural component, α-pleated sheet. It is a two-dimensional structure. It can be modeled as a sequence of one-dimensional structures of Strand via the Made-of ordered relationship. There are several types of sheets, such as circle and bi-fork. One instance of strands can be shared by several different sheets. So the cardinality constraint between Strand and Sheet is m:n. Strand is also one segment of a polypeptide chain, which is formed by consecutive residues via the Strand-Structure spatial relationship.
- *Turn:* This entity type models one of the secondary structural component, turn. There are three types of turns: 3-turn, 4-turn, and 5-turn [15].
- *Assemble:* This molecular spatial relationship denotes that two or more protein components can be assembled into a protein complex, thus forming dimers, trimers, tetramers, and so on. The Type attribute of the relationship denotes the type of assembly, whether it is composed of the same type of protein units (homo-multimer) or different types (hetero-multimer).
- *Motif/Domain:* Usually, a motif consists of a small number of secondary elements (helices, sheets, and turn), combined in local specific geometric arrangements. These motifs then coalesce to form domains. To simplify modeling, we do not distinguish between motifs and domains. Note that the Motif-Structure spatial relationship relates the entity Motif/Domain and SecondStructure. Many proteins can share the same type of domains, so the cardinality ratio between Protein and Motif/Domain is m:n.

3.5.3 The Molecular Interaction and Pathway Model

Figure 3.10 shows the EER conceptual schema of the molecular interaction and biological pathway. In our conceptual model, the entity Bioentity is the high level class of biological objects that are physical entities with attributes pertaining to their internal properties (e.g., the nucleotide base sequence of a gene, the molecular weight of a protein). So it is the union of all types of biological entities, such as genes, proteins, and cofactors (metal ions or small organic molecules). Another important entity is the Interaction that relates any pair of biological entities. These interactions include gene-gene and protein-protein interactions. Some complex formed by molecular interactions like DNA-protein binding can also be an instance of interactions. There exist three relationships between Bioentity and Interaction in our design. The Input and Output relationships are for any two pairs of interacting entities, and the Catalyst relationship is for other helping entities if they exist in the interaction. Here we name these three relationships using "input," "output," and "catalyst" for the purpose of process relationship representation (discussed in Section 3.6). Note that this design can also model the reaction concept with reactant (input), product (output), and catalyst roles. By definition, a pathway is a linked set of biochemical interactions (reactions). If we ignore the branch case of the pathway,

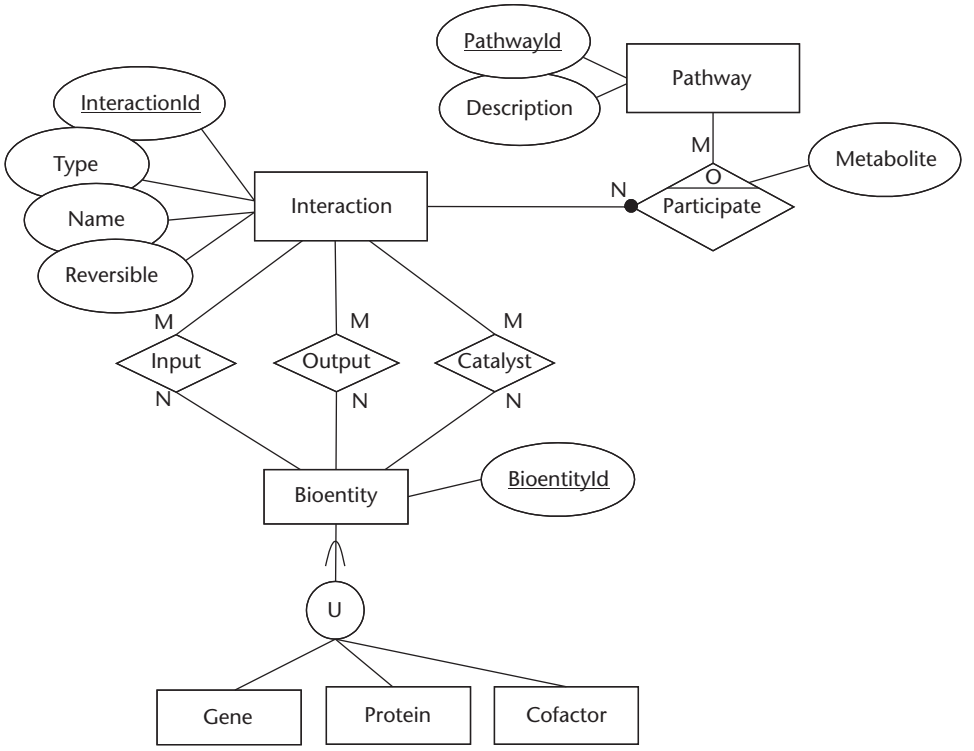


Figure 3.10 EER schema of the molecular interaction and the pathway.

it can be treated as a sequence of unique interactions. So we use the Participate ordered set relationship to denote the relation between Interaction and Pathway.

3.6 EER-to-Relational Mapping

In this section, we describe the implementation of the above-described new EER constructs to the relational database. We show how to map the ordered relationship, the process relationship, and the molecular spatial relationship to relational models. This allows us to implement a conceptual design on a relational database system, such as ORACLE or MySQL.

3.6.1 Ordered Relationship Mapping

In Section 3.3.1 we defined four types of the ordered relationships, shown in Figure 3.4. The mapping of the unordered set relationship is a standard procedure [13, Chapter 7]. For the ordered set relationship mapping, we create a new relation R, including the primary keys of E1 and E2 as foreign keys in R and rename them as *E1Id* and *E2Id*, respectively. The primary key of this relation is the combination of the attributes *E1Id* and *E2Id*. We also include additional *OrderNo* attribute to indicate the ordering of *E2Ids* related to the same *E1Id* value. The following constraint will hold on the *OrderNo* attribute: for all tuples with the same value for *E1Id*, the values of *OrderNo* will be distinct and numbered 1, 2, 3… (see Table 3.2).

Table 3.2 EER-to-Relational Mapping of Ordered Set Relationship (R)

E1Id	*E2Id*	*OrderNo*
v1	v2	1
v1	v3	4
v1	v5	2
v1	v7	3
…		

For the unordered bag relationship mapping, the relation R includes the primary key of E1 as *E1Id*, the primary key of E2 as *E2Id*, and attribute *BagDiscriminator*. The *BagDiscriminator* is to discriminate the tuples if the values of (*E1Id*, *E2Id*) are the same in the bag relationship, because the elements in the bag can be duplicate. The primary key of this relation is the combination of the foreign key attributes *E1Id*, *E2Id*, and *BagDiscriminator*. The following constraint will hold on the *BagDiscriminator* attribute: for all tuples with the same (*E1Id*, *E2Id*) combination of values, the values of *BagDiscriminator* will be distinct (they can be ordered 1, 2, 3). Table 3.3 shows one example of the mapping in relation table.

For the ordered bag relationship mapping, the relation R includes the primary key of E1, the primary key of E2, and attribute *OrderNo*. Like the attributes of the above relations, the *OrderNo* is to both discriminate and order the tuples with the same *E1Id* value. The same constraint on *OrderNo* for ordered set applies here. The primary key of this relation is (*E1Id*, *E2Id*, *OrderNo*). Table 3.4 shows one example of the mapping in relation table.

3.6.2 Process Relationship Mapping

As defined in Section 3.3.2 (Figure 3.5), the entities associated with the process relationship have three distinct types: i (input), o (output), and c (catalyst). For each process relationship, we can have a new relation R with three attributes (i, o, c) whose values are the primary keys of each participating entities, as shown in Table 3.5. Each such table holds the relationship instances for one of the process relationships. Another relation, called ProcessRelationDesc, is needed to describe the participating entities for all process relationships. Its attributes include *Relation*, *Entity*, and *Role*.

Table 3.3 EER-to-Relational Mapping of Unordered Bag Relationship (R)

E1Id	*E2Id*	*BagDiscriminator*
v1	v2	1
v2	v5	1
v1	v2	2
v1	v3	1
v1	v2	3
…		

Table 3.4 EER-to-Relational Mapping of Ordered Bag Relationship (R)

E1Id	*E2Id*	*OrderNo*
v1	v2	1
v1	v2	8
v1	v2	2
v1	v3	7
v1	v3	4
v1	v4	3
v1	v4	5
v1	v4	6
…		

Table 3.5 EER-to-Relational Mapping of Process Relationship (R1 and R2 of Basic Type)

Input	*Output*	*Catalyst*
v1	v2	v3
v4	v8	v6
…		
…		
Input	*Output*	*Catalyst*
v7	v2	v13
v6	v9	v5

Relation records the names of the process relationship. Entity records the names of the entity type that participate in a process relationship, while Role specifies their acting roles. Table 3.6 shows an example of the mapping results in relation table.

The above mapping works for process relationships that have one input, one output, and one catalyst only. If we want to map the general case, where there can be multiple inputs, outputs, or catalysts, we can name the input attributes *i1*, *i2*, …, the outputs *o1*, *o2*, …, and the catalysts *c1*, *c2*, …. An example is given in Table 3.7.

3.6.3 Molecular Spatial Relationship Mapping

As defined in Section 3.3.3 (EER notation shown in Figure 3.6), the molecular spatial relationship R associates a group of atoms (component objects) spatially with a molecule entity (a composite object) with specific connectivity among atoms. For the mapping, we can have a new relation called MolStructure with attributes (*MoleculeId*, *Atom*, *Discriminator*, *X*, *Y*, *Z*, *AtomOId*). *MoleculeId* refers to the primary key of the Molecule relation in Table 3.8. As described in the ordered relationship mapping, the attribute *Discriminator* distinguishes the atoms of the same type in a molecule. *X, Y, Z* attributes are the Cartesian coordinates of atoms. *AtomOId* is a system-generated unique object id for each instance in this relation. The primary key of this relation is *AtomOId*. The alternative keys can be

Table 3.6 EER-to-Relational Mapping of Process Relationship (ProcessRelationDesc)

Relation	*Entity*	*Role*
R1	E1	i
R1	E2	o
R2	E3	c
R2	E4	I
R2	E5	o
R2	E6	c
R3	E7	i1
R3	E8	i2
R3	E9	o1
R3	E10	o2
R3	E11	o3
R3	E12	c1
R3	E13	c2
…		

Table 3.7 EER-to-Relational Mapping of Process Relationship (R3 of General Type)

i1	*i2*	*o1*	*o2*	*o3*	*c1*	*c2*
v1	v1	v3	v4	v2	v2	v7
v2	v8	v6	v6	v8	v9	v12
…						

Table 3.8 EER-to-Relational Mapping of Molecular Spatial Relationship (Molecule)

MoleculeId	*Name*	*Formula*	*Isomer*
1	Water	H2O	0
2	Alanine	C3H7O2N	21
…			

(*MoleculeId*, *X, Y, Z*) or (*MoleculeId*, *Atom*, *Discriminator*). Table 3.9 shows the mapping example of water and alanine molecules. For the simplicity, H atoms are deliberately omitted.

To record the bond information, we should have associated connection relation called Bond with attributes (*AtomOId1*, *AtomOId2*). *AtomOId1* and *AtomOId2* refer to the primary key *AtomOId* of relation MolStructure in Table 3.9. We can enforce a constraint that the value of *AtomOId1* is always less than the value of *AtomOId2* because the connectivity between atoms (nodes) is undirectional (see Table 3.10).

Table 3.9 EER-to-Relational Mapping of Molecular Spatial Relationship (MolStructure)

MoleculeId	*Atom*	*Discriminator*	*X*	*Y*	*Z*	*AtomOId*
1	H	1	–1.4	1.3	0.0	1
1	H	2	1.4	1.3	0.0	2
1	O	1	0.0	0.0	0.0	3
2	N	1	1.6	1.5	0.0	4
2	C	1	0.0	0.6	0.0	5
2	C	2	–1.3	1.4	0.0	6
2	C	3	0.0	–0.6	0.0	7
2	O	1	1.7	–1.6	0.0	8
2	O	2	–1.7	–1.6	0.0	9
…						

Table 3.10 EER-to-Relational Mapping of Molecular Spatial Relationship (Bond)

AtomOId1	*AtomOId2*
1	2
2	3
4	5
5	6
5	7
7	8
7	9
…	

3.7 Introduction to Multilevel Modeling and Data Source Integration

Biological and medical researchers create large amounts of data, stored in diverse databases such as GenBank and PDB, which needs to be processed, integrated, and organized in order to query them efficiently. Some integration work has been done, as mentioned in [16, 17], and this work is mainly categorized into warehouse integration, mediator-based integration, and navigational integration. Most of this work [15] focuses on horizontal integration that integrates complementary sources. In the bioinformatics/biomedical research fields, however, some semantic data can be classified into different levels of abstraction and mapped to different schemas based on the degree of abstraction. The degree of abstraction of the data determines the amount of detail of information that is enclosed inside. The higher abstraction level data contains less detailed information than in lower abstraction levels [18]. Moreover, events occurring at one level can effect and be affected by events at different levels of scale or time. Between the levels, there may be some transitions and connections, which need what we call a vertical approach to integrate these data

sources to explore further information. Next, we will briefly describe some of the concepts of a human body and apply multilevel modeling on these concepts in order to illustrate what we mean by multilevel modeling.

Cell theory states that all living things are composed of cells, and cells are the basic units of structure and function in living things [19]. In a multicellular organism such as a human being, different types of cells perform different tasks. Some cells can be responsible for extracting nutrients, and others can function as receptors or be responsible for other functions. Tissues are groups of similar cells specialized for a single function, such as epithelial tissue which lines the chambers of the heart to prevent leakage of blood and nervous tissue which receives messages from the body's external and internal environment, analyses the data, and directs the response [19]. An organ is a group of tissues that work together to perform a complex function. For instance, the heart is the most important organ inside the human body, which is mainly composed of epithelial tissues, connective tissues, and nervous tissues. Biologists classify the human body into 11 organ systems (nervous system, circulatory system, skeletal system, muscular system, and so on), and each system is a group of organs that perform closely related functions. As we know, the circulatory system supports cells throughout the body with the nutrients and oxygen that they need to stay alive [19]. After we depict the human body, we define five different data abstraction levels for human biology: molecule, cell, tissue, organ, and system.

In the next section, we propose an extended EER model which incorporates multilevel concepts and relationships related to the biological field with the purpose of building a biological data model for further integration work. We give an example of why multilevel modeling may be useful in biological conceptual modeling.

3.8 Multilevel Concepts and EER Modeling

Example 1 Figure 3.11 shows the EER conceptual modeling of a cell system [20]. Usually a cell is surrounded by a plasma membrane (entity type Membrane). There are channels (entity type Pore) embedded within this membrane that allow different molecules to pass through the membrane. Cell surface membranes also contain receptor (entity type Receptor) proteins that allow cells to detect external signalling molecules such as hormones. Cells also have a set of "little organs," called organelles (entity type Compartment), specialized for carrying out one or more vital functions. In every second there are millions of biochemical reactions that happen in a cell, integrating into diverse types of biological processes (entity type Pathway). Genetic materials such as DNA are packaged into chromosomes (entity type Chromosome), which are stored in the nucleus (entity type Nucleus) of a cell. Cells grow through successive cell divisions. Relationship split denotes this cellular metabolism. The same type of cells can be assembled into a group of cells (entity type Tissue) executing special functions.

Due to the complexity and huge amount of data stored and processed in a single cell (not to mention the even higher information density in tissue, organ, and organism systems), it will be more efficient and easier to integrate if we can classify and relate data at different abstraction levels.

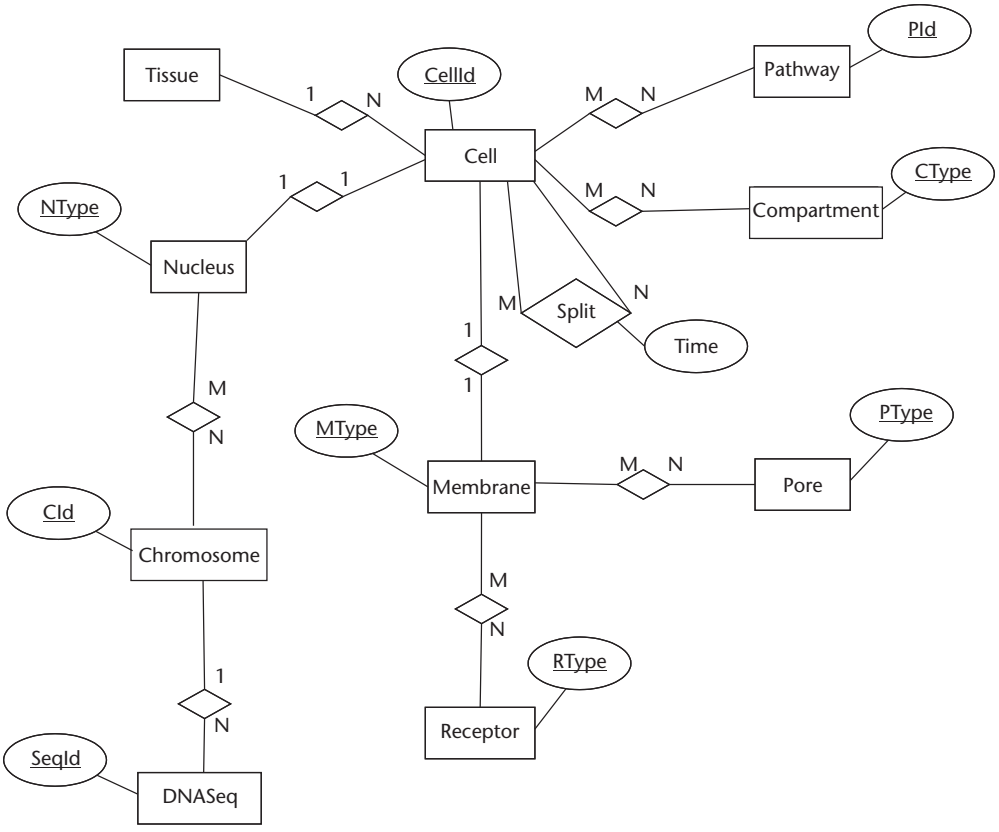


Figure 3.11 Conceptual EER diagram of a cell system.

In Figure 3.12, we categorize biological data based on their abstraction levels, concepts, and experimental result/evidences. For each concept or experimental result/evidence, it can be assigned to certain abstraction level. In addition, each con-

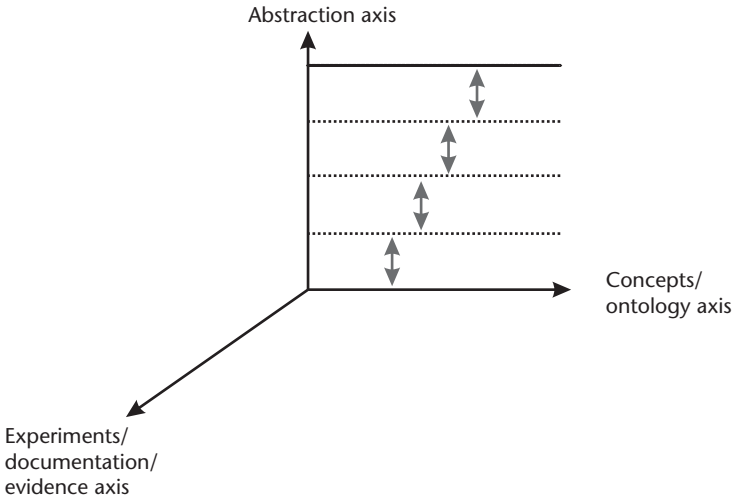


Figure 3.12 Abstraction, evidence, and concepts coordinates.

cept can have several experimental results or evidences related to it. For example, we may define five different levels (molecule, cell, tissue, organ, and system) for human anatomical data, and each data source can be classified into a certain level. Figure 3.13 demonstrates the data source integrations from the horizontal and vertical approaches. The horizontal approach integrates data sources at the same abstraction level. In contrast, the vertical approach integrates data sources from different abstraction levels. In order to integrate data in different abstraction levels, we need to establish connections/interactions between them. It will be necessary to propose some biological relationships that can describe the relationships and interactions between data at the different levels as building blocks of vertical data integration.

3.9 Conclusion

In this chapter we introduced three new types of relationships into the EER model: ordered relationship, process relationship, and molecular spatial relationship. We also extended the relationships to allow bags (or multisets) of relationship instances, since many relationships in molecular biology fall into this category. We illustrated the need for these relationships in modeling biological data and we proposed some special diagrammatic notation. By introducing these extensions, we anticipate that biological data having these properties will be made explicit to the data modeler, which would help direct future biological database implementation. In particular, our unordered bag relationship can be used for various reaction data, and our

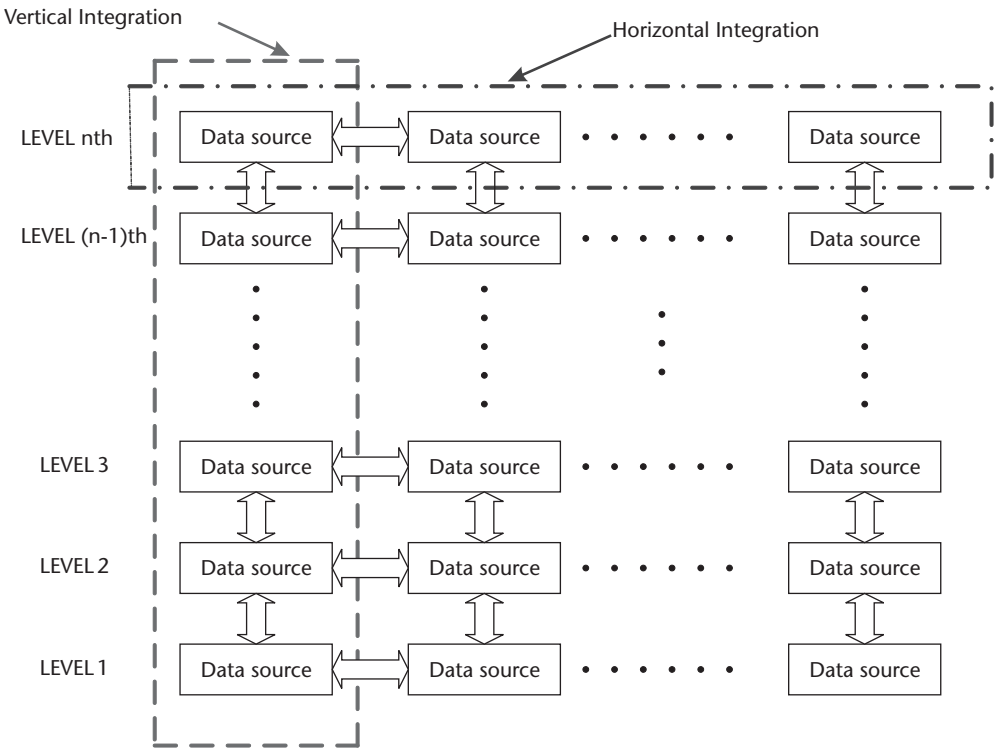


Figure 3.13 Horizontal and vertical integration.

ordered bag should be useful for sequence and genetic features. Our process relationship would be useful for reaction, interaction, and pathway data. These changes do not add much complexity to the existing EER model, thus making them easier for integration. We also gave the formal definitions for these new concepts and summarized their notation and usage. We also showed how these additional concepts could be mapped into relations for implementation in relational databases such as ORACLE or MySQL. We are currently implementing these relationships in our ontology-based mediator. We have already used these concepts to design initial conceptual schema for parts of GenBank, PDB, and Pathways data [21].

We then briefly introduced the concepts of multilevel modeling that can be utilized in integrating data sources from different abstraction levels.

References

[1] Benson, D. A., et al., "GenBank," *Nucleic Acids Res.*, Vol. 34 (Database Issue), 2006, pp. D16–D20.

[2] Bry, F., and P. Kroger, "A Computational Biology Database Digest: Data, Data Analysis, and Data Management," *Distributed and Parallel Databases*, Vol. 13, 2003, pp. 7–42.

[3] Ostell, J., S. J. Wheelan, and J. A. Kans, *The NCBI Data Model in Bioinformatics: A Practical Guide to the Analysis of Genes and Proteins*, New York: John Wiley & Sons, 2001.

[4] Riemer, C., et al., "A Database of Experimental Results on Globin Gene Expression," *Genomics*, Vol. 53, 1998, pp. 325–337.

[5] Han, H., and R. Elmasri, *Ontology Extraction and Conceptual Modeling for Web Information,* Hershey, PA: IDEA Group Publishing, 2003.

[6] Harris, M. A., et al., "The Gene Ontology Database and Informatics Resource," *Nucleic Acids Res.*, Vol. 32 (Database Issue), 2004, pp. D258–D261.

[7] Keet, C. M., "Biological Data and Conceptual Modelling Methods," *Journal of Conceptual Modeling*, Issue 29, October 2003.

[8] Chen, J. Y., and J. V. Carlis, "Genomic Data Modeling," *Information Systems*, Vol. 28, No. 4, 2003, pp. 287–310.

[9] Ram, S., and W. Wei, "Modeling the Semantic of 3D Protein Structures," *23rd International Conference on Conceptual Modeling (ER)*, LNCS 3288, 2004, pp. 696–708.

[10] Mitra, P., G. Wiederhold, and S. Decker, "A Scalable Framework for the Interoperation of Information Sources," *Proc. of SWWS'01, The First Semantic Web Working Symposium*, Stanford University, Stanford, CA, 2001, pp. 317–329.

[11] Karp, P. D., "Pathway Databases: A Case Study in Computational Symbolic Theories," *Science*, Vol. 293, 2001, pp. 2040–2044.

[12] Westbrook, J., et al., "The Protein Data Bank and Structural Genomics," *Nucleic Acids Res.*, Vol. 31, No. 1, 2003, pp. 489–491.

[13] Elmasri, R., and S. B. Navathe, *Fundamentals of Database Systems*, 5th ed., Reading, MA: Addison-Wesley, 2006.

[14] Guex, N., A. Diemand, and M. C. Peitsch, "Protein Modelling for All," *Trends in Biochemical Sciences*, Vol. 24, 1999, pp. 364–367.

[15] Kabsch, W., and C. Sander, "Dictionary of Protein Secondary Structure: Pattern Recognition of Hydrogen-Bonded and Geometrical Features," *Biopolymers*, Vol. 22, No. 12, 1983, pp. 2577–2673.

[16] Hernandez, T., and S. Kambhampati, "Integration of Biological Sources: Current Systems and Challenges Ahead," *SIGMOD Record*, Vol. 33, No. 3, 2004, pp. 51–60.

[17] Topaloglou, T., et al., "Biological Data Management: Research, Practice and Opportunities," *Proc. of the 30th VLDB Conference*, Toronto, Canada, 2004, pp. 1233–1236.

[18] Benjamin, P., et al., "Simulation Modeling at Multiple Levels of Abstraction," *WSC'98: Proc. of the 30th Conference on Winter Simulation*, Los Alamitos, CA, 1998, pp. 391–398.

[19] Miller, K. R., and J. Levine, *Biology*, Upper Saddle River, NJ: Prentice-Hall, 2002.

[20] http://en.wikipedia.org/wiki/Cell_(biology).

[21] Elmasri, R., et al., "Extending EER Modeling Concepts for Biological Data," *Proc. of 19th IEEE International Symposium on Computer-Based Medical Systems (CBMS 2006),* Salt Lake City, UT, June 2006.

CHAPTER 4

Fundamentals of Gene Ontology

Viroj Wiwanitkit

A main scientific interest in the post-genomic era is gene function. Gene ontology (GO) provides biological knowledge of a gene product in any organism. This chapter presents an overview of gene ontology. It discusses the design application in biomedical sciences and usage of gene ontology.

4.1 Introduction to Gene Ontology

A main scientific interest in the postgenomic era is gene function [1]. The exponential growth in the volume of accessible biological information has generated a confusion of voices surrounding the annotation of molecular information about genes and their products [2]. As a new focus, function and other information concerning genes are to be captured, made accessible to biologists, or structured in a computable form [1]. Since much of biology works by applying prior known knowledge to an unknown entity, the application of a set of axioms that will elicit knowledge and the complex biological data stored in bioinformatics databases is necessary. This often require the addition of knowledge to specify and constrain the values held in those databases, and a way of capturing knowledge within bioinformatics applications and databases is by using ontologies [3]. Until recently, the concept of ontology has been almost unknown in bioinformatics, and even more so in molecular biology [4]. Nowadays, many bioinformatics articles mention it in connection with text mining, data integration, or as a metaphysical cure for problems in standardization of nomenclature and other applications [4]. Efforts in genome annotation are most often based upon advances in computer systems that are specifically designed to deal with the tremendous amounts of data being generated by current sequencing projects [5]. These efforts in analysis are being linked to new ways of visualizing computationally annotated genomes [5]. It can be said that an ontology is the concrete form of a conceptualization of a public knowledge of a domain [3].

Since the formation of the Gene Ontology (GO) Consortium, its aim has been to provide a framework for both the description and the organization of such information in the genomics domain [1]. The Gene Ontology Project seeks to provide a set of structured vocabularies for specific biological domains that can be used to describe gene products in any organism [2]. The work includes building three extensive ontologies to describe molecular function, biological process, and cellu-

lar components, and providing a community database resource that supports the use of these ontologies [2]. The GO Consortium was initiated by scientists associated with three model organism databases—the Saccharomyces Genome database (SGD); FlyBase, the Drosophila genome database; and MGD/GXD, the Mouse Genome Informatics databases—then other databases were added [2]. The GO Consortium supports the development of the GO database resource and provides tools which enable curators and researchers to query and manipulate the vocabularies [2]. Two new databases providing new resources for gene annotation have been launched: the InterPro database of protein domains and motifs, and the GO database for terms that describe the molecular functions and biological roles of gene products [5]. Now, the GO data resources are accessible to the public at http://www.geneontology.org/, and the GO Web site can be used by the community both to recover the GO vocabularies and to access the annotated gene product datasets from the model organism databases [2]. The GO Consortium is presently concerned with three structured controlled vocabularies which describe the molecular function, biological roles, and cellular locations of gene products [1]. In 2005, Doms and Schroeder introduced GoPubMed (www.gopubmed.org), a Web server which allowed users to explore PubMed search results with the GO. GoPubMed provides the following benefits: (1) it gives an overview of the literature abstracts by categorizing abstracts according to the GO and thus allowing users to quickly navigate through the abstracts by category; (2) it automatically shows general ontology terms related to the original query, which often do not even appear directly in the abstract; (3) it enables users to verify its classification because GO terms are highlighted in the abstracts and each term is labeled with an accuracy percentage; and (4) exploring PubMed abstracts with GoPubMed is useful as it shows definitions of GO terms without the need for further look-up [6].

The aim of this chapter is to introduce the reader to the use of GO within bioinformatics. The concept of the construction process of an ontology as well as the application of GO in biological and medical science will be presented. Some fallacies and pitfalls that creators and users should be aware of will also be noted.

4.2 Construction of an Ontology

A large amount of knowledge about genes has been stored in public databases [7]. Detailed classifications, controlled vocabularies, and organized terminology are widely used in different areas of science and technology [8]. Their relatively recent introduction into molecular biology has been crucial for progress in the analysis of genomics and massive proteomics experiments [8]. One of the most challenging problems in bioinformatics, given all the information about the genes in the databases, is determining the relationships between the genes [7]. The interesting purpose for determining these relationships is to know if genes are related and how closely they are related based on existing knowledge about their biological roles [7]. A critical element of the computational infrastructure required for functional genomics is a shared language for communicating biological data and knowledge [9]. The construction of the ontologies, including terminology, classification, and entity relations, requires considerable effort including the analysis of massive

amounts of literature [8]. The GO provides taxonomy of concepts and their attributes for annotating gene products. As the GO increases in size, its ongoing construction and maintenance become more challenging [9].

Ontological frameworks can provide a shared language for communicating biological information and thereby integrate biological knowledge and generalize the data worldwide. Presently, there is a continuously increasing number of groups developing ontologies in assorted biological domains. These varied and disparate efforts can be beneficial if certain standard criteria are met. The general prerequisites are that the ontologies are not overlapping, that they are accepted and used by the community, and that they are well principled. A short summary of the process of ontology construction is presented in Table 4.1.

Since present electronic knowledge representation is becoming more and more pervasive both in the form of formal ontologies and less formal reference vocabularies, the developers of clinical knowledge bases need to reuse these resources [10]. Such reuse requires a new generation of tools for ontology development and management [10]. To assist users in developing and maintaining ontologies, a number of tools have been developed [11]. Lambrix et al. [11] tested several ontologies including Protege-2000, Chimaera, DAG-Edit, and OilEd and found that no system was preferred in all situations, but each system had its own strengths and weaknesses. Of the several mentioned tools, Protege-2000 is the best known. Protege-2000 (http://protege.stanford.edu) is an open source tool that assists users in the construction of large electronic knowledge bases [12]. It has an intuitive user interface that enables developers to create and edit domain ontologies [12]. Numerous plug-ins provide alternative visualization mechanisms, enable management of multiple ontologies, allow the use of interference engines and problem solvers with Protege ontologies, and provide other functionality [12]. As a summary, Protege-2000 is a general-purpose knowledge-acquisition tool that facilitates domain experts and developers to record, browse, and maintain domain knowledge in

Table 4.1 Principles of Ontology Construction

Principles	*Brief Description*
Background and fundamental principles	Delineating the criteria, upon which an ontology is made, is important. It must first define ontology research and its intersection with computer science. In order to reason upon and draw inferences from data to which ontology has been applied, it is absolutely essential that the relationships be carefully defined, otherwise the data entry is insecure and the results are unpredictable.
Survey of existing ontologies	Collection of the relevant biological ontologies that are currently available is necessary. This should include all of those included in the Open Biological Ontologies (OBO), the Gene Ontology (GO), the MGED Ontology, the NCI Thesaurus, eVOC, the Plant Ontology Consortium, the Foundational Model of Anatomy (FMA), Reactome, and the Sequence Ontology (SO). This should cover the scope of the ontology, the relationships used and the logical inferences these support, available data sets to which the ontology has been applied, and the format(s) in which the ontology is available.
Ontology creation and critique	Proper technique for developing high-quality ontologies should be selected, and then a process of ontology evaluation, in which the constructor can see the relative pros and cons for problematic terms and relationships, must be performed.

knowledge bases [12, 13]. In 2003, Shankar et al. [13] illustrated a knowledge-acquisition wizard that they built around Protege-2000. According to their presentation, the wizard provided an environment that was more intuitive to domain specialists for entering knowledge, and to domain specialists and practitioners for reviewing the knowledge entered [13]. In 2001, Tu and Musen reported the use of the Protege-2000 knowledge engineering environment to build: (1) a patient-data information model, (2) a medical-specialty model, and (3) a guideline model that formalizes the knowledge needed to generate recommendations regarding clinical decisions and actions. They also showed how the use of such models allows development of alternative decision-criteria languages and allows systematic mapping of the data required for guideline execution from patient data contained in electronic medical record systems [14]. Jiang et al. [15] performed an interesting study whose main objective was to explore the potential role of formal concept analysis (FCA) in a context-based ontology building support in a clinical domain. They developed an ontology building support system that integrated an FCA module with a natural language processing (NLP) module, and the user interface of the system was developed as a Protege-2000 Java tab plug-in [15]. In this work, a collection of 368 textual discharge summaries and a standard dictionary of Japanese diagnostic terms (MEDIS ver2.0) were used as the main knowledge sources [15]. Jiang et al. [15] reported that stability was shown on the MEDIS-based medical concept extraction with high precision. They said that under the framework of their ontology building support system using FCA, the clinical experts could reach a mass of both linguistic information and context-based knowledge that was demonstrated as useful to support their ontology building tasks [15]. In 2003, Yeh et al. [9] assessed the applicability of Protege-2000 to the maintenance and development of GO by transferring GO to Protege-2000 in order to evaluate its suitability for GO. The graphical user interface supported browsing and editing of GO. Using Protege-2000, they tested and proved the ability to make changes and extensions to GO to refine the semantics of attributes and classify more concepts [9]. In 2002 Kohler and Schulze-Kremer [16] implemented a system for intelligent semantic integration and querying of federated databases by using three main components: (1) a component which enables SQL access to integrated databases by database federation (MARGBench); (2) an ontology-based semantic metadatabase (SEMEDA); and (3) an ontology-based query interface (SEMEDA-query). Since SEMEDA is implemented as three-tiered Web application, database providers can enter all relevant semantic and technical information about their databases by themselves via a Web browser [16].

Li et al. [10] said that medical experts with little or no computer science experience need tools that will enable them to develop knowledge bases and provide capabilities for directly importing knowledge not only from formal knowledge bases but also from reference terminologies. They said that the portions of knowledge bases that were imported from disparate resources then need to be merged or aligned to one another in order to link corresponding terms, to remove redundancies, and to resolve logical conflicts [10]. In 2000, Li et al. [10] developed a suite of tools for knowledge base management based on the Protege-2000 environment for ontology development and knowledge acquisition, an application for incorporating information from remote knowledge sources such as UMLS into a Protege knowledge base.

In 2002, Demir et al. [17] presented the architecture of an integrated environment named Patika (Pathway Analysis Tool for Integration and Knowledge Acquisition), a composition of a server-side, scalable, object-oriented database and client-side editors to provide an integrated, multiuser environment for visualizing and manipulating networks of cellular events. Demir et al. [17] said that this tool features automated pathway layout, functional computation support, advanced querying, and a user-friendly graphical interface. They noted that Patika would be a valuable tool for rapid knowledge acquisition, microarray generated large-scale data interpretation, disease gene identification, and drug development [17].

With the rapid advancement of biomedical science and the development of high-throughput analysis methods, the extraction of various types of information from biomedical texts has become critical [18]. The description of genes in databases by keywords helps the nonspecialist to quickly grasp the properties of a gene and increases the efficiency of computational tools that are applied to gene data [19]. Since the association of keywords to genes or protein sequences is a difficult process that ultimately implies examination of the literature related to a gene, a procedure to derive keywords from the set of scientific abstracts related to a gene is necessary [19]. Because automatic functional annotations of genes are quite useful for interpreting large amounts of high-throughput data efficiently, the demand for automatic extraction of information related to gene functions from text has been increasing [18]. Blaschke and Valencia [8] proposed a method that automatically generates classifications of gene-product functions using bibliographic information. Blaschke and Valencia [8] noted that the analysis of a large structure built for yeast gene-products, and their detailed inspection of various examples showed encouraging properties. They also noted that the comparison with the well-accepted GO points to different situations in which the automatically derived classification could be useful for assisting human experts in the annotation of ontologies [8]. In 2004, Pérez et al. [19] developed a new system based on the automated extraction of mappings between related terms from different databases using a model of fuzzy associations that could be applied with all generality to any pair of linked databases. Pérez et al. [19] tested the system by annotating genes of the Swiss-Prot database with keywords derived from the abstracts linked to their entries (stored in the MEDLINE database of scientific references) and found that the performance of the annotation procedure was much better for Swiss-Prot keywords (recall of 47%, precision of 68%) than for GO terms (recall of 8%, precision of 67%). In 2005, Koike et al. [18] developed a method for automatically extracting the biological process functions of genes/protein/families based on GO from text using a shallow parser and sentence structure analysis techniques. When the gene/protein/family names and their functions are described in ACTOR (doer of action) and OBJECT (receiver of action) relationships, the corresponding GO-IDs are assigned to the genes/proteins/families and the gene/protein/family names are recognized using the gene/protein/family name dictionaries [18]. Koike et al. [18] noted that a preliminary experiment demonstrated that this method had an estimated recall of 54% to 64% with a precision of 91% to 94% for actually described functions in abstracts and it extracted more than 190,000 gene-GO relationships and 150,000 family-GO relationships for major eukaryotes when applied to the PubMed.

In order to aid in hypothesis-driven experimental gene discovery, a computer application is necessary for the automatic retrieval of signal transduction data from electronic versions of scientific publications using NLP techniques, as well as for visualizing representations of regulatory systems. In 2004, Zhou and Cui [7] developed GeneInfoViz, a Web tool for batch retrieval of gene information and construction and visualization of gene relation networks. With this tool, users can batch search for a group of genes and get the GO terms that were associated with the genes and directed acyclic graphs that were generated to show the hierarchical structure of the GO tree [7]. To summarize, GeneInfoViz calculates an adjacency matrix to determine whether the genes are related, and, if so, how closely they are related based on biological processes, molecular functions, or cellular components with which they are associated, and then it displays a dynamic graph layout of the network among the selected genes [7]. Schober et al. [20] mentioned a new tool, the Gandr (gene annotation data representation), an ontological framework for laboratory-specific gene annotation which could be visualized as an interactive network of nodes and edges representing genes and their functional relationships. Basically, Gandr uses Protege-2000 for editing, querying, and visualizing microarray data and annotations [20]. With this tool, genes can be annotated with provided, newly created, or imported ontological concepts, and annotated genes can inherit assigned concept properties and can be related to each other [20]. Schober et al. [20] said that Gandr could allow for immediate and associative gene context exploration.

4.3 General Evolution of GO Structures and General Annotation Strategy of Assigning GO Terms to Genes

GO is a set of controlled vocabulary modeled in directed acyclic graphs. It is both a term system by itself, and it is used as the dictionary to annotate other genes. This duality should be emphasized.

4.3.1 General Evolution of GO Structures

Basically, the three organizing principles of GO are molecular function, biological process, and cellular component (Table 4.2). For a gene product, it might have one or more molecular functions and can be used in one or more biological processes. Also it might be associated with one or more cellular components.

Table 4.2 The Three Organizing Principles of GO

Principles	*Brief Description*
Molecular function	Molecular function is the activity at the molecular level (e.g., transporter, binding).
Biological process	A biological process is series of events accomplished by one or more ordered assemblies of molecular functions. In one process, there must be at least one function. It should be noted, however, that the biological process does not reflex dynamics, and therefore, they are not equivalent to pathway (e.g., purine metabolism).
Cellular component	Cellular component is a part of a large anatomical structure or a gene product group (e.g., nucleus, endoplasmic reticulum).

In an evolution of a GO structure, identification of the three organizing principles is necessary. This identification can be a base for further annotation. An example of the identification of the three organizing principles in a recent study of human hemoglobin by Wiwanitkit [21] is presented in Table 4.3. This example is an easy one since the quoted gene product "human hemoglobin" has only one molecular function (transporter) and is used in only one biological process (oxygen transportation). Also, it associates with only one cellular component (cytosol).

4.3.2 General Annotation Strategy of Assigning GO Terms to Genes

As previously mentioned, assigning of GO terms to genes has a lot of benefit in clarifying the function of a gene product or gene. This can be applied for comparative study for the gene expressions of genes. Presently, many collaborating databases annotate their gene products with GO terms, providing references and indicating what kind of evidence is available to support the annotations (Table 4.4). By browsing any of the contributing databases, users can find that each gene or gene product has a list of associated GO terms. In addition, users can predict a function of a gene product by the GO tools. To simplify and help the reader better understand this, the author presents an example of using a public GO tool, GoFigure, to study the gene product of bacterial hemoglobin (cytochrome O: P04252) in a recent study by Wiwanitkit [21]. First, the known sequence of bacterial hemoglobin is submitted via GoFigure and the derived results are shown in Figure 4.1. Here, it can be shown that bacterial hemoglobin has more complicated biological functions and is involved in more biological processes than human hemoglobin (Table 4.3).

4.4 Applications of Gene Ontology in Biological and Medical Science

4.4.1 Application of Gene Ontology in Biological Science

Gene ontology has several advantages in molecular biology. Basically, molecular biology has a communication problem in that there are many databases each using their own labels and categories for storing data objects and some using identical labels and categories but with different meanings [22]. There are many databases that use their own labels and categories for storing data objects and some use identical labels and categories but with a different meaning [23]. Conversely, a concept is often found under different names [23]. Schulze-Kremer [22] said that this situation could only be improved by either defining individual semantic interfaces between each pair of databases (complexity of order n^2) or by implementing one agreeable, transparent, and computationally tractable semantic repository and linking each

Table 4.3 The Three Organizing Principles of Human Hemoglobin

Principles	*Brief Description*
Molecular function	Transporter
Biological process	Oxygen transportation
Cellular component	Cytosol

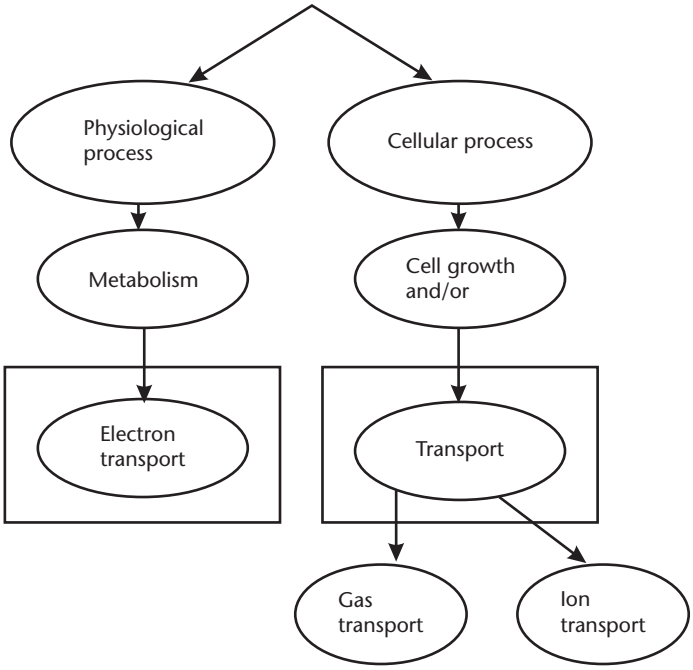


Figure 4.1 Hierarchies of bacterial hemoglobin derived from GoFigure study. (*From:* [21]. © 2005 Viroj Wiwanitkit. Reprinted with permission.)

database to it (complexity of order *n*). Ontology is a means to provide a semantic repository to systematically order relevant concepts in molecular biology and to bridge the different notions in various databases by explicitly specifying the meaning of and relation between the fundamental concepts in an application domain [22]. Several applied GO tools have been developed which are useful for biology and medicine (Table 4.4).

4.4.2 Application of Gene Ontology in Medical Science

Presently, gene ontology is applied for advance research in medicine. The gene expressions in many diseases are analyzed based on the gene ontology principle. A group of diseases, which is widely investigated, is malignancy. There are many recent publications in cancer research based on the advances in gene ontology. In 2003, Cunliffe et al. [24] studied the gene expression response of breast cancer to growth regulators. In this study, Cunliffe et al. defined the dynamic transcriptional effects elicited in MCF7, T-47D, and MDA-MB-436 breast cancer cell lines by nine regulators of growth and differentiation (17beta-estradiol, antiestrogens fulvestrant and tamoxifen, progestin R5020, antiprogestin RU486, all-trans-retinoic acid, epidermal growth factor, mitogen-activated protein/extracellular signal-regulated kinase 1/2 inhibitor U0126, and phorbol ester 12-O-tetradecanoylphorbol-13-acetate) and compared the patterns of gene regulation to published tumor expression profiles. Gene ontology analysis was used to highlight functionally distinct biological responses to different mitogens and significant correlations were identified between several clusters of drug-responsive genes and

Table 4.4 Some Examples of Applied GO Tools in Biological and Medical Science

Tools	*Description*
EBI SRS server [25]	This server (http://srs.ebi.ac.uk) has become an integration system for both data retrieval and sequence analysis applications. The EBI SRS server is a primary gateway to major databases in the field of molecular biology that are produced and supported at EBI, as well as a European public access point to the MEDLINE database provided by U.S. National Library of Medicine (NLM). The new additions include: concept of virtual databases, integration of XML databases like the Integrated Resource of Protein Domains and Functional Sites (InterPro), Gene Ontology, MEDLINE, and Metabolic Pathways.
Proteomic Investigation Strategy for Mammals (PRISM) [26]	This systematic, analytical approach combines subcellular fractionation, multidimensional liquid chromatography-tandem mass spectrometry-based protein shotgun sequencing, and two newly developed computer algorithms, STATQUEST and GOClust, as a means to rapidly identify, annotate, and categorize thousands of expressed mammalian proteins. Automated clustering of the identified proteins into GO annotation groups allowed for streamlined analysis of the large data set, revealing interesting and physiologically relevant patterns of tissue and organelle specificity. Therefore, this tool offers an effective platform for in-depth investigation of complex mammalian proteomes.
Genome Information Management System (GIMS) [27]	This tool is an object database that integrates genomic data with data on the transcriptome, protein-protein interactions, metabolic pathways and annotations, such as GO terms and identifiers. The resulting system supports the running of analyses over this integrated data resource and provides comprehensive facilities for handling and interrelating the results of these analyses.
ToPNet [28]	This is a new tool for the combined visualization and exploration of gene networks and expression data. It provides various ways of restricting, manipulating, and combining biological networks according to annotation data and presents results to the user via different visualization procedures and hyperlinks to the underlying data sources.
GoFish [29]	This is a Java application that allows users to search for gene products with particular GO attributes, or combinations of attributes. GoFish ranks gene products by the degree to which they satisfy a Boolean query.
Expressed Sequence Tag Information Management and Annotation (ESTIMA) [30]	This tool consists of a relational database schema and a set of interactive query interfaces. These are integrated with a suite of Web-based tools that allow a user to query and retrieve information. Further, query results are interconnected among the various EST properties. ESTIMA has several unique features. Users may run their own expressed sequence tag (EST) processing pipeline, search against preferred reference genomes, and use any clustering and assembly algorithm.
Gene Ontology Automated Lexicon (GOAL) [31]	This is a Web-based application for the identification of functions and processes regulated in microarray and Serial Analysis of Gene Expression (SAGE) experiments. GOAL allows a seamless and high-level analysis of expression profiles.
PeerGAD [32]	This tool is a Web-based database-driven application that allows community-wide peer-reviewed annotation of prokaryotic genome sequences. PeerGAD incorporates several innovative design and operation features and accepts annotations pertaining to gene naming, role classification, and gene translation and annotation derivation. The annotator tool in PeerGAD is built around a genome browser that offers users the ability to search and navigate the genome sequence.
FatiGO [33]	This is a Web tool for finding significant associations of GO terms with groups of genes. FatiGO includes GO associations for diverse organisms (human, mouse, fly, worm, and yeast) and the TrEMBL/SwissProt correspondences from the European Bioinformatics Institute.
GOTree Machine (GPTM) [34]	This tool generates a GOTree, a tree-like structure to navigate the Gene Ontology Directed Acyclic Graph for input gene sets. It has a broad application in functional genomic, proteomic and other high-throughput methods that generate large sets of interesting genes; its primary purpose is to help users sort for interesting patterns in gene sets.

genes that discriminate estrogen receptor status or disease outcome in patient samples [24]. Cunliffe et al. concluded that the majority of estrogen receptor status discriminators were not responsive in their dataset and are therefore likely to reflect underlying differences in histogenesis and disease progression rather than growth factor signaling. In 2004, Shi et al. [35] studied effects of resveratrol (RE) on gene expression in renal cell carcinoma. They profiled and analyzed the expression of 2,059 cancer-related genes in a RCC cell line RCC54 treated with RE [35]. In this study, biological functions of 633 genes were annotated based on biological process ontology and clustered into functional categories and 29 highly differentially expressed genes in RE-treated RCC54, and the potential implications of some gene expression alterations in RCC carcinogenesis were identified [35]. Arciero et al. [36] summarized the current literature on selected biomarkers for breast cancer. They also discussed the functional relationships, and grouped the selected genes based on a GO classification [36]. They noted that choosing the right combination of biomarkers was challenging, because (1) multiple pathways are involved and (2) up to 62 genes and their protein products are potentially involved in breast cancer-related mechanisms [36]. In 2004, Sulman et al. [37] reported a study on genomic annotation of the meningioma tumor suppressor locus on chromosome 1p34. In this study, a high-resolution integrated map of the region was constructed (CompView) to identify all markers in the smallest region of overlapping deletion (SRO) and a regional somatic cell hybrid panel was used to more precisely localize those markers identified in CompView as within or overlapping the region [37]. According to this study, a total of 59 genes were ordered within the SRO and 17 of these were selected as likely candidates based on annotation using GO Consortium terms, including the MUTYH, PRDX1, FOXD2, FOXE3, PTCH2, and RAD54L genes [37]. Sulman et al. [37] noted that this annotation of a putative tumor suppressor locus provided a resource for further analysis of meningioma candidate genes. In 2003, Ahn et al. [38] profiled differentially expressed transcriptome and proteome in six-paired leiomyoma and normal myometrium. In this study, screening up to 17,000 genes identified 21 unregulated and 50 down regulated genes, and the gene-expression profiles were classified into mutually dependent 420 functional sets, resulting in 611 cellular processes according to the gene ontology [38]. Also, protein analysis using two-dimensional gel electrophoresis identified 33 proteins (17 unregulated and 16 down regulated) of more than 500 total spots, which was classified into 302 cellular processes. Ahn et al. [38] mentioned that the gene ontology analysis could overcome the complexity of expression profiles of cDNA microarray and two-dimensional protein analysis via its cellular process-level approach; therefore, a valuable prognostic candidate gene with relevance to disease-specific pathogenesis could be found at cellular process levels. In addition to oncology, the applications of the gene ontology for the diseases in other groups are reported. The application in vaccine development is also noted [39]. The advances in gene ontology become a useful tool in medicine at present.

References

[1] Ashburner, M., and S. Lewis, "On Ontologies for Biologists: The Gene Ontology—Untangling the Web," *Novartis. Found. Symp.,* Vol. 247, 2002, pp. 66–80.

[2] Gene Ontology Consortium, "Creating the Gene Ontology Resource: Design and Implementation," *Genome. Res.,* Vol. 11, No. 8, 2001, pp. 1425–1433.

[3] Stevens, R., C. A. Goble, and S. Bechhofer, "Ontology-Based Knowledge Representation for Bioinformatics," *Brief. Bioinform.,* Vol. 1, No. 4, 2000, pp. 398–414.

[4] Schulze-Kremer, S., "Ontologies for Molecular Biology and Bioinformatics," *In Silico Biol.*, Vol. 2, No. 3, 2002, pp. 179–193.

[5] Lewis, S., M. Ashburner, and M. G. Reese, "Annotating Eukaryote Genomes," *Curr. Opin. Struct. Biol.,* Vol. 10, No. 3, 2000, pp. 349–354.

[6] Doms, A., and M. Schroeder, "GoPubMed: Exploring PubMed with the Gene Ontology," *Nucleic Acid Res.,* Vol. 33 (Web Server issue), 2005, pp. W783–W786.

[7] Zhou, M., and Y. Cui, "GeneInfoViz: Constructing and Visualizing Gene Relation Networks," *In Silico Biol.,* Vol. 4, No. 3, 2004, pp. 323–333.

[8] Blaschke, C., and A. Valencia, "Automatic Ontology Construction from the Literature," *Genome Inform. Series,* Vol. 13, 2002, pp. 201–213.

[9] Yeh, I., et al., "Knowledge Acquisition, Consistency Checking and Concurrency Control for Gene Ontology (GO)," *Bioinformatics,* Vol. 19, No. 2, 2003, pp. 241–248.

[10] Li, Q., et al., "Ontology Acquisition from On-Line Knowledge Sources," *Proc. AMIA Symp.,* Vol. 1, 2000, pp. 497–501.

[11] Lambrix, P., M. Habbouchem, and M. Perez, "Evaluation of Ontology Development Tools for Bioinformatics," *Bioinformatics*, Vol. 19, No. 12, 2003, pp. 1564–1571.

[12] Noy, N. F., et al., "Protege-2000: An Open-Source Ontology-Development and Knowledge-Acquisition Environment," *AMIA Annu. Symp. Proc.*, 2003, p. 953.

[13] Shankar, R. D., S. W. Tu, and M. A. Musen, "A Knowledge-Acquisition Wizard to Encode Guidelines," *AMIA Annu. Symp. Proc.,* Vol. 1, 2003, p. 1007.

[14] Tu, S. W., and M. A. Musen, "Modeling Data and Knowledge in the EON Guideline Architecture," *Medinfo*, Vol. 10, Pt. 1, 2001, pp. 280–284.

[15] Jiang, G., et al., "Context-Based Ontology Building Support in Clinical Domains Using Formal Concept Analysis," *Int. J. Med. Inform.,* Vol. 71, No. 1, 2003, pp. 71–81.

[16] Kohler, J., and S. Schulze-Kremer, "The Semantic Metadatabase (SEMEDA): Ontology Based Integration of Federated Molecular Biological Data Sources," *In Silico Biol.,* Vol. 2, No. 3, 2002, pp. 219–231.

[17] Demir, E., et al., "PATIKA: An Integrated Visual Environment for Collaborative Construction and Analysis of Cellular Pathways," *Bioinformatics*, Vol. 18, No. 7, 2002, pp. 996–1003.

[18] Koike, A., Y. Niwa, and T. Takagi, "Automatic Extraction of Gene/Protein Biological Functions from Biomedical Text," *Bioinformatics,* Vol. 21, No. 7, 2005, pp. 1227–1236.

[19] Pérez, A. J., et al., "Gene Annotation from Scientific Literature Using Mappings Between Keyword Systems," *Bioinformatics*, Vol. 20, No. 13, 2004, pp. 2084–2091.

[20] Schober, D., et al., "GandrKB—Ontological Microarray Annotation and Visualization," *Bioinformatics,* Vol. 21, No. 11, 2005, pp. 2785–2786.

[21] Wiwanitkit, V., "Bacterial Hemoglobin: What Is Its Molecular Function and Biological Process?" *HAEMA*, 2005.

[22] Schulze-Kremer, S., "Ontologies for Molecular Biology," *Pac. Symp. Biocomput.*, Vol. 1, 1998, pp. 695–706.

[23] Schulze-Kremer, S., "Adding Semantics to Genome Databases: Towards an Ontology for Molecular Biology," *Proc. Int. Conf. Intell. Syst. Mol. Biol.*, Vol. 5, 1997, pp. 272–275.

[24] Cunliffe, H. E., et al., "The Gene Expression Response of Breast Cancer to Growth Regulators: Patterns and Correlation with Tumor Expression Profiles," *Cancer Res.,* Vol. 63, No. 21, 2003, pp. 7158–7166.

[25] Zdobnov, E. M., et al., "The EBI SRS Server-New Features," *Bioinformatics*, Vol. 18, No. 8, 2002, pp. 1149–1150.

[26] Kislinger, T., et al., "PRISM: A Generic Large Scale Proteomic Investigation Strategy for Mammals," *Mol. Cell Proteomics*, Vol. 2, No. 2, 2003, pp. 96–106.

[27] Cornell, M., et al., "GIMS: An Integrated Data Storage and Analysis Environment for Genomic and Functional Data," *Yeast,* Vol. 20, No. 15, 2003, pp. 1291–1306.

[28] Hanisch, D., F. Sohler, and R. Zimmer, "ToPNet—An Application for Interactive Analysis of Expression Data and Biological Networks," *Bioinformatics*, Vol. 20, No. 9, 2004, pp. 1470–1471.

[29] Berriz, G. F., et al., "GoFish Finds Genes with Combinations of Gene Ontology Attributes," *Bioinformatics,* Vol. 19, No. 6, 2003, pp. 788–789.

[30] Kumar, C. G., et al., "ESTIMA: A Tool for EST Management in a Multi-Project Environment," *BMC Bioinformatics,* Vol. 5, No. 1, 2004, p. 176.

[31] Volinia, S., et al., "GOAL: Automated Gene Ontology Analysis of Expression Profiles," *Nucleic Acids Res.,* Vol. 1, No. 32 (Web Server issue), 2004, pp. W492–W499.

[32] D'Ascenzo, M. D., A. Collmer, and G. B. Martin, "PeerGAD: A Peer-Review-Based and Community-Centric Web Application for Viewing and Annotating Prokaryotic Genome Sequences," *Nucleic Acids Res.,* Vol. 32, No. 10, 2004, pp. 3124–3135.

[33] Al-Shahrour, F., R. Diaz-Uriarte, and J. Dopazo, "FatiGO: A Web Tool for Finding Significant Associations of Gene Ontology Terms with Groups of Genes," *Bioinformatics*, Vol. 20, No. 4, 2004, pp. 578–580.

[34] Zhang, B., et al., "GOTree Machine (GOTM): A Web-Based Platform for Interpreting Sets of Interesting Genes Using Gene Ontology Hierarchies," *BMC Bioinformatics*, Vol. 5, No. 1, 2004, p. 16.

[35] Shi, T., et al., "Effects of Resveratrol on Gene Expression in Renal Cell Carcinoma," *Cancer Biol. Ther.,* Vol. 3, No. 9, 2004, pp. 882–888.

[36] Arciero, C., et al., "Functional Relationship and Gene Ontology Classification of Breast Cancer Biomarkers," *Int. J. Biol. Markers,* Vol. 18, No. 4, 2003, pp. 241–272.

[37] Sulman, E. P., P. S. White, and G. M. Brodeur, "Genomic Annotation of the Meningioma Tumor Suppressor Locus on Chromosome 1p34," *Oncogene*, Vol. 23, No. 4, 2004, pp. 1014–1020.

[38] Ahn, W. S., et al., "Targeted Cellular Process Profiling Approach for Uterine Leiomyoma Using CDNA Microarray, Proteomics and Gene Ontology Analysis," *Int. J. Exp. Pathol.,* Vol. 84, No. 6, 2003, pp. 267–279.

[39] Schonbach, C., T. Nagashima, and A. Konagaya, "Textmining in Support of Knowledge Discovery for Vaccine Development," *Methods,* Vol. 34, No. 4, 2004, pp. 488–495.

CHAPTER 5

Protein Ontology

Amandeep S. Sidhu, Tharam S. Dillon, and Elizabeth Chang

Two factors dominate current developments in structural bioinformatics, especially in protein informatics and related areas: (1) the amount of raw data is increasing, very rapidly; and (2) successful application of data to biomedical research requires carefully and continuously curated and accurately annotated protein databanks. In this chapter, we introduce the concepts for annotating protein data using our protein ontology. We first describe and review existing approaches for protein annotation. We then describe the advantages of semantic integration of protein data using Web ontology language, in comparison to annotating using automatic search and analyzing using text mining. The rest of chapter is devoted to the use of the protein ontology for the annotation of protein databases. The protein ontology is available at http://www.proteinontology.info/.

5.1 Introduction

A large number of diverse bioinformatics sources are available today. The future of biological sciences promises more data. No individual data source will provide us with answers to all the queries that we need to ask. Instead, knowledge has to be composed from multiple data sources to answer the queries. Even though multiple databases may cover the same data, their focus might be different. For example, even though Swiss-Prot [1–3] and PDB [4–7] are both protein databases, we might want to get information about sequence as well as structure of a particular protein. In order to answer that query, we need to get data about the protein from both sources and combine them in a consistent fashion [8]. In this postgenomic era, isolating specific data from heterogeneous protein data sources has become a major issue. Extracting relevant protein data without omitting data and without introducing irrelevant information is a challenge. The main problems lie in interpreting protein nomenclature and the multitude of synonyms and acronyms, which can be used to describe a single protein, identifying data provenance, and extracting data from computationally unprocessed natural language.

5.2 What Is Protein Annotation?

As biological databases undergo a very rapid growth, two major consequences are emerging. First, an efficient utilization of databases to provide easy management and access to this data is continuously developing. A second consequence is a necessary formalization of biological concepts and relationships among these concepts via the creation of ontologies.

In the context of protein data, annotation generally refers to all information about a protein other than protein sequence. In a collection of protein data, each protein is labeled at least by an identifier and is usually complemented by annotations as free text or as codified information, such as names of authors responsible for that protein, submission date of protein data, and so on. Annotations become a challenge in proteomics considering the size and complexity of protein complexes and their structures.

For our purposes, we will mainly deal with two main sources of protein annotations: (1) those taken from various protein data sources submitted by the authors of protein data themselves from their published experimental results; and (2) those that we name annotation that are obtained by an annotator or group of annotators through analysis of raw data (typically a protein sequence or atomic structure description) with various tools that extract biological information from other protein data collections.

5.3 Underlying Issues with Protein Annotation

Traditional approaches to integrate protein data generally involved keyword searches, which immediately excludes unannotated or poorly annotated data. It also excludes proteins annotated with synonyms unknown to the user. Of the protein data that is retrieved in this manner, some biological resources do not record information about the data source, so there is no evidence of the annotation. An alternative protein annotation approach is to rely on sequence identity, structural similarity, or functional identification. The success of this method is dependent on the family the protein belongs to. Some proteins have a high degree of sequence identity, structural similarity, or similarity in functions that are unique to members of that family. Consequently, this approach cannot be generalized to integrate the protein data. Clearly, these traditional approaches have limitations in capturing and integrating data for protein annotation. For these reasons, we have adopted an alternative method that does not rely on keywords or similarity metrics, but instead uses ontology. Briefly, *ontology* is a means of formalizing knowledge; at the minimum, ontology must include concepts or terms relevant to the domain, definitions of concepts, and defined relationships between the concepts.

We have built protein ontology [9–17] to integrate protein data formats and provide a structured and unified vocabulary to represent protein synthesis concepts. Protein ontology (PO) provides integration of heterogeneous protein and biological data sources. PO converts the enormous amounts of data collected by geneticists and molecular biologists into information that scientists, physicians, and other health care professionals and researchers can use to easily understand the mapping

of relationships inside protein molecules, interaction between two protein molecules, and interactions between protein and other macromolecules at cellular level. PO also helps to codify proteomics data for analysis by researchers. Before we discuss the PO framework in detail, in the next section we provide an overview of various protein databanks and earlier attempts to integrate protein data from these data sources.

5.3.1 Other Biomedical Ontologies

In this section we will discuss various biomedical ontology works related to protein ontology. Gene ontology (GO) [18, 19] defines a hierarchy of terms related to genome annotation. GO is a structured network consisting of defined terms and relationships that describe molecular functions, biological processes, and cellular components of genes. GO is clearly defined and modeled for numerous other biological ontology projects. So far, GO has been used to describe the genes of several model organisms (*Saccharomyces cerevisiae*, *Drosophila melanogaster*, *Mus musculus*, and others).

RiboWEB [20] is an online data resource for Ribosome, a vital cellular apparatus. It contains a large knowledge base of relevant published data and computational modules that can process this data to test hypotheses about ribosome's structure. The system is built around the concept of ontology. Diverse types of data taken principally from published journal articles are represented using a set of templates in the knowledge base, and the data is linked to each other with numerous connections.

Protein Data Bank (PDB) has recently released versions of the PDB Exchange Dictionary and the PDB archival files in XML format, collectively named PDBML [21]. The representation of PDB data in XML builds from content of the PDB Exchange Dictionary, both for assignment of data item names and defining data organization. PDB exchange and XML representations use the same logical data organization. A side effect of maintaining a logical correspondence with PDB exchange representation is that PDBML lacks the hierarchical structure characteristic of XML data.

PRONTO [22] is a directed acyclic graph (DAG)-based ontology induction tool that constructs a protein ontology including protein names found in MEDLINE abstracts and in UniProt. It is a typical example of text mining the literature and the data sources. It cannot be classified as protein ontology as it only represents relationship between protein literatures and does not formalize knowledge about protein synthesis process. Ontology for protein domain must contain terms or concepts relevant to protein synthesis, describing protein sequence, structure, and function and relationships between them. While defining PO we made an effort to emulate the protein synthesis and describe the concepts and relationships that describe it.

There is a need for an agreed-upon standard to semantically describe protein data. PO addresses this issue by providing clear and unambiguous definitions of all major biological concepts of protein synthesis process and the relationships between them. PO provides a unified controlled vocabulary both for annotation data types and for annotation data itself.

5.3.2 Protein Data Frameworks

The identification of all the genes that encode proteins in the genome of an organism is essential but not sufficient for understanding how these proteins function in making up a living cell. The number of different fundamental proteins in an organism often substantially exceeds the number of genes due to generation of protein isoforms by alternative RNA processing as well as by covalent modifications of precursor proteins. To cope with the complexity of protein sequence and functional information, annotated databases of protein sequence, structure, and function with high interoperability are needed. The major protein databases are the most comprehensive sources of information on proteins. In addition to these universal databases that cover proteins from all the species, there are collections that store information about specific families or groups of proteins, or about proteins of specific organisms. Here we will give a brief overview of major protein collections and their corresponding annotations.

5.3.2.1 Worldwide PDB (wwPDB)

The wwPDB [23] represents a milestone in the evolution of PDB, which was established in 1971 at Brookhaven National Laboratory as the sole international repository for three-dimensional structural data of biological macromolecules. Since July 1, 1999, the PDB has been managed by three member institutions of the RCSB: Rutgers, The State University of New Jersey; the San Diego Supercomputer Center at the University of California, San Diego; and the Center for Advanced Research in Biotechnology of the National Institute of Standards and Technology. The wwPDB recognizes the importance of providing equal access to the database both in terms of depositing and retrieving data from different regions of the world. Therefore, the wwPDB members will continue to serve as deposition, data processing, and distribution sites. To ensure the consistency of PDB data, all entries are validated and annotated following a common set of criteria. All processed data is sent to the RCSB, which distributes the data worldwide. All format documentation will be kept publicly available and the distribution sites will mirror the PDB archive using identical contents and subdirectory structure. However, each member of the wwPDB will be able to develop its own Web site, with a unique view of the primary data, providing a variety of tools and resources for the global community.

5.3.2.2 Universal Protein Resource (UniProt)

UniProt [1] joins three major protein databases: PIR-PSD [24], Swiss-Prot [3], and TrEMBL [2]. The Protein Information Resource (PIR) provides an integrated public resource of protein informatics. PIR produces the Protein Sequence Database (PSD) of functionally annotated protein sequences. Swiss-Prot is a protein knowledge base established in 1986 and maintained collaboratively by the Swiss Institute of Bioinformatics (SIB) and the European Bioinformatics Institute (EBI). It strives to provide a high level of annotation, a minimal level of redundancy, a high level of integration with other biomolecular databases, and extensive external documentation. The translation of EMBL nucleotide sequence database (TrEMBL), a supple-

ment to Swiss-Prot, was created in 1996. This supplement contains computer annotated protein sequences not yet integrated with Swiss-Prot. Together, PIR, EBI, and SIB maintain and provide UniProt, a stable, comprehensive, fully classified, and accurately annotated protein knowledge base.

5.3.2.3 Classification of Protein Families

Classification of proteins provides valuable clues of structure, activity, and metabolic role. A number of different classification systems have been developed in recent years to organize proteins. The various existing classification schemes include: (1) hierarchical families of proteins, such as the superfamilies or families in the PIR-PSD, and protein groups in ProntoNet [25]; (2) protein family domains such as those in Pfam [26] and ProDom [27]; (3) sequence motifs or conserved regions as in PROSITE [28] and PRINTS [29]; (4) structural classes, such as in SCOP [30] and CATH [31]; and (5) integrations of various family classifications such as iProClass [32] and InterPro [33]. While each of these databases is useful for particular needs, no classification scheme is by itself adequate for addressing all protein annotation needs.

InterPro is an integrated resource of PROSITE, PRINTS, Pfam, ProDom, SMART [34], and TIGRFAMs [35] for protein families, domains, and functional sites. Each entry in InterPro includes a unique name and short name; an abstract, which provides annotation about protein matching the entry; literature references and links back to relevant databases; and a list of precomputed matches against the whole of SwissProt and TrEMBL.

PIR defines closely related proteins as having at least 50% sequence identity; such sequences are automatically assigned to the same family. The families produced by automatic clustering can be refined to make groups that make biological sense. A PIR superfamily is a collection of families. Sequences in different families in same superfamily have as little as 15–20% sequence identity. The PIR superfamily / family concept [36] is the earliest protein classification based on sequence similarity, and is unique in providing nonoverlapping clustering of protein sequences into a hierarchical order to reflect evolutionary relationships.

5.3.2.4 Online Mendelian Inheritance in Man

Online Mendelian Inheritance in Man (OMIM) is a continuously updated catalog of human genes and genetic disorders. OMIM focuses primarily on inherited or heritable, genetic diseases. It is also considered to be a phenotypic companion to the human genome project. OMIM is based upon the text *Mendelian Inheritance in Man* [37], authored and edited by Victor A. McKusick and a team of science writers and editors at Johns Hopkins University and elsewhere. *Mendelian Inheritance in Man* is now in its twelfth edition. The database contains textual information and references. OMIM is primarily used by physicians and other professionals concerned with genetic disorders, by genetics researchers, and by advanced students in science and medicine.

5.3.3 Critical Analysis of Protein Data Frameworks

Semantics of protein data are usually hard to define precisely because they are not explicitly stated but are implicitly included in database design. Proteomics is not a single, consistent domain; it is composed of various smaller focused research communities, each having a different data format. Data semantics would not be a significant issue if researchers only accessed data from within a single research domain, but this is not usually the case. Typically, researchers require integrated access to data from multiple domains, which requires resolving terms that have slightly different meanings across communities. This is further complicated by observations that the specific community whose terminology is being used by a data source is not explicitly identified and that the terminology evolves over time. For many of the larger, community data sources, the domain is oblivious, the PDB handles protein structure information, the Swiss-Prot protein sequence database provides protein sequence information and useful annotations, and so on. The terminology used in these major data banks does not reflect knowledge integration from multiple protein families. The wwPDB is an effort to integrate protein data from PDB and other major protein databases at EBI, but the work is at too early a stage at the moment to comment. Furthermore, in smaller community data sources terminology is typically selected based on functionality of data source or usage model. Consequently, as queries to protein data models discussed in this section can involve using concepts from other data sources, the data source answering the query will use whatever definitions are most intuitive, annotating the knowledge from various protein families as needed. This kind of protein annotation will be difficult to generalize for all kinds of proteins.

5.4 Developing Protein Ontology

One of the major motivations for developing protein ontology was introduction of the Genomes to Life Initiative [38, 39] close to the completion of Human Genome Project (HGP), which finished in April 2003 [40]. Lessons learned from HGP will guide ongoing management and coordination of GTL. The overall objective of the Protein Ontology Project is: "To correlate information about multiprotein machines with data in major protein databases to better understand sequence, structure and function of protein machines." The objective is achieved to some extent by creating databases of major protein families, based on the vocabulary of PO.

As technologies mature, the shift from single annotation databases being queried by Web-based scripts generating HTML pages to annotation repositories capable of exporting selected data in XML format, either to be further analyzed by remote applications or to undergo a transformation stage to be presented to user in a Web browser, will undoubtedly be one of the major evolutions of protein annotation process. XML is a markup language much like HTML, but XML describes data using hierarchy. An XML document uses the schema to describe data and is designed to be self-descriptive. This allows easy and powerful manipulation of data in XML documents. XML provides syntax for structured documents, but imposes no semantic constraints on the meaning of these documents.

Resource Description Framework (RDF) is a data model for objects or resources and relations between them; it provides a simple semantics for this data model, and these data models can be represented in XML syntax. RDF Schema is a vocabulary for describing properties and classes of RDF resources, with semantics for generalization hierarchies of such properties and classes.

To efficiently represent the protein annotation framework and to integrate all the existing data representations into a standardized protein data specification for the bioinformatics community, the protein ontology needs to be represented in a format that not only enforces semantic constraints on protein data, but can also facilitate reasoning tasks on protein data using semantic query algebra. This motivates the representation of the Protein Ontology Model in Web Ontology Language (OWL). OWL is designed for use by applications that need to process the content of information instead of just presenting information to humans. OWL facilitates greater machine interpretability of Web content than that supported by XML, RDF, and RDF Schema by providing additional vocabulary along with a formal semantics. OWL provides a language for capturing declarative knowledge about protein domain and a classifier that allows reasoning about protein data. Knowledge captured from protein data using OWL is classified in a rich hierarchy of concepts and their interrelationships. OWL is compositional and dynamic, relying on notions of classification, reasoning, consistency, retrieval, and querying. We investigated the use of OWL for making PO using the Protégé OWL Plug-in. OWL is flexible and powerful enough to capture and classify biological concepts of proteins in a consistent and principled fashion. OWL is used to construct PO that can be used for making inferences from proteomics data using defined semantic query algebra.

5.5 Protein Ontology Framework

The ultimate goal of protein annotator framework or PO is to deduce from proteomics data all its biological features and describe all intermediate structures: primary amino acid sequence, secondary structure folds and domains, tertiary three-dimensional atomic structure, quaternary active functional sites, and so on. Thus, complete protein annotation for all types of proteins for an organism is a very complex process that requires, in addition to extracting data from various protein databases, the integration of additional information such as results of protein experiments, analysis of bioinformatics tools, and biological knowledge accumulated over the years. This constitutes a huge mass of heterogeneous protein data sources that need to rightly represented and stored. Protein annotators must be able to readily retrieve and consult this data. Therefore protein databases and man-machine interfaces are very important when defining a protein annotation using protein ontology.

The process of development of a protein annotation based on our protein ontology requires a significant effort to organize, standardize, and rationalize protein data and concepts. First of all, protein information must be defined and organized in a systematic manner in databases. In this context, PO addresses the following problems of existing protein databases: redundancy, data quality (errors, incorrect annotations, and inconsistencies), and the lack of standardization in nomenclature.

The process of annotation relies heavily on integration of heterogeneous protein data. Integration is thus a key concept if one wants to make full use of protein data from collections. In order to be able to integrate various protein data, it is important that communities agree upon concepts underlying the data. PO provides a framework of structured vocabularies and a standardized description of protein concepts which help to achieve this agreement and achieve uniformity in protein data representation.

PO consists of concepts (or classes), which are data descriptors for proteomics data, and the relations among these concepts. PO has: (1) a hierarchical classification of concepts, from general to specific; (2) a list of attributes related to each concept, for each class; and (3) a set of relations between classes to link concepts in ontology in more complicated ways then implied by the hierarchy, to promote reuse of concepts in the ontology. PO provides the concepts necessary to describe individual proteins, but it does not contain individual protein instances. The PO Instance Store contains individual instances for protein complex in the Web Ontology Language (OWL) format.

5.5.1 The ProteinOntology Concept

The main concept in PO is ProteinOntology. For each instance of protein that is entered into PO, the submission information is entered for the ProteinOntology Concept. The ProteinOntology Concept has the following attributes: ProteinOntologyID and ProteinOntologyDescription. The ProteinOntologyID has the following format: PO000000029.

5.5.2 Generic Concepts in Protein Ontology

There are seven subconcepts of the ProteinOntology Concept, called generic concepts, which are used to define complex concepts in other PO classes: Residue, Chain, Atom, Family, AtomicBind, Bind, and SiteGroup. These generic concepts are reused for other definitions of complex concepts in PO. Details and properties of residues in a protein sequence are defined by instances of Residue Concept. Instances of chains of residues are defined in Chain Concept. Three-dimensional structure data of protein atoms is represented as instances of Atom Concept. Defining chain, residue, and atom as individual concepts has the benefit that any special properties or changes affecting a particular chain, residue, or atom can be easily added. Family Concept represents protein superfamily and family details of proteins. Data about binding atoms in chemical bonds like a hydrogen bond, residue links, and salt bridges is entered into the ontology as an instance of AtomicBind Concept. Similarly, the data about binding residues in chemical bonds like disulphide bonds and CIS peptides is entered into the ontology as an instance of Bind Concept. All data related to site groups of the active binding sites of proteins are defined as instances of SiteGroup Concept. Representation of instances of residues and chains of residues are shown as follows:

```
<Residues>
<Residue>LEU</Residue>
<ResidueName>LEUCINE</ResidueName>
<ResidueProperty>1-LETTER CODE: L; FORMULA: C6 H13 N1 O2;
```

```
MOLECULAR WEIGHT: 131.17</ResidueProperty>
</Residues>

<Chains>
<Chain>D</Chain>
<ChainName>CHAIN D</ChainName>
</Chains>
```

5.5.3 The ProteinComplex Concept

The root concept for the definition of protein complexes in the protein ontology is ProteinComplex. The ProteinComplex Concept defines one or more proteins in the complex molecule. There are six subconcepts of ProteinComplex: Entry, Structure, StructuralDomains, FunctionalDomains, ChemicalBonds, and Constraints. These subconcepts define sequence, structure, function, and chemical bindings of the protein complex.

5.5.4 Entry Concept

Entry Concept specifies the details of a protein or a protein complex which are entered into the knowledge base of protein ontology. Protein entry details are entered into Entry Concept as instances of EntrySuperFamily, EntryFamily, SourceDatabaseID, SourceDatabaseName, SubmissionDate, and Classification. These attributes describe the entry in the original protein data source from where it was taken. Entry has three subconcepts: Description, Molecule, and Reference. The Description subconcept describes data about the title of the entry, the authors of the entry, the experiment that produced the entry, and the keywords describing the entry. The second subconcept of Entry is Molecule, which is simply any chemically distinct molecule or compound in a protein complex. MoleculeID uniquely identifies a molecule. MoleculeName is the chemical name of the molecule. MoleculeChain refers to the chain description. BiologicalUnit instance describes the larger biological unit of which the molecule is a part. Engineered identifies whether the molecule is engineered using recombinant technology or chemical synthesis. A specific domain or region of the molecule is defined using Fragment. Mutated molecules of the protein have Mutations Information. Details about various mutations are described in the GeneticDefects Class. A list of synonyms for molecule name are in Synonyms. OtherDetails describes any other information. The Reference subconcept lists the various literature citations of the protein or protein complex described by the instances of CitationTitle, CitationAuthors, CitationEditors, CitationPublication, CitationReference, and CitationReferenceNumbers. A typical instance of Entry is as follows:

```
<Entry>
<ProteinOntologyID>PO00000000007</ProteinOntologyID>
<EntrySuperFamily>HUMAN</EntrySuperFamily>
<EntryFamily>PRION PROTEINS</EntryFamily>
<SourceDatabaseID>1E1P</SourceDatabaseID>
<SourceDatabaseName>PROTEIN DATA BANK</SourceDatabaseName>
<SubmissionDate>09-MAY-00</SubmissionDate>
<Title>HUMAN PRION PROTEIN VARIANT S170N</Title>
<Authors>L.CALZOLAI, D.A.LYSEK, P.GUNTERT, C.VON SCHROETTER,
```

```
R.ZAHN, R.RIEK, K.WUTHRICH</Authors>
<Experiment>NMR, 20 STRUCTURES</Experiment>
<Keywords>PRION PROTEIN</Keywords>
<CitationTitle>NMR STRUCTURES OF THREE SINGLE-RESIDUE VARIANTS OF
THE HUMAN PRION PROTEIN</CitationTitle>
<CitationAuthors>L.CALZOLAI, D.A.LYSEK, P.GUNTERT, C.VON
SCHROETTER, R.ZAHN, R.RIEK, K.WUTHRICH</CitationAuthors>
<CitationPublication>PROC.NAT.ACAD.SCI.USA</CitationPublication>
<CitationReference>V. 97 8340 2000</CitationReference>
<CitationReferenceNumbers>ASTM PNASA6 US ISSN
0027-8424</CitationReferenceNumbers>
</Entry>
```

5.5.5 Structure Concept

Structure Concept describes the protein structure details. Structure has two subconcepts: ATOMSequence and UnitCell. ATOMSequence is an example of the reuse of concepts in PO; it is constructed using generic concepts of Chain, Residue, and Atom. The reasoning is already there in the underlying protein data, as each chain in a protein represents a sequence of residues, and each residue is defined by a number of three-dimensional atoms in the protein structure. Structure Concept defines ATOMSequence, with references to definitions of Chain and Residues, as:

```
<ATOMSequence>
<ProteinOntologyID>PO00000000004</ProteinOntologyID>
<Chain>
<AtomChain>A</AtomChain>
<Residue>
<ATOMResidue>ARG</ATOMResidue>
<Atom>
<AtomID>364</AtomID>
<Symbol>HE</Symbol>
<ATOMResSeqNum>148</ATOMResSeqNum>
-23.549</X>
<Y>3.766</Y>
<Z>-0.325</Z>
<Occupancy>1.0</Occupancy>
<TempratureFactor>0.0</TempratureFactor>
<Element>H</Element>
</Atom>
</Residue>
</Chains>
</ATOMSequence>
```

Protein crystallography data like a, b, c, alpha, beta, gamma, z, and SpaceGroup is described in *UnitCell* Concept.

5.5.6 StructuralDomains Concept

Structural folds and domains defining secondary structures of proteins are defined in StructuralDomains Concept. The subconcepts SuperFamily and Family of generic concept Family are used for identifying the protein family here. The subconcepts of StructuralDomains are Helices, Sheets, and OtherFolds. Helix, which is a subconcept of Helices, identifies a helix using HelixNumber, HelixID, HelixClass,

and HelixLength Instances. Helix has a subconcept HelixStructure which gives the detailed composition of the helix. A typical instance of Helices Concept is:

```
<Helices>
<ProteinOntologyID>PO00000000002</ProteinOntologyID>
<StructuralDomainSuperFamily>HAMSTER</StructuralDomainSuperFamily>
<StructuralDomainFamily>PRION PROTEINS</StructuralDomainFamily>
<Helix>
<HelixID>1</HelixID>
<HelixNumber>1</HelixNumber>
<HelixClass>Right Handed Alpha</HelixClass>
<HelixLength>10</HelixLength>
<HelixStructure>
<HelixChain>A</HelixChain>
<HelixInitialResidue>ASP</HelixInitialResidue>
<HelixInitialResidueSeqNum>144</HelixInitialResidueSeqNum>
<HelixEndResidue>ASN</HelixEndResidue>
<HelixEndResidueSeqNum>153</HelixEndResidueSeqNum>
</HelixStructure>
</Helix>
</Helices>
```

Other secondary structures like sheets and turns (or loops) are represented using concepts of chains and residues in a similar way. Sheets has a subconcept Sheet which describes a sheet using SheetID and NumberStrands. Sheet has a subconcept Strands which describes the detailed structure of a sheet. A typical instance of Sheets Class is:

```
<Sheets>
<ProteinOntologyID>PO00000000001</ProteinOntologyID>
<StructuralDomainSuperFamily>MOUSE</StructuralDomainSuperFamily>
<StructuralDomainFamily>PRION PROTEINS</StructuralDomainFamily>
<Sheet>
<SheetID>S1</SheetID>
<NumberStrands>2</NumberStrands>
<Strands>
<StrandNumber>2</StrandNumber>
<StrandChain>NULL</StrandChain>
<StrandIntialResidue>VAL</StrandIntialResidue>
<StrandIntialResidueSeqNum>161</StrandIntialResidueSeqNum>
<StrandEndResidue>ARG</StrandEndResidue>
<StrandEndResidueSeqNum>164</StrandEndResidueSeqNum>
<StrandSense>ANTI-PARALLEL</StrandSense>
</Strands>
</Sheet>
</Sheets>
```

5.5.7 FunctionalDomains Concept

PO has the first Functional Domain Classification Model defined using FunctionalDomains Concept using: (1) data about cellular and organism source in SourceCell subconcept, (2) data about biological functions of protein in BiologicalFunction subconcept, and (3) data about active binding sites in proteins in ActiveBindingSites subconcept. Like StructuralDomains Concept, SuperFamily and Family subconcepts of generic concept Family are used for identifying the protein family here. SourceCell specifies a biological or chemical source of each biological

molecule (defined by Molecule Concept earlier) in the protein. Biological functions of the protein complex are described in BiologicalFunction. BiologicalFunction has two subconcepts, PhysiologicalFunctions and PathologicalFunctions, and each of these has several subconcepts and sub-subconcepts describing various corresponding functions. The third subconcept of FunctionalDomains is ActiveBindingSites which has details about active binding sites in the protein. Active binding sites are represented in our ontology as a collection of various site groups, defined in SiteGroup Concept. SiteGroup has details about each of the residues and chains that form the binding site. There can be a maximum of seven site groups defined for a protein complex in PO. A typical instance of SourceCell in FunctionalDomains is:

```
<SourceCell>
<ProteinOntologyID>PO0000000009</ProteinOntologyID>
<SourceMoleculeID>1</SourceMoleculeID>
<OrganismScientific>HOMO SAPIENS</OrganismScientific>
<OrganismCommon>HUMAN</OrganismCommon>
<ExpressionSystem>ESCHERICHIA COLI; BACTERIA</ExpressionSystem>
<ExpressionSystemVector>PLAMID</ExpressionSystemVector>
<Plasmid>PRSETB</Plasmid>
</SourceCell>
```

5.5.8 ChemicalBonds Concept

Various chemical bonds used to bind various substructures in a complex protein structure are defined in ChemicalBonds Concept. Chemical bonds that are defined in PO by their respective subconcepts are: DisulphideBond, CISPeptide, HydrogenBond, ResidueLink, and SaltBridge. These are defined using generic concepts of Bind and AtomicBind. The chemical bonds that have binding residues (DisulphideBond, CISPeptide) reuse the generic concept of Bind. In defining the generic concept of Bind in protein ontology we again reuse the generic concepts of Chains and Residues. Similarly, the chemical bonds that have binding atoms (HydrogenBond, ResidueLink, and SaltBridge) reuse the generic concept of AtomicBind. In defining the generic concept of AtomicBind we reuse the generic concepts of Chains, Residues, and Atoms. A typical instance of a ChemicalBond is:

```
<CISPeptides>
<ProteinOntologyID>PO0000000003</ProteinOntologyID>
<BindChain1>H</BindChain1>
<BindResidue1>GLU</BindResidue1>
<BindResSeqNum1>145</BindResSeqNum1>
<BindChain2>H</BindChain2>
<BindResidue2>PRO</BindResidue2>
<BindResSeqNum2>146</BindResSeqNum2>
<AngleMeasure>-6.61</AngleMeasure>
<Model>0</Model>
</CISPeptides>
```

5.5.9 Constraints Concept

Various constraints that affect final protein conformation are defined in Constraints Concept using ConstraintID and ConstraintDescription. The constraints currently described in PO are as follows: (1) monogenetic and polygenetic defects present in

genes that are present in molecules making proteins in the GeneDefects subconcept, (2) hydrophobicity properties in the Hydrophobicity concept, and (3) modification in residue sequences due to chemical environment and mutations in the ModifiedResidue concept. Posttranslational residue modifications are comprised of those amino acids that are chemically changed in such way that they could not be restored by physiological processes, as well as other rare amino acids that are translationally incorporated but for historical reasons are represented as modified residues. The RESID Database [41] is the most comprehensive collection of annotations and structures for protein modifications. The current version of RESID maps posttranslational modifications to both PIR and Swiss-Prot. Data in the GeneDefects class is entered as instances of GeneDefects class and is normally taken from OMIM [37] or scientific literature. A typical instance of a Constraint is:

```
<Constraints>
<ProteinOntologyID>PO00000000001</ProteinOntologyID>
<ConstraintID> 3 </ConstraintID>
<ConstraintDescription> MODIFICATION OF RESIDUES DUE TO
GLYCOSYLATION</ConstraintDescription>
</Constraints>
```

The complete class hierarchy of PO is shown in Figure 5.1. More details about PO are available at the Web site: http://www.proteinontology.info/.

5.5.10 Comparison with Protein Annotation Frameworks

In this section we compare the frameworks of our PO with PRONTO and PDBML.

5.5.10.1 PRONTO and PO

Machine-generated protein ontology generated by PRONTO is just a set of terms and relationships between those terms. PRONTO-generated ontology does not cover and map all the stages of the proteomics process from protein's primary structure to protein's quaternary structure. PRONTO uses iProLink literature-mining ontology to search and identify protein names in the MEDLINE database of biological literature. It then cross-references EBI's UniProt database to define relationships between these terms. PO, on the other hand, integrates data representation frameworks of various protein data sources—PDB, SCOP, RESID, and OMIM—to provide a unified vocabulary covering all the stages of proteomics process. PRONTO represents only two relationships between the terms of the ontology: is-a relation and part-of relation. Whereas PO represents five different relationships between the terms used in the ontology definition. They are: *SubConceptOf, PartOf, AttributeOf, InstanceOf,* and *ValueOf*.

5.5.10.2 PDBML and PO

PDBML is a XML Schema mapping the PDB Exchange Dictionary. In 2004, we did quite similar work [15–17] to PDBML by creating a XML Schema and RDF Schema mapping of PDB, Swiss-Prot, and PIR databases. PDBML lacks the hierarchical relationships as it is linked to the logical representation of PDB. The semantics of

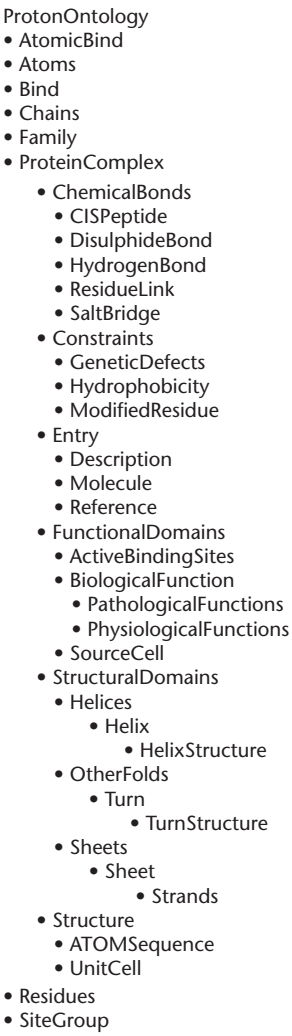


Figure 5.1 Concept hierarchy of protein ontology.

data is preserved and translation from PDB to XML Schema is simple, but it cannot be used to process the content. PO with the power of OWL has no limitations in processing the content.

5.6 Protein Ontology Instance Store

The Protein Ontology Instance Store is created for entering existing protein data using the PO format. PO provides a technical and scientific infrastructure to allow evidence-based description and analysis of relationships between proteins. PO uses data sources including new proteome information resources like PDB, SCOP, and RESID, as well as classical sources of information where information is maintained

in a knowledge base of scientific text files like OMIM and various published scientific literature in various journals. The PO Instance Store is represented using OWL. The PO Instance Store at the moment contains data instances of following protein families: (1) Prion Proteins, (2) B.Subtilis, and (3) Chlorides. More protein data instances will be added as PO becomes more developed. All the PO instances are available for download (http://proteinontology.info/proteins.htm) in OWL format which can be read by any popular editor like Protégé (http://protege.stanford.edu/).

5.7 Strengths and Limitations of Protein Ontology

PO provides a unified vocabulary for capturing declarative knowledge about protein domain and for classifying that knowledge to allow reasoning. Information captured by PO is classified in a rich hierarchy of concepts and their interrelationships. PO is compositional and dynamic, relying on notions of classification, reasoning, consistency, retrieval, and querying. In PO the notions classification, reasoning, and consistency are applied by defining new concepts from defined generic concepts. The concepts derived from generic concepts are placed precisely into the concept hierarchy of PO to completely represent information that defines a protein complex.

As the OWL representation used in PO is an XML-Abbrev based (i.e., Abbreviated XML Notation), it can be easily transformed to the corresponding RDF and XML formats without much effort using the available converters. The PO Instance Store currently covers various species of proteins from bacterial and plant proteins to human proteins. Such a generic representation using PO shows the strength of PO format representation.

We will provide a specific set of rules to cover these application-specific semantics over the PO framework. The rules use only the relationships whose semantics are predefined to establish correspondence among terms in PO. These rules will help in defining semantic query algebra for PO to efficiently reason and query the underlying instance store.

For protein functional classification, in addition to the presence of domains, motifs or functional residues, the following factors are relevant: (1) similarity of three-dimensional protein structures, (2) proximity to genes (this may indicate that the proteins they produce are involved in same pathway), (3) metabolic functions of organisms, and (4) evolutionary history of the protein. At the moment, PO's functional domain classification does not address the issues of proximity of genes and evolutionary history of proteins. These factors will be added in the future to complete the functional domain classification system in PO. Also, the constraints defined in PO are not mapped back to the protein sequence, structure, and function they affect. Achieving this in the future will interlink all the concepts of PO.

The limitations of PO in terms of defining new concepts for protein functions and constraints on protein structure do not limit the use of generalized concepts in PO to define any kind of complex concept for proteomics research in the future.

5.8 Summary

Our protein ontology is the first ever work to integrate protein data based on data semantics describing various phases of protein structure. PO helps to understand structure, cellular function, and the constraints that affect protein in a cellular environment. The attribute values in the PO are not defined as text strings or as set of keywords. Most of the values are entered as instances of generic concepts defined in PO which provide notions of classification, reasoning, and consistency when defining new concepts.

References

[1] Apweiler, R., et al., "UniProt: The Universal Protein Knowledgebase," *Nucleic Acids Res.*, Vol. 32, 2004, pp. 115–119.

[2] Apweiler, R., et al., "Protein Sequence Annotation in the Genome Era: The Annotation Concept of SWISS-PROT + TREMBL," *5th International Conference on Intelligent Systems for Molecular Biology (ISMB)*, Halkidiki, 1997.

[3] Boeckmann, B., et al., "The SWISS-PROT Protein Knowledgebase and Its Supplement TrEMBL in 2003," *Nucleic Acids Res.*, Vol. 31, 2003, pp. 365–370.

[4] Berman, H., et al., "The Protein Data Bank and the Challenge of Structural Genomics," *Nature Structural Biology*, Structural Genomics Supplement, 2000, pp. 957–959.

[5] Bhat, T. N., et al., "The PDB Data Uniformity Project," *Nucleic Acids Res.*, Vol. 29, 2001, pp. 214–218.

[6] Weissig, H., and P. E. Bourne, "Protein Structure Resources," *Biological Crystallography*, Vol. D58, 2002, pp. 908–915.

[7] Westbrook, J., et al., "The Protein Data Bank: Unifying the Archive," *Nucleic Acids Res.*, Vol. 30, 2002, pp. 245–248.

[8] Mitra, P., and G. Wiederhold, "An Algebra for Semantic Interoperatibility of Information Sources," *Infolab*, Stanford University, Stanford, CA, 2001.

[9] Sidhu, A. S., T. S. Dillon, and E. Chang, "Ontological Foundation for Protein Data Models," *1st IFIP WG 2.12 & WG 12.4 International Workshop on Web Semantics (SWWS 2005)*, in conjunction with On The Move Federated Conferences (OTM 2005), Agia Napa, Cyprus, 2005.

[10] Sidhu, A. S., T. S. Dillon, and E. Chang, "An Ontology for Protein Data Models," *27th Annual International Conference of the IEEE Engineering in Medicine and Biology Society 2005 (IEEE EMBC 2005)*, Shanghai, China, 2005.

[11] Sidhu, A. S., T. S. Dillon, and E. Chang, "Advances in Protein Ontology Project," *19th IEEE International Symposium on Computer-Based Medical Systems (CBMS 2006)*, Salt Lake City, UT, 2006.

[12] Sidhu, A. S., T. S. Dillon, and E. Chang, "Integration of Protein Data Sources Through PO," *17th International Conference on Database and Expert Systems Applications (DEXA 2006)*, Poland, 2006.

[13] Sidhu, A. S., T. S. Dillon, and E. Chang, "Towards Semantic Interoperability of Protein Data Sources," *2nd IFIP WG 2.12 & WG 12.4 International Workshop on Web Semantics (SWWS 2006)* in conjunction with OTM 2006, France, 2006.

[14] Sidhu, A. S., et al., "Protein Ontology: Vocabulary for Protein Data," *3rd International IEEE Conference on Information Technology and Applications, 2005 (IEEE ICITA 2005)*, Sydney, 2005.

[15] Sidhu, A. S., et al., "Comprehensive Protein Database Representation," *8th International Conference on Research in Computational Molecular Biology 2004 (RECOMB 2004)*, San Diego, CA, 2004.

[16] Sidhu, A. S., et al., "A Unified Representation of Protein Structure Databases," in *Biotechnological Approaches for Sustainable Development*, M. S. Reddy and S. Khanna, (eds.), India: Allied Publishers, 2004, pp. 396–408.

[17] Sidhu, A. S., et al., "An XML Based Semantic Protein Map," *5th International Conference on Data Mining, Text Mining and their Business Applications (Data Mining 2004)*, Malaga, Spain, 2004.

[18] Ashburner, M., et al., "Creating the Gene Ontology Resource: Design and Implementation," *Genome Research*, Vol. 11, 2001, pp. 1425–1433.

[19] Lewis, S. E., "Gene Ontology: Looking Backwards and Forwards," *Genome Biology*, Vol. 6, 2004, pp. 103.1–103.4.

[20] Altmann, R. B., et al., "RiboWeb: An Ontology-Based System for Collaborative Molecular Biology," *IEEE Intelligent Systems*, 1999, pp. 68–76.

[21] Westbrook, J., et al., "PDBML: The Representation of Archival Macromolecular Structure Data in XML," *Bioinformatics*, Vol. 21, 2005, pp. 988–992.

[22] Mani, I., et al., "PRONTO: A Large-Scale Machine-Induced Protein Ontology," *2nd Standards and Ontologies for Functional Genomics Conference (SOFG 2004)*, Philadelphia, PA, 2004.

[23] Berman, H., K. Henrick, and H. Nakamura, "Announcing the Worldwide Protein Data Bank," *Nature Structural Biology*, Vol. 12, 2003, p. 980.

[24] Wu, C. H., et al., "The Protein Information Resource," *Nucleic Acids Res.*, Vol. 31, 2003, pp. 345–347.

[25] Sasson, O., et al., "ProtoNet: Hierarchical Classification of the Protein Space," *Nucleic Acids Res.*, Vol. 31, 2003, pp. 348–352.

[26] Bateman, A., et al., "The Pfam Protein Families Database," *Nucleic Acids Res.*, Vol. 30, 2002, pp. 276–280.

[27] Corpet, F., et al., "ProDom and ProDom-CG: Tools for Protein Domain Analysis and Whole Genome Comparisons," *Nucleic Acids Res.*, Vol. 28, 2000, pp. 267–269.

[28] Falquet, L., et al., "The Prosite Database, Its Status in 2002," *Nucleic Acids Res.*, Vol. 30, 2002, pp. 235–238.

[29] Attwood, T. K., et al., "PRINTS and Its Automatic Supplement, prePRINTS," *Nucleic Acids Res.*, Vol. 31, 2003, pp. 400–402.

[30] Lo Conte, L., et al., "SCOP Database in 2002: Refinements Accommodate Structural Genomics," *Nucleic Acids Res.*, Vol. 30, 2002, pp. 264–267.

[31] Pearl, F. M. G., et al., "The CATH Database: An Extended Protein Family Resource for Structural and Functional Genomics," *Nucleic Acids Res.*, Vol. 31, 2003, pp. 452–455.

[32] Huang, H., et al., "iProClass: An Integrated Database of Protein Family, Function and Structure Information," *Nucleic Acids Res.*, Vol. 31, 2003, pp. 390–392.

[33] Mulder, N. J., et al., "The InterPro Database, 2003 Brings Increased Coverage and New Features," *Nucleic Acids Res.*, Vol. 31, 2003, pp. 315–318.

[34] Letunic, I., et al., "Recent Advancements to the SMART Domain-Based Sequence Annotation Resource," *Nucleic Acids Res.*, Vol. 30, 2002, pp. 242–244.

[35] Haft, D., et al., "TIGRFAMs: A Protein Family Resource for Functional Identification of Proteins," *Nucleic Acids Res.*, Vol. 29, 2001, pp. 41–43.

[36] Dayhoff, M. O., "The Origin and Evolution of Protein Superfamilies," *Fed. Proc.*, Vol. 35, 1976, pp. 2132–2138.

[37] McKusick, V. A., *Mendelian Inheritance in Man: A Catalog of Human Genes and Genetic Disorders*, 12th ed., Baltimore, MD: Johns Hopkins University Press, 1998.

[38] Frazier, M. E., et al., "Realizing the Potential of Genome Revolution: The Genomes to Life Program," *Science*, Vol. 300, 2003, pp. 290–293.

[39] Frazier, M. E., et al., "Setting Up the Pace of Discovery: The Genomes to Life Program," *2nd IEEE Computer Society Bioinformatics Conference (CSB 2003)*, Stanford, CA, 2003.

[40] Collins, F. S., M. Morgan, and A. Patrinos, "The Human Genome Project: Lessons from Large-Scale Biology," *Science*, Vol. 300, 2003, pp. 286–290.

[41] Garavelli, J. S., "The RESID Database of Protein Modifications: 2003 Developments," *Nucleic Acids Res.*, Vol. 31, 2003, pp. 499–501.

CHAPTER 6

Information Quality Management Challenges for High-Throughput Data

Cornelia Hedeler and Paolo Missier

In postgenomic biology, high-throughput analysis techniques allow a large number of genes and gene products to be studied simultaneously. These techniques are embedded in experimental pipelines that produce high volumes of data at various stages. Ultimately, the biological interpretation derived from the data analysis yields publishable results. Their quality, however, is routinely affected by the number and complexity of biological and technical variations within the experiments, both of which are difficult to control.

In this chapter we present an analysis of some of these issues, conducted through a survey of quality control techniques within the specific fields of transcriptomics and proteomics. Our analysis suggests that, despite their differences, a common structure and a common set of problems for the two classes of experiments can be found, and we propose a framework for their classification. We argue that the scientists' ability to make informed decisions regarding the quality of published data relies on the availability of metainformation describing the experiment variables, as well as on the standardization of its content and structure. Information management expertise can play a major role in the effort to model, collect, and exploit the necessary metainformation.

6.1 Motivation

With several genomes of model organisms now being fully sequenced and with the advent of high-throughput experimental techniques, research in biology is shifting away from the study of individual genes, and towards understanding complex systems as a whole, an area of study called *systems biology* [1]. Instead of studying one gene or protein at a time, a large number of genes or proteins are monitored simultaneously. Different kinds of experimental data are integrated and analyzed to draw biological conclusions, state new hypotheses, and ultimately generate mathematical models of the biological systems.

A single high-throughput experiment may generate thousands of measurements, requiring the use of data-intensive analysis tools to draw biologically significant conclusions from the data. The data and its biological interpretation are then

disseminated through public repositories and journal publications. Once published, this can be used within the scientific community to annotate gene and protein descriptions in public databases, and to provide input to so-called in silico experiments—that is, "procedures that use computer-based information repositories and computational analysis tools to test a hypothesis, derive a summary, search for patterns, or demonstrate a known fact" [2].

In the recent past, research into the quality of information available in public biology databases has focused mainly on the issue of data reconciliation across multiple and heterogeneous data sources [3, 4]. In this area, it has been possible to adapt techniques and algorithms for which a largely domain-independent theoretical framework exists, notably for record linkage [5] and for data integration in the presence of inconsistencies and incompleteness [6, 7].

Data reconciliation techniques, however, largely fail to address the basic problem of establishing the *reliability* of experimental results submitted to a repository, regardless of their relationships with other public data. This is a fundamental and pervasive information quality problem[1]: using unproven or misleading experimental results for the purpose of database annotation, or as input to further experiments, may result in the wrong scientific conclusions. As we will try to clarify in this chapter, techniques and, most importantly, appropriate metadata for objective quality assessment are generally not available to scientists, who can be only intuitively aware of the impact of poor quality data on their own experiments. They are therefore faced with apparently simple questions: Are the data and their biological implications credible? Are the experimental results sound, reproducible, and can they be used with confidence?

This survey offers an insight into these questions, by providing an introductory guide for information management practitioners and researchers, into the complex domain of post-genomic data. Specifically, we focus on data from transcriptomics and proteomics (i.e., the large-scale study of gene[2] and protein expression), which represent two of the most important experimental areas of the post-genomic era.

We argue that answering the scientists' questions requires a thorough understanding of the processes that produce the data and of the quality control measures taken at each step in the process. This is not a new idea: a generally accepted assumption in the information quality community [9, 10] has been to consider information as a product, created by a recognizable production process, with the implication that techniques for quality control used in manufacturing could be adapted for use with data. These ideas have been embedded into guidelines for process analysis that attempt to find metrics for measuring data quality [11, 12].

While we subscribe to this general idea, we observe that an important distinction should be made between business data, to which these methodologies have been applied for the most part (with some exceptions; see, for example, [13] for an analysis of data quality problems in genome data), and experimental scientific data. Business data is often created in a few predictable ways—that is, with human input or

1. The term "information" is often used in contrast with "data," to underline the difference between the ability to establish formal correctness of a data item, and the ability to provide a correct interpretation for it. In this sense, assessing reliability is clearly a problem of correct interpretation, hence of *information quality*.
2. The term "gene expression" refers to the process of DNA transcription for protein production within a cell. For a general introduction to the topics of genomic and proteomics, see [8].

input from other processes (this is the case, for example, in banking, public sector, and so on), and it has a simple interpretation (addresses, accounting information). Therefore, traditional data quality problems such as stale data, or inconsistencies among copies, can often be traced to problems with the input channels and with data management processes within the systems, and software engineering techniques are usually applied to address them.

The correct interpretation of scientific data, on the other hand, requires a precise understanding of the broad variability of the experimental processes that produce it. With research data in particular, the processes are themselves experimental and tend to change rapidly over time to track technology advances. Furthermore, existing quality control techniques are very focused on the specific data and processes, and are difficult to generalize; hence the wealth of domain-specific literature offered in this survey.

This variability and complexity makes the analysis of quality properties for scientific data different and challenging. In this domain, traditional data quality issues such as completeness, consistency, and currency are typically observed at the end of the experiment, when the final data interpretation is made. However, as the literature cited in this chapter shows, there is a perception within the scientific community that quality problems must be addressed at all the stages of an experiment.

For these reasons, we focus on the data creation processes, rather than on the maintenance of the final data output. We concentrate on two classes of experiments: microarray data analysis for transcriptomics, and protein identification for proteomics. In these areas, the quality of the data at the dissemination stage is determined by factors such as the intrinsic variability of the experimental processes, both biological and technical, and by the choice of bioinformatics algorithms for data analysis; these are often based on statistical models, and their performance is in turn affected by experimental variability, among other factors. A brief background on these technologies is provided in Section 6.2.

As a matter of method, we observe that these two classes can be described using the same basic sequence of steps, and that the corresponding quality problems also fall into a small number of categories. We use the resulting framework to structure a list of domain-specific problems, and to provide references for the techniques used to tackle them. This analysis is presented in Section 6.3.

Although the quality control techniques surveyed are rooted in the context of post-genomics, and this survey does not discuss specific techniques or solutions in depth, a few general points emerge from this analysis regarding technical approaches to quality management: (1) the importance of standards for accurately modeling and capturing *provenance metadata* regarding the experiments (i.e., details of the experimental design and of its execution), and (2) the standardization of their representation, in order to deal with heterogeneity among different laboratories that adopt different experimental practices. These points are discussed in Section 6.4.

A further potentially promising contribution offered by the information quality community is the study of information as a product, already mentioned [9, 10]. However, to the best of our knowledge, no general theory of process control for these data domains has been developed.

6.2 The Experimental Context

We describe the general steps of a rather generic biological experiment, starting from the experimental design, leading to a publication, and further, to the use of published literature for the functional annotation of genes and proteins in databases. An overview of this abstraction is shown in Figure 6.1.

The experiment begins with the statement of a scientific hypothesis to be tested; along with constraints imposed by the laboratory equipment, this leads to the choice of an appropriate experimental design. The so-called *wet lab* portion is executed by the biologist, starting from the preparation of the sample, and usually leading to the generation of some form of raw data.

It is common to build elements of repetition into the experiment, to take into account both technical and biological variability, specifically:

1. *Technical repeats:* After preparation, the sample is divided into two or more portions and each portion is run through exactly the same technical steps, leading to separate measurements for each portion. This is done to account for the variability of the technical process.
2. *Biological repeat:* Two or more samples are obtained from different individuals studied under exactly the same conditions. These samples are then prepared using the same protocol and run through the same technical process. These repeats allow for the estimation of biological variability between individuals.

The raw data generated in the lab is then analyzed in the so-called *dry lab*, a computing environment equipped with a suite of bioinformatics data analysis tools. The processed data is then interpreted in the light of biological knowledge, and scientific claims can be published. A growing number of scientific journals explicitly require that the experimental data be submitted to public data repositories at the same time [14].

The result of data analysis and interpretation is processed data, which can include not only the experimental data in analyzed form, but also additional information that has been used to place the data into context, such as the functional annotation of genes and proteins or pathways the proteins are involved in.

The repetition of this process for a large number of high-throughput experiments and over a period of time results in a body of literature about a particular gene or protein. This knowledge is used by *curators* as evidence to support the annotation of genes and proteins described in public databases, such as MIPS [15] and Swiss-Prot [16]. A protein annotation typically includes a description of its function, a description of the biological processes in which it participates, its location in the cell, and its interactions with other proteins.

Reaching conclusions regarding protein function requires the analysis of multiple pieces of evidence, the results of many experiments of different natures, and may involve a combination of manual and automated steps [17]. In this chapter, we concentrate on two classes of experiments, microarray analysis of gene expression and protein identification; they share the general structure outlined above, and are relevant for their contribution to the knowledge used by the curation process.

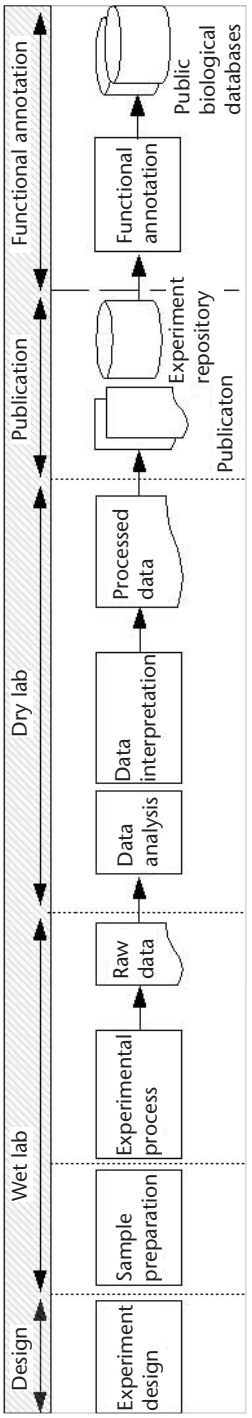


Figure 6.1 Sample high-throughput biological data processing pipeline.

We now briefly review some definitions regarding these experiments.

6.2.1 Transcriptomics

Transcriptome experiments use microarray technology to measure the level of transcription of a large number of genes (up to all genes in a genome) simultaneously, as an organism responds to the environment. They measure the quantity of mRNA produced in response to some environmental factor, for instance some treatment, at a certain point in time, by obtaining a snapshot of the gene activity at that time.[3]

Here we only provide a brief introduction to the experimental steps involved and the data analysis. For recent reviews on this topic, see [18–20]. In addition, [21–23] provide reviews of methods to normalize and analyze transcriptome data.

An array is a matrix of spots, each populated during manufacturing with known DNA strands, corresponding to the genes of interest for the experiment. When a sample consisting of mRNA molecules from the cells under investigation is deposited onto the array, these molecules bind, or *hybridize*, to the specific DNA templates from which they originated. Thus, by looking at the hybridized array, the quantity of each different mRNA molecule of interest that was involved in the transcription activity can be measured.

Among the many different array technologies that have become available within the last few years, we focus on the two that are most common: cDNA [24] and oligonucleotide arrays [25]. The choice between the two is dictated by the available equipment, expertise, and by the type of experiment. An oligonucleotide array accepts one sample and is suitable for measuring absolute expression values. Whereas cDNA arrays accept two samples, labeled using two different fluorescent dyes, which may represent the state of the organism before and after treatment; they are used to measure ratios of expression levels between the two samples. To obtain ratios using oligonucleotide technology, two arrays are necessary, and the ratios are computed from the separate measurements of each. This difference is significant, because the technical and biological variability of these experiments plays a role in the interpretation of the results.

The measurements are obtained by scanning the arrays into a digital image, which represents the raw data from the wet lab portion of the experiment. In the dry lab, the image is analyzed to identify poor quality spots, which are excluded from further analysis, and to convert each remaining spot into an intensity value (the "raw readings" in Figure 6.2). These values are normalized to correct for background intensity, variability introduced in the experiment, and also to enable a comparison between repeats.

In the subsequent high-level data analysis, the normalized data is interpreted in the light of the hypothesis stated and the biological knowledge, to draw publishable conclusions. Typically, the goal of the analysis is to detect genes that are differentially expressed after stimulation, or to observe the evolution of expression levels in time, or the clustering of genes with similar expression patterns over a range of conditions and over time. Statistical and machine learning approaches are applied in this phase [22, 26, 27].

3. For a tutorial on microarrays, see http://www.ncbi.nlm.nih.gov/About/primer/ microarrays.html.

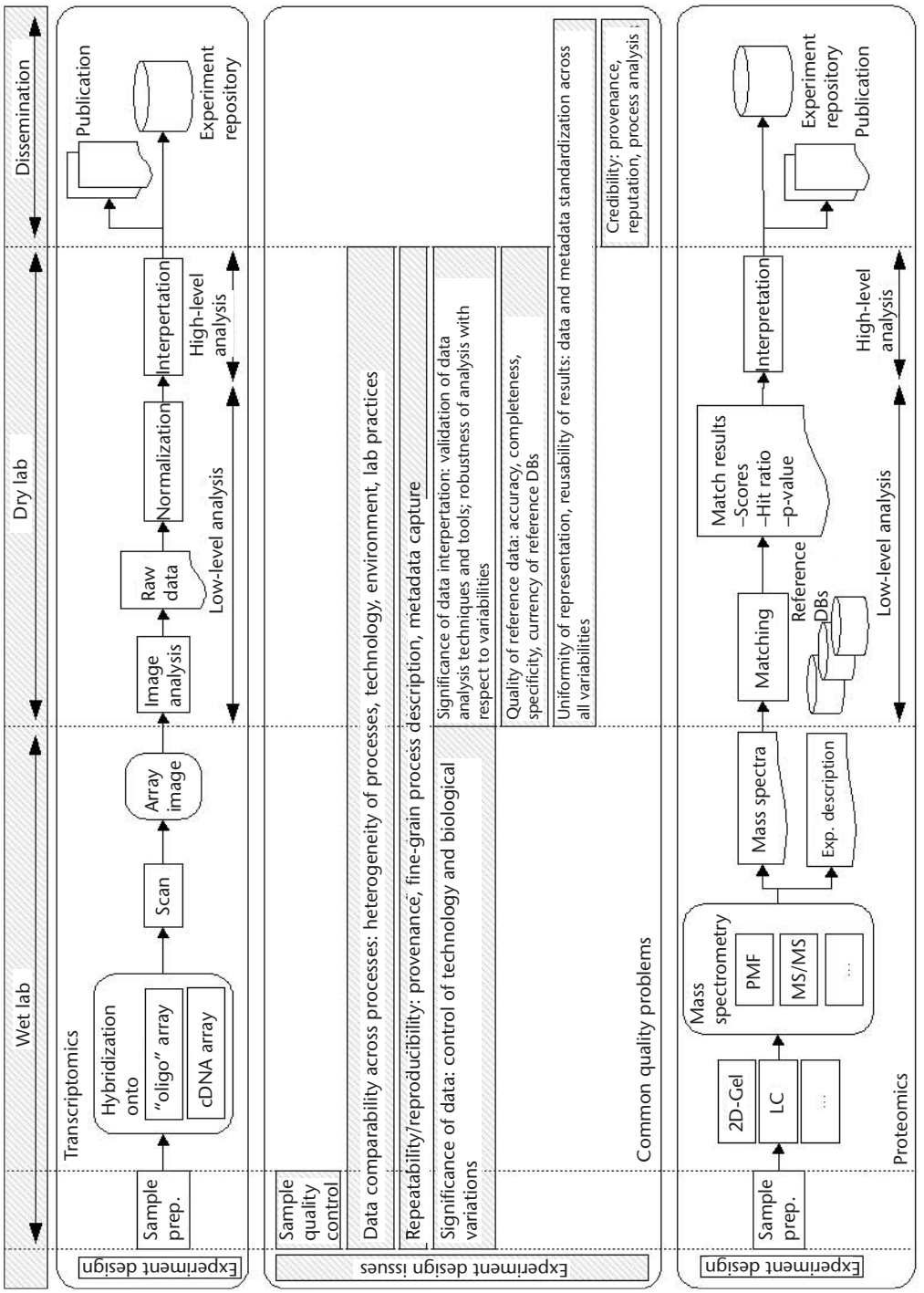


Figure 6.2 High-level biological experiment pipeline and common quality issues.

Each of these process steps involves choices that must be made (e.g., of technology, of experiment design, and of low-level and high-level data analysis algorithms and tools), which are interdependent and collectively affect the significance of the final result. We survey some of these factors in the next section.

6.2.2 Qualitative Proteomics

The term "proteomics" refers to large-scale analysis of proteins, its ultimate goal being to determine protein functions, and includes a number of areas of investigation. Here we only consider the problem of identifying the proteins within a sample, a problem of *qualitative* proteomics; this involves determining the peptide masses and sequences of the proteins present in a sample, and matching those against theoretically derived peptides calculated from protein sequence databases. For in-depth reviews of the field, see [28–30].

This technology is suitable for experiments in which the protein contents before and after a certain treatment are compared, ultimately leading to conclusions regarding their function, the biological processes in which they are involved, and their interactions. The main steps of the experimental process are shown at the bottom part of Figure 6.2.

A sample containing a number of proteins (possibly of the order of thousands) undergoes a process of separation, commonly by two-dimensional electrophoresis (2DE), resulting in the separation of the proteins onto a gel based on two orthogonal parameters: their charge and their mass. The separated proteins spotted on the gel are then excised and degraded enzymatically to peptides. An alternative technique for peptide separation involves liquid chromatography (LC) (see [31, 32] for reviews). LC is used to purify and separate peptides in complex peptide mixtures and can be used without the extra step of protein separation on a gel before digestion [29]. Peptides are separated by their size, charge, and hydrophobicity.

To identify the proteins, mass spectrometry (MS) is used to measure the mass-to-charge ratio of the ionized peptides. The spectrometer produces mass spectra (i.e., histograms of intensity versus mass-to-charge ratio). For single-stage experiments, these are called peptide mass fingerprints (PMF). Additionally, a selection of these peptides can be further fragmented to perform tandem MS, or MS/MS experiments, which generate spectra for individual peptides. From these spectra the sequence tag of the peptide can be derived. Using sequence information of several peptides in addition to their masses is more specific for the protein identification than just the masses.

The key parameters for this technology are sensitivity, resolution, and the ability to generate information-rich mass spectra. The issue of resolution arises when one considers that every cell may express over 10,000 genes, and that the dynamic range of abundance in complex samples can be as high as 10^6. Since 2DE technology can resolve no more than 1,000 proteins, clearly only the most abundant proteins can be identified, which creates a problem when interesting proteins are much less abundant [29]. Techniques have been developed to deal with these issues [33]; in general, however, limitations in the technology translate into inaccuracies in the resulting spectra.

Finally, in the dry lab the mass spectra are compared with masses and sequences of peptides in databases. Here the experimenter is confronted with more choices: a

number of different algorithms (e.g., Mascot [34], SEQUEST [35]) exist to compute a score for the goodness of the match between the theoretical peptide sequences in the database and the experimental data. Also, these algorithms may be applied to different reference databases, and provide different indicators to assess the quality of the match. Examples of indicators are the hit ratio (the number of peptide masses matched, divided by the total number of peptide masses submitted to the search), and the sequence coverage (the percentage of the number of amino acids in the experimental sequence, to those in the theoretical sequence).

The quality of the scoring functions in particular is affected by experimental variability, and statistical and computational methods have been proposed to deal with the uncertainty of the identification process (see [36] for a review).

6.3 A Survey of Quality Issues

We begin our analysis by presenting a common framework, illustrated in Figure 6.2. At the top and the bottom are the main steps of the protein identification and of the microarray experiments, respectively. The figure shows their common structure in terms of the general wet lab, dry lab, and dissemination steps, and highlights the key quality concerns addressed by experimenters at each step. We use this high-level framework to provide structure for the analysis of domain-specific issues, and the current techniques and practices adopted to address them.

In Section 6.3.4, a separate discussion is devoted to the problem of annotating information after it has been submitted to proteomic databases; this has only been addressed in the past by relatively few and isolated experiments.

6.3.1 Variability and Experimental Design

Both transcriptome and proteome experiments consist of a number of steps, each of which can introduce factors of variability. However, it is not only the variability introduced in the experimental process (the so-called technical variability) that can affect the quality of the results, but also biological variability. Systematic analyzes of variability in transcriptome studies [37, 38] and proteome studies [39] have shown that biological variability may have a greater impact on the result.

6.3.1.1 Biological Variability

This form of variability affects the results of both transcriptome and proteome experiments; it is of rather random nature and is hard to estimate. Examples include:

1. Variability between individuals studied under the same experimental condition [37, 40] due to genetic differences [39], minor infections resulting in inflammatory and immune responses of varying intensities [38], environmental stress, or different activity levels, but can also be due to tissue heterogeneity (varying distribution of distinct cell types in tissues);

2. Variability between individuals due to random differences in the experimental conditions, such as growth, culture, or housing conditions [39–41];
3. Variability within individuals and within the same tissue due to tissue heterogeneity [23, 37];
5. Variability within individuals in different tissues or cell types [41]—in this case the differences are more distinct than within the same tissue.

These variabilities can obscure the variation induced by the stimulation of the organism [40], leading to results that are meaningless in the context of the stated hypothesis. Biological variability can be addressed in part at the early stages by proper experimental design, for example by a sufficient number of biological repeats [21, 38, 40] that can be used to average over the data and validate the conclusions over a range of biological samples. For further validation of the results, they should be confirmed using alternative experimental techniques on a number of biological samples [40].

6.3.1.2 Technical Variability

This usually represents some kind of systematic error or bias introduced in the experimental process and once known can be corrected for in the data analysis, such as normalization. It can also be reduced by appropriate experimental design. Examples are mentioned in Table 6.1 [42–45]. To reduce technical variability, experimental protocols that result in reproducible, reliable results can be identified and then followed meticulously [40]. Dye swap cDNA microarray experiments, in which the labeling dye of the samples is reversed in the repeat [23, 43], are used to account for

Table 6.1 Examples of Technical Variability Introduced in Transcriptomics and Proteomics

Wet Lab		*Dry Lab*
Sample preparation	*Experimental process*	*Data analysis*
Variation in sample collection and preparation	Variation in experimental data collection processes	Variation in data processing and preparation analysis
RNA extraction and labeling [21, 37, 40–42]. Variability in the sample preparation can result in change of the gene expression profile. Sample contamination [23]. Dye-based bias (i.e., one dye might be "brighter" than the other dye) [23, 43].	Variation in hybridization process [21, 38, 40–42]. Variations introduced in the process can obscure changes caused by the stimulation of the organism (i.e., changes that the experiment actually seeks to determine).	Different data processing approaches [41, 42]. The wide range of available analysis approaches make it hard to assess the performance of each of them and to compare the results of experiments carried out in different labs.
Variability in sample preparation and processing for LC/MS/MS can lead to differences in the number of low intensity peaks measured [44]. This can result in the identification of fewer peptides and proteins.	Variability in tandem mass spectra collection (LC/MS/MS) [44, 45]. Variability introduced here can lead to errors in search algorithms and ultimately to false positives in peptide identification. Quantitative variation between matched spots in two 2D-gels and fewer spots that can be matched in repeated gels [39].	Variability in tandem mass spectra processing (LC/MS/MS) [44, 45]. Some of the search algorithms used to identify peptides might be more or less sensitive to the variability introduced during the collection of mass spectra [45], resulting in a different number of identified peptides in the same spectra using different algorithms.

dye-based bias. To estimate the influence of technical variability on results of both transcriptome and proteome experiments, technical repeats can be used [21, 23, 39, 40]. Formulae have been devised to determine the number of repeats and samples by taking into account the effects of pooling, technical replicates, and dye-swaps [46].

6.3.1.3 Experimental Design

Experimental design not only includes the decision about the number of biological and technical replicates, it also includes all the decisions about sample preparation methods, experimental techniques, and data analysis methods. All these decisions should be made to ensure that the data collected in the experiment will provide the information to support or reject the scientific hypothesis. Badly designed experiments might not only not provide the answers to the questions stated, but might also leave potential bias in the data that might compromise the analysis and interpretation of the result [38]. Reviews of experimental design of transcriptome and proteome experiments can be found in [21, 38, 43, 47].

The number of variabilities that affect the outcome of an experiment make it hard to assess the experiment's quality. As we argue in the next section, an accurate record of the experimental design and of the environmental variables involved is a necessary, but hardly sufficient, condition to provide objective indicators that can be used to assess confidence in the experimental results.

6.3.2 Analysis of Quality Issues and Techniques

Results of our survey analysis are presented in Tables 6.2 and 6.3 for transcriptomics and proteomics experiments, respectively. Each group of entries corresponds to one of the general quality concerns from Figure 6.2 (first column in the table); for each group, specific problems are listed in the third column, and in the last column there is a summary of associated current practices and techniques, including examples that illustrate the need to address the issues using those practices and techniques. An additional grouping of these issues by type of artifact produced during the process (second column) is provided where appropriate. For instance, "repeatability and reproducibility" (second group) in Table 6.2 maps to two problems: general adequacy of the process description for future reference, and control of variability factors. For the latter, the issues are sufficiently distinct to suggest grouping them by artifact (hybridized array, raw data, interpretation of normalized data).

These tables, along with the selected references associated to the entries, are designed as a sort of springboard for investigators who are interested in a deeper understanding of the issues discussed in this chapter.

Quality issues that are not addressed in the process of the experiment may result in poor data quality in the form of false positives or false negatives and may lead to incorrect conclusions. Since these high-throughput experiments are frequently used not only to test hypotheses, but also, due to their scale, to generate new hypotheses, these new hypotheses might be wrong and follow-up experimental expenses and time to test these hypotheses may be wasted.

Table 6.2 Quality Issues in Transcriptomics Experiments

Common Quality Issues	*Artifact*	*Specific Issues*	*Examples, Techniques, and References*
Quality of sample	Biological assay	RNA contamination control biological variability	Technical assessment of RNA quality [48]; low quality RNA may compromise results of data analyses
Process repeatability results reproducibility	(General)	Adequate process description	Provenance, metadata capture standards and techniques for fine-grain process description [2, 49]
	Raw data (image)	Biological variability [37] Technical variability: consistancy of image quality control parameter	Review [23, 50] Experimental design [43, 46]
	Normalized data	Significant of interpertation given biological and technical variability	See "significance of data interpertation"
Data comparability	(General)	Reproducibility across platforms, technologies, and labaratories	Methods to accommodate variability across platforms and labs [51, 52] Consistency of results across platforms [53, 54]
Significance of data	(General)	Variability control	Quantification of measurement errors [55]
	Raw data	Image accuracy: interpretation spots and their intensity levels nonuniform hybridization	Image analysis and quality control [23, 50]; bad spot detection, background identification, image noise modeling, manual inspection of spots; poor image quality may require costly manipulations and decrease the power of the analysis
	Normalized data	Choice of normalization algorithms	Review: [23]; choosing an inadequate normalization algorithm may lead to an incomplete removal of systematic errors and affect the power of the downstream analysis; low-level data analysis [21]; statistical error analysis, dye-bias control and reduction [56]; algorithms to control signal-to-noise ratios [57]
Significance of data interpertation	Data interpertation	Validity of data analysis techniques and tools; robustness of analysis with respect to variabilities	Review on design and selection of clustering algorithms: [26, 58]; computational methods to take variabilities into account [37, 41]; algorithm performance analysis by cross-validation [59]; identification of significant differences in gene expression: statstical analysis of replicated experiments [27]; analysis of threshold choice of characterise disease versus normality [23, 60]; use of false discovery rate for generating gene expression scores [61, 62]
Quality of reference data	(General)	Accuracy, completeness, specificity, current of reference databases functional annotations in reference DBs	Mostly based on practitioners' personal perception; systematic studies are needed (see Section 6.3.4)
Uniformity of representation reusability of results	Output data, publication	Heterogeneity of presentation	Data and metadata standardization of content and presentation format [14, 23]

Table 6.3 Quality Issues in Protein Identification Experiments

Common Quality Issues	*Artifact*	*Specific Issues*	*Examples, Techniques, and References*
Quality of sample	Biological assay	Biological variability, contamination control	Sample contamination with, for example, human proteins from the experimenter may obsure the results of downstream analysis
Process repeatability results reproducibility	(General)	Adequate process description	Data modelling for capturing experiment design and execution results [63] Also see uniformity, below
	Raw data (image)	Technical and biological variability	Analysis of reproducibility [64] Quantitative assessment of variability [65]
Data comparability	(General)	Reproducibility across platforms, technologies, and labaratories	Review [66]
Significance of data	(General)	Variability control	Review on statistical and computational issues across all phases of data analysis [67] Review on analysis of sensitivity [68]
	Raw data (mass spectra)	Sensitivity of spectra generation methods, dynamic range for relative protein abundance Technical and biological variability Limitations of technology for generating spectra	Review on strategies to improve accuracy and sensitivity of PI, quantification of relative changes in protein abundance [69] Studies on scoring models, database search algorithms, assessment of spectra quality prior to performing a search, analysis of variables that affect performance of DB search [36] (review) [70] Review on limitations of 2DE technology for low-abundance proteins [33]
Significance of data interpertation	Match results	Significance and accuracy of match results, limitations of technology for accurate identification	Definition [71] and validation of scoring functions Review on limitations of technology: [72] Statistical models [73] Studies on matching algorithms [36] (review), [74]
Quality of reference data	(General)	Redundancy of reference DB (same protein appears under different anmes and accession numbers in databases) Accuracy, completeness specificity, currency of reference databases	Criteria for the selection of appropriate reference DB [64]: using a species-specific reference datatbase will result in more real protein identifications than using a general reference database containing a large number of organisms; using the latter may result in a large number of false positives
Uniformity of representation reusability of results	Output data, publication	Heterogeneity of presentation	Need for represntation standards [75] The PEDRO proteomics data model [64] Guidelines for publication [76] Standards for metadata [66]

6.3.3 Specificity of Techniques and Generality of Dimensions

Most of the techniques mentioned in Tables 6.2 and 6.3, on which we will not elaborate due to space constraints, are specific and difficult to generalize into a reusable "quality toolkit" for this data domain. While this may be frustrating to some quality practitioners in the information systems domain, we can still use some of the common terms for quality dimensions, provided that we give them a correct interpretation. A good example is the definition of "accuracy": in its generality, it is defined as the distance between a data value and the *real* value. When applied to a

record or a field in a relational database table, this definition is specialized by introducing distance functions that measure the similarity between the value in the record and a *reference* value, for instance by computing the edit distance between strings. Further distinctions are made depending on the type of similarity that we seek to measure.

In the experimental sciences, the abstract definition for accuracy is identical (see, for instance, [42]); however, for a value that represents the numeric output of an experimental process, accuracy is interpreted in statistical terms, as a measure of systematic error (e.g., background noise in the experiment). Consequently, techniques for estimating accuracy (i.e., the equivalent of "distance functions") are grounded in the nature of the process, and are aimed at measuring and controlling noise. In [56], for example, a novel statistical model is proposed for the analysis of systematic errors in microarray experiments. Here, errors that lead to low accuracy are detected and corrected by introducing different normalization techniques, whose effectiveness is compared experimentally; different statistical models are applied depending on the specific microarray experiment design used.

Information quality practitioners will probably be on more familiar ground when quality concepts like accuracy, currency, and timeliness are applied to reference databases used in the experiments, for example, for protein-peptide matches, or to the last phase of our reference pipeline, when the differently expressed genes in a transcriptome experiment are functionally annotated. In this case, "accuracy" refers to the likelihood that a functional annotation is correct (i.e., that the description of the function of the gene or gene product corresponds to its *real* function [77].[4] As mentioned, annotations may be done either by human experts, based on publications evidence, or automatically by algorithms that try to infer function from structure and their similarity with that of other known gene products. In the first case, measuring accuracy amounts to supporting or disproving scientific claims made in published literature, while in the second, the predictive performance of an algorithm is measured.

In general, we observe a trade-off between the accuracy of curator-produced functional annotations, which have a low throughput, and the timeliness of the annotation (i.e., how soon the annotation becomes available after the gene product is submitted to a database). A notable example is provided by the Swiss-Prot and TrEMBL protein databases. While in the former, annotations are done by biologists, with great accuracy at the expense of timeliness, TrEMBL contains proteins that are automatically annotated, often with lower accuracy, but are made available sooner [78]. This gives the scientist a choice, based on personal requirements. For well-curated database such as UniProt, claims of nonredundancy (but not of completeness) are also made [79].

6.3.4 Beyond Data Generation: Annotation and Presentation

To conclude this section, we now elaborate further on the topic of functional annotations and their relationship to quality. The aim of annotation is, in general, to "bridge the gap between the sequence and the biology of the organism" [80]. In this

4. Assessing the completeness of an annotation is just as important; however, the intrinsic incompleteness of the biological interpretation of genes and gene products [77] makes this task even more challenging.

endeavor, three main layers of interpretation of the raw data are identified: nucleotide-level (where are the genes in a sequence?), protein-level (what is the function of a protein?), and process-level (what is the role of genes and proteins in the biological process and how do they interact?). The information provided by high-throughput transcriptomics and proteomics contributes to functional and process annotation. Thus, it participates in the cycle shown in Figure 6.3: publications are used by curators to produce functional annotations on protein database entries, which in turn may stimulate the proposal of new experiments (automatic annotations use information from other databases, as well).

Although a bit simplistic, this view is sufficient to identify the critical issue with annotation: erroneous functional annotation based on biased results and conclusions due to unaccounted variabilities in experiments can propagate through public databases and further lead to wrong conclusions.

Most of the studies on the percolation effects of annotation errors have focused on automated annotations, in which protein function is determined computationally, based on sequence similarity to other proteins in the same domain [81–85]. However, the issue of validating curated annotations that are based on published literature is more subtle. One approach is based on the observation that, when standard controlled vocabularies are used for annotation, the consistency of use of the terminology offered by these vocabularies in multiple independent annotations of the same data can be used as an indicator of annotation accuracy.

As an example, consider the Gene Ontology (GO), a well-known standard ontology for describing the function of eukaryotic genes [86]. GO is maintained by a consortium of three model organism databases, and it consists of three parts: molecular function, biological process, and cellular component (the subcellular structures where they are located). Up to the present, GO annotations have been used to annotate almost two million gene products in more than 30 databases. UniProt is the most prominent, accounting for almost 50% of the annotations.

The adoption of such a standard in biology has allowed researchers to investigate issues of annotation consistency. We mention two contributions here. The first [87] has studied measures of *semantic similarity* between Swiss-Prot entries, based on their GO annotations. The authors hypothesize that valid conclusions about protein similarity can be drawn not only based on their sequence similarity (as would be done, for instance, by BLAST), but also from the semantic similarity of the annotations that describe the biological role of the proteins. The latter is described

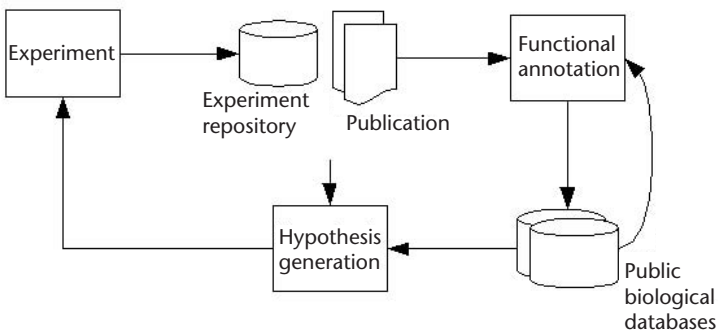


Figure 6.3 The annotation process.

by metric functions defined on the GO structure (GO is a directed acyclic graph). Based on statistical evidence, the authors conclude that the hypothesis is valid for various specific assumptions, for example, that the data set is restricted to those proteins whose annotations are supported by published literature, as opposed to being inferred from some indirect data source.

The second contribution has studied the consistency of annotations among orthologues[5] in different databases [88]. Experiments on sets of mouse and human proteins resulted in a useful classification of annotation errors and mismatches, as well as in effective techniques for their detection.

These studies offer a partial, but quantitative, validation of the main claim that standardization of terminology improves the confidence in the annotation process and facilitates the retrieval of information.

6.4 Current Approaches to Quality

Partly to dominate the complexity of the domain and the broad variability of available techniques, the information management community has been adopting a general approach towards standardization based on: (1) modeling, capturing, and exploiting metadata that describes the experimental processes in detail, known as *provenance*; and (2) creating controlled vocabularies and ontologies used to describe the metadata.

Information quality management may benefit greatly from this approach.

6.4.1 Modeling, Collection, and Use of Provenance Metadata

Throughout this chapter, we have mentioned a number of variability factors that affect the outcome of an experiment. The metainformation about these variables and their impact (i.e., the experimental design and details of experiment execution) is known as *provenance*. The importance of capturing provenance in a formal and machine-processable way has been recognized in the recent past, as a way to promote interoperability and uniformity across labs. The role of provenance in addressing quality issues, however, has not yet been properly formalized. Recent research efforts have been focusing on using provenance and other types of metadata, to allow scientists to formally express quality preferences (i.e., to define decision procedures for selecting or discarding data based on underlying quality indicators) [89].

Standards for capturing provenance are beginning to emerge, but much work is still to be done. Within the transcriptomic community, one initial response comes from the Microarray Gene Expression Data (MGED) society, which has proposed a standard set of guidelines called MIAME [90], for Minimal Information About a Microarray Experiment, prescribing minimal content for an acceptable database submission. Along with content standardization, the Microarray and Gene Expression (MAGE) group within MGED in collaboration with the Object Management Group (OMG) also defines MAGE-OM, an object model describing the conceptual structure of MIAME documents. The model has been mapped to MAGE-ML, an

5. Orthologues are similar genes that occur in the genomes of different species.

XML markup language for writing MAGE-OM documents, resulting in a complete standard for the preparation of MIAME-compliant database submissions.

Furthermore, MAGE prescribes that experiment descriptions be annotated using the MGED ontology, a controlled vocabulary for the gene expression domain. MGED is currently being redesigned, with the goal of encompassing a broader domain of functional genomics, and will hopefully include a structure and terminology for experimental variables, which is currently missing. Writing complete MAGEML documents is a lengthy process for nontrivial experiments. At present, adoption of the standard by the research community is driven mostly by the requirement that data submitted to major journals for publication be MIAME-compliant.

Similar efforts are under way in the proteomic field [91], although accepted standards do not yet exist for data models and format (although some proposed data models like PEDRo are being increasingly adopted by the community [92]).

The Human Proteome Organisation (HUPO) provides updated information on its Proteomics Standards Initiative (PSI).

The challenge for these standardization efforts is the rapid development of functional genomics. This requires these standards to be specific enough to capture all the details of the experiments, but at the same time to be generic and flexible enough to adapt and be extended to changes in existing or evolving experimental techniques. Furthermore, these standards need to cater for different communities within the large and diverse biological community. Examples of this diversity include the study of eukaryotes or prokaryotes, model organisms that have already been sequenced or nonmodel organisms with only limited amount of information available, inbred populations that can be studied in controlled environment, or outbred populations that can only be studied in their natural environment.[6]

To allow a systems biology approach to the analysis of data from different kinds of experiments, a further effort is undertaken by a number of standardization bodies to create a general standard for functional genomics (FuGE).[7] This effort is based on the independent standards for transcriptomics and proteomics mentioned above and seeks to model the common aspects of functional genomics experiments.

One of the practical issues with provenance data is that, in the wet lab, the data capture activity represents additional workload for the experimenter, possibly assisted by the equipment software. The advantage in the dry lab is that extensive information system support during experiment execution is available, in particular based on workflow technology, as proven in the myGrid project [2, 49]. In this case, provenance can be captured by detailed journaling of the workflow execution.

6.4.2 Creating Controlled Vocabularies and Ontologies

The second approach for a standardized representation of data and metadata is the development of controlled vocabularies and ontologies. A large number of ontologies are being developed, including ontologies to represent aspects of functional genomics experiments, such as the MGED ontology for transcriptomics[8] or

6. See, for example, http://envgen.nox.ac.uk/miame/miame env.html for a proposal to extend the MIAME standard to take into account requirements of the environmental genomics community.
7. http://fuge.sourceforge.net/ and http://sourceforge.net/projects/fuge/.
8. http://mged.sourceforge.net/ontologies/.

the PSI ontology for proteomics,[9] both of which will form part of the Functional Genomics Ontology (FuGO, part of FuGE).

As for the development of standardized models for metadata, the development of standardized controlled vocabulary faces similar challenges, such as the rapid development of the technologies that are described in the ontology or the knowledge presented in a controlled vocabulary. Furthermore, the representation of the ontologies varies, ranging from lists of terms to complex structures modeled using an ontology language, such as OWL.[10]

6.5 Conclusions

We have presented a survey on quality issues that biologists face during the execution of transcriptomics and proteomics experiments, and observed that issues of poor quality in published data can be traced to the complexity of controlling the biological and technical variables within the experiment.

Our analysis suggests that, despite their differences, a common structure and a common set of quality issues for the two classes of experiments can be found; we have proposed a framework for the classification of these issues, and have used it to survey current quality control techniques.

We argued that the scientists' ability to make informed decisions regarding the quality of published data relies on the availability of metainformation describing the experiment variables, as well as on standardization efforts on the content and structure of metadata.

The area of information management can play a major role in this effort, by providing suitable information management models for metadata, and tools to exploit it. Although the literature offers many more results on these topics that can be presented here, we have offered a starting point for in-depth investigation of this field.

Acknowledgments

We thank Dr. Simon Hubbard and his group at the University of Manchester for help during the preparation of this manuscript and Dr. Suzanne Embury and Prof. Norman Paton at the University of Manchester for valuable comments. This work is supported by the Wellcome Trust, the BBSRC, and the EPSRC.

References

[1] Ideker, T., T. Galitski, and L. Hood, "A New Approach to Decoding Life: Systems Biology," *Ann. Rev. Genomics Hum. Genet.*, Vol. 2, 2001, pp. 343–372.

[2] Greenwood, M., et al., "Provenance of E-Science Experiments: Experience from Bioinformatics," *OST e-Science Second All Hands Meeting 2003 (AHM'03)*, Nottingham, U.K., September 2003.

9. http://psidev.sourceforge.net/ontology/.
10. http://www.w3.org/TR/owl-features/.

[3] Lacroix, Z., and T. Critchlow, (eds.), *Bionformatics: Managing Scientific Data*, New York: Elsevier, 2004.

[4] Rahm, E., (ed.), *First International Workshop, DILS 2004*, Vol. 2994 of *Lecture Notes in Bioinformatics*, March 2004.

[5] Winkler, W., "Methods for Evaluating and Creating Data Quality," *Information Systems*, Vol. 29, No. 7, 2004.

[6] Naumann, F., et al., "Completeness of Integrated Information Sources," *Information Systems*, Vol. 29, No. 7, 2004, pp. 583–615.

[7] Motro, A., P. Anokhin, and A. Acar, "Utility-Based Resolution of Data Inconsistencies," *Intl. Workshop on Information Quality in Information Systems 2004 (IQIS'04)*, Paris, France, June 2004.

[8] Campbell, A., and L. Heyer, *Discovering Genomics, Proteomics, and Bioinformatics*, San Francisco, CA: Benjamin Cummings, 2003.

[9] Ballou, D., et al., "Modelling Information Manufacturing Systems to Determine Information Product Quality," *Journal of Management Sciences*, Vol. 44, April 1998.

[10] Wang, R., "A Product Perspective on Total Data Quality Management," *Communications of the ACM*, Vol. 41, No. 2, 1998.

[11] English, L., *Improving Data Warehouse and Business Information Quality: Methods for Reducing Costs and Increasing Profits*, New York: John Wiley & Sons, 1999.

[12] Redman, T., *Data Quality for the Information Age*, Norwood, MA: Artech House, 1996.

[13] Mueller, H., F. Naumann, and J. Freytag, "Data Quality in Genome Databases," *Proc. 8th International Conference on Information Quality (ICIQ03)*, MIT, Cambridge, MA, 2003.

[14] "Microarray Standards at Last," *Nature*, Vol. 419, No. 6905, 2002, p. 323.

[15] Mewes, H., et al., "MIPS: Analysis and Annotation of Proteins from Whole Genomes," *Nucleic Acids Res.*, Vol. 32, 2004, pp. D41–D44.

[16] Apweiler, R., et al., "UniProt: The Universal Protein Knowledgebase," *Nucleic Acids Res.*, Vol. 32, 2004, pp. D115–D119.

[17] Bairoch, A., et al., "Swiss-Prot: Juggling Between Evolution and Stability," *Brief Bioinform*, Vol. 5, 2004, pp. 39–55.

[18] Lockhart, D., and E. Winzeler, "Genomics, Gene Expression and DNA Arrays," *Nature*, Vol. 405, 2000, pp. 827–836.

[19] Bowtell, D., "Options Available—From Start to Finish—Obtaining Expression Data by Microarray," *Nat. Genet.*, Vol. 21, 1999, pp. 25–32.

[20] Holloway, A., et al., "Options Available—From Start to Finish—for Obtaining Data from DNA Microarrays II," *Nat. Genet.*, Vol. 32, 2002, pp. 481–489.

[21] Bolstad, B., et al., "Experimental Design and Low-Level Analysis of Microarray Data," *Int. Rev. Neurobiol.*, Vol. 60, 2004, pp. 25–58.

[22] Quackenbush, J., "Computational Analysis of Microarray Data," *Nat. Rev. Genet.*, Vol. 2, 2001, pp. 418–427.

[23] Leung, Y., and D. Cavalieri, "Fundamentals of cDNA Microarray Data Analysis," *Trends Genet.*, Vol. 19, 2003, pp. 649–659.

[24] Cheung, V., et al., "Making and Reading Microarrays," *Nat. Genet.*, Vol. 21, 1999, pp. 15–19.

[25] Lipshutz, R., et al., "High Density Synthetic Oligonucleotide Arrays," *Nat. Genet.*, Vol. 21, 1999, pp. 20–24.

[26] Kaminski, N., and N. Friedman, "Practical Approaches to Analyzing Results of Microarray Experiments," *Am. J. Respir. Cell Mol. Biol.*, Vol. 27, 2002, pp. 125–132.

[27] Dudoit, S., et al., "Statistical Methods for Identifying Differentially Expressed Genes in Replicated cDNA Microarray Experiments," *Statistica Sinica*, Vol. 12, 2000, pp. 111–139.

[28] Aebersold, R., and M. Mann, "Mass Spectrometry-Based Proteomics," *Nature*, Vol. 422, 2003, pp. 198–207.

[29] Pandey, A., and M. Mann, "Proteomics to Study Genes and Genomes," *Nature*, Vol. 405, 2000, pp. 837–846.

[30] Patterson, S., and R. Aebersold, "Proteomics: The First Decade and Beyond," *Nat. Genet.*, Vol. 33, 2003, pp. 311–323.

[31] Hunter, T., et al., "The Functional Proteomics Toolbox: Methods and Applications," *J. Chromatogr. B*, Vol. 782, 2002, pp. 165–181.

[32] de Hoog, C., and M. Mann, "Proteomics," *Annu. Rev. Genomics Hum. Genet.*, Vol. 5, 2004, pp. 267–293.

[33] Flory, M., et al., "Advances in Quantitative Proteomics Using Stable Isotope Tags," *Trends Biotechnol.*, Vol. 20, 2002, pp. S23–S29.

[34] Perkins, D., et al., "Probability-Based Protein Identification by Searching Sequence Databases Using Mass Spectrometry Data," *Electrophoresis*, Vol. 20, 1999, pp. 3551–3567.

[35] Eng, J., et al., "An Approach to Correlate Tandem Mass Spectral Data of Peptides with Amino Acid Sequences in a Protein Database," *J. Am. Soc. Mass. Spectrom.*, Vol. 5, 1994, pp. 976–989.

[36] Sadygov, R., D. Cociorva, and J. Yates, III, "Large-Scale Database Searching Using Tandem Mass Spectra: Looking Up the Answers in the Back of the Book," *Nat. Methods*, Vol. 1, 2004, pp. 195–201.

[37] Bakay, M., et al., "Sources of Variability and Effect of Experimental Approach on Expression Profiling Data Interpretation," *BMC Bioinformatics*, Vol. 3, 2002, p. 4.

[38] Yang, Y., and T. Speed, "Design Issues for cDNA Microarray Experiments," *Nat. Rev. Genet.*, Vol. 3, 2002, pp. 579–588.

[39] Molloy, M., et al., "Overcoming Technical Variation and Biological Variation in Quantitative Proteomics," *Proteomics*, Vol. 3, 2003, pp. 1912–1919.

[40] Novak, J., R. Sladek, and T. Hudson, "Characterization of Variability in Large-Scale Gene Expression Data: Implications for Study Design," *Genomics*, Vol. 79, 2002, pp. 104–113.

[41] Hatfield, G., S. Hung, and P. Baldi, "Differential Analysis of DNA Microarray Gene Expression Data," *Mol. Microbiol.*, Vol. 47, 2003, pp. 871–877.

[42] van Bakel, H., and F. Holstege, "In Control: Systematic Assessment of Microarray Performance," *EMBO Reports*, Vol. 5, 2004, pp. 964–969.

[43] Kerr, M., and G. Churchill, "Experimental Design for Gene Expression Microarrays," *Biostatistics*, Vol. 2, 2001, pp. 183–201.

[44] Stewart, I., et al., "The Reproducible Acquisition of Comparative Liquid Chromatography/Tandem Mass Spectrometry Data from Complex Biological Samples," *Rapid Commun. Mass Spectrom.*, Vol. 18, 2004, pp. 1697–1710.

[45] Venable, J., and J. Yates, III, "Impact of Ion Trap Tandem Mass Spectra Variability on the Identification of Peptides," *Anal. Chem.*, Vol. 76, 2004, pp. 2928–2937.

[46] Dobbin, K., and R. Simon, "Sample Size Determination in Microarray Experiments for Class Comparison and Prognostic Classification," *Biostatistics*, Vol. 6, 2005, pp. 27–38.

[47] Riter, L., et al., "Statistical Design of Experiments as a Tool in Mass Spectrometry," *J. Mass Spectrom.*, Vol. 40, 2005, pp. 565–579.

[48] Imbeaud, S., et al., "Towards Standardization of RNA Quality Assessment Using User-Independent Classifiers of Microcapillary Electrophoresis Traces," *Nucleic Acids Res.*, Vol. 33, 2005.

[49] Zhao, J., et al., "Using Semantic Web Technologies for Representing e-Science Provenance," *3rd International Semantic Web Conference (ISWC2004)*, Hiroshima, Japan, November 2004.

[50] Hess, K., et al., "Microarrays: Handling the Deluge of Data and Extracting Reliable Information," *Trends Biotechnol.*, Vol. 19, 2001, pp. 463–468.

[51] Members of the Toxicogenomics Research Consortium, "Standardizing Global Gene Expression Analysis Between Laboratories and Across Platforms," *Nat. Methods*, Vol. 2, 2005, pp. 1–6.

[52] Larkin, J., et al., "Independence and Reproducibility Across Microarray Platforms," *Nat. Methods*, Vol. 2, 2005, pp. 337–343.

[53] Wang, H., et al., "A Study of Inter-Lab and Inter-Platform Agreement of DNA Microarray Data," *BMC Genomics*, Vol. 6, 2005, p. 71.

[54] Petersen, D., et al., "Three Microarray Platforms: An Analysis of Their Concordance in Profiling Gene Expression," *BMC Genomics*, Vol. 6, 2005, p. 63.

[55] Huber, W., et al., "Variance Stabilization Applied to Microarray Data Calibration and to the Quantification of Differential Expression," *Bioinformatics*, Vol. 18, 2002, pp. S96–S104.

[56] Fang, Y., et al., "A Model-Based Analysis of Microarray Experimental Error and Normalisation," *Nucleic Acids Res.*, Vol. 31, 2003, p. e96.

[57] Seo, J., et al., "Interactively Optimizing Signal-to-Noise Ratios in Expression Profiling: Project-Specific Algorithm Selection and Detection P-Value Weighting in Affymetrix Microarrays," *Bioinformatics*, Vol. 20, 2004, pp. 2534–2544.

[58] Levenstien, M., and J. Ott, "Statistical Significance for Hierarchical Clustering in Genetic Association and Microarray Expression Studies," *BMC Bioinformatics*, Vol. 4, 2003, p. 62.

[59] Pepe, M., et al., "Selecting Differentially Expressed Genes from Microarray Experiments," *Biometrics*, Vol. 59, 2003, pp. 133–142.

[60] Pan, K. -H., C. -J. Lih, and S. Cohen, "Effects of Threshold Choice on Biological Conclusions Reached During Analysis of Gene Expression by DNA Microarrays," *Proc. Natl. Acad. Sci. USA*, Vol. 102, 2005, pp. 8961–8965.

[61] Pawitan, Y., et al., "False Discovery Rate, Sensitivity and Samples Size for Microarray Studies," *Bioinformatics*, Vol. 21, 2005, pp. 3017–3024.

[62] Reiner, A., D. Yekutieli, and Y. Benjamini, "Identifying Differentially Expressed Genes Using False Discovery Rate Controlling Procedures," *Bioinformatics*, Vol. 19, 2003, pp. 368–375.

[63] Fenyoe, D., and R. Beavis, "Informatics and Data Management in Proteomics," *Trends Biotechnol.*, Vol. 20, 2002, pp. S35–S38.

[64] Taylor, C., et al., "A Systematic Approach to Modeling, Capturing, and Disseminating Proteomics Experimental Data," *Nat. Biotechnol.*, Vol. 21, 2003.

[65] Challapalli, K., et al., "High Reproducibility of Large-Gel Two-Dimensional Electrophoresis," *Electrophoresis*, Vol. 25, 2004, pp. 3040–3047.

[66] Hancock, W., et al., "Publishing Large Proteome Datasets: Scientific Policy Meets Emerging Technologies," *Trends Biotechnol.*, Vol. 20, 2002, pp. S39–S44.

[67] Listgarten, J., and A. Emili, "Statistical and Computational Methods for Comparative Proteomic Profiling Using Liquid Chromatography-Tandem Mass Spectrometry," *Mol. Cell Proteomics*, Vol. 4, 2005, pp. 419–434.

[68] Smith, R., "Trends in Mass Spectrometry Instrumentation for Proteomics," *Trends Biotechnol.*, Vol. 20, 2002, pp. S3–S7.

[69] Resing, K., and N. Ahn, "Proteomics Strategies for Protein Identification," *FEBS Lett.*, Vol. 579, 2005, pp. 885–889.

[70] Bern, M., et al., "Automatic Quality Assessment of Peptide Tandem Mass Spectra," *Bioinformatics*, Vol. 20, 2004, pp. i49–i54.

[71] Others, J. C., "A Systematic Analysis of Ion Trap Tandem Mass Spectra in View of Peptide Scoring," *Proc. 3rd International Workshop on Algorithms in Bioinformatics (WADI)*, Budapest, September 2003.

[72] Nesvizhskii, A., and R. Aebersold, "Analysis, Statistical Validation and Dissemination of Large-Scale Proteomics Datasets Generated by Tandem MS," *Drug Discov. Today*, Vol. 9, 2004, pp. 173–181.

[73] Nesvizhskii, A., et al., "A Statistical Model for Identifying Proteins by Tandem Mass Spectrometry," *Anal. Chem.*, Vol. 75, 2003, pp. 4646–4658.

[74] Zhang, N., et al., "ProbID: A Probabilistic Algorithm to Identify Peptides Through Sequence Database Searching Using Tandem Mass Spectral Data," *Proteomics*, Vol. 2, 2002, pp. 1406–1412.

[75] Ravichandran, V., and R. Sriram, "Toward Data Standards for Proteomics," *Nat. Biotechnol.*, Vol. 23, 2005, pp. 373–376.

[76] Carr, S., et al., "The Need for Guidelines in Publication of Peptide and Protein Identification Data," *Mol. Cell Proteomics*, Vol. 3, 2004, pp. 531–533.

[77] King, O., et al., "Predicting Gene Function from Patterns of Annotation," *Genome Res*, Vol. 13, 2003, pp. 896–904.

[78] Junker, V., et al., "The Role SWISS-PROT and TrEMBL Play in the Genome Research Environment," *J. Biotechnol.*, Vol. 78, 2000, pp. 221–234.

[79] O'Donovan, C., et al., "Removing Redundancy in SWISS-PROT and TrEMBL," *Bioinformatics*, Vol. 15, 1999, pp. 258–259.

[80] Stein, L., "Genome Annotation: From Sequence to Biology," *Nat. Rev. Genet.*, Vol. 2, 2001, pp. 493–503.

[81] Karp, P., S. Paley, and J. Zhu, "Database Verification Studies of SWISS-PROT and GenBank," *Bioinformatics*, Vol. 17, 2001, pp. 526–532.

[82] Gilks, W. et al., "Modeling the Percolation of Annotation Errors in a Database of Protein Sequences," *Bioinformatics*, Vol. 18, 2002, pp. 1641–1649.

[83] Devos, D., and A. Valencia, "Intrinsic Errors in Genome Annotation," *Trends Genet.*, Vol. 17, 2001, pp. 429–431.

[84] Wieser, D., E. Kretschmann, and R. Apweiler, "Filtering Erroneous Protein Annotation," *Bioinformatics*, Vol. 20, 2004, pp. i342–i347.

[85] Prlic, A., et al., "WILMA-Automated Annotation of Protein Sequences," *Bioinformatics*, Vol. 20, 2004, pp. 127–128.

[86] The Gene Ontology Consortium, "Gene Ontology: Tool for the Unification of Biology," *Nat Genet*, Vol. 25, 2000, pp. 25–29.

[87] Lord, P., et al., "Investigating Semantic Similarity Measures Across the Gene Ontology: The Relationship Between Sequence and Annotation," *Bioinformatics*, Vol. 19, 2003, pp. 1275–1283.

[88] Dolan, M., et al., "A Procedure for Assessing GO Annotation Consistency," *Bioinformatics*, Vol. 21, 2005, pp. i136–i143.

[89] Missier, P., et al., "An Ontology-Based Approach to Handling Information Quality in e-Science," *Proc. 4th e-Science All Hands Meeting*, 2005.

[90] Brazma, A., et al., "Minimum Information About a Microarray Experiment (MIAME)-Towards Standards for Microarray Data," *Nat. Genet.*, Vol. 29, 2001, pp. 365–371.

[91] Orchard, S., H. Hermjakob, and R. Apweiler, "The Proteomics Standards Initiative," *Proteomics*, Vol. 3, 2003, pp. 1374–1376.

[92] Garwood, K. et al., "PEDRo: A Database for Storing, Searching and Disseminating Experimental Proteomics Data," *BMC Genomics*, Vol. 5, 2004, p. 68.

CHAPTER 7

Data Management for Fungal Genomics: An Experience Report

Greg Butler, Wendy Ding, John Longo, Jack Min, Nick O'Toole, Sindhu Pillai, Ronghua Shu, Jian Sun, Yan Yang, Qing Xie, Regis-Olivier Benech, Aleks Spurmanis, Peter Ulycznyj, Justin Powlowski, Reg Storms, and Adrian Tsang

Data management for our fungal genomics project involves the development of four major databases, numerous analysis pipelines, and their integration. The project is vertically integrated—that is, it covers the stages from gene discovery to enzyme classification, including the construction of cDNA libraries, EST sequencing and assembly, gene annotation, microarray transcription profiling, gene expression, and enzyme assays. We describe our bioinformatics platform and our issues, challenges, and priorities. Two primary concerns have been the ease of collecting data and the tracking of the quality and provenance of data.

7.1 Introduction

Over the last 3 years we have been developing a bioinformatics platform to support a comprehensive project on fungal genomics for the discovery of novel enzymes with industrial or environmental applications. This is a large-scale, 3-year project employing more than 40 personnel, with 10 investigators and six collaborators across four institutions in Montreal. The total budget was more than $7 million.

Enzymes are protein catalysts that perform a wide range of chemical reactions. They drive the metabolic activities of microbes that have been used for thousands of years in the production of food and alcohol, and are commonly used food additives today. Enzymes are specific in their action and can be used efficiently and safely to modify fats, carbohydrates, and proteins in the production of specialized foods and in other industrial processes (see Table 7.1 for a small sample). Applications of enzymes include the decomposition of wood, bleaching of wood fibers, deinking of used paper for the paper industry, the production of bioethanol and biodiesel, and the biodegradation of toxic chemicals.

All organisms make enzymes which support their lifestyles. Most fungi adopt a nutritional strategy in which they secrete extracellular enzymes to break down complex substrates and then transport the resulting nutrients into the cells for consumption. To accommodate this way of life, fungi have evolved effective, diverse, and

Table 7.1 Some Industrial Applications of Enzymes

Industrial Process	*Enzymes*	*Applications*
Household and industrial detergents	Proteases, lipases, amylases, cellulases	Cleaning laundry and dishes at a wide range of temperatures
Textile	Cellulases, proteases, amylases, catalases	Dye removal, desizing of starch, improve color brightness and smoothness, degradation of hydrogen peroxide, degumming
Brewing	Alpha-acetolactate decarboxylases, betaglucanases, cellulases, xylanases, proteases	Reduce beer maturation time, improve yield and filterability, extract protein to give desirable nitrogen level
Baking	Alpha-amylases, glucose oxidases, lipases, lipoxygenases, xylanases, proteases	Maximize fermentation, oxidize sulfhydryl groups, dough conditioning, bleaching and strengthening dough
Personal care	Proteases, glucoamylases, glucose oxidases, catalases	Toothpaste, cleaning solution for contact lens

comprehensive arrays of catalytic activities. We search for new enzymes using a functional genomic approach to identify fungal genes that encode extracellular proteins.

Our project is not a genome sequencing project; rather, it samples mRNA transcripts of expressed genes and sequences so-called expressed sequence tags (ESTs). The project is comprehensive in that it is vertically integrated (see Figure 7.1). In essence, it is several major subprojects each carried out on several fungal species and many genes and enzymes, namely: (1) a sequencing project which constructs cDNA libraries, sequences ESTs, assembles ESTs into unigenes, and analyses unigenes; (2) a microarray project which constructs cDNA microarrays for each species and carries out transcription profiling experiments; (3) a curation project where selected target genes and enzymes are manually curated and resequenced to obtain the full-length genes; (4) a gene expression project where full-length genes are spliced into host organisms in order to secrete the selected enzyme; (5) an enzymology project which biochemically assays the enzymes; and (6) an applications testing project. Where the project differs from many other genomic projects is in its vertical integration, the focus on industrial enzymes, and the breadth of species under investigation.

The bioinformatics platform supports data collection and analysis across the entire project. Our aim was to use existing best practice, adopt widely used software, and minimize our internal efforts at software development. Nevertheless, there were several significant systems developed as they had to be customized to our genomics project and its complexity. This being our first project of this scale and breadth, and also our first bioinformatics platform, it was very much a learning process. We are now in the process of reassessing our first version, learning from our experience and mistakes, and designing version 2 of the platform.

The fungal genomics project [1] at Concordia University is identifying novel enzymes in 14 species of fungi shown in Table 7.2. The species were selected to meet the following criteria: (1) available from the American Type Culture Collection

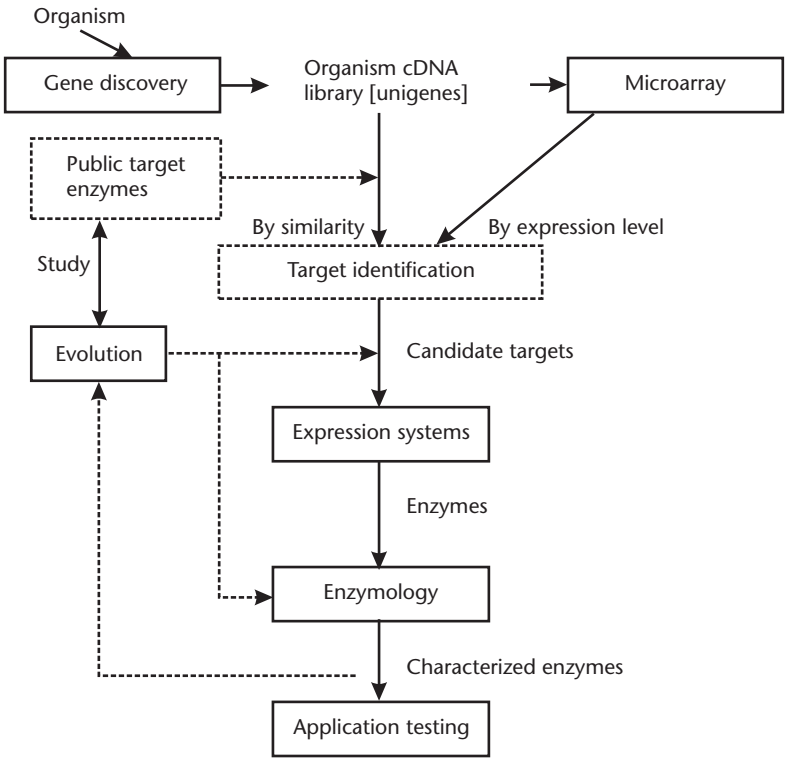


Figure 7.1 Overview of the fungal genomics project.

(ATCC); (2) able to be cultivated (easily); (3) known to have interesting activity; and (4) be a diverse selection.

The general structure of the project is to grow the fungi under a range of conditions of interest, and then sample the mRNA present in the organisms under these conditions. From this, we expect to sample 18,000 cDNA clones per species (equals 252,000 in total). These clones, allowing for duplicates, should provide representatives of approximately 70,000 genes, of which at least 30% will be new genes. From these genes we expect to identify about 3,000 target genes of interest using similarity and using the gene expression studies. Those targets identified by similarity will be the "easy" ones with known relatives amongst the enzyme families. They should be easier to characterize, but will offer lower potential payback in terms of commercialization because they will need to be superior (in some way) to existing known enzymes. Those targets without any known homologues will be more difficult to characterize, but are guaranteed to be novel in some sense, and likelier to have new means of activity and higher commercial payback.

The bioinformatics platform supports five main activities:

1. Creation of a Web site with information on fungal genomics for use by the project researchers, to link to external data, and to provide a focus for dissemination of results of the project;
2. Automated data management and analysis facilities to support the project activities in sequencing, microarray development and gene expression

Table 7.2 The 14 Species of Fungi

Phanerochaete chrysosporium white-rot fungus	Lignin peroxidase and manganese peroxidase were first isolated from nitrogen-limited static cultures of this white-rot fungus. It is known to mineralize polycyclic aromatic hydrocarbons and polychlorinated biphenyls.
Trametes versicolor white-rot fungus	It is able to bleach kraft pulp, solubilize coal, detoxify wood extractives, and degrade synthetic polymers. It is a prolific producer of laccases and peroxidases associated with delignification and degradation of polycyclic aromatic hydrocarbons.
Lentinula edodes shiitake mushroom	It is a selectively ligninolytic white-rot fungus known to produce Mnperoxidase and laccase and to biodegrade chlorinated phenolics in soil and phenolics in wastewaters.
Gloeophyllum trabeum brown-rot fungus	It is involved in wood decay, and produces peroxidases and laccases. Lignin modifying enzymes are poorly characterized in brown-rot fungi compared to white-rot fungi.
Ophiostoma piliferum Blue-stain fungus grows on and discolors wood.	It is used as a source of enzymes that remove pitch, which causes various problems in papermaking. It has been shown to detoxify wood extractives.
Coprinus cinereus	This mushroom synthesizes many peroxidases and laccases that have found application in a variety of industrial processes.
Chrysosporium pannorum	A moderately cellulolytic fungus identified in temperate soils including those of tundra with a minimum growth temperature of –5°C and involved in meat spoilage.
Cryptococcus laurentii	A freeze-tolerant fungus with killer activity which is indigenous to both deciduous and coniferous forests.
Thermomyces lanuginosa	This thermophilic composter, known to produce xylanases and lipases, is one of the most ubiquitous thermophilic fungi. It has been shown to grow at temperatures up to 60°C.
Sporotrichum thermophile	A thermophilic composter isolated initially from straw and leaf matter with an optimum temperature for growth around 50°C.
Aureobasidium pullulans	A black yeast that grows at the expense of aromatics, produces xylanases and antifungals, and colonizes plastic surfaces.
Amorphotheca resinae "kerosene fungus"	Strains of this species grow vigorously in jet fuel.
Leucosporidium scottii	This is one of the most common yeasts of the forest floor and can grow at the expense of a variety of lignin-derived aromatics.
Cunninghamella elegans	This is a nonligninolytic fungus that has been extensively used to study metabolism of aromatic compounds, especially polyaromatic hydrocarbons.

experiments, wet lab chemical assays and characterization of selected enzymes, and the sequencing of genomic DNA for selected enzymes;

3. Environment to support the work of annotation that selects, presents, and links the available data on sequences and genomes;
4. Data quality monitoring across the range of activities that generate data: sequences, microarrays, and assays;
5. General project management.

In terms of hardware, there is a mirrored file server with 2.3-TB capacity, a dedicated Web server, several compute servers, a small cluster, and numerous workstations. We use public domain software for database and Web infrastructure (Apache, mySQL, PHP, Perl); applications in Perl and Java; critical algorithms in C++; data communication in XML; and visualization in Java.

In many scientific databases [2–4], the data is stored in specialized formats in files, and the database tracks these files, their contents, and the associated metadata. Metadata records descriptions of data such as the who, when, where, what, and why of data capture, interpretation, and analysis. This is also true in part of bioinformatics [5]. The raw data in terms of chromatograms of sequences, or TIFF image files of microarray scans, are kept as files. The databases store the interpreted data (i.e., sequences and image intensity values), as well as the metadata.

There are four major databases (see Figure 7.2):

1. A *materials tracking database* records physical entities such as microtiter plates, membranes, slides, and so on, their contents, their location, and tracks the material transfer between them. It tracks the construction of cDNA libraries up to the stage where template plates are physically sent to the sequencing centre (marked "S&T" in Figure 7.2). This is a Laboratory Information Management System (LIMS) custom built in-house for our project.
2. An *annotation database* records information about sequences (ESTs, unigenes, genes, and proteins). It stores the results of EST base calling and assembly, as well as annotations derived from tools for sequence analysis.

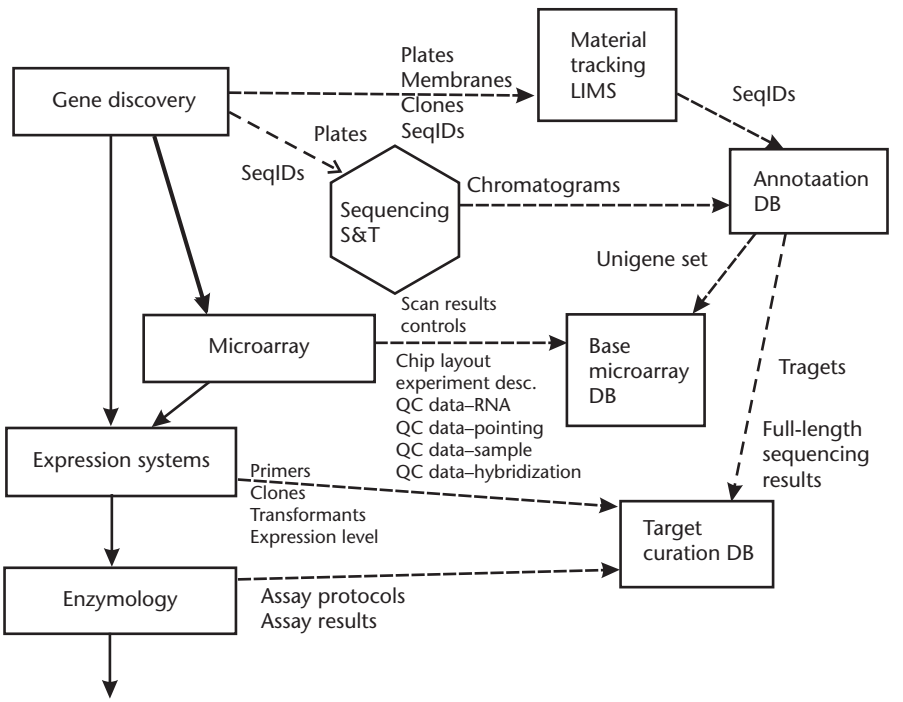


Figure 7.2 Flow of data in the fungal genomics project.

3. A *microarray database*, the BASE system [6] from Lund, Sweden, is MIAME-compliant [7]. It records all the metadata, data, and data analysis results for microarray experiments.
4. A *target curation database* records information about potential and selected target genes and enzymes. It also supports the work on the expression of genes in hosts, the biochemical assays of enzymes, and the manual curation of targets.

The processing of bioinformatics data can conceptually be thought of as several processes, though each of them contributes to each other, and to the overall information available to the curators:

1. To process EST sequences by performing base calling, quality control, and assembly;
2. To annotate sequences with GO terms [8] and collect information about sequences using a range of widely used tools;
3. To identify potential target genes and enzymes based on sequence information;
4. To support full-length resequencing of selected target genes by designing primers, processing sequences, and reassembly;
5. To transfer gene information from the annotation DB to the microarray DB when designing a cDNA microarray for a species;
6. To create data warehouses for our partners to access;
7. To create data warehouses for the public;
8. To submit sequence data to the Genbank archive at NCBI;
9. To create tables, figures, and supplementary material in preparation for common types of publications.

Several important tasks are done manually or semimanually. The transfers to the sequencing center are done physically for the actual microtiter plates, and via ftp for information about the plate contents. The resulting chromatograms are downloaded via ftp from the sequencing center. The analysis of microarray data is done interactively within BASE using tool plug-ins and following a standard protocol. The interpretation of the microarray results for identifying potential target genes and enzymes is done manually. The selection and curation of targets is done manually.

The focus of this chapter is to report on the data management aspects of the fungal genomics project. Data management for our fungal genomics project involves four major databases: we describe each of these in turn and the related work. The data models are not described in detail due to lack of space. However, we do provide pointers to descriptions of typical data models for each database. In [9] we address issues of data modeling for genomics based on our experience and offer a set of guidelines. Current best practices for modeling and data integration are discussed there. Here we discuss our experience in overall data management, primarily two key issues and challenges: the ease of collecting data and the tracking of the quality and provenance of data.

The development and use of our bioinformatics databases follows three phases: (1) data entry, (2) data access, and (3) data analysis. The first phase begins with trial entry of data during system development and test, followed by high volume data entry as the wet lab processes achieve full production. The second phase begins with the generation of reports which are used to monitor the productivity and quality of the lab processes, followed by queries to help diagnose problems, and then more general ad hoc queries to study the results. The third phase incorporates existing data analysis tools, statistical packages, and data mining techniques which other research groups have demonstrated to be useful for analysis of the particular data in the database. We have gone through all three phases with each of our four databases, except for the target curation DB.

We provide high-level descriptions of the requirements and design of our databases, system components, and pipelines. We believe that these are sufficient to set the context for the experience report. In any case, the details will vary for another research project. Requirements for LIMS track closely the actual procedures and techniques used in the wet lab, and these are highly project specific. Annotation databases are now well described, for example, for Ensembl [10] which annotates the human genome, for the Saccharomyces Genome Database (SGD) [11] which annotates the yeast genome, and for Pedant [12] which annotates many genome and EST projects. Microarray databases have an enormous literature: we use the BASE system. A full description of the target curation DB will be forthcoming. We are only now at high volume data entry, and initial data querying. We need a few more months before we have the requisite experience with data analysis of enzyme assays.

7.2 Materials Tracking Database

Genomics takes advantage of high-throughput, small-scale procedures. In the gene discovery process, individual cDNA clones are produced. Each is placed in one well of a 96-well or 384-well microtiter plate. As each clone is studied, portions of the material in a well may be transferred to a well on another plate. After a series of steps, a plate will be sent to the sequencing center (marked "S&T" in Figure 7.2). Each well will correspond to a chromatogram ?le containing the sequence of nucleotides in the cDNA clone. From the analysis of the sequence, we identify interesting cDNA clones—our targets—so we have to be able to track back from the sequence to the well that holds the cDNA clone. This is the purpose of our material tracking DB: it is a LIMS for the manipulation of cDNA clones.

While the general purpose of the system is clear, it seems that each LIMS is highly customized to its lab setting. There are many commercial LIMS systems available, but we could find no open source LIMS. The commercial products were for unrelated lab work such as pathology labs or cancer labs handling tissue and fluid samples or for sequencing centers. The latter recorded the sequencing results but did not cover the front-end processes in gene discovery.

Our development took 8 months initially, followed by continual refinement due to full-length resequencing of clones during the curation, cloning, and expression of targets. The initial development focused heavily on modeling the domain of gene discovery—that is, the lab processes and materials. Then there was refinement of

data entry interfaces and procedures to eliminate errors and make data entry more efficient. One key aspect to this refinement of data entry concerned the use of membranes.

Membranes are large sheets, about 9 inches by 9 inches, which you can spot with clones. A membrane is large enough to hold colonies from 24 384-well plates. The spots on a membrane are read using digital cameras or CCDs to record fluorescence or intensity of transmitted light (i.e., from an X-ray image of a membrane where the spots have been labeled with radioactive isotopes). In gene discovery, the membranes are used in the selection of clones for sequencing. There are three sets of data that each membrane can provide under our lab protocols: growth information, virtual subtraction information, and direct subtraction information. These may be used separately or together in the selection process.

Initial automation of the use of membranes cut the time to process the data for one membrane from 3 or more days to 1 day. Further improvements to the data entry interfaces, including tuning of the database transaction for data entry, and the analysis of the data reduced the time to 3 hours or less.

7.3 Annotation Database

The term "annotation" has a slightly different meaning when discussing EST annotation and genome annotation. Genome annotation is first concerned with predicting which segments of the genome are genes and what are the coding sequences within the gene. Only then does genome annotation seek to understand the role of the gene product: in our case, the enzyme. In an EST project, you immediately have the (partial) coding sequence within the gene, so the only focus is on annotating the role of the gene product. As a result, many systems related to genome annotation were not at all useful. But many of the principles were, and the domain was now an in silico domain.

The needs of an annotation database keep growing because our ability to analyze sequence data and predict the function of an enzyme is so limited. As each new analysis tool comes along, there are cases where it helps understand a sequence, and so an argument is made to include the tool in the annotation pipeline and the results in the annotation database.

The initial pipeline [13] for base calling and assembly uses "standard" software tools phred, Lucy, and phrap. The annotation pipeline [14] uses a set of common software tools and databases as well as two in-house tools. Our in-house tool TargetIdentifier [15] matches our sequences against a database of known enzymes and checks whether our cDNA is full-length or not. Our in-house tool OrfPredictor [16] translates the nucleotide sequence to a sequence of amino acids in any of the six frames.

The design of the annotation database [14] is influenced by the Gene Indices of TIGR [17] and the decoupling of sequence and annotation found in the Distributed Annotation System (DAS) [18]. The following systems have excellent descriptions of their data models: Ensembl [10], the Saccharomyces Genome Database (SGD) [11], and Pedant [12]. Pedant includes a comprehensive list of the annotation tools it uses.

When this work began, there were few systems available for EST processing, EST annotation, and curation. Since then, several systems for EST processing and initial annotation have become available [19–24].

7.4 Microarray Database

Finally, there was a system that we did not have to develop ourselves. Just prior to our project there was an enormous interest in the new technology of microarrays, particularly the concern to be able to archive and share the experimental data in a meaningful way. This led to a standard (MIAME) for recording data, several open source systems, and an open source effort (Bioconductor) for statistical analysis tools [25]. We adopted the BASE system [6] from Lund, Sweden, which is MIAME-compliant [7]. It records all the metadata, data, and data analysis results for microarray experiments. There is a plug-in architecture for adding new analysis tools: we have added several from the MEV suite from TIGR [17], from Bioconductor, and our own.

The metadata requirements of MIAME are extensive. They cover the design and production of the microarrays themselves, the source and preparation of samples, and the protocols for the extraction, labeling, hybridization, scanning, and analysis steps of the experiment. Quality control data as well as experimental results are collected (see Figure 7.2).

A microarray provides information about the expression level of every gene of the organism under a particular condition: in our case a condition of growth for the fungi. Basic data analysis up to the determination of the set of genes which have statistically significant changes in expression is well understood now. Other techniques include clustering [26] and pathway analysis [27]. However, the interpretation of the biological reasons for these changes is extremely hard, and varies from experiment to experiment. The interpretation relies heavily on the annotation of the genes for the species: typically there are 30% of the genes which are well annotated, 30% of the genes with no annotation, and 20% of the genes in a gray area where much may be known yet the role of the gene is not truly understood.

7.5 Target Curation Database

The potential target enzymes are further investigated. First, the sequence for the complete enzyme is required: a cDNA sequence gives about the first 800 base pairs, and assembly of several cDNA sequences can yield up to 3,000 or more base pairs, but there is no guarantee it is complete. Knowing the full-length sequence allows more accurate in silico analysis to predict the function of the enzyme, and is required for subsequent cloning, expression, and hence assaying.

We could find no examples of existing systems to record the activities and results of the enzyme assay process. Most biochemical labs use lab notebooks, spreadsheets, and math software for curve fitting. The high-throughput approach we were taking with microtiter plates made our requirements unique.

Again, we went through extensive domain analysis. This time we had the luxury of starting this analysis with master's-level students [28] about 18 months before the lab work began. Our experience with the metadata requirements mandated for microarray data allowed us to readily see similar needs here to record protocols, personnel, batches of reagents, and batches of enzyme and purified enzyme, as well as the actual results of the assays.

Even so, the nature and variety of assays evolved, and we changed strategy from only screening enzymes on those substrates as indicated by in silico analysis to broader screening of enzymes across all assays. This translates to some 20,000 assays in a 4- to 6-week period when we process a batch of enzymes.

As indicated in Section 7.1, we are only at phase 1 of the development and deployment of the target curation DB. We expect it to undergo modifications as we learn more about the data access requirements, especially in sharing data with our partners, and in analyzing the data and utilizing it to improve our in silico annotation. In due course, a full description will be published.

7.6 Discussion

We adopted an iterative development process for each of our databases. The initial version typically took one developer about 4 months, including intensive discussions with scientists and lab technicians to define the requirements. Once deployed, there were one or two further versions developed in response to the experience of the lab personnel. Ease of data entry and the minimization of errors in data entry were the focus of most improvements. One situation required extensive database tuning in order to speed up the data entry of large data files containing some 10,000 datapoints, but in general we did not need to tune the databases. The lab practices evolved as a result of the data management systems being in place. Data was collected differently; more data could be collected and analyzed; analysis was faster; and feedback on the quality of the lab work was more timely.

The project overall was a pipeline where different platforms came into play over a 2-year period. This allowed us to concentrate on one new data management system at a time. As the project involved a significant learning curve for all the bioinformatics personnel, this was a good thing. Having built all the platforms, now is the time to review our experience, redesign the systems, and consider how best to integrate them.

Basic data management was impacted once later platforms came online. These often demanded variations on existing features that we did not foresee. For example, the expression platform involved determining the full-length sequence of a gene. Now each clone was associated with many reads; these were both three and five reads; the assembly process required modification; and we had to track plates containing templates from multiple species. On the positive side, having a full-length gene allowed us to infer more during the annotation process.

The target curation DB differed from the others in that it was planned as a multistage implementation. Stage 1 covered the management of the collection of potential targets with links to the annotation data. It allowed the selection and prioritization of targets for full-length sequencing, including the design of primers, the clone-based

assembly, and further curation of the full-length gene. Stage 2 managed the data for the expression platform. Stage 3 managed the assay results.

In all cases, there was organizational resistance to adopting the data management systems at the level of the lab personnel. The acceptance of the lead scientists was always there. Often the solution was to explain the needs and benefits more fully to the lab personnel, though there was more than one occasion when the lead scientists simply had to mandate the use of the systems. Once they were in use, the resistance disappeared within 4 to 6 weeks. One reason for the resistance is that most of the benefits accrued to either supervisory staff or to scientists analyzing the final data rather than to the lab personnel themselves. So we had to encourage lab personnel to identify with the project as a whole and not just with their piece of the work.

A key issue from the beginning was data integrity and data quality. We knew the issue of data integration would arise, but we also saw that we could defer its consideration until we had the big picture: that is, each platform had its own data management support. Our approach to data integration in the short term was to either link across databases using well-defined identifiers (e.g., for clones, sequences, and unigenes), or to replicate subsets of data (e.g., the target curation DB is incrementally loaded with batches of newly identified target sequences from the annotation DB).

One issue that we had hoped to avoid was the open-ended scope of research. We did not want to have to support any and all kinds of data analysis that creative scientists might desire. The project was already large enough, and we hoped was well defined enough so that the deliverables were clear: manage the data, ensure quality and integrity, and provide analysis to identify targets and record their assay results. Even for the microarray platform, we had a well-defined deliverable: identify the differentially expressed genes, as this sufficed to highlight some targets with unknown function and potential application. However, our view of scope omitted the need to write publications and the fact that most scientific publications need to explain the data: in our case, tell the biological story behind the results of the analysis. This has led to sustained pressure to support more analysis tools, more integrative views of the data, and seek information on metabolic pathways, secretion, and regulation from the data (and the literature). These are open questions in bioinformatics research rather than issues of data management for genomics.

7.6.1 Issue of Data and Metadata Capture

Data entry is a time-consuming and error-prone activity in any field. Much of the effort of user interface design is to minimize the amount of manual data entry, especially data entry that requires typing. When first introducing automation to any setting—and laboratory settings are no exception—there is a cultural change required before technicians and scientists fully adopt centralized data collection and analysis. We have found that the adoption of data entry proceeds faster than the adoption of metadata entry, primarily because the need and benefit of the former is more obvious to the technicians and scientists. This has also been the experience of groups implementing the MIAME standards [7] for metadata of microarray experiments. Yet the capture of metadata for data provenance is critical in science [2, 3]. It is vital to know the history of the data: that is, the source of the data, the data manipula-

tions carried out, and the analysis and interpretation of the data. This history is what we call the *provenance* of the data.

One purpose of our redesign is to have lab workers follow a uniform work pattern so that the capture of metadata can occur behind the scenes. This will make the capturing of metadata on the provenance of data unobtrusive to the users, thus overcoming a major hurdle to its collection.

Provenance metadata includes the context of the laboratory work which produces the data, the source of materials and data from outside the laboratory, the quality of the data, the workflow process description, and the confidence in computational results and manual interpretations.

The context of the lab work is described in terms of people, projects, roles, and activities. The capture of such data can be achieved through activity-based access.

The history of data collection, manipulation, and analysis can be achieved unobtrusively by adopting an activity-based initiation of tasks and having the implementation of those tasks pass through a wrapper that records the metadata.

The quality of the data requires each activity to be able to associate a measure of quality with its results. For consistency, comparison, combination, and interpretation, there needs to be a common measurement system with a well-understood calculus. Similarly, there needs to be a measure of confidence in the results. For a scientist to truly have confidence in a set of results, there needs to be facilities which allow the scientist to see how the results were obtained.

7.6.1.1 Activity-Based Access

Access control is a must for us, as we have a mixture of public and private data, and we have several academic and industrial partners. The Role-Based Access Control (RBAC) approach [29–31] provides a framework based on *users* and their *roles* in a project, and the *permissions* to access *data* and perform *operations* on data are associated with the roles. In its full generality, RBAC organizes roles into a *role hierarchy* and includes *conditions* which constrain the permissions under certain contexts. In an object-based system, we can view the data in units of *objects*. In a workflow-based system, we can view the operations on data as *activities*.

Users access the bioinformatics platform by logging on to the portal. The user database records the roles that a user plays in the projects. It also records the activities associated with each role and the permissions. Users select an activity to perform. This sets the *context* for subsequent interactions of this user session with the bioinformatics platform. Henceforth, a user session carries along this context.

Each activity is associated with underlying software which performs the operations and data manipulations. Typically this is a script executing mySQL commands. The portal provides a Web interface to drive that activity and provides a wrapper which automatically records the desired metadata. The Web interface is typically a form where users input or select values into fields and/or upload files.

Activities are parts of workflows. Scientists think of workflow in terms of experimental *protocols*. These are the descriptions for their work processes in the lab and on the computer. They typically do not make a distinction between manual steps and computer-based steps, as even "manual" lab work involves the use of various instruments (i.e., machines) both large and small. At a basic level, the description of

a workflow can be captured as text, as is the standard practice for representing lab protocols. Essentially we just need identifiers for protocols and activities, and to be able to relate activities to protocols. We do not require a formal process description. Each protocol has an identifier and a (human-readable) description. An activity is also described by an identifier and a description. A protocol is associated with a sequence of activities.

This simplistic view is sufficient. Note that we are not regarding a protocol as a sequential composition of activities, though this may be the typical case. The sequence of activities implies a general flow of work, but allows for activities to be skipped, or repeated, or performed in parallel.

Of course, versioning of protocols and activities is required. This is explicit in their identifiers: the full identifiers distinguish between different versions of the same protocol or activity.

7.6.1.2 Life Cycle of Files and Data

The material tracking DB tracks physical entities within our lab adequately. However, it needs to track the external movement of material and also track digital entities (i.e., files and data) in order to fully capture data provenance. This comes about in several situations due to use of ftp and archiving.

In our transfers to the sequencing center, we physically transfer microtiter plates, and transfer by ftp a spreadsheet documenting the plates and download the set of chromatogram files. At intervals we archive the chromatogram files from our hard disk. All files are archived actually, but there is no easy access to information in the archive or about the archive for users.

Some compute-intensive steps for sequence annotation are off-loaded to facilities at BioNEQ in Montreal or Canadian Bioinformatics Resource (CBR) at Halifax and Calgary. These are done by ftp-ing data files and scripts to the facility within our account there, and later retrieving the result files.

We download software and data from various sites. Data download is generally regular for those data archives that update periodically but software download is not. Often they are done manually.

It is important that these manipulations be captured as metadata for two reasons. First, they are steps within the data processing and need to be fully and accurately recorded so that data provenance is clear. Second, the versioning of databases and software tools requires capturing those downloads and reinstallations. Proper recording of versions of tools and their associated data resources is important for correctly recording data analysis using those tools.

Data integrity can be affected by archiving since datafields may refer to file names as the source of raw data or input to a processing step. Yet archiving may remove the file.

Data integrity can be affected also by references to external data via URLs or other identifiers such as Genbank accession numbers for sequences. These external identifiers are not under our control. They may disappear, or they may no longer refer to the same content. Even Genbank allows submitters of sequence data to edit a submission at a later date.

For these reasons, the metadata must include tracking of files and external identifiers.

7.7 Conclusion

Data management for our fungal genomics project is a large-scale, domain-specific task that faces many of the same challenges of other large-scale database systems. Understanding the goals of the project, the processes in the labs, and the capabilities of the personnel are some of the common challenges. The common solutions to the challenges include an iterative development approach, intensive discussion with lab personnel, and a focus on delivering useful systems. We aimed for many small successes to build a connection with the lab personnel and to ease our own learning curve and doubts. "Big bang" delivery of a complete system would just not have worked: we still do not understand all the requirements for data analysis, even though we are on top of data collection, data integrity, data quality, and data provenance.

Here we have emphasized the need for managing metadata to capture the provenance of scientific data. We have encountered several obstacles which we plan to address in the next version of our data management systems. We have discussed those issues and plans in Section 7.6.

Bioinformatics has developed novel techniques to support the evolution of data models and the integration of data [32], particularly the approach of the Distributed Annotation System (DAS) [18]. We have used these ideas for decoupling data from annotation to good effect [9]. In other work, we are exploring how best to provide access to genomics data either using visual queries [33] or XML form-based queries and reports [34]. Another major need is to move beyond data management to knowledge management. We are exploring the role of the semantic Web with formal ontologies, software agents, and AI tools in the FungalWeb project [35] to assess how well the semantic Web meets this need.

Acknowledgments

This work has been partially funded by NSERC, Genome Canada, and Genome Quebec. Many master's students in the bioinformatics courses at Concordia University, Canada, participated in the study and discussions of the issues, challenges, and existing systems. We wish to thank our colleagues Chellappa Gopalakrishnan, Yueqin Chen, and Longchang Fu, who worked with the bioinformatics team.

References

[1] Fungal Genomics Project Web site, https://fungalgenomics.concordia.ca.

[2] Bose, R., and J. Frew, "Lineage Retrieval for Scientific Data Processing: A Survey," *ACM Computing Surveys*, Vol. 37, No. 1, 2005, pp. 1–28.

[3] French, J. C., A. K. Jones, and J. L. Pfaltz, *Scientific Database Management*, Technical Report Technical Report 90-21, Department of Computer Science, University of Virginia, 1990.

[4] Gupta, A., "Special Section on Data Engineering for Life Sciences," *ACM SIGMOD Record*, Vol. 33, 2004.

[5] Frishman, D., et al., "Comprehensive, Comprehensible, Distributed and Intelligent Databases: Current Status," *Bioinformatics*, Vol. 14, 1998, pp. 551–561.

[6] Saal, L. H., et al., "BioArray Software Environment (BASE): A Platform for Comprehensive Management and Analysis of Microarray Data," *Genome Biology*, Vol. 3, 2002, pp. 0003.1–0003.6

[7] Brazma, A., et al., "Minimum Information About a Microarray Experiment (MIAME)—Toward Standards for Microarray Data," *Nature Genetics*, Vol. 29, 2001, pp. 365–371.

[8] The Gene Ontology Consortium, "Creating the Gene Ontology Resource: Design and Implementation," *Genome Research*, Vol. 11, 2001, pp. 1425–1433, http://www.geneontology.org.

[9] Butler, G., et al., "Guidelines for Data Modeling in Bioinformatics," *Atlantic Symposium on Computational Biology and Genome Informatics, 2003*, Association for Intelligent Machinery, 2003, pp. 907–910.

[10] The Ensembl Genome Browser, http://www.ensembl.org.

[11] Saccharomyces Genome Database, http://www.yeastgenome.org.

[12] Frishman, D., et al., "Functional and Structural Genomics Using PEDANT," *Bioinformatics*, Vol. 17, 2001, pp. 44–57.

[13] Chen, Y., "Pipeline for the Quality Control of Sequencing," Master's thesis, Department of Computer Science, Concordia University, 2003.

[14] Sun, J., "The Design and Implementation of an EST Annotation Database for Fungal Genomics Project," Master's thesis, Department of Computer Science and Software Engineering, Concordia University, 2004.

[15] Min, X. J., et al., "TargetIdentifier: A Webserver for Identifying Full-Length cDNAs from EST Sequences," *Nucleic Acids Res*., Vol. 33, 2005, pp. W669–W672.

[16] Min, X. J., et al., "OrfPredictor: Predicting Protein-Coding Regions in EST-Derived Sequences," *Nucleic Acids Res*., Vol. 33, 2005, pp. W677–W680.

[17] The Institute for Genomics Research, http://www.tigr.org.

[18] Dowell, R. D., et al., "The Distributed Annotation System," *BMC Bioinformatics*, Vol. 2, 2001, p. 7, http://www.biodas.org.

[19] Ayoubi, P., et al., "PipeOnline 2.0: Automated EST Processing and Functional Data Sorting," *Nucleic Acids Res*., Vol. 30, No. 21, 2002, pp. 4761–4769.

[20] Hotz-Wagenblatt, A., et al., "ESTAnnotator: A Tool for High Throughput EST Annotation," *Nucleic Acids Res*., Vol. 31, 2003, pp. 3716–3719.

[21] Kumar, C. G., et al., "ESTIMA: A Tool for EST Management in a Multi-Project Environment," *BMC Bioinformatics,* Vol. 5, 2004, p. 176.

[22] Mao, C., et al., "ESTAP—An Automated System for the Analysis of EST Data," Bioinformatics, Vol. 19, 2003, pp. 1720–1722.

[23] Matukumalli, L. K., et al., "EST-PAGE—Managing and Analyzing EST Data," *Bioinformatics*, Vol. 20, 2004, pp. 286–288.

[24] Xu, H., et al., "EST Pipeline System: Detailed and Automated EST Data Processing and Mining," *Genomics Proteomics Bioinformatics*, Vol. 1, 2003, pp. 236–242.

[25] Gardiner-Garden, M., and T. G. Littlejohn, "A Comparison of Microarray Databases," *Brief Bioinform.*, Vol. 2, 2001, pp. 143–158.

[26] Yang, Y., "Data Storage for Cluster Analysis of Microarray Gene Expression Data," Master's thesis, Department of Computer Science, Concordia University, 2004.

[27] Liu, Y., "Data Modeling for Biochemical Pathways and Microarray Gene Expression," Master's thesis, Department of Computer Science, Concordia University, 2003.

[28] Shu, R., "Design of an Enzyme Activity Mapping Database," Master's thesis, Department of Computer Science, Concordia University, 2003.

[29] Role-Based Access Control (RBAC) official Web site, http://csrc.nist.gov/rbac.

[30] Sandhu, R. S., et al., "Role-Based Access Control Models," *IEEE Computer*, Vol. 29, 1996, pp. 38–47.

[31] Sandhu, R. S., and P. Samarati, *Access Control: Principles and Practice*, New York: IEEE, 1994.

[32] Achard, F., G. Vaysseix, and E. Barillot, "XML, Bioinformatics and Data Integration," *Bioinformatics*, Vol. 17, 2001, pp. 115–125.

[33] Butler, G., et al., "A Graph Database with Visual Queries for Genomics," *Proc. of 3rd Asia-Pacific Bioinformatics Conference*, 2005, pp. 31–40.

[34] Liang, V., and G. Butler, "WISH Query Composer," *6th International Conference on Enterprise Information Systems (ICEIS 2004),Volume 1: Databases and Information Systems Integration,* 2004, pp. 566–569.

[35] FungalWeb Semantic Web Project Web site, http://www.cs.concordia.ca/FungalWeb/.

CHAPTER 8

Microarray Data Management: An Enterprise Information Approach

Willy A. Valdivia-Granda and Christopher Dwan

The extraction of information from high-throughput experiments is a key aspect of modern biology. Early in the development of microarray technology, researchers recognized that the size of the datasets and the limitations of both computational and visualization techniques restricted their ability to find the biological meaning hidden in the data. In addition, most researchers wanted to make their datasets accessible to others. This resulted in the development of new and advanced data storage, analysis, and visualization tools enabling the cross-platform validation of the experiments and the identification of previously undetected patterns. In order to reap the benefits of this microarray data, researchers have needed to implement database management systems providing integration of different experiments and data types. Moreover, it was necessary to standardize the basic data structure and experimental techniques for the standardization of microarray platforms. In this chapter, we introduce the reader to the major concepts related to the use of controlled vocabularies (ontologies) and to the definition of Minimum Information About a Microarray Experiment (MIAME), and provide an overview of different microarray data management strategies in use today. We summarize the main characteristics of microarray data storage and sharing strategies including warehouses, datamarts, and federations. The fundamental challenges involved in the distribution and retrieval of microarray data are presented, along with an overview of some emerging technologies.

8.1 Introduction

A microarray is a high-density, two-dimensional matrix where thousands of nucleic acid, proteins, or tissues are immobilized on the surface of a glass slide, nylon filter, or silicon wafer. The primary purpose of a microarray is to perform biological screening experiments at the whole genome scale. Each *spot* represents a single biochemical assay *probe* against a particular object of biological interest, perhaps measuring the expression level of a gene, or the binding efficiency of a genomic regulatory element. Using this technology, researchers effectively perform tens of thousands of measurements in parallel.

There are many ways to perform the *spotting* process by which samples are placed on a microarray. In contact printing, mechanical pins can be used to robotically transfer micrograms of probe from storage trays onto slides or membranes. In noncontact printing, ink-jet style printing techniques spray various amounts and configurations of probe. Finally, in situ synthesis using photolithographic methods can build cDNA or RNA strands, residue by residue. Because of the distinction between sample spotting and photolithography, the latter are sometimes referred to as *DNA chips*. For the purposes of this chapter, we refer to both techniques as microarrays. Both contact and noncontact printing give spots of 100 μm in diameter, while photolithography spots are about 20 μm. These processes produce microarrays with spot densities from 10,000 to 250,000 spots per cm^2.

Because the spots printed on an array surface are typically less than 200 mm in diameter, microarrays need to be read by specialized scanners. Most commercially available microarray scanners are inverted florescent microscopes that acquire data at two wavelengths (generally used to record a test and a control signal) using 532-nm (17 mW) and 635-nm (10 mW) lasers. The output of this process will be an image file (~5 Mb) and a text file (~1.5 Mb). The text file provides primary data on the intensity ratios of the two wavelengths, averaged over the area of each spot. In order to assess the contribution of experimental noise and error inherent in this new technology, it has become standard process, in contact and noncontact array manufacture, to place an abundance of replicates of each probe on a single microarray. In addition, most experiments involve multiple copies/instances of each microarray. A single microarray experiment might involve measuring the expression of a particular set of genes at 1-hour intervals during the 24 hours following exposure to some environmental stress. This would produce, with even modest experimental redundancy, nearly half a gigabyte of primary data.

In less than a decade, microarrays have become a widespread technology used for the exploration of molecular activity of biological systems. Since their development, more than 12,000 publications have relied on them for primary experimental results. This demonstrates their impact on biological sciences. The wide use of microarrays is the result of two factors: the decreasing cost of reagents and instruments, and the fact that they are so effective as an experimental technique. Today, the end cost to a researcher to measure the expression of a gene is approximately $0.05 [1]. Of course, this assumes that the researcher is willing to measure gene expression in batches of tens of thousands. The high number of probes permits the exploration of complete genomes, including noncoding regions [2, 3]. The diversification of microarray technology to include tissues [4–6], proteins, and peptides permits interrogation of the molecular activity of the cell at many levels of resolution [1, 7].

An increasing number of laboratories are using microarray-based analysis for disease fingerprinting, toxicological assessment, single nucleotide polymorphism (SNP) analysis, reconstruction of signal transduction pathways, and phylogenomic and epigenetic analysis [8–12]. Microarrays are also ideal for fast, sensitive, specific, and parallelized detection and diagnosis of microorganisms [13] and infection biomarkers. Several researchers have used microarrays for the genotyping of influenza viruses [14, 15]; drug resistant HIV-1 [16]; polioviruses [16]; human papiloma [17]; RNA respiratory viruses [18, 19]; hepatitis B and C [20]; and African swine fever [17, 21]. These applications in primary research and clinical medicine make

microarray technology one of the most progressive approaches to understanding living systems.

In the early stages of microarray technology development, researchers recognized that, due to the size of the datasets involved, computational analysis would be required to properly exploit the information. Early microarrays were very expensive, and for this reason several researchers restricted themselves to analyzing datasets published by others. At this stage, the sharing of microarray data was mainly accomplished by exchanging flat files. This enabled progress, despite the lack of standards to exchange genomic information. The key to this success, however, was the personal communication between the researcher who had done the physical experiment and the one doing the analysis. The use of flat files coupled with the lack of direct communication has several limitations. The primary problem is in the exchange of experimental parameters, the *metadata* without which the raw data is meaningless. Most microarray experiments are composed of many different gene expression data files. To understand the biological significance of its content, it is necessary to integrate several types of genomic information (e.g., the assignment of the molecular function of genes, the history of the samples used on the microarray, the batch and lot numbers of the slide, the settings of the scanner, and so on). There is also difficulty involved in retrieving a subset of genes and expression values from flat files without extensive script programming information. Nonetheless, a main advantage of the use of flat file format is that microarray data could be provided *as is*.

Spreadsheets are another file format used to store and share microarray data. This format not only allows sorting and filtering, but makes it possible to perform basic calculations and to produce graphical representations using add-ins and collections of macros developed specifically to analyze microarray data [22–24].

Unfortunately, spreadsheets are difficult to update or manage remotely. Moreover, the proprietary format of this platform has limited impact in the extensive exchange of microarray data. For this reason, researchers typically link spreadsheets with Web pages in the context of their publication. While requiring little effort to implement, the content and quality of the information contained within the spreadsheets is dependent on the algorithms used for normalizing, filtering, and analyzing. Of course, the above mentioned limitations of metadata transfer apply just as much to spreadsheets.

The wide availability of microarray data has fueled the development of *exploratory research* and the generation of new hypotheses about specific biological processes based on the analysis of large amounts of data. A typical example is the dataset published by Golub et al. [25]. It has been analyzed by different researchers using a variety of statistical and computational methods [26–34]. Because different algorithms applied to the same data can provide new insights about a particular biological process, the integration of different experiments through automated database management systems can have a significant impact on understanding/interpretation. This phenomenon has already been seen with databases storing genomic and protein sequence data. With the emergence of the study of biological systems in a holistic manner (also known as biocomplexity or systems biology), the analysis of microarray data is placed in conjunction with that of the other *omic*

datasets [18, 35–37]. This has enabled the development of multiresolution molecular maps of specific biological processes.

Currently, around 3% of more than 400 biological databases store microarray data [35, 38, 39]. However, many researchers performing microarray experiments are unfamiliar with database concepts and perceive data management systems as *black boxes* for data input and retrieval. With this in mind, the objective of this chapter is to introduce the reader to the basic concepts related to the storage, use, and exchange of microarray data including:

- A description of the use of ontologies to provide a structured vocabulary for cataloging molecular biological components and details about microarray experiments;
- An overview of different data models to exchange genomic information, including the minimum information about a microarray experiment (MIAME);
- A description of different microarray database management systems and the main characteristics of microarray data integration projects, including data warehouses, datamarts, and federated databases;
- An overview of new developments in data storage, exchange, and high-performance computing for the implementation of enterprise data and microarray knowledge management systems;
- A highlight of the main challenges and opportunities related to the development of new exchange systems and the access to data streams.

8.2 Microarray Data Standardization

The issue of data standards, integration, and interoperability has long been of interest to biologists. DNA and protein sequence formats like those used by Genbank, Swiss-Prot, and PDB reflect such need. The structure of this information allows researchers to write specific parsers to retrieve subsets of information which are in an XML or flat file format. When analyzing nucleic or amino acid sequences, researchers are interested in obtaining information other than the sequence data. For example, they might want to know about genomic context: the length of the open reading frame, the frequency and location of known introns, the chromosomal location, and any putative molecular function. In most cases, this information is stored in separate databases.

Because most microarray experiments measure the transcriptional activity of genes, the information about a particular gene is very relevant. Additionally, since the variability and reliability of the experiment is affected by multiple factors, microarray analyses require detailed information about the experiment itself before the raw data can be interpreted at all.

A typical experiment using microarrays involves a team of researchers. Each member has skills in a particular process: from tissue preparation, microarray production (or selection from a corporate provider), RNA extraction and cDNA dye labeling, operation of the microarray scanner, normalization and data analysis. Some of these steps may even be outsourced to external sources. During each step,

different sources of noise and variability are introduced. As a result, missing data, outliers, and variability across replications and laboratories is very common. A researcher integrating different microarray datasets must know the strengths and weaknesses of each, as well as their relative level of appropriateness for the current investigation.

To integrate different databases, we must establish points of reference in the metadata and compare the data from various experiments in light of those reference points. Comparing free text definitions is very difficult. Different research groups may come up with different definitions for a particular experiment or biological process. They also may use very similar words to describe fundamentally different processes. For instance, a researcher might use the term DAG to mean directed acyclic graph, but for most cell biologists it will be the shorthand for diacylglycerol, a key intracellular signaling component in the calcium transduction cascade. Therefore, when integrating genomic information, the reader should be very aware that biology is a massive and dynamic field of experimental study. Word meanings are not stable between experimental domains, and as new discoveries are made, new data definitions of genes, genomes, and biological systems emerge.

To facilitate the retrieval of genomic information and the exchange of microarray data, researchers recently have begun to agree on a common set of terminologies and a minimum set of parameters that should be used to describe experiments involving this technology. The formation of government initiatives for the standardization of protocols and reagents, as well as the use of microarrays in clinical studies and for the diagnosis of pathogens, has prompted this need. In order to provide the reader with the overall understanding of the significance of these implementations, we will review concepts related to the gene and microarray ontologies and the MIAME standard.

8.2.1 Gene Ontologies

The abstraction of real-world concepts is very important in the creation of information exchange systems and the management of knowledge. Most applied mathematics is based on this fundamental truth. In the early 1990s the artificial intelligence community developed a framework for the use of controlled vocabularies to capture and formalize the knowledge in a particular domain. Ontologies specify the terms or concepts and relationships among terms and their intended correspondence to objects and entities that exist in the world. Domain ontologies are specialized collections of names for concepts and relations organized in a particular order. These descriptions and rules are accepted by a community in an interdependent fashion. They allow computer-generated queries to filter and retrieve information based on user-defined constraints [40–42].

The implementation of ontologies can be accomplished using specialized development environments, including the Knowledge Interchange Format (KIF), Ontolingua, WebOnto, μKosmos, Cyc, and Protégée. However, ontologies vary in their coverage, quality, and resolution. From an implementation point of view, three main types of knowledge representation can be distinguished:

1. *Upper ontologies*, also called high-level, core, or reference ontologies, describe common general concepts across different communities (e.g., SUMO and WorldNet).
2. *Intermediate ontologies* are shared ontologies among domains that allow for scalability and join domain and upper ontologies.
3. *Domain ontologies* are restricted in their scope and coverage to the interest of a particular domain (e.g., plant ontology, human anatomy ontology, gene ontology, microarray ontology). Domain ontologies join and leave intermediate and upper ontologies and are in constant development.

The sequencing of genes and genomes led to the proliferation of many biological databases. The information contained in these repositories was designed to be populated and accessed by humans, rather than by computers, and was littered with inconsistencies. The functional role of genes tended to be annotated as free text phrases. Many of these where classified into arbitrary categories. At the very least, competing spellings of common terms made simple text searching unwieldy. As a result, it was difficult to search the databases for the function of a particular gene or biological process. Integrating these repositories was a herculean task, usually only undertaken within a fairly small community surrounding a particular area of research.

To address these issues, Schulze-Kremer [40] proposed the use of ontologies to provide a standardized description of objects and process related to molecular biology. An ontology for the molecular function, biological process, and cellular components of genes was proposed by The Gene Ontology Consortium (GOC) [43]. Their effort lead to the implementation of independent terminologies for species, as well as classifications related to genes.

The gene ontology (GO) now has approximately 17,000 terms and several million annotated instances describing how gene products behave in a cellular context. A particular term is linked directly to some datum in a public database. The GO is used by at least 30 major bioinformatic databases serving researchers interested in more than 140 organisms. Each term in the gene ontology is accessible by a unique identifier (GO ID) and every annotation must be attributed to a source which may be a literature reference or a computer-generated annotation.

In a relatively short time, the GO has been adopted by the biological community. Its impact is due to the strictness and expressiveness that allows software architectures to compute and associate biological information from disparate databases. The GO has also gained considerable credibility for simply starting with a large, overlapping set of definitions, rather than haggling over an exact data modeling standard. For these reasons, the GO has become the de facto standard for biological database ontologies [44].

The graphical representation of the gene ontology is made as a semantic net or conceptual graph—both of which are instances of a directed acyclic graph (DAG). A DAG consists of a set of nodes, and a set of edges. An edge is a pair of nodes, and the order of the nodes in the edge makes a difference—that is, the edge (a,b) is different from the edge (b,a). This type of representation is ideal for path analysis and for understanding the relationships between different hierarchical categories of GO.

8.2.2 Microarray Ontologies

An effective microarray database should allow researchers involved in data analysis to pose a query in terms used by an experimentalist, and retrieve a unified dataset from multiple sources. This, however, requires knowledge of the experimental parameters affecting the reliability and the quality of a particular microarray experiment. To properly join the concepts and definitions describing these experiments and to facilitate automated querying and exchange of this microarray data, a group of public and private researchers formed the MGED Ontology Working Group [45]. This effort is standardizing the terminology required to publish a microarray experiment. The MGED Ontology Working Group is composed of computer scientists, developmental biologists, toxicologists, and the whole microarray community. This group is collaborating on the makeup of a microarray ontology (MO) using each member's knowledge for their area of expertise. MO uses the Protégée development environment and is divided into two parts [46]. The *core layer* is a static ontology describing only essential concepts about microarray experiments. This layer is intended to be relatively static. The *extended layer* describes concepts related to microarray experiments and changes as biological knowledge and microarray platforms evolve.

MO and GO are the first attempts to formalize in a consistent way the description of experiments and the molecular components of the cell. Although the design and implementation of these integration infrastructures is still under development, Soldatova and King [46] have pointed out several awkward linguistic issues in the naming policy and the design of the GO and in particular, the MO. The fact that GO and MO do not contain enough terms to describe actual microarray experiments or biological processes limits its mapping, alignment, and merging to intermediate and upper ontologies. Also in several instances MO uses the same name at different levels of abstraction and allows multiple inheritances of properties. Despite the obvious limitations, MO and GO avoid subjective interpretations of the meaning of microarray experiments and gene descriptions. However, as new experiments become available, the need for its redesign or reconstruction is becoming obvious.

8.2.3 Minimum Information About a Microarray Experiment

To achieve the integration of microarray datasets, researchers need to agree not only on the GO (what we are using or observing) and MO (what data we are collecting), but also on the manner in which the experiment is being conducted. There is a considerable variability in both reagents and reference controls, and therefore, it is difficult to compare microarray data generated by different laboratories [7, 47]. The MIAME strictly defines each of the parameters that should be reported in order to provide sufficient information to allow an outsider to interpret the experiment [48]. Most importantly, the MIAME is facilitating microarray applications in clinical and diagnostic settings. The MIAME annotation has six major sections:

1. Experimental design;
2. Array design;
3. Samples;

4. Hybridization;
5. Measurements;
6. Normalization.

An updated summary of the MIAME guidelines is available in the MGED society Web site. In addition, the MIAME is also serving as a blueprint for the standardization of specific type of experiments [49, 50]. MIAME-Tox includes descriptors for the inclusion of cell types, anatomy terms, histopathology, toxicology, and chemical compound nomenclature in the context of toxicogenomics and pharmacogenomics research [51–53].

8.3 Database Management Systems

The storage, exploration, and exchange of microarray data require computer systems capable of handling many simultaneous users, performing millions of data transactions, and transferring many terabytes of data in a secure and reliable way. Fortunately, there is a robust field of software development known as database management systems (DBMS) dedicated to exactly this task. DBMS tools are frequently referred to as "databases," which leads to confusion between the software infrastructures used to manage the data (the DBMS) and the collection of data being managed. In this section, we are discussing DBMS software. Examples include products such as Oracle, Sybase, DB2, and MySQL. The use of a DBMS can provide many benefits: secure access to both journal published and unpublished data, the elimination of redundant, inconsistent and outdated information, reliable data storage and retrieval, data provenance, and historical recovery.

There is no reason to limit a DBMS to storing only primary data. It is also possible to use DBMS to store data about data, or metadata. However, metadata requirements must be identified a priori, and should include scientific, computing, and administrative considerations. Using the metadata, researchers can compose queries that incorporate the quality, condition, or even physical location. From an implementation point of view, we can divide metadata into the following types:

- *Technical metadata:* This information is primarily used to support the work of the staff that is deploying and implementing a particular DBMS. Technical metadata describes the physical organization of the database, the access policies, user accounts, and the integrity constraints that allow the system to operate effectively.
- *Microarray metadata:* In the context of this document, is the data annotated using the MIAME and GO standards, including the use of the MO.

Using a DBMS, one can vastly accelerate the process of data exchange and analysis and therefore, researchers can improve their understanding of specific biological processes. However, in contrast to data shared via flat files, data stored in a DBMS must conform to specific rules within a mathematical framework known as the *data model*. A data model is a conceptual representation of the mathematical rules that define the relationships between different components of a database. In

other words, the data model defines what data is required, and how it should be organized. Over the years, database researchers have proposed six main data models: file processing, hierarchical, network, relational, object-oriented, and the object-relational. In this document, we focus in on the last three data models which are commonly used to exchange microarray information.

8.3.1 Relational Data Model

The relational data model (R-DM) was developed by Codd (1970). The main idea behind this approach is the representation of data in two-dimensional tables. This data structure in many ways drove the enterprise adoption of computers in financial, business, and research applications. The basic elements of the relational data model are the table (or *relation*) that is composed of rows (*tuples*) and columns (*attributes*). Each table has a unique attribute known as the *primary key* that identifies a tuple. Relationships between two tables are made by matching their primary key values. While the primary key of each table can never be a null value, a *foreign key* permits the association of multiple tables defined by a *schema*. The term schema is often used to refer to a graphical depiction of the database structure and defines the fields in each table, and the relationships between fields.

8.3.1.1 Notation of the Relational Model

Whether the implementation of a relational database is intended to serve the needs of a large number of researchers or small workgroup, the planning of its design is an important step ensuring future performance of the database. Notation is a logical and graphical design technique often used to allow designers, implementers, and users to understand in advance the relationships encoded by the database. The notation is also a valuable graphical representation that facilitates the redesign and update of the database.

The relational model is simple to understand and use, even for those who are not experienced programmers. However, the use of the R-DM is poorly suited to the integration of microarray experiments with other types of genomic information. The relational model does not handle well certain forms of data. This includes images (a key component of microarray experiments), sequence data, and digital documents. These limitations can restrict the scalability and interoperability of a relational microarray database or the type of services that the implementation can provide.

Since most microarray experiments are very complex, the design of the relational database needs to consider the possibility of creating or updating new tables. As the number of tables increase, more complex phrasing becomes necessary. As this information grows and scatters across relations, the query process becomes dependent on the scalability of the system. Because adding and updating tables may be cumbersome, a single very large table with many attributes may be generated. Many of these tables might contain empty tuples which affect the performance of the database and applications reading the output of these files.

Another main disadvantage of the R-DM is the separation of the schema from the application software. This makes updating the schema difficult. This is further

complicated due to the constant evolution of biological databases and their respective schemas. To change the schema, the user needs to understand, at some level, the entire set of tables and the intricate relations of whole design. Since schemas are more valuable when they represent a clear view of the components of the database, schemas should not be affected by implementation considerations, such as limits on the number of classes, tables, or attributes. Therefore, while constructing global schemas it is necessary to detect semantic conflicts among existing tables (such as naming inconsistencies and identical entities entered multiple times).

8.3.2 Object-Oriented Data Model

Object-oriented programming languages originated to overcome some of the scalability limitations of relational databases and quickly become one of the dominant forms for the development of data environments with relatively large-scale software systems. Beginning in the 1980s, the Object Oriented Data Model (OO-DM) was proposed to scale the access of biological and genomic information and to address some of the limitations of the relational data model [55–65]. Many sequencing and genome projects acquired a considerable amount of data in a short period of time. The OO-DM associates actions and functional information along with data. It has been referred to as "data with attitude."

The OO data model *encapsulates* each *tuple* as an *object* into a single unit called *class*. Since the underlying details of a class are masked behind access methods, objects from radically different implementations can be combined in a single query. This allows the OO-DM to provide access to the data via methods or functions which can conceal a certain amount of complexity. This leads to increased portability and interoperability, since interfaces, rather than direct access to underlying data model features, are used. Since the OO-DM provides a more intuitive structure for human access, and because of its inherently modular structure, OO systems tend to be easier to maintain and reuse than purely relational ones. Also the use of object identifiers (OIDs) used to reference the accession methods in objects makes the code more scalable. This can lead to significant performance improvements over relational databases.

Generally speaking, objects have three features: state, identity, and extensibility. Identity assures that we are accessing the correct object. The state is characterized by a set of attributes (the data contained in the object) as well as any history of modification or ownership. Behavior is characterized by a set of methods that are applicable to the object. Extensibility is an especially powerful concept in software development and refers to the ability to add functionality to an existing system without fundamentally changing it. Most important is the idea that old methods of accessing the data should continue to work, even if new features are added. An object-oriented approach to programming provides extensibility in two ways: behavioral extension and inheritance. Objects may be extended by simply adding additional methods. This is tremendously valuable because developers can rely on existing behaviors in building tools that reference the information in the object. An OO approach further promotes extensibility through reuse or inheritance. It is important to note that while the terminology of the OO-DM is inspired in part by

biology, the analogy is limited at best, and the biological metaphors should be taken with a grain of salt.

8.3.2.1 Notation of the OO-DM

Object-oriented notation and modeling is one of the key aspects in the development of an OO database. During this process the use case scenarios, class/object diagrams which represent the main functionality as well as the structural aspects of the system, are presented in an intuitive manner. The procedural control flow of the whole OO database is represented schematically using standardized stereotypes.

8.3.2.2 The eXtensible Markup Language

XML is derived from the Standard Generalized Markup Language (SGML), the international standard for defining descriptions of the structure and content of different types of electronic documents [60]. The XML is a data source in that its presentation is separate from its structure and content. The manipulation of genomic information using XML represents an interesting alternative and is currently implemented in different bioinformatic applications including microarray data integration efforts.

XML not only allows information in different representations to be exchanged between applications in a generic format, but also offers an opportunity to access information managed by heterogeneous DBMSs. The XML data defines the structure and content, and then a stylesheet is applied to it to define the presentation. Since XML data is stored in plain text format, XML provides a software and hardware-independent way of sharing data. Furthermore, XML can be used to represent the query results as datagrams, and Extensible Style Language Transformation (XSLT) provides a mechanism for transforming the datagrams into XML.

The relevance of the XML framework is particularly useful for the reordering of microarray gene expression data. XML provides a framework for tagging structured data that can be used for specific tag sets and therefore for defining standard specifications. An XML document is either well formed, obeying the syntax of XML, or XML valid, conforming to the logical structure defined by document type description (DTD) [60, 66]. The DTD is the classification system that defines the different types of information in any XML document. Any Web page that indicates the DTD to which it conforms will instantly allow the user of an XML-enabled search engine to restrict queries to that DTD-defined space.

The Extensible Markup Language/Resource Description Format (XML/RDF) was developed by the W3C to enhance the XML model and encode metadata concerning Web documents. Instead of defining a class in terms of the properties its instances may have, the RDF vocabulary describes properties in terms of the classes of resource to which they apply. XML/RDF as is, without a higher level formalism that encompasses the expressivity present in frame-based languages, does not go far enough to allow the kind of modeling needed in the bioinformatics community. Three main elements are part of an XML file.

- *XML tag:* A start tag is an element type name enclosed in angle brackets that opens an element. Every start tag must have a corresponding end tag. An end tag finishes the content of an element, comprised of an angle slash and then the element type name, all enclosed by angle brackets.
- *XML attribute:* Attributes are name value pairs that are associated with an element type. They follow the element type name inside the start tag. They can be thought of as the "adjectives" of XML.
- *XML element:* An element consists of a start/end tag pair, some optional attributes defined as key/value pairs, and the data between the tags.

8.3.2.3 The Microarray Gene Expression Markup Language

Microarray Gene Expression Object Management (MAGE-OM) is a data-centric Universal Modeling Language (UML) that contains 132 classes grouped into 17 packages, containing in total 123 attributes and 223 associations between classes reflecting the core requirements of MIAME [45]. MAGE-OM is a framework for describing experiments performed on all types of DNA-arrays. It is independent of the particular image analysis and data normalization algorithms, and it allows representation of both raw and processed microarray data. Since MAGE-OM defines the objects of gene expression data independent of any implementation, it allows users to describe the experimental process using free-text descriptions. There are three abstract classes in MAGE-OM from which all the classes in the model derive from: extendable, describable, and identifiable.

The MGED Society implemented the Microarray Gene Expression Markup Language (MAGE-ML) as an XML representation of the MAGE-OM. A major advantage of the MAGE-ML format is that while it supports information from a variety of gene expression measurements including related data collection methodologies; it does not impose any particular data analysis method [45, 67, 68]. MAGE-ML also has advantages in the sense that many laboratories can verify microarray experiments with other methodologies such as real-time PCR. MAGE-ML is organized into subvocabularies in such a way that the subvocabularies are independent of each other. These subvocabularies are driven by the packages and identifiable classes of the MAGE-OM. The MAGE software toolkit (MAGEstk) is well developed for Perl and Java applications.

8.3.2.4 Limitations of the OO-DM

OO-DMs often assume a network of computers, with processing on the back or front end, as well as intermediate tiers, caching on each level of the database. However, there are very few software systems capable of implementing a full-scale object-oriented data model. While the OO-DM offers scalability, there are more requirements to identify accurately different classes. Therefore, the initial design is important in ensuring the future performance of the database. Without a proper management of each class, the design will not work as *per specification* and the database will be severely impaired.

8.3.3 Object-Relational Data Model

Databases with an Object-Relational Data Model (OR-DM) were developed with the aim of extending the relational information with three key features of the OO-DM: inheritance, behavior, and extensibility. This functionality not only permits the management of native SQL data types, but also the handling of object-oriented multimedia information (e.g., sequences, images, and video). The OR-DM is still relational because the data is stored in relations, but, loosely organized into OO hierarchical categories. As a result, the OR-DM extends the R-DM by transforming the tuple as object and the table as class. While column holds primitive data types, the class can hold data of any type of data. This allows attributes of tuples to have complex types, including nonatomic values such as nested relations while preserving the declarative relational access to data. This results in a very complex data structures known as LOBs (large objects).

Databases designed with the OR-DM are very attractive for the integration of genomic and microarray information. They are frequently used in Web applications and specialized data warehouses; although a more significant impact can be seen in data federations. A database with OR-capabilities can execute complex analytical and multimedia data manipulations (i.e., images, normalized microarray data, as well sequence information) and transform these manipulations into new, complex objects, making OR-DMs ideal for a research enterprise. An OR-DBMS is represented by the PIR database [69], ooTFD (object-oriented Transcription Factors Database) [59]. OR vendors provide products such Oracle, Informix, FirstSQL/J, OpenODB DB2, and Postgre Object-relational mapping.

8.3.3.1 Limitations of the OR-DM

One of the challenges in the implementation of OR-DM is the design of a modular schema capable to allow the reuse when dealing with complex structures. Moreover, the translation layer between relational and object oriented can be slow, inefficient, and very costly. This can result in programs that are slower and use considerable memory.

8.4 Microarray Data Storage and Exchange

Once microarray experiments are in digital format, all the components can be shared, copied, processed, indexed, and transmitted from computer to computer, quickly and flexibly. The development of new technologies to store digital information is transforming the life sciences and enabling scientists to record vast quantities of data. These advances and the improvement in the sensitivity of microarray technology have motivated the development of a considerable number of specialized databases. As the relevance of microarray experiments increases, the use of this technology for diagnostics and clinical research present a new paradigm in the storage of this information. The scientific community has been enthusiastic about microarray technology for pharmacogenomic and toxicogenomic studies in the hope of advancing personalized medicine and drug development. The U.S. Food and Drug Administration (FDA) is proactive in promoting the use of

pharmacogenomic data in drug development. This progress means that in the future, microarray data related to clinical studies and diagnostics needs to comply with regulations mandating data preservation and access.

The scope of different databases provides users with a variety of services while maintaining specific types of information associated with microarray experiments. These databases can store at least five levels of information: (1) the scanned images (raw data), (2) quantitative outputs from image analysis, (3) normalized data, (4) a list of important genes after the analysis process, and (5) the metadata associated with each experiment.

Microarray raw data (images) are the starting point of the analysis process. Storing this information, however, poses practical limitations including the size of and access to the image files. Nonetheless, considering the ongoing development in image analysis software, the storage of any processed form of the original image, without keeping the original image itself, can lead to the argument that the data is outdated as new image analysis methods become available. In early 2001, there was considerable discussion about who should maintain original microarray images and if this was the responsibility of journals, public repositories, or research institutes. Despite the intense debate, no consensus has been reach about whether or not it is cost-effective to store all this information, and, at this point, the publishing authors themselves are responsible for storing (and providing on request) original image files. Certainly, no decision has been made regarding if this task should be ensured by public repositories or the institutions hosting the author of a particular paper [35, 67].

Sharing the extracted (but not normalized, i.e., CEL, GPR) files solves some of the practical limitations related with raw images. This level of data level sharing is well suited for many microarray public and local databases. However, it requires the implementation of appropriate DBMS as well preprocessing tools. Another approach to store microarray data consists of the sharing of normalized expression ratios or summarized values such as signal intensities. In this form, much information about the experiment is lost because the diversity of microarray data normalization and probe level analysis techniques. The last form of microarray data exchange consists of providing a list of genes that significantly differ between experimental samples. Due to the wide variability in accuracy across different analysis methods, this information should be limited only to publications. Finally, the sharing of microarray metadata is another component of the data exchange process; however, it has not received considerable attention.

Considering that microarray experiments are done by different communities and have different scope, we can classify these implementations as:

- *Public*: These types of microarray DBMSs cover different microarray experiments by single or different researchers, and allow users to query, retrieve, and analyze both unpublished and published microarray information.
- *Institutional*: The configuration of this type of microarray DBMS resembles public databases but is built around a particular organism and/or restricts the access to a limited number of researchers depending on some set of permissions that are defined by the institution.

- *Private*: These microarray DBMSs are limited to researchers within a research group and are not available to other researchers.

8.4.1 Microarray Repository

Microarray data repositories are data collections that, in general, are implemented by one institution to serve a research community [61]. These storage and exchange systems allow the submission of data from both internal and external investigators [70, 71]. Although often used synonymously with "data warehouse," a repository does not have the analysis functionality of a warehouse. The maintenance and curation of data repositories has made these data exchange systems of considerable value to specific research communities. Since repositories need to be able to store, access, filter, update, and manipulate large data sets quickly and accurately, the information requires systematic knowledge management, proper representation, integration, and exchange.

8.4.2 Microarray Data Warehouses and Datamarts

Data warehouses are databases devoted to storing relevant information from other sources into a single accessible format [72]. These systems have the advantage that they can import and analyze data that cannot otherwise communicate with each other. Since they incorporate a time factor, data warehouses can present a coherent picture of heterogeneous genomic sources integrated at different time points. In fact, very often, their requirement is to capture the incrementally changed data (delta) from the source system with respect to the previous extract.

Data warehouses are populated from the primary data stores in three main steps often through sophisticated compression and hashing techniques. First, data are extracted from the primary data sources. This process uses monitors/wrappers that are capable of both collecting the data of interest and sending it to the warehouse. The monitor is also responsible for identifying changes in external databases and updating the warehouse automatically. Second, the data are transformed and cleaned. Specific logic for data standardization or for resolving discrepancies between data can be implemented in this step. Third, the data are loaded into the database, and indexes are built to achieve optimum query performance. This configuration facilitates the direct access of microarray data for analysis, allowing for both good performance and extensive analysis and visualization capabilities.

In order to standardize data analysis, data warehouses are organized as problem-driven small units called datamarts. These implementations are subsets of larger data warehouses and contain data that has further been summarized or derived from a main data warehouse. Datamarts are an attractive option because they take less time to implement than a centralized data warehouse and initially cost less. However, datamarts can be more costly in the long run because they require duplicated development and maintenance efforts, as well as duplicated software and hardware infrastructure. Additionally, scattered data marts can hinder enterprise performance because they often store inconsistent data, making one version of "the truth" impossible to obtain.

The development of data warehouses like MaxD and DataFoundry, which integrated Swiss-Prot, PDB, Scop, Chat, and dbEST in a unified schema, represent a clear success of genomic data warehousing [72, 73]. First, it must obviate the need for the conversion and migration of data and must require no change to the local database system. Second, it must allow users to interact in such a way that both users and applications are shielded from the database heterogeneity. Third, allowing the interoperability of heterogeneous databases must allow reads and updates of these databases without introducing changes to them. By their nature, data federations (datamarts) do not modify the primary data sources and a great effort must be paid in the cleaning and transformation before their placement in the warehouse. Since data are drawn directly from the primary data stores, detection and cleaning of redundant data is not easily incorporated [74, 75].

Microarray data warehouses have two costly drawbacks: (1) considerable effort is required for planning the integration; and (2) a great deal of investment is required for data cleaning and transformation. This situation affects reliability and overall system maintenance of the system.

8.4.3 Microarray Data Federations

Most microarray databases are specialized collections of information for a particular organism or biological process. They are scattered in different locations and managed under difference policies. In order to integrate this information, a data federation schema seeks to join isolated, heterogeneous repositories into a single virtual main database. This process is accomplished without modifying the primary data sources and by avoiding the creation of a large warehouse. Their use is motivating the emergence of "virtual organizations" which take advantage of the standardization of microarray protocols and the use of reference probes. In addition, federations rely on the development of GO, MO, MIAME, and MAGE-ML standards which permit the development of wrappers that explore and query multiple data sources and may have different characteristics including:

- *Public data:* Data from public sources, such as ArrayExpress and NCBI-GEO; copies of raw data may be held locally for performance reasons or shared throughout the federation.
- *Processed public data:* Public data that has additional annotation or indexing to support the analyses needed by different analysis algorithms. This information can serve as the common link for joining different databases within the federation.
- *Sensitive data:* In many cases, an individual user will be generating data which remains to be analyzed or is unpublished. This requires careful enforcement of privacy and may be restricted to one site, or even part of a site.
- *Personal research data:* Data specific to a researcher, as a result of experiments or analyses that that researcher is performing. This is not shared even among the local team. It may later become team research data.
- *Team research data:* Data that is shared by the team members at a site or within a group at a site. It may later become consortium research data (e.g.,

when the researchers are confident of its value or have written about its creation and implications).

- *Consortium research data:* Data produced by one site or a combination of sites that is now available for the whole consortium.

While data federations could accelerate the development of data standards, traditional federations might be too rigid and labor-intensive to adapt to an open environment where new sources are integrated dynamically. Therefore, before implementing or joining a data federation, researchers interested in this possibility need to address issues related with the design, interoperability, and security of each transaction and the transfer of high volumes of information.

In most cases, member databases are geographically distributed, hosted on a variety of platforms, and administered independently according to differing policies, which might be independent of the federation policies. This means that the federated system must be designed under the assumption that not all resources will be available and consistent at all times. This makes the quality control very difficult. Because data federations perform a considerable number of data transformations, query performance is one of the main concerns.

8.4.4 Enterprise Microarray Databases and M-KM

From an institutional perspective, the information generated by microarray experiments can be interpreted as data, values, and relations generated by different researchers with a shared common and main institutional goal [76, 77]. In this context, enterprise systems can be defined as computer architectures designed as intranet systems capable of performing pipeline operations. Using specific hardware, software, database management systems, agent software, analysis, and visualization algorithms, enterprise systems integrate information and find patterns and relations over large periods of time. The result of this process is the constant transformation of data into an intellectual asset. The implementation of enterprise microarray data management systems is being enhanced by the development of semantic Web, grid computing, and Internet-2.

The implementation of enterprise microarray data management systems is resulting in a new generation of infrastructures known as knowledge management (KM) systems. The KM concept evolved from information management tools, not only to integrate data, but to integrate many aspects of computer-supported collaborative environments including blogs and wikies. Microarray KM tries to consolidate knowledge that is not easily codified in digital form, such as the intuition of key individuals, with considerable experience interpreting data from a particular biological process, organism, or cellular process. These individuals and/or their collective thinking might recognize various patterns of gene expression profiles that individuals may not recognize. While promising, microarray KM implementation requires a series of standards to enable genomic information to be captured, analyzed, understood, and reapplied in new contexts. Therefore, the implementation of an enterprise analysis system requires:

- *Technical integration:* Use nonproprietary platforms, open standards, and methodologies in the design of the system architecture which ensure long-term scalability, robustness, performance, extensibility, and interoperability with other systems and platforms.
- *Semantic integration:* Use all levels of linked biological concepts and their dependencies in biological, genetic, and microarray ontologies. Manual intervention to map data between different data sources should not be required.
- *Interoperability:* Provide users with the ability to directly import and export gene expression data as a single flat files derived from separate microarray DBMSs.
- *Allow configurable combinations of data sources:* It should be possible to integrate and combine different sources of biological information.

8.5 Challenges and Considerations

Microarray technology has added an important dimension and depth to the analysis of different and dynamic biological processes. The scientific value of this technology is enormous; however, the quality of this information is highly variable. Problems in data quality have been observed from analyzing published datasets, and many laboratories have been struggling with technical troubleshooting rather than generating reliable datasets. Therefore, it is important to recognize that not all datasets are suitable for storage and distribution. Unless a clear description of the experimental design and quality experiment itself is provided (i.e., technical and biological replicates, and the use of appropriate protocols), the query and retrieval of datasets should be limited to published results. The fact that many of these datasets do not provide appropriate metadata makes difficult the incorporation of quality assessment methods. Therefore, it is necessary to implement semiautomated approaches that score the level of reliability of the data. Developing better systems for collecting metadata, either manually or automatically, is one of the most urgent issues needing attention.

Several microarray databases and analysis software overcome national boundaries. This is particularly true in the sharing of microarray data, where scientists on a global basis deposit and retrieve data irrespective of who funded the information production. Some microarray databases have already surpassed a terabyte scale. The implications of the accumulation of this information has been not fully recognized. There are several critical design issues in databases which affect how new databases and analysis systems are implemented. Performance and efficiency not only needs to be measured by query response time, but by the time it takes a scientist to extract knowledge from the data. Adopting standards which are likely to survive and/or are well described for the future is difficult. Therefore, it is necessary to motivate the reuse of software and the development of approaches to decrease the risk of data loss or the expense of data resurrection.

Large data repositories, computationally intensive data analysis, and visualization tools pose difficult problems for the implementation of open access enterprise microarray data management and KM systems. Commonly, database schemas are changed without any notification, explanatory documentation, or appropriate nota-

tion. This makes the maintenance and improvement of these systems difficult. These challenges are complicated by the fact that Internet bandwidth and data compression technologies have not kept pace with the growth of scientific data sets. Many data repositories still provide data access primarily via FTP. While FTP-based data sharing is a valuable starting point, we need to encourage more robust interfaces, capable of retrieving specific datasets automatically. This is perhaps a main bottleneck in the automatic retrieval of databases since there is poor communication on the part of the resource maintainers. Moreover, large data archives are becoming increasingly "isolated" in the network sense.

In order to work with large datasets, it might be necessary to send computations to the data, rather than copying or moving the data across the Internet. A limiting aspect in the development of microarray data storage and exchange systems is related to the complexity and dynamics of the data itself. Complexity arises from the lack of unique spot identifiers and the existence of a large number of many-to-many relationships among clones, accession numbers, chromosomal location, mutation types, and so on. In addition, microarray datasets derive the treatment of biological samples (with different genetic background) to multiple experimental conditions and time courses. The dynamics of microarray data result from the terminology used for the description of a biological sample and the functional role for a particular gene or its transcriptional variants. These attributes can change as new discoveries update this information. As a result, the interpretation of a particular microarray dataset is highly dependent on ever-growing and dynamic annotation information. Although the use of microarray data analysis tools is beyond the scope of this chapter, the reader should be aware that the annotation of unknown genes using ontologies depends on analysis algorithms and the amount of information used in the analysis process. It is now more evident that the "guilt by association" is not always true.

The reader must be aware that deciding on appropriate terms that could be used in the development of microarray ontologies and mapping them to other middle and upper ontologies entails main decision points. First, the implementation of a large and comprehensive ontology versus several smaller task-oriented ontologies is still a subject of discussion. One alternative (large ontologies) presents challenges regarding agreement across subdisciplines. Second, coordination between different small ontologies could be very expensive. In both situations, it is necessary to consider how the dynamics of the ontology will affect a database. This is important in biological ontologies because they do not remain static; they evolve as new discoveries are made. By restricting access to or simplifying assumptions about a particular dataset in order to accommodate it to a particular ontological definition, the user risks the trivializing the queries and results.

The annotation of new genes based on combined gene expression values using an integrated view of different microarray datasets can lead to a "transitive catastrophe" or "data poisoning," in which one piece of inaccurate information can corrupt a large number of derived results. This legacy issue is becoming more significant since the functional inference of genes and transcriptional interactions changes with time and is not straightforward. As more microarray data becomes available, it is becoming evident that biological systems are organized as

transcriptional networks with specific modular components, rather than in a particular class or cluster of similar gene expression values.

8.6 Conclusions

Since the early 1990s, when scientists first began using microarray devices to study gene expression, they have widened the use of this technology to studying how genes interact at the transcriptional, proteomic, and metabolomic levels. The rapid increase in the size and diversity of this type of information has highlighted the need for efficient computational techniques for data storage and exchange. The Internet has made it possible to access large amounts of information from multiple microarray databases distributed across the world. This is stimulating a growing demand for analysis and visualization systems of multiple and heterogeneous biological data sources. However, even when a global network infrastructure provides the foundation for the microarray data sharing and exchange, the location, retrieval, and the combination of disparate and poorly annotated microarray datasets has proven to be a complex and a time-consuming task.

Researchers recognize the benefits of integrating microarray with other genomic information. Investing in these efforts not only saves time, but also makes more effective experimental designs and reduces experimental resource expenses. Due to the large number of data points and since the analysis of the same data using different computational techniques can lead to a better understanding of the biological process, different microarray data repositories are playing a vital role in biological sciences. *Data exploration research* is now impacting traditional wet lab experiments from hypothesis generation to experimental design and data analysis. However, how good genomic data mining is made depends on the time and care that is spent when designing and implementing a data storage and exchange system, especially now that a new generation of researchers no longer "do" wet lab experiments. Instead they "mine" available microarray databases, looking for new patterns and discoveries.

The integration of data is an active research field in the computational sciences. However, as new technologies collect large amounts of genomic information in a near real-time fashion, the storage and exchange of data streams will continue to challenge a new generation of researchers. Therefore, important questions in database design will need to be addressed. The inclusion of different data types and the communication with other very large databases will be one of the most important challenges for an integrated initiative toward the understanding of complex biological systems.

Acknowledgments

We thank Dr. Eric M. Blalock at the University of Kentucky, Department of Molecular and Biomedical Pharmacology, for providing insightful suggestions and editorial comments.

References

[1] Stears, R. L., T. Martinsky, and M. Schena, "Trends in Microarray Analysis," *Nat. Med.*, Vol. 9, No. 1, 2003, pp. 140–145.

[2] Lim, L. P., et al., "Microarray Analysis Shows That Some MicroRNAs Downregulate Large Numbers of Target mRNAs," *Nature*, Vol. 433, No. 7027, 2005, pp. 769–773.

[3] Shingara, J., et al., "An Optimized Isolation and Labeling Platform for Accurate MicroRNA Expression Profiling," *RNA*, Vol. 11, No. 9, 2005, pp. 1461–1470.

[4] Rui, H., and M. J. Lebaron, "Creating Tissue Microarrays by Cutting-Edge Matrix Assembly," *Expert Rev. Med. Devices*, Vol. 2, No. 6, 2005, pp. 673–680.

[5] Warford, A., "Tissue Microarrays: Fast-Tracking Protein Expression at the Cellular Level," *Expert Rev. Proteomics*, Vol. 1, No. 3, 2004, pp. 283–292.

[6] Manley, S., et al., "Relational Database Structure to Manage High-Density Tissue Microarray Data and Images for Pathology Studies Focusing on Clinical Outcome: The Prostate Specialized Program of Research Excellence Model," *Am. J. Pathol.*, Vol. 159, No. 3, 2001, pp. 837–843.

[7] Maziarz, M., et al., "Integrating Global Proteomic and Genomic Expression Profiles Generated from Islet Alpha Cells: Opportunities and Challenges to Deriving Reliable Biological Inferences," *Mol. Cell Proteomics*, Vol. 4, No. 4, 2005, pp. 458–474.

[8] Cutler, D. J., et al., "High-Throughput Variation Detection and Genotyping Using Microarrays," *Genome Res.*, Vol. 11, No. 11, 2001, p. 1913–1925.

[9] Kaneta, Y., et al., "Prediction of Sensitivity to STI571 Among Chronic Myeloid Leukemia Patients by Genome-Wide cDNA Microarray Analysis," *Jpn. J. Cancer Res.*, Vol. 93, No. 8, 2002, pp. 849–856.

[10] Pan, J. Z., R. Jornsten, and R. P. Hart, "Screening Anti-Inflammatory Compounds in Injured Spinal Cord with Microarrays: A Comparison of Bioinformatics Analysis Approaches," *Physiol. Genomics*, Vol. 17, No. 2, 2004, pp. 201–214.

[11] Huopaniemi, L., et al., "Diazepam-Induced Adaptive Plasticity Revealed by Alpha1 GABAA Receptor-Specific Expression Profiling," *J. Neurochem.*, Vol. 88, No. 5, 2004, pp. 1059–1067.

[12] Page, G. P., et al., "A Design and Statistical Perspective on Microarray Gene Expression Studies in Nutrition: The Need for Playful Creativity and Scientific Hard-Mindedness," *Nutrition*, Vol. 19, No. 11-12, 2003, pp. 997–1000.

[13] Striebel, H. M., et al., "Virus Diagnostics on Microarrays," *Curr. Pharm. Biotechnol.*, Vol. 4, No. 6, 2003, pp. 401–415.

[14] Li, J., S. Chen, and D. H. Evans, "Typing and Subtyping Influenza Virus Using DNA Microarrays and Multiplex Reverse Transcriptase PCR," *J. Clin. Microbiol.*, Vol. 39, No. 2, 2001, pp. 696–704.

[15] Ellis, J. S., and M. C. Zambon, "Molecular Diagnosis of Influenza," *Rev. Med. Virol.*, Vol. 12, No. 6, 2002, pp. 375–389.

[16] Cherkasova, E., et al., "Microarray Analysis of Evolution of RNA Viruses: Evidence of Circulation of Virulent Highly Divergent Vaccine-Derived Polioviruses," *Proc. Natl. Acad. Sci. USA*, Vol. 100, No. 16, 2003, pp. 9398–9403.

[17] Cho, N. H., et al., "Genotyping of 22 Human Papillomavirus Types by DNA Chip in Korean Women: Comparison with Cytologic Diagnosis," *Am. J. Obstet. Gynecol.*, Vol. 188, No. 1, 2003, pp. 56–62.

[18] Zeeberg, B. R., et al., "GoMiner: A Resource for Biological Interpretation of Genomic and Proteomic Data," *Genome Biol.*, Vol. 4, No. 4, 2003, p. R28.

[19] Wong, C. W., et al., "Tracking the Evolution of the SARS Coronavirus Using High-Throughput, High-Density Resequencing Arrays," *Genome Res.*, Vol. 14, No. 3, 2004, pp. 398–405.

[20] Perrin, A., et al., "A Combined Oligonucleotide and Protein Microarray for the Codetection of Nucleic Acids and Antibodies Associated with Human Immunodeficiency Virus, Hepatitis B Virus, and Hepatitis C Virus Infections," *Anal. Biochem.*, Vol. 322, No. 2, 2003, pp. 148–155.

[21] Afonso, C. L., et al., "African Swine Fever Virus Multigene Family 360 and 530 Genes Affect Host Interferon Response," *J. Virol.*, Vol. 78, No. 4, 2004, pp. 1858–1864.

[22] Breitkreutz, B. J., et al., "AFM 4.0: A Toolbox for DNA Microarray Analysis," *Genome Biol.*, Vol. 2, No. 8, 2001, p. SOFTWARE0001.

[23] Schageman, J. J., et al., "MarC-V: A Spreadsheet-Based Tool for Analysis, Normalization, and Visualization of Single cDNA Microarray Experiments," *Biotechniques*, Vol. 32, No. 2, 2002, pp. 338–340, 342, 344.

[24] Anbazhagan, R., "Microarray Data Assembler," *Bioinformatics*, Vol. 19, No. 1, 2003, pp. 157–158.

[25] Golub, T. R., et al., "Molecular Classification of Cancer: Class Discovery and Class Prediction by Gene Expression Monitoring," *Science*, Vol. 286, No. 5439, 1999, pp. 531–537.

[26] Dudoit, S., and T. P. Speed, "A Score Test for the Linkage Analysis of Qualitative and Quantitative Traits Based on Identity by Descent Data from Sib-Pairs," *Biostatistics*, Vol. 1, No. 1, 2000, pp. 1–26.

[27] Getz, G., E. Levine, and E. Domany, "Coupled Two-Way Clustering Analysis of Gene Microarray Data," *Proc. Natl. Acad. Sci. USA*, Vol. 97, No. 22, 2000, pp. 12079–12084.

[28] Alizadeh, A. A., et al., "Towards a Novel Classification of Human Malignancies Based on Gene Expression Patterns," *J. Pathol.*, Vol. 195, No. 1, 2001, pp. 41–52.

[29] Troyanskaya, O. G., et al., "A Bayesian Framework for Combining Heterogeneous Data Sources for Gene Function Prediction (in Saccharomyces Cerevisiae)," *Proc. Natl. Acad. Sci. USA*, Vol. 100, No. 14, 2003, pp. 8348–8353.

[30] Li, L., et al., "Gene Selection for Sample Classification Based on Gene Expression Data: Study of Sensitivity to Choice of Parameters of the GA/KNN Method," *Bioinformatics*, Vol. 17, No. 12, 2001, pp. 1131–1142.

[31] Pavlidis, P., et al., "Learning Gene Functional Classifications from Multiple Data Types," *J. Comput. Biol.*, Vol. 9, No. 2, 2002, pp. 401–411.

[32] Olshen, A. B., and A. N. Jain, "Deriving Quantitative Conclusions from Microarray Expression Data," *Bioinformatics*, Vol. 18, No. 7, 2002, pp. 961–970.

[33] Thomas, J. G., et al., "An Efficient and Robust Statistical Modeling Approach to Discover Differentially Expressed Genes Using Genomic Expression Profiles," *Genome Res.*, Vol. 11, No. 7, 2001, pp. 1227–1236.

[34] Yan, X., et al., "Detecting Differentially Expressed Genes by Relative Entropy," *J. Theor. Biol.*, Vol. 234, No. 3, 2005, pp. 395–402.

[35] Gardiner-Garden, M., and T. G. Littlejohn, "A Comparison of Microarray Databases," *Brief Bioinform.*, Vol. 2, No. 2, 2001, pp. 143–158.

[36] Diehn, M., et al., "SOURCE: A Unified Genomic Resource of Functional Annotations, Ontologies, and Gene Expression Data," *Nucleic Acids Res.*, Vol. 31, No. 1, 2003, pp. 219–223.

[37] Ruse, C. I., et al., "Intregrated Analysis of the Human Cardiac Transcriptome, Proteome and Phosphoproteome," *Proteomics*, Vol. 4, No. 5, 2004, pp. 1505–1516.

[38] Baxevanis, A. D., "The Molecular Biology Database Collection: 2003 Update," *Nucleic Acids Res.*, Vol. 31, No. 1, 2003, pp. 1–12.

[39] Galperin, M. Y., "The Molecular Biology Database Collection: 2004 Update," *Nucleic Acids Res.*, Vol. 32 Database issue, 2004, pp. D3–D22.

[40] Schulze-Kremer, S., "Adding Semantics to Genome Databases: Towards an Ontology for Molecular Biology," *Proc. Int. Conf. Intell. Syst. Mol. Biol.*, Vol. 5, 1997, pp. 272–275.

[41] Schulze-Kremer, S., "Ontologies for Molecular Biology," *Pac. Symp. Biocomput.*, 1998, pp. 695–706.

[42] Schulze-Kremer, S., "Ontologies for Molecular Biology and Bioinformatics," *In Silico Biol.*, 2002. 2, No. 3, 2002, pp. 179–193.

[43] Ashburner, M., et al., "Gene Ontology: Tool for the Unification of Biology," The Gene Ontology Consortium, *Nat. Genet.*, Vol. 25, No. 1, 2000, pp. 25–29.

[44] Camon, E., et al., "The Gene Ontology Annotation (GOA) Database—An Integrated Resource of GO Annotations to the UniProt Knowledgebase," *In Silico Biol.*, Vol. 4, No. 1, 2004, pp. 5–6.

[45] Spellman, P. T., et al., "Design and Implementation of Microarray Gene Expression Markup Language (MAGE-ML)," *Genome Biol.*, Vol. 3, No. 9, 2002, p. RESEARCH0046.

[46] Soldatova, L. N., and R. D. King, "Are the Current Ontologies in Biology Good Ontologies?" *Nat. Biotechnol.,* Vol. 23, No. 9, 2005, pp. 1095–1098.

[47] Bammler, T., et al., "Standardizing Global Gene Expression Analysis Between Laboratories and Across Platforms," *Nat. Methods*, Vol. 2, No. 5, 2005, pp. 351–356.

[48] Brazma, A., K. Ikeo, and Y. Tateno, "Standardization of Microarray Experiment Data," *Tanpakushitsu Kakusan Koso*, Vol. 48, No. 3, 2003, pp. 280–285.

[49] Bao, W., et al., "A Database for Tracking Toxicogenomic Samples and Procedures," *Reprod. Toxicol.*, Vol. 19, No. 3, 2005, p. 411–419.

[50] Knudsen, T. B., and G. P. Daston, "MIAME Guidelines," *Reprod. Toxicol.*, Vol. 19, No. 3, 2005, p. 263.

[51] Fostel, J., et al., "Chemical Effects in Biological Systems—Data Dictionary (CEBS-DD): A Compendium of Terms for the Capture and Integration of Biological Study Design Description, Conventional Phenotypes, and 'Omics Data," *Toxicol. Sci.*, Vol. 88, No. 2, 2005, pp. 585–601.

[52] Lindon, J. C., et al., "Summary Recommendations for Standardization and Reporting of Metabolic Analyses," *Nat. Biotechnol.*, Vol. 23, No. 7, 2005, pp. 833–838.

[53] Mattes, W. B., et al., "Database Development in Toxicogenomics: Issues and Efforts," *Environ. Health Perspect.*, Vol. 112, No. 4, 2004, pp. 495–505.

[54] Saal, L. H., et al., "BioArray Software Environment (BASE): A Platform for Comprehensive Management and Analysis of Microarray Data," *Genome Biol.*, Vol. 3, No. 8, 2002, p. SOFTWARE0003.

[55] Schmeltzer, O., et al., "Building Large Knowledge Bases in Molecular Biology," *Proc. Int. Conf. Intell. Syst. Mol. Biol.*, Vol. 1, 1993, pp. 345–353.

[56] Pinciroli, F., C. Combi, and G. Pozzi, "Object-Orientated DBMS Techniques for Time-Oriented Medical Record," *Med. Inform. (Lond.)*, Vol. 17, No. 4, 1992, pp. 231–241.

[57] Dolin, R. H., "A High-Level Object-Oriented Model for Representing Relationships in an Electronic Medical Record," *Proc. Annu. Symp. Comput. Appl. Med. Care*, 1994, pp. 514–518.

[58] Harrington, J., "Recommendations for an Object Oriented Healthcare Information Model," *Stud. Health Technol. Inform.*, Vol. 6, 1993, pp. 52–59.

[59] Ghosh, D., "Object-Oriented Transcription Factors Database (ooTFD)," *Nucleic Acids Res.*, Vol. 28, No. 1, 2000, pp. 308–310.

[60] Achard, F., G. Vaysseix, and E. Barillot, "XML, Bioinformatics and Data Integration," *Bioinformatics*, Vol. 17, No. 2, 2001, pp. 115–125.

[61] Brazma, A., et al., "ArrayExpress—A Public Repository for Microarray Gene Expression Data at the EBI," *Nucleic Acids Res.*, Vol. 31, No. 1, 2003, pp. 68–71.

[62] Freier, K., et al., "Tissue Microarray Analysis Reveals Site-Specific Prevalence of Oncogene Amplifications in Head and Neck Squamous Cell Carcinoma," *Cancer Res.*, Vol. 63, No. 6, 2003, pp. 1179–1182.

[63] Ghosh, D., "Object Oriented Transcription Factors Database (ooTFD)," *Nucleic Acids Res.*, Vol. 27, No. 1, 1999, pp. 315–317.

[64] Robert, J. J., et al., "A Computational Model of Information Retrieval with UMLS," *Proc. Annu. Symp. Comput. Appl. Med. Care*, 1994, pp. 167–171.

[65] Xirasagar, S., et al., "CEBS Object Model for Systems Biology Data, SysBio-OM," *Bioinformatics*, Vol. 20, No. 13, 2004, pp. 2004–2015.

[66] Barillot, E., and F. Achard, "XML: A Lingua Franca for Science?" *Trends Biotechnol.*, Vol. 18, No. 8, 2000, pp. 331–333.

[67] Brazma, A., et al., "Minimum Information About a Microarray Experiment (MIAME)-Toward Standards for Microarray Data," *Nat. Genet.*, Vol. 29, No. 4, 2001, pp. 365–371.

[68] Brazma, A., et al., "Microarray Data Representation, Annotation and Storage," *Adv. Biochem. Eng. Biotechnol.*, Vol. 77, 2002, pp. 113–139.

[69] Wu, C. H., et al., "The Protein Information Resource: An Integrated Public Resource of Functional Annotation of Proteins," *Nucleic Acids Res.*, Vol. 30, No. 1, 2002, pp. 35–37.

[70] Barrett, T., et al., "NCBI GEO: Mining Millions of Expression Profiles—Database and Tools," *Nucleic Acids Res.*, Vol. 33 (Database issue), 2005, pp. D562–D566.

[71] Boyle, J., "Gene-Expression Omnibus Integration and Clustering Tools in SeqExpress," *Bioinformatics*, Vol. 21, No. 10, 2005, pp. 2550–2551.

[72] Fellenberg, K., et al., "Microarray Data Warehouse Allowing for Inclusion of Experiment Annotations in Statistical Analysis," *Bioinformatics*, Vol. 18, No. 3, 2002, pp. 423–433.

[73] Kasprzyk, A., et al., "EnsMart: A Generic System for Fast and Flexible Access to Biological Data," *Genome Res.*, Vol. 14, No. 1, 2004, pp. 160–169.

[74] Durinck, S., et al., "Importing MAGE-ML Format Microarray Data into BioConductor," *Bioinformatics*, Vol. 20, No. 18, 2004, pp. 3641–3642.

[75] Durinck, S., et al., "BioMart and Bioconductor: A Powerful Link Between Biological Databases and Microarray Data Analysis," *Bioinformatics*, Vol. 21, No. 16, 2005, pp. 3439–3440.

[76] Masseroli, M., et al., "GAAS: Gene Array Analyzer Software for Management, Analysis and Visualization of Gene Expression Data," *Bioinformatics*, Vol. 19, No. 6, 2003, pp. 774–775.

[77] Burgarella, S., et al., "MicroGen: A MIAME Compliant Web System for Microarray Experiment Information and Workflow Management," *BMC Bioinformatics,* Vol. 6, Suppl. 4, 2005, p. S6.

CHAPTER 9

Data Management in Expression-Based Proteomics

Zhong Yan, Jake Chen, Josh Heyen, Lee W. Ott, Cary Woods, Maureen A. Harrington, and Mark G. Goebl

This chapter introduces the current status of data management in expression-based proteomics studies. Proteomics studies generate a tremendous volume of data that must be processed, stored, analyzed, and interpreted. Data management is of critical importance in proteomics studies. Existing data management approaches include file management, simple database systems developed in-house, public data repositories, and integrated data management tools. The challenges that still exist include the unification of different data formats, the creation of a database that allows the capture of experimental parameters to describe proteomics experiments, and the development of more sophisticated data management tools to facilitate proteomic studies. This chapter also describes the need to develop a customized infrastructure for systematic studies of proteomics in the context of systems biology. A relational data model that allows intensive data analysis for proteomics is introduced. The data model design is based on mass spectrometry proteomics data collected from variety of experimental systems. It focuses on the analysis of processed data from mass spectrometry proteomics experiments and microarray experiments.

9.1 Background

The rapid advances in high-throughput microarray and proteomics analysis have introduced a new era of research. While microarray analysis has the ability to investigate changes in the relative levels of gene expression, proteomics has a distinct application in unraveling the levels of protein abundance, posttranslational modifications (e.g., phosphorylation), and protein-protein interactions, which are the formative drive in a cell. Proteomics creates opportunities to identify target proteins that are differentially regulated under different conditions, for example, in health and disease. It helps biologists elucidating the dynamics of signaling and regulatory networks important in disease progression, which will ultimately lead to the development of novel strategies for the treatment of human disease.

Proteomics, in general, falls into two categories: expression proteomics and cell-mapping proteomics. Cell-mapping proteomics studies how proteins interact with each other. Expression proteomics focuses on global changes in protein expression: what proteins are expressed under a specific condition and at a certain time [1, 2]. Since protein expression is dynamically controlled, the quantitative information is also important to address the function of proteins. Different labeling methods have been developed for quantitative proteomics in recent years [3–5], for example, isotope coded affinity tags (ICAT) and the cultured isotope tags (BISCUIT) method. Nana-scale liquid chromatography coupled with tandem mass spectrometry (nano LC/MS/MS) is one of the analytical approaches developed for quantitative proteomics [5].

Proteomic studies generate enormous volumes of data that need to be managed. In this chapter, the data management in proteomics studies will be introduced with a specific emphasis on expression-based proteomics. To enable a better understanding of the need for data management in expression-based proteomics, we will begin with an introduction of the background information about the expression-based proteomics.

Expression-based proteomic profiling is carried out in what can be considered a modular configuration where a wide variety of separation techniques, ionization sources, mass spectrometers, and analysis platforms can be combined in many different ways to optimize for specific analyses (Figure 9.1). Any global proteomic expression analysis begins with a complex mixture of either protein or peptides. This mixture can be directly introduced into a mass spectrometer and analyzed; however, the simplicity of this system is tremendously inefficient, resulting in large sample and data loss. In order to more effectively analyze the constituents of these complex mixtures, a separation system must be employed first. Methods of separa-

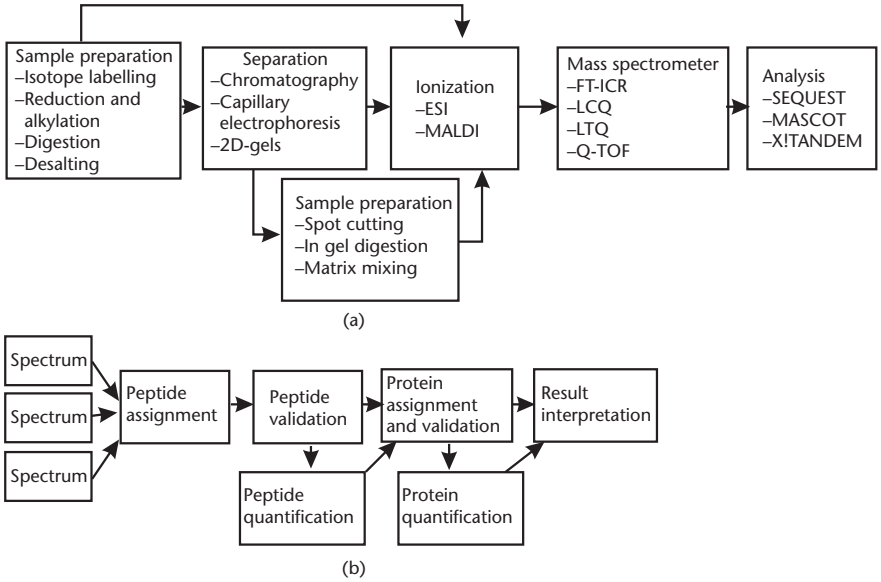


Figure 9.1 (a) Flow chart of popular MS-based proteomics data generation process. (b) Proteomics data analysis pipeline.

tion include two-dimensional gel electrophoresis [6], high performance liquid chromatography (HPLC) [7], capillary electrophoresis (CE) [8], and one day chips [9].

The separation of proteins or peptides is usually followed by their subsequent ionization. Though it is more than possible to ionize and analyze whole proteins using the appropriate system, usually proteins are digested into fragment peptides to facilitate most analyses. Ionization sources convert peptides or proteins into gas phase ions that can then be manipulated by the mass spectrometer. Two of the most popular ionization sources available include matrix assisted laser desorption ionization (MALDI) [10] and electrospray ionization (ESI) [11]. Each of these two ionization sources possesses distinctively unique qualities that are exploited to enhance analyses. Following the separation and ionization steps, the mass-to-charge ratio of each ion is measured by the mass spectrometer of choice. Mass spectrometers vary in robustness, resolution, sampling rate, mass accuracy, cost, and so on, and thus certain types of mass spectrometers are more suited for specific types of analysis. Types of mass spectrometers include the quadrupole time of flight (Q-TOF), the linear trap–Fourier transform mass spectrometer (LT-FTMS), magnetic sectors, the MALDI time of flight (MALDI TOF) [12, 13]. Despite the plethora of possible modular configurations employable, the bulk of global protein expression studies are performed utilizing two robust methodologies termed peptide mass fingerprinting [14] and "shotgun" proteomics [12]. Since these two methodologies are utilized most often, the modularity of proteomics systems will be discussed while outlining the process that these two systems encompass.

Traditionally, experiments determining global protein expression have been performed by utilizing two-dimensional gel electrophoresis as a separation strategy, MALDI as an ionization source, and a TOF or TOF-TOF as the mass spectrometer. Briefly, whole proteins are separated by 2DGE, visualized by staining, extracted from the gel, and the isolated protein(s) is then fragmented using a protease (typically trypsin). Whole proteins are first separated by mass and pI utilizing two-dimensional gel electrophoresis. The proteins are then visualized by staining them with silver stain or Coomassie [13]. At this point, relative quantification is achieved by comparing the staining intensities of conserved spots between two gels representing two different samples. There are a number of analysis tools that facilitate this function, including PD Quest, Delta 2-D, Melanie, and Z3 [15]. Each of these software programs utilizes digitized images of the 2D gel to compare spot intensities and produce various data output formats [15]. Protein identification is achieved by excising stained protein spots from the gel followed by the protein's digestion with a sequence-specific protease (usually trypsin). Digestion of the protein produces many sequence-specific peptide fragments with varying masses. This subset of masses is then analyzed simultaneously by ionizing the digested peptide mixture using either ESI or MALDI and analyzing this mixture in any number of mass spectrometer types (usually TOF or TOF-TOF instruments). The resulting data is in the form of a mass spectrum where many mass-to-charge ratios can be determined for the protein-specific peptides. The resulting mass spectra are the peptide mass fingerprints. These fingerprints, the molecular mass of the protein, and the pI of the protein are then used to search against sequence databases in which each protein sequence entry from the database is theoretically fragmented with trypsin. The best match between the experimental and the theoretical "peptide mass finger-

prints" would thus identify the protein. MASCOT (http://www.matrixscience.com) and Protein Prospector (http://www.prospector.ucsf) are just two of the programs that can carry out database searches for peptide mass fingerprinting and each employs its own proprietary data output format.

The term "shotgun" proteomics is used to define a global protein expression analysis where a complex mixture of peptides is separated by HPLC, ionized, and analyzed in a totally automated fashion [12]. Typically, whole cell lysates are first digested with a protease (usually trypsin) to attain a complex mixture of peptides. This peptide mixture can then be separated using a number of HPLC-based separation techniques. Most often one-dimensional or two-dimensional liquid chromatography is used since these methods offer high resolution separation and separated peptides can be ionized and analyzed by ESI-MS as they elute from the column [16]. Shotgun proteomics employs the use of collision-induced dissociation to generate peptide-specific fingerprints as opposed to protein-specific peptide fingerprints. As in peptide mass fingerprinting, the mass-to-charge ratio is determined for all the peptide ions that enter the mass spectrometer at a given time. Unlike peptide mass fingerprinting, peptide-specific fingerprints can be generated by isolating the peptide of interest, inducing this peptide to fragment, and collecting the mass-to-charge ratios of the peptide-specific fragments; this process is known as collision-induced dissociation or MS/MS [17]. The original mass-to-charge ratio of each ion as well its specific fragment spectrum is used to search a database of theoretical peptide fragmentation spectra often resulting in unambiguous peptide identification. The data from each of these methodologies is represented as output peak files adherent to a specific file format that is dependent on the instrument used for analysis. Programs such as SEQUEST [18] and MASCOT correlate the experimentally acquired MS/MS spectra to the computer-generated MS/MS spectra and produce various scores used to assess the validity of this correlation. Each correlation program uses different algorithms to assign peptides and thus each program produces overlapping but variable outputs. Various laboratories have used different approaches to exploit the advantages of both software algorithms [19] and to validate more thoroughly the results of these algorithms individually [20, 21]. It is apparent that no one analysis system has been universally accepted.

Quantification in shotgun proteomic systems has been performed using label free techniques [22], but most analyses still employ the use of metabolic or chemical labeling techniques [23]. In this type of analysis, the two samples to be compared are mixed before separation, and then subjected to the shotgun analysis described previously. The measurement of differences in protein expression between two samples is facilitated by labeling one of the samples with a mass tag. This mass tag can be applied in vivo using methods such as SILAC [24], or after protein or peptide extraction using ICAT [25] or GIST [26]. Utilizing one of these mass tagging technologies, peptides of the same sequence but from different samples can be discerned in the mass spectrometer, since the labeled peptide would have a greater mass. All processes described previously can be performed in this analysis, but now the full mass spectra are used over a specific time frame to generate a chromatographic elution profile for the peptide of interest and its heavy-labeled partner. Relative quantification of peptides from different samples is then achieved by comparing the integrated chromatographic elution profile of the peptide of interest and its heavy partner.

Software programs that automate peptide quantification and calculate the protein expression differences between samples include ASAPRatio [27], Xpress [28], and RelEx [29]. Each of these uses different algorithms to determine the elution profile of the peaks and calculate the relative expression changes, and thus each possesses its own subset of strengths and weaknesses. The variability inherent in these types of analysis is best exemplified at this level and stresses a need for an integrative analysis platform that can combine these results to arrive at a more accurate expression level calculation.

9.2 Proteomics Data Management Approaches

Mass spectrometry based proteomics studies produce tremendous amounts of data that need to be stored, processed, and correctly interpreted. Data management is of critical importance in proteomics studies. Currently there are four types of approaches for proteomics data management including:

1. *File management*. The files are usually stored in personal computers or archived to compact discs. The data can be easily accessed by individual researchers who keep and maintain the files. A major drawback of the file system approach is that it is very difficult to keep track of many experiments at the same time, especially when doing comparative analysis. A typical mass spectrometry experiment can generate many intermediate files and final output files. The total size of the files for just one experiment is usually very large, which can reach up to several gigabytes. It is not practical to use a file system to manage data if the data is to be manipulated and to be reanalyzed by other investigators.
2. *Simple in-house developed database systems*. Simple in-house developed database systems have the advantage of customized design to support existing workflow, but their scalability is an issue. The system may not be suitable for comprehensive analyses which may involve the integration of numerous large datasets from different biological perspectives. The maintenance of in-house database systems also relies on effective collaborations between the developer and the biologist.
3. *Public data repositories*. (See Section 9.4 for details.) Data repositories are databases used for data storage purposes. Public data repositories not only serve as sites of data storage, but also play a key role in data dissemination and exchange. The repositories also promote the development of new computational algorithms leading to new discoveries. There is a need for public proteomics repositories [30].
4. *Integrated data management tools*. (See Section 9.5 for details.) Mass spectrometry based proteomic studies generate large volumes of different types of data including gel image data, mass spectrometry spectra (peak list), and protein identification data. Simple data storage and retrieval approaches become difficult for data analysis when different types of data need to be integrated. Smart tools are urgently needed to meet the needs of the fast growing and evolving field of proteomics. Ideally these tools will

facilitate the discovery of underlying biological connections. Robust integrated proteomics data management tools will provide the ability to store large scale experimental data, mimic experimental data flow, and perform comprehensive analysis. The data management tools will need to integrate multiple modules and technologies including database, data visualization module, a statistical analysis module, and a powerful query interface.

Proteomics data management is still in its infancy. Many challenges exist in the proteomics field. These challenges are typically reflected in the following two aspects:

1. *Diverse data formats*. The different types of mass spectrometers have different designs, which produce data in different formats [31]. For example, the Micromass Q-TOF mass spectrometer is run by MassLynx software, whereas ThermoFinnigan LCQ or LTQ mass spectrometers are run by Xcalibur. The use of different software creates problems for data comparisons even within a single laboratory. The diverse spectra data formats also complicate the integration of new instruments into the existing infrastructure. In addition to the diverse data formats for mass spectrometer spectra files, the assignment of peptides to MS/MS spectra, and the inference of protein identification from lists of peptides also generate different output formats [32]. All these complexities hinder the analysis, exchange, comparison, and publication of results from different laboratories, and prevent the bioinformatics community from accessing data sets required for software development. The biologists and computer scientists must account for the differences in techniques, equipments, and data formats, and overcome the challenges of accessing and sharing large datasets. Initiatives have been launched by Human Proteomics Organization (HUPO) towards data standards in the proteomics field. (See Section 9.3 for details.)
2. *Databases integration and software infrastructure*. The integration of proteomic databases with other biological databases such as genetic and genomic databases to allow cross-database queries and perform comprehensive analysis is another major challenge of proteomics data management. Successful integration of all these databases will involve the assignment of standard terms across databases, mechanisms for periodic updates, and provisions for standardizing data acquisition across different database management systems. These needs will also require software infrastructure. Software infrastructure is recognized as a critical component and major bottleneck for current proteomics data management. While methods and infrastructures have been well established to determine the response of genes to various conditions by microarray analysis and genomic phenotyping analysis, infrastructure for proteomic data management to allow for systematic analyses is still at an early stage (see Sections 9.5 and 9.6 for details).

9.3 Data Standards in Mass Spectrometry Based Proteomics Studies

We need common standards within the proteomics field to promote data comparison, exchange, and verification. The Proteomics Standards Initiative (PSI) was founded at the Human Proteome Initiative Workshop on April 29, 2002, that was held at the National Institutes of Health, Bethesda, Maryland. The goal of PSI is to develop standards for data representation, in collaboration with the users, instrumentation and software manufacturers, journals, and database producers. So far, much effort has been made in mass spectrometry (PSI-MS) standards, protein-protein interaction (PSI-PPI) data standards, and general proteomics standards (PSI-GPS). In mass spectrometry proteomics, significant progress has been made in the last 2 years [33–38].

Due to the unique features of XML technology—which is extensible and platform neutral—the PSI mass spectrometry group is currently producing two XML-based data formats [33, 35]:

1. *mzData standard* allows the capture and interchange of peak list information by unifying the large number of current formats (pkl, dta, mgf) into one mzData. mzData has been released in a stable production version (v1.05). A detailed XML schema documentation, annotated XML schema, and some complete mzData example files are available online (http://psidev.sourceforge.net/ms/index.html). An example of an mzData file is shown in Figure 9.2. Software supporting the mzData standard are Mascot (in release 2.1), Proteios, Phenyx, and open source projects X! Tandem and The GPM. There are many other companies and organizations currently working on the implementation of mzData, such as Bruker

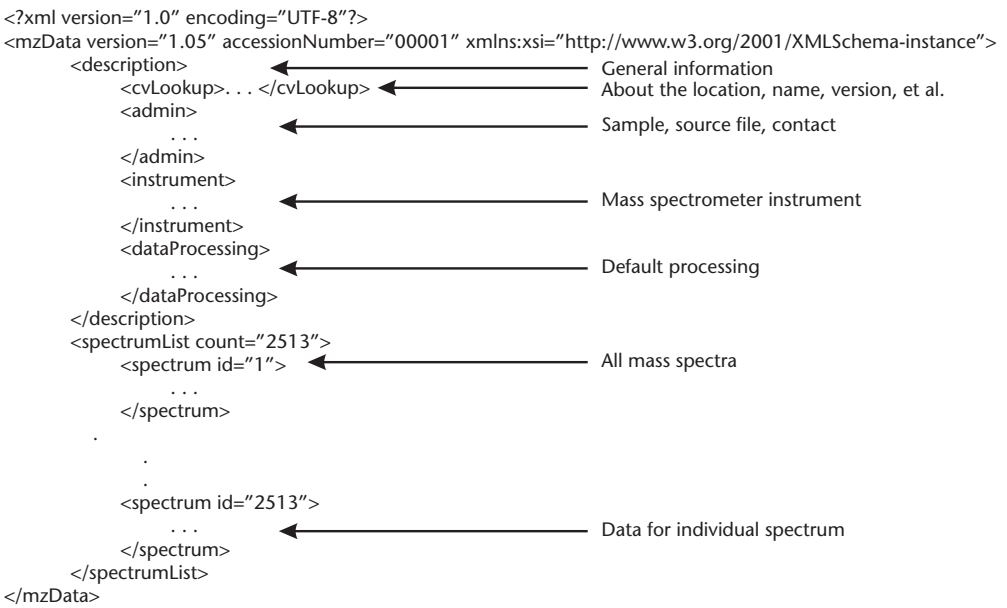


Figure 9.2 An example mzData file. This figure shows only a part of the data. "…" means subelements, and data in the terminal branches were omitted here.

Daltonics and Kratos. A number of converters are available to convert peak files generated by MS instruments into mzData standard files. The converters will be required until manufacturers have implemented the function into their software to export data directly in mzData format [33].

2. *mzIdent standard* captures parameters and results of search engines such as Mascot and Sequest. The mzIdent standard describes both protein identity and the corresponding peptides from which the identification was made. The mzIdent file can potentially be treated as an input file to a search engine since search parameters are described in mzIdent. It is still in a conceptual phase. The draft mzIdent XML schema and schema documentation are available at the Web site (http://psidev.sourceforge.net/ms/index.html).

In 2004, the Institute for Systems Biology (ISB) published another solution to standardize data formatting, mzXML (Figure 9.3), which is also an XML-based technology that provides a universal data interface between mass spectrometers and data analysis pipelines. Similar to mzData, mzXML unites the heterogeneous spectra data formats generated by different MS instruments by creating instrument-specific converters which can convert peak files into the common mzXML format [31, 32]. In a separate publication from ISB, a MS/MS analysis platform, the Trans-Proteomic Pipeline (Figure 9.4), was described. The Trans-Proteomic Pipeline uses XML-based pepXML and protXML (Figure 9.5) file standards to unite the file formats for the analysis pipeline [32].

Several pilot projects, including the liver, plasma, and brain proteome initiatives [33, 39], have already been adopted in several HUPO Proteomics Initiatives.

The pilot studies of the HUPO Brain Proteome Project (BPP) are nearly finished now (http://www.hbpp.org/). In the two pilot studies of the HUPO BPP, mouse brains of different age stages and human brain autopsy versus biopsy samples are being analyzed. One purpose is to elaborate the common data standards. Most of the participating laboratories used ProteinScape bioinformatics platform as the local database system to manage the data [39]. The data generated by the HUPO tissue initiative is available in the PRIDE mzData-compatible database (http://www.ebi.ac.uk/pride) from where it can be downloaded for subsequent reanalysis.

A future plan for data file standardization is a possible merging of HUPO-MS mzData and ISB mzXML. The merger would be designed to adopt the Functional Genomics Experimental Model (FUGE) concept. A possible merger between HUPO-MS mzIdent and pepXML/protXML from the ISB is also expected. The PSI-MS group and the ISB developers will work together to ensure the first analysis standard, analysisXML (formerly mzIdent), and later the two peak list standards can be merged [35, 40].

With several HUPO proteome pilot projects being analyzed and many other large-scale projects about to disseminate their data, it is essential to establish standards for minimum information to describe the proteomics experiments. However, the publication mechanism of these experiment results has been left behind. Descriptions of sample acquisition, preparation, and storage vary enormously. Protein identities are often confusing [41, 42].

In early 2003, the PEDRo proteomics data model was released. PEDRo is an object model described in Unified Modeling Language (UML). The PEDRo can cap-

```
<?xml version="1.0"encoding="ISO-8859-1"?>
<mzXML
xmlns="http://sashimi.sourceforge.net/schema_revision/mzXML_2.0"
xmlns:xsi="http://www.w3.org/2001/XML_Schema-instance"
xsi:schemaLocation="http://sashimi.sourceforge.net/.../mzXML_Idx_2.0.xsd">
<msRun scanCount="1232"startTime="PT5.001835"endTime="PT59.971S"
  <parentFile fileName=file://Rdf3/data2/search/ppatrick/sshimi_repository/LCQ/AD0BDB.RAW
    fileType="RAWData"fileSha1="9a930a1400d4a08ff401e00c7efa878f31cc1253"/>
  <msInstrusment>
    <msManufacturer category="msManufacturer"value="ThermoFinnigan"/>
    <msModel category="msModel"value="LCQ Deca"/>
    <mslonisation category="mslonisation"value="ESI"/>
    <msMassAnalyzer category="msMassAnalyzer"value="Ion Trap"/>
    <msDetector category="msDetector"value="EMT"/>
    <software type="acquisition"name="Xcalibur"version"1.3 alpha 8"/>
  </msInstrument>
  <dataProcessing centroided="1">
    <software type="conversion"name="Thermo2mzXML"version="1"/>
  </dataProcessing>
  <scan num="1"msLevel="1"peaksCount="0"polarity="+"retentionTime="PT300.11S"
      lowMz="400"highMz="1800"basePeakMz="0"basePeakIntensity="0"totlonCurrent="0">
      <peaks precision="32"byteOrder="network"pairOrder="m/z-int">AAAAAAAAAAA=</peaks>
  </scan>
   .
   .
   .
  <scan num="1231"msLevel="1"peaksCount="1405"polarity="+"retentionTime="PT3595.94S"
      lowMz="400"highMz="1800"basePeakMz="870.541"basePeakIntensity="2.7515e+006"totlonCurrent="2.11785e+008">
      <peaks precision="32"byteOrder="network"pairOrder="m/z-int>O8g04EgPulB...BAAAA=</peaks>
  </scan>
  <scan num="1232"msLevel="2"peaksCount="61"polarity="+"retentionTime="PT3598.26S"
      collisionEnergy="35"lowMz="224"highMz="1755"basePeakMz="795.083"basePeakIntensity="13910"totionCurrent="156025"
      <precursorMzpresursorIntensity="147170">870.54</precursorMz>
      <peaks precision="32"byteOrder="network"pairOrder="m/z-int">Q4FZDk...EScYAA=</peaks>
  </scan>
  </scan>
</msRun>
<index name="scan">
 <offset id="1">1196</offset>
  .
  .
  .
 <offset id="1232">12269539</offset>
 <index>
<indexOffset>12270666</indexOffset>
<sha1>26613fba29a69da7bf10410925111ceb2c66eecf</sha1>
</mzXML>
```

Information about MS instrument

Spectrum data

Figure 9.3 An example mzXML file. (This figure was created based on the downloaded data from http://sashimi.sourceforge.net/repository.html.)

ture parameters about mass spectrometry, chromatography, or two-dimensional gel electrophoresis. It also allows the storage of general information about samples and experiments, information about peaks, database search parameters, peptide hits, and protein hits. The data model can be divided into four sections: sample generation, sample processing, mass spectrometry, and result analysis. The release of the PEDRo data model stimulated the development of the data standards for minimum information to describe a proteomics experiment [43].

PEDRo has been accepted as the working model of PSI for protein separation based experiments. The PSI-GPS group is working on generating a document about minimum information to describe a proteomics experiment (MIAPE). MIAPE (http://psidev.sourceforge.net/gps/carros/) represents a subset of the total information available from any proteomics experiment, containing just enough information

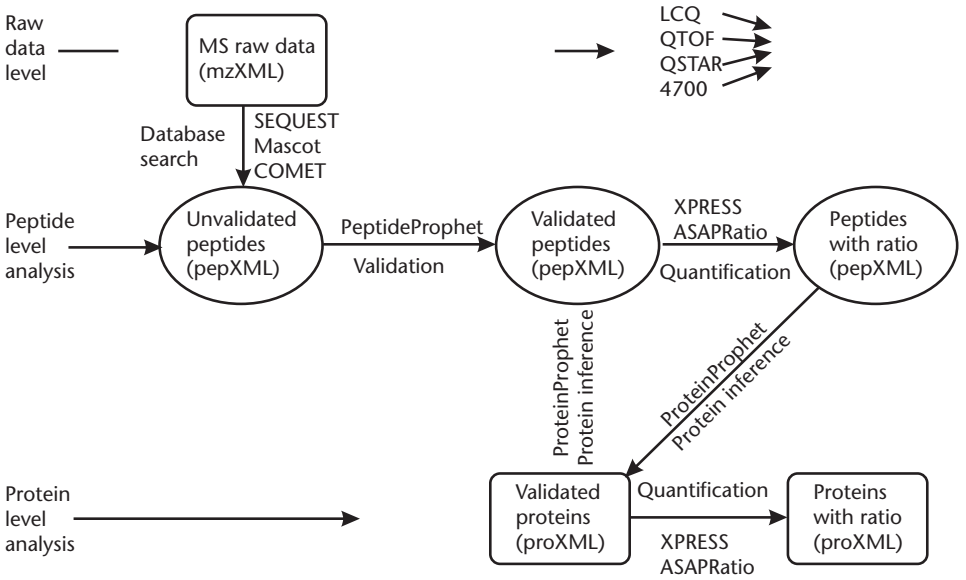


Figure 9.4 Trans-Proteomic Pipeline. (This picture was redrawn based on the picture in [5]. The literature in parenthesis is data file format for each step.)

```
<protein_group group_number="301" probability="1.00">
  <protein protein_name="IP100010154" n_indistinguishable_proteins="1" probability= "1.00" percent_coverage="7.60626">
  <annotation protein_description="&gt;IP|00010154 IPI;IP100010154.1|SWISS-PROT:P31150|REFSEQ_NP:NP_001484|
              ENSEMBLE:ENSP0000002 Tax_Id=9606 GDP dissociation inhibitor 1"/>
  <peptide peptide_sequence="SPYLYPLYGLGELPQGFAR" charge="2" initial_probability="1.00" nsp_adjusted_probability="1.00"
            weight="0.59" is_nondegenerate_evidence="N" n_tryptic_termini="2" n_sibling_peptides_bin="3" n_instances="2">
  <peptide_parent_protein protein_name="IP100031461"/>
  </peptide>
  <peptide peptide_sequence="TDDYLDQPC"LETVNR charge="2" initial_probability="1.00" nsp_adjusted_probability="0.99"
           weight="1.00" is_nondegenerate_evidence="Y" n_tryptic_termini="2" n_sibling_peptides_bin="0" n_instances="2">
  </peptide>
  </protein>
</protein_group>
```

Figure 9.5 A part of an example ProtXML file. This figure shows only a single identified protein and the associated peptides, along with the detailed information such as protein probability, annotations, and peptide sequences.

to assess the relevance of experimental methods, results, and conclusions. PSI-GPS is also working on developing a consensus object model (PSI-OM) and producing an XML format (PSI-ML) for data exchange which will be compatible with the standard data formats. The HUPO PSI-GPS and PSI-MS groups are starting to build the ontology to support all formats generated by the various PSI projects.

9.4 Public Repositories for Mass Spectrometry Data

To meet the demands of the fast growing proteomics field, building centralized public proteomics data repositories is critical. Recently, a proteomics identifications

(PRIDE) database (http://www.ebi.ac.uk/pride) has been developed jointly by the European Bioinformatics Institute at Hinxton near Cambridge, United Kingdom, and Ghent University in Belgium. The PRIDE Proteomics IDEntification database is a centralized, HUPO standards compliant, public repository for protein and peptide identifications and related evidence (e.g., example, peptide sequence, start and end location, and score). General information about the experiment such as protocols, samples, instrument, database searched, and search engine name can also be stored and retrieved. As this chapter was being written, the PRIDE contained 1,627 experiments, 178918 protein identifications, and 501296 supporting peptide identifications. The PRIDE was developed in Java 2 and J2EE, and uses many open source tools, components, and libraries. XML was chosen to form the basic data structure for the project. The infrastructure and XML schema of PRIDE can be accessed from its Web site. During the development, it used MySQL as the database management system, which was ultimately ported to Oracle. The PRIDE database, source code, and support tools are freely available either through Web access or downloading for local installation. The future development of PRIDE is closely linked to HUPO PSI [44].

Open Proteomics Database (OPD) is another public repository for storing and disseminating mass spectrometry based proteomics data (http://bioinformatics.icmb.utexas.edu/OPD/). Information stored includes detailed descriptions about the experiment, settings, sample processing, raw mass spectrometry data, raw analysis files, parameters used to generate the analysis files, and summarized report files [30]. As this chapter was being written, the database contained roughly 1.2 million spectra representing experiments from four different organisms. The OPD implementation and database schema is not publicly available from their Web site.

The Peptide Atlas Raw Data Repository contained 168 mass spectrometry experiment datasets for download (http://www.peptideatlas.org/repository/) as of July 2007. These datasets have been either published or released by the data producers. It is expected that many more datasets will become available to the public upon publication. The downloadable files for a dataset include spectra data in their native format or in mzXML format, database search result files, and ProteinProphet files. PeptideAtlas builds are a multiorganism, publicly accessible compendium of peptides identified in a large set of LC-MS/MS proteomics experiments (http://www.peptideatlas.org/builds/) [45, 46]. For example, the human PeptideAtlas contains samples from human tissues, cell lines, and fluids. The April 2005 build contains 35,391 distinct peptides above p>=0.9 from 90 samples. PeptideAtlas builds are available for download in three different file formats: FASTA format (containing PeptideAtlas peptide accession number and amino acid sequence), CDS coordinates file (containing peptides accession number and the position of the peptide relative to the protein start), and CDS and chromosomal coordinate file (containing CDS information and chromosomal location). The PeptideAtlas database schema can be found at the Web site (http://www.peptideatlas.org/dbschema.php).

The UAB Proteomics Database was developed by the University of Alabama (http://proteomics.biosccc.uab.edu/Proteomics/servlet/proteinmenu). This database focuses on annotated gel data with limited details on mass spectrometry or analysis

[47]. As this article was being written, the searching and browsing for the datasets was not supported for the public, as queries returned only SQL script.

The Sashimi Data Repository currently stores eight datasets about mass spectrometry spectra in both native format and mzXML format (http://sashimi.sourceforge.net/repository.html).

9.5 Proteomics Data Management Tools

The scale and complexity of proteomics data require software tools to facilitate data management. Compared with microarray data management tools, there are few tools available for mass spectrometry proteomics studies.

PEDRo database tool (http://pedro.man.ac.uk) is an open source tool for proteomics data entry and modeling. However, it is not a comprehensive query and analysis tool. The PEDRo tool implemented the PEDRo data model (refer to Section 9.3) which was released early in 2003. The schema of the PEDRo data model is available at the Web site. PEDRo supports an ontology service. It stores the XML directly in an open-source XML storage system, Xindice. The data are presented to the users by gathering Web pages from the stored XML using XSLT [43, 48].

SBEAMS-Proteomics (http://www.sbeams.org/Proteomics/) is one of the modules of SBEAMS integrated platform developed by ISB that is used for proteomics experimental data storage and retrieval. These experiments can be correlated later under the same framework. The integrated open source system SBEAMS adopts a relational database management system backend and a Web interface front end. Information about the quality of identification can be stored with the data; peptides which could not be properly identified from mass spectra can be flagged and reanalyzed with additional searches. The database schema for SBEAMS-Proteomics is available at the Web site (http://www.sbeams.org/Proteomics/).

ProteinScape is a commercial client-server platform for proteomics data management (http://www.bdal.com/proteinscape.html). It organizes data such as gel data, mass spectra, process parameters, and search results. It can manage gel-based or LC-based workflows, as well as quantitative proteomics. ProteinScape also enables automated analysis through interactions with database search engines such as Mascot, Phenux, and Profound. ProteinScape's relational database system can be Microsoft SQL or Oracle 9.1.

Proteios (http://www.proteios.org/) is an mzData-compliant open source client-server application that implements mass spectrometry data storage, organization, and annotation [49]. The server is a relational database that can be MySQL or Oracle, and it can utilize other alternatives. The client side runs as a Java application. One of the main objectives of Proteios is to provide a GUI that enables queries based on experiment data and annotation data. The schematic diagram is available at the Web site. Currently the input data files must be in XML format. It is working on imports of tab-separated files.

PROTICdb is a Web-based proteomics data management tool used for plant proteomics data storage and analysis [50]. The data can come from 2D-GEL and MS. The data stored can also be in the form of quantitative measurements. To support data interpretation, PROTICdb allows the integration of information from the

user's own expertise and other sources into a knowledge base. It also provides links to external databases.

ProDB is an open source tool (http://www.cebitec.uni-bielefeld.de/groups/brf/software/prodb_info/index.html) that can handle data conversion between different mass spectrometer software, automate data analysis, and allow the annotation of MS spectra (i.e., assigning gene names or storing data on protein modifications). The system is based on an extensible relational database to store the mass spectra together with the experimental setup [51]. The first release will be available to the public soon.

There are several other proteomics data management tools not described here, such as PROTEUS [52], Proteomics Rims (developed by Bruker BioSciences), Xome, and Mass Navigator [5].

9.6 Expression Proteomics in the Context of Systems Biology Studies

Proteomic expression profiling offers enormous hope for the study of gene regulation and biological pathways. With data standards established or being established, and more and more proteomics data available to the public, integrating proteomics data with microarray data and public annotation data is paramount for the systematic study of biological processes. Systems biology is an emerging cross-disciplinary science that aims to understand global cellular changes in response to mutational or conditional perturbations. As large-scale studies rely heavily on information technology to interpret complex biological systems, future trends need to emphasize the development of integrated data management tools to link the findings across mRNA, protein, and phenotypic datasets. Several initial studies have been started [53–55]. The infrastructure of these systems emphasizes generic support for mass spectrometry data processing. To provide customized support for systematic data analysis, the development of a system that can integrate multiple data sets, both from large scale experiments and public biological annotation databases, and that has a broad coverage for processed data to facilitate the analysis is needed. For this purpose, we are developing a customized infrastructure that will initially combine datasets from proteomic experiments, microarray datasets, and genotyping datasets related to cell stress. This infrastructure will allow cluster and network analyses of proteins and genes, which will ultimately lead to cell stress related systems biology discoveries. To develop this customized platform, we have initially designed a relational data model. The data model defines the grammar and vocabulary required to facilitate data exchange and knowledge discovery. The following focuses on the introduction of the data model.

The data model was designed based on several large-scale experimental datasets collected by several of the authors. It can be integrated with our existing knowledge base and visualization tool [56]. Similar to genomic data modeling [57], modeling proteomics data should consider representing intermediate and derived data—a practice frequently underemphasized by data modelers to support biological discoveries. In our data model, processed data such as paired group analysis results will be stored, since they are more meaningful from a biological point of view, and can be directly used for discovery analyses. Both single group analyses results and paired

group analyses results have been captured in the data model so that the performance will be optimized in comprehensive studies. The trade-off is faster retrieval at the cost of more storage and more expensive updating. Its implementation into an integrated proteomics data mining and analysis platform will enable us to track the presence and relative abundance of globally expressed proteins in a given condition in parallel. It will help us organize new findings based on known protein functions or related signaling pathways. It will help us identify protein functions and pathways important for a cellular condition in one organism and compare this response to that seen in other organisms. These capabilities are essential to creating new hypotheses in systems biology studies. Figure 9.6 shows the schema of the data model.

The Entity–Relationship Diagram (ERD) data modeling approach was used to describe the relational data model. The diagram uses Information Engineering notation [57]. The data model captures proteomics experimental data (blue), microarray data (yellow), genotyping data (orange), and mappings to the public annotation databases. The stored data allows comparative analysis between the groups under different experimental conditions. For example, by comparing the results between a gene knock-out sample and a wild-type sample, we may find the regulatory pathways affected by this mutation. The entity AnalysisProject holds information about the analysis goal. It is associated with the entity SingleGroupExperiment through an intermediate entity ProjGroup to reflect their many-to-many relationships. SingleGroupExperiment has two subtypes, one is for microarray studies, and the other is for proteomics studies. A few standalone tables can be created later by acquisition from public annotation databases, such as EBI Protein Identification Index (IPI), which can provide the mappings between a collection of genes and proteins. This allows for comparative studies between microarray and proteomics experiments. For example, studying the context of gene regulation: Does regulation occur at the transcriptional level? A detailed explanation about the proteomics component is given in the following (see Figure 9.7).

At the top of Figure 9.7, there are four entities created to store different mass spectrometry experimental conditions:

1. PreSamplePreparation stores chemical label identification, as well as chemical or proteolytic cleavage information specific for the sample and additional descriptions.
2. SampleInduction stores information regarding the instrument used for sample separation prior to introduction into the mass spectrometry, for example, 2DGE or a specific type of chromatography. It also stores the separation parameters associated with sample. The additional information can be stored in the field "Notes."
3. MS_Machine stores data for the mass spectrometer types, software, and parameters associated with each part of the instrument.
4. AnalysisPipeline stores information about the software pipelines used for the spectra analysis.

For some experiments, samples will be labeled and mixed before the separation process. PreSample and MSSample were created to store information regarding the

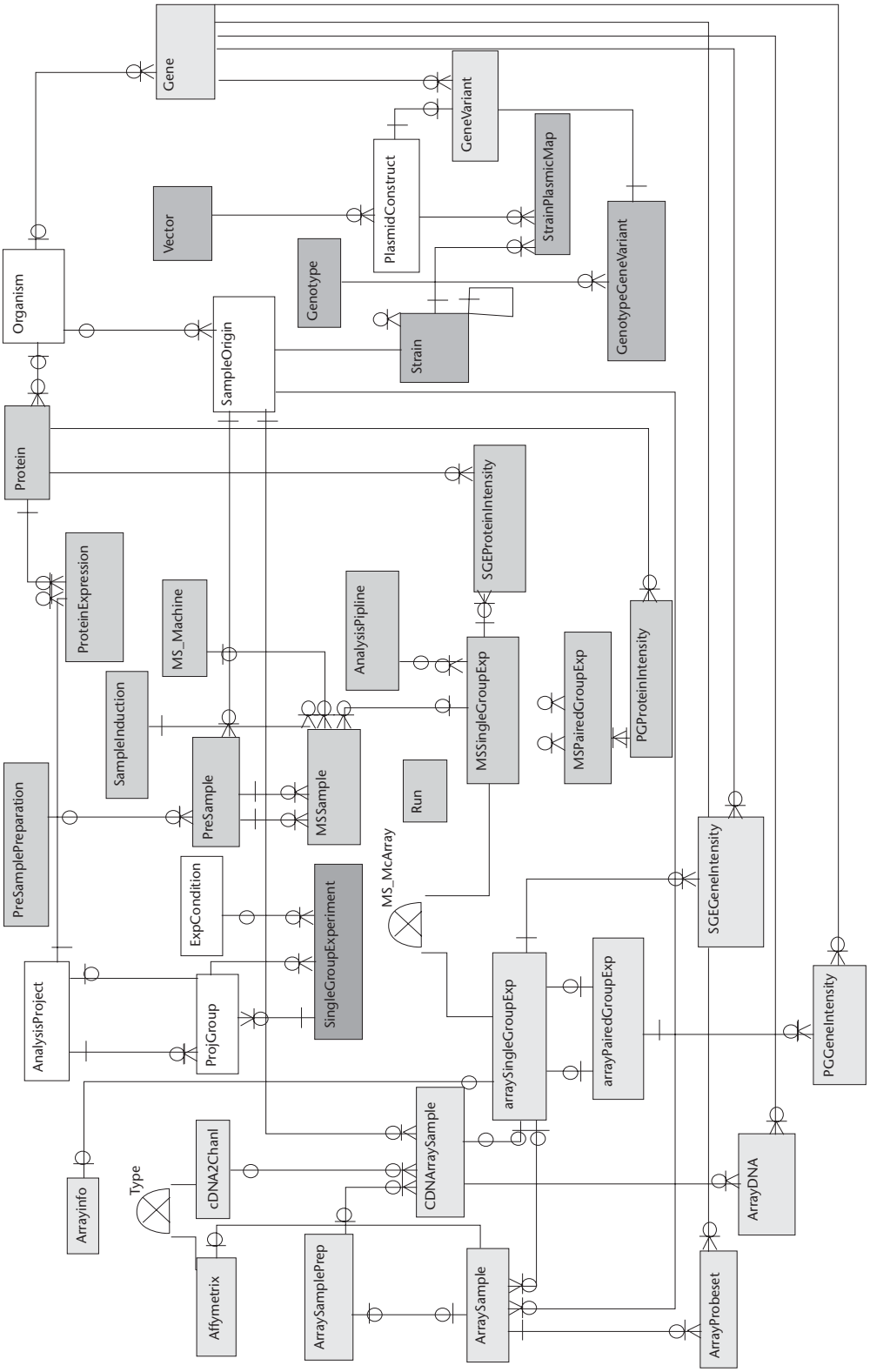


Figure 9.6 Relational data model schema. This figure shows only the entities and relationships.

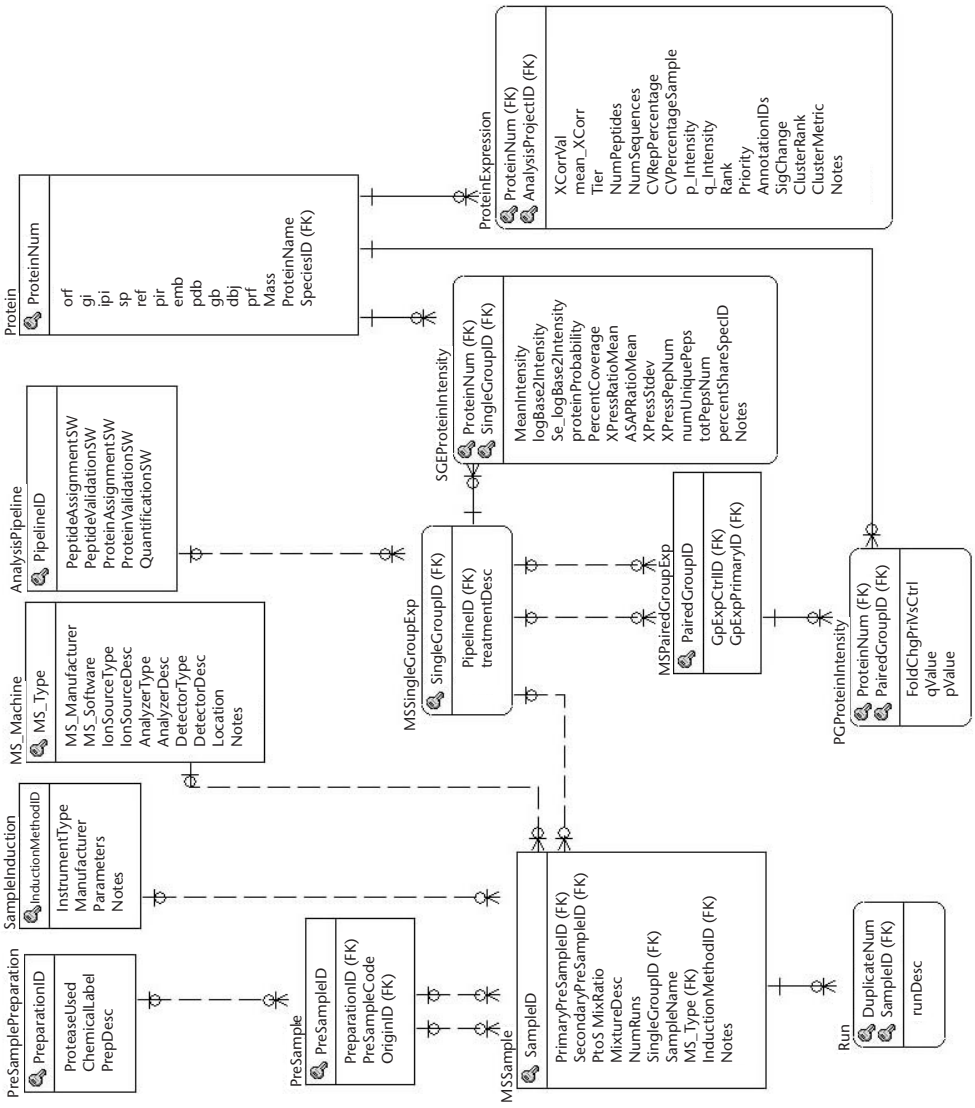


Figure 9.7 Schema for proteomics analysis.

samples before and after they are combined, respectively. If a sample is not going to be mixed with another sample before peptide separation, the field SecondaryPreSampleID can simply be set. By merging similar situations in the same set of tables, the data model can be kept compact and more efficient for future queries. Each sample may be present in multiple experiments or runs, the entity Run stores the information about this type of data.

The entities MSSingleGroupExp and MSPairedGroupExp are used to store experiment group related information, for example, group treatment descriptions. SGEProteinIntensity stores protein identification data along with the protein probabilities in the experiment group. PGProteinIntensity stores paired group comparison information, for example, protein intensity fold change between two groups and p values for statistical significance. ProteinExpression is associated with a specific analysis project, and contains the summarized information for the identified proteins for that analysis project. The latter allows for a global comparison between different analysis projects. For example, by comparing the differences of expressed proteins in different diseases, we may identify disease markers.

9.7 Protein Annotation Databases

The identification of a protein from peptides identifications derived from mass spectra has been facilitated by the annotated information extracted from protein databases, such as UniProt, Protein Information Resource (PIR), Protein Research Foundation (PRF), and Protein Data Bank (PDB).

The UniProt knowledge base consists of two sections: Swiss-Prot, a curated protein sequence database which strives to provide a high level of annotation (such as the description of the function of a protein, its domains structure, post-translational modifications, variants, and so on); and TrEMBL, a computer-annotated supplement of Swiss-Prot that contains all the translations of EMBL nucleotide sequence entries not yet integrated in Swiss-Prot. PIR is another excellent protein database but it is not well integrated with other databases. Protein sequence annotations in PIR are not as rich as those in Swiss-Prot. PRF collects information related to amino acids, peptides, and proteins. The PDB is the single worldwide repository for the processing and distribution of 3D structure data of large molecules of proteins and nucleic acids.

9.8 Conclusions

Proteomic studies routinely produce enormous amounts of data. Management of this data involves data acquisition, storage, modeling, integration, and interpretation. It is important that data standards be created to facilitate data exchange and dissemination and allow maximal acquisition of data from different experiments or laboratories. Several standards have been established and published. More standards will be established in the future. Centralized public repositories will provide a valuable mechanism for others to search for novel factors in existing analyses. However, there are limited public repositories currently available for mass spec-

trometry proteomics data. As large-scale studies rely heavily on information technology, integrated data management tools should be emphasized. In this regard, several proteomics data management tools have been developed in the last 2 years.

The future of proteomics data management will be in developing integrated tools that capture proteomics expression data, link it to gene expression data and public annotation data, which provide powerful query interface and flexible visualization choices with correct statistical summaries, and perform comprehensive analyses. Data models are essential to achieve this goal. In this chapter a relational data model focusing on integrated analysis of proteomics experimental data was introduced. The implementation of this data model into an integrated proteomics data management and analysis platform will provide opportunities for new discoveries.

References

[1] Simpson, R., *Proteins and Proteomics: A Laboratory Manual*, New York: Cold Spring Harbor Laboratory Press, 2003.

[2] Lesney, M., "Pathways to the Proteome: From 2DE to HPLC," *Modern Drug Discovery*, Vol. 4, No. 10, 2001, pp. 32–34, 36, 39.

[3] Ong, S. -E., and M. Mann, "Mass Spectrometry-Based Proteomics Turns Quantitative," *Nat. Chem. Biol.*, Vol. 1, No. 5, 2005, pp. 252–262.

[4] Bronstrup, M., "Absolute Quantification Strategies in Proteomics Based on Mass Spectrometry," *Expert Review of Proteomics*, Vol. 1, No. 4, 2004, pp. 503–512.

[5] Yanagisawa, K., et al., "Universal Proteomics Tools for Protein Quantification and Data Management -Xome & Mass Navigator," *Genome Informatics, 15th Intl. Conf. on Genome Informatics*, 2004.

[6] Monteoliva, L., and J. P. Albar, "Differential Proteomics: An Overview of Gel and Non-Gel Based Approaches," *Brief Funct. Genomic Proteomic*, Vol. 3, No. 3, 2004, pp. 220–239.

[7] Delahunty, C., and J. R. Yates, III, "Protein Identification Using 2D-LC-MS/MS," *Methods*, Vol. 35, No. 3, 2005, pp. 248–255.

[8] Simpson, D. C., and R. D. Smith, "Combining Capillary Electrophoresis with Mass Spectrometry for Applications in Proteomics," *Electrophoresis*, Vol. 26, No. 7-8, 2005, pp. 1291–1305.

[9] Schasfoort, R. B., "Proteomics-on-a-Chip: The Challenge to Couple Lab-on-a-Chip Unit Operations," *Expert Rev. Proteomics*, Vol. 1, No. 1, 2004, pp. 123–132.

[10] Tanaka, K. W., et al., "Protein and Polymer Analysis Up to m/z 100.000 by Laser Desorption Time-of-Flight Mass Spectrometry," *Rapid Commun. Mass Spectrom.*, Vol. 2, No. 151, 1988.

[11] Fenn, J. B., et al., "Electrospray Ionization for Mass-Spectrometry of Large Biomolecules," *Science*, Vol. 246, No. 4926, 1989, pp. 64–71.

[12] Wu, C. C., and M. J. MacCoss, "Shotgun Proteomics: Tools for the Analysis of Complex Biological Systems," *Curr. Opin. Mol. Ther.*, Vol. 4, No. 3, 2002, pp. 242–250.

[13] Witzmann, F. A., et al., "Gels and More Gels: Probing Toxicity," *Curr. Opin. Mol. Ther.*, Vol. 6, No. 6, 2004, pp. 608–615.

[14] Thiede, B., et al., "Peptide Mass Fingerprinting," *Methods*, Vol. 35, No. 3, 2005, pp. 237–247.

[15] Marengo, E., et al., "Numerical Approaches for Quantitative Analysis of Two-Dimensional Maps: A Review of Commercial Software and Home-Made Systems," *Proteomics*, Vol. 5, No. 3, 2005, pp. 654–666.

[16] McNulty, D. E., and J. R. Slemmon, "Peptide Proteomics," *Methods Mol. Biol.*, Vol. 244, 2004, pp. 411–423.

[17] Herbert, C. G., and R. A. W. Johnstone, *Mass Spectrometry Basics*, Boca Raton, FL: CRC Press, 2003.

[18] Yates, III, J. R., et al., "Method to Correlate Tandem Mass Spectra of Modified Peptides to Amino Acid Sequences in the Protein Database," *Anal. Chem.*, Vol. 67, No. 8, 1995, pp. 1426–1436.

[19] Resing, K. A., and N. G. Ahn, "Proteomics Strategies for Protein Identification," *FEBS Lett.*, Vol. 579, No. 4, 2005, pp. 885–889.

[20] Keller, A., et al., "Empirical Statistical Model to Estimate the Accuracy of Peptide Identifications Made by MS/MS and Database Search," *Anal. Chem.*, Vol. 74, No. 20, 2002, pp. 5383–5392.

[21] Nesvizhskii, A. I., et al., "A Statistical Model for Identifying Proteins by Tandem Mass Spectrometry," *Anal. Chem.*, Vol. 75, No. 17, 2003, pp. 4646–4658.

[22] Old, W. M., et al., "Comparison of Label-Free Methods for Quantifying Human Proteins by Shotgun Proteomics," *Mol. Cell Proteomics*, Vol. 4, No. 10, 2005, pp. 1487–1502.

[23] Guerrera, I. C., and O. Kleiner, "Application of Mass Spectrometry in Proteomics," *Biosci. Rep.*, Vol. 25, No. 1-2, 2005, pp. 71–93.

[24] Ong, S. E., et al., "Stable Isotope Labeling by Amino Acids in Cell Culture, SILAC, as a Simple and Accurate Approach to Expression Proteomics," *Mol. Cell Proteomics*, Vol. 1, No. 5, 2002, pp. 376–386.

[25] Gygi, S. P., et al., "Quantitative Analysis of Complex Protein Mixtures Using Isotope-Coded Affinity Tags," *Nat. Biotechnol.*, Vol. 17, No. 10, 1999, pp. 994–999.

[26] Chakraborty, A., and F. E. Regnier, "Global Internal Standard Technology for Comparative Proteomics," *J. Chromatogr. A*, Vol. 949, No. 1-2, 2002, pp. 173–184.

[27] Li, X. J., et al., "Automated Statistical Analysis of Protein Abundance Ratios from Data Generated by Stable-Isotope Dilution and Tandem Mass Spectrometry," *Anal. Chem.*, Vol. 75, No. 23, 2003, pp. 6648–6657.

[28] Han, D. K., et al., "Quantitative Profiling of Differentiation-Induced Microsomal Proteins Using Isotope-Coded Affinity Tags and Mass Spectrometry," *Nat. Biotechnol.*, Vol. 19, No. 10, 2001, pp. 946–951.

[29] MacCoss, M. J., et al., "A Correlation Algorithm for the Automated Quantitative Analysis of Shotgun Proteomics Data," *Anal. Chem.*, Vol. 75, No. 24, 2003, pp. 6912–6921.

[30] Prince, J. T., et al., "The Need for a Public Proteomics Repository," *Nat. Biotechnol.*, Vol. 22, No. 4, 2004, pp. 471–472.

[31] Pedrioli, P. G., et al., "A Common Open Representation of Mass Spectrometry Data and Its Application to Proteomics Research," *Nat. Biotechnol.*, Vol. 22, No. 11, 2004, pp. 1459–1466.

[32] Keller, A., et al., "A Uniform Proteomics MS/MS Analysis Platform Utilizing Open XML File Formats," *Mol. Syst. Biol.*, Vol. 1, No. 1, 2005, pp. msb4100024-E1–msb4100024-E8.

[33] Orchard, S., et al., "Further Steps Towards Data Standardisation: The Proteomic Standards Initiative HUPO 3, Annual Congress, Beijing, October 25–27, 2004," *Proteomics*, Vol. 5, No. 2, 2005, pp. 337–339.

[34] Orchard, S., et al., "Common Interchange Standards for Proteomics Data: Public Availability of Tools and Schema," *Proteomics*, Vol. 4, No. 2, 2004, pp. 490–491.

[35] Orchard, S., et al., "Further Steps in Standardisation: Report of the Second Annual Proteomics Standards Initiative Spring Workshop, Siena, Italy, April 17–20, 2005," *Proteomics*, Vol. 5, No. 14, 2005, pp. 3552–3555.

[36] Orchard, S., et al., "Current Status of Proteomic Standards Development," *Expert Rev. Proteomics*, Vol. 1, No. 2, 2004, pp. 179–183.

[37] Orchard, S., et al., "Advances in the Development of Common Interchange Standards for Proteomic Data," *Proteomics*, Vol. 4, No. 8, 2004, pp. 2363–2365.

[38] Ravichandran, V., and R. D. Sriram, "Toward Data Standards for Proteomics," *Nat. Biotechnol.*, Vol. 23, No. 3, 2005, pp. 373–376.

[39] Bluggel, M., et al., "Towards Data Management of the HUPO Human Brain Proteome Project Pilot Phase," *Proteomics*, Vol. 4, No. 8, 2004, pp. 2361–2362.

[40] "Minutes of the HUPO PSI Autumn Workshop," Geneva, Switzerland, 2005, http://paidevsourceforgenet/meetings/2005-09/reporthtml.

[41] Orchard, S., *Report from HUPO 2005 Munich*, 2005, http://wwwbio-itworldcom/newsitems/2005/sept2005/09-29-05-news-hupo?page:int=-1.

[42] Carr, S., et al., "The Need for Guidelines in Publication of Peptide and Protein Identification Data: Working Group on Publication Guidelines for Peptide and Protein Identification Data," *Mol. Cell Proteomics*, Vol. 3, No. 6, 2004, pp. 531–533.

[43] Taylor, C. F., et al., "A Systematic Approach to Modeling, Capturing, and Disseminating Proteomics Experimental Data," *Nat. Biotechnol.*, Vol. 21, No. 3, 2003, pp. 247–254.

[44] Martens, L., et al., "PRIDE: The Proteomics Identifications Database," *Proteomics*, Vol. 5, No. 13, 2005, pp. 3537–3545.

[45] Desiere, F., et al., "Integration with the Human Genome of Peptide Sequences Obtained by High-Throughput Mass Spectrometry," *Genome Biol.*, Vol. 6, No. 1, 2005, p. R9.

[46] Deutsch, E. W., et al., "Human Plasma Peptide Atlas," *Proteomics*, Vol. 5, No. 13, 2005, pp. 3497–3500.

[47] Hill, A., and H. Kim, "The UAB Proteomics Database," *Bioinformatics*, Vol. 19, No. 16, 2003, pp. 2149–2151.

[48] Garwood, K., et al., "PEDRo: A Database for Storing, Searching and Disseminating Experimental Proteomics Data," *BMC Genomics*, Vol. 5, No. 1, 2004, p. 68.

[49] Garden, P., R. Alm, and J. Hakkinen, "PROTEIOS: An Open Source Proteomics Initiative," *Bioinformatics*, Vol. 21, No. 9, 2005, pp. 2085–2087.

[50] Ferry-Dumazet, H., et al., "PROTICdb: A Web-Based Application to Store, Track, Query, and Compare Plant Proteome Data," *Proteomics*, Vol. 5, No. 8, 2005, pp. 2069–2081.

[51] Wilke, A., et al., "Bioinformatics Support for High-Throughput Proteomics," *J. Biotechnol.*, Vol. 106, No. 2-3, 2003, pp. 147–156.

[52] Cannataro, M., G. Cuda, and P. Veltri, "Modeling and Designing a Proteomics Application on PROTEUS," *Methods Inf. Med.*, Vol. 44, No. 2, 2005, pp. 221–226.

[53] Jones, A., et al., "An Object Model and Database for Functional Genomics," *Bioinformatics*, Vol. 20, No. 10, 2004, pp. 1583–90.

[54] Xirasagar, S., et al., "CEBS Object Model for Systems Biology Data, SysBio-OM," *Bioinformatics*, Vol. 20, No. 13, 2004, pp. 2004–2015.

[55] Deutsch, E. W., *Systems Biology Experiment Analysis Management System (SBEAMS) Proteomics*, 2005, http://dbsystemsbiologyorg/projects/sbeams/.

[56] Chen, J., M. Wang, and C. Shen, "An Integrated Computational Proteomics Method to Extract Protein Targets for Fanconi Anemia Studies," *Proc. of 21st Ann. ACM Symp. on Applied Computing*, Dijion, France, 2006.

[57] Chen, J., and J. Carlis, "Genomic Data Modeling," *Information Systems*, Vol. 28, No. 4, 2003, pp. 287–310.

CHAPTER 10

Model-Driven Drug Discovery: Principles and Practices

Karthik Raman, Yeturu Kalidas, and Nagasuma Chandra

The science of drug discovery has been witnessing paradigm shifts in the recent past. The new directions can be attributed to several factors such as availability of genome sequences, development of system-level models, significant growth of databases in the public domain that host functional and structural data of macromolecules, as well as adaptation and application of computational methods to biological problems.

Traditionally, computational tools to aid drug discovery have been focused on statistical or chemical models such as QSAR and SAR, used by medicinal chemists, primarily using the ligand data. Enhancement of our knowledge about the mechanisms of drug action facilitated by the in-depth understanding of the target macromolecules has led not only to the evolution of improved models, but also to the development of newer methods, shifting the focus from the lead to the target, and thus leading to the emergence of 4D/5D QSAR, structure-based design models, system-level models, and virtual cell models, some of which are still in their infancy. The promises and limitations of these methods in context of the levels of abstraction they encode are discussed in this chapter along with appropriate examples from literature. Perspectives for future development in these areas and their potential for tackling the existing challenges are presented. Tackling the problem from multiple angles, as is increasingly becoming the trend, has the advantage of leveraging the latest advances from several areas, leading to the genesis of integrated models.

10.1 Introduction

Despite the fact that the beginnings of drug discovery can be traced back to 4,000 to 5,000 years ago, with the first texts of Vedas from India or those from ancient Chinese medicine, we still do not have a "magic strategy" for designing drugs, let alone "magic bullets" or super-drugs that have most of the desirable drug properties. Drug discovery, in addition to having many uncertain phases, is also an extremely laborious and expensive process, requiring millions of dollars and about 12 to 15 years for a drug to reach the market from its initial discovery stage. This is not surprising in the least, especially when one considers the complexity of biological sys-

tems, most of which is only now beginning to be understood. Until the recent few decades, the science of drug discovery has relied heavily upon serendipity or systematic screening, with the entire system between the disease and the drug being treated as a black box (see the American Chemical Society Pharmaceutical Century Supplement at http://pubs.acs.org/journals/pharmcent/index.html for an account of the history of drug discovery). This eventually gave way to increased usage of medicinal chemistry and allied disciplines wherein the chemistry of the potential drugs were understood and exploited in great detail to design more effective molecules. In fact, there are a number of drugs in current clinical use which have emerged out of traditional medicinal chemistry approaches involving organic synthesis of potential drug candidates and pharmacological testing. The scenario, however, is changing rapidly and the science of drug discovery has witnessed several paradigm shifts [1, 2]. Many of these can be attributed to advances in molecular biology, delineation of the molecular bases of pathological processes, as well as those of drug actions in many cases, leading to a shift in the discovery focus—from a ligand (lead) to the *target* molecule (Figure 10.1).

The genomics and the postgenomics eras, with the parallel advances in high-throughput experimental methods and screening techniques to analyze whole genomes and proteomes, are witnessing an explosion in the types and amount of information available, not only with respect to the genome sequences and protein structures but also with respect to gene-expression, regulation and protein-protein

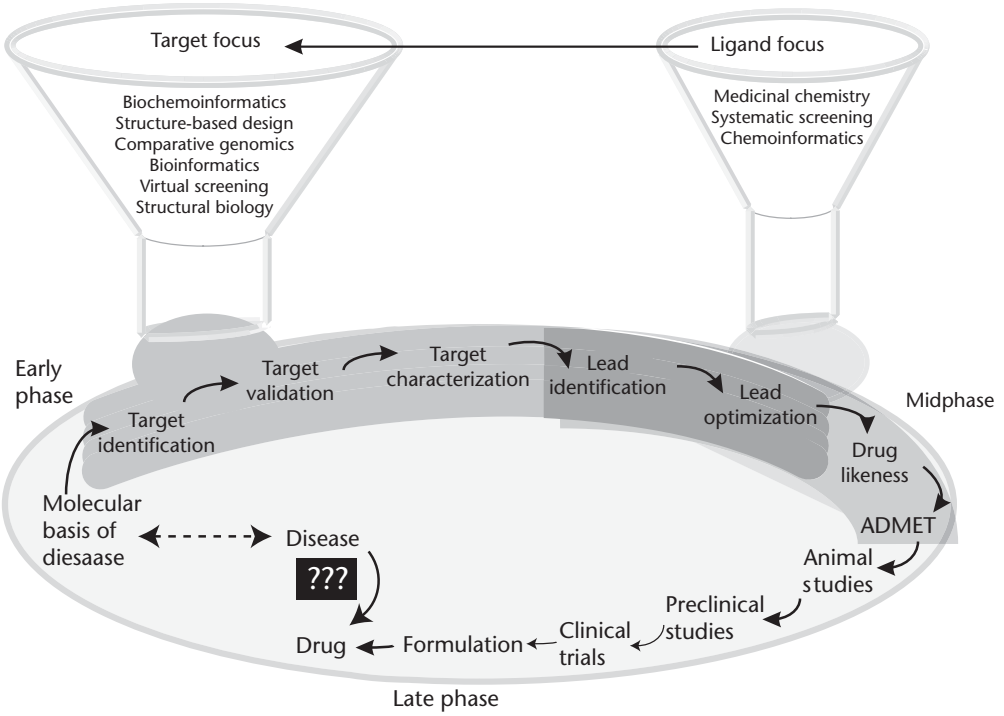


Figure 10.1 The drug discovery pipeline and processes involved. The figure also illustrates the paradigm shifts from the initial "disease to drug" practice, with most things in between being black boxes, to the more recent ligand-based approach and subsequently to the current practice of target-based approaches.

interactions. The availability of such information in publicly accessible databases (Table 10.1) and the advances in both computing power as well as in computational methods for data mining, have led to the emergence of several in silico approaches to systematically address several questions in biology, with an obvious impact on drug discovery [3, 4].

The emergence of computational models appears to have followed almost automatically, given the sheer size and complexity in content of the different types of information available. Comprehension of large volumes of complex information and its application in a higher order understanding of the biological systems has necessitated the use of systematic mathematical analyses, leading to evolution of models from ligand-driven statistical models that have been in vogue in the last decade or so, to target-enriched structural and simulation models. The growth of pathway databases and the dawn of model databases, coupled with enhancements in ligand and chemical structure databases are of immense assistance in the drug discovery process, due to the enormous predictive power they confer. This chapter provides an overview of the different types of models used in drug discovery and how they have been and can be converted into predictive models. The emphasis is on recent developments in target-based and system-based models, while the older ligand-based models are also briefly discussed.

10.2 Model Abstraction

Models are created to simulate a process or a set of processes observed in the natural world in order to gain insights into process mechanisms and predict outcomes for a given set of specific input parameters. Conceptual and theoretical modeling constructs are expressed as sets of algorithms and implemented as software packages. What constitutes a model depends upon what is understood about a given process and how best it is computationally tractable. Thus, in drug discovery, a model can refer to the relationship of the structure of a target molecule to its ability to bind a certain type of ligand at one end of the spectrum, while at the other end, it can refer to a statistically derived relationship of a set of ligands to a particular biological activity, with no explicit consideration of the mechanism or the basis of such activities. The advantages of having a model are manifold: (1) it gives the precise definition of the components of a given system (or) the genotype; (2) it allows performing simulations and monitoring the end-effect, which may be the given phenotype in this context; (3) it helps in dissecting the role of every component in the system through the analysis of perturbations; (4) it helps both in designing minimal systems that can result in a particular phenotype, as well as analyzing the effect of the addition of newer components into the framework; and (5) it is highly amenable for high-throughput simulations, useful especially in applications such as drug discovery. Thus, models not only provide significant insights into the underlying biology and application opportunities, but also enable the efficient study of what may be highly impractical, or even impossible through biochemical and molecular biology experiments. It must be pointed out, however, that a model is only as good as our understanding of what constitutes a system or how to build it. Model building is thus a critical step in in silico analysis and is often iterated and refined with valida-

tion steps. It is therefore important to understand the abstraction levels of the models, so that conclusions are drawn at appropriate levels from the analyses.

Given that biological systems and processes are understood at many different levels (Figure 10.2) and in many different aspects, it is no wonder that many different kinds of models should exist in practice. These are used at different stages of drug discovery, as will be discussed in the following sections. Figure 10.2 illustrates that models span a wide range, emanating from the organizational hierarchy in which biological systems are understood. On one hand, we have structural models at atomic levels implying certain functions, whereas on the other hand, we have whole genome-based mathematical models of either pathways or entire organisms implying functions at very different levels. We also have several statistical models, which correlate symbolic molecules or systems with generalized functions, such as those used in quantitative structure–activity relationship studies.

10.2.1 Evolution of Models

Until about two decades ago, the knowledge of the target macromolecules involved in drug action was sketchy, if it existed at all. Medicinal chemistry, on the other hand, was significantly advanced and the structure–activity models that were used extensively concerned themselves with the structure of the ligand molecule. Therefore, a majority of the effort was concentrated only on the knowledge of the ligand and its chemical modifications.

As biochemical and molecular biology experiments began to provide details about proteins involved in various diseases, it became apparent that inhibiting or modulating the activities of such proteins were useful strategies in drug design. The models, therefore, were simple and pertained to the ability of a ligand to inhibit that particular protein, based on a hypothesis that the target protein was responsible for a particular pathological process and that inhibiting it would alleviate that particular disease. Typically, the structures of a number of ligand molecules (analogs) designed over a parent scaffold were tested experimentally for inhibition of that protein. These methods were used in conjunction with biochemical and biological assays to evaluate the therapeutic potential in the context of the change in the ligand chemical structure. Various parameters, which were commonly referred to as "descriptors" of the ligand were then used to train statistical learning algorithms that could correlate individual or sets of descriptors to the inhibition capability. Such algorithms can subsequently be used for prediction of the inhibition potential for a new analogue. Since these parameters are derived quantitatively, the methods are referred to as quantitative structure–activity relationship (QSAR) or quantitative structure–property relationship (QSPR) methods, which are used extensively in the pharmaceutical industry to date. However, these models have not been adequate to aid in the design process in a number of cases, quite possibly because they consider only one part of the system—the target in these approaches is still an elusive entity and there is no direct application of any target-related information.

This scenario has changed considerably, with the growth of protein sequence databases and a concomitant development of sequence analysis methods, leading to the wide usage of "sequence-to-function" models, where the sequence of a particular protein is compared with other related sequences in databases to infer their basic

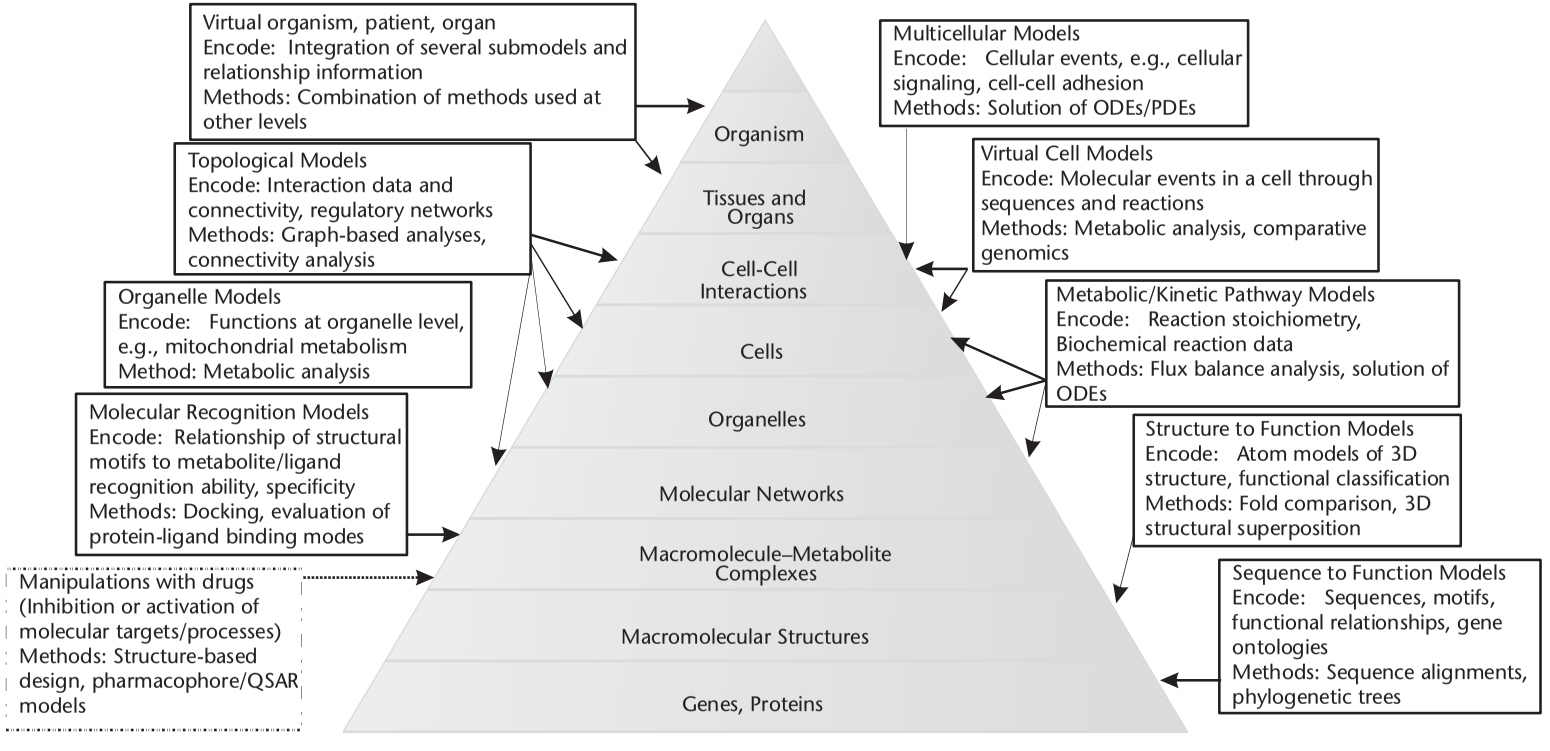


Figure10.2 Levels of hierarchies for understanding and modeling biological systems. The figure illustrates different types of models (see text for details) that are appropriate at a given level of hierarchy. The information they encode (abstraction level) are listed for each of them, as well as the methods that are in current practice to design, build, and analyze the models.

function and apply in the drug design process. With the complete sequencing of several genomes, comparative genomics became feasible, leading to direct clues about sets of proteins unique to a pathogenic organism or even virulence determinants, hence assisting in the rational identification of possible drug targets. In parallel, the successes of structural genomics projects and the overall growth of protein structure databases have: (1) enabled comparison of protein molecules at a structural level, leading to the annotation of proteins in some cases; (2) led to the understanding of structural basis for function in some other cases; and (3) led to derivation of structure–function relationships in yet other cases—"structure" here referring to the macromolecular "target" structure. Combined with the medicinal chemistry knowledge from traditional structure–activity data, this has also led to the evolution of QSAR models, resulting in 4D and 5D QSAR methodologies.

Though these QSAR models have been successful to an extent, they still fail to take into account the complete picture of the system of the host and pathogen and their interactions. Systems biology, on the other hand, attempts to take a holistic view of systems and attempts to integrate data on metabolic pathways, gene regulation, and protein–protein interactions, to build more complete and even genome-scale models of systems. Systems approaches provide a global view of drug action and can yield more reliable hypotheses for experimental evaluation.

10.2.1.1 Model Validation

Models are routinely built from a variety of sources, which vary in the degree of accuracy of experiments recorded and often depend on the interpretation of data available. Model validation is a critical quality control step that endorses the results obtained through simulation of the model. Typical model validation involves comparison of model predictions against known biochemical and genetic evidences obtained by various experiments, particularly when models are not fit to data [5–7]. Stoichiometric analyses on a network can yield conservation laws, which may be examined for feasibility. Any absurd conservation laws that emerge should point to the incompleteness and loopholes in the curated model. Model validation is thus a critical step in model building itself, where such examinations may be used to iteratively refine models.

10.3 Target Identification

Fundamental to all aspects of drug action, as for any biological process, are the molecular recognition events such as the specific binding of a substrate or a metabolite to a cellular macromolecule. Knowledge of the relevant macromolecules is a crucial requirement in the process of drug design. Two of the main components of drug discovery, therefore, are: (1) the target macromolecule of interest, which is often an enzyme or a receptor; and (2) the ligand, which is itself the molecule that has the potential to interact with the target and manipulate its function to alter the pathological process beneficially.

Advances in molecular pharmacology have led to the understanding of specific target molecules for many drugs in current clinical practice. With the availability of

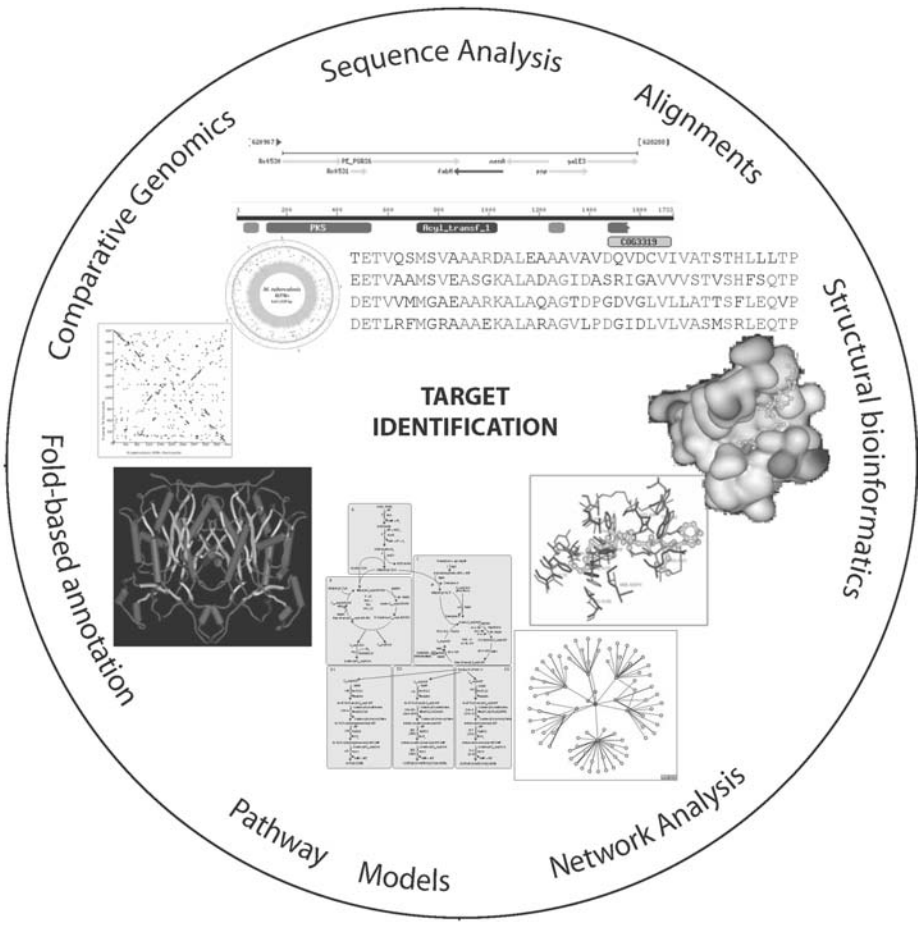


Figure 10.3 A schematic diagram representing different strategies for identifying targets. Different methods are illustrated with FabH from *M. tuberculosis*, as an example. The individual icons refer to the genome sequence of *M. tuberculosis*; its difference plot with the genome of *Escherichia coli*; the architecture of the FabH gene in the tuberculosis genome; identification of the FabH domain in another protein of *M. tuberculosis*; a multiple sequence alignment to identify key residues; the crystal structure of FabH from *M. tuberculosis*; a structural bioinformatics study with its neighbors to identify the binding site residues (a homologue bound to malonylCoA can be seen) and an analysis of its binding site (surface representation). A diagram of the mycolic acid pathway containing FabH is shown, along with an illustration of a network. Network analyses can also be very useful in target identification.

sequence and structural data that are ever-increasing and high-throughput experimental methods, it is now more realistic to identify targets on a rational basis [8, 9]. Information about sets of protein molecules involved in diseases is also being accumulated rapidly, leading to identification of potential targets that can be explored for designing more efficient and safer drugs. This section presents computational methods and concepts used for target identification, characterization, and validation. A schematic summary of the different levels at which target identification can be carried out is depicted in Figure 10.3, and these are discussed in detail in the following sections.

10.3.1 Sequence-to-Function Models

These models are based on the fundamental concepts used in bioinformatics, that of sequences containing all the necessary information to dictate the structure and more importantly the function of protein molecules. Given the facts that sequence information is much more abundant than the structure information in a number of cases and also that sequence comparison is a much simpler and faster task than structural comparison, sequences have often been used directly to gain insights into the function, by mining sequence patterns from databases and correlating them with experimentally determined functions. The term "function" can have many meanings, ranging from the basic function of binding a substrate, for example, to a larger function of being responsible for processes such as genetic recombination. Depending on the experimental data available for a given protein, the level of detail of functional annotation also varies. It is important to note that any clue at all, obtained with reasonable certainty, is helpful in identification and characterization of that protein molecule. The power of these methods has led protein characterization to become largely a data-driven science. Integrated databases and knowledge bases now exist that contain functional annotations and gene ontologies of several entire genomes [10, 11]. Some examples of such resources are presented in Table 10.1. The first functional characterizations of proteins have, in a number of instances, emerged from genome sequencing and analysis. Identification of cyclin proteins in *Plasmodium falciparum* is one such example. Three of four cyclin-like sequences identified have biochemical properties of typical cyclins, to include association with kinase activity, the latter being an attractive drug target [12].

10.3.2 Sequence Alignments and Phylogenetic Trees

Alignments are fundamental tools for analyzing protein or nucleic acid sequences. When two sequences are similar to each other over a certain threshold, they are generally considered to have similar fold and similar functions. Many sophisticated algorithms are available for aligning pairs of sequences as well as for sets of sequences. Some examples are the Smith and Waterman algorithm for pair-wise alignment [13, 14] for multiple sequence alignment, and [15] for profile-based database searching. These usually form the first steps of characterization of a protein molecule, given its sequence. The clues obtained from this can be further refined using known sequence motifs [16] and fingerprints [17], especially to analyze active sites and other functionally important regions in the proteins. Several algorithms also exist for computing phylogenetic relationships [18], which further enrich protein characterizations.

10.3.2.1 Comparative Genomics

The availability of complete genome sequences has enabled comparison of entire organisms systematically. Comparing whole genomes has several advantages, the most significant being its ability to identify proteins unique to a particular organism. Such data would be invaluable for identifying antibacterial drug targets. This method can provide information, not only on individual proteins that are unique to an organism, but also on entire pathways that may be unique to an organism [19].

Table 10.1 Examples of Online Database Resources Useful for Target-Based Drug Discovery

Sequence Resources	
NCBI Genome Database	http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?db=Genome
MicroBial Genome DataBase (MBGD)	http://mbgd.genome.ad.jp/
PEDANT	http://pedant.gsf.de/
SwissPROT	http://us.expasy.org/sprot/
Genomes OnLine Database (GOLD)	http://www.genomesonline.org/
Structure Resources	
PDB CATH SCOP Cambridge Structural Database Pathway Resources	http://www.rcsb.org/pdb/ http://cathwww.biochem.ucl.ac.uk/latest/ http://scop.mrc-lmb.cam.ac.uk/scop/ http://www.ccdc.cam.ac.uk/products/csd/
KEGG BioCyc EMP aMAZE BRENDA Pathway Hunter Tool	http://www.genome.ad.jp/kegg/ http://www.biocyc.org/ http://www.empproject.com/ http://www.amaze.ulb.ac.be/ http://www.brenda.uni-koeln.de/ http://pht.uni-koeln.de/
Protein–Protein Interaction Databases	
Database of Interacting Proteins (DIP)	http://dip.doe-mbi.ucla.edu/
Biomolecular Interaction Database (BIND)	http://bind.ca/
Modeling Tools	
E-Cell	http://www.e-cell.org/
Virtual Cell	http://www.nrcam.uchc.edu/
BioSPICE	https://www.biospice.org/
Systems Biology Workbench	http://sbw.kgi.edu/
Systems Biology Markup Language	http://www.sbml.org/
Model Resources	
Biomodels.NET	http://www.biomodels.net/
KEGG Models	http://www.systems-biology.org/001/001. html
BioCyc Models	http://genome.dfci.harvard.edu/~zucker/ BPHYS/ biocyc-open/

Some examples of unique proteins in *M. tuberculosis* are those involved in mycolic acid synthesis and the PE/PPE families of proteins [20]. Typically, these analyses are carried out by comparing a genome with all other genomes, which at present run into several hundreds (Sequence Resources from Table 10.1). At the same time, sets of proteins common to a group of organisms such as several Gram positive bacteria, can also be obtained from this type of analysis. Such analyses provide inputs that facilitate the pursuit of rational strategies for designing broad-spectrum antibiotics (e.g., [21]). It is also possible to profile several genomes simultaneously and identify proteins characteristic to a particular organism or species. Thus, this analysis can be used to generate hypotheses for identifying different types of targets.

Several high-throughput technologies have also emerged in the recent past, such as DNA microarrays, Serial Analysis of Gene Expression (SAGE), and large-scale Reverse-Transcriptase Polymerase Chain Reaction (RT-PCR). These experiments often provide a wealth of data that is complementary to that obtained by pure

sequence analyses, making their integration highly useful in target identification and validation [22]. Temporal DNA microarray data give an indication of the variation of gene expression levels with time. It is possible that a particular protein that is involved in the disease process manifests abruptly at the onset of the disease, as in case of some neurodegenerative disorders. By examining the gene expression data computationally, through clustering, or by using metrics such as entropy, it is possible to identify the corresponding genes, which may be possible drug targets [23]. Differential gene expression analysis between infected and uninfected samples may also reveal possible targets of interest. Methodologies also exist for performing systematic gene-deletions, which again provide very useful information for target identification.

10.3.2.2 Similarity Analysis with the Human Proteome

A special case of comparative genomics which is very important in target identification is the comparison of either the individual target or sets of potential targets identified from pathogenic organisms with that of the human proteome [24]. It is important that the chosen target in the pathogenic organism does not have close homologues in the human proteome, so that the drug does not exhibit unforeseen adverse effects. This is, in fact, a first-level feasibility measure of the usefulness of the identified protein as a good target and can be a step in target validation. This is particularly important in the design of antibacterial or antiviral drugs. High rates of attrition and failure at this stage, due to such adverse effects, can prove very expensive to the design process in a pharmaceutical industry.

10.3.3 Structure-to-Function Models

These models strive to capture the structure–function relationships, following the same fundamental assumptions as in the sequence models, but have the advantage of utilizing higher resolution data provided by three-dimensional structures of protein molecules. Though sequence analysis can help in identifying homologies and similarities, function can be better appreciated by analyzing protein structures. It is also often said that structure is more conserved than sequence during evolution and hence can provide better frameworks for functional annotation. It is therefore desirable to analyze molecules at a structural level, wherever appropriate data is available. Besides providing functional clues, the structures also provide a framework to understand the molecular basis of recognition, which is required both for lead design (as discussed later), as well as for analyzing the feasibility of the target molecule [25, 26].

10.3.3.1 Fold-Based Annotation

On a number of occasions, structures of the protein are determined before they are fully biochemically characterized. Structural genomics projects too often lead to the derivation of structures of proteins whose functions are yet to be identified. Owing to the growth in structure databases, a wealth of data about the functional categories that proteins of particular folds exhibit has also accumulated (Structure

Resources from Table 10.1). It is therefore possible to carry out structural comparisons and match whole or part structures to those in the database and infer functions from such similarities [27]. Many sophisticated algorithms have also emerged for this purpose. A number of instances of fold-based functional annotations are found in the literature. For example, the active site of barley endochitinase involved in plant defense against chitin-containing pests could not be identified by site-directed mutagenesis, but its fold was identified to be related to that of lysozymes from animals and phages [28] through a database search, which provided clues about its ability to hydrolyze complex carbohydrates. Recently, fold recognition methods have been applied to the prediction of the structure and catalytic mechanism of two potential drug targets, arginine succinyltransferase and succinylarginine dihydrolase [29].

10.3.3.2 Use of Homology Models

Apart from the experimentally determined protein structures, structures of a large number of proteins can be predicted using homology modeling techniques, based on the sequence–structure equations that we have come to understand. In these cases, apart from the sequences, the modeled structures can also provide significant clues about the functions of such proteins. A classic example of utilizing homology and structural information in drug discovery is the design of Captopril and its analogues, which are inhibitors of Angiotensin converting enzymes [30], for use in treating hypertension. Molecular models have also provided clues about functions that are significantly different as compared to that in their sequence or structure homologues, due to the differences in detail in the active site or other functionally important regions, despite having the same overall fold. For example, bacterial luciferase and triose phosphate isomerase have the same fold, but the differences in their fine structure lead them to have significantly different function [31]. A detailed comparison of such structural pairs provides the first clues about differences in functions. There are numerous examples where protein homology models have supported the discovery and the optimization of lead compounds with respect to potency and selectivity. One such case is the use of a homology model of Thrombin activatable fibrinolysis inhibitor (TAFI), an important regulator of fibrinolysis and hence an attractive drug target in thrombolytic therapy, in order to design appropriately substituted imidazole acetic acids, which were subsequently found to be potent and selective inhibitors of activated TAFI [32].

10.3.4 Systems-Based Approaches

Systems biology has the potential to address several important issues that arise in drug discovery such as interaction between the drug, the target, and the system as a whole, possible side effects, and causes of drug toxicity [3]. A complete knowledge of metabolic reactions in an organism helps to analyze all possible interactions between the drug and the system and helps narrow down possible causes for adverse effects and drug toxicity. Targeting multiple points in a metabolic pathway can also be a useful strategy in drug design and this is thought to be the reason for the success of several “natural” crude drugs [33]. Given the fact that cellular systems are

extremely complex, a systematic analysis of all reactions taking place in a cell across various biochemical pathways is a challenging task. With an increased popularity for systems-based approaches in biology, a wide spectrum of techniques have been applied for the simulation and analysis of biochemical systems. These include stoichiometric techniques that rely on reaction stoichiometry and other constraints, kinetic pathway modeling techniques that rely on a comprehensive mechanistic model, interaction-based analyses, Petri nets, and qualitative modeling formalisms.

10.3.4.1 Kinetic Modeling and Flux Analysis

It has been said that drug design has often not included the following basic idea: what cells do is metabolism, and a major thing drugs are supposed to do is to alter metabolism. Of the 500 well-known targets, 30% are enzymes and 45% are receptors [34]. Metabolic analysis can yield profound insights into the behavior of a cell and would be of immense importance in drug design. Eisenthal and Cornish-Bowden [35] have used pathway modeling to identify drug targets in the African trypanosome *Trypanosoma brucei*. Based on stoichiometric and kinetic analyses, they have proposed inhibition of pyruvate export as a potential strategy for inhibiting trypanosomal activity. In silico gene deletion studies help in identifying those enzymes in a metabolic network, which when deleted, adversely affect the fluxes across the entire network. Large-scale gene deletion studies for organisms such as *Saccharomyces cerevisiae* [36, 37] and *E. coli* [38] have been reported in literature. Deletions that are lethal serve as a first list of putative drug targets, which can be further characterized by sequence analyses and structural studies. Recently, we have carried out a flux balance analysis on the mycolic acid biosynthesis pathway in *M. tuberculosis*, which led to the identification of seven new possible targets, apart from InhA, an already well-known target [39]. Systems level modeling of microbial systems is on the rise [38, 40–42] and a centralized repository for such models can prove to be of great assistance in drug discovery.

10.3.4.2 Pathway Models

A pathway model is the lowest level of abstraction in system-based models. It looks at only the reactions in the metabolome of an organism and accounts for several of the interactions between the gene products of an organism and its metabolites. However, this is a significant improvement on the mere sequence data that is often employed for modeling and analysis. Several paradigms exist for pathway modeling and they are reviewed in literature [43–46]. Based on the availability of data, a suitable paradigm can be chosen for modeling; this affects the accuracy of the simulations performed on the systems.

10.3.4.3 Network-Based Analysis

Barabási and Oltvai [47] have shown that tools from network theory may be adapted to biology, providing profound insights into cellular organization and evolution. Hubs or heavily connected components in a graph may be identified and targeted to "knock out" a system. In a typical interaction-based modeling of metabolic

pathways, connections between the various proteins and metabolites in a system are obtained. When further analyzed, specific hubs emerge to be more connected; biological networks typically display a power-law degree distribution. These hubs may serve as interesting targets as they have the potential to affect several other connections in the system. The advantage of interaction-based modeling is that the amount of data required is relatively less and it is possible to generate interaction networks from existing databases [48]. There is a need for such derived databases, which would be of immense use in drug discovery.

10.3.4.4 Virtual Cells

This is the next level in the modeling hierarchy. In addition to the pathways and regulatory networks in systems, it is often very important to consider the localization of various system components (proteins/metabolites) as well as several transport processes that exist in a cell. A number of efforts have been initiated in the past few years to create models of entire cells and even entire organs, at various levels of abstraction. Metabolic models of cells, where biochemical reactions are represented mathematically and their behavior analyzed in terms of reaction kinetics or fluxes, have been developed for a number of prokaryotes [38, 40, 42] and for yeast cells [41]. Significant concepts in organ modeling are exemplified by models of cardiac cells describing cell signaling events, mitochondrial metabolism, pathways such as that of purine metabolism, flow of ionic currents, and muscle excitation and contraction. These could be further integrated into tissue-level descriptions of action potential propagations and myocardial mechanics and further with whole ventricular anatomic models and circulatory system dynamics, with links to system-level neurohumoral regulation [49]. Models of different cell types in the heart have led to the creation of the first virtual organ, which is being used in drug discovery and testing and in simulating the action of devices such as cardiac defibrillators [50].

10.3.4.5 Virtual Organisms and Patients

The culmination of systems modeling lies in the modeling of complete systems, accounting for all component reactions, the localization of these components and their interactions. The interaction between these organelles or compartments and the interface with the physical world, in terms of external temperature, pH, and other effects becomes more relevant in the final layer of hierarchy. Computational models of human physiology come into play both to relate to whole animal models used in traditional pharmacology and more importantly to build integrated data-driven models that can be refined to mimic the human physiology more closely.

The Physiome project (http://www.physiome.org/) is a project that is aimed at describing the human organism quantitatively in order to understand key elements of physiology and pathophysiology. The salient features of the project are the databasing of physiological, pharmacological, and pathological information on humans and other organisms, and integration through computational modeling. Models span a wide range, from diagrammatic schema suggesting relationships

among system components, to fully quantitative computational models describing the behavior of physiological systems and the response of an organism to environmental change. Each mathematical model is an internally self-consistent summary of available information and thereby defines a working hypothesis about how a system operates. Predictions from such models are subject to tests, with new results leading to new models. The goal is to understand the behavior of complex biological systems through a step-by-step process of building upon and refining existing knowledge [51].

Efforts are underway to extend these concepts further to virtual patients. Entelos' PhysioLab (http://www.entelos.com/) has developed models of human physiology that supplement animal model systems. For example, Entelos' Diabetes PhysioLab has more than 60 virtual patients, each one representing a hypothesis of the pathophysiology of diabetes, constrained by the pathway networks and consistent with validation experiments. Such models have the potential for performing patient profiling, classifying patient types, and even tailor-designing treatment regimes, with a long-term pharmacogenomics goal of personalized medicine.

10.3.5 Target Validation

Validation is an essential step of the target identification process, just as in any model building exercise. It is often integrated with the identification step itself. Traditionally, validation of targets has been achieved through experimental techniques such as animal experiments, gene knock-out, or site-directed mutagenesis that lead to loss-of-function phenotypes. The need for systematic and large-scale validation in the post-genomic era has led to the usage of computational methods for validation. At the sequence level, potential targets can be analyzed to assess their feasibility for manipulating their function with a drug [52]. Comparison with the human (host) proteome can be useful in filtering out those targets that have detectable homologues in the human cells, in order to reduce the risk of adverse effects. Besides overall homology, a detailed analysis of the target sequence can be performed to gain additional insights into the functional motifs and patterns. A higher-level feasibility analysis can be achieved by considering the structural information of the proteins either through experimental methods or through structure predictions in order to identify and analyze "druggable" sites in the target molecule. One such example is the study of the aurora kinases, a novel oncogenic family of mitotic serine/threoninekinases, which are over-expressed in a number of solid tumors such as in pancreatic and colorectal cancers. Molecular dynamics and docking simulations targeting the ATP binding site of aurora2 with adenylyl imidodiphosphate (AMPPNP), staurosporine, and six small molecular kinase inhibitors, identified active-site residues that interact with these inhibitors differentially, validating the choice of targets [53]. In some situations, a target molecule that has a homologue in the host, but exhibits sufficient structural differences in the functional sites may still be explored for druggability [54]. Broader insights about the appropriateness of a potential target can be obtained by considering pathways and whole-system models relevant to that disease. For example, an enzyme that may be identified as a good target for a particular disease may not actually be critical or essential when viewed in the context of the entire metabolism in the cell. Analyzing system-level models can help in

assessing criticality of the individual proteins by studying any alternate pathways and mechanisms that may naturally exist to compensate for the absence of that protein. In some other situations, especially for metabolic disorders, where the target protein is also from the human proteome, it is important to consider if inhibition of that target will lead to effects other than the intended one, owing to the involvement of that target in additional processes that may be important for maintaining normal health. System-level models will prove to be invaluable in such validations.

10.4 Lead Identification

A "lead" is generally defined as representative of a compound series with sufficient potential to progress into a full drug development program [55]. Lead compounds have been derived from diverse sources, ranging from clues from natural medicinal compounds to those specifically designed with the knowledge of target structures. From the times of total reliance on the intuition of the medicinal chemist, the methods available for lead identification have come a long way. Computational methods used widely for this purpose can be classified based on two scenarios: (1) where the target macromolecular structure is known either through experiment or through prediction; and (2) where either such structural information about a target is not available or the target molecule is not even identified. For the first class, structure-based design algorithms exist, which utilize the detailed information about the precise geometry and chemistry that exist at binding sites of protein molecules, while for the second class, the methods are predominantly based on statistical correlations between the properties of a series of ligand molecules with a testable biological activity. This section describes some of the concepts and methods used for this purpose. A schematic overview of the lead identification strategies is illustrated in Figure 10.4. Some of the well-recognized resources useful for lead identification are listed in Table 10.2.

10.4.1 Target Structure-Based Design

Knowledge of the structure of the target macromolecule facilitates computational docking of the ligand molecule into its binding site. Some examples of drugs designed by structure-based methods are Zanamivir and Oseltamivir (against influenza neuraminidase), Nelfinavir, Amprenavir, and Lopinavir (targeting HIV protease) [25]. Prior to docking, it is important to identify the binding site in the target protein, information for which is available many times through the structures of the complexes of the protein with its natural substrate. Chemical modification or site-directed mutagenesis data of the target protein can also provide clues about the binding site residues, where structures of complexes are not known. Several methods also exist for computationally predicting the binding sites—the α-shapes, Delaunay triangulation, and evolutionary trace methods being good examples [56, 57].

Docking refers to the optimal positioning of a ligand molecule with respect to the binding site of a target structure. Many methods have been developed to perform ligand docking. The simplest is the rigid-body docking, which includes clique-based searches, geometric hashing and pose clustering, which are fast but do

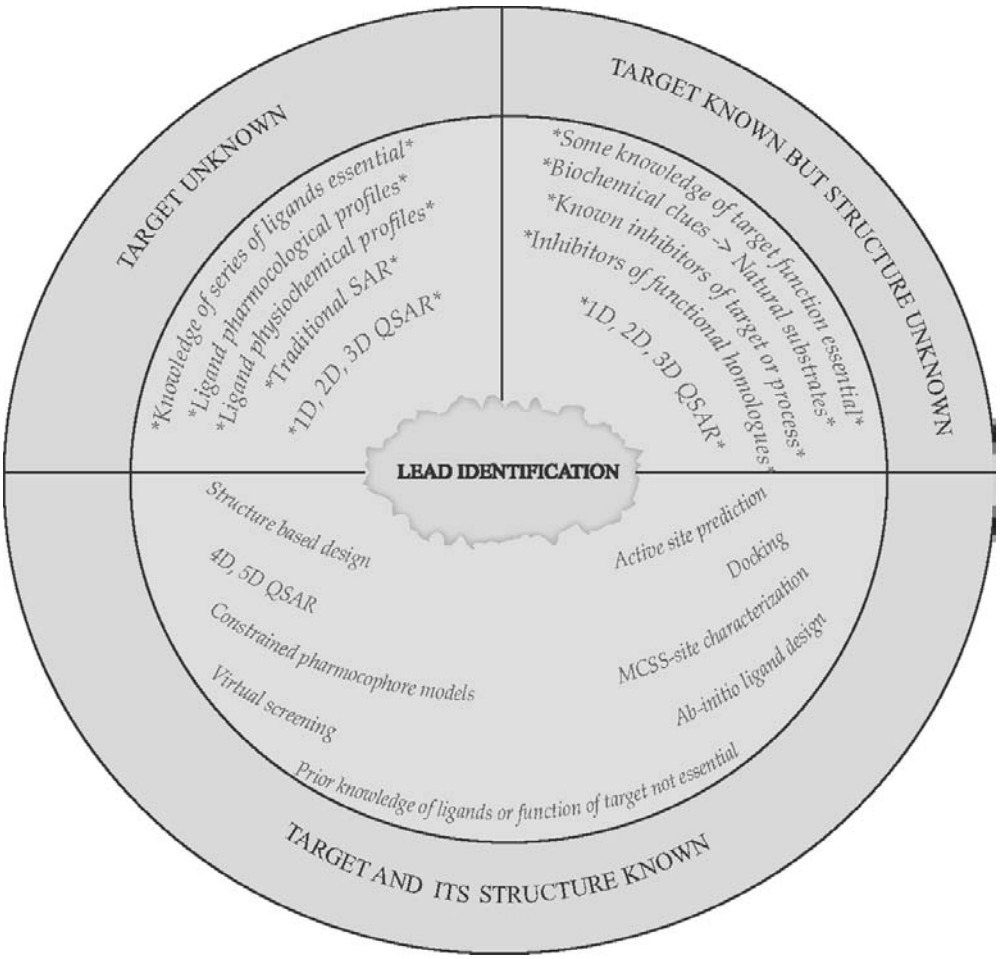


Figure 10.4 Broad strategies for lead identification in drug discovery. This figure illustrates the different approaches that can be adopted for identifying lead compounds, depending on the extent of information available for a particular case. The upper half of the figure largely refers to the ligand-based method, while the lower half indicates techniques used in target-based design. Minimal requirements in each case are also shown.

not consider the conformational flexibility in either the target or the ligand molecules. This drawback has been overcome to varying extents by the development of flexible docking methods such as place-and-join techniques, incremental construction (e.g., FlexX [58]), molecular dynamics simulations, stochastic search techniques such as simulated annealing and Monte Carlo simulations, and evolutionary algorithms (e.g., AUTODOCK [59]).The strength of binding of the ligand to the target is usually determined by considering the intermolecular energies contributed by the interaction forces arising from electrostatic, hydrogen-bond, van der Waals, and hydrophobic interactions [60, 61]. The contribution of the solvent in ligand binding can also be explicitly considered. Quantum chemical models for evaluating interac-

Table 10.2 Examples of Online Ligand-Database Resources Useful for Drug Discovery

Ligand Databases	
PubChem	http://pubchem.ncbi.nlm.nih.gov/
ChemDB	http://www.bioscreening.com/db/
KEGG LIGAND Database	http://www.genome.ad.jp/kegg/ligand.html
Daylight Fingerprints	http://www.daylight.com/
Available Chemicals Directory (ACD)	http://cds.dl.ac.uk/cds/datasets/orgchem/isis/acd.html
Ligand Structure Resources	
The Cambridge Structural Database (CSD) PDB-Ligand Maybridge Database NCI Ligand Property Resources	http://www.ccdc.cam.ac.uk/ http://www.idrtech.com/ PDB-Ligand/ http://www.maybridge.com/ http://cactus.cit.nih.gov/ncidb/
RELIBASE Hetero-compound Information Centre (HICCUP) WOMBATDatabase PRODRG2 MolInspiration ALOGPS	http://relibase.ebi.ac.uk/ http://alpha2.bmc.uu.se/hicup/ http://sunsetmolecular.com/ http://davapc1.bioch.dundee.ac. uk/programs/prodrg/ http://www.molinspiration.com/ http://vcclab.org/

tion potential are also available [62, 63]. Figure 10.5 illustrates examples of drugs designed based on the target structures.

10.4.2 Ligand-Based Models

These are also often referred to as "analog-based" models and are usually carried out when there is no adequate knowledge of the target molecules, but where the class of ligands has been shown to exhibit the required biological effect. Such effects are usually measured through animal experiments or whole-cell biochemical assays, which indicate a final pharmacological effect but do not have the resolution of information at the molecular level. Series of analogs are usually synthesized and tested for activity using the same assay method. Data from these experiments can be exploited to mine various patterns that might reflect their biological properties, as described below.

10.4.2.1 Pharmacophore Models

A pharmacophore is defined as the specific three-dimensional arrangement of functional groups within a molecular framework that are necessary to bind to a macromolecule [64]. These models capture the geometric and chemical features of those portions of the ligand, which are sufficient to relate to a given function. Pharmacophore detection is carried out by comparing a series of analogues designed over a parent scaffold and identifying common features among those that exhibit similar functional profiles. Newer molecules are either designed to fit the identified pharmacophores, or a chemical database of existing compounds is screened to identify those that contain the required pharmacophore (e.g., D2 antagonists). Pharmacophore models have been particularly useful in identifying ligands for targeting G-Protein Coupled Receptors (GPCRs) [65]. Pharmacophores can be based on shape, size, topology, and chemical nature of the fragments; they may also include three-dimensional distance constraints among the different fragments that

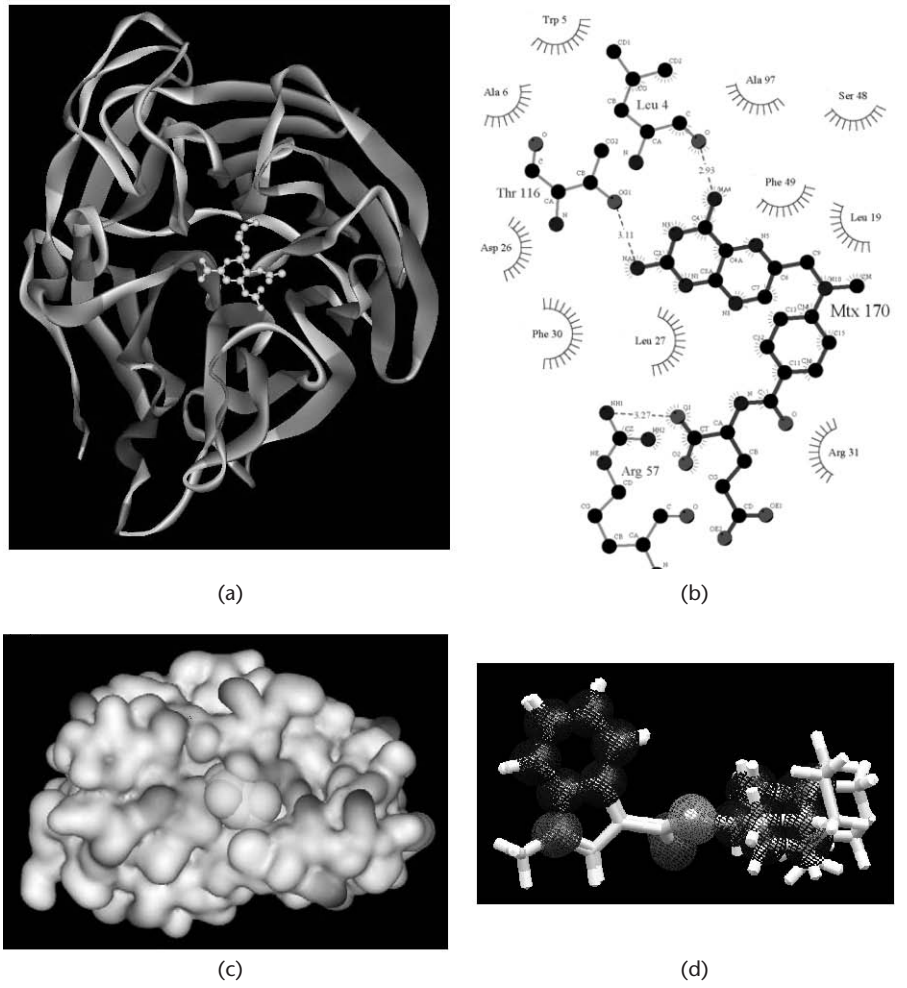


Figure 10.5 Snapshots of structures of protein-drug complexes. (a) Cartoon representation of influenza virus neuraminidase with Zanamivir (PDB:1A4G). (b) Interactions of dihyrdofolate reductase with methorexate—LIGPLOT diagram (PDB:1AO8). (c) Surface representation of HIV Protease with tripeptide (EDL) inhibitor (PDB:1A30). (d) An example of a pharmacophore model.

make up a pharmacophore, or even with the target residues [66]. A recent report describes a pharmacophore model for the main proteinase of severe acute respiratory syndrome coronavirus (SARS-CoV), which when used to search the NCI 3D database, identified 30 existing drugs containing the pharmacophore query. Six of these were found to be experimentally known SARS-CoV protease inhibitors, thus demonstrating the usefulness of the pharmacophore approach [67]. A number of methods exist to represent ligand molecules, each with its own set of merits and demerits. SMILES strings, adjacency and distance matrices, and graphs are some common examples. Based on the mathematical representation of the ligand, various methods are used for searching, such as correlation vectors, feature trees, structural keys, and fingerprints [68].

10.4.2.2 QSAR Models

QSAR models are quantitative regression methods that attempt to relate various physical and chemical properties to biological activity. QSAR models have attracted broad scientific interest particularly in the pharmaceutical industry, since it could be used without the explicit knowledge of the target molecule or even the explicit knowledge of the mechanism of action. Physical and chemical properties such as size, shape, solubility, hydrophobic, electronic, and steric properties, derived from a series of compounds can be correlated with activity through statistical methods. Recent years have seen the development of 3D QSAR models, which incorporate knowledge from the ligand structures as well as 4D and 5D QSAR, which in addition incorporate target structures and target conformational flexibility where such information is known. Many applications and review articles are available in literature, for example [69]. Comparative Molecular Field Analysis (CoMFA) and Comparative Analysis of Molecular Similarity Indices (CoMSIA) [70], where fields of different physicochemical and structural properties are computed, compared and correlated with biological activities, have also been used extensively.

10.4.2.3 Virtual Screening

A common requirement in the drug discovery process is to screen large libraries of compounds for identifying an appropriate lead compound. The libraries can be made of all available chemicals, or corporate libraries containing classified compounds or even combinatorial libraries, designed virtually. A typical library can contain over a million molecules necessitating computationally efficient methods for screening. A large variety of computational tools to select potentially active and even bio-available compounds from libraries are available. These tools range from fast filter criteria and crude molecular similarity measures to sophisticated molecular docking and scoring techniques [68, 71]. Virtual high-throughput screening (vHTS) applies in silico approaches such as docking and alignment to large virtual molecular databases, to enrich biologically active compounds in order to yield lead structures [72, 73]. Novel inhibitors targeted at the catalytic domain of ABL tyrosine kinase by using three-dimensional database searching techniques have been identified through virtual screening in order to overcome the problem of drug resistance that is beginning to be seen for the classic anticancer drug Gleevec. Two promising compounds identified exhibited significant inhibitions in further ABL tyrosine phosphorylation assays. It is anticipated that those two compounds can serve as lead compounds for further drug design and optimization [74].

10.4.2.4 Lead Optimization

Lead optimization is the process of modifying the lead compound in order to enhance properties such as specificity and bio-availability. Once a lead is identified, its closer structural analogs can be tested either for improvement in activity, or even for other practical considerations such as higher solubility, lower cost of synthesis, stability, and pharmacokinetic parameters. This can be carried out by a finer sampling of the libraries or by using more focused scoring functions. Pyrrole derivatives

with good in vitro activity against mycobacteria have been used to build a four-point pharmacophore model. Molecular modeling and SAR studies were performed to rationalize their activity in terms of superposition onto a pharmacophore model for antitubercular compounds in terms of their lipophilic properties. These studies suggest the design of new derivatives bearing the thiomorpholinomethyl moiety and different aryl substituents at N_1 and C_5, which were found to exhibit significantly enhanced activity profiles [75].

Structural information of the target molecules can also be used for lead optimization, as has been demonstrated in [76], for the design of influenza neuraminidase inhibitors. Several other examples are featured in the recent literature, some examples of which are the successes in optimization of inhibitors of several protein kinases, HIV protease [25], reverse transcriptase [77], and thymidylate synthase [78].

10.5 Lead to Drug Phase

A lead compound, to progress to a drug, usually has to traverse a long and bumpy path. In fact, several lead compounds fail at this stage, leading to a high attrition in the pipeline of drug discovery. This is because, for a compound to be a useful drug, besides exhibiting the required biological activity, it should also exhibit: (1) specificity to the target, so as to minimize the risk of adverse effects; (2) predictable dose–response relationships; (3) acceptable pharmacokinetics; (4) no undue pharmacodynamic effects with common drugs or food substances, or transformation into substances that may be toxic to the system; and (5) stability in the appropriate formulation. Besides these, there are also cost factors and other practical considerations that need to be met. The computational models that are applied in this phase of drug discovery are geared towards predicting drug-likeness of a compound and also its absorption, distribution, metabolism, elimination, and toxicity properties, collectively referred to as ADMET properties.

10.5.1 Predicting Drug-Likeness

One of the extensively used concepts to predict drug-likeness is through the analysis of the physico-chemical properties of the drug, followed by statistical correlations through the use of machine learning methods against databases of known drugs. The most well-known effort in this respect is Lipinski's rule of five, based on the observation that most orally administered drugs have (1) not more than five hydrogen bond donors, (2) not more than 10 hydrogen bond acceptors, (3) a molecular weight under 500, and (4) a LogP under five [79]. Lipinski's work has since been extended to include properties such as the number of rings and rotatable bonds.

10.5.2 ADMET Properties

Unfavorable ADMET properties have often caused the failure of leads in drug development. There is an increasing need for good predictive tools of ADMET properties to serve two key aims: first, at the design stage of new compounds and compound libraries so as to reduce the risk of late-stage attrition; and second, to optimize the

screening and testing by looking at only the most promising compounds. Several approaches exist for in silico modeling of ADMET properties and these have been reviewed in [80–83]. Two of the main classes of approaches use:

1. Data modeling and QSAR techniques, which typically use statistical tools to search for correlations between a given property and a set of molecular and structural descriptors of the molecules in question;
2. Molecular models, which use quantum mechanical methods to assess the potential for interaction between the small molecules under consideration and proteins known to be involved in ADME processes, such ascytochrome P450s.

A recent report shows a successful classification of CNS active/inactive compounds through the use of a topological substructural molecular approach. The model has also been able to identify a potential structural pharmacophore, showing its usefulness in the lead generation and optimization processes [84].

Molecular level understanding of the processes involved in the pharmacokinetics, bio-availability, and toxicity are still very poor. Development of comprehensive system-level models that encode most of the features of a system will enable a better understanding of drug toxicity and hence eliminate poor candidates early in the discovery pipeline.

10.6 Future Perspectives

The enormous progress in the development of new methods in different branches of biology and their abstraction through computational models that we are currently witnessing has already shown enormous potential, both for understanding biological processes at a better level, as well as for any application opportunities. These opportunities, which are expected to increase even more in the coming years, promise to make the mission of creating data-driven models and simulations a reality, leading to fundamental changes in the way we discover drugs. The predictive power provided by data-driven computation has long been a critical component in product development and safety testing in other industries, from aerospace engineering to circuit design.

Similarly, data-driven modeling will have a great impact in transforming biomedical research and pharmaceutical drug development into a more predictive science. In particular, substantial growth in pathway databases and a paradigm shift from annotations of genes to annotations of entire pathways or processes can be expected. The improvements in databases such as the KEGG and BioCyc indicate this trend. The new resource, Biomodels.net, which provides system models that may be exploited for various purposes, is another effort in this direction. Development of systematic methods to measure interactions between molecules and indeed between different cells will play a crucial role in the availability of reliable data at such levels and will directly influence the types and quality of the computational models that can be built. Thus, in some sense, generation and curation of higher-order data hold the key to future applications.

On the computational front, development of high performance methods for computationally intense tasks such as docking, could lead to the routine use of structure-based methods in virtual screening of millions of compounds for lead design. With the current rate of advances in systems biology, we can also expect significant enhancements in pathway models, process models, and indeed in entire system models, both in terms of mathematically representing such complex phenomena as well as in terms of mimicking and simulating the biological events. The success of the virtual heart project and the creation of virtual patients representing different pathophysiologies are suggestive of this trend. We can also envisage that the use of pharmacogenomics, and tailor-made medicines could be distinct possibilities in the near future. Systems biology approaches are also likely to impact development of molecular level pharmacokinetic and pharmacodynamic models for individual drugs, to provide comprehensive profiles of drug actions. In short, the stage is set for the integration and application of skills from mathematics and computer science to address complex problems in biology and medicine, in a big way.

Acknowledgments

The use of facilities at the Bioinformatics Centre and Interactive Graphics Based Molecular Modeling Facility (both supported by the Department of Biotechnology) and the Supercomputer Education and Research Centre are gratefully acknowledged. We also thank Preethi Rajagopalan and Nirmal Kumar for assisting in the preparation of the manuscript.

References

[1] Searls, D. B., "Data Integration: Challenges for Drug Discovery," *Nature Reviews Drug Discovery*, Vol. 4, January 2005, pp. 45–58.

[2] Reiss, T., "Drug Discovery of the Future: The Implications of the Human Genome Project," *Trends in Biotechnology*, Vol. 19, December 2001, pp. 496–499.

[3] Apic, G., et al., "Illuminating Drug Discovery with Biological Pathways," *FEBS Letters*, Vol. 579, No. 8, March 2005, pp. 1872–1877.

[4] Claus, B. L., and D. J. Underwood, "Discovery Informatics: Its Evolving Role in Drug Discovery," *Drug Discovery Today*, Vol. 7, September 2002, pp. 957–966.

[5] Hopkins, A., "Are Drugs Discovered in the Clinic or the Laboratory?" *Drug Discovery Today: TARGETS*, Vol. 3, October 2004, pp. 173–176.

[6] Chen, J. Y., and J. V. Carlis. "Genomic Data Modeling," *Information Systems*, Vol. 28, June 2003, pp. 287–310.

[7] Balant, L. P., and M. Gex-Fabry, "Modelling During Drug Development," *European Journal of Pharmaceutics and Biopharmaceutics*, Vol. 50, 2000, pp. 13–26.

[8] Roses, A. D., et al., "Disease-Specific Target Selection: A Critical First Step Down the Right Road," *Drug Discovery Today*, Vol. 10, February 2005, pp. 177–189.

[9] Cunningham, M. J., "Genomics and Proteomics: The New Millennium of Drug Discovery and Development," *Journal of Pharmacological and Toxicological Methods*, Vol. 44, July 2000, pp. 291–300.

[10] Chalker, A. F., and R. D. Lunsford, "Rational Identification of New Antibacterial Drug Targets That Are Essential for Viability Using a Genomics-Based Approach," *Pharmacology and Therapeutics*, Vol. 95, July 2002, pp. 1–20.

[11] Attwood, T. K., "The Quest to Deduce Protein Function from Sequence: The Role of Pattern Databases," *The International Journal of Biochemistry and Cell Biology*, Vol. 32, February 2000, pp. 139–155.

[12] Geyer, J. A., S. T. Prigge, and N. C. Waters, "Targeting Malaria with Specific CDK Inhibitors," *Biochimica et Biophysica Acta—Proteins and Proteomics*, Vol. 1754, No. 1–2, September 2005, pp. 160–170.

[13] Smith, T. F., and M. S. Waterman, "Identification of Common Molecular Subsequences," *Journal of Molecular Biology*, Vol. 147, 1981, pp. 195–197.

[14] Chenna, R., et al., "Multiple Sequence Alignment with the Clustal Series of Programs," *Nucleic Acids Res.*, Vol. 31, No. 13, July 2003, pp. 3497–3500.

[15] Altschul, S. F., et al., "Gapped BLAST and PSI-BLAST: A New Generation of Protein Database Search Programs," *Nucleic Acids Res.*, Vol. 25, No. 17, 1997, pp. 3389–3402.

[16] Sigrist, C. J., et al., "PROSITE: A Documented Database Using Patterns and Profiles as Motif Descriptors," *Briefings in Bioinformatics*, Vol. 3, No. 3, September 2002, pp. 265–274.

[17] Scordis, P., D. R. Flower, and T. K. Attwood, "FingerPRINTScan: Intelligent Searching of the PRINTS Motif Database," *Bioinformatics*, Vol. 15, No. 10, 1999, pp. 799–806.

[18] Kumar, S., K. Tamura, and M. Nei, "MEGA3: Integrated Software for Molecular Evolutionary Genetics Analysis and Sequence Alignment," *Briefings in Bioinformatics*, Vol. 5, No. 2, June 2004, pp. 150–163.

[19] Galperin, M. Y., and E. V. Koonin. "'Conservedhypothetical' Proteins: Prioritization of Targets for Experimental Study," *Nucleic Acids Res.*, Vol. 32, No. 18, 2004, pp. 5452–5463.

[20] Camus, J. -C., et al., "Re-Annotation of the Genome Sequence of *Mycobacterium Tuberculosis* H37Rv," *Microbiology*, Vol. 148, No. 10, 2002, pp. 2967–2973.

[21] Moir, D. T., et al., "Genomics and Antimicrobial Drug Discovery," *Antimicrobial Agents and Chemotherapy*, Vol. 43, No. 3, 1999, pp. 439–446.

[22] Bajorath, J., "Rational Drug Discovery Revisited: Interfacing Experimental Programs with Bio- and Chemo-Informatics," *Drug Discovery Today*, Vol. 6, October 2001, pp. 989–995.

[23] Fuhrman, S., et al., "Target Finding in Genomes and Proteomes," in *Bioinformatics: From Genomes to Drugs*, Vol. 14(II) of *Methods and Principles in Medicinal Chemistry*, T. Lengauer, (ed.), Weinheim, Germany: Wiley-VCH, 2002, Ch. 5, pp. 119–135.

[24] Sanseau, P., "Impact of Human Genome Sequencing for *In Silico* Target Discovery," *Drug Discovery Today*, Vol. 6, March 2001, pp. 316–323.

[25] Congreve, M., C. W. Murray, and T. L. Blundell, "Keynote Review: Structural Biology and Drug Discovery," *Drug Discovery Today*, Vol. 10, No. 13, July 2005, pp. 895–907.

[26] Hillisch, A., L. F. Pineda, and R. Hilgenfeld, "Utility of Homology Models in the Drug Discovery Process," *Drug Discovery Today*, Vol. 9, No. 15, 2004, pp. 659–669.

[27] Mizuguchi, K., "Fold Recognition for Drug Discovery," *Drug Discovery Today, TARGETS*, Vol. 3, February 2004, pp. 18–23.

[28] Holm, L., and C. Sander, "Structural Similarity Between Plant Endochitinase and Lysozymes from Animals and Phage: An Evolutionary Connection," *FEBS Letters*, Vol. 340, 1994, pp. 129–132.

[29] Shirai, H., and K. Mizuguchi, "Prediction of the Structure and Function of AstA and AstB, the First Two Enzymes of the Arginine Succinyltransferase Pathway of Arginine Catabolism," *FEBS Letters*, Vol. 555, December 2003, pp. 505–510.

[30] Ondetti, M. A., B. Rubin, and D. W. Cushman, "Design of Specific Inhibitors of Angiotensin-Converting Enzyme: New Class of Orally Active Antihypertensive Agents," *Science*, Vol. 196, No. 4288, 1977, pp. 441–444.

[31] Fisher, A. J., et al., "The 1.5-A Resolution Crystal Structure of Bacterial Luciferase in Low Salt Conditions," *Journal of Biological Chemistry*, Vol. 271, No. 36, September 1996, pp. 21956–21968.

[32] Barrow, J. C., et al., "Synthesis and Evaluation of Imidazole Acetic Acid Inhibitors of Activated Thrombin-Activatable Fibrinolysis Inhibitor as Novel Antithrombotics," *Journal of Medicinal Chemistry*, Vol. 46, November 2003, pp. 5294–5297.

[33] Csermely, P., V. Agoston, and S. Pongor, "The Efficiency of Multi-Target Drugs: The Network Approach Might Help Drug Design," *Trends in Pharmacological Sciences*, Vol. 26, 2005, pp. 178–182.

[34] Cornish-Bowden, A., and M. L. Cardenas, "Metabolic Analysis in Drug Design," *C. R. Biologies*, Vol. 326, 2003, pp. 509–515.

[35] Eisenthal, R., and A. Cornish-Bowden, "Prospects for Antiparasitic Drugs: The Case of *Trypanosoma Brucei*, the Causative Agent of African Sleeping Sickness," *The Journal of Biological Chemistry*, Vol. 273, No. 10, 1998, pp. 5500–5505.

[36] Duarte, N. C., M. J. Herrgard, and B. O. Palsson, "Reconstruction and Validation of *Saccharomyces Ccerevisiae* iND750, a Fully Compartmentalized Genome-Scale Metabolic Model," *Genome Research*, Vol. 14, 2004, pp. 1298–1309.

[37] Forster, I., et al., "Large-Scale Evaluation of *In Silico* Gene Deletions in *Saccharomyces Cerevisiae*," *OMICS*, Vol. 7, No. 2, 2003, pp. 193–202.

[38] Edwards, J. S., and B. O. Palsson, "The *Escherichia Coli* MG1655 *In Silico* Metabolic Genotype: Its Definition, Characteristics, and Capabilities," *Proc. Natl. Acad. Sci. USA*, Vol. 97, No. 10, 2000, pp. 5528–5533.

[39] Raman, K., P. Rajagopalan, and N. Chandra, "Flux Balance Analysis of Mycolic Acid Pathway: Targets for Anti-Tubercular Drugs," *PLoS Computational Biology*, Vol. 1, 2005, pp. 349–358.

[40] Becker, S. A., and B. O. Palsson, "Genome-Scale Reconstruction of the Metabolic Network in *Staphylococcus Aureus* N315: An Initial Draft to the Two-Dimensional Annotation," *BMC Microbiology*, Vol. 5, 2005, p. 8.

[41] Forster, J., et al., "Genome-Scale Reconstruction of the *Saccharomyces Cerevisiae* Metabolic Network," *Genome Research*, Vol. 13, No. 2, 2003, pp. 244–253.

[42] Edwards, J. S., and B. O. Palsson, "Systems Properties of the *Haemophilus influenzae* Rd Metabolic Genotype," *The Journal of Biological Chemistry*, Vol. 274, No. 25, 1999, pp. 17410–17416.

[43] Stelling, J., "Mathematical Models in Microbial Systems Biology," *Current Opinion in Microbiology*, Vol. 7, 2004, pp. 513–518.

[44] Wiechert, W., "Modeling and Simulation: Tools for Metabolic Engineering," *Journal of Biotechnology*, Vol. 94, 2002, pp. 37–63.

[45] Cascante, M., et al., "Metabolic Control Analysis in Drug Discovery and Disease," *Nature Biotechnology*, Vol. 20, 2002, pp. 243–249.

[46] Covert, M. W., et al., "Metabolic Modeling of Microbial Strains *In Silico*," *Trends in Biochemical Sciences*, Vol. 26, No. 3, 2001, pp. 179–186.

[47] Barabási, A. L., and Z. N. Oltvai, "Network Biology: Understanding the Cell's Functional Organization," *Nature Reviews Genetics*, Vol. 5, No. 2, February 2004, pp. 101–113.

[48] Hongwu, M., and A. -P. Zeng, "Reconstruction of Metabolic Networks from Genome Data and Analysis of Their Global Structure for Various Organisms," *Bioinformatics*, Vol. 19, No. 2, 2003, pp. 270–277.

[49] Hunter, P. J., P. Kohl, and D. Noble, "Integrative Models of the Heart: Achievements and Limitations," *Philosophical Transactions of the Royal Society of London A*, Vol. 359, 2001, pp. 1049–1054.

[50] Noble, D., "Modelling the Heart: Insights, Failures and Progress," *BioEssays*, Vol. 24, 2002, pp. 1155–1163.

[51] Bassingthwaighte, J. B., "Strategies for the Physiome Project," *Annals of Biomedical Engineering*, Vol. 28, No. 8, August 2000, pp. 1043–1058.

[52] van Duin, M., et al., "Genomics in Target and Drug Discovery," *Biochemical Society Transactions*, Vol. 31, No. 2, 2003, pp. 429–432.

[53] Vankayalapati, H., et al., "Targeting Aurora2 Kinase in Oncogenesis: A Structural Bioinformatics Approach to Target Validation and Rational Drug Design," *Molecular Cancer Therapeutics*, Vol. 2, No. 3, March 2003, pp. 283–294.

[54] An, J., M. Totrov, and R. Abagyan, "Comprehensive Identification of 'Druggable' Protein Ligand Binding Sites," *Genome Informatics Series: Workshop on Genome Informatics*, Vol. 15, 2004, pp. 31–41.

[55] Valler, M. J., and D. Green, "Diversity Screening Versus Focussed Screening in Drug Discovery," *Drug Discovery Today*, Vol. 5, No. 7, July 2000, pp. 286–293.

[56] Binkowski, T. A., S. Naghibzadeh, and J. Liang, "CASTp: Computed Atlas of Surface Topography of Proteins," *Nucleic Acids Res.*, Vol. 31, No. 13, 2003, pp. 3352–3355.

[57] Lichtarge, O., H. R. Bourne, and F. E. Cohen, "An Evolutionary Trace Method Defines Binding Surfaces Common to Protein Families," *Journal of Molecular Biology*, Vol. 257, No. 2, March 1996, pp. 342–358.

[58] Rarey, M., et al., "A Fast Flexible Docking Method Using an Incremental Construction Algorithm," *Journal of Molecular Biology*, Vol. 261, No. 3, August 1996, pp. 470–489.

[59] Morris, G. M., et al., "Automated Docking Using a Lamarckian Genetic Algorithm and an Empirical Binding Free Energy Function," *Journal of Computational Chemistry*, Vol. 19, No. 14, 1998, pp. 1639–1662.

[60] Muegge, I., and Y. C. Martin. "A General and Fast Scoring Function for Protein-Ligand Interactions: A Simplified Potential Approach," *Journal of Medicinal Chemistry*, Vol. 42, No. 5, 1999, pp. 791–804.

[61] Sobolev, V., et al., "Automated Analysis of Interatomic Contacts in Proteins," *Bioinformatics*, Vol. 15, No. 4, 1999, pp. 327–332.

[62] Xiang, Y., D. W. Zhang, and J. Z. H. Zhang. "Fully Quantum Mechanical Energy Optimization for Protein-Ligand Structure," *Journal of Computational Chemistry*, Vol. 25, No. 12, 2004, pp. 1431–1437.

[63] Zoete, V., O. Michielin, and M. Karplus, "Protein-Ligand Binding Free Energy Estimation Using Molecular Mechanics and Continuum Electrostatics. Application to HIV-1 Protease Inhibitors," *Journal of Computer-Aided Molecular Design*, Vol. 17, No. 12, December 2003, pp. 861–880.

[64] Neamati, N., and J. J. Barchi, Jr., "New Paradigms in Drug Design and Discovery," *Current Topics in Medicinal Chemistry*, Vol. 2, No. 3, 2002, pp. 211–227.

[65] Klabunde, T., and G. Hessler. "Drug Design Strategies for Targeting G-Protein-Coupled Receptors," *ChemBioChem*, Vol. 3, No. 10, 2002, pp. 928–944.

[66] Bleicher, K. H., et al., "Ligand Identification for G-Protein-Coupled Receptors: A Lead Generation Perspective," *Current Opinion in Chemical Biology*, Vol. 8, June 2004, pp. 287–296.

[67] Zhang, X. W., Y. L. Yap, and R. M. Altmeyer, "Generation of Predictive Pharmacophore Model for SARScoronavirus Main Proteinase," *European Journal of Medicinal Chemistry*, 40, No. 1, January 2005, pp. 57–62.

[68] Stahl, M., M. Rarey, and G. Klebe, "Screening of Drug Databases," in *Bioinformatics: From Genomes to Drugs*, Vol. 14(II) of *Methods and Principles in Medicinal Chemistry*, T. Lengauer, (ed.), Weinheim, Germany: Wiley-VCH, 2002, Ch. 6, pp. 137–170.

[69] Perkins, R., et al., "Quantitative Structure-Activity Relationship Methods: Perspectives on Drug Discovery and Toxicology," *Environmental Toxicology and Chemistry*, Vol. 22, No. 8, August 2003, pp. 1666–1679.

[70] Klebe, G., U. Abraham, and T. Mietzner, "Molecular Similarity Indices in a Comparative Analysis (CoMSIA) of Drug Molecules to Correlate and Predict Their Biological Activity," *Journal of Medicinal Chemistry*, Vol. 37, No. 24, 1994, pp. 4130–4146.

[71] Waszkowycz, B., et al., "Large-Scale Virtual Screening for Discovering Leads in the Postgenomic Era," *IBM Systems Journal*, Vol. 40, No. 2, 2001, pp. 360–376.

[72] Lengauer, T., et al., "Novel Technologies for Virtual Screening," *Drug Discovery Today*, Vol. 9, January 2004, pp. 27–34.

[73] Markus, H., et al., "Virtual High-Throughput *In Silico* Screening," *Drug Discovery Today Biosilico*, Vol. 1, September 2003, pp. 143–149.

[74] Peng, H., et al., "Identification of Novel Inhibitors of bcr-abl Tyrosine Kinase Via Virtual Screening," *Bioorganic and Medicinal Chemistry Letters*, Vol. 13, November 2003, pp. 3693–3699.

[75] Biava, M., et al., "Antimycobacterial Compounds: Optimization of the BM 212 Structure, the Lead Compound for a New Pyrrole Derivative Class," *Bioorganic and Medicinal Chemistry*, Vol. 13, February 2005, pp. 1221–1230.

[76] Stoll, V., et al., "Influenza Neuraminidase Inhibitors: Structure-Based Design of a Novel Inhibitor Series," *Biochemistry*, Vol. 42, No. 3, 2003, pp. 718–727.

[77] Jorgensen, W. L., et al., "Computer-Aided Design of Non-Nucleoside Inhibitors of HIV1 Reverse Transcriptase," *Bioorganic and Medicinal Chemistry Letters*, Vol. 16, No. 3, November 2006, pp. 663–667.

[78] Shiying, L., et al., "The 3DQSAR Analysis of 4(3H)-Quinazolinone Derivatives with Dithiocarbamate Side Chains on Thymidylate Synthase," *Bioorganic and Medicinal Chemistry*, Vol. 14, No. 5, March 2006, pp. 1425–1430.

[79] Lipinski, C. A., et al., "Experimental and Computational Approaches to Estimate Solubility and Permeability in Drug Discovery and Development Settings," *Advanced Drug Delivery Reviews*, Vol. 23, No. 1-3, January 1997, pp. 3–25.

[80] Bugrim, A., T. Nikolskaya, and Y. Nikolsky, "Early Prediction of Drug Metabolism and Toxicity: Systems Biology Approach and Modeling," *Drug Discovery Today*, Vol. 9, No. 3, 2004, pp. 127–135.

[81] Yu, H., and A. Adedoyin. "ADME-Toxin Drug Discovery: Integration of Experimental and Computational Technologies," *Drug Discovery Today*, Vol. 8, No. 18, September 2003, pp. 852–861.

[82] Butina, D., M. D. Segall, and K. Frankcombe, "Predicting ADME Properties In Silico: Methods and Models," *Drug Discovery Today*, Vol. 7, No. 11, May 2002, pp. S83–S88.

[83] Eddershaw, P. J., A. P. Beresford, and M. K. Bayliss, "ADME/PK as Part of a Rational Approach to Drug Discovery," *Drug Discovery Today*, Vol. 9, September 2000, pp. 409–414.

[84] Perez, M. A. C., and M. B. Sanz, "In Silico Prediction of Central Nervous System Activity of Compounds, Identification of Potential Pharmacophores by the TOPS–MODE Approach," *Bioorganic and Medicinal Chemistry*, Vol. 22, November 2004, pp. 5833–5843.

CHAPTER 11

Information Management and Interaction in High-Throughput Screening for Drug Discovery

Preeti Malik, Tammy Chan, Jody Vandergriff, Jennifer Weisman, Joseph DeRisi, and Rahul Singh

High-throughput screening (HTS) assays provide a way for researchers to simultaneously study interactions between large numbers of potential drug candidates with a target of interest. Significant volumes of data are generated as a consequence of conducting HTS experiments, making it very difficult to store, interact, and thoroughly mine the data for biological significance. Thus management of HTS information is one of the key challenges for drug discovery data management and retrieval. In this chapter, we present our research towards the development of an information management, interaction, and visualization system for HTS data. Our goal is to provide a solution to this challenge that addresses many issues associated with HTS including managing and storing vast amounts of data, controlling process flow, and providing intuitive, highly interactive interfaces for analysis and visual mining of large datasets. We have developed a prototype system, called FreeFlowDB, specifically to meet these challenges and validate our methodologies in the context of antimalarial drug discovery research.

11.1 Introduction

The recent past in therapeutic drug discovery has been witness to several significant events in the evolution of science and technology. These include among others, the sequencing of the human genome and mapping of genomic DNA [1], as well as the somewhat simultaneous, recent developments in industrial robotics, combinatorial chemistry, and high-throughput screening. Taken together, these factors promise to provide a proliferation of targets, leading to newer or improved therapeutics and a significantly increased number of lead compounds synthesized in pharmaceutical drug-discovery settings [2, 3]. The process of drug discovery is a complex and multistage one. Generally, but not exclusively, a drug is a small molecule that interacts with a target, typically a protein or an enzyme, in a therapeutically beneficial manner. The primary stages of a standard drug discovery process are shown in

Figure 11.1. This process begins by determining potentially interesting targets. The identification of a target for drug discovery is done by screening gene databases using homology (sequence similarity) to known targets, colocation (which organ/tissue the gene is expressed in), or other criteria, like similar expression profiles that can indicate a biological connection between the gene and targeted symptoms or phenotypes. Once identified, the target is validated using molecular biology techniques like gene-knockouts and/or computational approaches like expression analysis along with pathway elucidation. A validated target is then screened against a large number of small molecules (potentially millions) to determine which of (called *hits*) interact with the target. The hits are further analyzed and optimized in terms of properties like binding potency, pharmacokinetics (PK), pharmacodynamics (PD), and efficacy, to come up with the molecules that are most suited to undergo clinical trials. In typical settings, the number of hits against a target represents a two to three order of magnitude reduction in the number of molecules being considered at the start. During the lead optimization stage, the initial focus is on maximizing potency by minimizing the concentration at which the drug-target interaction occurs. This further reduces the candidate list of molecules to (typically) a few hundred. Of these, after testing for PK/PD, only tens of molecules or fewer may remain to be considered for clinical trials.

Within the above process, high-throughput screening is a recent technique that has been aggressively adopted in academic and pharmaceutical drug discovery [4]. It involves miniaturized biochemical assays that are performed to enable researchers to test a large number of compounds at a fixed known dosage for binding activity or biological activity against the target molecule. These assays are often conducted in parallel in multiwell plates. Positive or active results are then collected for the next round of detailed testing to locate the optimal concentration and structure. The key advantage of HTS lies in its ability to study the interactions between large numbers of possible drug candidates with a target of interest. This not only significantly reduces the time and cost associated with lead discovery, but also allows exploring the ligand structural space at fine granularities.

The significant volume of information generated as a consequence of conducting an HTS experiment is easy to discern. The data volume constitutes the primary motivation for designing information management systems for HTS data. However, an analysis of the HTS process highlights further critical challenges an information management approach needs to address in this domain. These include:

1. Developing efficient techniques for data processing and analysis;
2. Capturing and presenting a process or *workflow-centric view* of the information as well as an *information-centric view* of the information;
3. Modeling *heterogeneity* in the information arising possibly due to the presence of different data types (physical heterogeneity) or different ways of organizing various parts of the available information (logical heterogeneity);

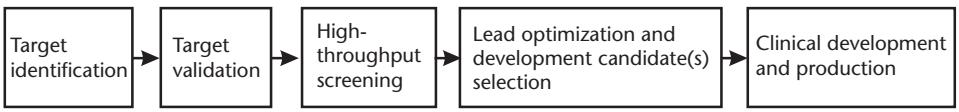


Figure 11.1 Stages in a typical drug discovery process.

4. Modeling evolutionary data as experimental logic changes;
5. Developing novel paradigms that combine visualization and querying to support interactions (query-retrieval) as well as assimilation of the data.

At the current state of the art, industrial research and development in the area of drug discovery data management has lead to a few systems [5–7] that can store large volumes of drug discovery data, potentially involving millions of molecules tested across hundreds of high-throughput screens as well as data from later stages. In contrast, the scope of academic research has typically been limited. For instance, there exists a marked paucity of publicly available systems for management of drug discovery information. Furthermore, in spite of the obvious scientific importance and broad impact of drug discovery, there is little computer science research in issues associated with high-throughput drug discovery data processing, data management, and paradigms for user-data interaction.

Towards addressing this goal, in this chapter, we present our investigations towards the development of an information management system for managing high-throughput screening data from drug discovery. Within this research, we consider three key aspects of the information management problem: data processing, data modeling for storage and retrieval, and paradigms for user-data interaction. Based on our research, we have developed a prototype information management system for high-throughput screening, which is validated through its application in antimalarial drug discovery. While we use the context of antimalarial drug discovery to describe our research, it should be noted that our research results extend to management of information in other disease-oriented settings. We begin this chapter in Section 11.2 with an overview of the current research in designing information management systems for drug discovery. A brief description of antimalarial drug discovery is presented in Section 11.3, which is intended to provide readers with the necessary biological background and motivate the ensuing description in light of a specific context. Next, Section 11.4 presents an overview of the system architecture. Using the context of antimalarial drug discovery, Section 11.5 describes a typical HTS workflow and discusses details of data formats and data preprocessing algorithms. Section 11.6 describes the data model that lies at the heart of the proposed system and shows how its design is motivated by the key information management challenges in HTS. This is followed by a description of the user interface that acts both as an information visualization tool as well as an interface mediating user-data interactions in Section 11.7. We conclude this chapter in Section 11.8 by discussing the contributions of this research and future goals.

11.2 Prior Research

The development of information management systems for drug discovery has been the focus of researchers in both academia and industry. The problem of small-molecule query retrieval and structure-activity modeling has received considerable attention both in academic and industrial research (see, for example, [8, 9] and references therein). From the information management perspective, an anticancer drug discovery database has been developed that screens a large number of compounds in vitro

against 60 human cancer cell lines from different organs of origin, and each tested compound is characterized by its "fingerprint" values [10]. Various statistical and artificial intelligence methods, including component analysis, hierarchical cluster analysis, multidimensional scaling, neural network modeling, and genetic function approximation, can be used to analyze this large activity database. Also, researchers have employed data automation using a multitier scheme (HTDS). In such architectures, the data processing tier produces reports and deposits the results into a database. The next tier is centered on a structure-property database and operations related to data mining or analyses can also be performed. Techniques for processing data from biological screens have also been the focus of software tools [11]. Among industrial research and development in this area, IDBS solutions provide a platform for information management across the full spectrum of drug discovery activities, starting from initial data capture, to long-term data management including results analysis and reporting [6]. MDL's Chemscape Server and Chime-Pro family of products are an integrated set of tools for chemical structure illustration, communication, and database searching [7]. Accelrys offers CombiMat, a data management system for materials science high-throughput experimentation (HTE) that supports the high-throughput design, characterization, testing, and analysis of materials samples that can be shared throughout an organization [5]. BACIIS uses a semantic data model to integrate dynamic, heterogeneous, autonomous, geographically distributed, and semistructured Web databases focusing on total data source transparency [12, 13].

11.3 Overview of Antimalarial Drug Discovery

Malaria is a devastating disease that claims more than 1 million lives annually. The majority of these deaths occur in children under the age of 5 in Sub-Saharan Africa. The most prevalent and lethal form of human malaria is due to the protozoal parasite, *Plasmodium falciparum*. *P. falciparum* has three distinct life-cycle stages: sexual reproduction occurs in Anopheles mosquitoes; a transition stage of asexual reproduction occurs in the human liver; and long-term asexual reproduction occurs in human erythrocytes. The parasite is transmitted to humans via a bite from its insect host, proceeds to the liver where it multiplies, and then exits the liver and enters the bloodstream. Methods of reducing the number of deaths caused by malaria include control of the insect vector (use of DDT), prevention of transmission to humans (vaccines), and rapid clearance of the parasite once transmitted (antimalarial drugs). Mosquito control successfully eradicated malaria from North America, however the disease is still endemic in tropical regions, including Africa, Central and South America, and Southeast Asia. Despite significant efforts, no commercial vaccine exists for widespread use in endemic regions. The burden, therefore, falls on antimalarial drugs to limit the death and suffering caused by the disease.

Chloroquine has been the mainstay of malaria treatment for the past several decades. Unfortunately, the emergence of chloroquine-resistant strains of the parasite has made this cheap and previously effective therapy practically useless in many regions of the world. Resistant strains of the parasite are also developing for other currently used therapy alternatives, such as mefloquine and sulfadoxine/

pyrimethamine. The need for novel antimalarial therapeutics is critical, and the drug discovery process is long and difficult. Potential malaria drugs must show antimalarial inhibition activity in a series of model systems before human clinical trials are undertaken. The first system that potential drugs are tested in is *Plasmodium falciparum* cultured in human erythrocytes. Compounds that inhibit growth of the parasite in culture at low concentrations are then candidates to test in rodent models of malaria. Rodents are not susceptible to infection by *Plasmodium falciparum*, but are susceptible to malaria infection by other species of the genus Plasmodium. Following favorable behavior of the potential drug compounds in rodent models, various larger animal models must be pursued before human clinical trials. Our work focuses on the identification of potential new antimalarial therapeutics at the preanimal model stage.

There are three main strategies of antimalarial drug discovery and development at the preanimal stage. One strategy is to synthetically alter the chemical structures of known antimalarial drugs (e.g., chloroquine) in order to recover efficacy [14–16]. Another strategy is to develop small molecule compounds that inhibit the normal action of specific parasite enzymes required for survival. A third strategy is to perform large-scale screening of small molecule libraries against the parasite cultured in human red blood cells to identify completely novel potential drugs with unknown mode of action. This third strategy is particularly important for identifying completely new chemical scaffolds for antimalarial drug design. Large compound libraries can be tested for inhibition activity against malaria cultured in erythrocytes with a high-throughput screening assay. One assay that is well suited for high-throughput screening of blood cells is a flow cytometry-based inhibition technique that is outlined in Section 11.5.

11.4 Overview of the Proposed Solution and System Architecture

Our research in data management for high-throughput drug discovery centers around the development of a system called FreeFlowDB (enabling the "free flow" of information between different stages/aspects of drug discovery). The proposed system takes a multiple perspective view of the problem and supports data processing, data storage, and highly intuitive interactions with the data. FreeFlowDB has been implemented as a Web-based application running on an Apache Web server. The data is persisted in the backend using MySQL and Berkeley DB XML databases. The system is organized using a three-tier methodology such that the Web tier (presentation), business logic, and data access layers are loosely coupled.

The Web tier primarily consists of the presentation logic. The middle tier is responsible for handling all the calls from the Web tier and it encapsulates the business logic for the application. The data access layer is responsible for persistence of data. This layered approach serves well from a maintenance and scalability perspective.

For the process flow, FreeFlowDB is materialized using custom modules built on open source tools. Figure 11.2 depicts a high-level representation. The open source components are comprised of: (1) Apache Web server configured for PHP for hosting dynamic Web content; (2) Java bridge for PHP to enable communication

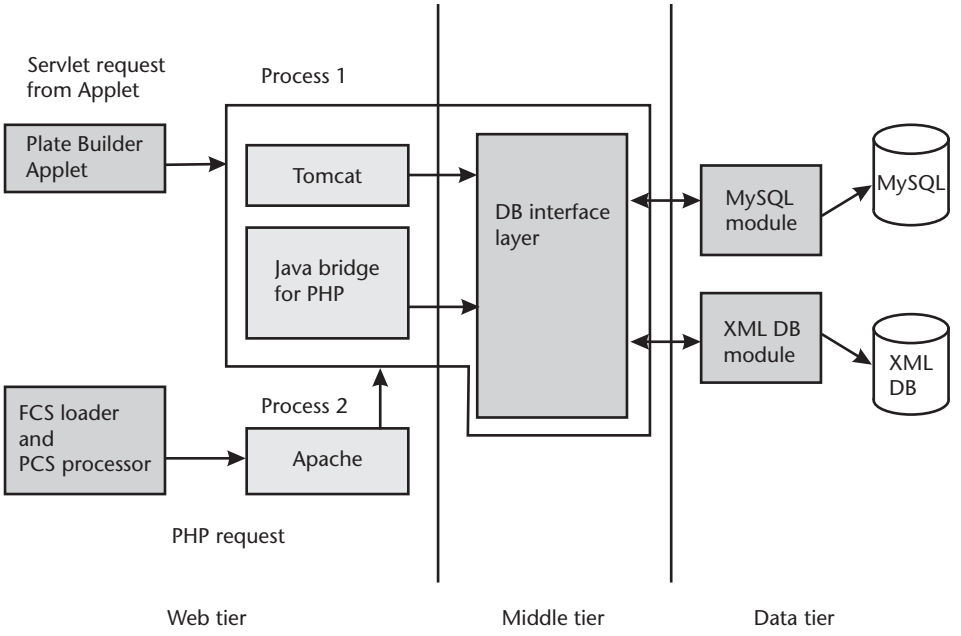


Figure 11.2 Three-tier architecture diagram of FreeFlowDB.

from PHP to Java; (3) Tomcat servlet container; (4) MySQL open source relational database; and (5) Berkeley DB XML open source database. The custom modules developed for FreeFlowDB (described in the following sections) are comprised of the following:

- *Data Loader and Data Processor:* This is an important component of the Web tier and is implemented in PHP.
- *Plate Builder/Viewer:* This is implemented as a Java applet in order to provide a much richer set of GUI controls and client-side dynamic behaviors.
- *DB Interface:* Serving as the middle tier of FreeFlowDB, it acts as the controller in the process flow of the system. It is responsible for all the business logic and delegates database calls to the appropriate DB module.
- *Data Access Modules:* These comprise the data access layer and are implemented using Java JDBC and Java-on-XML technologies.

11.5 HTS Data Processing

11.5.1 Introduction to HTS

High-throughput screening is a key step to reduce cost and time in the drug discovery process. A large number of compounds in a chemical library, ranging in size from 25,000 to 500,000, can be simultaneously screened for efficacy against a biological target in a multiwell tissue culture plate. A single plate can consist of 96, 384, 1,536, or more wells in which the biological target, reagents, and drug compound are placed for screening. An example of a 96-well tissue culture plate is shown in

Figure 11.3. With automated technology, several hundred plates can be screened each day [17]. Intuitively, the number of raw data files generated by HTS can accumulate rapidly and thus creates challenges for the storing and processing of data for further analysis. Therefore a system to provide an efficient way to capture and process data collected from these high-throughput drug screening experiments is necessary to facilitate drug discovery research.

11.5.2 Example of HTS for Antimalarial Drug Screening

Researchers screen hundreds of new compounds against various malaria strains and use flow cytometry to detect the effectiveness of these compounds. The process for doing so involves plating malaria-infected and uninfected red blood cells in 96-well plates. Growth inhibition of malaria strain *P. falciparum* cultures may be quantified using a fluorescent-active cell sorting (FACS) assay [18, 19]. Cultures of *P. falciparum* are grown in purified human erythrocytes and supplemented RPMI 1640 media. The erythrocytic lifecycle of *P. falciparum* is 48 hours. Synchronous cultures of parasite of known parasitemia and hematocrit are incubated for 72 hours in the presence of potential drug compounds. After incubation, the erythrocytes are fixed and stained with a fluorescent DNA-intercalator, or a fluorescent DNA binding molecule. The fixed and stained cells are then analyzed on a flow cytometer, which detects single cell fluorescence. Since erythrocytes do not have DNA, the dye will stain only those cells infected with the parasite. Counting the number of infected versus uninfected cells by flow cytometry allows for quantitative determination of drug-treated parasite growth relative to control cultures without the drug. All of this is done in high-throughput using a 96-well plate format.

The flow cytometer produces a series of binary files, each representing raw data collected for one well of a multiwell plate. Data contained in the binary files includes a record of each event, or cell, and the properties that were recorded for that event. Properties such as cell size, cytoplasmic granularity, cell shape and aggregation, and fluorescent intensity are all recorded. In the context of this particular assay, the fluorescent intensity of each cell is of greatest interest to researchers because it is the quantifiable measure of inhibition of parasite growth.

The processing of raw data files from HTS screening involves a data loading and processing step. This step is performed by the three modules described next.

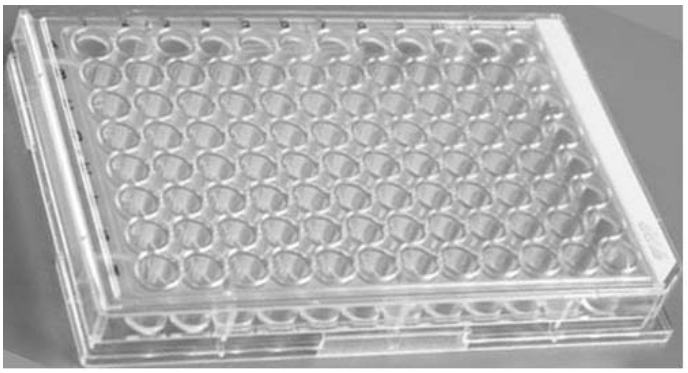


Figure 11.3 A plate with 96 wells, arranged in 8 rows and 12 columns.

Data Loader Module The Data Loader module provides an efficient mechanism for a user to upload binary files generated from a flow cytometer. The user creates a compressed data file containing all the binary data files generated by the flow cytometer and a mapping file that specifies the association of these binary files to their well location on the plate. In the Data Loader user interface, shown in Figure 11.4, a researcher can specify the following information related to the uploaded zip file:

- Plate name;
- Plate description;
- The location of the zip file selected either through a drop down menu (if the data file is preuploaded) or through a file chooser.

The Data Loader module ensures that the plate name is unique in the database. The binary data files in the zip file are then extracted and stored in a designated directory on the server. Only the file pointer of each binary file is stored in the database. Since an accurate association of the binary files and their respective well locations is important for later data analysis, the Data Loader will check if the number and name of the binary files uploaded to the system match the information stated in the mapping file.

Plate Configuration Module After storing the binary data files from the flow cytometer via the Data Loader module, it is essential to collect and track the experimental data of each well for the corresponding raw data files. For this antimalaria drug screening assay, experimental data such as drug ID, drug concentration, malaria strain grown, and the cell type is captured. The Plate Builder module discussed in Section 11.6 illustrates a powerful virtual plating environment for users to provide such experimental data to the system for later analysis.

Data Processor Module Once both experimental data and binary data files of a plate are entered in the system, the Data Processor module can be used to analyze the data. In the context of the antimalarial assay, the indicator of interest is called the *Parasitemia value* and is defined as the percentage of the infected cells that survive in the presence of the drug compound. The reader may note that this value is directly correlated to the effectiveness of a drug. Computing the Parasitemia value involves

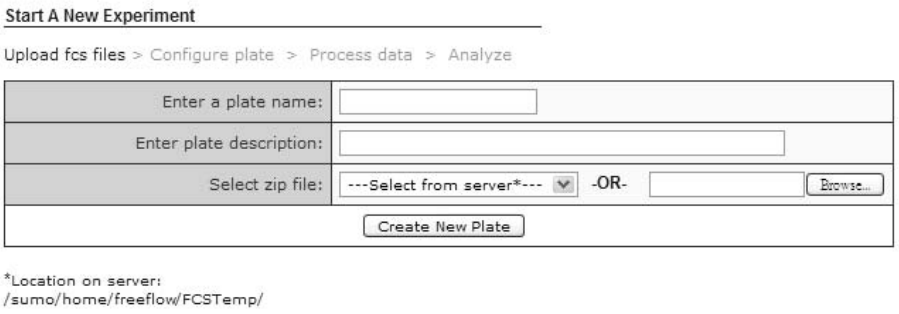


Figure 11.4 User interface for data loader.

analyzing the fluorescence intensity of cells infected with malaria and subsequently filtering the data using quality control parameters defined for the assay.

It should be noted that the Data Processor module can hold different algorithms, each designed for a specific assay. For instance, in a cancer drug discovery setting, an algorithm can be designed to calculate the ratio of live and dead tumor cells after a potential drug is supplied. Subsequently, the mean and standard deviation of data points from the control assay and the experiment can be analyzed using Z-factor analysis [20]. Next, this algorithm can be uploaded in the Data Processor Module and appropriately applied to the data.

The selection of parameters for analyzing a given assay is done using the Data Processor interface shown in Figure 11.5. In context of the antimalarial assay, for example, a user can specify two types of parameters:

1. Process parameters, which include: the window size and the threshold, used for data filtering;
2. Quality control (QC) indicators, including the minimum number of events captured per well and the percentage of allowable filtered data.

In the following, we present detailed description of the data processing steps involved in the antimalarial assay. Three major components of this processing step are:

- Calculating the gating value;
- Calculating the Parasitemia value;
- Displaying the QC indicators.

Calculating the Gating Value(s) Fluorescent intensity (FI) values are captured by the flow cytometer for every event. In this context, an event is defined as the passing of a single cell through the cytometer. The FI value is often noisy and high, especially during the starting and ending phase of the measurements, as shown in Figure 11.6. The data is therefore preprocessed and filtered before calculating the gating value. The data is first censored by the window size and threshold factor specified so that outlier FI data are filtered out. The overall median FI value of all events in the binary file is calculated next. A shifting window of size "m" (meaning "m" number of consecutive FI readings) is then used to analyze the FI values. If the average FI

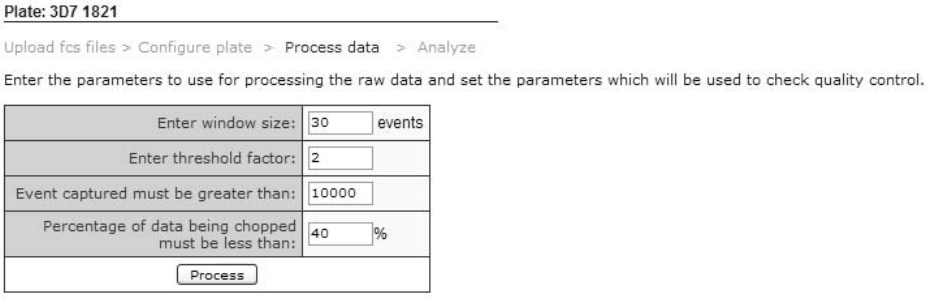


Figure 11.5 User interface for capturing processing parameters.

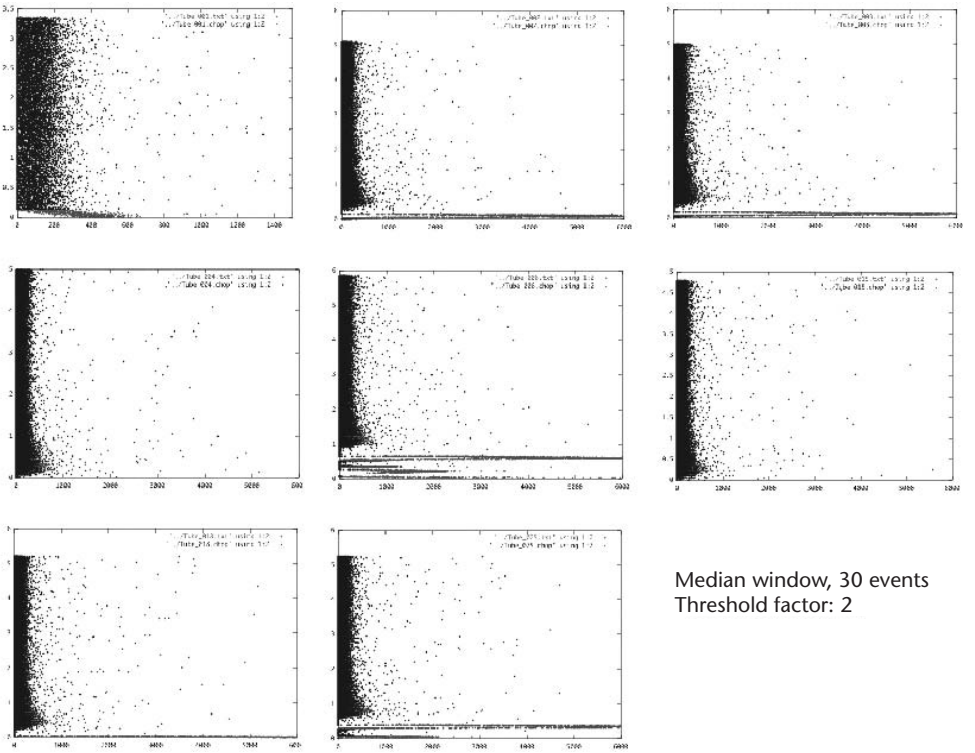


Figure 11.6 The FI distribution.

value within the window size is greater than the value of overall median FI multiplied by the threshold factor, all events, or cells, within that window are eliminated.

In order to determine an FI value cutoff, a gating on the FI reading is used. If the FI value of a particular event is higher than the gating value, it means that an infected cell is recorded. Ideally, an uninfected cell should have an absolute zero FI value. However, it is not easy to achieve an exact calibration of the measuring equipment and avoid an autofluorescence problem. In reality, an uninfected cell can generate a very small measure of fluorescence. In order to minimize the effect of background noise, the gating value is determined empirically as the FI value at the lowest 5% among all recorded events when arranged in descending order. Usually, the FI gate is calculated based on redundant uninfected control wells to enhance accuracy.

To summarize, the gating value is calculated only on uninfected control wells, which are specified in the plate configuration step by the Plate Builder. The FI values in each control well are sorted in descending order. The gating value of a control well is then computed as the FI value that is in the (0.005 * number of events in well + 1) position of the sorted list. The final FI gating value of the experiment is the average FI gate value obtained in all control wells.

Calculating the Parasitemia Value The Parasitemia value is computed as the percentage of the number of infected cells over the total number of cells in a well. With the average gating value computed from multiple control wells, the number of

infected cells in a particular well is obtained by counting the events of all FI values, after filtering the data points that are greater than the gating value. The resulting Parasitemia value of that well is then stored in the database. Intuitively, a higher Parasitemia value is an indication of a less effective drug. The formula to calculate Parasitemia is:

$$Parasitemia(\%) = \frac{Number\ of\ events\ that(FI\ value \geq FI\ gate)}{Total\ number\ of\ events\ after\ data\ cleaning} \times 100\%$$

Displaying the QC Indicators The QC indicators simply serve as a way for the user to specify acceptable cutoff values so that the data outside the range can be flagged. For each well, the QC indicators are calculated and displayed alongside the Parasitemia value to show the validity and trustworthiness of that measurement. If the number of events collected in a well is less than the requirement specified by the user, the Parasitemia value is highlighted to alert the user of the fact that it was computed based on insufficient data. Similarly, the percent of data that has been filtered, or removed from consideration, provides clues to the user about the reliability of the instrumental readings for a well. A user can then either decide to tolerate the deficiency or discard the data.

Once all three processing steps are finished, the number of events and the Parasitemia value for each well of the plate are stored in the database. Such information can be used to detect trends in the data during plate analysis. Figure 11.7 shows an example of the processed data.

11.6 Data Modeling

The key challenges facing database design for HTS include:

- *Schema extensibility:* Different HTS assays have different data organization logic, requiring different schemata. Further, changing a research approach even within a specific HTS assay can lead to changes in the experimental parameters (e.g., assays, organism to be researched). Schemata for storing such data should therefore be inherently extensible.
- *Data heterogeneity:* A drug-discovery project often involves different assays. This introduces heterogeneity in the data that needs to be tackled. Here, our notion of heterogeneity includes both *physical heterogeneity* [21], since the data elements present in the databases are stored as different data types (e.g., numbers, images, structures, visualizations), as well as *logical heterogeneity* due to possibly different logics of data organization inherent to each assay.
- *Support for data interaction:* Owing to the complexity in the data, there exists a need to provide user-information interaction paradigms that are focused not just on interactive query retrieval, but also on aiding information visualization, visual-data analysis, and assimilation.
- *Supporting a dual data model:* There is a need to support a data model that is both workflow-centric and data-centric owing to the volatile nature of data.

Processed Results [Process Plate]

Processed Result with window size=30, threshold=2, event captured should be greater than 10000 and data being chopped should be less than 40%

Statistics mode: median
avg_FITC_gate: 1029.1

Well	% Para	Total Event	% Data Censored	Median	Para Events	Events After Censored	Window Censored	Censored Events
A1 (+)	0.5	16705	0	85	81	16705	0	0
A2	1.3	17680	2.2	80	222	17290	13	390
A3	0.9	18265	1.6	122	157	17965	10	300
A4	0.7	20410	2.9	86	133	19810	20	600
A5	4.1	15074	7.6	62	568	13934	38	1140
A6	5.5	23985	24.3	24	1007	18165	194	5820
A7	6.1	22685	15.6	42	1162	19145	118	3540
A8	7.9	18330	35.7	4	929	11790	218	6540
A9	7.4	20020	41.2	0	876	11770	275	8250
A10	7.5	16055	31.4	17	829	11015	168	5040
A11	7.5	16250	24.6	24	921	12260	133	3990
A12 (+)	7.7	12285	44.4	0	527	6825	182	5460
B1 (+)	0.5	31135	7	73	154	28945	73	2190
B2	6.1	17290	26	25	777	12790	150	4500
B3	6.5	19500	26.5	22	935	14340	172	5160

[row][col] (-): negative control well that has no strain and no drug
[row][col] (+): positive control well that has strain but no drug
% of Para: (Para Event / Total Event) * 100%
Total Event: number of row in a fcs file
% of Data Censored: (Censored Event / Total Event) * 100%
Median: median of fluorescence value in fcs file
Para Event: fluorescence value that is greater than the gate
Event After Censored: Total Event - Censored Event
Window Censored: number of window of noisy data being censored out
Censored Event: Window Censored * window size

Figure 11.7 The processed result is displayed.

In order to address the database extensibility issue in the current setup, an XML database was implemented in conjugation with the relational database. An XML database is not constrained by the static structure used by any relational database like fixing the characteristics (columns) as well as the data types of any particular entity (table). This is a requirement owing to the reasons discussed earlier.

The problem of heterogeneity was solved using the generic APIs (to serve the UI demands) and wrapper classes over the two databases being used (to query the databases separately). For example, a call for getting plate information can be serviced by the database interface by delegating the call to appropriate database(s). Finally, FreeFlowDB was designed support a workflow-oriented environment rather than simply being data-centric. This environment is described in the following.

The FreeFlowDB database is designed to reflect the user workflow and aid in workflow-dependent data management. To ensure this, the system was designed to support different stages of the HTS workflow as described here:

- A user creates an experimental plate by uploading the raw data to the FreeFlowDB server and subsequently doing the virtual plating using the GUI provided by FreeFlowDB. This step stores the complete plate information into the database which includes the plate metadata and the configuration information for each well inside the plate (well metadata). The raw data related to

each well is persisted in the file system along with any associated pointers or metadata.

- The user can then use the system to process the plate by providing the processing parameters (e.g., window size, percentage of plate data that has been filtered out, threshold value). The processed data as well as the processing parameters are subsequently stored in the database. The processing parameters are stored as plate metadata, while the processed results are stored as well properties in the database. These are used to visualize the information at the plate level.

The process can be described diagrammatically in Figure 11.8 which shows: (1) data entry from the flow cytometer; (2) storage of the experimental data input during plate configuration; and (3) data generation by preprocessing the plates. In addition to its workflow orientation, FreeFlowDB is also designed to support a data-centric perspective. In this context, the key data in HTS experiments include:

- Descriptive experimental plate-related information;
- Information related to each well in any experimental plate;
- Experimental protocols;
- Processed plate results;
- Miscellaneous auditing information like time and owner for an experiment.

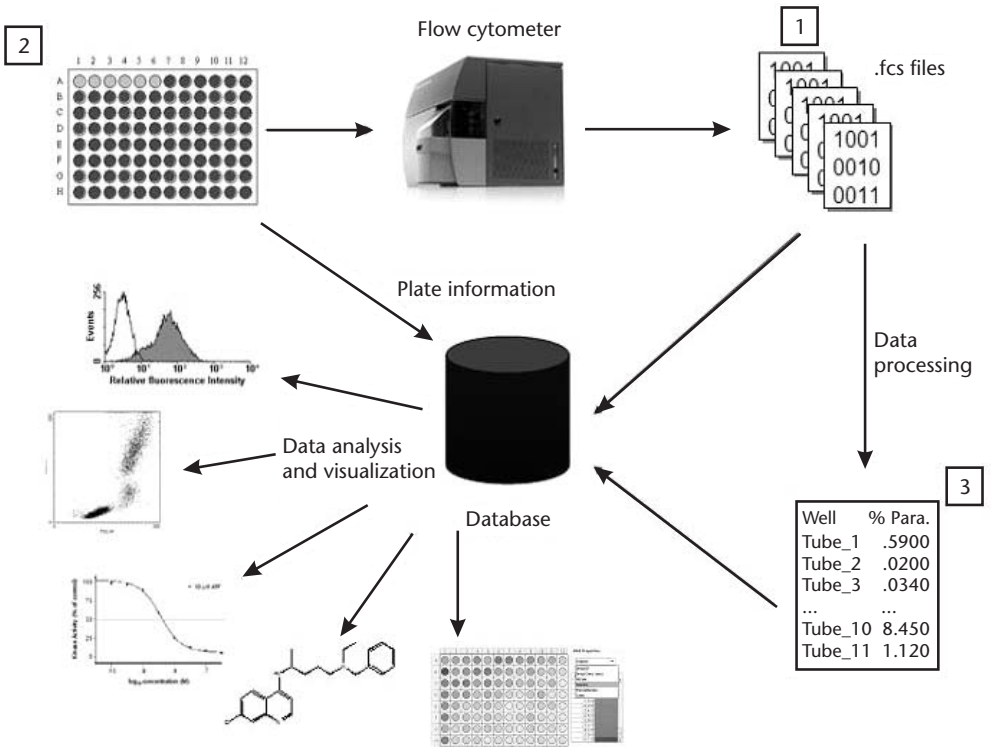


Figure 11.8 Block diagram depicting the workflow-oriented database environment.

11.6.1 The Database Design

The database is designed such that most of the data specific to the experiments is stored in the database itself (see Figure 11.9 for the UML diagram). Additionally, the actual binary result files are stored in the file system with pointers to them stored in the database. The binary files are owned by the Web application and permissions are set such that they cannot be modified outside of the Web application. This is done to reduce the volume of raw data in the database, which does not need indexing. The sources for data input include: (1) binary result files; (2) plate configuration files; and (3) data processing parameters entered via the user interface. Furthermore, the data is output as: (1) processed plate visuals for drug sensitivity in each well within the plate; (2) QC reports for each plate consisting of event counts, data processing statistics, and gate calculations; and (3) visualization of raw data in the form of histograms and dot plots.

The main entities in the database schema are:

- *Experiment*: a high-throughput screening experiment that consists of a single or multiple experimental plate(s);
- *Plate*: an experimental plate that consists of 96 (or 384) wells;
- *Well*: a particular well in an experimental plate;
- *Protocol*: an entity to store complete information for each of the different assays that are being used;

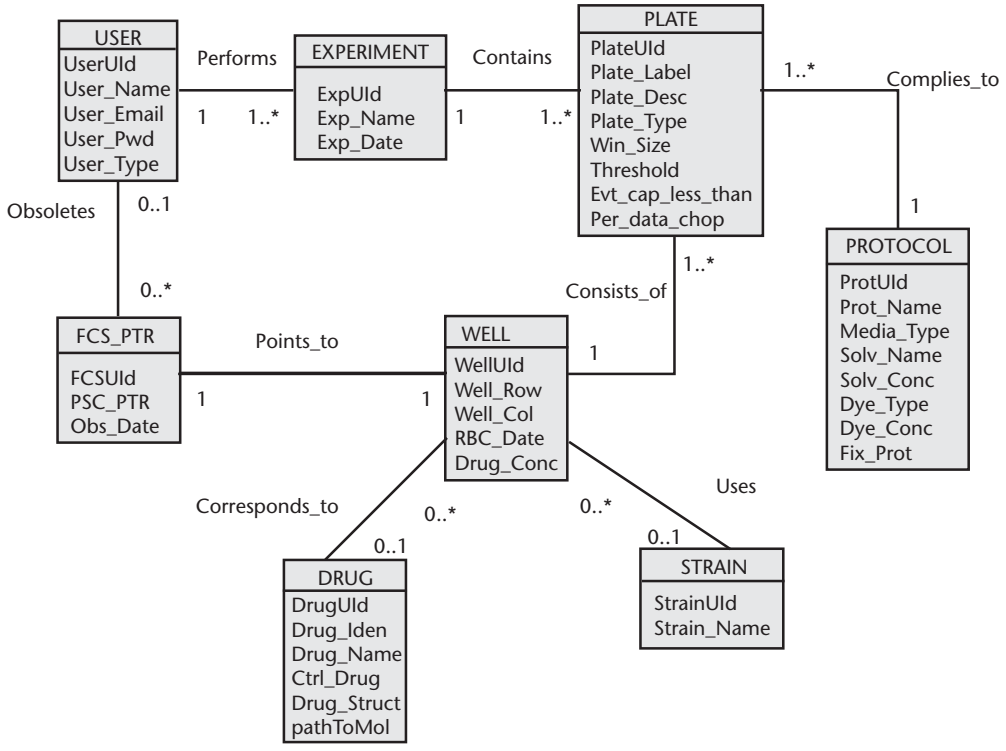


Figure 11.9 UML diagram.

- *Strain*: an entity to store information about the strain of the organism over which research is being conducted;
- *Drug*: an entity to store drug details for the drugs that are being tested;
- *User*: an entity to store researcher information.

The process of data persistence into the database is materialized with the user actions. A registered user has the options to create and process a new experimental plate, examine other experimental plates (that have already been processed), create or examine protocols (or assays) to be used in an experiment, and change the personal information.

In order to create a new experiment, the user is asked to give details for the new plate and upload a zip file that contains a set of binary files obtained from the flow cytometer (which contain valuable information about the experiment conducted that needs to be processed). Once this has been done, the plate information is stored as relational data as a Plate entity and simultaneously some well data of the 96-well plate is stored as a Well entity. The zip file uploaded usually contains 96 binary files with the file extension .fcs, which are stored in the file system with pointers to them in the relational database as an FCS_ptr entity.

The Plate entity contains a unique plate id, plate name, plate type (96 or 384), plate description, and some statistical information about the plate that will be stored and used when the plate is being processed later. Such statistical information includes window size for the events, threshold value(s), the upper bound on the number of events captured by the flow cytometer, and the percentage of noisy data that needs to be filtered out.

The Well entity has a unique well ID for each well, well row, well column, date of RBC culture, drug ID, drug concentration, strain ID, the binary file pointer ID, and some other well-dependent biochemical properties of interest.

Once the user has created the experimental plate, the plate then needs to be configured. The drug information, strain information, and other information such as well dependent properties (mentioned in the earlier point) are added to the database. These properties are stored in an XML database. The semistructured nature of the XML database allows the addition of well properties, such as cell type and malaria strain, are added by the user at the time of plate configuration. Other properties, such as the Parasitemia values, are saved after processing the plate data. It should be noted that the use of a semistructured data model, in this context, enhances database extensibility.

After configuring the plate, the user can proceed with the task of processing it. The parameters to process the plate are stored in the Plate entity while the processed results are stored as read-only well properties in the XML database, since it is unknown beforehand what kinds of results may be expected. A sample format of how well properties are persisted in XML is shown in Figure 11.10.

Once the processing is completed, the user is ready to view the results. This is achieved using color gradients overlaid on the plate GUI as well as displaying data in simple tables.

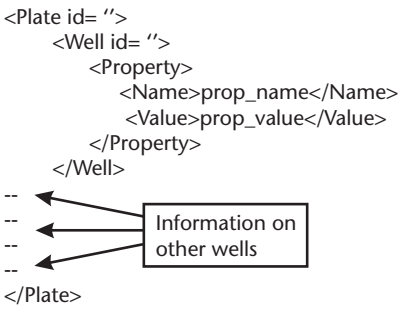


Figure 11.10 Block diagram depicting the XML file format.

11.7 User Interface

User experience is often undervalued in scientific applications where most of the emphasis is placed on storing and processing data. Recently, however, with the growing number of publicly available molecular biology Web servers [22], the importance of usability and user experience is becoming very clear [23]. Most bench biologists do not have the database know-how to explore their data directly. So, they are limited by the options presented to them on the interface. For this reason, developing scientific applications requires close collaboration between scientists and software engineers; otherwise, important patterns in the data may be missed. This problem is compounded as the data volume grows. Here, we explore intuitive interfaces for aiding in high-throughput screening experiments where a huge amount of data needs to be addressed.

Our goal in developing the FreeFlowDB interfaces was to provide a highly interactive and intuitive environment for interacting with large amounts of HTS data. The Plate GUI component of the FreeFlowDB application provides a rich graphical interface serving two main purposes. First, it provides an intuitive interface for capturing details of the experimental design, known as Plate Builder. Second, it provides a unique method for visualizing both the input data (experimental parameters, such as drug and drug concentrations) and the output data (numeric values calculated by the flow cytometer), known as Plate Viewer.

An extensive amount of metadata related to the experimental protocol and design needs to be captured in order to fully explore the generated data. For instance, in the high-throughput antimalarial assays using 96-well tissue culture plates, the configuration of experimental samples and controls need to be captured so that the data generated from laboratory automation equipment can be associated for more advanced data analysis and mining. Lab Information Management Systems (LIMS) are commonly used to track samples, experimental design, and results, but these are usually developed to generically handle laboratory information and are oftentimes difficult to integrate with specific laboratory instruments [24]. Providing an intuitive interface, modeled after the laboratory techniques being conducted, would greatly enhance user experience, thereby promoting accurate and thorough data entry.

Plate Builder provides an intuitive interface for the user to convey experimental parameters directly to a relational database. The interface provides a graphical representation of a 96-well plate, allowing the researcher to use a similar workflow for

data entry as what is used for the tissue culture experiment. Figure 11.11 shows the Plate Builder interface in "configuration" mode, which allows the input of four key parameters in this experiment: drug ID, drug concentration, malaria strain grown, and the presence or absence of red blood cells. Selection of an entire row or column of "wells" is available for quick and intuitive data entry.

On the other hand, Plate Viewer provides a unique approach for visualizing the data collected from the flow cytometer. Generated, numeric data, such as events (the number of cells collected in the well) and Parasitemia calculations (a measure of the degree of parasite infection in the sample) can be visualized on the 96-well plate interface. The advantage of doing this is that positional patterns can be quickly detected. For instance, Figure 11.12 illustrates the Plate Viewer interface in "analysis" mode. The events (or cell counts) are normally given as a gradient of the color red, not shown here. From this view, it is very apparent that there is a correlation between the number of living cells present in a well and the position of the well on the plate. The measure of events in this particular experiment is used primarily as a quality control check for the cell growth conditions and this view of the data is a quick and efficient way to detect problems.

In assessing efficacy of a particular drug compound against the malaria parasite, the Parasitemia values are the most important piece of data collected. These values can also be displayed in Plate Viewer using a color gradient that corresponds to the degree of parasitized red blood cells. Visualizing data in this manner provides a very quick way to screen samples at a superficial level, which can then be used to conduct more extensive analysis or experimentation.

In addition to providing visual clues about the data, Plate Viewer also allows the biologist to directly and intuitively interact with the experimental results at a level that would otherwise be impossible. A user can select a well of interest, and the corresponding numeric value is highlighted. For instance, in Figure 11.12, well E7 looks particularly pale shade when compared with surrounding wells. This is an indication that cells in this well did not grow as well as expected over the 4-day incubation period, even though they were exposed to the same reagents and parasite

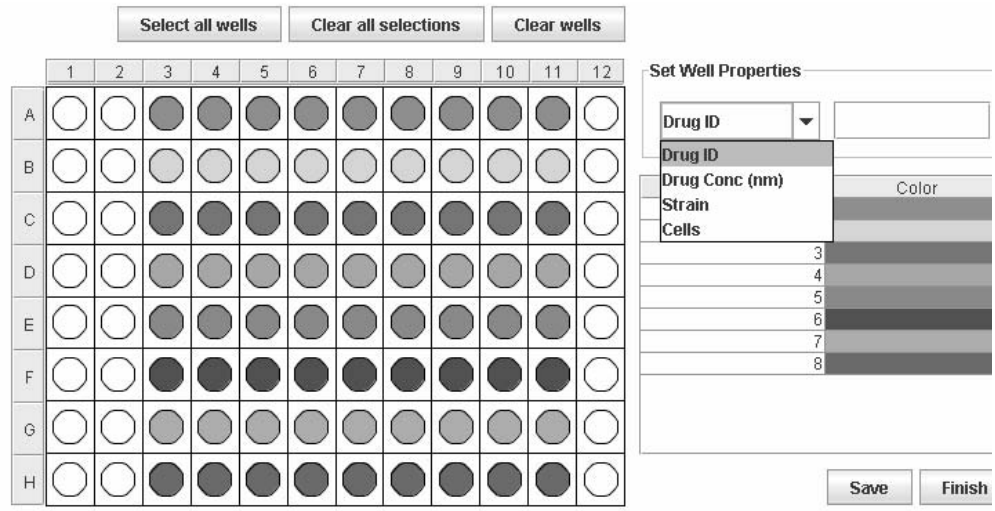


Figure 11.11 Plate Builder interface for capturing experimental parameters.

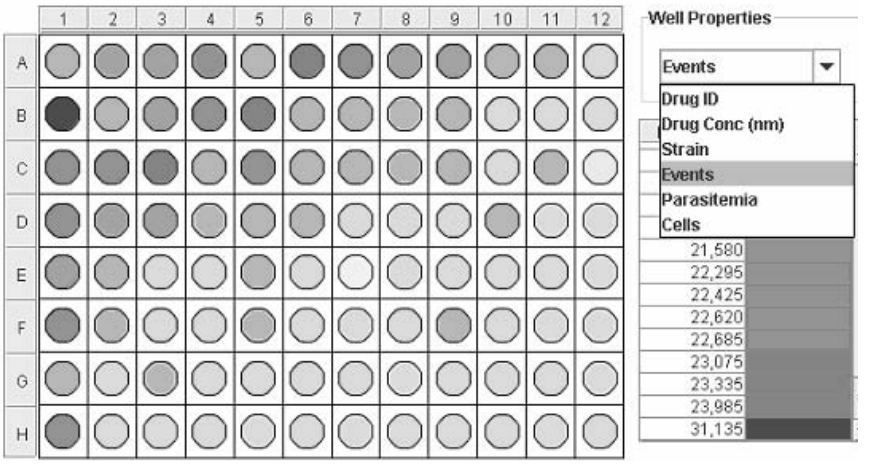


Figure 11.12 Interface showing correlation between cell growth and well position.

as other wells on the plate. Selecting this well scrolls to and highlights the corresponding value for the number of events. The well remains selected as the user toggles between other important parameters, such as the drug ID that was added to this well, the drug concentration, and the Parasitemia values. Since many factors affect biological responses such as cell growth, it is critical to be able to quickly compare various factors in the given system. Many times, much of this analysis is done programmatically using empirically determined thresholds, but human interaction is almost always required or otherwise, important clues would be missed. Plate Viewer provides an efficient way to do this.

11.8 Conclusions

The FreeFlowDB architecture aims to solve the problem of information management in HTS-based drug discovery. The current prototype solves many issues associated with high-throughput screening data including controlling process flow, managing and storing data, and providing intuitive, initial phase analysis of large amounts of data.

Our research in this area attempts to comprehensively address the information processing, storage, and query retrieval issues involved in high-throughput screening. While this research is presented in the specific context of antimalarial drug screening in this book chapter, our implementation is generic and can be extended to drug discovery directed against other diseases. Our future research is directed at extending FreeFlowDB through novel indexing techniques as well as providing a broader class of visualization and data interaction mechanisms. We also plan on introducing the drug compound structures into the database and data visualization interfaces in order to provide a complete picture of the HTS data.

Acknowledgments

This work has been partially supported by National Science Foundation Grant IIS 0644418, The SFSU Presidential Award for Profession Development awarded to Rahul Singh, and CSUPERB (California State University Program for Education and Research in Biotechnology). Tammy Chan has also been supported by the SFSU Graduate Fellowship Award for Continuing Graduate Students.

References

[1] Genome International Sequencing Consortium, "Initial Sequencing and Analysis of the Human Genome," *Nature*, Vol. 409, February 2001, pp. 860–921.

[2] Flickinger, B., "Using Metabolism Data in Early Development," *Drug Discovery and Development,* September 2001.

[3] Lehman Brothers and McKinsey & Company, *The Fruits of Genomics*, January 2001.

[4] Hertzberg, R. P., and A. J. Pope, "High-Throughput Screening: New Technology for the 21st Century," *Current Opinion in Chemical Biology*, Vol. 4, 2000, pp. 445–451.

[5] Accelrys, http://www.accelrys.com/chemicals/doe/.

[6] IDBS, http://www.idbs.com/solutions/.

[7] MDL, http://www.mdli.com/products/framework/chemscape/index.jsp.

[8] Singh, R., "Reasoning About Molecular Similarity and Properties," *Proc. IEEE Computational Systems Bioinformatics Conference (CSB),* 2004.

[9] Singh, R., "Issues in Computational Modeling of Molecular Structure-Property Relationships in Real-World Settings," *Proc. Conference on Systemics, Cybernetics, and Informatics (SCI),* Vol. VII, 2004, pp. 63–68.

[10] Leming M., et al., "Mining and Visualizing Large Anticancer Drug Discovery Databases," *J. Chem. Inf. Comput. Sci.*, Vol. 40, 2000, pp. 367–379.

[11] Shamu, C., "Building a High Throughput Screening Facility in an Academic Setting," http://iccb.med.harvard.edu/screening/faq_hts_facility.htm, 2003.

[12] Miled, B., et al., "A Decentralized Approach to the Integration of Life Science Web Databases," *Informatica (Slovenia)*, Vol. 27, No. 1, 2003, pp. 3–14.

[13] Miled, B., Y. Webster, and Y. Liu, "An Ontology for Semantic Integration of Life Science Web Databases," *International Journal of Cooperative Information Systems*, Vol. 12, No. 2, 2003.

[14] Elueze, E. I., S. L. Croft, and D. C. Warhurst, "Activity of Pyronaridine and Mepacrine Against Twelve Strains of Plasmodium Falciparum In Vitro," *J. Antimicrob. Chemother.*, Vol. 37, 1996, pp. 511–518.

[15] Lin, A. J., A. B. Zikry, and D. E. Kyle, "Antimalarial Activity of New Dihydroartemisinin Derivatives 7.4-(P-Substituted Phenyl)-4(R or S)-[10(Alpha or Beta)-Dihydroartemisininoxy] Butyric Acids," *J. Med. Chem.*, Vol. 40, 1997, pp. 1396–1400.

[16] Madrid, P. B., et al., "Synthesis of Ring-Substituted 4-Aminoquinolines and Evaluation of Their Antimalarial Activities," *Bioorg. Med. Chem. Lett.,* Vol. 15, 2005, pp. 1015–1018.

[17] Mullin, R., "Drug Discovery," *Chemical & Engineering News*, 82.30, http://pubs.acs.org/cen/coverstory/8230/8230drugdiscovery.html, July 26, 2004.

[18] Barkan, D., H. Ginsburg, and J. Golenser, "Optimisation of Flow Cytometric Measurement of Parasitaemia in Plasmodium-Infected Mice," *Int. J. Parasitol.*, Vol. 30, 2000, pp. 649–653.

[19] Madrid, P. B., et al., "Parallel Synthesis and Antimalarial Screening of a 4-Aminoquinoline Library," *J. Comb. Chem.*, Vol. 6, 2004, pp. 437–442.

[20] Zhang, J., T. Chung, and K. Oldengburg, "A Simple Statistical Parameter for Use in Evaluation and Validation of High Throughput Screening Assays," *Journal of Biomolecular Screening,* Vol. 4, No. 2, 1999.

[21] Singh, R., and R. Jain, "From Information-Centric to Experiential Environments," in *Interactive Computation: The New Paradigm*, D. Goldin, S. Smolka and P. Wegner, (eds.), New York: Springer-Verlag, 2005.

[22] Fox, J. A., et al., "The Bioinformatics Links Directory: A Compilation of Molecular Biology Web Servers," *Nucleic Acids Res*., Vol. 33, Suppl. 2, 2005, pp. W3–W24.

[23] Fulton, K., et al., "CLIMS: Crystallography Laboratory Information Management," *System. Acta Cryst.,* Vol. D60, 2004, pp. 1691–1693.

[24] Salamone, S., "LIMS: To Buy or Not to Buy?" *Bio-IT World,* http://www.bio-itworld.com/archive/061704/lims.html, 2004.

Selected Bibliography

Andritsos, P., et al., "Kanata: Adaptation and Evolution in Data Sharing Systems," *SIGMOD Record,* Vol. 33, No. 4, 2004, pp. 33–37.

Chen, L. H., M. Jamil, and N. Wang, "Automatic Composite Wrapper Generation for Semi-Structured Biological Data Based on Table Structure Identification," *SIGMOD Record Web Edition,* Vol. 33, No. 2, June 2004.

Deng, M., A. P. Sistla, and O. Wolfson, "Temporal Conditions with Retroactive and Proactive Updates," *Proc of Intl. Workshop on Active and Real-Time Database Systems (ARTDB-95),* 1995, pp. 122–141.

Dittrich, K. R., S. Gatziu, and A. Geppert, "The Active Database Management System Manifesto: A Rulebase of ADBMS Features," *Rules in Database Systems,* 1995, pp. 3–20.

Frauenheim, E., "IBM Joins Genetics Firm in Drug Search," News.Com, http://news.com.com/2100-1001-981749.html, January 22, 2003.

Hernandez, T., and S. Kambhampati, "Integration of Biological Sources: Current Systems and Challenges Ahead," *SIGMOD Record Web Edition*, Vol. 33, No. 2, 2004.

Jagadish, H. V., and F. Olken, "Database Management for Life Sciences Research," *SIGMOD Record Web Edition*, Vol. 33, No. 2, June 2004.

Jain, R., "Experiential Computing," *CACM*, Vol. 46, No. 7, July 2003.

Liu, H. -F., Y. –H. Chang, and K. -M. Chao, "An Optimal Algorithm for Querying Tree Structures and Its Applications in Bioinformatics," *SIGMOD Record,* Vol. 33, No. 2, 2004, pp. 21–26.

Sistla, A. P., and O. Wolfson, "Temporal Triggers in Active Databases," *IEEE Trans. on Knowledge Data Engineering,* Vol. 7, No. 3, June 1995.

About the Authors

Jake Chen is an assistant professor of informatics at the Indiana University School of Informatics and an assistant professor of computer science at the Purdue University School of Science, Department of Computer and Information Science, Indianapolis, Indiana. He is an IEEE senior member. He holds a Ph.D. and an M.S. in computer science from the University of Minnesota at Twin Cities and a B.S. in molecular biology and biochemistry from Peking University, China. He has been active in bioinformatics R&D in the biotech industry and academia. From 1998 to 2002, he was a bioinformatics computer scientist at Affymetrix, Inc., Santa Clara, California. From 2002 to 2003, he headed the Bioinformatics Department at Myriad Proteomics as Head of Computational Proteomics to map the first human protein interactome. Since coming back to academia in 2004, he became increasingly interested in the area of network and systems biology, where he has authored more than 30 research publications and filed for a worldwide patent for his invention at Indiana University of a technique to find highly relevant disease-specific genes/proteins from microarray, proteomics, and pathway datasets. He is currently the chief informatics officer and cofounder of Predictive Physiology and Medicine, Inc., Blooming, Indiana, a biotech company focused on the personalized medicine market.

Amandeep S. Sidhu is a senior researcher at Digital Ecosystems and Business Intelligence Institute at Curtin University of Technology, Perth, Australia, with expertise in protein informatics. He has been, and is currently leading, the innovative Protein Ontology Project since 2003. His research interests include biomedical ontologies, biological data modeling, structural bioinformatics, proteomics, XML-enabled Web services, and artificial intelligence. His work in these fields has resulted in more than 30 scientific publications. He is currently involved with many semantic Web and bioinformatics conferences and workshops as an organizer or as a program committee member. He is current acting as a coordinator for the IEEE Bioinformatics and Systems Biology community. He is also serving as the founding vice-chair of the NSW Chapter of IEEE Engineering in Medicine and Biology Society.

Regis-Olivier Benech is a postdoctoral fellow on a project working with the enzymology platform at the Centre for Structural and Functional Genomics (CSFG) at Concordia University.

Greg Butler is a professor in the Department of Computer Science and Software Engineering at Concordia University, Montreal, Canada. He leads the development of bioinformatics at the Centre for Structural and Functional Genomics (CSFG) at

Concordia, which hosted a large-scale fungal genomics project. Professor Butler has broad international experience, including periods visiting the University of Delaware, Universität Bayreuth, Universität Karlsruhe, and RMIT, Melbourne. His research goal is to construct knowledge-based scientific work environments, especially those for genomics and bioinformatics. In order to succeed at this, he has been investigating software technology for complex software systems, including software architectures, object-oriented design, and software reuse, to complement his experience in scientific computation and knowledge-based systems.

Tammy Chan is currently an application developer at Exelixis, where she works on developing heterogeneous n-tiered Web application infrastructure. She graduated from San Francisco State University with an M.S. in computer science with concentration in computing for life sciences. Her technical interests are bioinformatics, computational drug discovery, software engineering, distributed system, multimedia systems, and networking. She received two scholarships (SFSU Graduate Fellowship Award for Continuing Graduate Students & SFSU College of Science & Engineering Advisory Board Scholarships) for her outstanding research.

Nagasuma Chandra holds a faculty position at the Indian Institute of Science, Bangalore. She has a Ph.D. in structural biology, after initial training (B.Pharm and M.Pharm) in pharmaceutical sciences. The active areas of pursuit in her laboratory are systems biology approaches to drug discovery, modeling and simulation of virtual cells at different levels of hierarchy, and their integration with computational genomics and structural bioinformatics.

Elizabeth Chang is currently the director of the Digital Ecosystems and Business Intelligence Institute at Curtin Business School, Curtin University of Technology, Perth, Australia. She is the vice-chair of the Work Group on Web Semantics (WG 2.12/12.4) in the Technical Committee for Software: Theory and Practice (TC2) for the International Federation for Information Processing (IFIP). Professor Chang has published more than 300 scientific conference and journal papers as well as chapters in three authored books and two coedited books. The themes of these papers are in the areas of ontology, software engineering, object/component based methodologies, e-commerce, trust management and security, Web services, user interface and Web engineering, and logistics informatics.

Joseph DeRisi is currently an associate professor at the University of California at San Francisco, in the Department of Biochemistry and Biophysics, and is a Howard Hughes Medical Institute Investigator. Prior to joining the faculty at UCSF in 1999, Professor DeRisi was a UC Fellow for approximately one year. During his graduate work at Stanford University in the laboratory of Patrick O. Brown, he was one of the early pioneers of DNA microarray technology and whole genome expression profiling. He is nationally recognized for his efforts to make this technology accessible and freely available. Professor DeRisi was the lead instructor of the popular annual Cold Spring Harbor Laboratories course "Making and Using cDNA Microarrays." He was a Searle Scholar and is now a Packard Fellow. He was also the recipient of the 2001 JPMorgan Chase Health award and the 2004 WIRED RAVE award. In addition, Professor DeRisi holds the first Tomkins Chair at UCSF. Most recently, he

was chosen for a Macarthur Award in 2004. He has extended his work to the study of infectious diseases, including malaria and viral pathogens.

Tharam S. Dillon is currently working as a professorial fellow with Digital Ecosystems and Business Intelligence Institute at Curtin University of Technology, Perth, Austrailia. He was previously working as the Dean of Faculty of Information Technology at the University of Technology Sydney (UTS) until 2006. He was the Foundation Professor in Computer Science and Engineering at La Trobe University where he worked as head of the School of Engineering & Mathematical Sciences before he joined UTS in July 2003. He is the chair of the Work Group on Web Semantics (WG 2.12/12.4) in Technical Committee for Software: Theory and Practice (TC2) for the International Federation for Information Processing (IFIP). Professor Dillon is an expert in object component-based conceptual modeling and design, XML modeling, ontology development, and knowledge engineering. He has published six authored books and six coedited books. He has also published more than 600 scientific papers and has more than 1,000 citations of his work in refereed journals and conferences.

Wendy Ding is a bioinformatician and programmer. She developed the bioinformatics platform at the Centre for Structural and Functional Genomics (CSFG) at Concordia University.

Christopher Dwan trained in computer science, with a focus in artificial intelligence, at the University of Michigan in Ann Arbor. As a research engineer at the Environmental Research Institute of Michigan, Mr. Dwan developed pattern recognition algorithms for identifying military vehicles based on infrared and radar imagery. He also developed sensor fusion algorithms for the detection of land mines and unexploded ordinance, in collaboration with researchers at the Army Research Lab in Adelphi, Maryland. At the University of Minnesota's Center for Computational Genomics and Bioinformatics, Mr. Dwan provided technical support and software development for a wide variety of life sciences researchers. While there, he developed a customized version of the EnsEMBL genome annotation system for use with the Medicago truncatula and Lotus japonicus genome projects. This system was notable for its use of grid computing techniques to distribute workload across a variety of geographically and administratively discrete clusters. Mr. Dwan is an international lecturer and consultant on distributed computing for bioinformatics and is a principal investigator with BioTeam.

Ramez Elmasri is a professor in the CSE Department at the University of Texas at Arlington. He has more than 120 refereed research publications and is the lead coauthor of the widely used textbook *Fundamentals of Database Systems*, now in its fifth edition (Addison-Wesley). His recent research has been in the areas of bioinformatics and sensor networks, in particular in applying modeling, querying, and indexing techniques to these new challenging areas, and developing new techniques that take into account the specific characteristics of bioinformatics data and sensor networks data. His previous research contributions were in the areas of conceptual database modeling, temporal database querying and indexing, database query languages and interfaces, and systems integration. His current research interests also include using ontologies and XML in mediator and integration systems for accessing bioinformatics data sources.

Jack Fu received an M.S. in computer science from the University of Texas at Arlington in 2001. He is currently working toward a Ph.D. degree at the same university under the advisement of Professor Ramez Elmasri. His research areas include bioinformatics multilevel modeling, biomedical ontologies integration, XML, and semantic Web.

Mark G. Goebl graduated from the University of Notre Dame in 1980 with a B.S. degree in microbiology. In 1985, he received his Ph.D. in molecular genetics and cell biology from the University of Chicago working with Dr. Thomas Petes on the issue of essential DNA in the simple eukaryote *Saccharomyces cerevisiae*, otherwise known as Baker's or Brewer's yeast. From 1986 through 1988, he was a postdoctoral fellow in the Department of Genetics at the University of Washington where he worked with Dr. Breck Byers. While working with Dr. Byers, Dr. Goebl began his long-standing interest in ubiquitin-dependent protein turnover as it relates to the control of cell division, again focusing on the yeast system. Since 1988, Dr. Goebl has continued these studies while moving through the ranks from assistant professor to full professor of biochemistry and molecular biology at the Indiana University School of Medicine in Indianapolis.

Maureen A. Harrington is a professor of biochemistry and molecular biology at Indiana University, Indianapolis. She holds a B.S. in home economics from Purdue University, an M.S. in population studies from the University of Texas, School of Public Health, and a Ph.D. in pharmacology from the University of North Carolina, Chapel Hill.

Cornelia Hedeler is a research associate in the School of Computer Science at the University of Manchester, United Kingdom. She received a Ph.D. in bioinformatics from the University of Manchester and an M.Sc. in computer science from the University of Karlsruhe, Germany.

Josh Heyen is a fifth-year Ph.D. student in the laboratory of Dr. Mark Goebl at the Indiana University School of Medicine, Department of Biochemistry and Molecular Biology, in Indianapolis, Indiana. He currently holds a B.S. in biology from Illinois State University in Normal, Illinois.

Feng Ji received an M.S. in physical chemistry from Nanjing University (P.R. China) in 1997 and an M.S. in computer science from the University of Texas at Arlington in 2002. He is currently a Ph.D. student in the Department of Computer Science and Engineering at the University of Texas at Arlington. His current research areas focus on bioinformatics data modeling and integration using ontologies in mediator systems.

Yeturu Kalidas has a B.Tech in computer science and is currently a Ph.D student, working on structure-based virtual cells and applications in drug discovery.

John Longo is a bioinformatician and programmer. He developed the bioinformatics platform at the Centre for Structural and Functional Genomics (CSFG) at Concordia University.

Preeti Malik graduated from San Francisco State University with an M.S. in computer science with a concentration in computing for life sciences. Her technical interests are in bioinformatics, computational drug discovery, database management,

and interactive UI development. She is a recipient of the Outstanding Culminating Research Award of the Computer Science Department.

Jack Min is a postdoctoral fellow on a project working with the sequencing and annotation platform at the Centre for Structural and Functional Genomics (CSFG) at Concordia University.

Paolo Missier is a research fellow with the School of Computer Science at the University of Manchester, United Kingdom, since 2004. Prior to joining academia, he was a rsearch scientist at Telcordia Technologies (formerly Bellcore), New Jersey, from 1994 through 2001, where he gained research and industrial experience in the area of information management and software architectures. He has also been a lecturer in databases at the Universita' of Milano Bicocca, in Italy; an independent IT consultant for the Italian public sector; and has contributed to research projects in Europe in the area of information quality management and information extraction from Web sources. He holds a M.Sc. in computer science from the University of Houston, Texas, and a B.Sc. and an M.Sc. in computer science from the Universita' di Udine, Italy, in 1990. He has been an ACM member since 1998 and an IEEE member since 2001.

Nick O'Toole is a postdoctoral fellow on a project working with the passport platform at the Centre for Structural and Functional Genomics (CSFG) at Concordia University.

Lee W. Ott is a Ph.D. candidate in the Department of Biochemistry and Molecular Biology at Indiana University. He received an M.S. in biology from Wright State University and a B.A. in biology from Wittenberg University.

Sindhu Pillai is a bioinformatician and programmer. He developed the bioinformatics platform at the Centre for Structural and Functional Genomics (CSFG) at Concordia University.

Justin Powlowski is an associate professor in the Department of Chemistry and Biochemistry at Concordia University. His research revolves around the biochemistry of microbial detoxification pathways. Enzymes and proteins involved in the degradation of aromatic compounds feature most prominently. A variety of approaches and techniques are used, including classical enzymology, protein engineering by design or by directed evolution, and proteomics, to examine processes such as bacterial degradation of phenols and polychlorinated biphenyls, and fungal degradation of phenols and components of wood. In addition, his group is characterizing the proteins involved in bacterial resistance to mercuric ion.

Karthik Raman has a B.Tech. in chemical technology and is currently a Ph.D. student, working on systems-biology approaches to drug discovery, with an emphasis on tuberculosis.

Ronghua Shu is a research student and developed the first version of the enzymology database as his master's thesis at the Centre for Structural and Functional Genomics (CSFG) at Concordia University.

Rahul Singh is currently an assistant professor in the Department of Computer Science at San Francisco State University. His technical interests are in computational drug discovery and biology, multimedia systems, search, and retrieval. Prior to joining academia, he was in the industry (in the San Francisco Bay Area) at Scimagix,

where he worked on various problems related to multimedia biological information management, and at Exelixis, where he founded and headed the computational drug discovery group, which worked on various problems across the genomics-drug discovery spectrum. Dr. Singh received an M.S.E. in computer science with excellence from the Moscow Power Engineering Institute and an M.S. and a Ph.D. in computer science from the University of Minnesota. His research has received several awards, including the CAREER award (2007–2012) of the National Science Foundation and the Frost & Sullivan Technology Innovation Award.

Aleks Spurmanis was a technician in the Centre for Structural and Functional Genomics (CSFG) at Concordia University.

Reg Storms is an associate professor of the Biology Department at Concordia University. He is using genome-based methods to characterize fungal secretomes with the goal of identifying enzymes for environmental and industrial applications. This research includes the construction of fungal cDNA libraries, the development of EST analysis software, the development of expression vectors and host strains for protein production, and high throughput assays for genome scale enzyme characterization and applications testing. He also uses genome approaches to determine gene function through gene deletion, conditional gene expression and parallel analysis.

Jian Sun is a bioinformatician and programmer. He developed the bioinformatics platform at the Centre for Structural and Functional Genomics (CSFG) at Concordia University.

Adrian Tsang is a professor of the Biology Department at Concordia University. He received a B.Sc. in genetics from the University of Alberta in 1973 and a Ph.D. in biochemistry and cell biology from York University in 1978. He pursued his postdoctoral studies on a NATO Science Fellowship at the Imperial Cancer Research Fund in London working on molecular mechanisms of development. Prior to joining Concordia in 1991, Professor Tsang was a faculty member at York University and McGill University. Halfway through his academic career, he switched his research focus to protein production in fungi and more recently to fungal genomics. In 1999, he became the founding director of the Centre for Structural and Functional Genomics at Concordia University.

Peter Ulycznyj is the manager of the Centre for Structural and Functional Genomics (CSFG) at Concordia University.

Willy A. Valdivia-Granda is the founder and CEO of Orion Integrated Biosciences, Inc., in New York. This corporation is developing a new generation of integrated analysis systems and biodatabases to untangle the complexity of life. Mr. Valdivia-Granda is a molecular biologist and computational scientist leading the development of the algorithms for the reconstruction of molecular gene interactions in model organisms and to identify pathogen early infection biomarkers. He is also involved in several collaborative efforts for pathogen detection and the integration of nanotechnology, genomics, and bioinformatics. In 2004 Mr. Valdivia-Granda was voted Model.Org by *Genome Technology Magazine*. He serves as the assistant editor of the *Bioinformation Journal* and as a reviewer for the *Journal of Applied Genomics and Proteomics* and *PLoS Computational Biology* and as the editor-in-chief of *Computational and Genomic Methods for Infectious Diseases*.

Jody Vandergriff received a B.S. in molecular genetics from Ohio State University in 1998 and is currently pursuing an M.S. in computer science from San Francisco State University. Her research interests include usability and user interface design for bioinformatics applications and exploring novel ways to interact with biological data over the Web. Ms. Vandergriff has worked as a bioinformatics scientist at such companies as Celera Genomics, Applied Biosystems, and Genentech, where her work focused on protein classification ontologies, protein function prediction, biological pathway visualization tools, and Web applications for capturing and annotating biological data with high efficiency. She also works as an independent consultant in the area of user interface design for Web-based, data-rich applications.

Jennifer Weisman received a B.S. from The College of William and Mary in 1999 and a Ph.D. from the Department of Chemistry at the University of California, Berkeley, in 2003. She is currently a postdoctoral fellow in the Department of Cellular and Molecular Pharmacology at the University of California, San Francisco. Her research interests focus on enzyme-specific strategies to develop new antimalarial therapeutics.

Viroj Wiwanitkit is a visiting professor. He holds an M.D. and Board of Family Medicine from Thai Medical Council. He is on the board of scientific publication of the Thai Medical Association. He is the editor of many international journals in science and medicine, and acts as a reviewer for more than 50 international scientific and medical journals. He had more than 350 international publications and has authored many books at the international level. He was the Thai scientist who had the greatest number of international publications from 2004 to 2006. He has been invited to give lectures in many universities in many countries. He has been speaker/visiting professor for many congresses. His present affiliation is at Wiwanitkit House, Bangkhae, Bangkok, Thailand.

Cary Woods is the principal of a life science contract research firm and teaches bioinformatics at University of Indianapolis. He did his doctoral work in executive development at Ball State University.

Qing Xie is a bioinformatician and programmer. He developed the bioinformatics platform at the Centre for Structural and Functional Genomics (CSFG) at Concordia University.

Zhong Yan is a business information analyst at the Hong Kong and Shanghai Banking Corporation. She obtained a B.S. in physiology from Nanjing University, Jiangsu, China, an M.S. in cellular biology from the Institute of Pharmacology and Toxicology, Beijing, China, and an M.S. in informatics from Indiana University School of Informatics, Indiana.

Yan Yang is a bioinformatician and programmer. He developed the bioinformatics platform at the Centre for Structural and Functional Genomics (CSFG) at Concordia University.

Index