DESIGN AND USE OF THE BIOMOLECULAR INTERACTION NETWORK DATABASE (BIND) FOR STORING AND ANALYZING PROTEIN-PROTEIN INTERACTION DATA

by

Gary David Bader

A thesis submitted in conformity with the requirements
for the degree of Doctor of Philosophy
Graduate Department of Biochemistry
University of Toronto

DESIGN AND USE OF THE BIOMOLECULAR INTERACTION NETWORK DATABASE (BIND) FOR STORING AND ANALYZING PROTEIN-PROTEIN INTERACTION DATA

A thesis submitted in conformity with the requirements for the degree of Doctor of Philosophy

Gary David Bader
Graduate Department of Biochemistry, University of Toronto, 2003

Abstract

As genomics and proteomics technologies such as mass spectrometry, yeast two-hybrid, phage display and genetic interaction screens become more sensitive and robust, they are becoming more automated and high-throughput. These experimental systems are currently providing a wealth of data on genetic and molecular interactions and post-translational protein modifications. The Biomolecular Interaction Network Database (BIND - http://bind.ca) has been designed to store details about these molecular and genetic interactions, complexes and pathways and thus captures proteomics data in a computer readable format. Chemical reactions, photochemical activation and conformational changes can be described down to the atomic level of detail. Everything from small molecule biochemistry to signal transduction is abstracted in such a way that graph theory methods may be applied for data mining. The database can be used to study networks of interactions, to map pathways across taxonomic branches and to generate information for full pathway kinetic simulations. Currently, BIND is a web-based system that allows the database to be queried and for records to be entered. A Java applet to visually navigate the database and a BLAST against BIND service are both available via the web. BIND is an open community effort. All BIND records are in the public domain and source code for the project is made freely available under the GNU Public License. The system is designed so that both users and a curation staff can submit interactions

described in the literature, which are then vetted. BIND has been used to manage and automatically discover new knowledge residing in large yeast protein-protein and genetic interaction networks in *Saccharomyces cerevisiae* determined using mass-spectrometry, phage-display, yeast two-hybrid and roboticized synthetic lethal screens. A system, called MCODE (Molecular Complex Detection), for automatically recognizing molecular complexes in large molecular interaction networks, has been devised. MCODE is based on the notion that densely connected regions of a molecular network, or graph, represent molecular complexes. The BIND project illustrates how a structured software development process focusing on the design phase provides a sturdy foundation for the future implementation of bioinformatics tools that solve real biological problems.

Acknowledgements

First and foremost, I would like to thank my parents, Hazel and Dennis, as well as my sisters Danielle and Andrea for their unending love and support that has allowed and encouraged me to focus on my Ph.D. work while living with them in Toronto. I would also like to thank the many professors, mentors, colleagues and friends at the University of Toronto who have created such a unique and exciting environment for me to study, learn about and be a part of the amazing scientific achievements, driven by the completion of many genome projects, currently taking place in Biochemistry, Molecular and Cellular Biology and Bioinformatics.

I must especially thank my supervisor, Dr. Chris Hogue, for seeing the potential of a young undergraduate student and giving me the opportunity to have fun creating Bioinformatics software and learning about Biology and the field of science in general. All of the work, conferences, publications, lectures, poster presentations, advice, constructive criticism, summer student supervisory responsibilities, examples and other opportunities that I have been given have been the perfect mix of ingredients for an enjoyable and successful learning experience. I am most grateful to Dr. Tony Pawson for developing the original idea for the BIND project before I arrived at the Samuel Lunenfeld Research Institute (SLRI) in September 1998 and for being a member of my supervisory committee. His prescience in identifying the need for such a database and his advice and support along the way towards its creation have been extraordinary. Thanks go to Dr. Chris Yip for being a committee member and for sharing his enthusiasm in anything technology or computer related.

I have been extremely fortunate to have an opportunity to collaborate on world-class Biology projects led by talented and enthusiastic scientists. These projects were enjoyable and provided a fantastic learning experience. Dr. Charlie Boone led two projects, both of which led to papers in Science and could likely not have been finished without Charlie's unending 'wammo!' enthusiasm. Dr. Mike Tyers led the HMS-PCI project, which was another enjoyable and memorable learning experience. Another enjoyable collaboration, more closely related to BIND, was with Professor Francis Ouellette and his group, Patrick Franchini, Graeme Campbell and Sohrab Shah at the

Centre for Molecular Medicine and Therapeutics (CMMT) at the University of British Columbia (UBC) in Vancouver. While not direct collaborators, the authors of the original NCBI ASN.1 data specifications, James Ostell, Steve Bryant, Hitomi Ohkawa and others, as well as the authors of the NCBI programming toolkit, including Denis Vakatov, must be graciously thanked for providing the software development base and elegant software design principles upon which BIND is built.

My lab colleagues, past and present, Katerina Michalickova, Howard Feldman, Moyez Dharsee, Cheryl Wolting, Ian Donaldson, Michel Dumontier, Doron Betel, Sherrie Kelly, Sue Sroka and many summer students and past and present members of the Hogue lab were part of a friendly environment conducive to work and I thank them for that and their hard work on projects related to BIND. I especially thank Katerina for developing the SeqHound database system, which is used to support most research in the lab. I also appreciate the work of administrative assistants and staff of the Biochemistry Department and the SLRI for their support.

I was financially supported over the past few years by an Ontario Graduate Scholarship (OGS), the Ontario Student Opportunity Trust Fund (OSOTF) Bank of Montreal Fellowship in Medical Research and by scholarships from the University of Toronto Biochemistry Department. The Canadian Institutes of Health Research (CIHR), the Canadian Foundation for Innovation (CFI) and Connaught (Aventis), originally supported the BIND project. Via the work of Drs. Hogue, Pawson and others, BIND has earned support for continued development from Genome Canada, the Ontario Research and Development Challenge Fund (ORDCF), IBM, MDS Proteomics and the Singapore Economic Development Board (EDB).

Table of Contents

List of Tables

List of Figures

List of Appendices

List of Abbreviations

ANSI – America National Standards Institute
APC – Anaphase Promoting Complex
API – Application Programming Interface
ARP2 – Actin Related Protein 2
ASCII - American Standard Code for Information Interchange
ASN.1 – Abstract Syntax Notation 1
BER – Basic Encoding Rules
BIND – Biomolecular Interaction Network Database
BIND-ID – BIND Identifier
BKA – BIND Key Assigner
BLAST – Basic Local Alignment Search Tool
BMOID – BIND Molecular Complex Object Identifier
CAD – Computer Aided Design
CAS – Chemical Abstracts Service
CGI – Common Gateway Interface
CORBA – Common Object Request Broker Architecture
CPU – Central Processing Unit
CSNDB – Cell Signaling Network Database
CVS – Concurrent Version Control System
DDBJ - DNA Data Bank of Japan
DDR – DNA Damage Response
DFC – Dense Fibrillar Component
DIP – Database of Interacting Proteins
DTD – Document Type Definition
EGFR – Epidermal Growth Factor Receptor
ELISA – Enzyme-Linked Immunosorbent Assay
EMBL – European Molecular Biology Laboratory
FC – Fibrillar Component
FTP – File Transfer Protocol
GC – Granular Component
GI – GenInfo Identifier
GIF – Graphics Interchange Format
GNU – GNU Not Unix
GO – Gene Ontology
GPL – GNU Public License
GRB2 – Growth factor Receptor Bound Protein 2
GST – Glutathione-*S*-Transferase
HA - Hemagglutinin
HLA – Human Leukocyte Antigen
HMS-PCI – High Throughput Mass Spectrometric Protein Complex Identification
HTML – Hypertext Markup Language
HTP – High Throughput
HT-MS – High Throughput Mass Spectrometry
IC50 – Median Inhibition Concentration

IAID – Internal Action Identifier
IBM DB2 – International Business Machines Database 2
ICID – Internal Conditions Identifier
ISID – Internal State Identifier
IDL – Interface Definition Language
IID – Interaction Identifier
IUPAC – International Union of Pure and Applied Chemistry
KEGG – Kyoto Encyclopedia of Genes and Genomes
LAT – Linker for Activation of T- cells
LC-MS/MS – Liquid Chromatrography MS/MS
LS – Large Scale
LIMS – Laboratory Information Management System
MALDI-TOF – Matrix Assisted Laser Desorption/Ionization Time of Flight
MAP – Mitogen-Activated Kinase
MCID – Molecular Complex Identifier
MCODE – Molecular Complex Detection
MHC – Major Histocompatibility Complex
MIPS – Munich Information Center for Protein Sequences
mmCIF – macromolecular Crystallographic Information File
MMDB – Molecular Modeling Database
MS – Mass Spectrometry
NCBI – National Center for Biotechnology Information
OMIM – Online Mendelian Inheritance in Man
ORF – Open Reading Frame
PDB – Protein Data Bank
PFI – Polyadenylation Factor I
PID – Pathway Identifier
PMID – PubMed Identifier
PRP – Proteasome Regulatory Particle
PSI-BLAST – Position-Specific Iterated BLAST
PSSM – Position Specific Score Matrix
PTB – Phosphotyrosine Binding
RNAi – Ribonucleic Acid Interference
SAGA - SPT-ADA-GCN5 Acetyltransferase
SGD – Saccharomyces Genome Database
SDS-PAGE – Sodium Dodecyl Sulphate Polyacrylamide Gel Electrophoresis
SELEX – Systematic Evolution of Ligands By Exponential Enrichment
SGA – Synthetic Genetic Array
SH2 – Src Homology 2
SH3 – Src Homology 3
SLRI – Samuel Lunenfeld Research Institute
SMART – Simple Modular Architecture Research Tool
SQL – Structured Query Language
SRS – Sequence Retrieval System
SVG – Scalable Vector Graphics
SVM – Support Vector Machine

TAP – Tandem Affinity Purification
TraDES – Trajectory Directed Ensemble Sampling
TRAPP – Transport Protein Particle
UML – Unified Modeling Language
URL – Universal Resource Locator
VWP – Vertex Weight Percentage
WASP – Wiskott-Aldrich Syndrome Protein
WIT – What is There?
WWW – World Wide Web
XML – Extensible Markup Language
Y2H – Yeast Two-Hybrid
YPD – Yeast Proteome Database

Chapter 1 – Introduction

The majority of the work presented in this chapter has been published as follows (reprinted with permission, copyright Wiley-VCH):

Gary D. Bader, Christopher W.V. Hogue
Chapter 18 - Interaction Databases, in Genomics and Bioinformatics
(Volume 5B of the Series: Biotechnology, 2nd Edition. Eds Rehm, H.-J., Reed, G., Pühler, A., Stadler, P.)
Wiley-VCH, Germany

Introduction

Given estimates based on the draft sequence of the human genome (Lander et al., 2001; Venter et al., 2001) of between 30,000 to 80,000 human genes, it is apparent that a minority of these genes encode conventional metabolic enzymes or transcription-translation apparatus. Genomic sequencing of metazoans and more specifically vertebrates has uncovered large numbers of complex multi-domain proteins, many containing interacting modules, such as SH3 domains, which generally bind proline-rich protein regions. The complexity of the DNA blueprint is augmented in an exponential fashion when one considers the possibility that these multi-domain proteins could bind to several other biomolecules either simultaneously or at different points in the cell cycle or in different cell types. A molecular interaction is a specific binding event resulting from atomic-level physicochemical forces (Jones and Thornton, 1996). Multiple binding events among many different proteins in a cell form "interaction networks". These networks form conventional signaling cascades, classical metabolic pathways, transcription activation complexes, vesicle mechanisms, and cellular growth and differentiation systems, indeed all of the systems that make cells work (Pawson et al., 2001).

The ultimate manifestation of gene function is through intermolecular interactions. It is impossible to disentangle the mechanistic description of the function of a biomolecule from a description of other molecules with which it interacts. One of the best forms of the annotation of a gene's function, from the perspective of a machine-readable archive, is information linking specific molecular interactions together, an interaction database. Thus, interactions, defining molecular function, and interaction databases are critical components as we move towards a complete and dynamic functional description of the cell at a molecular level of detail. Interaction databases are essential to the future of bioinformatics as a new science. In this review, what can be achieved through integration of current interaction information into a common framework is broadly considered, and a number of databases that contain interaction information are examined.

Scientific Foundations of Biomolecular Interaction Information

Interaction information is based on the experimental observation of a specific interaction between two or more molecules. For the purposes of this discussion, natural, biological molecules spanning the entire range of biochemistry are spoken of, including proteins, nucleic acids, carbohydrates and small molecules, both organic and inorganic and even light. Interaction information could also be considered for genes, as in a synthetic lethal genetic interaction, although this interaction is less direct. Interaction information is an inference that two or more molecules have a preferred specific affinity for each other, within a living organism, and that inference is based on experimental evidence using conventional experimental molecular and cell biology techniques.

The number of experimental types that can provide this evidence is large. Primary interaction experiments can be broadly described as being based on direct observation of two molecules directly interacting or of a measurable phenomenon directly related to that interaction. This may be *in-vivo*, such as a yeast two-hybrid assay, or *in-vitro* as in a fluorescence polarization experiment using purified reagents in a cuvette. Experimental genetic evidence provides another type of information. For example a tandem gene knockout in an organism may cause a certain phenotype to appear such as a growth defect or lethality. This phenotypic readout provides evidence that the two genes are involved in pathways affecting the phenotype, and may imply a molecular interaction between the two gene products or that the two genes act in redundant pathways. This data is indirect and possibly dependent on other genes in the background of the experimental system. Nonetheless all this information is important in helping us to understand gene and protein function in dynamic molecular interaction networks. By storing primary interaction data in a common machine-readable archive, as is currently done for gene sequence and molecular structure information, we would have a tremendous resource for research biology and data retrieval.

Interaction databases ideally should contain information that is in the form of a correlated pair or group of molecules, some link to the experimental evidence that led to the interaction, and machine-readable information about what experimental interaction parameters are known. For example, did the interacting molecules undergo a chemical

change during the interaction? Was the binding reversible? What are the kinetic and thermodynamic parameters, if they were measured in the experiment? Were the forms of the molecules in the experiment wild type, or mutated variants? What are the binding sites on the molecules?

The Graph Abstraction for Interaction Databases

Consider the collection of molecules in a cell as a graph. Each molecule is a vertex or node, and each interaction is an edge. Classical bioinformatics databases hold protein sequence, DNA sequence and small molecule chemistry databases, collectively, hold molecules, which are the vertices of this graph. In contrast, the ideal interaction database will hold the edge information – which two molecules come together, under what cellular conditions, location and stage, how they interact, and what happens to them in the course of the interaction. This concept is referred to as the graph-theory abstraction for interaction databases. It is a powerful data abstraction as it simplifies the underlying concepts and allows one to apply algorithms that are well understood from the field of computer science to the larger problems of data mining and visualization. Having a clear picture of a general graph abstraction for interaction databases is the key to the integration of data into a universal framework.

Nodes in a graph do not have to represent single parts of a cell; a single node can represent multiple related parts. For example, a protein and its orthologues could be represented by one node in a graph of a metabolic pathway. The graph would thus correspond to a generalized pathway across more than one organism. Edges in graphs can have direction from cell signal information flow (e.g. from cell surface to nucleus) or from chemical action direction (e.g. kinase phosphorylates a protein substrate). Nodes and edges can be assigned a weight that could be mapped from information in a database. For instance, a node could be assigned a higher weight if it is a larger protein and this could be translated to a larger node in a graph visualization system. An edge can be assigned a weight based on the confidence in an interaction. This probability value could be derived from a function of the type and number of experiments that were done to conclude that two molecules interact.

The Biomolecular Interaction Network Database (BIND) (Bader and Hogue, 2000) seeks to create a database of interaction information around a generalized graph theory abstraction of interaction data.

Why Contemplate Integration of Interaction Data?

In building the BIND data model, a prototyping approach was pursued, which is very different from the way most biological databases are created. A comprehensive data model was designed that allowed interaction information to be represented in a machine-readable format, spanning all type of molecular interactions, including protein, RNA, DNA and small molecules and the biochemical reactions, complexes and pathways they are involved in. The BIND data specification was created following the NCBI ASN.1 architectural model (Ostell and Kans, 1998) and the NCBI software development toolkit for implementing early versions of the BIND database and its tools. A considerable amount of time was spent focusing on designing the data model for BIND, contemplating the way molecular interaction and molecular mechanism information would be stored, from inferences as broad as a genetic experiment, to as precise as the atomic level of details found in a crystal structure of an interacting complex. The hypothesis was that there is a plausible universal computer readable description of molecular interactions and mechanisms that can suffice to drive whole cell visualization, simulation and data retrieval services. The design phase involved asking ourselves and others questions such as: What data should be represented? What abstractions should be used? How can interactions be described together with chemical alterations to the interacting molecules? The outcome of this hypothesis testing exercise is described in detail in the BIND specification (Bader and Hogue, 2000).

A Requirement for More Detailed Abstractions

Molecular interaction data must be abstractly represented so that computations may be carried out and data maintained in a machine-readable archive more easily. This is a simple idea with an analogy in biological sequence information. Biopolymer

molecules, DNA and proteins, are abstracted for the computer as strings of letters. This information tells us nothing about conformation or structure of the molecule, just of composition and biopolymer sequence. The IUPAC single letter code for DNA and for amino acids are abstractions that contain sufficient information to reconstruct chemical bonding information, provided that a standard form of the biopolymer is being represented, and not a phosphorylated, methylated or otherwise modified form.

One cannot imagine a database of cellular biomolecular assembly instructions without first having an enumeration of the contents of the cell, the biomolecular parts list. Sequence databases only partially fulfill this parts list requirement, as precise information about post-translational modification of biopolymers is not encoded. Also, small molecules, such as metabolites are not included in sequence databases. In order to encode exact information about biomolecules, one must have the capacity to describe the biopolymer both as sequence and as an atom-and-bonds representation, the chemical graph.

A chemical graph description of a biomolecule is sufficient to recreate the atoms, bonds and chirality of the molecule, but without specifying the exact location of the atoms in 3-D space. In other words, a chemical graph is an atomic structure without coordinate information. A chemical graph data abstraction exists within the NCBI MMDB data specification and database of molecular structure information (Wang et al., 2000). This specification is the only example of a chemical graph based structure abstraction, and a complete chemical structure may be encoded in MMDB without knowing a single X, Y, Z atom coordinate. The PDB and the newer mmCIF molecular structure file format both do not have a chemical graph data structure that can describe the complete chemistry of a molecule without atomic coordinate information (Berman et al., 2000).

Sequence alphabet abstractions have been invaluable in bioinformatics, having enabled all computer based sequence analysis. This would have been very difficult to compute had an exact database of atoms and bonds making up each biopolymer sequence been chosen as the abstraction. While this information might bog down sequence comparison, it is required for a more precise record of the chemical state of a biopolymer following post-translational changes. These chemical states, once accessible through a

precise database query, are important to have recorded as they form the control points for uncounted pathways and mechanisms for cellular regulation.

Abstractions are rarely applicable universally for all kinds of computations. As computing power increases, abstractions can be expanded in detail to fulfill the requirements of more kinds of computations. So far, there has been resistance to expand the abstractions of sequence information to more complete descriptions like a chemical graph, but it is clear these will be required to describe large and important parts of molecular biology such as phosphorylation, carbohydrate or lipid modification, and other post-translational changes upon which many molecular mechanisms depend.

Interaction databases can be contemplated now because it has been demonstrated that computer infrastructure can keep up with genomic information. However the representational models that are selected need to be carefully chosen in order that they not preclude a computation that may be required in future research. It may be time to find an abstraction that can accommodate the most complete description for molecular information one can imagine. With adequate standard data representations for molecules that are unambiguous for the purposes of general computation, specifying sequences, structures and small molecule chemistry, it should be possible to move ahead with annotation of molecular function in a very complete fashion. Without this, machine-readable descriptions of knowledge will be ambiguous and will be limited in the precision which biological simulation, visualization and data mining tools will require.

An Interaction Database as a Framework for a Cellular CAD System

In order to achieve the goal of a computing and software system that can achieve whole cell simulation, something like a CAD (Computer Aided Design) system must be built. CAD systems are used in engineering, for example, in the design of electronic circuitry. In biology, such a system could be used for the representation and possible design of cellular circuitry. Unlike engineering, the biological CAD system could be used backwards as a reverse-engineering tool to understand the complexity of cellular life. This system would have a detailed representation of biochemistry sufficient to allow output of a data description of a snapshot of a living cell to a simulation, data mining or

visualization system. In engineering, CAD systems are database driven software, and the utility of a particular CAD system is proportional to the content of its database of parts, the symbols used to describe electronic components. Likewise, a biological CAD system will require a complete list of parts as an integrated software and database system. The fragmentation in the Bioinformatics parts list community must obviously be resolved to achieve such a list (Stein, 2002). Federated databases with highly latent network interconnections and imprecise data models will not suffice for large cellular information systems. Interaction and parts information is best stored as an integrated system in order to meet the data demands of whole-cell simulation, visualization and data mining. Overall, such a system requires a formal data model for molecular interactions that provides a good abstraction of the data with precise computability without sacrificing complexity of the information. The emergence of a standard will allow diverse groups to collaborate and work towards their common goals more efficiently.

BIND – The Biomolecular Interaction Network Database

The Biomolecular Interaction Network Database (BIND) has been designed as a system to store biomolecular interactions possessing the positive attributes of an interaction database discussed above. BIND is a web-based database system that is based on a data model written in ASN.1 (Abstract Syntax Notation - http://asn1.elibel.tm.fr/). ASN.1 is a hierarchical data description language used by the NCBI to describe all of the data in PubMed, GenBank, MMDB and other NCBI resources (Benson et al., 2002). ASN.1 is also used extensively in air traffic control systems, international telecommunications and Internet security schemes. The advantages of ASN.1 compared to other computer readable data description languages such as XML include being strongly typed and having an efficient cross-platform binary encoding scheme that saves space and CPU resources when transmitting data. Disadvantages are that commercial ASN.1 tools are very expensive and that the ASN.1 standard process is closed. The NCBI, however, provides public domain cross-platform software development toolkits written in the C and C++ languages to deal with the NCBI data model and with ASN.1. Each toolkit can read an ASN.1 defined data model and generate C code that allows

automatic reading (parsing), writing and management of ASN.1 objects. Also supported is the ability to automatically translate ASN.1 defined objects to and from XML as well as the automatic generation of an XML DTD for an ASN.1 data specification. These toolkits are currently available at ftp://ncbi.nlm.nih.gov/toolbox/. This powerful data description language and toolkit combination allows us to circumvent the large and time-consuming problem in Bioinformatics of parsing primary databases to integrate data for effective research use. With the toolkit, parsing is automatic through the use of machine generated parsing code. The use of ASN.1 also allows the BIND specification to use mature NCBI data types for biological sequence, structure and publications.

Recently, the XML (Extensible Markup Language - http://www.w3.org/XML/) language has gained popularity for data description. XML matches ASN.1 in its ease of use, although it does not provide strong types. For instance, ASN.1 recognizes integers and can validate them, while XML treats numerical data as text. The advantages of XML are its open nature and familiarity, since it is similar to HTML. Many tools currently use XML, although free code-generation and rapid application development tools are only beginning to mature. XML also wastes space because it does not have a binary encoding scheme and because it is tag based (Figure 1). An XML message will be many times larger than a binary encoded ASN.1 message. In the future, the XML Schema standard (http://www.w3.org/XML/Schema) will partially solve some of the problems of XML, such as lack of strong types, and will likely mature enough to be considered a replacement of ASN.1 because of wider commercial and development community support.

A)

```
Date ::= CHOICE {
     str VisibleString,
     std Date-std
}

Date-std ::= SEQUENCE {
     year INTEGER,
     month INTEGER OPTIONAL
}
```

B)

```
date
     std {
          year 1974 ,
          month 7 ,
          day 7
     }
```

C)

```
<Date>
     <Date_std>
          <Date-std>
               <Date-std_year>1974</Date-std_year>
               <Date-std_month>7</Date-std_month>
               <Date-std_day>7</Date-std_day>
          </Date-std>
     </Date_std>
</Date>
```

Figure 1: Examples of ASN.1 and Equivalent XML

A) An example of how a date data type is specified in ASN.1. B) An example of how an instance of specific date data is represented in the print form of ASN.1. The BER binary encoded form of this ASN.1 would only take up less than 20 bytes. C) An example of how the same date data as in B) is represented in XML. XML does not have a binary encoded form. Note the excess of information required to specify a date.

The BIND data specification describes biomolecular interaction, molecular complex and molecular pathway data. Both genetic and physical interactions can be stored. Chemical reactions, photochemical activation and conformational changes can be described down to the atomic level of detail. Everything from small molecule biochemistry to signal transduction is abstracted in such a way that graph theory methods may be applied for data mining. The database can be used to study networks of interactions, to map pathways across taxonomic branches and to generate models for kinetic simulations. The database can be visually navigated using a Java applet and queried using a text search or the BLAST against BIND service. BIND is an open effort; all records are in the public domain and source code for the project is made freely available under the GNU Public License. Users are encouraged to submit their favorite interactions. BIND has been used to manage and automatically discover new knowledge residing in large yeast protein-protein and genetic interaction networks in *Saccharomyces cerevisiae* determined using mass-spectrometry, phage-display, yeast two-hybrid and roboticized synthetic lethal screens.

Other Molecular Interaction Databases

Most molecular interaction data resides in the scientific literature, in unstructured text, tables and figures in thousands of papers in molecular and cellular biology. It is currently impossible to retrieve information from this archive using computational tools such as natural language query methods with the accuracy required by scientists. Many databases currently available, mainly over the Internet, contain interaction information, although most of these databases are not focused on storing biomolecular interactions. Most of these databases are small and have very select niches of interaction information, for example, the restriction enzyme database REBase (Roberts and Macelis, 2001) maintained by New England Biolabs, while not an interaction database *per se* does contain interactions between restriction enzymes and the specific patterns of DNA that they recognize and cleave. These protein-DNA interactions satisfy the node and edge criterion of the graph abstraction of interaction data and are thus a very valuable source of information.

Examples of Interaction Databases

Examining both the literature and the Internet results in a large and varied list of databases that contain interaction information covering proteins, DNA, RNA and small molecules. The number of projects indicates the importance of this data. However, the variety of data representation paradigms, file formats, data architectures and license agreements is a daunting challenge to integration of this information into a common scheme. One can classify databases according to whether they are linked back to primary experimental data in the literature, or are secondary sources of information based on review articles or the knowledge of expert curators. The databases based on primary information are few in this list, yet are amongst the most valuable.

The following database review highlights whether the data is present in a machine-readable form. Many databases are packed with information, but the information is entered in such a way that it cannot be unambiguously matched to other databases. For example some databases are missing key data descriptors like sequence accession numbers, CAS chemical compound numbers, PubMed identifiers for publication references, or unambiguous taxonomy information when data from multiple organisms is present. This impedes the usefulness of the information, since it is difficult to tie it to other knowledge, which is required on a large scale for it to be mined and more broadly understood. It is critical that these projects move towards sound database principles when describing data such that it may be computed upon unambiguously and precisely. Where possible, the primary reference and license terms of the database to academic and industrial users of the data is listed to aid future data integrators when choosing databases to import into a data warehouse.

Aminoacyl-tRNA Synthetase Database

URL: http://rose.man.poznan.pl/aars/index.html
Ref: (Szymanski and Barciszewski, 2000)

Contains aminoacyl-tRNA synthetase (AARS) sequences for many organisms. This database is simply a sequence collection, but collated pairs of AARS + tRNA can be used to create RNA-protein interaction records. The database is available freely over the web.

ASEdb (Alanine Scanning Energetics Database)

URL: http://www.asedb.org
Ref: (Thorn and Bogan, 2001)

ASEdb is a database of protein sidechain interaction energetics determined by alanine-scanning mutagenesis manually curated by a single group. The database is not very large, but does provide valuable information on proteins binding with other molecules, mainly other proteins. This is derived from alanine scanning mutagenesis followed by a measurement of the change in free energy of binding that the mutation caused. The database is web-based and text searchable, but only contains a few specialized database fields.

BBID (Biological Biochemical Image Database)

URL: http://bbid.grc.nia.nih.gov/

The BBID is a searchable database of images from publications about cellular pathways and other biological relationships. It focuses on signal transduction pathways. The molecules in the figures and the publications that the figures are from are indexed in a database that allows searching the figures. While molecular interaction information is

available in the figures, it is not extracted in a machine-readable fashion, thus BBID remains a human reference only and cannot be computed upon.

BindingDB (The Binding Database)

URL: http://www.bindingdb.org/

Ref: (Chen et al., 2001a; Chen et al., 2001b; Chen et al., 2002)

BindingDB is a public, web-based database containing kinetic and thermodynamic binding constants for interacting biomolecules. The data is only from isothermal titration calorimetry and enzyme inhibition experimental methods, but may include data from other methods in the future. The database is rigorously designed and implemented using the latest database technology. The search interface is very advanced and even allows searching for small molecules that are similar to an input structure. While it does contain information about biomolecular interactions, the data specification is focused on binding constant information and experimental method description for two specific methods.

Biocarta

URL: http://www.biocarta.com/

Biocarta is a commercial venture whose purpose is to provide manually created clickable pathway maps for signal transduction as a resource to the scientific community. The presence of a standard set of symbols to represent various different protein components of pathways make the pathway maps clear and easy to understand. Proteins are linked to many different primary databases including PubMed, GenBank, OMIM, Unigene (Wheeler et al., 2000), KEGG, SWISS-PROT (Bairoch and Apweiler, 2000) and Genecard. Companies may sponsor genes and links are present to commercially available reagents. Biocarta invites volunteer users to supply pathways as figures, and Biocarta then creates clickable linked maps and makes them available via the web. The

data model is not public and the database has not been published in peer-reviewed literature.

Biocatalysis/Biodegradation Database

URL: http://www.labmed.umn.edu/umbbd/
Ref: (Ellis et al., 2001)

Contains microbial biocatalytic reactions and biodegradation pathways primarily for xenobiotic, chemical compounds. Currently contains about 125 pathways, over 830 reactions, 750 compounds, 510 enzymes and 320 microorganisms are represented. The data model is chemical reaction based with a graph abstraction for pathways. The graph abstraction allows the 'Generate a pathway starting from this reaction' function. PDB files for some of the small molecules are available. Graphics (clickable GIFs) are available for the various pathways. The work is funded by several organizations and is free to all users. Data is entered on a volunteer basis and records contain literary references to PubMed.

BRENDA

URL: http://www.brenda.uni-koeln.de/
Ref: (Schomburg et al., 2002b; Schomburg et al., 2002a)

BRENDA is a database of enzymes. It is based on EC number and contains much information about each particular enzyme including reaction and specificity, enzyme structure, post-translational modification, isolation/preparation, stability and cross references to structure databanks. Information about the chemical reaction is extensive, but is in free-text form and thus is not machine-readable. The database is copyright and is free to academics. Commercial users must obtain a license.

BRITE (Biomolecular Reaction pathways for Information Transfer and Expression)

URL: http://www.genome.ad.jp/brite/

BRITE is a database of binary relations based on the KEGG system. It contains protein-protein interactions, enzyme-enzyme relations from KEGG, sequence similarity, expression similarity and positional correlations of genes on the genome. The database mentions that it is based on graph theory, but no path finding tools are present. BRITE contains some cell cycle controlling pathways that have now been incorporated into KEGG.

COMPEL (Composite Regulatory Elements)

URL: http://compel.bionet.nsc.ru/
Ref: (Kel-Margoulis et al., 2000)

Contains protein-DNA and protein-protein interactions for Composite Regulatory Elements (CREs) affecting gene transcription in eukaryotes including the positions on the DNA that the protein binds to. The database is organized in a fielded flat-file format and provides links to TRANSFAC. The data model does not use a graph theory abstraction. COMPEL 3.0 in January 1999, contained 178 composite elements.

COPE (Cytokines Online Pathfinder Encyclopedia)

URL: http://www.copewithcytokines.de/

COPE is an encyclopedia of cytokines and related biological terms. COPE provides a free-text textbook like entry describing each of the many terms and a dictionary for term definitions. Protein and other biomolecular interactions relating to the terms in the encyclopedia are described. The database can be browsed and searched using keywords but contains no formal data model and is thus not natively machine-readable.

CSNDB (Cell Signaling Networks Database)

URL: http://geo.nihs.go.jp/csndb/
Ref: (Takai-Igarashi et al., 1998)

CSNDB contains cell signaling pathway information in *Homo sapiens*. It has a data model that is specific only to cell signaling and is constructed on ACeDB (Eeckman and Durbin, 1995). It is based on both interactions and reactions and stores information mainly as unstructured text in fields within a structured record. The data model is sound and some fields contain controlled vocabulary. An extensive graph theory abstraction is present. It is probably one of the first databases to use a simple graph theory abstraction since its first publication in 1998 and contains the most elaborate pathway finder using shortest path algorithms. It can limit the graph to a specific organ and can mask sub-trees for this feature. Fields have been added as they are needed and the system is not general. CSNDB contains interesting pharmacological fields for drugs, such as IC50. The database can represent proteins, complexes and small molecules. It is linked to PubMed and TRANSFAC. Recently, TRANSFAC has imported the CSNDB to seed its TRANSPATH database of regulatory pathways that link with transcription factors. Contains an extensive license agreement that limits corporate use. Free to academics. Funded by the Japanese National Institute of Health Sciences.

Curagen Pathcalling

URL: http://curatools.curagen.com/

The commercial Curagen Pathcalling program visualizes information from the Stanley Fields lab high-throughput yeast two-hybrid screen of the yeast genome along with other yeast protein-protein interaction from the literature. It contains only protein-protein interactions. Pathcalling uses a graph theory abstraction that allows the use of a Java applet to visually navigate the database. Each protein may be linked to SGD (Ball et al., 2000), GenBank or SWISS-PROT. Because it is proprietary, the database does not make any of its information, software or data model available.

DIP (Database of Interacting Proteins)

URL: http://dip.doe-mbi.ucla.edu
Ref: (Xenarios et al., 2000; Xenarios et al., 2002)

The DIP database stores only protein-protein interactions and recently began storing chemical actions and chemical states of those proteins. It is based on a binary interaction scheme for representing interactions and uses a graph abstraction for its tools. A visual navigation tool is present. DIP does not use a formal grammar for its data specification. The DIP data model allows the description of the interacting proteins, the experimental methods used to determine the interaction, the dissociation constant, the amino acid residue ranges of the interaction site and references for the interaction. DIP contains over 17,500 protein-protein interactions representing about 110 different organisms. Academic users may register to download the database for free if they agree to the click-through license. Commercial users must contact DIP for a license.

DRC (Database of Ribosomal Cross-links)

URL: http://www.mpimg-berlin-dahlem.mpg.de/~ag_ribo/ag_brimacombe/drc
Ref: (Baranov et al., 1999)

This database keeps a collection of all published cross-linking data in the *E. coli* ribosome. This is a database of hand-curated dBASE IV files with a web interface (last updated March 7th, 1998). The possibilities for machine parsing the database seem limited since the field data is non-standardized and meant to be human-readable only.

DPInteract

URL: http://arep.med.harvard.edu/dpinteract/
Ref: (Robison et al., 1998)

DPInteract is a curated relational database of *E. coli* DNA binding proteins and their target genes. Provides BLASTN searching for DNA. Has links to SWISS-PROT,

EcoCyc, PubMed and Prosite (Hofmann et al., 1999). The database is text based with a limited data specification. Interestingly, position specific matrices are available to describe the DNA binding motif. Records are organized by protein structure family (e.g. Helix-turn-helix family proteins). Updating of the database continued from 1993-1997 and has now stopped. The database is copyright, but is freely available over the web and contains information about 55 *E. coli* DNA binding proteins with known binding sites.

EcoCyc (and MetaCyc)

URL: http://biocyc.org/

Ref: (Karp et al., 1999; Karp et al., 2002b; Karp et al., 2002a)

EcoCyc is a private database (freely available to academics) that contains metabolic and signaling pathways from *E. coli*. EcoCyc is one of the oldest pathway databases. It is based on an object-oriented data model. Chemical reactions are used to describe the data, which is intuitive in this case, since EcoCyc's main goal is to catalogue metabolic pathways from *E. coli*. It is currently being retrofitted to deal with protein-protein interactions in cell signaling pathways, although data is still described using a chemical reaction scheme. The fields of this database are mostly free text based. All types of molecules from small molecules to molecular complexes may be represented. The data model is not based on a chemical graph, however, and atomic level detail is not present. EcoCyc uses a graph abstraction model that has allowed pathway traversing and visualization tools to be written. EcoCyc contains interactions of proteins with proteins and small molecules. MetaCyc contains EcoCyc and also pathways from over 150 other organisms. Recently, BioCyc has been created to contain EcoCyc, MetaCyc and computationally derived pathway databases for recently sequences genomes, similar to the WIT project.

EMP (Enzymes and Metabolic Pathways Database)

URL: http://www.empproject.com/
Ref: (Selkov et al., 1996)

EMP is an enzyme database that is chemical reaction based. It stores information as detailed as chemical reaction and k_m. Over 300 fields are stored as semi-structured text that may allow most of the database to be easily machine-readable. The database is part of the WIT project and can also be accessed from the WIT system. GIF and SVG images of many pathways are available and the project is heavily curated. Recently this project underwent a major website reorganization and is now very user friendly and easily searchable. Some source code is available for the project via a CVS server and the database is freely available over the web.

ENZYME

URL: http://www.expasy.ch/enzyme/
Ref: (Bairoch, 2000)

This database contains enzyme, substrate, product and cofactor information for over 3,850 enzymes. It has been a crucial resource for metabolic databases including EcoCyc. It is chemical reaction based. This database can be translated to an interaction model by breaking down the chemical reactions into substrate-enzyme, product-enzyme and cofactor-enzyme groups. ENZYME links to BRENDA, EMP/PUMA, WIT and KEGG. The database is free and is run by the not-for-profit Swiss Institute of Bioinformatics. There are no restrictions on its use by any institutions as long as its content is in no way modified.

FIMM (Functional Molecular Immunology)

URL: http://sdmc.krdl.org.sg:8080/fimm/

Ref: (Schonbach et al., 2000)

The FIMM database contains information about functional immunology. It is primarily not an interaction database, but it contains information about major histocompatibility complex (MHC)/ Human Leukocyte Antigen (HLA) associated peptides, antigens and diseases. The database contains over 1,400 peptides and almost 1,400 HLA records at time of writing. It is linked to GenBank, SWISS-PROT, MHCPEP, OMIM, and PubMed, among others. This data provides protein-peptide interaction records that are important immunologically and some records contain HLA class I structure models. The database is provided 'as-is' by Kent Ridge Digital Labs in Singapore.

FlyNets

Ref: (Sanchez et al., 1999)

FlyNets is now defunct, but originally stored information about molecular interactions (protein-DNA, protein-RNA, protein-protein interactions) and genetic interaction networks in the fruit fly, *Drosophila melanogaster*, focusing on developmental pathways. Information was linked to PubMed and FlyBase. Version 3.0 was available in May 1999 and contained 200 interactions. FlyNets was based on a graph abstraction and provided a visual graph navigation tool to draw networks from the database.

GeneNet (Genetic Networks)

URL: http://wwwmgs.bionet.nsc.ru/systems/mgl/genenet/
Ref: (Kolpakov et al., 1998; Kolpakov and Ananko, 1999)

GeneNet describes genetic networks from gene through cell to organism level using a chemical reaction based formalism, i.e. substrates, entities affecting course of reaction and products. The database is based on a formal object-oriented data model. GeneNet contains 23 gene network diagrams and over 1,000 genetic interactions (termed relations in GeneNet) from varied organisms including human. The database is current and is regularly updated. Visual tools are present for examining and querying the pathway data in the context of a simple diagram of a cell, but are plagued by network latency problems that can prevent complete loading.

GeNet (Gene Networks Database)

URL: http://www.csa.ru/Inst/gorb_dep/inbios/genet/genet.htm
Ref: (Serov et al., 1998)

GeNet curates genetic networks for a few example species. It provides Java visualization tools for the genetic networks. The database contains extensive information about each example network in free-text form. This database is not machine-readable, although is a good genetic interaction resource.

HIV Molecular Immunology Database

URL: http://hiv-web.lanl.gov/immunology/index.html
Ref: (Korber et al., 1998)

This database contains information about binding events between HIV and the immune system including HIV epitope and antibody binding sites that could provide data for an interaction database. HLA binding motifs are included that allow prediction of

HLA-peptide interactions. This information is freely available from the database's FTP site.

HOX Pro

URL: http://www.iephb.nw.ru/hoxpro
Ref: (Spirov et al., 2000)

The main purpose of this database is to provide a curated human readable resource for homeobox genes. It also stores extensive information about genetic networks of homeobox genes in a few model organisms. Clickable picture and a Java applet are available to visualize the networks. The visualization system is the same one used for GeNet.

InBase (The Intein Database)

URL: http://www.neb.com/neb/inteins.html
Ref: (Perler, 2000)

The main purpose of this database is to be a curated resource for protein splicing. The database contains descriptions of intein proteins (self-catalytic proteins) that are good examples of intramolecular interactions. The database records are present in a machine-readable format. Each record could be used by an interaction database to generate intramolecular interaction records containing chemical reaction description using information about the mechanism of protein splicing present on the InBase website.

Indigo

URL: http://195.221.65.10:1234/Indigo/

Indigo contains information of codon usage, operons, gene neighbors and metabolic pathways for *Escherichia coli* and *Bacillus subtilis*. The metabolic pathway

information contains information about enzymatic reactions and can be accessed using clickable images in a Java applet. Enzyme names in the pathway map are linked back to primary sequence databases. The database is difficult to use and slow because of the overhead of the Java query tool.

Interact

Ref: (Eilbeck et al., 1999)

Interact is an object oriented protein-protein interaction database based on Java and the POET database (www.poet.com) that is now defunct. It has a formal data-model that describes interactions, molecular complexes and genetic interactions. It stores information about experimental method and is based on an object-oriented description of proteins and genes. The database does not provide other details about the interaction and the underlying description of genes and proteins is simplified compared to that of GenBank. The database is not publicly available, but the object-oriented design approach has been described in the literature. The database contains over 1,000 interactions.

ICBS (Inter-Chain Beta-Sheets)

URL: http://www.igb.uci.edu/servers/icbs/

Ref: (Baisnee et al., 2002)

ICBS contains protein-protein interactions mediated by beta-sheets taken from the PDB database. The database contains over 2600 PDB structures that contain protein complexes mediated by this type of interaction. Basic information about each PDB file is provided as well as detailed physical and structural information about the beta sheets at the interaction interface. This database is similar to MMDBind, but is more limited.

JenPep

URL: http://www.jenner.ac.uk/JenPep/
Ref: (Blythe et al., 2002)

JenPep is a peptide binding database that contains more than 8,000 peptide-protein interactions for MHC Class I, II, CD8 and CD4 T cells and TAP (Transport of Antigen) complex. All information in JenPep, such as IC50 and peptide origin, is from published experiments. Peptide epitopes can be searched over the web using a simple query interface.

KEGG (Kyoto Encyclopedia of Genes and Genomes)

URL: http://www.genome.ad.jp/kegg/
Ref: (Kanehisa et al., 2002)

KEGG represents most of the known metabolic pathways and some of the regulatory pathways as graphical diagrams that are manually drawn and updated. Each of the metabolic pathway drawings is intended to represent all chemically feasible pathways for a given system. As such, these pathways are abstractions onto which enzymes and substrates from specific organisms can be mapped. KEGG does not explicitly represent specific biomolecular interactions, however, the pathway representations are a valuable source of information for someone assembling pathway information from interaction records. The database is machine-readable, except for the pathway diagrams. Each enzyme entry contains a substrate and a product field that can be used to translate between the chemical reaction description scheme and a binary interaction scheme. The KEGG project distributes all databases freely for academics via FTP. KEGG is one of the best freely available resources of metabolic and small molecule information (the LIGAND database).

Kohn Molecular Interaction Maps

URL: http://discover.nci.nih.gov/kohnk/interaction_maps.html
Ref: (Kohn, 1999)

Kohn molecular maps represent one researchers attempt to create a standard for representing biochemical pathways and molecular interactions using a symbolic language similar to electronic circuit diagrams. Kohn created detailed maps of the mammalian cell cycle control and DNA repair systems as an example. The maps are pictures only and thus are not machine-readable, although they do have a grid system as in normal street maps. A separate annotation list is provided that allows mapping of molecules from the list of the map using the coordinate system. The ideas represented in these maps are useful for further research on pathway visualization systems and the initial two maps provide a resource for manual extraction of molecular interaction information.

MDB (Metalloprotein Database)

URL: http://metallo.scripps.edu/
Ref: (Castagnetto et al., 2002)

MDB contains the metal-binding sites from entries in the PDB database. The database is based on open source software and is freely available. The data is present down to the atomic level of detail. An extensive Java applet is available to query and examine the data in detail. Ad-hoc queries of the database using SQL are available and tools are being developed to predict a metal binding site in a given protein structure.

MHCPEP

URL: http://wehih.wehi.edu.au/mhcpep/
Ref: (Brusic et al., 1998)

MHCPEP is a database comprising over 13,000 peptide sequences known to bind MHC molecules compiled from the literature and from direct submissions. It has not

been updated since mid-1998. While this database is not a typical interaction database, it provides peptide-protein interaction information relevant to immunology. The database is freely available via FTP in a text based machine-readable format.

MINT (Molecular Interaction Database)

URL: http://cbm.bio.uniroma2.it/mint/
Ref: (Zanzoni et al., 2002)

MINT is a database of molecular interactions gathered from the literature and manually input. Apart from a simple relational schema to store binary relations, MINT can store some protein post-translational modifications, experiments, cellular location, pathways and complexes. MINT contains more than 3,800 binary interactions and only a handful of complexes. An extensive graph abstraction is present which allows the use of a graphical Java viewer for the interactions. Interestingly, the size of the molecules is represented relative to each other in the visualization so that heavier proteins are drawn as larger circles.

MIPS Comprehensive Yeast Genome Database

URL: http://mips.gsf.de/proj/yeast/
Ref: (Mewes et al., 2002)

The MIPS Comprehensive Yeast Genome Database (CYGD) presents a database that summarizes the current knowledge regarding the more than 6,200 ORFs encoded by the Yeast Genome. This database is similar to SGD and YPD and is not primarily an interaction database. The MIPS center, however, makes available large tables for direct protein-protein interactions as well as genetic interactions in yeast free for download at http://mips.gsf.de /proj/yeast/tables/interaction/index.html. Each interaction contains an experimental method used and usually contains a literature reference. Manually created clickable pathway maps are also available for various metabolic and regulatory pathways in yeast. The MIPS Yeast Genome Database uses a relational model, but most fields use

unstructured text. For example, the experimental method used to determine the interaction field is unstructured and the same experimental type may be represented in many different ways. This makes the database difficult to parse with a computer, but the CYGD is an extremely useful resource for yeast protein-protein interaction information. Recently, MIPS has made available a protein-protein interaction, complex and genetic interaction query tool for searching this data.

MMDB (Molecular Modeling Database)

URL: http://www.ncbi.nlm.nih.gov/Structure/
Ref: (Wang et al., 2002)

This database is an NCBI resource that contains all of the data in the PDB database in ASN.1 form. The MMDB validates all PDB file information and describes all atomic level detail data explicitly and in a formal machine-readable manner. While this database is not an interaction database, it does contain atomic level detail of molecular interactions present in some records that describe molecular complexes. Sequence linkage is improved and MMDB is easily accessed by machine-readable methods that can obtain information about molecular interactions. MMDB is in the public domain and all software and data is freely available to academics or corporations.

NetBiochem

URL: http://www.auhs.edu/netbiochem/NetWelco.htm

NetBiochem is primarily an education resource that focuses on teaching detailed biochemistry of specific metabolic pathways, such as fatty acid metabolism at the level of an introductory biochemistry course at a university. There is no formal data model, but the available pathways represent a good collection of different ways of presenting biochemical pathway data to an untrained audience. Thus, this site would be useful as a resource for curators to enter data into a molecular interaction database and as a source of ideas for pathway visualization research.

ooTFD (Object Oriented Transcription Factors Database)

URL: http://www.ifti.org/
Ref: (Ghosh, 2000)

The ooTFD contains information on transcription factors from various organisms including transcription factor binding sites on DNA and transcription factor molecular complex information. Thus it contains protein-DNA and protein-protein interactions. The database is based on a formal machine-readable object-oriented format and is available in numerous forms. The database contains thousands of sites and transcription factors and is freely available (including software) from http://ncbi.nlm.nih.gov/repository/TFD/.

ORDB (Olfactory Receptor Database)

URL: http://senselab.med.yale.edu
Ref: (Crasto et al., 2002)

The ORDB is primarily a database of sequences of olfactory receptor proteins. It contains a section on small molecule ligands that bind to olfactory receptors. About 80 ligand-protein interactions are present in the database with about 40 small molecules. Structures of these small molecules are available as well.

PATIKA (Pathway Analysis Tool for Integration and Knowledge Acquisition)

URL: www.patika.org
Ref: (Demir et al., 2002)

PATIKA is a combination of a Java pathway modeling tool and an object-oriented pathway database. A data specification is present using a state and transition notion for pathway descriptions. This data model combines elements from BIND, EcoCyc and Petri Nets. Interestingly, the data model allows multiple levels of abstraction to allow the description of cellular events when not all of the details are known. For instance, a

transition can describe the change of one state to another and that state can be very detailed chemically or be a very general cellular state. The Java tool allows one to build pathways and query the database remotely over the Internet. The data model is currently quite simple and is only designed to store human pathway information.

PFBP (Protein Function and Biochemical Networks Project)

URL: http://www.ebi.ac.uk/research/pfmp/
Ref: (van Helden et al., 2000)

The aim of the PFBP is to describe metabolism, gene regulation, molecular transport and signal transduction in the aMAZE database. PFBP is based on a formal object oriented data model that will be integrated with CORBA. The database is chemical reaction based, was started by describing metabolic pathways only and was seeded from data from BRENDA (Schomburg et al., 2002b). PFBP uses a graph abstraction for the interaction data and can describe chemical reactions and pathways. This has allowed pathway finding and visualization tools to be implemented. The aMAZE database has an extensive web site describing it, but is not yet publicly available over the web.

PhosphoBase

URL: http://www.cbs.dtu.dk/databases/PhosphoBase/
Ref: (Kreegipuu et al., 1999)

This database contains information on kinases and phosphorylation sites. The phosphorylation sites are stored along with kinetics information and references for each kinase. While this is not an interaction database directly, information is present about protein-protein interactions involved in cell signaling and their chemistry. Recently, a neural network based phosphorylation site prediction tool has been made available.

PIMRider (Protein Interaction Map - Hybrigenics)

URL: http://pim.hybrigenics.com/pimriderlobby/current/PimRiderLobby.htm

PIMRider is a graphical Java applet based protein interaction network visualization tool that is driven by a database of protein-protein interactions. All interactions have been determined using the sequence fragment (domain) based two-hybrid screen experimental approach by the Hybrigenics company. All of the data and the data model is proprietary and is only partially publicly available. The PIM database contains information on *Helicobacter pylori*, HIV (Human Immunodeficiency Virus), HCV (Hepatitis C virus) and *Homo sapiens*.

PIMdb (Drosophila Protein Interaction Map Database)

URL: http://cmmg.biosci.wayne.edu/finlab/PIMdb.htm

PIMdb is a collection of two-hybrid generated protein-protein interactions for *Drosophila melanogaster*. A single lab is generating this data and the data is currently unpublished. A simple binary interaction data model is used to store the information. Presently, PIMdb does not make available any query tools, but is rather just a manually created list of experimental results from one academic research group. Without peer review, the quality of this data is in question. The group asks that they be contacted if any data will be used for other projects.

ProChart (Axcell)

URL: http://www.axcellbio.com/products.asp

The ProChart database is sold by Axcell Biosciences and contains proprietary data on protein-protein interactions garnered using Axcell's proprietary experimental methods. No part of the database or data model is publicly accessible or has been published.

ProNet (Myriad Genetics)

URL: http://www.myriad-pronet.com/

This commercial database provides protein-protein interaction information to the public from Myriad Genetics proprietary high-throughput yeast two-hybrid system for human proteins and from published literature. Each protein record describes interacting proteins and a Java applet is available to navigate the database. The database stores only protein interaction information with links to primary sequence databases and PubMed. It uses a graph abstraction to display the interactions. The database is fully proprietary and has not been published.

REBASE

URL: http://rebase.neb.com
Ref: (Roberts and Macelis, 2001)

REBASE is a comprehensive database of information about restriction enzymes and related proteins, such as methylases. While it is not an interaction database, restriction enzymes and methylases take part in specific DNA-protein interactions. REBASE describes the enzyme and the recognition site, thus can be used to create binary interaction records with chemical actions. Useful links are present to commercially available enzymes. REBASE is freely available in many different formats to the academic community.

Relibase

URL: http://relibase.ebi.ac.uk/
Ref: (Hendlich, 1998)

Relibase is a software query tool that allows powerful searches to be conducted on PDB entries containing protein-ligand interactions, where a ligand is anything that is not a protein. DNA and RNA are also considered ligands, but are ignored in searches.

The purpose of Relibase is to help examine small molecules, such as therapeutics, that are currently in the PDB binding to proteins. Full crystal structure and binding sites of ligands are available. The database may be searched by text, sequence, SMILES strings and 2-D/3-D small molecule structures. The Relibase project is currently run by the Cambridge Crystallographic Data Centre, which makes the tool available over the web.

RegulonDB

URL: http://www.cifn.unam.mx/Computational_Genomics/regulondb/
Ref: (Salgado et al., 2001)

RegulonDB is mainly an *E. coli* operon database, although it does contain protein-DNA interactions (e.g. ribosome binding sites and promoters) and protein complexes. The database is free for non-commercial use. Commercial users require a license.

SELEX_DB

URL: http://wwwmgs.bionet.nsc.ru/mgs/systems/selex/
Ref: (Ponomarenko et al., 2002)

SELEX_DB is a curated resource that stores experimental data for functional site sequences obtained by using SELEX-like random sequence pool technologies to study interactions. The database contains interactions, including binding sites, between random DNA sequences and various types of ligands, most of which are proteins. It is available over the web and via SRS and the records are available in a machine-readable flat-file format.

SoyBase

URL: http://soybase.ncgr.org/

SoyBase is an ACeDB (Eeckman and Durbin, 1995) database that contains information about the soybean, including metabolism. Metabolic pathways are based on a chemical reaction abstraction. SoyBase contains over 850 automatically generated diagrams of metabolic pathways covering over 1,500 enzymes and over 1,200 metabolites. Clicking on an enzyme or ligand on the diagram triggers a query for that molecule in the database. SoyBase is based on a formal machine-readable data model, as is any AceDB installation and is available over the web.

SPAD (Signaling Pathways Database)

URL: http://www.grt.kyushu-u.ac.jp/eny-doc/

SPAD provides clickable image maps for a handful of pathways. Clicking on an element of the pathway diagram links to sequence information of the protein or gene. Protein-protein and protein-DNA interactions are covered with respect to signal transduction. The database does not have a formal data model. SPAD has not been updated since 1998 but still gives useful overviews of the pathways it contains.

SPIN-PP (Surface Properties of Interfaces – Protein-Protein Interfaces)

URL: http://trantor.bioc.columbia.edu/cgi-bin/SPIN/

SPIN-PP is a database of all protein-protein interfaces in the PDB. Molecular surfaces are organized in a taxonomy based on surface curvature, electrostatic potential, sequence variability and hydrophobicity. SPIN-PP contains 855 protein-protein interfaces and is searchable by PDB code and the various surface structural properties listed above. Surfaces of interest can be viewed using the GRASS server(Nayal et al., 1999). The database does not seem to have been updated regularly since 1999, but is freely available.

STKE (Signal Transduction Knowledge Environment)

URL: http://www.stke.org/
Ref: (Gough and Ray, 2002)

STKE is a curated resource for signal transduction information. It provides a manually created clickable picture of various signal transduction pathways linked to primary database, the Connections Map. The data model is based on an upstream and downstream components view, which is a graph abstraction. Database fields are unstructured and thus are not machine-readable. STKE is available via a paid subscription to Science magazine.

SYFPEITHI

URL: http://www.uni-tuebingen.de/uni/kxi/
Ref: (Rammensee et al., 1999)

SYFPEITHI is a database of MHC ligands and peptide motifs. It contains over 3,500 peptide sequences known to bind class I and class II MHC molecules. All entries have been compiled from the literature. While this database is not a typical interaction database, it provides peptide-protein interaction information relevant to immunology.

TRANSFAC

URL: http://transfac.gbf.de/TRANSFAC/
Ref: (Wingender et al., 2001)

TRANSFAC is a database of transcription factors containing genomic binding sites and DNA-binding profiles. As such, it is not a typical interaction database, but it does contain protein-DNA interactions. A transcription factor DNA-binding site prediction tools is available. TRANSFAC is freely available to academics for download via FTP and is based on a formal relational database model.

TRANSPATH

URL: http://transfac.gbf.de/TRANSFAC/
Ref: (Wingender et al., 2001)

TRANSPATH is an effort underway at TRANSFAC to link regulatory pathways to transcription factors. The database is based on a chemical reaction view of interactions and contains a strong graph abstraction. Graph algorithms have been implemented to navigate the data. The data can describe regulatory pathways, their components and the cellular locations of those components. It can store information about various species. TRANSPATH includes all of the data from the CSNDB and it is obvious that TRANSPATH is using graph theory ideas from the CSNDB. TRANSPATH is free for academic users and can be downloaded in machine-readable XML format.

TRRD (Transcription Regulatory Regions Database)

URL: http://wwwmgs.bionet.nsc.ru/mgs/dbases/trrd4/
Ref: (Kolchanov et al., 2002)

TRRD contains information about regulatory regions including over 3,600 transcription factor binding sites (DNA-protein interactions). This database is very similar to TRANSFAC. It is available via an SRS database interface freely over the web.

WIT (What Is There?)

URL: http://wit.mcs.anl.gov/WIT2
Ref: (Overbeek et al., 2000)

WIT is a database project whose purpose is to reconstruct metabolic pathways in newly sequenced genomes by comparing predicted proteins with proteins in known metabolic networks. Predicted metabolic networks are stored in a chemical reaction based scheme with a graph abstraction. All information in the database may be queried

and pathways can be viewed as a computer generated diagram, which is hyper-linked back to the database.

YPD (Yeast Proteome Database – Incyte Genomics)

URL: https://www.incyte.com/proteome/index.html
Ref: (Costanzo et al., 2000)

This proprietary commercial curated proteome database effort by Incyte Inc. contains extensive information about all proteins in yeast. Extensive data about protein interactions, molecular complexes and sub-cellular location is present. Most of the database fields are free form text, but there is enough structure in the data model to make it amenable to machine reading of protein-protein interaction information. Incyte also makes available other proteomes for other model organisms including *Caenorhabditis elegans* and Human, but YPD is the most completely annotated. All of Incyte's proteome databases are proprietary and are available on a subscription basis.

Chapter 2 – BIND specification

The majority of the work presented in this chapter has been published as follows (reprinted with permission, copyright Oxford University Press):

Bader, G.D., Hogue, C.W.V.
BIND - A data specification for storing and describing biomolecular interactions, molecular complexes and pathways
Bioinformatics May 2000 16(5): 465-477

Abstract

Proteomics is gearing up towards high-throughput methods for identifying and characterizing all of the proteins, protein domains and protein interactions in a cell and will eventually create more recorded biological information than the Human Genome Project. Each protein expressed in a cell can interact with various other proteins and molecules in the course of its function. A standard data specification is required that can describe and store this information in all its detail and allow efficient cross-platform transfer of data. A complete specification must be the basis for any database or tool for managing and analyzing this information. A complete data specification has been defined in ASN.1 that can describe information about biomolecular interactions, complexes and pathways. This data specification is being used in the Biomolecular Interaction Network Database (BIND). An interaction record is based on the interaction between two objects. An object can be a protein, DNA, RNA, small molecule, molecular complex, photon or gene. Interaction description encompasses cellular location, experimental conditions used to observe the interaction, conserved sequence, molecular location, chemical action, kinetics, thermodynamics, and chemical state. Molecular complexes are defined as collections of at least one interaction that form a complex, with extra descriptive information such as complex topology. Pathways are defined as collections of at least one interaction that form a pathway, with additional descriptive information such as cell cycle stage. A request for proposal of a human readable flat-file format that mirrors the BIND data specification is also tendered for interested parties. The ASN.1 data specification for biomolecular interaction, molecular complex and pathway data is available at ftp://bioinfo.mshri.on.ca/pub/BIND/Spec/bind.asn. An interactive browser for this document is available via the web at http://bioinfo.mshri.on.ca/BIND/asn-browser/.

Introduction

Technological advances and mounting interest have pushed proteomics into the scientific spotlight. This growing field encompasses the study of proteins, both in

structure and in function, contained in a proteome - the protein equivalent of a genome. Because of increased interest and technique automation (Mendelsohn and Brent, 1999), the rate of proteomic data production is growing in a similar fashion as that of genomics a decade ago. For example, mass spectrometers, gene chips, and two-hybrid systems have made cellular signaling pathway mapping faster and easier and consequently these are becoming large producers of data. Protein-protein interaction and more general biomolecule-biomolecule (protein-DNA, protein-RNA, protein-small molecule, etc.) interaction information is being generated and recorded in the literature. Lessons from the genomic era have taught us that large amounts of related data recorded in scientific journals soon becomes unmanageable. A well designed common data specification based on a model of the biological information is therefore required to describe and store biomolecular interaction data.

Any well designed data specification for the storage and management of biomolecular interaction and biochemical pathway data should possess certain properties:

1. It should be able to describe all of the details of the biological data, from simple binary interactions to large-scale molecular complexes and networks of pathways and interactions. It must be possible to store protein, DNA, RNA, and other molecules in full atomic detail, since character based sequence abstractions of biomolecules often miss important chemical features, such as methylation on DNA or protein post-translational modification. This allows as much data as possible to be stored for scientific use in electronic form rather than in print.

2. It should be easily computable. A computer should be able to easily read, write and traverse the specification. This facilitates maintenance of a database of such information, creation of advanced queries and querying tools and development of computer programs that use the information for data visualization, data mining and visual data entry.

3. It should be platform and database independent. Tools written for one platform should be able to read data created on another platform directly. Any database management system should be able to handle the data structure without modification as well.

4. It should be succinct and easy for humans to understand. Field to data correspondence should be very clear and a human readable format of the specification should be available.

This paper describes a data specification for biomolecular interaction, molecular complex, and molecular pathway data that holds the above-mentioned properties. It has been designed for a database called BIND (Biomolecular Interaction Network Database) and has been written in a data specification language called Abstract Syntax Notation.1 (ASN.1) (http://www.oss.com/asn1/index.html). The U.S. National Center for Biotechnology Information (NCBI) uses ASN.1 to describe and store all of its biological and publication data and all of GenBank, MMDB and PubMed (Ostell and Kans, 1998). BIND inherits the NCBI data model, which provides a solid foundation for the BIND data specification through the use of mature NCBI data types that describe sequence, 3-D structure and publication reference information.

Although the specification is written in ASN.1, it is important to realize that it is not restricted to this syntax. The data structures can be readily translated to other common data specification languages such as CORBA IDL (Object Management Group, 1996) (Object Management Group, 1996) or XML (http://www.w3.org/XML) if the need arises. Aside from ASN.1, no other biological data specification is sufficiently rich in mature data types to use as a foundation for BIND without first building and testing those base data types.

With the BIND data specification, an effort was made to answer the question "Can complex cellular pathway information be efficiently represented in a computer?" BIND defines three main data types: interactions, molecular complexes, and pathways. Each of these objects is composed of various component and descriptor objects that are either defined in the specification proper or inherited from the NCBI ASN.1 data specifications. For example, an interaction record contains, among other data objects, two BIND-objects. A BIND-object describes a molecule of any type and is itself defined using simpler sub-objects. Normally, a BIND-object describing a biopolymer sequence will store a simple link to a sequence database, such as GenBank (Benson et al., 2002). If, however, the sequence is not present in a public database, it can be fully represented using an embedded NCBI-Bioseq object. The NCBI-Bioseq object is how NCBI stores

all of the sequences in GenBank and is a mature data structure. BIND also inherits the NCBI taxonomy model (also used and supported by EMBL, DDBJ and SWISS-PROT) and data, via an inherited NCBI-BioSource, and is designed so that interactions can be both inter- and intra-organismal. Sequence, structure, publication, taxonomy and small molecule databases have provided a strong foundation for BIND.

The Need for the BIND Specification

It is important to design well thought out methods for the electronic management of complex biological data, such as molecular interactions, now before the information becomes overwhelming for any one expert. This scenario has already occurred with current resources containing biomolecular sequence information such as GenBank or SWISS-PROT (Bairoch and Apweiler, 2000). It is becoming apparent that the complexity of genomics may be overshadowed by the complexity of molecular and, in particular, protein interactions in the cell. Of the 30,000 to 80,000 estimated human genes, only a small fraction encode classical "enzymes", perhaps only a few thousand. It is probable that most of the proteins encoded in the human genome are large, multi-domain molecules that participate in molecular interactions with other proteins, DNA, carbohydrates and small molecules. Thus it is not unreasonable to say that there are more protein-protein interactions than sequences (Marcotte et al., 1999).

Other interaction databases have been developed such as DIP (Xenarios et al., 2002), BRITE (http://www.genome.ad.jp/brite/), CSNDB (Igarashi and Kaminuma, 1997) and Interact (Eilbeck et al., 1999). Of these efforts, none are general for all biological molecular interactions and all lack a data specification that can handle the complexity and scale of the anticipated data. Even the GenBank/EMBL (Stoesser et al., 2002)/DDBJ (Tateno et al., 2002) feature table (DDBJ/EMBL/GenBank, 1997) contains space for recording interactions. Certain keys such as `misc_binding` allow a sequence submitter to enter and maintain interaction information within sequence records. Other standard feature table keys to indicate binding events are the `protein_bind` key used to annotate non-covalent protein binding sites on nucleic acid sequences, and the `RBS` key used to indicate a ribosome binding site. Each of these

feature table entries has only one single mandatory qualifier, `/bound_moiety="text"`, that allow the user to describe in plain text the bound moiety. There are other optional qualifiers that include `citation`, `db_xref`, and a series of free text fields that can be used to enter completely unformatted text data.

One problem in using these feature keys within sequence records is that this part of the specification is not suited to generate machine-readable information necessary to allow computer programs or individuals to explore the vast information space of interactions. Larger problems with the feature table are that it is DNA centric and thus poor for protein annotations and it does not fully represent the richness of the NCBI ASN.1 specification. Sequence depositors underutilized the feature tables as demonstrated for *Drosophila melanogaster* (Mohr et al., 1998). Features as described by GenBank/EMBL/DDBJ are not sufficient and not widely used, and it should not be expected that they be used, to capture molecular interaction information.

The BIND Data Model

This section describes the three main types of data objects in the BIND specification - interaction, molecular complex and pathway - as well as useful database management and data exchange objects. Explanations of the various objects in the specification are given along with examples. The specification will be explained as if it were being used to describe a single record in a database. The specification is available via FTP from ftp://bioinfo.mshri.on.ca/pub/BIND/Spec/bind.asn.

It is suggested that the reader follow the specification along with this paper. The data model is diagrammed in Figure 2 through to Figure 8 using UML (Unified Modeling Language, see http://www.rational.com/uml). Wherever possible, this specification is meant to reference information from other databases rather than storing the information as a copy. This avoids unnecessary duplication of information among databases and helps maintain data integrity (if the information in a referenced record in one database is updated, the other databases that reference the record are all automatically updated). All fields are non-optional unless stated otherwise.

An Object - A BIND-object

A BIND-object represents any chemical object - atom, molecule or complex of molecules and can be extended to represent any abstract object. A BIND-object contains:

1. *A short-label field to contain a short name for a molecule*. For example, ATP, IP3, S4 and HSP70 are acceptable short labels for ligands and proteins, respectively. Having a non-optional short label ensures that at least some descriptive data is entered for a molecule. This information is also useful to construct top-level descriptions regarding a particular record. For example, a simple description of an interaction between two proteins can be constructed using the short labels of the two BIND-objects in an interaction record. A graphical view of an interaction would be labeled with the short label field.

2. *A sequence of strings in the other-names field to contains synonyms for a molecule.* This field is optional, but is required to deal with normal genetic nomenclature. For instance, there are over 19,000 different gene names for only about 6,200 genes from the budding yeast *Saccharomyces cerevisiae*.

3. *A BIND-object-type-id object to contain the type of the molecule and a reference to another database containing a record for that molecule*. In this way, for instance, large DNA records are referenced rather than duplicated. A molecule type may be `not-specified`, `protein`, `dna`, `rna`, `small-molecule`, `complex`, `gene` or `photon`. Molecules of unknown type may be stored by specifying the type of molecule as `not-specified`. This type requires no further data input.

Protein, DNA, RNA and gene types all require a BIND-id object. This object can store accession numbers to any other database. It has special fields `gi` or Geninfo and `di` or domain identifier for the NCBI Entrez system (Schuler et al., 1996) and a database of domains under development at the Samuel Lunenfeld Research Institute, respectively. Any other accession number or numbers/strings to reference records in other databases can be stored in a set of NCBI Seq-id's present in the data object. All fields in BIND-id are optional so molecules stored internally in a BIND record that are not present in other databases (and so do not have accession numbers) can be properly saved. If A and B are genes, the interaction is a genetic interaction. These are important, even though they are

not as exact as direct physical interactions, as they have provided a large amount of protein functional information in the literature.

Molecules of type `small-molecule` require a BIND-small-molecule-id object. This object can contain a reference to an internal small molecule database or any other small molecule database via a database name and an integer and/or character based accession number.

BIND-objects of type `complex` require an integer accession number to a BIND molecular complex record.

The `photon` choice requires a BIND-photon object, which stores the wavelength and intensity of electromagnetic radiation. This can be used for light-protein interactions as occurs with rhodopsin in the visual transduction pathway in the eye.

4. *A BIND-object-origin data structure*. This structure contains a choice of origin between `not-specified`, `org` or organismal, and `chem` or chemical. BIND-objects of unknown origin would have origin type `not-specified`. Chemical objects that are derived directly from organisms, such as DNA, would be specified to be origin type `org` and are required to be associated with an NCBI BioSource object. A BioSource object can contain much descriptive data about an organism and the biological source of a compound. It also contains a reference to a taxonomy database. This information can be entered automatically if a GI is known for a biological sequence molecule, since a BioSource is part of the NCBI Bioseq object that stores biological sequences in Entrez. If a GI is not given, a BioSource can be created.

Molecules derived purely from chemical means are of origin type `chem` and require a BIND-chemsource object. The BIND-chemsource object contains a set of names for the chemical, usually a common name and any synonyms, a SMILES string (Weininger, 1988), the chemical formula, molecular weight (a RealVal-Units object), a CAS registry number (http://www.cas.org) and a BioSource if this is a natural product, such as a small molecule synthesized by a specific type of plant. A SMILES string is a standard way of representing a molecule's structure using ASCII characters. Many chemistry computer applications are available to manipulate and use data of this type (http://www.daylight.com/). Three-dimensional structure of a molecule can be predicted from a SMILES string to a high degree of accuracy using commercial chemistry

applications such as Corina (Gasteiger, 1996) and others. A CAS number is a reference number to the information regarding a chemical compound in the Chemical Abstracts Service. This service contains data on 43,640,331 chemical compounds (as of October 23th, 2002). Of all the fields in a BIND-chemsource object, only `names` is required. This means that for a BIND-object to be declared a small molecule of chemical origin, one must only provide a pointer to a small molecule database and one name of the chemical.

5. *An optional BIND-cellstage list to contain a list of cell cycle stages in which this object is found, or expressed, in the given organism*. This information is only relevant for BIND-objects of organismal origin. A BIND-cellstage object is an enumeration of all of the basic cell stages in the cell cycle. It contains an optional text description field that can describe other cell stages that are not present in the enumeration.

6. *An optional BIND-place-set to describe a cellular localization of this BIND-object.* The BIND-place-set data type describes a start and an end location in the cell and is described further below. Generally, only the start place is relevant for a BIND-object.

7. *An optional NCBI Bioseq object to store a biological sequence if a record for the sequence is not present in any public database*. The Bioseq may also be used to store a local copy of a sequence, as may be needed in a private database that has not yet submitted sequences to GenBank for an accession number. This field is only relevant for biological sequences. Bioseqs can be prepared using Sequin (Kans and Ouellette, 1998) and can be exchanged with NCBI.

8. *An optional NCBI Biostruc object to store a three dimensional atomic structure of any chemical object, from an atom to a complex of molecules, if the data is not present in any public database*. The Biostruc specification allows a chemical graph to be stored with or without coordinates. This is most useful for storing small molecule structures or post-translationally modified forms of a biomolecule. Thus, chemical entities within a BIND object can be described in precise detail.

The presence of these powerful and mature data structures in this part of the specification signifies that BIND is not completely reliant on other databases. Most of

the information present in any public sequence or 3-D molecular structure database can be stored using the BIND specification if necessary.

9. *An optional free flow text description of the BIND-object*. This field could contain, for example, a full name for a molecule such as Adenosine Triphosphate (ATP).

10. *An* `extref` *field for an external reference to another database.* Since the BIND specification may be used in a larger database warehouse type setting potentially where other databases exist that store similar information to a BIND-object, records in those databases may be referenced here. This allows one to more easily integrate BIND with other databases instead of being forced to convert molecules from one database to BIND-objects for use with BIND.

An Interaction - BIND-Interaction

The BIND-Interaction object is the fundamental component for storing data in this specification. It defines and describes the interaction between any two molecules, or even abstract objects. The majority of the information that can be stored is, however, used to describe interactions between proteins, DNA, RNA and genes. Interactions will only be referred to between molecules only rather than between molecules, atoms or other object types from this point on.

An interaction contains an NCBI Date object, a sequence of updates for an audit trail, an Interaction Identifier (IID) accession number, two interacting molecules (BIND-object), a description of the interaction, a series of publications, a list of record authors, a database division description, a private flag and an external reference object. Accession numbers for the three main types of records in BIND, interaction, complex and pathway are all in the same primary key space. That means that no two records in BIND can have the same accession number even if they are different record types. The BIND project plans to control BIND IID number space using a unique key server. Molecule A binds to molecule B and both are stored using BIND-objects (described above).

The BIND-descr object stores most of the information in an interaction object. It contains text description of the interaction, information on cellular place of interaction, experimental conditions used to observe the interaction, conserved sequence comment of

molecules A and/or B if they are biological sequences, location of binding sites on molecule A and B, chemical actions mediated by the interaction, chemical states of the molecules A and B, and an intramolecular interaction Boolean type to flag this type of interaction.

A BIND-pub-set is included to store empirical evidence references, usually publications that `support`, `dispute` or have `no-opinion` regarding the actual interaction. The dispute flag allows the database to track experimental trends and offer a machine-readable way to find discrepancies or differences of opinion in the literature. This will also allow query tools to be built that can generate the current view of a pathway and weight interactions by reliability. Interactions with more dispute flags would be treated as less reliable than interactions with many support flags.

A list of NCBI Author objects record the authors of the record. An author of a database record is the person who contributed to the creation of that record by performing the task of data entry. An author of a BIND record is not automatically the author of the publication that described the data unless that person enters the data into BIND. Authorship is rewarded by recognition, if desired, and by ownership. Ownership means that only the person who enters data can later change that data; it does not entail any copyright to the data as all information in BIND is in the public domain.

BIND may be organized into logical divisions based on the type of data. An optional BIND-Rec-coll-descr object in the interaction record determines if the record is part of a collection. For example, genetic interactions may be considered a division of BIND because they are fundamentally different than physical interactions from a biochemical sense. Divisions allow BIND to be more flexible in the types of queries that are supported as one can quickly select to search only those records of interest.

A private flag that defaults to FALSE is included in an interaction record. The flag indicates whether or not to export this record during a data exchange procedure. In a public database, a private record is not available to the public. This may be because a record has not been completed or information in the record has not yet been verified. In a private database, the private flag means that the record could be viewed internally, but it would never be exported. In this situation, a private record might contain proprietary

information and the database may contain a mix of these and public records imported from a public database.

Finally, an external reference is made available as a BIND-other-db object to allow one to reference other databases of interactions that may be available in the context of a larger information system.

Figure 2: Graphical Representation of the BIND Data Model in UML

This figure expands upon sub-types of BIND-Interaction except for BIND-descr, which is shown in Figure 3. Data fields preceded by an asterisk are optional in the specification. Short ASN.1 "ENUMERATED" lists in are shown in full, while long lists are only described in the specification and referenced using a UML note. ASN.1 "CHOICE" elements are marked in the figure. Referenced NCBI data types are marked "NCBI Type" and are not expanded. See the NCBI data model for further details on those types. Integer types are marked as such.

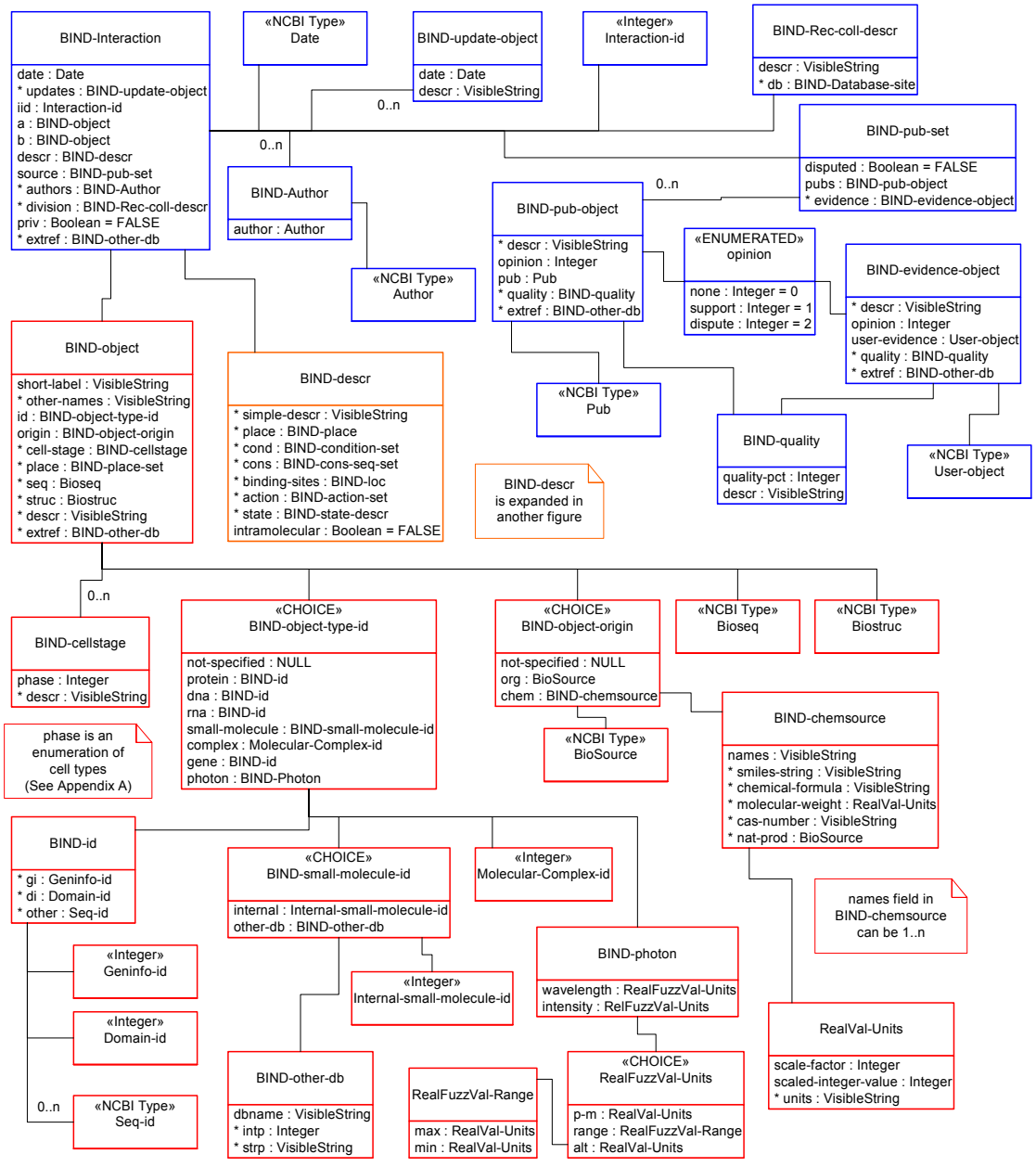

BIND-Interaction
date : Date
* updates : BIND-update-object
iid : Interaction-id
a : BIND-object
b : BIND-object
descr : BIND-descr
source : BIND-pub-set
* authors : BIND-Author
* division : BIND-Rec-coll-descr
priv : Boolean = FALSE
* extref : BIND-other-db
«NCBI Type»
Date
BIND-update-object
date : Date
descr : VisibleString
0..n
«Integer»
Interaction-id
BIND-Rec-coll-descr
descr : VisibleString
* db : BIND-Database-site
BIND-pub-set
disputed : Boolean = FALSE
pubs : BIND-pub-object
* evidence : BIND-evidence-object
0..n
BIND-Author
author : Author
«NCBI Type»
Author
BIND-pub-object
* descr : VisibleString
opinion : Integer
pub : Pub
* quality : BIND-quality
* extref : BIND-other-db
«ENUMERATED»
opinion
none : Integer = 0
support : Integer = 1
dispute : Integer = 2
BIND-evidence-object
* descr : VisibleString
opinion : Integer
user-evidence : User-object
* quality : BIND-quality
* extref : BIND-other-db
BIND-object
short-label : VisibleString
* other-names : VisibleString
id : BIND-object-type-id
origin : BIND-object-origin
* cell-stage : BIND-cellstage
* place : BIND-place-set
* seq : Bioseq
* struc : Biostruc
* descr : VisibleString
* extref : BIND-other-db
BIND-descr
* simple-descr : VisibleString
* place : BIND-place
* cond : BIND-condition-set
* cons : BIND-cons-seq-set
* binding-sites : BIND-loc
* action : BIND-action-set
* state : BIND-state-descr
intramolecular : Boolean = FALSE
«NCBI Type»
Pub
BIND-quality
quality-pct : Integer
descr : VisibleString
«NCBI Type»
User-object
BIND-descr
is expanded in
another figure
0..n
BIND-cellstage
phase : Integer
* descr : VisibleString
phase is an
enumeration of
cell types
(See Appendix A)
«CHOICE»
BIND-object-type-id
not-specified : NULL
protein : BIND-id
dna : BIND-id
rna : BIND-id
small-molecule : BIND-small-molecule-id
complex : Molecular-Complex-id
gene : BIND-id
photon : BIND-Photon
«CHOICE»
BIND-object-origin
not-specified : NULL
org : BioSource
chem : BIND-chemsource
«NCBI Type»
BioSource
«NCBI Type»
Bioseq
«NCBI Type»
Biostruc
BIND-chemsource
names : VisibleString
* smiles-string : VisibleString
* chemical-formula : VisibleString
* molecular-weight : RealVal-Units
* cas-number : VisibleString
* nat-prod : BioSource
names field in
BIND-chemsource
can be 1..n
BIND-id
* gi : Geninfo-id
* di : Domain-id
* other : Seq-id
«Integer»
Geninfo-id
«Integer»
Domain-id
0..n
«NCBI Type»
Seq-id
«CHOICE»
BIND-small-molecule-id
internal : Internal-small-molecule-id
other-db : BIND-other-db
«Integer»
Internal-small-molecule-id
«Integer»
Molecular-Complex-id
BIND-photon
wavelength : RealFuzzVal-Units
intensity : RelFuzzVal-Units
RealVal-Units
scale-factor : Integer
scaled-integer-value : Integer
* units : VisibleString
BIND-other-db
dbname : VisibleString
* intp : Integer
* strp : VisibleString
RealFuzzVal-Range
max : RealVal-Units
min : RealVal-Units
«CHOICE»
RealFuzzVal-Units
p-m : RealVal-Units
range : RealFuzzVal-Range
alt : RealVal-Units

Figure 2

Interaction Description - BIND-descr

All of the objects directly linked in this structure are optional to allow any level of richness of data to be stored. BIND-descr contains:

1. *A simple text description of the interaction*. This free flow text is meant to be a short description of the interaction such as, "Transcription factor X binds to a region of human DNA in section Y of chromosome 11."

2. *A sequence of BIND-place objects in a BIND-place-set*. A BIND-place object stores information about the location of the interaction with respect to the cell. The place of an interaction is meant to be the location where molecule A and B come together in a biologically meaningful way. This object contains a BIND-place-id integer that is unique among this BIND-place-set object to allow the place object to be referenced from anywhere in the database. A BIND-gen-place-set object is available for storing general place data, an optional BIND-spec-place-set object for storing specific place data, an optional BIND-pub-set for storing publications referring to the localization of an interaction, and an optional text description field. A BIND-gen-place-set contains a start and an optional end place for the interactions, specified by a somewhat hierarchical enumerated list of general places in the cell. An optional text field is also present for free text such as the description of the mechanism of translocation. Storing a start and an end place for an interaction takes into account the possibility of an interaction translocating across membranes and ending up in different sub-cellular compartments. The relatively simple enumeration of 33 cell places allows a computer to understand the location of the interaction. Some cell places contain other data objects to further specify the location. One example is the `golgi` choice, which contains a BIND-membrane object that specifies if the interaction is at a surface or integral to the membrane or in the Golgi lumen. Thus, the location description is somewhat hierarchical. If the hierarchy were to be flattened, over 150 distinct cellular places would be present. Having a general list is important for data visualization programs that need to be able to draw molecules in the correct places on a diagram of a cell. A human readable description of cellular place can be stored in the BIND-spec-place-set. This object contains a text description of a start and an optional end place for an interaction. More specific data regarding the location of

interaction, such as in what part of a membrane, apical or basal, an interaction occurs can be stored in the BIND-spec-place-set object. An optional pointer to a database of cellular locations, such as the Gene Ontology (The Gene Ontology Consortium, 2000), is present for reference purposes.

Multiple BIND-place objects are present to allow storage of an interaction that may be present only at certain separate places within and around the cell. More than one place object can also be used to describe an interaction occurring between two molecules over multiple sub-cellular compartments, as might be the case for transmembrane receptor proteins with large extra and intra-cellular domains. These two cases might need to be differentiated by cell place information in the BIND-object or by external information, such as if the protein has a transmembrane region.

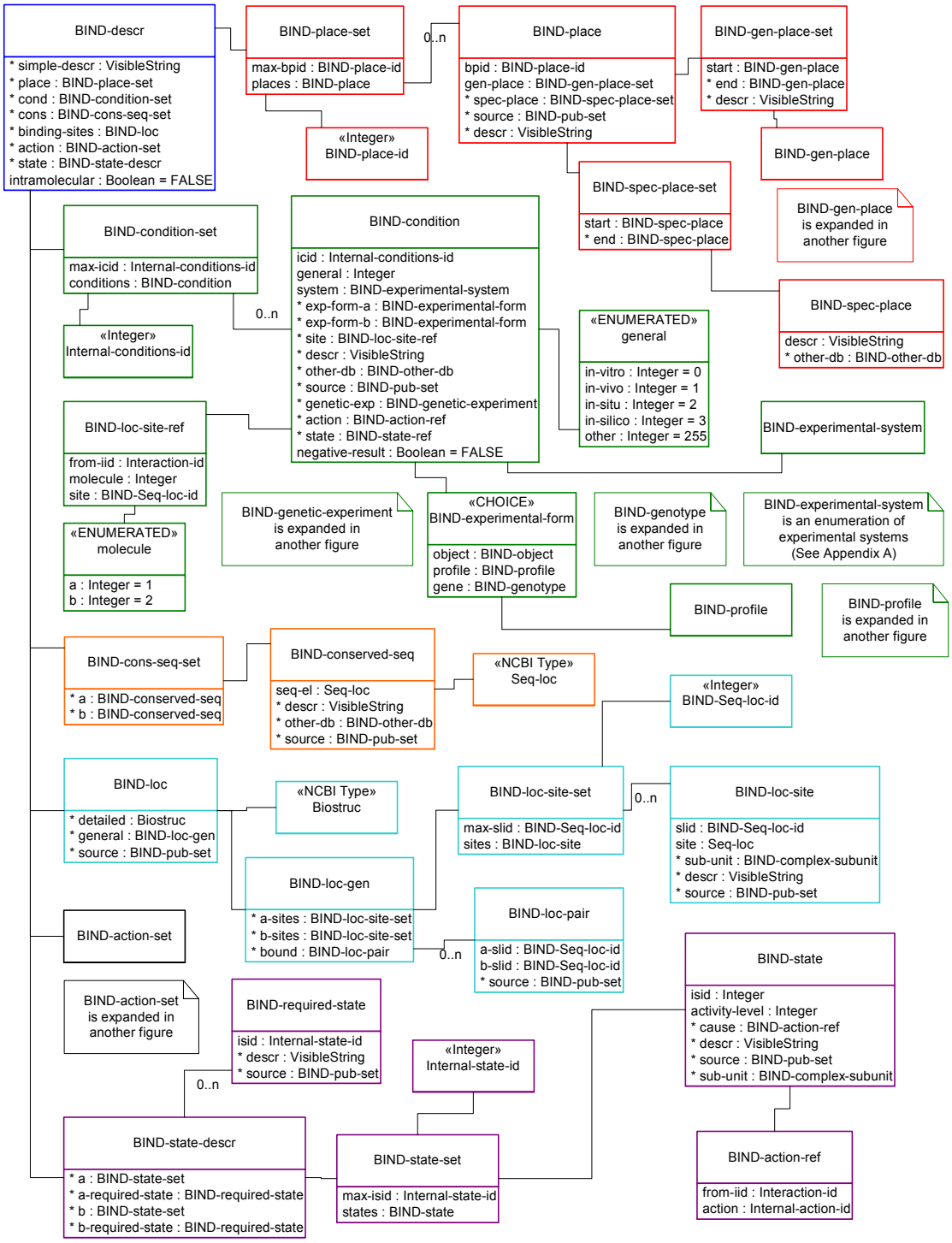


Figure 3: Continued UML Representation of the BIND Data Model Showing BIND-descr

This figure shows the BIND-descr type and some of its sub-types. Sub-types not represented here are shown in subsequent figures. See Figure 2 caption for notation.

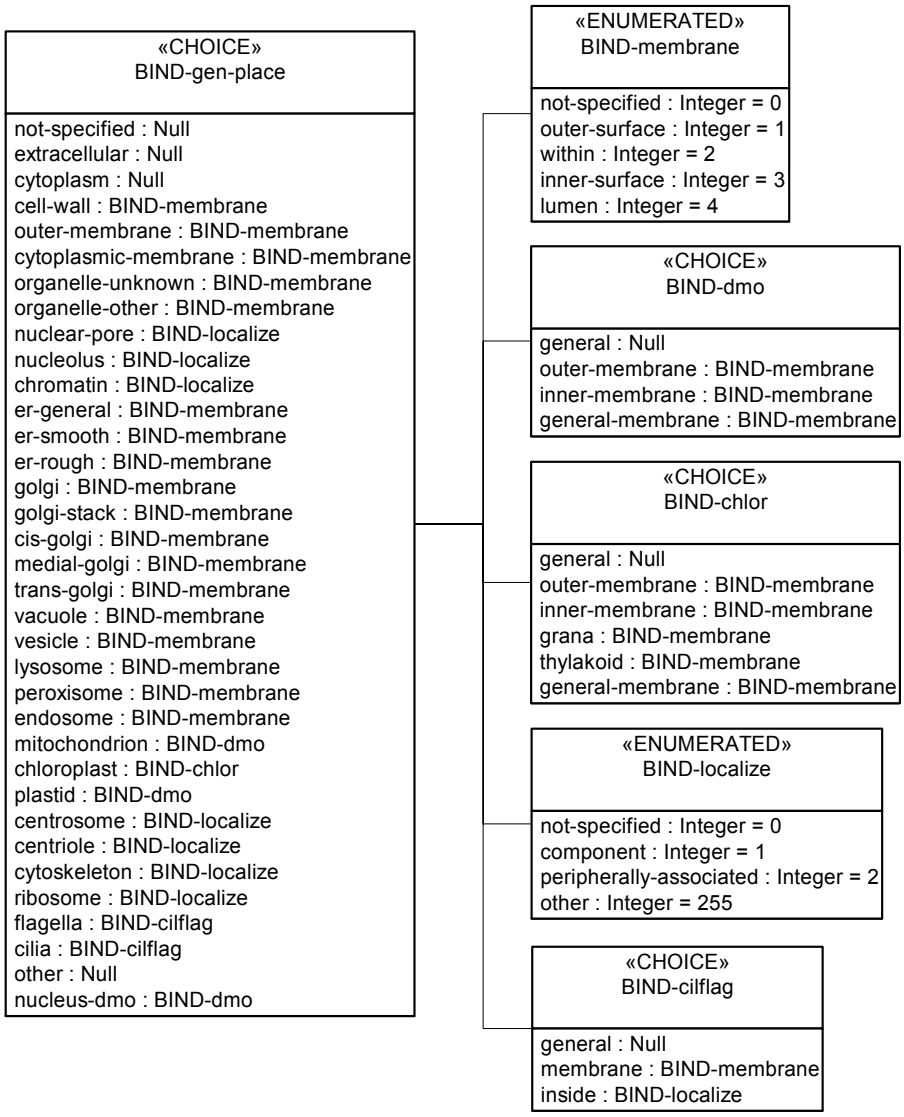


Figure 4: Continued UML Representation of the BIND Data Model Showing BIND-gen-place

This figure shows the BIND-gen-place type. See Figure 2 caption for notation.

3. *A BIND-condition-set to store a list of experimental conditions used to observe the interaction*. While actual data from the experiment is not stored here, experimental condition information stored should be sufficient to allow recreation of the original experiment. An experimental condition is described using a BIND-condition object. This object contains an Internal-conditions-id (ICID) number that can be used to reference a particular experimental condition in the BIND-condition-set from another part of BIND. A general experimental condition is an enumeration of five general conditions, *in-vitro*, *in-vivo*, *in-situ*, *in-silico* and other. A BIND-experimental-system object is present and is an enumeration of most popular experimental techniques, with 37 techniques listed in the specification. This field has been simply declared as an INTEGER enumeration type so that it can be easily extended with new experimental systems as they become available. Declaring a type as INTEGER in ASN.1 instead of enumeration prevents generated code from checking the name of the enumerated value against the specification. This means that items may be added to the list at a later date without disrupting tools that are based on previous specifications. An experimental form of one of the interacting objects can optionally be described here in the `exp-form-a` and `exp-form-b` fields, which are BIND-experimental-form objects. This data type is a choice of either a BIND-object, which could represent e.g. an epitope tagged form or truncated form of a protein, a BIND-profile, which is meant to represent a position specific score matrix (PSSM) for describing protein and nucleic acid sequence motifs as defined by the PROSITE database at http://www.expasy.ch/txt/profile.txt, or a BIND-genotype, which is used to store the experimental form of a gene in a genetic interaction experiment. The BIND-profile choice can describe the sequence pattern that a molecule binds to. For instance, a transcription factor or a restriction enzyme can bind to a pattern of DNA and many SH3 domains prefer binding to nonapeptide proline-rich motifs. Once a preferred binding motif is experimentally determined, it is common to use this to predict binding sites for these molecules. Either an experimentally determined or *in-silico* predicted interaction with a motif can be stored. A BIND-condition also contains an optional list of BIND-loc-site-ref, BIND-action-ref and BIND-state-ref objects to respectively reference binding sites, chemical actions or chemical states of molecule A or B that are involved in or were determined by this experiment. A BIND-other-db data

type is present to reference an experimental method database that may exist in the future. A `negative-result` Boolean flag is present to signify if the negative result of the experiment helps prove that two molecules interaction. For example, if a mutation of a specific residue of a binding-site ablates an interaction, then that residue may be important for binding. The experimental form of the molecule should also be described as having the mutation. Another enumerated type, `bait-condition`, records if the molecule is 'bait' in the experiment. This is only relevant for certain experimental conditions, such as co-immunoprecipitation and two-hybrid screens. Fundamentally, there are generally two types of molecular interaction experiments. In one type of method only two molecules are in an experimental system and their interaction is assayed. This is a binary experimental system, as the molecules either interact or don't interact. In the other type of method, one molecule, the 'bait', is screened against a collection of more than one other molecule and the result is a set of molecules from the collection that bind to the bait. One posits that the bait binds to all other molecules, but the interaction may be indirect if other molecules from the collection mediate it. The result should be recorded as a series of interactions of the bait to the associated molecules, but knowledge of what molecule was the bait helps one to determine if an indirect interaction is possible. Another type of interaction experiment could be considered where many baits are screened against a collection at the same time, such as the matrix yeast two-hybrid approach (Uetz et al., 2000), but this case deconvolves into a multiplexed version of the single bait screen case. A BIND-condition object also contains a free human readable text description. This field could be used to describe a system further or could be used to name a system if `other` has been specified as the BIND-experimental-system object. A BIND-pub-set is also provided in order to store publications related to the experimental systems described in the BIND-condition object.

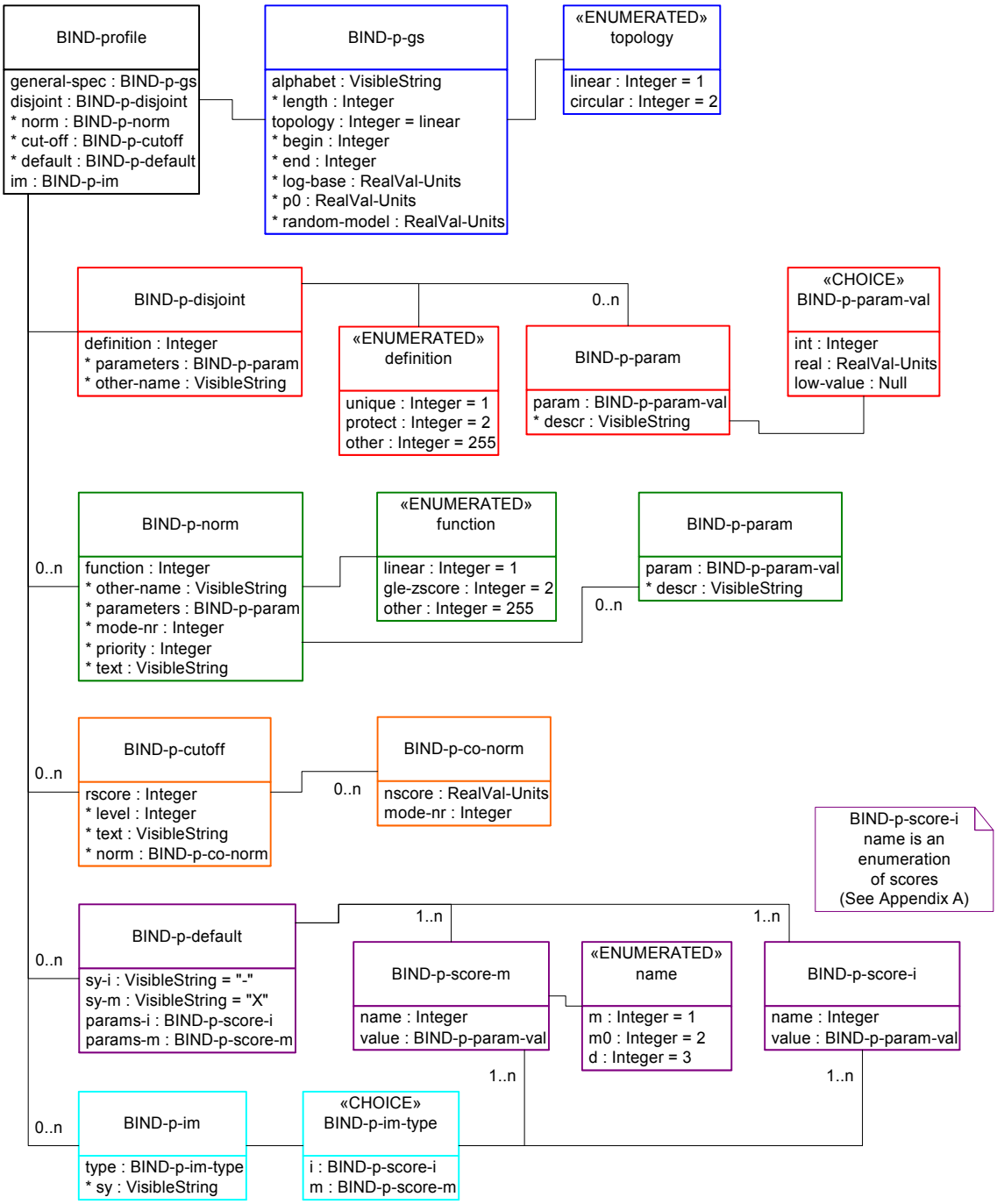


Figure 5: Continued UML Representation of the BIND Data Model Showing BIND-profile

This figure shows the BIND-profile type from PROSITE. See Figure 2 caption for notation.

A large part of the BIND-condition object is devoted to describing genetic interaction experiments, which consists of a combination of the `gene` choice in the experimental form and the BIND-genetic-experiment object in the `genetic-exp` field. These data types are only relevant if object A and B in the interaction are genes. As mentioned above, a BIND-genotype object describes the experimental form of a gene, which is a collection of all of the alleles of a gene present in the biological system, the allelic composition. This object consists of an optional BIND-allele-copy-num object to describe the total copy number of all alleles of the gene on chromosomal and extra chromosomal genetic elements. This can be general or specific with a choice of `high`, `single`, `wild-type`, `reduced` when the exact number of copies is not known and a possibly fuzzy number when a more specific or exact copy number is known. The actual sequence of alleles for the genotype of the gene is stored as a list of BIND-allele objects and this is the only non-optional element in the BIND-genotype object. The phenotype expressed with this collection of alleles is described with an optional BIND-phenotype object and the genetic background of the genotype is defined using a BIND-genetic-background object. All genetic objects described in the BIND specification are in relation to the wild-type form of the genome, which is operationally defined as the sequenced strain present in the database attached to the NCBI taxonomy ID listed in the interaction BIND-object for the gene.

BIND-allele describes the form of a gene and is comprised of:

i. A BIND-id to reference the 'archetypal' gene on the genome. The actual allele is described in relation to this wild-type form.

ii. An optional sequence of names for the allele that should correspond to the accepted genetic nomenclature for the organism in question. For example, in yeast, the first discovered allele of ARP2 would be represented as ARP2-1.

iii. The experimental form of the allele, which can be a choice of `not-specified`, `genomic` – the wild-type allele, `knock-out` – the gene has been completely deleted and `mutation`, which is a BIND-object that can hold a new DNA sequence for the allele if any base has changed or any number of bases has been added or deleted.

iv. A `copy-num` field to record how many copies of this experimental allele are present.

v. A BioSource object that can describe where the allele resides in the biological system, whether on a chromosome or on a plasmid from a specific strain.

vi. Optional free text description and a BIND-pub-set for evidence of this allele are also present.

The BIND-phenotype object stores a general description of a phenotype and consists of, an optional text field for the trait that is described, for example, "colour", a name for the phenotype, for example, "red", whether the phenotype is wild-type or not, or not specified, an optional text description and a list of links to a phenotype database. The DGAP project lists certain phenotypes (http://dgap.harvard.edu). BIND-genetic-background describes the genetic background of the system as a series of changes from the 'wild-type' genome. The organism and strain of the background is held in a BioSource object, the ploidy number of the organism is stored as well as if any chromosomes are present in different copy numbers than the organism ploidy, as is trisomy 21 (Down Syndrome), and the allelic changes in the background from the standard genome. Changes are represented as a list of BIND-allele-change objects that contain one BIND-allele object for the `old-form` and one for the `new-form` along with optional free text description.

Finally, the actual genetic experiment and its result are described in detail using the BIND-genetic-experiment object of the BIND-condition type. A genetic experiment is performed by crossing two parents and observing the resulting phenotype in the progeny to try to determine the genotype of the parents or the genes that are involved in specific phenotypes. The resulting phenotype of the experiment is stored as a list of BIND-genetic-exp-result objects. These, in turn, contain a BIND-phenotype object, an optional BIND-genetic-relation, which stores the relationship of the progeny's phenotype to the parents', and a BIND-genetic-background object to describe the organism and strain background of the progeny. A BIND-genetic-relation object describes if the phenotype of parent A (from gene BIND-object A in the interaction and experimental form of gene A in the BIND-condition container), if parent B is wild-type, if A and B

parents have the same phenotype, and the phenotype of the progeny as a BIND-genetic-ab-phenotype, which can be `wild-type`, `a-type` if it is the same as the A parent, `b-type` or novel in which case a BIND-synthetic-phenotype is present. The latter object is composed of a choice of being a modulation percentage of parent A's phenotype, a modulation of parent B's phenotype, a mix of the parents' phenotypes or a completely novel phenotype. For example if the phenotype of parent A is red and that of parent B is white, then if the progeny is pink, it is a mix of A and B's phenotype. A modulation of 0% means no trace of the parent phenotype is present and above 100% means that there is an enhanced phenotype or a stronger phenotype than the parent in this particular trait. Thus, the BIND-genetic-exp-result data type can describe the full range of possible results of a genetic experiment. The rest of the BIND-genetic-experiment object contains the type of the experiment as an extensible enumerated list of possible experiments, such as synthetic lethal, an optional series of BIND-allele-change objects to describe changes to the genetic background, other and those to genes A and B, that are required to see this experimental result. For example, gene disruptions of genes A and B only show synthetic lethality when gene C is mutated at a specific residue. Also present is an optional text description of the experiment, an optional list of BIND-objects to describe if molecules, such as DNA damage chemicals, are present during the experiment, and a general, but structured description of the environmental conditions used for the experiment, such as temperature.

4. *A BIND-cons-seq-set to store information about evolutionarily conserved sequence if either molecule A or B is a biological sequence*. This information is simply meant to be a comment on the possible importance of certain sequence elements that have been noticed to be conserved via phylogenetic or other evolutionary analysis. It is possible that information about conserved sequence is known for molecules in an interaction that is not very well characterized. This data might be useful to investigators interested in further studying the interaction, for example when deciding to make mutant gene constructs to find amino acids involved in the interaction. A BIND-cons-seq-set contains conserved sequence information about molecule A and B in a BIND-conserved-seq object. Semantically, a BIND-conserved-seq object may only be instantiated with data if the molecule that it refers to is a biological sequence. A BIND-conserved-seq

object contains an NCBI Seq-loc object. A Seq-loc can contain a location or a set of locations for any linearly numbered biological sequence. A free text description is also included in a BIND-conserved-seq as well as a BIND-other-db object to reference a conserved sequence database, such as BLOCKS (Henikoff et al., 2000). It is suggested that the method of determining the conserved sequence, for example a phylogenetic tree program such as PHYLIP (http://evolution.genetics.washington.edu/phylip.html) or an alignment program such as PSI-BLAST (Altschul et al., 1997) or CLUSTAL (Higgins et al., 1996) be stored in the `descr` field. A BIND-pub-set object is provided to store publications pertaining to a conserved sequence comment.

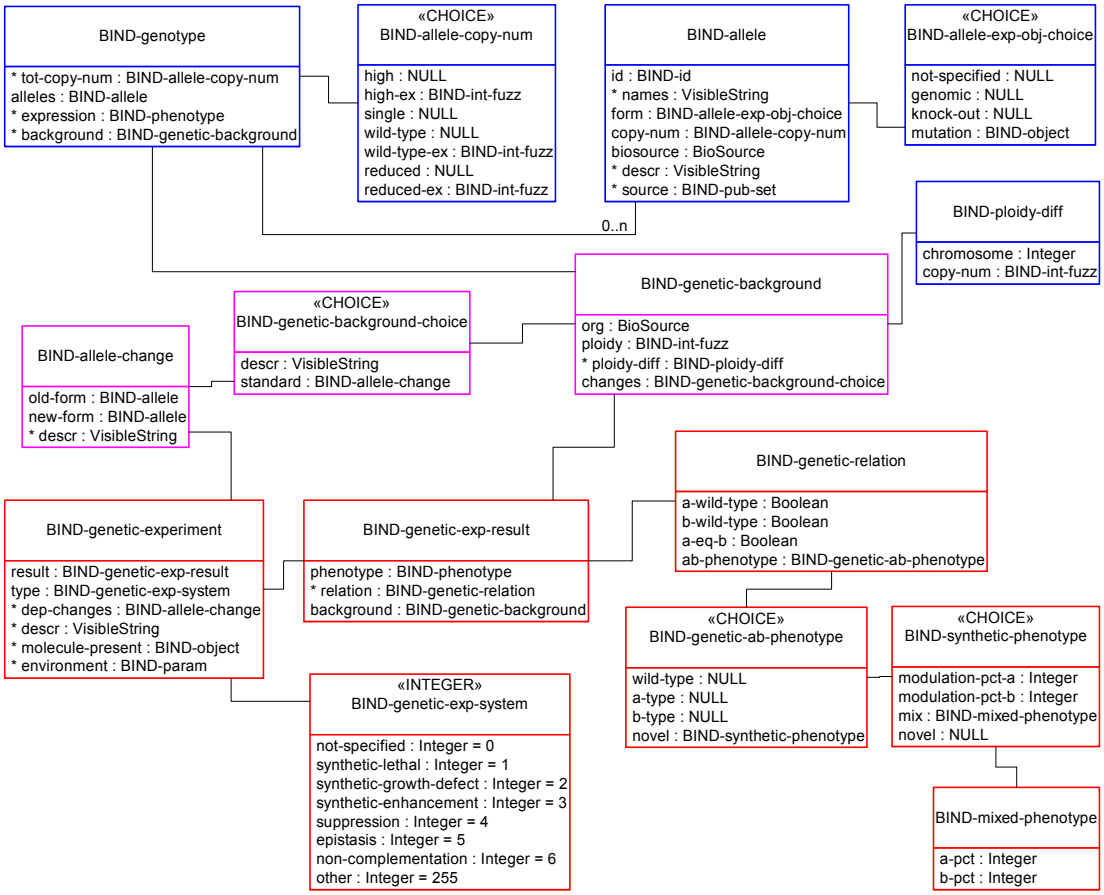


Figure 6: Continued UML Representation of the BIND Data Model Showing BIND-genotype and BIND-genetic-experiment

This figure shows the BIND-genotype and BIND-genetic experiment types found in a BIND-condition object. See Figure 2 caption for notation.

5. *A BIND-loc to store binding site information from very detailed to general*. A BIND-loc can store 3-D atomic level detail of an interaction site using an NCBI Biostruc. A BIND-loc-gen object is also present to store binding sites in an interaction at the sequence element level of detail. Therefore, only interactions involving biological sequences can hold general binding site information. The BIND-loc object also includes a BIND-pub-set for storing publications related to binding site. All top-level fields are optional allowing detailed, general and/or source information to be represented. Expanding further, the BIND-loc-gen object contains a list of binding sites on molecule A and a list of binding sites on molecule B. This information is contained in a BIND-loc-site-set object that contains a sequence of binding sites defined in BIND-loc-site objects. Each BIND-loc-site element contains an NCBI Seq-loc element and an internal reference integer ID called a BIND-Seq-loc-id. Since each binding site is numbered in a BIND-loc-site-set, other objects in the database can reference it. A BIND-loc-site also contains an optional reference to a subunit of a molecular complex as a BIND-complex-subunit object if object A or B is a complex and the binding site on one of its subunits is known. An optional text description for the site and a BIND-pub-set for publication information is also available.

A BIND-loc-gen object also contains an optional BIND-loc-pair object that specifies which binding sites on A bind to which binding sites on B. The binding sites are referenced from the BIND-loc-site-set objects so in order to use a BIND-loc-pair object, binding sites on molecule A and B must already be defined. This simple binary mapping allows most experimental binding information, such as that generated from foot-printing analysis, to be stored. An optional BIND-pub-set is present here as well to store evidence for the binding site pair.

6. *A BIND-action-set to describe the chemical action(s) mediated by this interaction*. A set of actions is required because there are many examples of interactions having multiple chemical actions. For instance, a kinase may phosphorylate a protein more than once in separate chemical actions or a restriction enzyme may cleave a molecule of DNA in more than one place. A BIND-action-set contains a set of elaborate BIND-action objects. Each BIND-action object in a set is numbered with an Internal-action-id (IAID) integer so that other data types can reference it.

A BIND-action object contains the IAID number, an optional text description field for free flow text description of the chemical action and an optional BIND-pub-set for storing publications pertaining to this chemical action. A BIND-direction type is included to specify the direction of the chemical action, which is an enumerate type that can be `none`, `a-to-a`, `a-to-b`, `b-to-b`, `b-to-a` or `other`. This represents all possible directions between two objects. The type of action is defined in the BIND-action-type object. The BIND-action-type object is a choice element that stores the type of chemical action and an associated data object. The possible choices of actions are `not-specified` for an unknown chemical action type, `none` for no action, `add` for adding a chemical object, `remove` for removing a chemical object, `bond-break` for a non-sequence cut action, `cut-seq` for a cut in a biological sequence, `change-conformation` for a change in conformation, `change-configuration` for a change in configuration, e.g. by an epimerase or isomerase, `change-other` for another type of change, such as a metal ion exchange, and `other` for any other chemical action. Since the type of an action is required, the type `none` can be used to store information in the BIND-action object, such as its result, even if there is no chemical action. Types `add`, `remove` and `cut-seq` are associated with a BIND-action-object to store related data. A BIND-action-object is a choice element that can store nothing, with a choice of NULL, a BIND-object, or a site on a sequence using a Seq-loc. The `object` choice of the BIND-action-object is only relevant for the `add` and `remove` choices of the BIND-action-type. The BIND-object is meant to store a description of the chemical compound that is added or removed. An example would be a phosphate group added by a kinase enzyme or removed by a phosphorylase enzyme. The `location` choice of the BIND-action-object is only relevant for the `cut-seq` choice of the BIND-action-type. The Seq-loc is meant to store the position(s) after which a biological sequence is cut. An example would be the locations after which a restriction enzyme cuts DNA or the sites after which a protease cleaves in a protein. The choice of `none` can be used for `add`, `remove` or `cut-seq` if information that would otherwise be stored is not known.

Continuing with the description of the BIND-action object, an optional result field is present as a sequence of BIND-result-object types to store the resulting molecule(s) from a chemical action. The BIND-result-object contains a BIND-object and an Internal-

result-id integer that allows the result to be referenced from other parts of the database. For instance, if a molecule of DNA was methylated, the description of the methylated DNA could be stored in a BIND-result-object. If a protein molecule was cut at various locations by a protease, all resulting protein molecule fragments could be described with the BIND-result-object sequence. With a sequence of interacting proteins where A binds to B, B binds to C, etc., the result field storing the full chemical form of B in the A-B interaction, for example, could be used directly in the B-C interaction record. This allows the exact description of sequential chemical modifications on a biological sequence that would otherwise not be possible given the standard sequence representation alone.

A Biostruc-feature-set that can contain residue or atomic level of detail differences in a molecule created by this chemical action is also present. The molecule that is different in this case is based on the direction of the chemical action. If the direction is molecule A to B, any information stored in the `diff` field would pertain to molecule B, not A. This field allows even small changes in molecules to be represented, as in the example of a chemical action reducing a double bond by adding two hydrogen atoms across it. The addition of the two hydrogens could be recorded as differences on an atomic structure. This information requires the presence of atomic level detail data for the molecule being changed. The diff field can also represent changes made to the substrate of the chemical action. In an example of a phosphate added to a protein on a specific tyrosine residue by a tyrosine kinase enzyme, the diff field would simply be the position in the protein sequence of the tyrosine that was being changed.

An optional BIND-signal object is included in the BIND-action object to store directional information related to chemical signal as it is found in cell signaling pathways. This data is really a more general notion of kinetics describing signal transduction. The signal could, for example, be the activation of proteins in a signaling cascade via phosphorylation such as in a MAP kinase pathway. BIND-signal contains an enumerated type describing the signal modification from a top-level viewpoint. Possible values are `none`, `amplify`, `repress` and `other`. The direction of the signal is stored in a BIND-direction object. An optional RealVal-Units object in the `factor` field can store the factor of signal amplification or repression if they occur. Signal amplification in

the cell is really just the recruitment of molecules one step further down in the pathway by the molecule at the current step. So, if molecule A activates molecule B by removing a phosphate in a signaling pathway and there is amplification at this step, in the cell, molecule A activates many molecules of B causing a strengthening of the chemical signal by a measurable factor that may be stored. An optional free text description that should contain some description of the signal action if `other` is specified in the `action` field and a BIND-pub-set are available in the BIND-signal object as well.

Kinetic and thermodynamic data may also be optionally stored in the BIND-action object using the BIND-kinetics object. The BIND-kinetics object offers specified real value and text description fields for common kinetics (e.g. Michaelis-Menten) and thermodynamic values as well as providing a sequence of BIND-kinetics-other objects to store any other text or real number values that may be pertinent. A BIND-pub-set object is also present to store publications that relate to any of the information stored. All objects in the BIND-kinetics object are optional to allow any combination of values to be stored.

Also in the BIND-action object, a link to a sequence of experimental conditions used to observe this chemical action is optionally provided using a sequence of BIND-condition-ref objects. The BIND-condition-ref object references a previously defined experimental condition by Interaction-id and Internal-conditions-id number. In this way, any experimental condition in a database using this specification may be uniquely referenced.

If molecule A or B in the interaction is a molecular complex, the subunit to which the chemical action applies can be optionally specified. If specific sites on molecules A or B are involved in the action, they can be specified by a list of BIND-action-site objects. For instance, the action could be 'performed' by a site such as an active site of an enzyme or the site could be affected by the action, as in a protein binding domain that gets phosphorylated so that if can no longer bind its substrate. The site is either a reference to a predefined site in the interaction record or, if a site cannot be referenced, a BIND-loc-site-set to represent a newly defined site. A BIND-other-db object is also present to allow referencing to possible future databases of chemical actions.

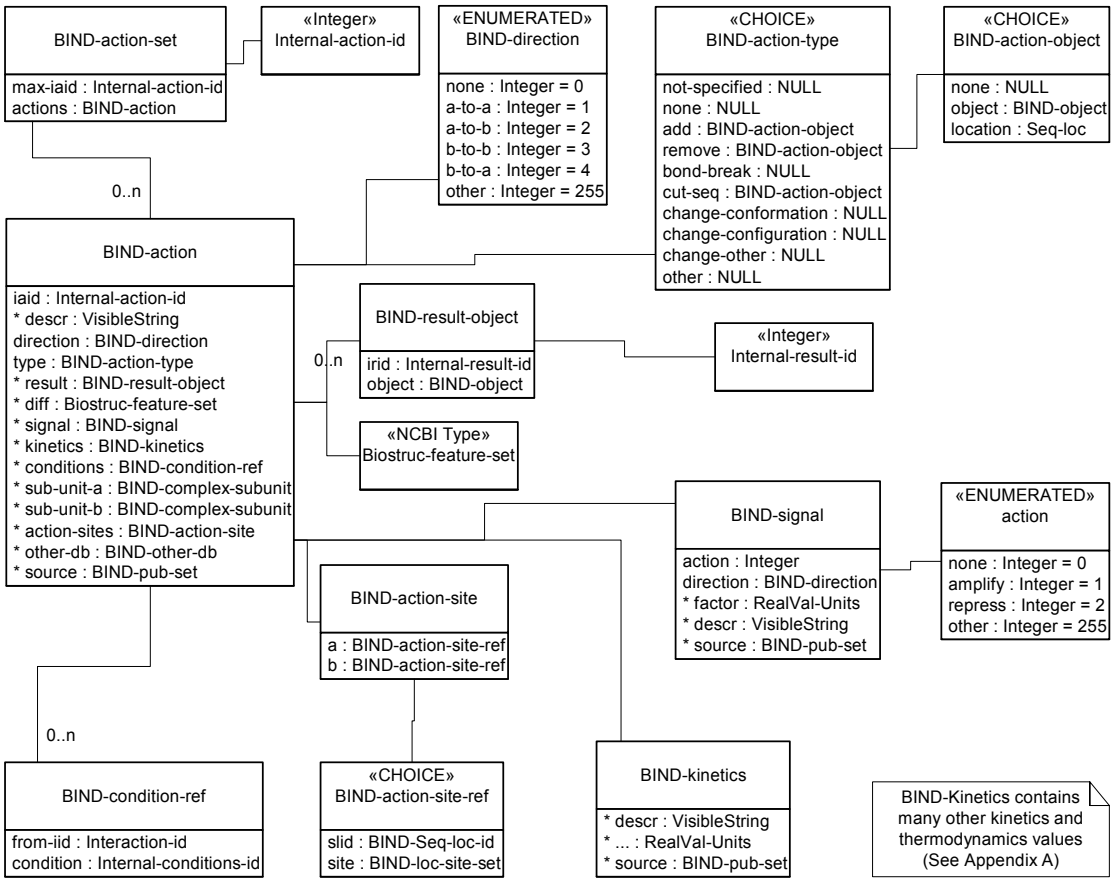


Figure 7: Continued UML Representation of the BIND Data Model Showing BIND-action-set

This figure shows the BIND-action-set type found in the BIND-descr object. See Figure 2 caption for notation.

7. *A BIND-state-descr object for storing information on chemical state of molecule A or B*. The BIND-state-descr object stores a list of possible chemical states for molecules A and B in BIND-state-set objects as well as references to defined chemical states of A and B that are required for the interaction to take place, in BIND-required-state objects. More than one possible state can be saved because certain molecules can assume multiple states. One example is a protein enzyme that may be multiply phosphorylated to bring about different enzymatic activity levels, depending on the phosphorylation level. All fields in the BIND-state-descr object are optional allowing any combination of data objects to be stored. A BIND-state-set contains a sequence of BIND-state objects each numbered by an Internal-state-id (ISID) integer so that other data types can uniquely reference them. Apart from the ISID, a BIND-state object contains an optional enumerated list describing the general activity of the molecule, an optional sequence of BIND-action-ref objects in the `cause` field, optional free text description, an optional BIND-pub-set for storing publications related to this chemical state and a reference to a molecular complex subunit if A or B is a complex and if the chemical state refers to only one subunit. The `activity-level` list is a simple description and is purely subjective, but is still useful for discriminating various states of different activity, especially by a data visualization program that could colour molecules based on this information. The BIND-action-ref object can be used to uniquely reference previously defined chemical actions from this or other interactions that bring about this state. It contains an IID and an IAID. This functionality is very important in the specification because it allows full chemistry to be described when chemical actions and chemical states are taken together. Full chemistry means that all substrates, enzymes, products, bioprocessed compounds etc. may be represented in full atomic level detail for all steps in a pathway. A certain chemical action can have a result (in the `result` field of a BIND-action object) and a certain chemical state can reference the action that occurred to create it. In this way bi-directional linked lists can form networks that represent true chemical networks in a cell. This is in effect a second level of graph abstraction that can describe the chemical events and their order in a biochemical pathway. The idea of storing a chemical state was recently borrowed in the LiveDIP

section (http://dip.doe-mbi.ucla.edu/ldipc/tmpl/browse-main.cgi) of the DIP project (Xenarios et al., 2002).

The BIND-required-state object contains a reference to a state within this BIND-state-set that is required for the interaction as well as a free text description and a BIND-pub-set to store evidence that the state is required.

8. *An intramolecular interaction Boolean flag*. This flag is set to true if the interaction is intramolecular. This is only meaningful if both molecule A and B are the same molecule and serves to differentiate an intramolecular interaction from a homodimer, where molecule A and B are also the same molecule.

A Molecular Complex - BIND-Molecular-Complex

The BIND-Molecular-Complex object is the second of three top-level biological objects in the BIND specification. It is meant to store a collection of at least one interaction that forms a complex, i.e. two or more BIND-objects that interact to form a stable complex and function as a unit. One example is the ribosome. In this way, it is useful to store knowledge of molecular complexes and as shorthand for use when defining interactions and pathways (see BIND-pathway).

A BIND-Molecular-Complex object contains similar administrative information fields as a BIND-Interaction. A Molecular-Complex-id (MCID) integer accession number is stored to uniquely identify molecular complexes. A BIND-pub-set is present to store publications that concern this molecular complex and a private flag is provided to mark this record as private using the same rules as the private flag of the interaction record. A list of record authors is present, a database division field as well as an optional external reference to other molecular complex databases.

Seven other fields in the molecular complex store data directly relating to the complex. The `descr` field optionally provides space for a human readable free text description of the molecular complex. The `sub-num` field contains a BIND-mol-sub-num object that stores the number of subunits (BIND-objects) in the molecular complex. The subunit number is a choice of an exact integer using the `num` field or a fuzzy integer in the `num-fuzz` field. The fuzzy number is stored using an NCBI Int-fuzz object that

can store a number in a range, plus or minus a fixed or percentage amount, or store a set of alternatives for the number. Using a fuzzy number, complexes can be stored even when the exact number of subunits is not known. Examples of such complexes are actin filaments or other parts of the cytoskeleton and virus coat proteins, both of which typically form using repeated units of certain proteins. Continuing with the BIND-Molecular-Complex, the `sub-units` field can store the actual subunits of the complex as a sequence of BIND-mol-object data types. The BIND-mol-object is mainly a wrapper for a BIND-object that allows the BIND-object to be numbered using a BIND-mol-object-id integer (BMOID). Numbering the subunit BIND-objects allows the BIND-mol-object-pair to reference them for e.g. topology information, as discussed below. The BIND-mol-object also contains an optional state of the subunit as a reference to a result of a chemical action elsewhere in the database. This allows the chemical action and state graph to extend into the complex subunits. The core component of the BIND-Molecular-Complex is the list of Interaction-ids which references previously defined interactions in a database. This means that most of the data for function, state, location, etc. for a molecular complex is actually stored in BIND-Interaction objects. This avoids some duplication of information. A Boolean flag marks the interaction list as being ordered or not. This should be true if the temporal order of interactions that form the complex is known and the IID list is ordered in that way. Ordering of subunit binding for some well-studied biological complexes, such as the ribosome, is known.

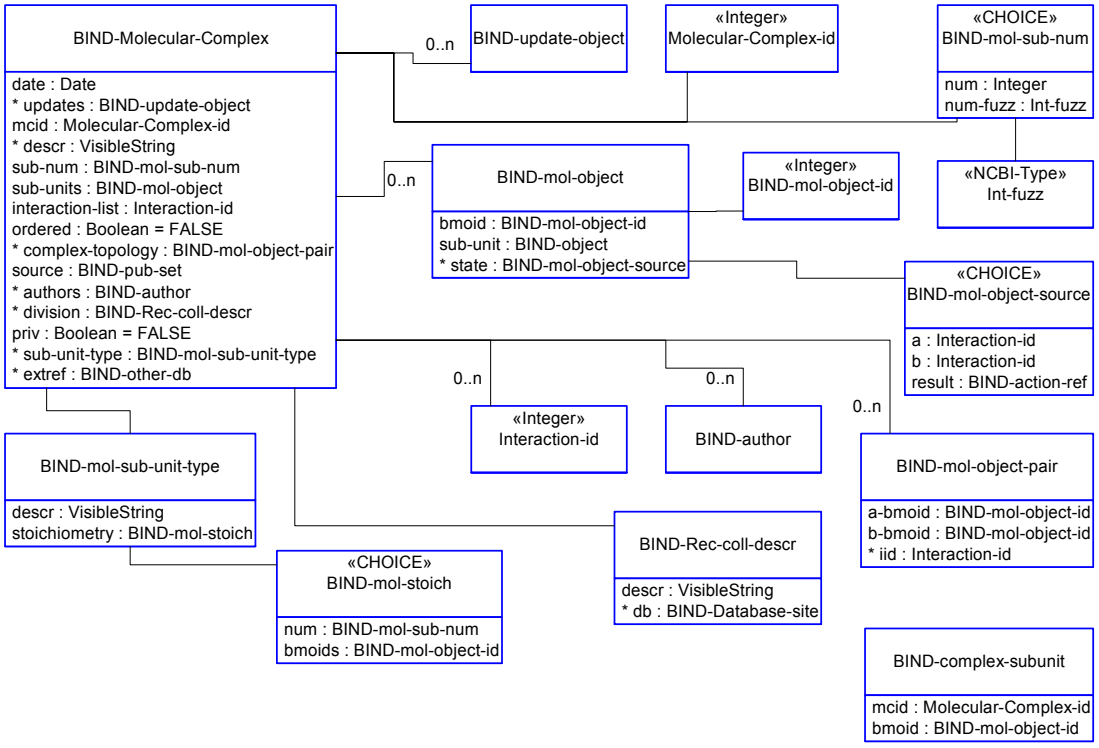
An optional sequence of BIND-mol-object-pair objects is present in the BIND-Molecular-Complex and is meant to store a simple graph-based topology of the molecular complex. A BIND-mol-object-pair simply records a connected pair of BIND-mol-objects in the molecular complex by making a reference to two BMOID numbers of the subunits that are connected and optionally references the Interaction-id that this link refers to. Together the BIND-mol-objects, as nodes, and the BIND-mol-object-pairs, as edges can describe the computer science concept of a graph. The topology information can allow a data visualization program to draw a representation of the actual shape of the complex. The topology can be used, for example, to describe that the subunits of the complex form a ring versus a straight line. Often, complex topology information is disputed in the literature and the topology field in conjunction with the publication opinion can

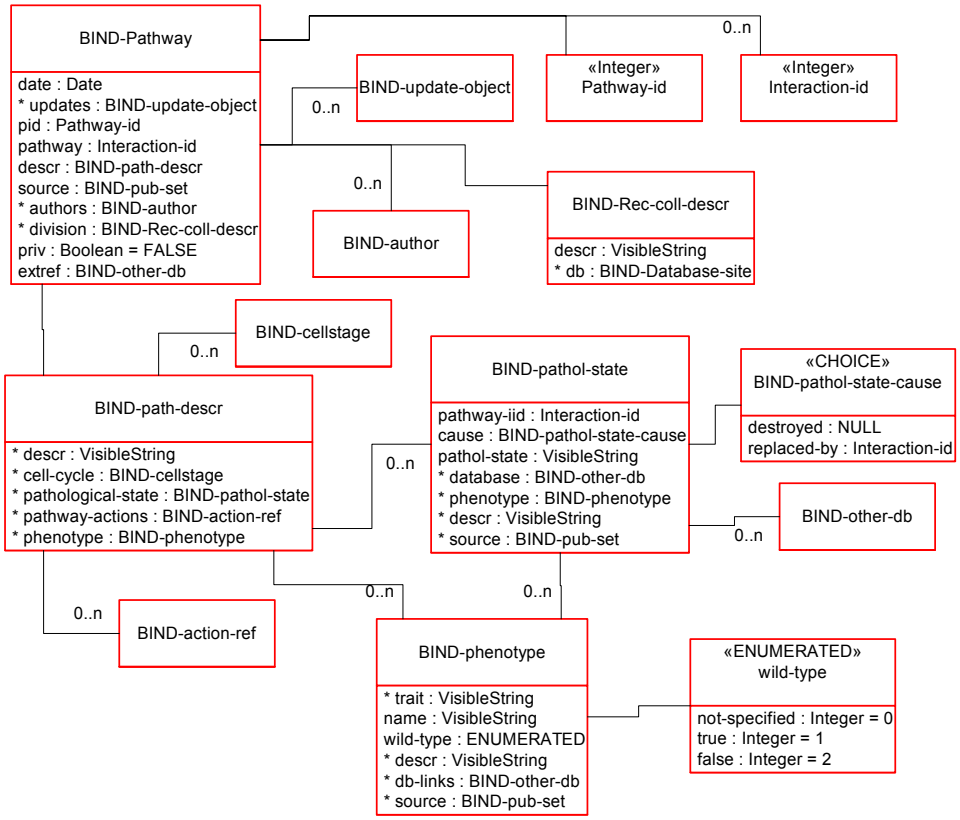
accommodate this discussion. Because most of the data for complexes is referenced from interaction records, a certain amount of automatic data entry can be used. Fetching the data from the given list of interaction records can automatically enter a list of subunits and the number of subunits. Such automatic data entry might not properly represent the stoichiometry of the complex, so the `sub-unit-type` field is present to describe this with BIND-mol-sub-unit-type objects. The BIND-mol-sub-unit-type data type describes the number and type of objects in the complex and can be used to represent e.g. a complex of eight subunits of three proteins $A_4B_3C_1$. A description of the type of subunit is present as well as its stoichiometry as defined by the BIND-mol-stoich object composed of the possibly fuzzy number of subunits and the BMOID numbers that are of this type. If the exact stoichiometry is known, then all subunits must be represented under BIND-Molecular-Complex→sub-units. If only a fuzzy stoichiometry is known, then only the ones that are referenced in the BMOID field must be present under BIND-Molecular-Complex→sub-units.

It can also be noted that a molecular complex can be defined if the pairwise interactions of which it is composed are not completely known. This can be done by creating a set of interaction objects with molecule A as a subunit of the complex and molecule B as `not-specified`. This is useful since many preliminary studies of a molecular complex observe only that certain molecules interact, e.g. from gel data, but not how they interact.

Figure 8: Continued UML Representation of the BIND Data Model Showing BIND-Molecular-Complex and BIND-Pathway

This figure shows the BIND-Molecular-Complex and BIND-Pathway top-level data types. See Figure 2 caption for notation.


BIND-Molecular-Complex
date : Date
* updates : BIND-update-object
mcid : Molecular-Complex-id
* descr : VisibleString
sub-num : BIND-mol-sub-num
sub-units : BIND-mol-object
interaction-list : Interaction-id
ordered : Boolean = FALSE
* complex-topology : BIND-mol-object-pair
source : BIND-pub-set
* authors : BIND-author
* division : BIND-Rec-coll-descr
priv : Boolean = FALSE
* sub-unit-type : BIND-mol-sub-unit-type
* extref : BIND-other-db
0..n
BIND-update-object
«Integer»
Molecular-Complex-id
«CHOICE»
BIND-mol-sub-num
num : Integer
num-fuzz : Int-fuzz
«NCBI-Type»
Int-fuzz
BIND-mol-object
bmoid : BIND-mol-object-id
sub-unit : BIND-object
* state : BIND-mol-object-source
«Integer»
BIND-mol-object-id
«CHOICE»
BIND-mol-object-source
a : Interaction-id
b : Interaction-id
result : BIND-action-ref
«Integer»
Interaction-id
BIND-author
BIND-mol-object-pair
a-bmoid : BIND-mol-object-id
b-bmoid : BIND-mol-object-id
* iid : Interaction-id
BIND-mol-sub-unit-type
descr : VisibleString
stoichiometry : BIND-mol-stoich
«CHOICE»
BIND-mol-stoich
num : BIND-mol-sub-num
bmoids : BIND-mol-object-id
BIND-Rec-coll-descr
descr : VisibleString
* db : BIND-Database-site
BIND-complex-subunit
mcid : Molecular-Complex-id
bmoid : BIND-mol-object-id


BIND-Pathway
date : Date
* updates : BIND-update-object
pid : Pathway-id
pathway : Interaction-id
descr : BIND-path-descr
source : BIND-pub-set
* authors : BIND-author
* division : BIND-Rec-coll-descr
priv : Boolean = FALSE
extref : BIND-other-db
0..n
BIND-update-object
«Integer»
Pathway-id
«Integer»
Interaction-id
BIND-author
BIND-Rec-coll-descr
descr : VisibleString
* db : BIND-Database-site
BIND-cellstage
BIND-path-descr
* descr : VisibleString
* cell-cycle : BIND-cellstage
* pathological-state : BIND-pathol-state
* pathway-actions : BIND-action-ref
* phenotype : BIND-phenotype
BIND-pathol-state
pathway-iid : Interaction-id
cause : BIND-pathol-state-cause
pathol-state : VisibleString
* database : BIND-other-db
* phenotype : BIND-phenotype
* descr : VisibleString
* source : BIND-pub-set
«CHOICE»
BIND-pathol-state-cause
destroyed : NULL
replaced-by : Interaction-id
BIND-other-db
BIND-action-ref
BIND-phenotype
* trait : VisibleString
name : VisibleString
wild-type : ENUMERATED
* descr : VisibleString
* db-links : BIND-other-db
* source : BIND-pub-set
«ENUMERATED»
wild-type
not-specified : Integer = 0
true : Integer = 1
false : Integer = 2

A Pathway - BIND-Pathway

The final top-level biological object in the BIND specification is the BIND-pathway data type. It describes a collection of at least one interaction whose molecules (BIND-objects) form an ordered network of interactions, but are generally free from each other. Common examples include metabolic pathways and cell signaling pathways. Metabolic pathways are usually connected by a series of chemical actions and results of those actions, for the purpose of changing one molecular species into another. Cell signaling pathways are generally connected by binding events sometimes involving chemical actions (e.g. conformational changes or phosphorylation events), for the purpose of transducing information from one place to another

A BIND-Pathway object contains similar administrative information fields as a BIND-Interaction and a BIND-Molecular-Complex. The pathway accession number is called a Pathway-id and is globally unique in BIND. An optional BIND-pub-set is present to store empirical evidence of the pathway. Two other fields in the BIND-pathway object store information describing the pathway. A sequence of Interaction-ids that reference previously defined interactions that make up this pathway is stored. Extra descriptive information regarding the pathway is stored using a BIND-path-descr object. This object can optionally store free text describing the pathway and an optional sequence of BIND-cellstage objects that represent the phases of the cell cycle in which this pathway is in effect. Parts of the pathway may be constitutively present in the cell, while other parts that complete the pathway and allow activation may only be expressed at certain times during the cell cycle. An optional list of BIND-pathol-state objects is also present in the description to store a disease or abnormal phenotype that may be caused by a change from a 'physiologically normal' pathway. BIND-pathol-state object is composed of an Interaction-id that is changed in the abnormal state, the change to the interaction, whether it was destroyed or replaced by another interaction, a list of names describing the pathological state, a list of external database references for the disease, such as OMIM (Hamosh et al., 2002), an optional free text description and an optional BIND-pub-set to store evidence of this pathological state. If multiple actions exist for interactions that define this pathway, a list of actions in the `pathway-actions` field

may be stored to specify the exact list of actions that occur in this pathway. This is required because of the possibility that an interaction is shared between multiple pathways and has slightly different chemical actions in each one. Finally, a BIND-phenotype object may be present to describe the normal phenotype associated with this pathway. For example, the cell may normally be red because of a pigment produced by this pathway.

Other BIND ASN.1 Objects

Publication Set

A BIND-pub-set is used to hold all publications and other evidence in BIND. It contains a list of BIND-pub-objects, a dispute flag and a list of BIND-evidence-object data types. A BIND-pub-object contains an optional free text description of the publication, an enumerated opinion of the publication field, an NCBI Pub object, an optional BIND-quality object and an optional external reference to another publication database. The description field may hold any text data pertaining to the publication referenced by this object. The opinion field may hold the values: `none`, `support` and `dispute`. It is meant to convey the general opinion of the referenced publication in regard to the information in the ASN.1 object that contains the BIND-pub-set. The NCBI Pub object is used to store most of the data in PubMed and can represent almost any publication. It should be used to store a reference to PubMed whenever possible using a PubMed unique identifier (PMID) only. The BIND-quality object stores a quality of information measure as taken from the publication. This is not a database user-based quality assessment. Occasionally, especially in large-scale experiments, data is published accompanied by a quality measure of each data point, possibly based on how many times that data point was tested. This quality measure can be roughly mapped to a percentage in the BIND-quality object and the mapping must be described in the BIND-quality description field. For example, if a paper rates data in four categories of A, B, C and D, then this could be mapped to percentages 100%, 75%, 50% and 25%, with A and 100% representing the best quality data.

The BIND-evidence-object type is designed to describe a user defined piece of evidence when using BIND in a private setting such as a single academic lab or a company. As with a publication, it also contains a free text description, an opinion and quality measure and an external reference. Instead of an NCBI Pub object, it contains an NCBI User-object field that can store any kind of data, even a picture of a gel. Importantly here, the quality measure for the data can be user defined and the external reference may point to a Laboratory Information Management System (LIMS) that stores actual experimental data.

Record Update

If a record is updated in BIND, a description of the update should be added to a BIND-update-object. This object contains a NCBI Date object and a text description field. The description field may contain any information that a database implementation decides to store, but it should be complete and stored in a standard and automatic way within each implementation so that it can be easily parsed. Any information may be stored up to and including the entire previous record in ASN.1 value notation. This data is not meant to be human entered but rather maintained as a machine generated audit trail of any changes made to the record.

Data Exchange and Data Cross-referencing

Data exchange systems and database management data structures have been included in the specification as powerful tools to make implementations more robust.

BIND-Submit is the top-level object for data exchange while the cross referencing system involves many separate top-level data objects.

Data Exchange - BIND-Submit

The BIND-Submit object can be used to exchange any number of the top-level data types in the BIND specification, BIND-Interaction, BIND-Molecular-Complex, and/or BIND-Pathway objects. BIND-Submit stores an NCBI Date object, an optional BIND-Database-Site, a BIND-Submitter object, an optional BIND-Submit-id integer for identifying the submission, a list of BIND accession numbers present in the submission

and fields for optionally storing BIND-Interaction-set, BIND-Complex-set, and BIND-Pathway-set objects.

A BIND-Database-site is a description of a database site. This object could be used if data was being submitted to BIND from any other database. It contains free text description of the database site, usually the database name. Also present is a text field for database country of origin and an optional field used to store the World Wide Web Universal Resource Locator (WWW URL) of the homepage of the database on the Internet. An optional NCBI Pub object can store a PubMed reference for this database.

A BIND-Submitter object contains information about a submitter to a BIND database. BIND-Submitter stores a BIND-Contact-info object, which contains information about a person. A 'hold until published' Boolean flag is present which defaults to false to allow data submission prior to publication. Also present is an optional enumeration of possible submission types, either `not-specified`, `new`, `update`, `revision`, `import`, `export` or `other`. An update is a change by an author while a revision is a non-author update. An optional BIND-Submission-tool contains the name, version and free text description of the tool used to submit the record.

Personal contact information should be kept separate from BIND records to keep the submitter and ownership information anonymous and protected from improper use.

Actual records are stored in the BIND-Submit object in data set data types. The BIND-Interaction-set, BIND-Complex-set and BIND-Pathway-set are all present in the BIND-Submit object and are analogous in that they optionally store the date on which the set was collected, optionally the database from which the record set originates using a BIND-Database-site, and the respective sequence of records.

Cross-referencing the Data

Since the BIND specification describes biological data from interactions to pathways and networks of pathways, the information space represented resembles a largely undirected graph with molecules as vertices and their interactions as edges. Cross-referencing information allows the graph to be easily traversed using simple indexed lookup techniques. If cross-referencing were not used in a system such as this, all records would have to be examined at each traversal of the data space. Instead of

creating traditional large, unwieldy indexes and tables to speed the traversal process, ASN.1 objects are directly specified to store cross-reference information. This represents an object oriented database index system. A BIND database accession number as well as NCBI GI/SLRI DI, PMID and taxonomy ID accession numbers has its own associated cross-reference object. This information may be easily exported and used by other databases to link their sequence or structure data back to BIND. One advantage of having these indexes present over a typical relational index system is that they are fast to load into memory as they only require a single contiguous disk read instead of having to traverse over a large table. Thus, these indexes are optimized for reading and would be most useful in a mainly read-only system. For a system that is mainly write-only, such as the BIND submission system, a relational index would provide better performance, since it is easier to add to than the ASN.1 indexes described above.

When updating cross-reference information, only one level of the graph is traversed, so as not to make the index overly complicated. Any time one of the three top-level objects is created that contains a cross-referenced accession number, the BIND-Cross-Ref object lists can be updated, although as the database scales, the updates to these indexes may need to be performed less often. In this way, any search using a cross-referenced accession number instantly retrieves all of the interaction, complex and pathway records that contain it.

The BIND record cross-reference data is stored in a BIND-Bid-XRef object. This data type contains the BID (generalized BIND accession number, which can be IID, MCID or PID) of the interaction, complex or pathway record being cross-referenced in this object. The `interaction-bids`, `complex-bids` and `pathway-bids` fields contain a list of IIDs, MCIDs and PIDs respectively of interactions, pathways and complexes that contain this BIND-Id. An interaction can be part of a complex and a pathway, a complex can be part of an interaction, a complex, and a pathway and a pathway cannot be part of another record. Thus, indexes only need to be maintained for interactions and molecular complex records.

The GI/DI cross-reference information is stored in a BIND-Seq-XRef object. This object links a biological sequence or domain to a list of interactions, molecular complexes and pathways that contain it.

PMID cross-reference data is maintained in a BIND-Pub-XRef object. This cross-reference scheme is analogous to that of GI/DI accession numbers.

NCBI taxonomy IDs are cross-referenced in a BIND-Tax-XRef object that is again analogous to that of GI/DI accession numbers.

The full cross-reference system allows quick and easy searching of relationships in the database by any of the four indexed accession numbers.

Exported Data Types

Typical ASN.1 data specifications make certain data types available for use by other ASN.1 specifications by exporting them. BIND currently exports the top-level data types BIND-Submit, BIND-Interaction, BIND-Interaction-set, BIND-Pathway, BIND-Pathway-set, BIND-Molecular-Complex and BIND-Complex-set, BIND-cellstage, BIND-object, BIND-object-type-id, BIND-place-set, BIND-condition-set, BIND-loc, BIND-action-set, BIND-state-set, RealVal-Units, Interaction-id, Molecular-Complex-id, Pathway-id and BIND-bid, although more types may be exported for convenience in the future.

Implementation

This section gives an overview of the BIND database. The BIND database may be accessed from the web page http://bind.ca. The implementation allows data entry and data retrieval supporting most of the BIND 3.0 ASN.1 specification. Programmed fully using the C programming language for maximum speed and compatibility, the BIND application programming interface (API) has been written to allow applications to easily use data in the BIND database. The API makes use of two C libraries, the NCBI Toolkit (ftp://ncbi.nlm.nih.gov/toolbox) for ASN.1 handling and more and the CodeBase (http://www.sequiter.com/) database library for a database implementation. Using this API, web-based applications have been developed for data entry, retrieval and management. All data is entered and retrieved using web-based forms generated by CGI

programs written in C. Interaction data is currently being entered using this web-based user interface and the system is constantly being updated with the help of user feedback.

The BIND API has been released under the GNU Public License (GPL) and is available in the SLRI Bioinformatics Toolkit at http://sourceforge.net/projects/slritools.

The BIND database uses the SeqHound database system as a resource (Michalickova et al., 2002). SeqHound is an in-house mirror of GenBank, the NCBI taxonomy database, the PDB (Bernstein et al., 1978) data in NCBI MMDB form (Hogue et al., 1996) and various other bioinformatics data resources. SeqHound derived data allows BIND to quickly and easily use sequence, taxonomy, 3-D molecular structure and molecular function information for validation and for information retrieval.

Future Work

The data specification is under constant examination, since it is already being used in the implementation of BIND. As time passes, the process of modifying the specification will yield mature and stable data types. This process has now been occurring for almost four years with reduced changes being required in the specification with every passing year. Feedback is welcome from anyone using the BIND database or specification. Data visualization and data mining systems have been designed and some of them have been implemented and are described further elsewhere.

Conclusion

A data specification has been presented for a standard way of representing biomolecular interaction, molecular complex and pathway information using the internationally standard ASN.1 data description syntax. The need for such a representation is paramount at this time as the scientific community, and specifically the proteomics community, gears up for an explosion of interaction, molecular complex and pathway data.

The use of and comments on this data specification and the related software tools that members of the BIND project will provide and maintain are encouraged. Data specifications require community input in order to mature and become useful.

Chapter 3 – The Biomolecular Interaction Network Database - Implementation

The majority of the work presented in this chapter has been published as follows (reprinted with permission, copyright Oxford University Press):

Bader, G.D., Donaldson I., Wolting C., Ouellette B.F., Pawson T., Hogue, C.W.V.
BIND-The Biomolecular Interaction Network Database
Nucleic Acids Research Jan 1, 2001 29(1): 242-245

Data contributors:

Cheryl Wolting defined the amino acid post-translational modification library with help from myself, Howard Feldman and Van Le. Ian Donaldson, Cheryl Wolting and Berivan Baskin together entered over 200 database records to help test the data specification.

Abstract

The Biomolecular Interaction Network Database (BIND; http://bind.ca) is a database designed to store full descriptions of interactions, molecular complexes and pathways. Development of the BIND 3.0 data model has led to the incorporation of virtually all components of molecular mechanisms including interactions between any two molecules composed of proteins, nucleic acids, and small molecules. Chemical reactions, photochemical activation and conformational changes can also be described. Everything from small molecule biochemistry to signal transduction is abstracted in such a way that graph theory methods may be applied for data mining. The database can be used to study networks of interactions, to map pathways across taxonomic branches and to generate information for kinetic simulations. BIND anticipates the coming large influx of interaction information from high-throughput proteomics efforts including detailed information about post-translational modifications from mass spectrometry. Implementation, content and the open nature of the BIND project is discussed. The BIND data specification is available as ASN.1 and XML DTD.

Introduction

The Biomolecular Interaction Network Database (BIND) has been designed to store information about biomolecular interactions, molecular complexes and pathways in a computer readable form. This type of data is typically stored as written English text in traditional journal publications and in PubMed, where it is difficult to mine. Because of technological advances and heightened interest, the field of proteomics is generating increasing amounts of scientific data on molecular interactions, pathways and post-translational modification of proteins. Proteomics techniques that generate large amounts of data include high throughput two-hybrid studies and mass spectrometry (Mendelsohn and Brent, 1999). The genome era has taught us that it is important to design and use effective tools for storing and managing data before they become too large. A concerted effort by the biological community is required now to prepare for the interaction information of the near future (Cassman et al., 2000).

The BIND project encompasses a data specification, a database and associated data mining and visualization tools. Goals of the project are to be a public proteomics resource to the community at large and to become a platform for data mining and visualization of interaction information. It is hoped that BIND will help in understanding complex cell signaling networks that play an important role in a number of cellular processes, from development to disease (Pawson, 1995).

Methods

The BIND database is implemented using an object-relational scheme. ASN.1 binary objects are stored with accompanying indexed accession numbers in a relational database using the CodeBase database C library (http://www.sequiter.com). The database layer is completely modular and only a single source file needs to be modified to implement the system with another Database Management System (DBMS). This has already been done with the DB2 system from IBM (http://www.ibm.com/). All programs have been written using the NCBI C Software Toolkit (http://www.ncbi.nlm.nih.gov/Toolbox/ and http://bioinfo.mshri.on.ca/tkcourse/) in ANSI C or in Java. The freely available NCBI toolkit provides automatic C/C++ code generation directly from an ASN.1 specification, for parsing and dealing with ASN.1 objects. Binary ASN.1 objects are compactly encoded and thus efficient to read, write and transmit. The combination of ASN.1 and automatically generated C/C++ code means that programs are rapidly developed to run very quickly across many platforms. The BIND data manager server can run on, and has been tested with, Windows, Linux, Solaris and Mac OS X. It has been developed on primarily Windows NT/2000, Linux and Solaris. The BIND data specification provides a basis for tools to be developed that will be able to communicate with each other and the database across platforms and networks with minimal effort via ASN.1 or XML object transmission. Java was used for the visual navigation applet (Figure 9) because it is natively cross-platform and thus supports running in web browsers on any computer that supports Java. The number of lines of source code that are used by the BIND project are detailed in Table 1

Project	Lines of C Source Code	Perl	Java
BIND	79,074*	-	3,524
BIND Associated (e.g. Collaborations)	25,836	1,720	-
SLRI Library	14,380	-	-
Text Index	4,427	-	-
SeqHound	63,958	3,898	-

Table 1: Physical Source Lines of Programming Code Supporting BIND

Physical source lines of code for BIND and supporting projects are shown, which does not count blank or comment lines, as of September 1st, 2002. BIND Associated code includes analysis programs written for collaborative and other projects. SLRI Library is a library of commonly used functions, originally written for support of BIND, but has since been adopted by other projects in the Hogue lab. The Text Index code enables the word search feature on BIND. SeqHound, as an integrated biological information database system similar to Entrez, is used in parts of BIND when, for example, a protein sequence is required. *21,737 lines of C were automatically generated from ASN.1 by NCBI's asntool program, which is included in this total.

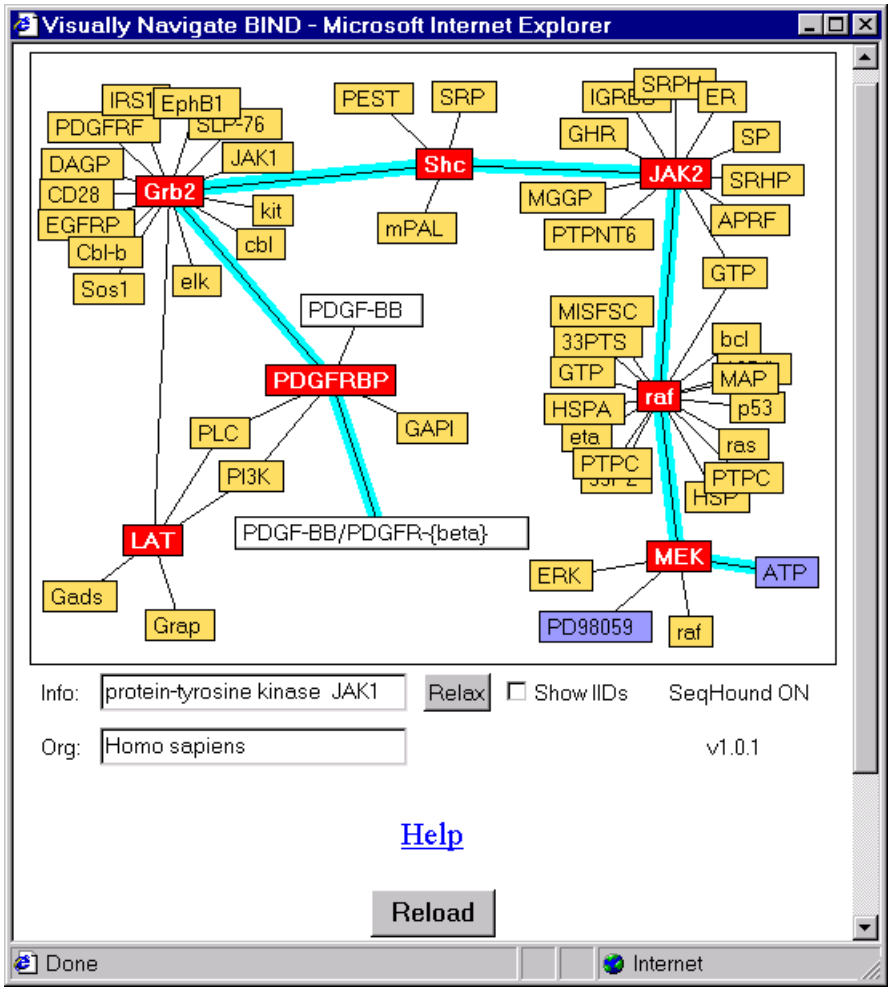


Figure 9: BIND Interaction Viewer Java Applet Showing How Molecules Can be Connected in the Database From Molecular Complex to Small Molecule

In this figure, yellow represents a protein, purple represents a small molecule and white represents a molecular complex. Red signifies that a square is fixed in place and won't be moved by the graph layout algorithm. This session was seeded by the interaction between human LAT and Grb2 proteins involved in cell signaling in the T-cell.

The user interface to BIND is web-based (Figure 10 and Figure 11). Currently a data entry tool allows most data in the specification to be entered and changed. The database may be queried using text from any field, or directly by accession number. An integrated Java applet, as shown in Figure 9, has been written to visually navigate the database starting from any interaction. A BLAST against BIND service has been written, by modifying NCBI's version of webBLAST, that allows a user to search for DNA or protein sequences similar to ones in BIND. The results of the search are linked back to BIND and SeqHound. An online help guide is available via the help link on the Data Manager menu.

BIND has also been designed to function in a distributed manner. Multiple BIND databases may be set up, all using a common Internet based key-server to assign unique accession numbers. Collaborations are easy as information is efficiently shared. The key-server has been designed, but not implemented.

BIND, as a public record submission site, is designed to be composed of four major subsystems (Figure 12):

1. *A submission system for new records entering the database*. As users and indexers enter records, they should be provided a temporary workspace to modify the new record until it is ready to become submitted for indexer validation. Since the system maintains temporary records, it should be purged of unfinished records after a published grace period of a certain number of days. The system must be purged to limit its growth so that it can maintain quick response speed for users. This system has been implemented and should be optimized for multiple record update use. Once the user has decided that the record is finished, they must press a final submit button to submit the record to the indexer queue.

2. *A queue for indexers to examine submitted records*. New records that are entered from users, automated data entry systems, indexers and by any other means should be input into the head of a queue where they can wait to be processed by an indexer. The goal of this subsystem is to ensure that the quality of the record is maintained at a high level. Once an indexer has validated a record, it can be assigned an official accession number from the key server and be submitted into the final public database where it will be made available in its final state unless the user who originally

submitted the record desires to update the information. This system also must be optimized for multiple record updates and needs to keep track of queue related information, such as the status of the record over the validation process and the indexer user rights for modifying records. This system is a modification of the already implemented data submission system, which will allow indexers to use web-based HTML forms to modify the records and follow the submission process.

3. *A final database that is made available for searching to the public*. This database has been implemented and is optimized for querying, as records will only be added to this system once, after they are validated and assigned a unique key. A querying API should access this database only, as it contains final, vetted records.

4. *A universal key server for tracking BIND accession numbers globally for all instances of the BIND database where new records are being created*. This system has been designed and is still under development. Any BIND system linked to the Internet that requires accession number will be able to request them from the key server. This allows the central BIND authority to maintain a stable accession number space. This is required in BIND because records reference one another by accession number and depend on that accession number not to change. The nature of the data to describe biomolecular interaction networks is such that an integrated data set is more useful than the sum of the usefulness of the smaller data sets that it is composed of. Thus, an efficient method of data integration across multiple instances of a BIND database depends on a centrally maintained key server as much as the interoperability of the subsystems of the Internet depends on a central authority for managing Internet Protocol (IP) hardware addresses.

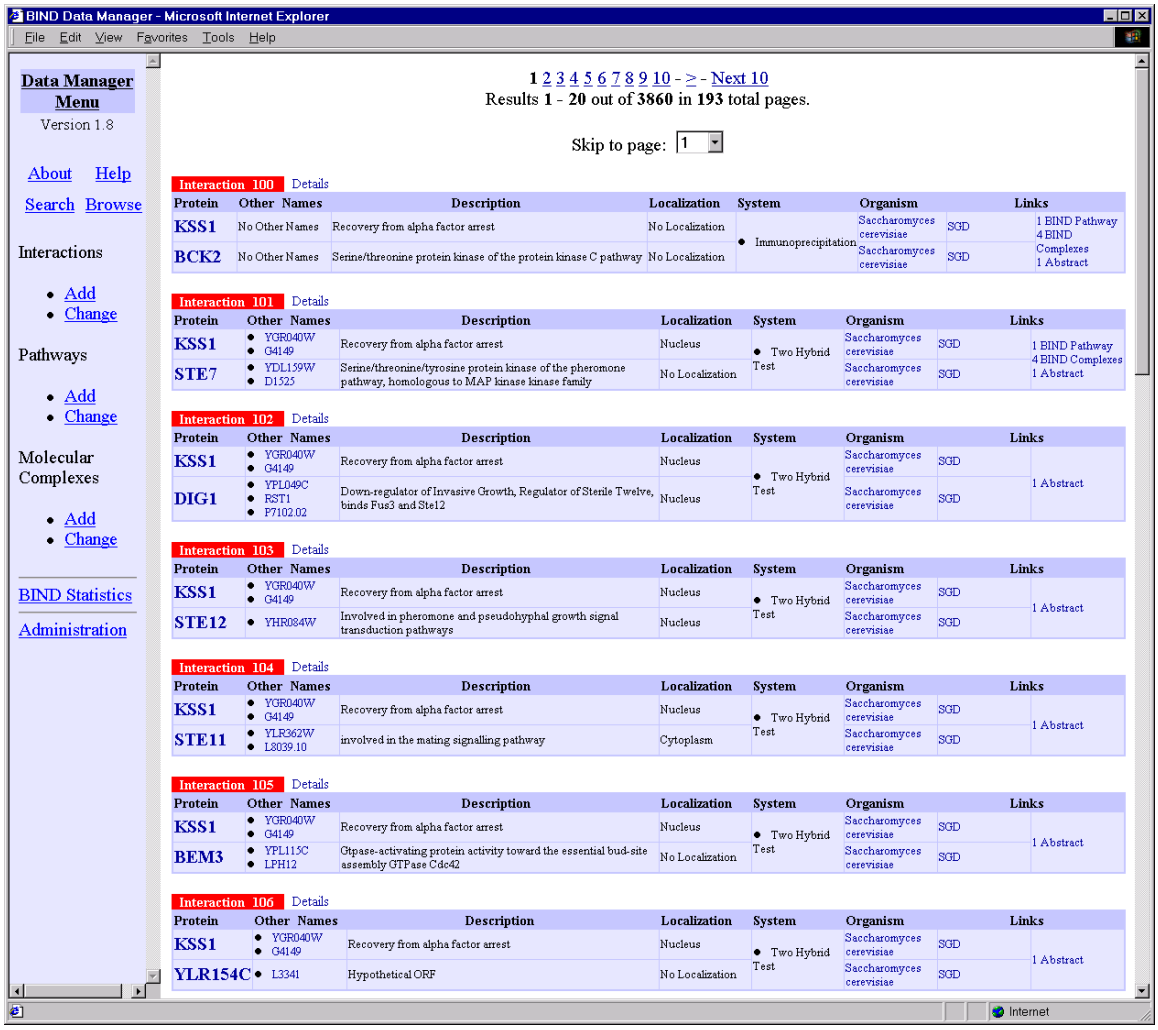


Figure 10: Browsing BIND Via the Web

A summary of each record is provided when browsing the database. Clicking on the 'Details' button sends the user to the view shown in Figure 11 for an interaction record. Summary view browsing is also available for complex and pathway records.

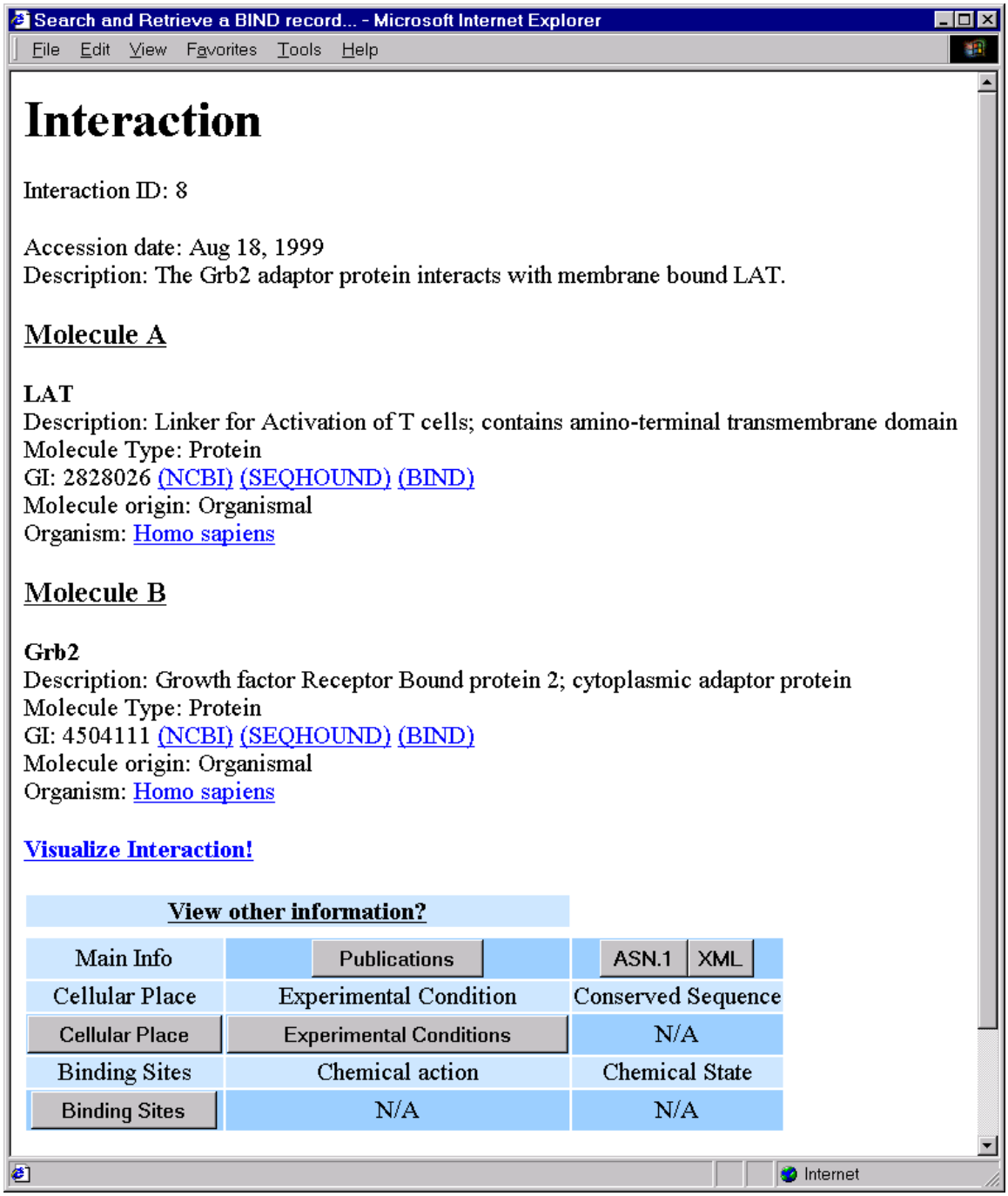


Figure 11: The Detailed View of an Interaction Record

From this view, the user may access more detailed information about this interaction. Detailed views are also available for complex and pathway records.

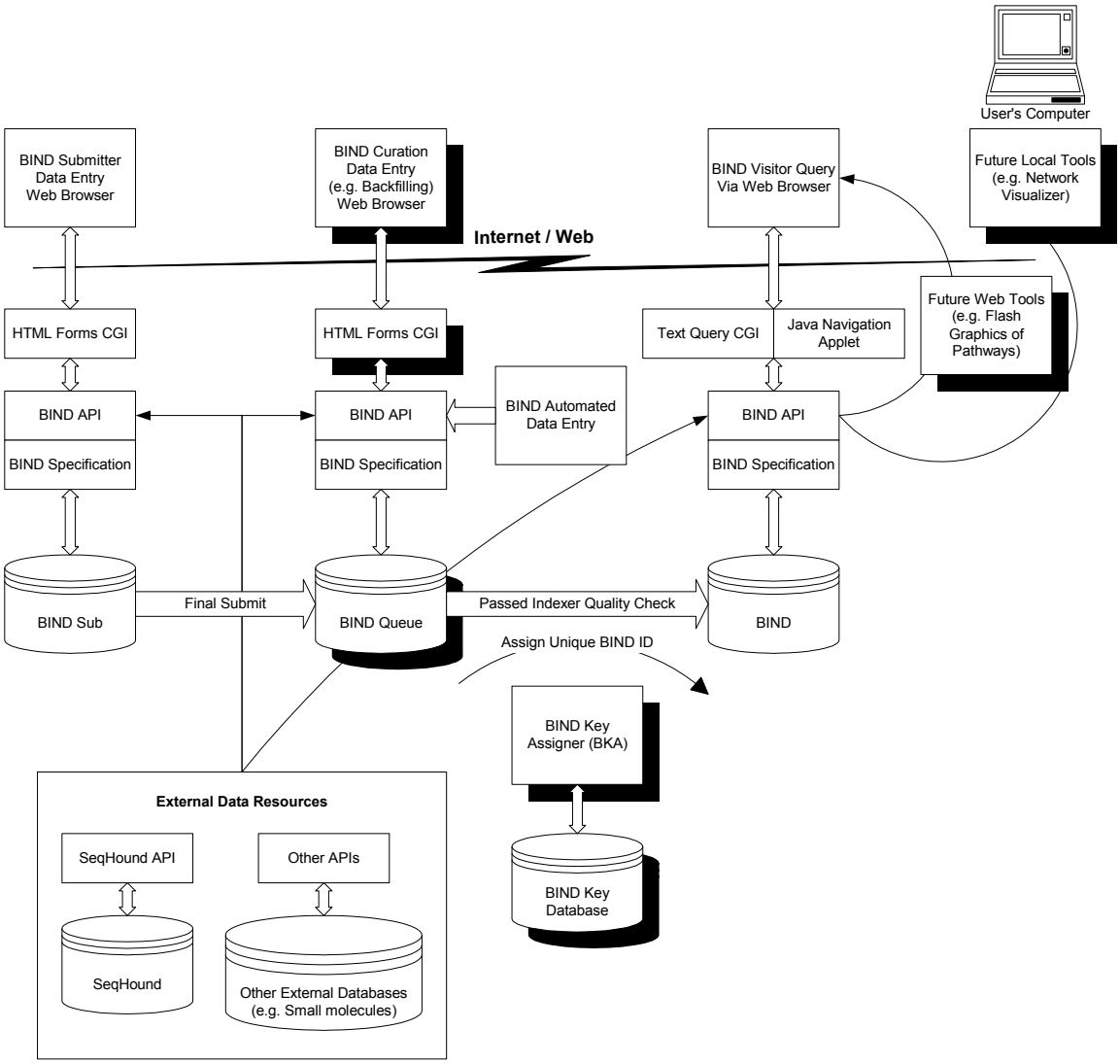


Figure 12: System Diagram of an Integrated BIND Database

An integrated BIND database encompassing data submission (BINDSub), curation/indexer record validation (BINDQueue), the BIND Key Assigner (BKA) and the final quality checked BIND website (BIND). Shadowed boxes represent system components that are not finished or implemented. BINDQueue has not yet been implemented. BKA is currently unfinished. External data resources, such as SeqHound are used for some functionality of the BIND API (e.g. biological sequence retrieval). Future database query, visualization and analysis tools are shown.

The BIND Data Specification

Version 1.0 of the BIND specification was finalized in June 1999 (Bader and Hogue, 2000). Since then, further implementation of the BIND database, data record entry into the database, user feedback, and discussion with other groups working on standard representations of biological function (Karp, 2000), (The Gene Ontology Consortium, 2000), (Cassman et al., 2000) have led to many improvements in version 3.0 (See Chapter 2).

Post-Translational Modifications

Mass spectrometry will provide much information about post-translationally modified proteins (Ficarro et al., 2002) and how these post-translational modifications affect interactions. The current IUPAC nomenclature for amino acids of single or three letter codes is not sufficient for easily representing modified amino acids. An extension to the IUPAC amino acid codes was developed using the infrastructure of the NCBI toolkit to represent 60 common naturally occurring post-translationally modified amino acids such as phospho-tyrosine, hydroxy-proline and hypusine (Table 2). Representative structures for each amino acid in both residue, N and C terminal forms (where appropriate) have been integrated into a custom version of the NCBI8aa encoding rules and the amino acid structure look up table files for Cn3D (Hogue, 1997). Classes of modifications that are represented include acetylation, amidation, formylation, hydroxylation, methylation, phosphorylation, palmitoylation, myristoylation, and geranyl geranylation. Each modified amino acid has a standard symbol in this scheme. This extension allows us to represent the most commonly modified amino acids easily in a sequence code. For example, O4'-phospho-L-tyrosine is represented as [Y:po] which can be used to describe the phosphopeptide ligand of Grb2 as [Y:po]VNV (Salcini et al., 1994) (See Figure 13). This system can be extended to represent more amino acid modifications in the future and was recently adopted in a symbolic system for describing protein-protein interaction domains that is being proposed as a standard (Aasland et al., 2002).

Modified Residue	Symbol	At what position?
ACETYLATED		
N-acetyl-L-alanine	[A:ac]	Amino
N-acetyl-L-arginine	[R:ac]	Amino
N-acetyl-L-asparagine	[N:ac]	Amino
N-acetyl-L-aspartic acid	[D:ac]	Amino
N-acetyl-L-cysteine	[C:ac]	Amino
N-acetyl-L-glutamine	[Q:ac]	Amino
N-acetyl-L-glutamic acid	[E:ac]	Amino
N-acetylglycine	[G:ac]	Amino
N-acetyl-L-histidine	[H:ac]	Amino
N-acetyl-L-isoleucine	[I:ac]	Amino
N-acetyl-L-leucine	[L:ac]	Amino
N2-acetyl-L-lysine	[K:ac]	Amino
N6-acetyl-L-lysine	[K:N6ac]	Any
N-acetyl-L-methionine	[M:ac]	Amino
N-acetyl-L-phenylalanine	[F:ac]	Amino
N-acetyl-L-proline	[P:ac]	Amino
N-acetyl-L-serine	[S:ac]	Amino
N-acetyl-L-threonine	[T:ac]	Amino
N-acetyl-L-tryptophan	[W:ac]	Amino
N-acetyl-L-tyrosine	[Y:ac]	Amino
N-acetyl-L-valine	[V:ac]	Amino
AMIDATED		
L-alanine amide	[A:am]	Carboxy
L-arginine amide	[R:am]	Carboxy
FORMYLATED		
N-formyl-L-methionine	[M:form]	Amino
HYDROXYLATED		
4-hydroxy-L-proline	[P:hy_g]	Any

LIPID MODIFIED		
S-farnesyl-L-cysteine	[C:farn]	Any
S-geranylgeranyl-L-cysteine	[C:ger]	Any
N-palmitoyl-L-cysteine	[C:palm_n]	Amino
S-palmitoyl-L-cysteine	[C:palm_s]	Any
N-myristoyl-glycine	[G:myr]	Amino
N6-myristoyl-L-lysine	[K:myr]	Any
METHYLATED		
N-methyl-L-alanine	[A:meth_n]	Amino
N,N,N-trimethyl-L-alanine	[A:meth_n3]	Amino
omega-N,omega-N-dimethyl-L-arginine	[R:meth_n7]	Any
L-beta-methylthioaspartic acid	[D:meth_b]	Any
N5-methyl-L-glutamine	[Q:meth_n5]	Any
L-glutamic acid 5-methyl ester	[E:meth_o5]	Any
3'-methyl-L-histidine	[H:meth_n4]	Any
N6-methyl-L-lysine	[K:meth_1]	Any
N6,N6-dimethyl-L-lysine	[K:meth_2]	Any
N6,N6,N6-trimethyl-L-lysine	[K:meth_3]	Any
N-methyl-L-methionine	[M:meth]	Amino
N-methyl-L-phenylalanine	[F:meth]	Amino
PHOSPHORYLATED		
omega-N-phospho-L-arginine	[R:po]	Any
L-aspartic 4-phosphoric anhydride	[D:po]	Any
S-phospho-L-cysteine	[C:po]	Any
1'-phospho-L-histidine	[H:po_e]	Any
3'-phospho-L-histidine	[H:po_d]	Any
O-phospho-L-serine	[S:po]	Any
O-phospho-L-threonine	[T:po]	Any
O4'-phospho-L-tyrosine	[Y:po]	Any
OTHER		
L-selenocysteine	[C:sel]	Any
L-selenomethionine	[M:sel]	Any
L-3-oxoalanine	[S:oxal]	Any

2-pyrrolidone-5-carboxylic acid	[E:pyro]	Amino
L-glutamyl 5-glycerylphosphorylethanolamine	[E:gpe]	Any
2'-[3-carboxamido-3-(trimethylammonio)propyl]-L-histidine (diphthamide)	[H:diph]	Any
N6-biotinyl-L-lysine	[K:biotin]	Any
N6-(4-amino-2-hydroxybutyl)-L-lysine (hypusine)	[K:hypu]	Any
N6-retinal-L-lysine	[K:retin]	Any

Table 2: The List of Modified Amino Acids Currently Available for Use by BIND

Symbols that extend the IUPAC one letter code for describing the 20 naturally occurring amino acids. Amino = N-terminal; Carboxy = C-terminal

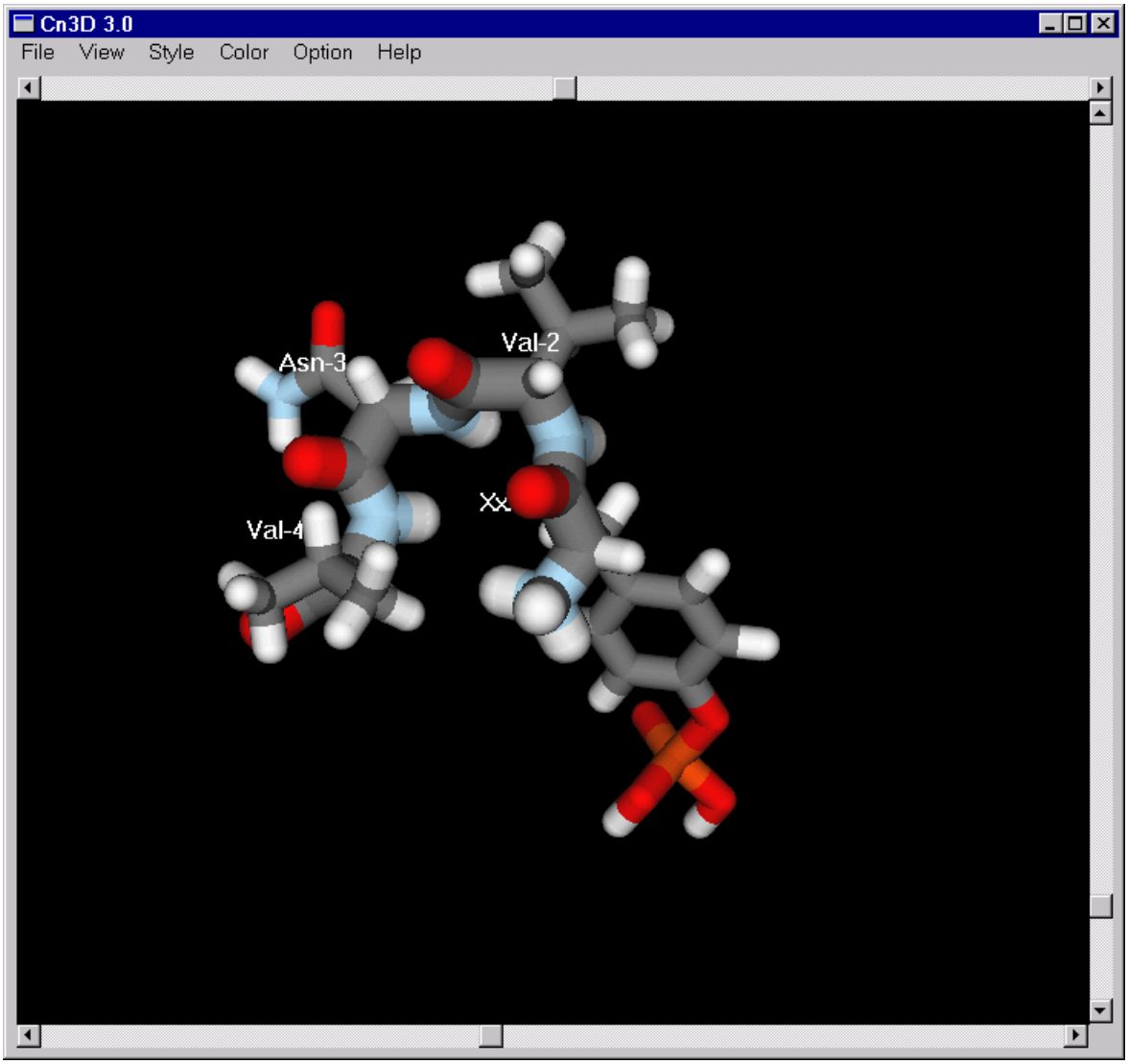


Figure 13: Graphical Representation of the Phosphopeptide Ligand of Grb2 as [Y:po]VNV

This phosphopeptide was generated as a random conformer using the TraDES algorithm (Feldman and Hogue, 2002) and visualized with Cn3D. Note the orange and red phosphate group attached to the tyrosine on the bottom right portion of the peptide.

Data Submission

Data is entered into BIND either by manual or automatic methods. Expert curators on the BIND team are entering high quality records on a continuing basis. Users are encouraged to enter records into the database via the web-based HTML form submission system, or to contact the BIND staff if they have large data sets they want to process. A simple submission involves entering contact information (which only needs to be done the first time the user submits to BIND), the PubMed identifier and two interacting molecules (which can easily be identified by their GIs). Every record that is entered in this way should be validated by BIND indexers and by at least one other expert before it is made available in any public data release.

A system for automatically searching abstracts in the literature for journal articles that contain information about protein and genetic interactions called PreBIND has been written by Ian Donaldson, a post-doctoral fellow in the Hogue lab, and by Joel Martin, a professor at the National Research Council in Ottawa. PreBIND is based on a Support Vector Machine (SVM), a machine learning tool which has been trained to classify abstracts into 'interaction' or 'non-interaction' categories. Papers classified as containing interactions are then manually examined to verify the SVM classification. Once verified, PreBIND can automatically enter the record into BIND. Automated searching systems such as this will speed the backfilling task of curators who can then spend more time entering records from the literature.

The GenBank policy on record ownership is followed as it is hoped that BIND becomes a primary public submission database for interaction, molecular complex and pathway data. Such a policy requires that the person who submits a record owns it and possesses the sole right to edit that record. Records in the public version of BIND are in the public domain.

Tools may also be written using the BIND API to import data from other sources. Such tools have been written to import information from the DIP database (Xenarios et al., 2000) and from recent yeast two-hybrid protein-protein interaction mapping projects (Uetz et al., 2000), (Ito et al., 2000). Databases that contain subsets of the interaction information that can be stored in BIND are increasing in number and are prime

candidates for data import tools. In cases where such databases are free for academic use but are not allowed to be distributed by a third party, the BIND project will make import tools available.

The Open Nature of BIND

BIND is meant to be an open effort to catalogue molecular interactions, complexes and pathways. Not only are records created by BIND indexers being released into the public domain, but all source code has also been released under the GNU general public license (GPL: http://www.fsf.org/) on the Sourceforge open-source project management system (http://sourceforge.net/projects/slritools). Copyright of the software is maintained by the BIND project, but the GPL allows anyone to freely distribute and modify the software source code provided they make their changes available under the GPL. Anyone may then install a copy of BIND on a private web server for laboratory data management use. Allowing anyone to install BIND locally will hopefully encourage people to submit their private data to the public version of BIND once that data is published.

Importantly, people involved in the BIND project strongly believe that standard methods used in a community increase productivity and progress. Thus, the BIND specification is being proposed as an open standard for describing, storing and exchanging biomolecular interaction data in the scientific community.

A 1.0 data release that contains over 1,000 interaction records, 6 pathways and 40 molecular complexes has been made available at ftp://ftp.bind.ca/BIND/DB/ in both XML and ASN.1 formats. Since then, the database has grown to contain over 6,000 interaction records, over 850 molecular complexes and 8 pathways. A project, called MMDBBIND, has also been undertaken to import all molecular interaction present in the PDB database (Westbrook et al., 2002) into BIND format and this has created over 65,000 interaction records (Salama et al., 2002). It has been estimated that there are 2 to 10 protein-protein interactions per protein in a cell (Marcotte et al., 1999). This estimate does not include other types of interactions such as protein-small molecule, of which there are undoubtedly at least as many. For example, this means that the approximately

5,500 *Saccharomyces cerevisiae* interactions so far in BIND may represent about 7% to 45% of the total protein-protein interactions in yeast. It is clear to us that yeast will be the first completely understood organism given the high-throughput experiments currently being undertaken in laboratories around the world.

Future Directions

Now that BIND has a stable data specification, a firm record base and a data release data mining methods are being designed for homologous interaction network finding, for finding pathways and for comparing interactions and scoring their similarity (analogous to BLAST for sequences). The data specification abstracts cellular interactions as a computer science concept of a graph, thus tools from the field of graph theory can be applied to data mining. A similarity algorithm can be used by the homologous interaction network algorithm and to create neighbor tables to deal with redundancy in the database. It is also possible that novel drug targets may be found by examining highly connected nodes in an interaction network (Albert et al., 2000). Investigating automatic data record generation directly from experimental sources, such as mass spectrometric data, is planned. Since BIND can contain information on the cellular place of all involved components of interaction networks and associated kinetics and thermodynamics data, models of cellular processes can be generated automatically for input into kinetics modeling software such as the Virtual Cell (Schaff and Loew, 1999).

Implementation of more advanced query tools, such as searching for proteins with specific domains or searching for interactions where both molecules have a solved 3-D structure, as well as ad-hoc querying need to be implemented. A program needs to be developed to allow visualization of pathways, complexes and networks from BIND in a more advanced way than the BIND Java database navigation tool does currently. Such a program could also allow users to enter their records visually, much as one would create a flowchart on a computer by dragging symbols onto a page. In this case, the symbols would represent molecules and the connections between them. To build BIND into a large repository of useful information, many records must be entered from the literature.

This ‘backfilling’ process will require a large team of curators. Automated data entry tools must be written to import molecular interaction data from other databases, as outlined in Chapter 1, and from published data sets that have not already been imported into BIND. Some of these import tools will require the creation of other databases to be used as resources. For instance, import of metabolic pathway data will depend on the creation of a small molecule database to allow unambiguous matching of small molecule names and aliases to structures. Certain small molecule databases exist, such as LIGAND of the KEGG project (Kanehisa et al., 2002) and Klotho of the Moirai project (http://www.biocheminfo.org/klotho/), although none are free, comprehensive and contain 3-D structures of molecules at the same time.

Implementing the full BIND design, as outlined in Figure 12, will allow scale-up of the system so that all known molecular interaction data can be efficiently stored, queried and analyzed.

Chapter 4 – Representing and Analyzing Protein and Genetic Interactions

The majority of the work presented in this chapter has appeared in the following publications (reprinted with permission from respective copyright holders):

Tong AH, Evangelista M, Parsons AB, Xu H, Bader GD, Page N, Robinson M, Raghibizadeh S, Hogue CW, Bussey H, Andrews B, Tyers M, Boone C.
Systematic Genetic Analysis with Ordered Arrays of Yeast Deletion Mutants
Science 2001 Dec 14;294(5550): 2364-8

Yuen Ho, Albrecht Gruhler, Adrian Heilbut, Gary D. Bader, Lynda Moore, Sally-Lin Adams, Anna Millar, Paul Taylor, Keiryn Bennett, Kelly Boutilier, Lingyun Yang, Cheryl Wolting, Ian Donaldson, Soren Schandorff, Juanita Shewnarane, Mai Vo, Joanne Taggart, Marilyn Goudreault, Brenda Muskat, Cris Alfarano, Danielle Dewar, Zhen Lin, Katerina Michalickova, Andrew R. Willems, Holly Sassi, Peter A. Nielsen, Karina J. Rasmussen, Jens R. Andersen, Lene E. Johansen, Lykke H. Hansen, Hans Jespersen, Alexandre Podtelejnikov, Eva Nielsen, Janne Crawford, Vibeke Poulsen, Birgitte D. Sorensen, Jesper Matthiesen, Ronald C. Hendrickson, Frank Gleeson, Tony Pawson, Michael F. Moran, Daniel Durocher, Matthias Mann, Christopher W. V. Hogue, Daniel Figeys & Mike Tyers
Systematic identification of protein complexes in *Saccharomyces cerevisiae* by mass spectrometry
Nature 2002 Jan 10;415: 180–183

Tong AH*, Drees B*, Nardelli G*, Bader GD*, Brannetti B, Castagnoli L, Evangelista M, Ferracuti S, Nelson B, Paoluzi S, Quondam M, Zucconi A, Hogue CW, Fields S, Boone C, Cesareni G.
A Combined Experimental and Computational Strategy to Define Protein Interaction Networks for Peptide Recognition Modules.
Science 2002 Jan 11;295(5553): 321-324
* equal contribution

Data contributors:

All computational experiments described in this chapter were done by myself except for the yeast proteome searching step in the final section, which was carried out by the lab of Dr. Gianni Cesareni at the University of Rome Tor Vergata, with help from myself and Dr. Charlie Boone at the University of Toronto. Adrian Heilbut from MDS-Proteomics helped with data management and defining noise filters in the HMS-PCI section. MDS-Proteomics contributed data to the HMS-PCI section, Joel Martin at the National Research Council Institute for Information Technology and Ian Donaldson at MDS Proteomics developed the PreBIND search engine and Cheryl Wolting and her group at MDS Proteomics entered data to create the PreBIND literature benchmark in the HMS-PCI section.

Introduction

Advances in the field of proteomics are making current technologies such as DNA microarray, mass spectrometry and yeast two-hybrid, more sensitive and robust (Dutt and Lee, 2000; Mendelsohn and Brent, 1999; Yates, 2000). This has allowed the adaptation of such techniques to a more automated and high throughput experimental approach. The implementation and use of high throughput proteomics systems will generate an immense amount of data on gene expression, molecular interactions and post-translational protein modifications (Blackstock and Weir, 1999; Lockhart and Winzeler, 2000; Pandey and Mann, 2000). This mirrors the events of more than a decade ago, when advances in genomics techniques such as PCR, recombinant libraries, and DNA sequencing led to high-throughput genomic sequencing projects, which created a plethora of information. Just as the genomics sequencing projects required robust information systems to manage their generated data in the past, proteomics projects need such systems now (Cassman et al., 2000).

Two major types of high throughput proteomics projects currently under way are whole genome gene expression profiles and full cell interaction maps. The data that is being generated from these efforts are complementary and can be combined to produce a model of the interaction network (including metabolic and signaling pathways) of the cell over time. Single cell organisms, such as *Saccharomyces cerevisiae,* can be mapped over the full range of the cell cycle, and multi-celled organisms, like *Caenorhabditis elegans* or *Homo sapiens*, can be mapped not only over the range of the cell cycle, but over the full range of the developmental cycle (Lockhart and Winzeler, 2000; Pandey and Mann, 2000). These undertakings, while possible, might require even more effort than the human genome project on which they are based.

The BIND database has been designed with proteomics projects in mind. The database can store descriptions of biomolecular interactions, molecular complexes and pathways. The data generation potential of genomics and proteomics as well as the data requirements of data mining and systems modeling (both kinetic and physiological) have been taken into account in the design of BIND. It is hoped that proteomics databases such as BIND will aid in the combination of diverse data and fields to provide a deeper

understanding of biology that will eventually create a full and detailed computer model of *Homo sapiens*.

Data Mining

Information management is only part of the work of dealing with large amounts of scientific data. Knowledge value comes from work involved in analyzing and understanding the data. It is obvious that manual data analysis does not scale well with the size of the data. Computer tools are required to intelligently search, filter and present data to human experts in such a way that the information of interest can be examined quickly and easily. These tools form the basis for data mining.

Data mining can be defined as the analysis of existing data for finding relationships that have not previously been discovered. It is hoped that new knowledge may be gleaned from BIND by various data mining methods. One major advantage of developing and implementing data mining algorithms is that they can be automatically run on a regular basis. As new records fill in information gaps in the database, automated data mining will regularly produce new results.

A key idea, since the conception of the BIND data model, is that biological interaction data can be represented as a connectivity graph. This allows the application of computer science graph theory algorithms to biological data mining. Other methods of data mining exist, most notably statistical clustering (Eisen et al., 1998), and many are fundamentally mathematically similar and can be used to solve the same problems. Graph theory provides a very intuitive representational abstraction here and is convenient for problem solving.

Graph theory is based on the notion of a *graph* (Figure 14), a representation of connected data as a set of vertices (or nodes) and a set of connecting edges possibly containing cycles. Acyclic graphs are called trees and a collection of trees is termed a forest.

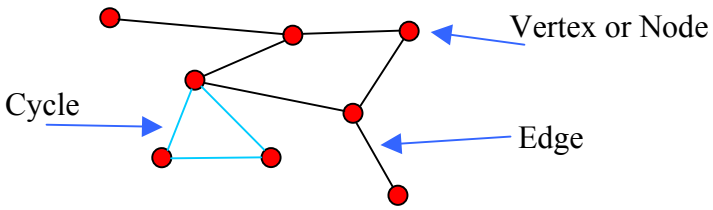


Figure 14: Basic Concepts of a Graph

Edges may be directed and may have an associated weight or a colour. Nodes may also have weight and colour (Figure 15). Some graph algorithms make use of edge direction, weight and colour. The number of edges that are directed into a node is called the in-degree of the node while the number of edges that are directed out of a node is called the out-degree.

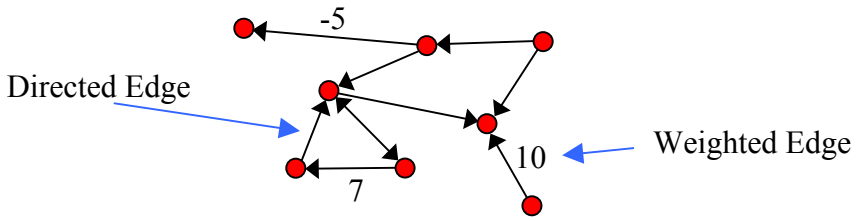


Figure 15: Further Basic Graph Theory Concepts

Note that a graph is a completely abstract mathematical concept and can be mapped to any problem where a mapping can be imagined, thus direction and weight do not have meaning until a mapping is made.

The data in BIND can be mapped to this connectivity model by representing biomolecules as nodes on the graph and interaction information as edges. Edge direction may be mapped from cell signaling and chemical action information. Edge weight may be derived from kinetics, publication opinion, experimental system type, quality of data, or from user defined weighting functions.

There is a distinction between metabolic pathways and cell signaling pathways that must be taken into account when building the graph. Metabolic pathways are usually connected by a series of chemical actions and the chemical results of those actions, for

the purpose of changing one molecular species into another. Cell signaling pathways are generally connected by binding events, sometimes involving chemical actions (e.g. phosphorylation events), for the purpose of communicating information from one place in an organism to another. This means that more information than just the fact that two molecules interact must be taken into account to faithfully map the biology to the graph. This point is missed by other groups using graph theory to examine interactions such as PFBP (van Helden et al., 2000) and others (Albert et al., 2000; Eisenberg et al., 2000).

There is also a distinction between pathways and interaction networks. Pathways are human constructs that help us to functionally organize interaction networks while interaction networks are not necessarily pathways. The EGFR pathway from the cell membrane to the nucleus may represent a single path in a cellular network, but is generally thought of as being separate from other pathways, at least in function. Thus, cross talk with other pathways is generally not taken into account. The holistic network view does not assume the modularity of a path in a network, although it may very well exist, thus should be able to uncover previously unseen connections among known cellular components.

Another problem in mapping BIND data to a graph is the redundancy of the underlying DNA and protein databases. A given node in the graph must represent a molecule in BIND uniquely, yet any biological sequence molecule may be referenced in BIND using one of many GenBank accession numbers. This has been dealt with by using a database of redundant accession numbers that is integrated into the SeqHound database system (Michalickova et al., 2002), an integrated biological database system similar to Entrez (Benson et al., 2002). SeqHound will return the accession numbers of proteins that have exactly the same sequence as a given protein, which Entrez does not do.

Once a mapping has been defined from the data of interest to a graph, graph theory algorithms may be implemented for data mining. The result of mapping BIND to a graph has was used in preliminary data mining work in which a shortest path algorithm was implemented to examine the properties of two different BIND data sets. One set was imported from a protein-protein interaction database (DIP) and one from the results of a recent high-throughput yeast two-hybrid screen (Uetz et al., 2000). Unfortunately, after

examining the results, it was found that these data sets contain many physiologically irrelevant records. DIP contains many cross-species interactions, since many are taken from the PDB molecular structure database. The yeast two-hybrid data set may contain false positive interactions that are inherent in the experimental technique. Together, these problems lead to artificially extended interaction networks that do not represent physiological conditions, such as a path that extends from human to yeast to cow and back again to human. To remedy this situation, high quality data sets must be developed. One such data set is the curated set of BIND records. Another is the set of molecular interactions in the Yeast Proteome Database (YPD) (Costanzo et al., 2001) or the Munich Information Center for Protein Sequences (MIPS) *Saccharomyces cerevisiae* database (Mewes et al., 2002).

Visualizing and Analyzing Genetic Interaction Networks

Introduction

For the budding yeast *Saccharomyces cerevisiae*, large-scale gene deletion analysis has shown that over 80% of the ~6,200 predicted or known yeast genes are not required for viability. Thus, many genes and pathways of eukaryotic cells may be functionally redundant or buffered from phenotypic consequences after genetic perturbation (Hartman et al., 2001). Due to the remarkable degree of genetic redundancy in yeast, the functions of thousands of yeast genes remain obscure. In collaboration with the lab of Dr. Charlie Boone at the Banting and Best Department of Biomedical Research at the University of Toronto, data mining of a large data set of synthetic lethal genetic interactions was undertaken. To evaluate function, the Boone lab developed an automated method for systematic construction of double mutants, termed synthetic genetic array (SGA) analysis, in which a yeast strain that carries a mutation in a query gene was crossed to an ordered array of ~4,600 viable gene deletion mutants. Double mutant meiotic progeny that were inviable or compromised for growth identified functional relationships between genes. The results of eight screens with genes involved

in actin cytoskeleton control and DNA synthesis and repair generated a network containing 204 genes and 291 genetic interactions. (Tong et al., 2001)

Experimental Method

Redundant functions can often be uncovered by synthetic genetic interactions, usually identified when a specific mutant is screened for second-site mutations that either suppress or enhance the original phenotype. In particular, two genes show a "synthetic lethal" interaction if the combination of two mutations, neither by itself lethal, causes cell death (Guarente, 1993). Synthetic lethal relationships may occur for genes acting in a single biochemical pathway or for genes within two distinct pathways if one process functionally compensates for or buffers the defects in the other (Hartman et al., 2001). Synthetic lethal screens have been applied successfully to identify genes involved in cell polarity, secretion, DNA repair, and numerous other processes (Bender and Pringle, 1991; Mullen et al., 2001). Despite the utility of this approach, just one or two different interactions are typically identified in a single screen (Hartman et al., 2001).

To enable high-throughput synthetic lethal analysis, the Boone lab assembled an ordered array of ~4,600 viable yeast gene deletion mutants and developed a series of robotic pinning procedures in which mating and meiotic recombination are used to generate haploid double mutants. The final pinning results in an ordered array of double mutant haploid strains, whose growth rate is monitored by visual inspection or image analysis of colony size on a growth plate. This procedure is referred to as synthetic genetic array (SGA) analysis. As assessed by tetrad dissection, the method is subject to ~20 to 50% false-positive interactions, depending upon the number of times a particular screen has been repeated. SGA analysis identified almost all (~92%) previously known interactions associated with the viable deletion mutants.

Visualization

Because genetic interactions can be represented as binary gene-gene relationships, multiple SGA screens should generate a network of genetic interactions that depicts the functional relationships between genes and pathways (Figure 16). Data from the SGA synthetic lethal interaction network was assembled as a list of yeast gene name pairs. The yeast import tool for the BIND project was used to convert the gene name pairs into BIND gene-gene interaction records, which were imported into BIND (BIND-IDs 6,600 to 6,890). This tool integrates yeast information from SGD (http://genome-www.stanford.edu/Saccharomyces/), YPD (http://www.incyte.com/), RefSeq (http://www.ncbi.nlm.nih.gov/LocusLink/refseq.html), the SGD yeast gene registry, the list of essential genes from the yeast deletion consortium (Winzeler et al., 1999) and GO terms (Dwight et al., 2002; The Gene Ontology Consortium, 2000) in order to unambiguously assign any yeast gene name, present in these resources, to an NCBI RefSeq biological sequence. For network visualization and analysis, BIND can export an arbitrary molecular interaction network as a Pajek network file, which can be viewed with the Pajek program for large network analysis (Batagelj and Mrvar, 1998). Pajek was originally designed for the graphical analysis of social interactions (White et al., 1999). The format of the Pajek network file can be found on the Pajek web site (http://vlado.fmf.uni-lj.si/pub/networks/pajek/). Network visualization allows rapid human understanding of relationships among graph components compared to simple lists of gene interactions.

Figure 16: Genetic Interaction Network Representing the Synthetic Lethal/Sick Interactions Determined by SGA Analysis

Genes are represented as nodes and interactions are represented as edges that connect the nodes, 291 interactions and 204 genes are shown. The genes are colored according to their YPD cellular roles. This was done automatically by BIND upon network export using selected cellular role annotation for each protein involved in the network. For genes assigned multiple cellular roles, one was picked that was considered the most probable based upon a review of published abstracts for studies concerning the gene. The network was visualized with Pajek using the Kamada-Kawai automatic layout algorithm (Kamada and Kawai, 1989) with subsequent manual alterations to remove node overlap and to visually cluster the nodes by cellular role. The network contains the interactions observed for the eight query genes, *BNI1*, *BBC1*, *ARC40*, *ARP2*, *BIM1*, *NBP2*, *SGS1*, and *RAD27,* screened with SGA. The function of the genes with unknown cellular roles (colored black) is predicted by the roles of surrounding genes that show a similar connectivity.

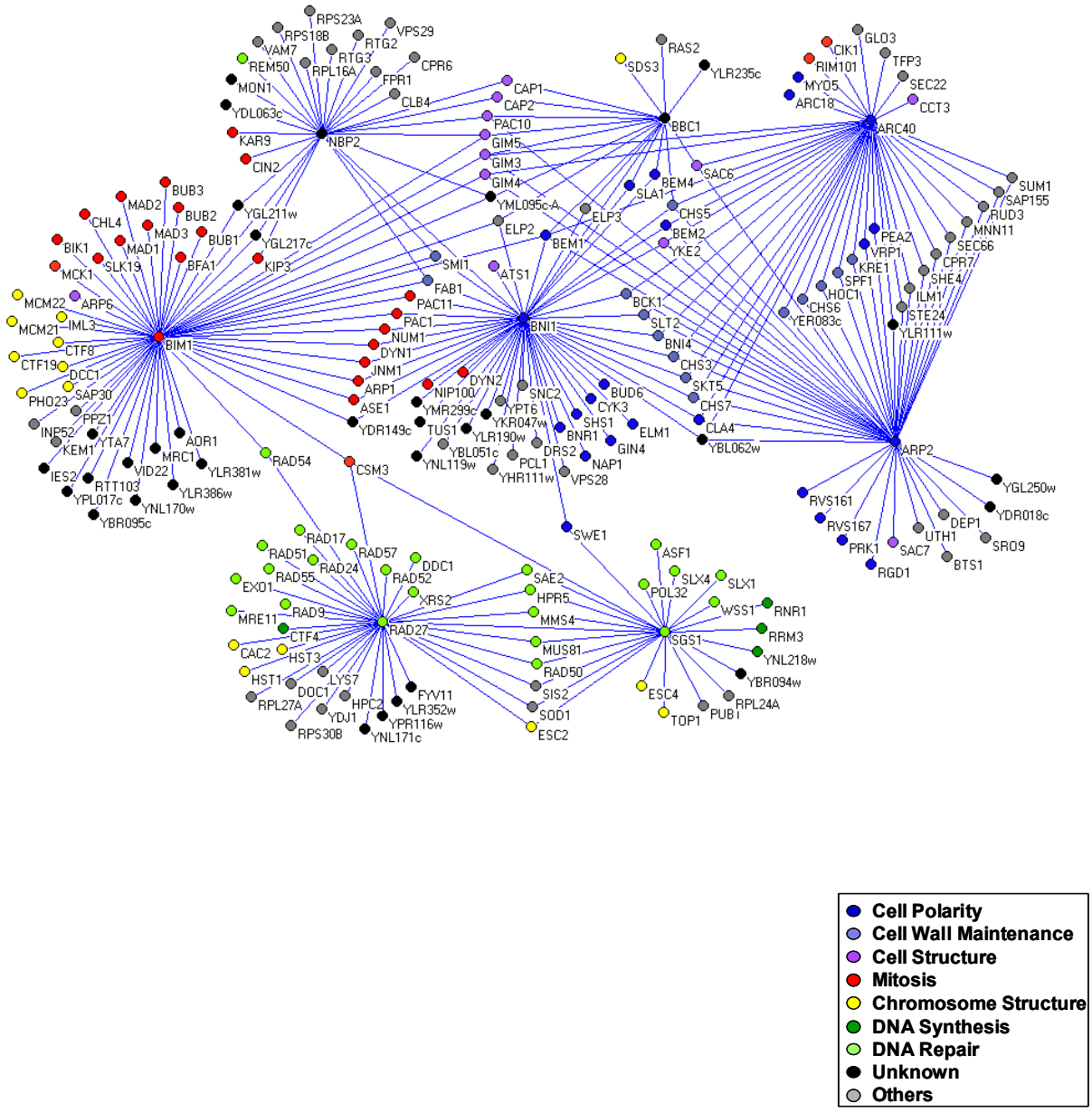

RPS23A
RPS18B
VPS29
VAM7
RTG2
RTG3
REM50
RPL16A
CPR6
MON1
FPR1
CLB4
YDL063c
KAR9
NBP2
CIN2
CAP1
CAP2
PAC10
GIM5
GIM3
GIM4
YML095c-A
ELP2
ELP3
RAS2
SDS3
YLR235c
BBC1
SAC6
BEM4
SLA1
CHS5
BEM2
YKE2
BEM1
GLO3
CIK1
TFP3
RIM101
MYO5
SEC22
ARC18
CCT3
ARC40
SUM1
SAP155
RUD3
MNN11
SEC66
PEA2
VRP1
CPR7
KRE1
SPF1
SHE4
HOC1
ILM1
CHS6
YER083c
STE24
YLR111w
BUB3
MAD2
BUB2
CHL4
MAD3
BUB1
YGL211w
YGL217c
BIK1
MAD1
SLK19
BFA1
KIP3
MCK1
SMI1
ATS1
FAB1
PAC11
PAC1
NUM1
DYN1
JNM1
ARP1
ASE1
YDR149c
BNI1
BCK1
SLT2
BNI4
CHS3
SKT5
CHS7
CLA4
YBL062w
DYN2
NIP100
YMR299c
TUS1
YBL051c
YNL119w
SNC2
YPT6
YKR047w
YLR190w
PCL1
YHR111w
BUD6
CYK3
SHS1
BNR1
DRS2
NAP1
ELM1
GIN4
VPS28
MCM22
ARP6
MCM21
IML3
CTF8
CTF19
DCC1
PHO23
SAP30
PPZ1
INP52
KEM1
YTA7
IES2
RTT103
YPL017c
YNL170w
YBR095c
AOR1
MRC1
VID22
YLR381w
YLR386w
BIM1
RAD54
CSM3
SWE1
ARP2
YGL250w
YDR018c
RVS161
RVS167
PRK1
SAC7
DEP1
UTH1
SRO9
RGD1
BTS1
RAD17
RAD51
RAD24
RAD57
DDC1
RAD55
RAD52
EXO1
XRS2
MRE11
RAD9
RAD27
CTF4
CAC2
HST3
HST1
LYS7
DOC1
FYV11
RPL27A
HPC2
YLR352w
YDJ1
YPR116w
RPS30B
YNL171c
SAE2
HPR5
MMS4
MUS81
RAD50
SIS2
SOD1
ESC2
ASF1
SLX4
SLX1
POL32
WSS1
RNR1
SGS1
RRM3
YNL218w
YBR094w
ESC4
RPL24A
PUB1
TOP1
Cell Polarity
Cell Wall Maintenance
Cell Structure
Mitosis
Chromosome Structure
DNA Synthesis
DNA Repair
Unknown
Others

Figure 16

Further Analysis

A program was written to find yeast genetic or physical interactions in a list where either one or both members of the interaction are of interest. This program was used to identify 72 known physical interactions where both proteins in the interaction correspond to products of genes within the SGA synthetic lethal network. The list of known yeast physical interactions that was used (8,429 protein-protein interactions) were imported from YPD, MIPS and previous large-scale genome-wide screens for comparison (Drees et al., 2001; Ito et al., 2001; Uetz et al., 2000). The network was then visualized using Pajek (Figure 17). To assess the significance of observing 72 genes whose products interact within this data set, 1,000 random networks were constructed that each contained 204 genes chosen randomly from the yeast genome, the same number as in the SGA network. On average, 7.8 genes (~4%; SD = 4.1) within the random data set had products that occurred within the protein-protein interaction data sets. Thus, the synthetic lethal network enrichment for genes whose products interact was highly unlikely to occur by chance. The products of many of the interacting genes occur within pathways probably because the readout of the pathway is required for life in strains carrying the synthetic lethal query mutation.

A list of previously known synthetic lethal interactions was created by merging 1,142 known synthetic lethal interactions provided by YPD (as of July 29, 2001; http://www.incyte.com/) with 535 known synthetic lethal interactions from MIPS, which resulted in 1,291 unique interactions. The MIPS list was downloaded on Aug 7, 2001 (http://mips.gsf.de/proj/yeast/tables/interaction/genetic_interact.html) and manually edited to extract synthetic lethal interactions. The overlap with the SGA genetic interactions with this list was used to determine the rate at which the SGA method uncovers previously known synthetic lethal interactions (~92%), as mentioned above.

Having both a large list of known physical protein-protein interactions and a list of known synthetic lethal interactions allowed an overlap of these lists to be calculated. The result is that 240 physical protein-protein interactions overlap with the synthetic lethal interaction set, and overlap rate of ~19% of the genetic interactions. Thus genetic

interactions not only provide a significant indication of gene function, but also may suggest a physical protein-protein interaction.

Conclusion

By linear extrapolation of the results presented here to the entire yeast genome of over 6,200 genes, it can be estimated that on the order of 300 SGA screens covering judiciously selected query genes will provide an effective working genetic scaffold, which should reveal many of the molecular mechanisms behind genetic robustness and buffering. As gene function is often highly conserved, a comprehensive functional genetic map of *S. cerevisiae* will provide a template to understand the relationships among analogous pathways in metazoans. With the advent of systematic genetic perturbation methodologies, such as large-scale RNAi analysis of gene function in *C. elegans* (Barstead, 2001), the SGA approach is in principle applicable to metazoan systems.

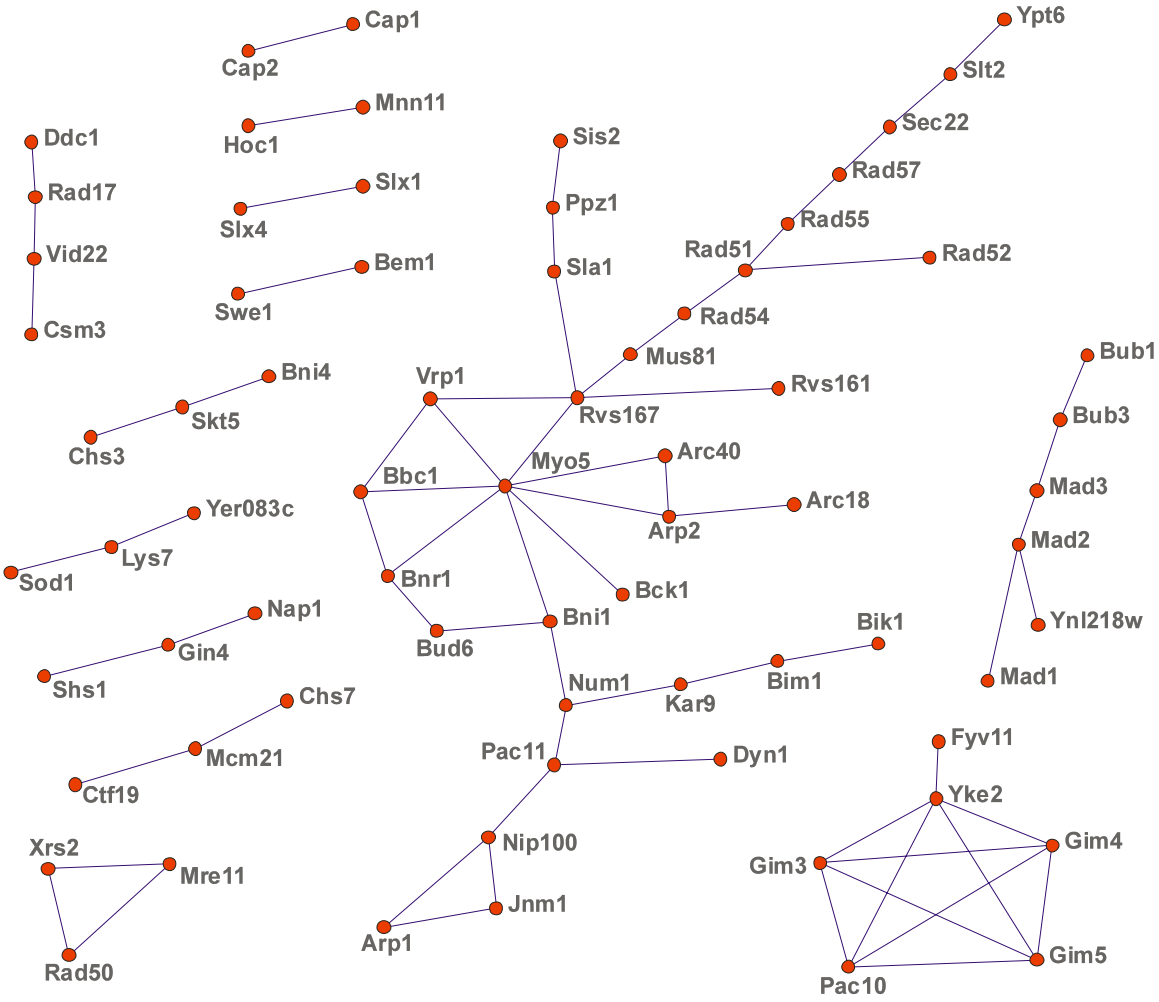


Figure 17: Overlap of SGA Genetic Interaction Network With the Known Physical Protein-Protein Interaction Network

72 protein-protein interactions encompassing 72 proteins corresponding to genes within the SGA genetic interaction network are shown, visualized using Pajek.

Visualizing and Analyzing Protein Interaction Networks from a Large-Scale Mass Spectrometry Experiment

Introduction

The recent deluge of genome sequence data has brought an urgent need for systematic proteomics to decipher the encoded protein networks that dictate cellular function (Pawson and Nash, 2000). To date, generation of large-scale protein-protein interaction maps has relied on the yeast two-hybrid system, which detects binary interactions via activation of reporter gene expression (Fields and Song, 1989; Ito et al., 2001; Uetz et al., 2000). With the advent of ultrasensitive mass spectrometric protein identification methods, it is feasible to directly identify protein complexes on a proteome-wide scale (Neubauer et al., 1997). In collaboration with the lab of Dr. Mike Tyers at the Samuel Lunenfeld Research Institute at Mount Sinai Hospital affiliated with the University of Toronto and MDS Proteomics based in Toronto, data mining of a large data set of protein-protein interactions was undertaken. Using budding yeast as a test case, Dr. Tyers *et al.* reported the first example of this approach, which is termed high-throughput mass spectrometric protein complex identification (HMS-PCI). Beginning with 10% of predicted yeast proteins as baits, 3,618 associated proteins were detected covering 25% of the yeast proteome. Numerous protein complexes were identified, including many novel interactions in various signaling pathways and in the DNA damage response. Comparison of the HMS-PCI data set to interactions reported in the literature revealed an average 3-fold higher success rate in detection of known complexes compared with large-scale two-hybrid studies (Ito et al., 2001; Uetz et al., 2000). Given the high degree of connectivity observed in this study, even partial HMS-PCI coverage of complex proteomes, including that of humans, should allow comprehensive identification of cellular networks. (Ho et al., 2002)

Experimental Method

To survey the yeast proteome, an initial set of 725 bait proteins was chosen representing a variety of different functional classes, including 100 protein kinases, 36 phosphatases and regulatory subunits, and 86 proteins implicated in the DNA damage response (DDR). A small scale, one-step immunoaffinity purification based on the FLAG epitope tag was used to capture bait proteins, which were transiently over-expressed from the heterologous *GAL1* or *tet* promoters. Proteins from 1,558 individual immunoprecipitations were resolved by SDS-PAGE, visualized by colloidal Coomassie stain, excised from the gel and subjected to tryptic digestion prior to mass spectrometric analysis (Figure 18). As the isolation procedure often yielded multiple proteins from single excised bands, which cannot be resolved by peptide-mass-fingerprinting alone, MS/MS fragmentation was used to unambiguously identify proteins in each gel slice (Mann et al., 2001). 15,683 gel slices were processed, yielding approximately 940,000 MS/MS spectra that matched sequences in the protein sequence database. Over 35,000 protein identifications were made in total, corresponding to 8,118 potential interactions with a set of 600 bait proteins that were expressed at detectable levels. Ubiquitous non-specifically binding proteins, defined empirically based on frequency of occurrence, were subtracted from the raw data set to yield 3,618 interactions with 493 baits, representing 1,578 different interacting proteins or 25% of the yeast proteome. In a preliminary direct validation of the HMS-PCI data set, 64 of 86 interactions (74%) in a random set of novel associations detected by HMS-PCI were recapitulated in immunoprecipitation-immunoblot experiments. The HMS-PCI method was able to identify known complexes from a variety of subcellular compartments, including the cytoplasm, cytoskeleton, nucleus, nucleolus, plasma membrane, mitochondrion and vacuole (Table 5). Of all the proteins identified, 531 corresponded to hypothetical uncharacterized proteins predicted from the yeast genome sequence (Ho et al., 2002). Data is available at http://www.mdsp.com/yeast and http://bind.ca.

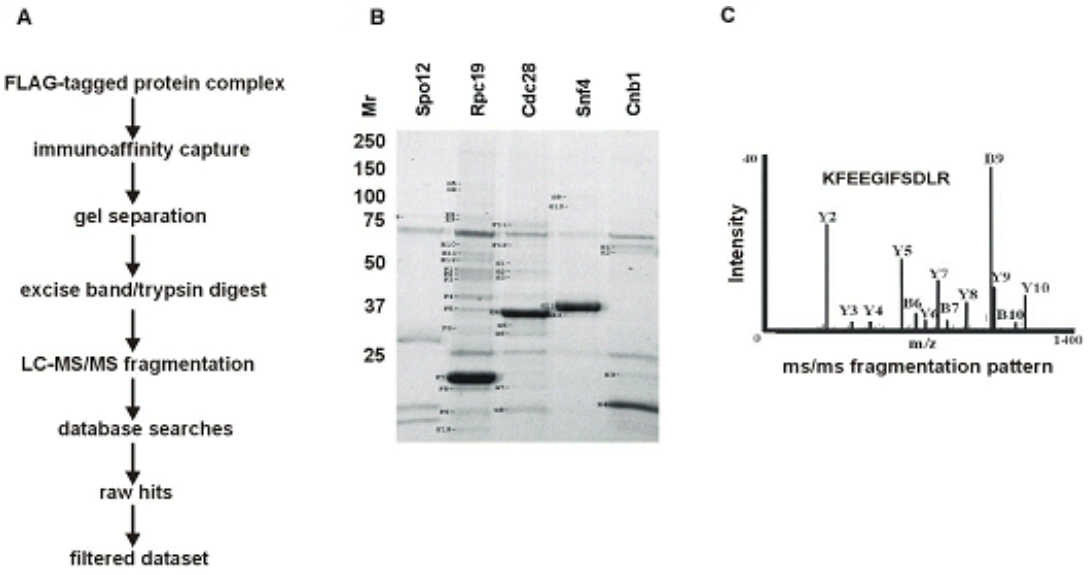


Figure 18: HMS-PCI Experimental Method Strategy

A) Flow diagram of approach **B**) Protein complexes captured onto anti-FLAG agarose resin, eluted and resolved by SDS-PAGE **C**) Proteins specific to the elution are excised, digested with trypsin and subject to LC-MS/MS. Matches of fragmentation spectra to databases unambiguously identify proteins in the sample, as shown here for Ste12.

Annotating the Resulting Data

In order to perform functionally relevant queries with the resulting data set, yeast proteins were annotated using terms from the Gene Ontology (GO) project (http://www.geneontology.org) (Dwight et al., 2002; The Gene Ontology Consortium, 2000). A subset of terms from the 'Biological Process' and 'Cellular Component' GO ontologies were selected to form a generalized categorization of *Saccharomyces cerevisiae* cellular localizations and biological processes. Some related GO terms were collapsed into a single category. For example, "endoplasmic reticulum" and "Golgi apparatus" were combined to form the "endoplasmic reticulum/Golgi" category. This annotation system was created in consultation with Drs. Yuen Ho and Mike Tyers, two experts in yeast biology. The set of terms was designed to be small, yet specific enough to differentiate among annotation considered to be significantly different according to criteria important in yeast biology. For instance, carbohydrate metabolism was separated out of general metabolism because sugar pathways have been one of the most intensely studied in yeast (Ideker et al., 2001). Annotation was performed from the set of GO terms downloaded from the GO FTP site on November 6, 2001. The GO selected term subset is shown below for the two ontologies used here. If a category is the result of combining more than one GO term or changing the name of a GO term, the original individual term(s) are shown in brackets (Table 3 and Table 4).

Derived Annotation Term	Combination of these GO terms (if applicable)
ascus	
bud	
cell wall	(external protective structure + cell wall)
cytoplasm	
cytoskeleton	
endoplasmic reticulum/Golgi	(endoplasmic reticulum + Golgi apparatus)
extracellular	
intracellular	
lysosome/peroxisome/vacuole	(lysosome + peroxisome + vacuole)
mitochondrion	
nucleolus	
nucleus	
plasma membrane/nuclear membrane	(plasma membrane + nuclear membrane)
shmoo	
unknown	(unknown + unlocalized + cell + obsolete)

Table 3: GO Cellular Component Ontology Selected Term Subset

Derived Annotation Term	Combination of these GO terms (if applicable)
aging	
autophagy	
budding	
carbohydrate metabolism	
cell adhesion	
cell cycle	(cell cycle + cell proliferation)
cell growth and/or maintenance	
cell organization and biogenesis	
cell shape and cell size control	
chromosome organization and biogenesis	
DNA damage response and repair	(DNA damage response + DNA repair)
DNA metabolism	
DNA recombination	
DNA replication	
general metabolism	(metabolism + respiration)
mating	(mating (sensu Saccharomyces))
mating-type determination	
nucleolar and ribosome biogenesis	(nucleologenesis + nucleolus organization and biogenesis + ribosome biogenesis)
nutritional response pathway	
protein amino acid phosphorylation/dephosphorylation	(protein amino acid phosphorylation + protein amino acid dephosphorylation)
protein biosynthesis	
protein degradation	(vacuolar protein degradation + protein degradation)
protein metabolism and modification	
protein transport	
RNA localization and processing	(RNA processing + RNA localization)
signal transduction	
sporulation	(sporulation (sensu Saccharomyces))
stress response	(stress response + osmotic response)
transcription	
transport	
unknown	

Table 4: GO Biological Process Ontology Selected Term Subset

GO Cellular Component Annotation Term	Attempted Baits	Expressed Baits	Co-localized Associations
ascus	2	2	2
bud	11	8	12
cell wall	1	0	0
cytoplasm	82	53	139
cytoskeleton	21	13	25
endoplasmic reticulum/Golgi	8	4	3
extracellular	2	2	2
intracellular	149	93	285
lysosome/peroxisome/vacuole	5	3	3
mitochondrion	11	4	2
nucleolus	14	11	32
nucleus	102	70	95
plasma membrane/nuclear membrane	26	18	17
shmoo	5	4	2

Table 5: Summary of GO Protein Localization Annotation in HMS-PCI Data Set

Creating a Literature Validated Protein-Protein Interaction Benchmark

To systematically compile a set of published interactions as a benchmark, a search engine called PreBIND (http://bioinfo.mshri.on.ca/prebind/) was used. PreBIND is a support vector machine and natural language processing based algorithm designed to identify abstracts that describe protein-protein interactions. All abstracts involving *Saccharomyces cerevisiae* in PubMed were searched for interactions involving the 600 HMS-PCI expressed baits. Interactions found in this way were manually verified by reading the original abstract. The resulting data set contained 697 non-redundant interactions involving 574 proteins and was formatted for import into BIND. The MIPS table of protein-protein interactions from *Saccharomyces cerevisiae* was downloaded from http://mips.gsf.de/proj/yeast/tables/interaction/physical_interact.html on November 3rd, 2001 and formatted for import into BIND after removing interactions generated purely by high-throughput yeast two-hybrid (HTP-Y2H) methods (Uetz et al., 2000) (Ito et al., 2000) and interactions not involving HMS-PCI expressed baits (Fromont-Racine et

al., 1997; Ito et al., 2001; Mayes et al., 1999). The resulting filtered data set contained 545 unique interactions among 511 proteins. The PreBIND data was combined with these 545 interactions derived from the MIPS protein interaction table (Mewes et al., 2002) to create a literature-based set, "PreBIND+MIPS", of 747 proteins involved in 1,003 non-redundant interactions that involve the HMS-PCI bait set.

To address possible methodological bias in the literature benchmark, the MIPS data set was sorted into two-hybrid interactions (MIPS Two-Hybrid) and interactions based on biochemical purification such as immunoprecipitation, coimmunoprecipitation, purification, or copurification methods (MIPS Biochemical), according to experimental method annotation in MIPS. Again, these data sets did not include interactions from HTP-Y2H methods or interactions not involving HMS-PCI expressed baits. The MIPS Two-hybrid data set contained 282 interactions involving 323 proteins and the MIPS Biochemical data set contained 311 interactions involving 308 proteins. The MIPS data set contains fewer interactions than the sum of the MIPS Two-hybrid and MIPS Biochemical data sets because some interactions in MIPS were found using both two-hybrid and purification methods. As these two sets are of roughly equal size, the MIPS benchmark is impartial in this aspect.

A Statistical Method to Remove Noise from the HMS-PCI Data Set

As a consequence of both the gentle isolation methods used to recover protein complexes from concentrated extracts and the ultra-sensitive mass spectrometry used to identify proteins in each gel slice, non-specific contaminants in each complex purification were detected. These recurrent background species were filtered from the data set according to the following criteria: (i) any protein found in association with 3% or more of the baits assayed (ii) structural components of the ribosome, which were detected in many preparations (iii) all proteins that detectably bound to anti-FLAG resin in the absence of a FLAG-tagged bait protein (Gasteiger, 1996).

One distinct advantage of the HMS-PCI approach is that non-specific interactions are more readily identified as the size of the data set increases, as more information is

provided for determining the frequency based filter. An inherent difficulty with any data-filtering scheme is that proteins that participate in many *bona fide* interactions are at risk of being excluded from analysis. Proteins of note in this category included actin, tubulin, karyopherins, chaperonins and heat shock proteins, all of which are known to form numerous distinct and biologically relevant complexes, but were excluded because they appeared in association with too many bait proteins. Application of these filtering criteria reduced the data set to 3,618 distinct protein identifications in association with 493 baits. The filtered interaction set contains 1,578 different proteins or approximately 25% of the yeast proteome.

Potential non-specific interactions, excluded from the final data set, were based on the number of different baits an interactor protein bound (Ito et al., 2001). A 3% binding frequency exclusion was found to remove background interactions while retaining interactions that are meaningful, as defined by literature validation. In Figure 19, roughly 94% of the known interactions found in the PreBIND+MIPS literature validated benchmark are retained (166/177) when the 3% frequency exclusion is used to eliminate frequently binding proteins (Figure 19, dotted line). 3% frequency exclusion means that proteins that associate with more than 3% of the tested bait proteins are filtered away. The higher the frequency cutoff is set, the more of the 8,118 bait-associated proteins are kept. Typically, the excluded proteins are abundant proteins that are involved in metabolic processes, cell structure or biogenesis.

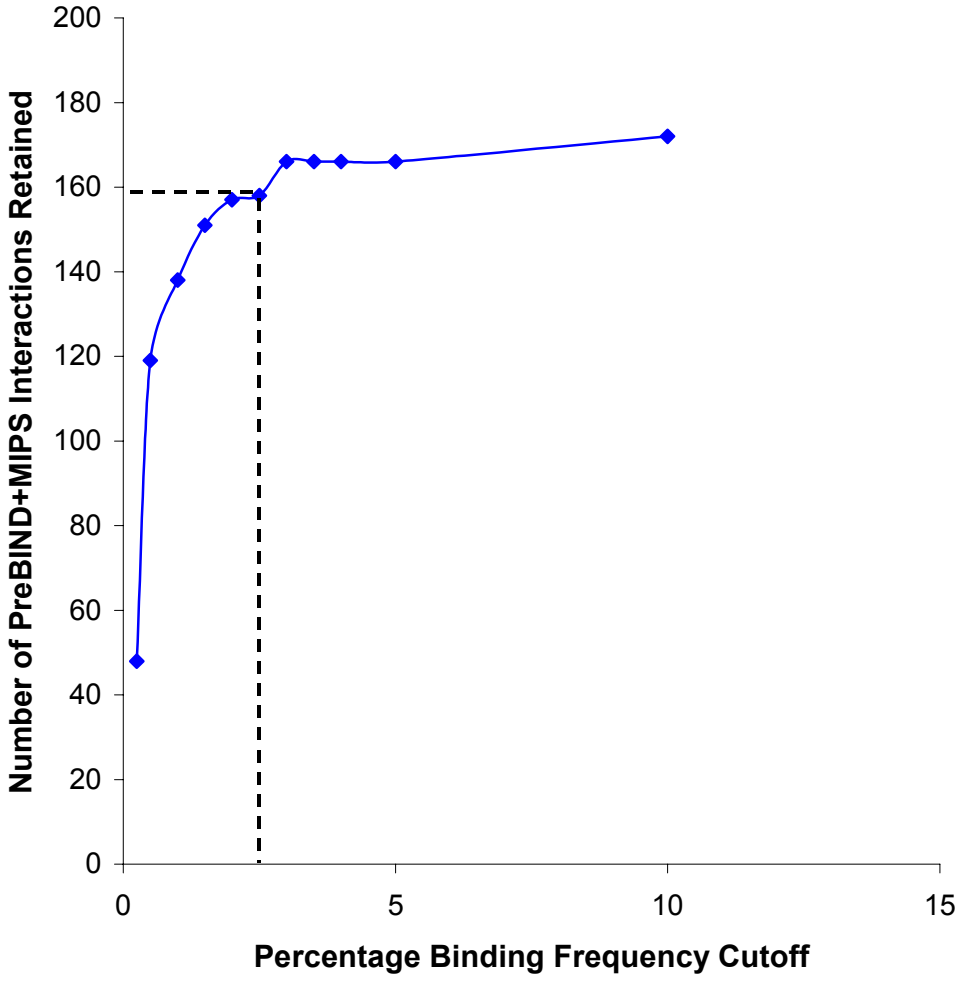


Figure 19: Graphical Analysis of Frequency Filter Cut-off

Graph of number of PreBIND+MIPS interactions retained in the data set as a function of interaction frequency exclusion. The dotted line indicates the number of interactions excluded from the data set when proteins binding 3% or more of the bait proteins were excluded.

Method Validation Based on Comparisons With Previous Large-Scale Data Sets

The HMS-PCI data set was compared to comprehensive HTP-Y2H data sets (Ito et al., 2001; Uetz et al., 2000) using interactions reported in the literature as a benchmark. It should be emphasized that in many cases the HMS-PCI detected interactions are bridged by intermediary partners, and could even be better considered a population of complexes of unknown topology. However, in the absence of additional evidence, there is no a priori means to elucidate connectivity of the interactions, and so each interaction is represented as a direct interaction between bait and associated proteins. Interaction data sets were entered into BIND as protein-protein interactions using the BIND Yeast import tool described above (BIND-IDs 11,509 to 12,408). When compared against the PreBIND+MIPS literature benchmark, described above, the HMS-PCI data set contained 2.6 to 3.4 fold more literature-derived interactions per bait than each large-scale HTP-Y2H data set and 1.9 fold more interactions when compared to the combination of both comprehensive HTP-Y2H data sets (Figure 20 a, b; Table 6) (Ito et al., 2001; Uetz et al., 2000). Interaction comparisons for overlap calculation purposes were treated as reflexive (i.e. A-B = B-A). To determine if the HMS-PCI data set is biased towards finding previously known interactions from a certain experimental technique, it was also compared to this method split benchmark (Table 6). No clear bias was revealed by this comparison, although yeast-two hybrid did match more previously known two-hybrid results compared to HMS-PCI and HMS-PCI matched more known biochemical purification results that the yeast two-hybrid data sets. In addition to published interactions, a number of novel interactions were shared by the HMS-PCI and HTP-Y2H data sets (Figure 20c).

Literature set / HTP data set	PreBIND	MIPS	PreBIND + MIPS	MIPS Two-Hybrid	MIPS Biochemical
HMS-PCI	113	119	166	55	81
Uetz	42	53	63	37	22
Ito-full	37	39	49	27	18
Ito-core	25	26	32	19	9
Ito-full + Uetz	60	71	86	47	37

Table 6: Literature-Derived Interactions Found in HMS-PCI and Large-Scale Two-Hybrid Interaction Data Sets

Figure 20: Comparison of Large-Scale Protein Interaction Networks to Interactions Reported in the Literature

A) Overlap of HMS-PCI data set and PreBIND+MIPS data set **B**) overlap of a comprehensive HTP-Y2H data set (Ito et al., 2001) and PreBIND+MIPS data set **C**) overlap of HMS-PCI and the HTP-Y2H data set (Ito et al., 2001). Blue edges are literature-derived interactions from PreBIND+MIPS; red edges are novel interactions detected by HTP approaches. For clarity, binary interactions are not shown: panel **A**), 37 interactions removed; panel **B**), 23 interactions removed panel **C**), 31 interactions removed. Visualization of protein interaction networks was performed using Pajek and were manually laid out. Pajek input network files were automatically generated from BIND by a custom program so that arrows pointing from bait protein to an experimentally determined associated protein and/or with previously known interactions from the PreBIND+MIPS set were highlighted. A poster-size Pajek visualization of the entire HMS-PCI network where each protein is colored by GO Biological Role is available as a vector-based PDF at:
http://www.nature.com/cgi-taf/DynaPage.taf?file=/nature/journal/v415/n6868/abs/415180a_fs.html.

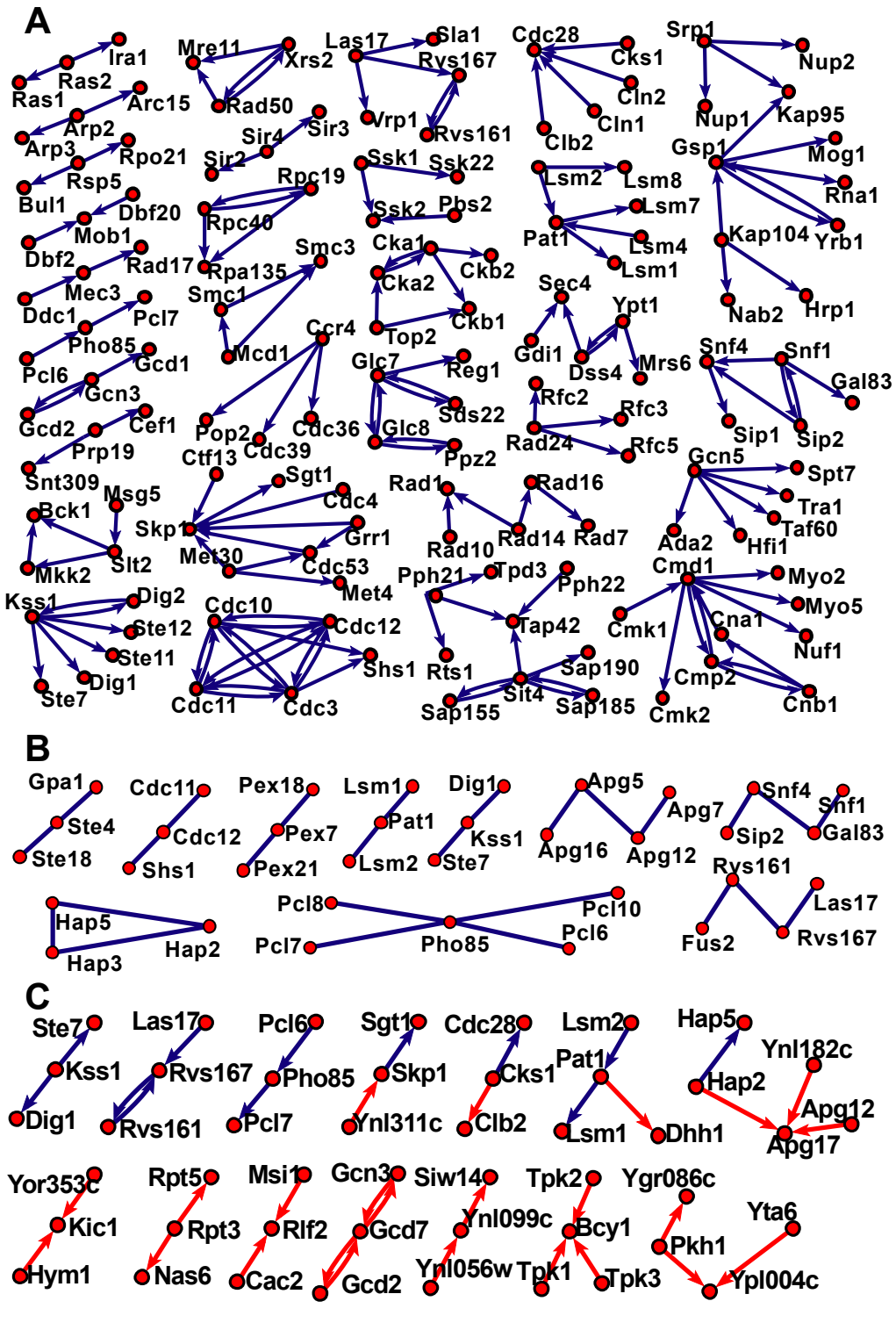

A
Ira1
Ras2
Ras1
Arc15
Arp2
Arp3
Rpo21
Rsp5
Bul1
Dbf20
Mob1
Dbf2
Rad17
Mec3
Ddc1
Pcl7
Pho85
Pcl6
Gcd1
Gcn3
Gcd2
Cef1
Prp19
Snt309
Msg5
Bck1
Slt2
Mkk2
Kss1
Dig2
Ste12
Ste11
Dig1
Ste7
Mre11
Xrs2
Rad50
Sir4
Sir3
Sir2
Rpc19
Rpc40
Rpa135
Smc3
Smc1
Mcd1
Ccr4
Pop2
Cdc36
Ctf13
Cdc39
Sgt1
Cdc4
Skp1
Grr1
Met30
Cdc53
Met4
Cdc10
Cdc12
Shs1
Cdc11
Cdc3
Las17
Sla1
Rvs167
Vrp1
Rvs161
Ssk1
Ssk22
Ssk2
Pbs2
Cka1
Ckb2
Cka2
Ckb1
Top2
Glc7
Reg1
Sds22
Glc8
Ppz2
Rad1
Rad16
Rad10
Rad14
Rad7
Pph21
Tpd3
Pph22
Tap42
Rts1
Sit4
Sap190
Sap185
Sap155
Cdc28
Cks1
Cln2
Cln1
Clb2
Lsm2
Lsm8
Lsm7
Pat1
Lsm4
Lsm1
Sec4
Ypt1
Gdi1
Dss4
Mrs6
Rfc2
Rfc3
Rad24
Rfc5
Srp1
Nup2
Nup1
Kap95
Gsp1
Mog1
Rna1
Kap104
Yrb1
Nab2
Hrp1
Snf4
Snf1
Gal83
Sip1
Sip2
Gcn5
Spt7
Tra1
Taf60
Ada2
Hfi1
Cmd1
Myo2
Myo5
Cna1
Nuf1
Cmp2
Cmk1
Cmk2
Cnb1
B
Gpa1
Ste4
Ste18
Cdc11
Cdc12
Shs1
Pex18
Pex7
Pex21
Lsm1
Pat1
Lsm2
Dig1
Kss1
Ste7
Apg5
Apg16
Apg7
Apg12
Snf4
Snf1
Sip2
Gal83
Rvs161
Las17
Rvs167
Fus2
Hap5
Hap2
Hap3
Pcl8
Pcl10
Pcl6
Pcl7
Pho85
C
Ste7
Kss1
Dig1
Las17
Rvs167
Rvs161
Pcl6
Pho85
Pcl7
Sgt1
Skp1
Ynl311c
Cdc28
Cks1
Clb2
Lsm2
Pat1
Lsm1
Dhh1
Hap5
Hap2
Ynl182c
Apg12
Apg17
Yor353c
Kic1
Hym1
Rpt5
Rpt3
Nas6
Msi1
Rlf2
Cac2
Gcn3
Gcd7
Gcd2
Siw14
Ynl099c
Ynl056w
Tpk2
Bcy1
Tpk1
Tpk3
Ygr086c
Pkh1
Yta6
Ypl004c

Figure 20

The HMS-PCI Data Connectivity Distribution Follows a Power Law

The connectivity distribution of the HMS-PCI data, hypothetically modeled as a network of direct bait protein-associated protein pairwise interactions, was calculated using the Pajek software package (Batagelj and Mrvar, 1998) by partitioning the network by node (protein) degree (k). The resulting partition was exported to Microsoft Excel where the graph of the probability that a node in the network interacts with k other nodes, $P(k)$, was plotted versus k. The resulting graph could be fitted using a power-law with an R^2 value of 0.90 (Figure 21). The power-law relationship was $P(k) = 1{,}042\ k^{-1.8}$. The fit of the connectivity distribution to this power-law is likely affected by the filtering criteria that were applied to the raw HMS-PCI data to remove background and from the fact that the hypothetical model does not take indirect interactions in the immunoprecipitated protein complexes into account. Metabolic (Jeong et al., 2000) and protein interaction (Jeong et al., 2001; Wagner and Fell, 2001) networks have been previously discovered to follow a power-law connectivity distribution (Barabasi and Albert, 1999). Such networks are robust and maintain their integrity when subjected to random disruption of components (Albert et al., 2000; Wagner, 2000).

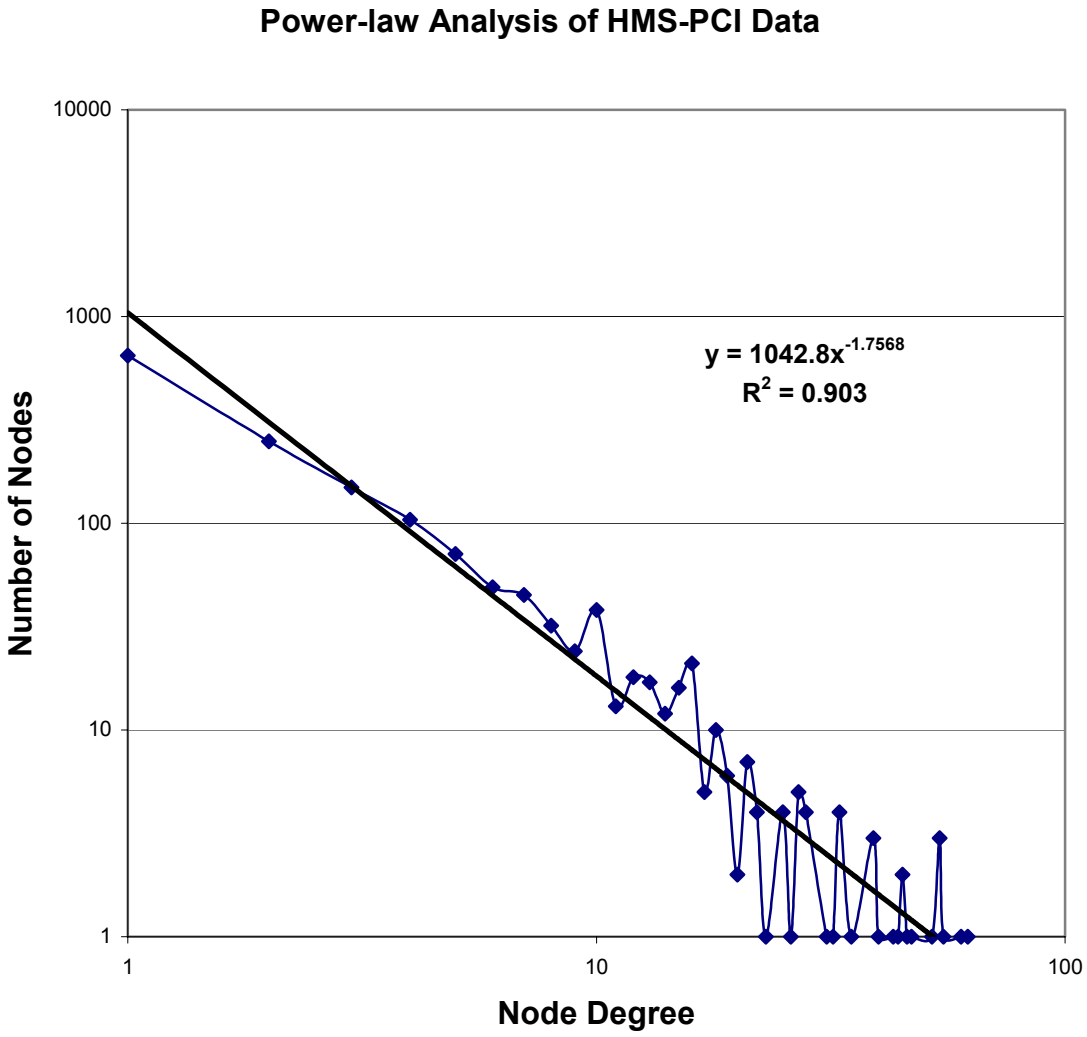


Figure 21: Power-Law Analysis of HMS-PCI Data

The number of nodes of each degree is shown. The Y and X axes are logarithmic scale. Note that there are many nodes of small degree and few nodes of large degree.

Conclusion

Proteome-wide analysis of native protein complexes by highly sensitive mass spectrometric methods allows the detection of complex cellular networks that might otherwise elude more focused approaches (See Chapter 6 – An Automated Method for Finding Molecular Complexes in Large Protein Interaction Networks). Given that approximately 40% of yeast proteins are conserved through eukaryotic evolution (Chervitz et al., 1998), the global yeast protein interaction map will provide a partial framework for understanding more complex proteomes. Imminent technical advances, such as gel-free analysis of protein complexes, higher sensitivity mass spectrometers, systematic analysis of post-translational modifications and protein microarrays will undoubtedly extend the reach of the approach described here (Mann et al., 2001; Zhu et al., 2001). As the set of proteins nominally encoded by the human genome is only 5-fold greater than the total number of yeast proteins, comprehensive analysis of the human proteome is feasible with current technology.

A Combined Experimental and Computational Strategy to Define Protein Interaction Networks for Peptide Recognition Modules

Introduction

Peptide recognition modules mediate many protein-protein interactions critical for the assembly of macromolecular complexes (Pawson and Scott, 1997). These modules bind to ligands containing a core structural motif; for example, SH3 and WW domains recognize proline-rich peptides, EH domains bind to peptides containing the NPF motif, and SH2 and PTB domains bind to peptides containing a phosphorylated tyrosine (Moran et al., 1990; Ren et al., 1993; Salcini et al., 1997). For particular modules within the same family, binding-partner specificity is determined by key residues flanking the core binding motif (Paoluzi et al., 1998). Although the complete genome sequence for an organism provides all of the potential peptide recognition modules and binding partners, a major challenge is to use these data to construct protein-protein interaction networks in which every module is linked to its physiologically relevant cognate partners. In collaboration with the lab of Dr. Charlie Boone at the Banting and Best Department of Biomedical Research at the University of Toronto, the lab of Dr. Gianni Cesareni at the University of Rome Tor Vergata and the lab of Dr. Stanley Fields at the University of Washington, a strategy was developed to combine computational prediction of interactions from phage-display ligand consensus sequences with large-scale two-hybrid physical interaction tests. Application to yeast SH3 domains generated a phage-display network containing 394 interactions among 206 proteins and a two-hybrid network containing 233 interactions among 145 proteins. Graph theoretic analysis identified 59 highly likely interactions common to both networks. Las17, a member of the Wiskott-Aldrich Syndrome protein (WASP) family of actin-assembly proteins, showed multiple SH3 interactions, many of which were confirmed *in vivo* by coimmunoprecipitation. (Tong et al., 2002)

Experimental Method

A four-step strategy is applied for the derivation of protein-protein interaction networks mediated by peptide recognition modules:

1) Screen random peptide libraries by phage display to define the consensus sequences for preferred ligands that bind to each peptide recognition module.
2) On the basis of these consensus sequences, computationally derive a protein-protein interaction network that links each peptide recognition module to proteins containing a preferred peptide ligand.
3) Experimentally derive a protein-protein interaction network by testing each peptide recognition module for association to each protein of the inferred proteome in the yeast two-hybrid system.
4) Determine the intersection of the predicted and experimental networks and test *in vivo* the biological relevance of key interactions within this set.

Because this strategy identifies ligands that bind directly to specific peptide recognition modules and defines interacting partners from the intersection of data sets derived independently, it is anticipated that the resultant network will be enriched for physiologically relevant interactions.

Results

This approach was applied to *Saccharomyces cerevisiae* SH3 domains as a test case. With the SH3 domain of the protein kinase Src as a query sequence for PSI-BLAST analysis (Altschul et al., 1997), 24 SH3 proteins were identified within the predicted *S. cerevisiae* proteome. Apart from Fus1, which controls cell fusion during mating, and Pex13, which participates in peroxisome biogenesis, most yeast SH3 proteins have been implicated in either signal transduction (Bem1, Boi1, Boi2, Cdc25, Sdc25, and Sho1) or reorganization of the cortical actin cytoskeleton (Abp1, Bud14, Cyk3, Hof1, Myo3, Myo5, Rvs167, and Sla1). A set of eight SH3 proteins (Bbc1, Bzz1, Nbp2, Yfr024c, Ygr136w, Yhl002w, Ypr154w, and Ysc84) remains to be characterized. Bem1

and Bzz1 contain 2 SH3 domains and Sla1 contains 3, with a total of 28 SH3 domains analyzed in this study.

Step 1: Phage display was used from Dr. Cesareni's lab to select SH3 domain ligands from a random amino acid nonapeptide library and screened all but four SH3 domains (Bem1-2, Cdc25, Sla1-1, and Sla1-2), which could not be expressed in a soluble form as glutathione-*S*-transferase (GST)-SH3 fusion proteins in *Escherichia coli*. After three selection cycles, positive clones were sequenced, and a consensus ligand was determined for 20 different SH3 domains (Figure 22). Four SH3 domains — Bud14, Sdc25, Cyk3, and Hof1 — did not select a ligand from the nonapeptide library, suggesting that they may not bind to a simple linear peptide with micromolar affinity. To further explore the subset of peptides containing the PxxP motif, a biased library (xxxxPxxPxxxx) was screened; however, the same SH3 domains failed to select a preferred ligand. In general, the ligand-binding surface of SH3 domains binds to a core PxxP ligand motif. Class I peptides conform to the consensus RxLPPZP (Z, hydrophobic residues or Arg) and bind in an orientation opposite to that of class II peptides, Px#PxR (Mayer, 2001). Most of the yeast SH3 domains selected proline-rich peptides that aligned with the typical Class I or Class II consensus sequence (Figure 22). Because of ancient chromosomal duplications, several SH3 proteins occur as pairs of paralogs (Myo3/Myo5, Yfr024c/Ysc84, and Ygr136w/Ypr154w). The SH3 domains of paralogs selected highly similar peptides, resulting in a similar consensus (Figure 22). A few SH3 domains selected peptides conforming to a highly unusual consensus. Bem1-1 SH3 domain selected peptides containing a PpxVxPY and Fus1 SH3 domain selected peptides with an RxxR(s/t)(s/t)Sl consensus.

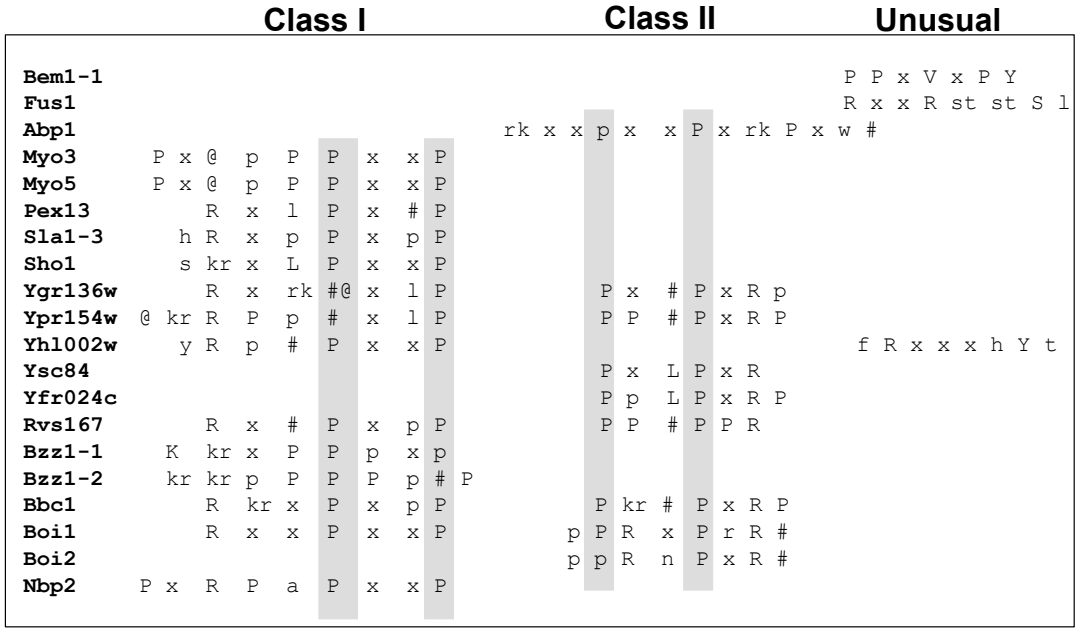


Figure 22: Consensus Sequence of Yeast SH3 Peptide Ligands

The consensus peptides were derived from an alignment of the selected phage-display peptides (x, any amino acid; lowercase letters, residues conserved in 50 to 80% of the selected peptides; uppercase letters, residues conserved in more than 80% of the selected peptides). Abbreviations for the amino acid residues are as follows: A, Ala; H, His; K, Lys; L, Leu; N, Asn; P, Pro; R, Arg; S, Ser; T, Thr; V, Val; W, Trp; Y, Tyr; #, hydrophobic residues; @, aromatic residues. The consensus sequences corresponding to Class I peptides, first column; Class II peptides, second column; unaligned, third column.

Step 2: The consensus sequences were used to search the yeast proteome for potential natural SH3 ligands in collaboration with the Cesareni and Boone labs. For 18 SH3 domains, a position-specific scoring matrix (PSSM) was compiled by calculating the frequency with which each amino acid was found at each position of the selected nonapeptides. The PSSM contained 9 columns (one for each peptide position) and 20 rows (one for each amino acid). To infer the ligands, a basic consensus pattern was first defined — for example, RxxPxxP or PxxPxR — for each SH3 domain, and then the PSSM was used to score all yeast peptides containing the consensus pattern. Peptides with the top 20% scores were considered potential ligands.

Because many of the yeast SH3 domain proteins have functionally connected roles in signal transduction and actin assembly, it was tested whether they could be represented as a network of interacting proteins (Schwikowski et al., 2000). The data were first imported into BIND (BIND-IDs 6,181 to 6,413), then formatted with BIND tools and exported for visualization in the Pajek package (Batagelj and Mrvar, 1998). The resulting protein-protein interaction map derived from the phage-display analysis (Figure 23) contains several known interactions, for example Sho1 SH3-Pbs2 (Maeda et al., 1995) and Rvs167 SH3-Abp1 (Lila and Drubin, 1997).

Abstracting the network as a graph permits analysis of the interactions with graph theoretical algorithms. Proteins are represented as nodes in the graph and interactions are represented as edges connecting the nodes. A subset of interconnected proteins in which each protein has at least k interactions (where k is a positive integer) forms a k-core. These cores represent proteins that are associated with one another by multiple interactions, as may occur in a molecular complex. The k-cores for the phage-display network were computed by using a core finding function in BIND and colored accordingly (Figure 23). The most highly connected core of the phage-display network was a single six-core subgraph, i.e., each protein in the subgraph has at least six interactions with other proteins in the subgraph (Figure 24). This core may represent a single complex; however, because the network does not take into account temporal expression or protein localization information, other interpretations are possible.

Figure 23: The Phage-Display Predicted SH3 Network

Yeast SH3 domain protein-protein interaction network predicted by means of phage display–selected peptides. In total, 394 interactions and 206 proteins are shown. The proteins are colored according to their *k*-core value (6-core, black; 5-core, cyan; 4-core, blue; 3-core, red; 2-core, green; 1-core, yellow), identifying subsets of interconnected proteins in which each protein has at least *k* interactions. Here, lower core numbers encompass all higher core numbers (e.g., a 4-core includes all the nodes in the 4-core, 5-core, and 6-core). The interactions of the 6-core subgraph are highlighted in red and are shown in more detail in Figure 24.

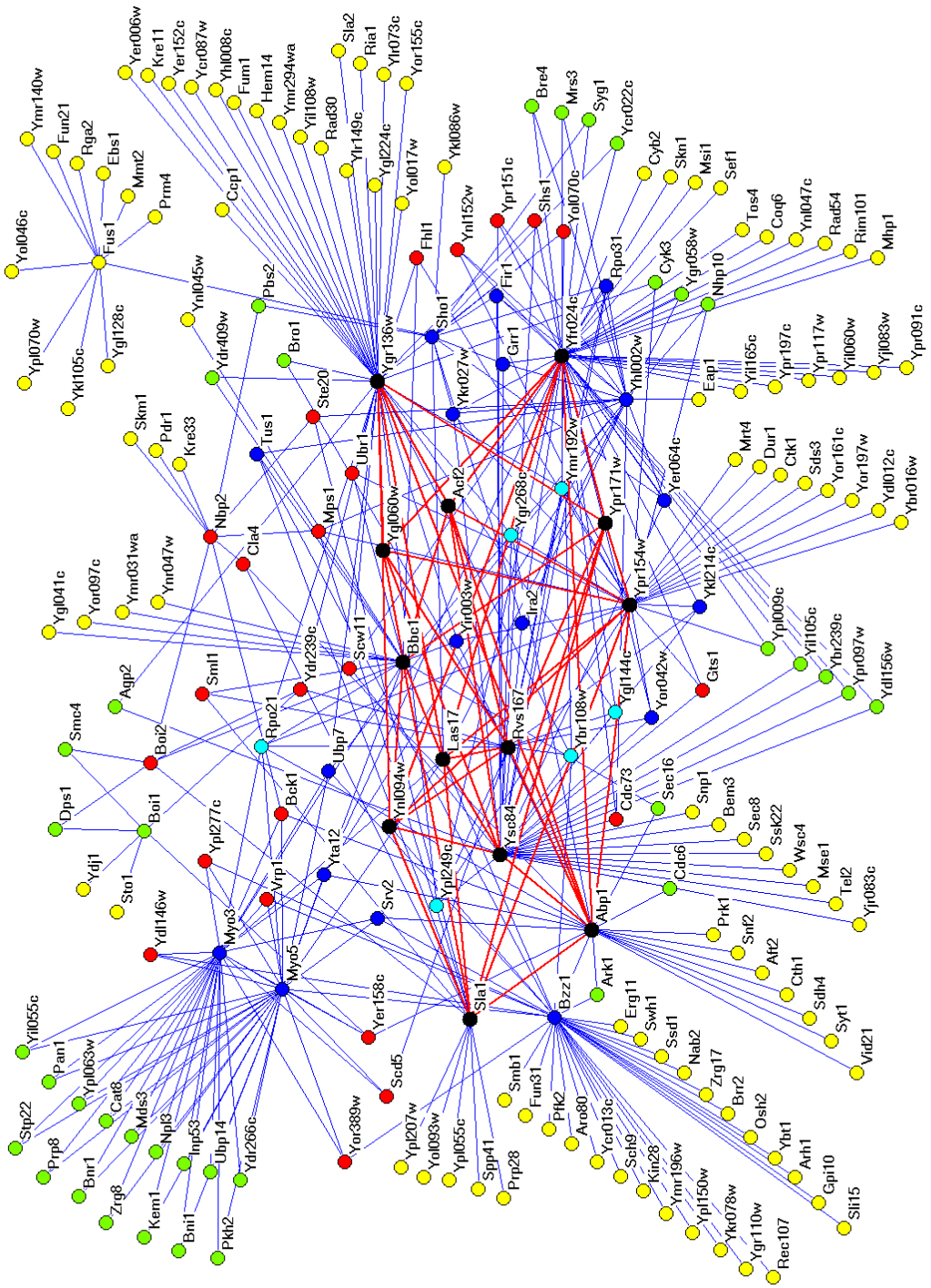

Yer006w
Kre11
Yer152c
Ycr087w
Yhl008c
Fum1
Hem14
Ymr294wa
Yil108w
Rad30
Ylr149c
Ygl224c
Yol017w
Ykl086w
Sla2
Ria1
Ylr073c
Yor155c
Ymr140w
Fun21
Rga2
Ebs1
Mmt2
Prm4
Yal046c
Fus1
Ypl070w
Ykl105c
Ygl128c
Ccp1
Ynl045w
Pbs2
Ydr409w
Bro1
Ste20
Ygr136w
Fhl1
Ynl152w
Ypr151c
Shs1
Yal070c
Bre4
Mrs3
Syg1
Ycr022c
Cyb2
Skn1
Msi1
Sef1
Tos4
Coq6
Ynl047c
Rad54
Rim101
Mhp1
Ypr091c
Yil083w
Ypr117w
Ypr197c
Yil060w
Yil165c
Eap1
Yfr024c
Yhl002w
Rpo31
Cyk3
Ygr058w
Nhp10
Fir1
Grr1
Ykr027w
Sho1
Ubr1
Ypr171w
Ymr192w
Yer064c
Mrt4
Dur1
Ctk1
Sds3
Yor161c
Yor197w
Ydl012c
Ybr016w
Ykl214c
Ypr154w
Ygr268c
Act2
Ygl060w
Mps1
Tus1
Skm1
Pdr1
Kre33
Nbp2
Cla4
Ynr047w
Ymr031wa
Yor097c
Ygl041c
Ydr239c
Scw11
Bbc1
Yir003w
Ira2
Ypl009c
Yil105c
Ybr239c
Ypr097w
Ydl156w
Gts1
Yor042w
Ygl144c
Ybr108w
Las17
Rvs167
Ubp7
Rpo21
Smi1
Agp2
Smc4
Boi2
Boi1
Dps1
Ydj1
Sto1
Ydl146w
Myo3
Myo5
Ypl277c
Bck1
Vrp1
Yta12
Srv2
Ynl094w
Ypl249c
Ysc84
Cdc73
Sec16
Snp1
Bem3
Sec8
Ssk22
Wsc4
Mse1
Tel2
Yjr083c
Cde6
Abp1
Prk1
Snf2
Aft2
Cth1
Sdh4
Syt1
Vid21
Ark1
Bzz1
Erg11
Swh1
Ssd1
Nab2
Zrg17
Bnr2
Osh2
Ybt1
Arh1
Gpi10
Sli15
Rec107
Ygr110w
Ykr078w
Ypl150w
Ymr196w
Kin28
Sch9
Ycr013c
Ara80
Pfk2
Fun31
Smb1
Spp41
Prp28
Ypl055c
Yal093w
Ypl207w
Yor389w
Scd5
Yer158c
Sla1
Yil055c
Pan1
Ypl063w
Cat8
Mds3
Npl3
Inp53
Ubp14
Ydr266c
Pkh2
Bni1
Kem1
Zrg8
Bnr1
Prp8
Stp22

Figure 23

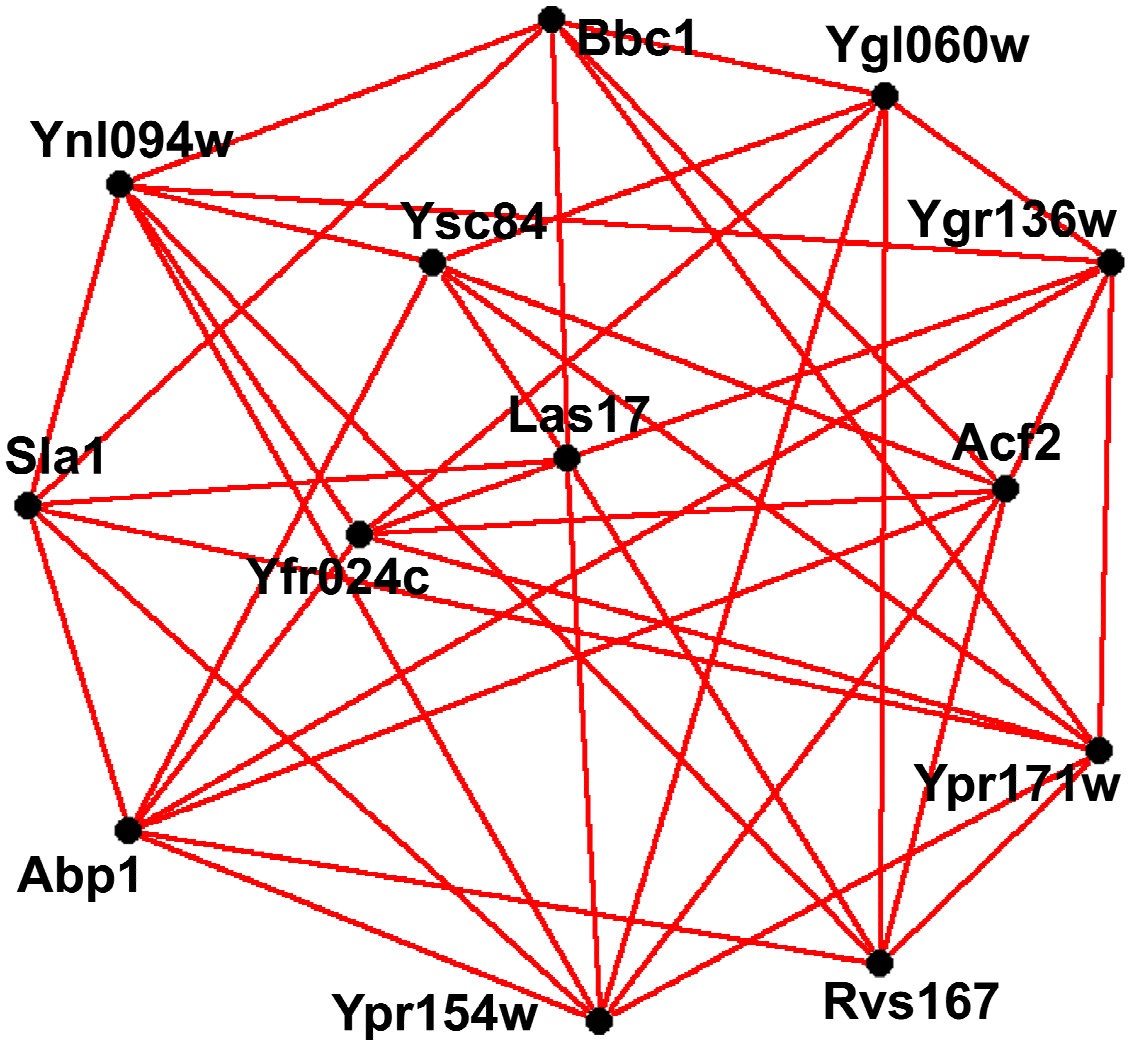


Figure 24: The Highest K-Core, a Six-Core, in the Phage-Display Predicted Protein Interaction Network

The 6-core subgraph derived from the phage-display protein-protein interaction network, expanded to allow identification of individual proteins. The 6-core subset contains eight SH3 domain proteins (Abp1, Bbc1, Rvs167, Sla1, Yfr024c, Ysc84, Ypr154w, and Ygr136w) and five proteins predicted to bind to at least six different SH3 domains (Las17, Acf2, Ypr171w, Ygl060w, and Ynl094w).

To assess the significance of this six-core, models were constructed of the phage-display network by randomly permuting its interactions. Modeling 1,000 different random networks resulted in an average core number of 4.01 (SD = 0.12); therefore, the observation of a highly connected six-core within the phage-display network was unlikely to occur by chance. Computer programs were written to automatically perform the random network modeling. A core finding algorithm is present in the Pajek package and can be applied to single networks. For convenience, a core finding function was written into the BIND API for use in the random network model programs. This algorithm takes as input a connected graph and proceeds by first removing nodes from the graph of degree less than k and then iteratively removing other nodes in the graph that are not connected by at least k edges to remaining nodes. A core finding algorithm finds a subset of highly connected nodes that are central to the network. The core analysis seems to work well with this particular data set because it is relatively small and focused on functionally related proteins. Other measures of connectivity, targeting specific regions of a network, are likely more informative for larger data sets (See Chapter 6 – An Automated Method for Finding Molecular Complexes in Large Protein Interaction Networks). The proteins within the six-core include several SH3 proteins — Abp1, Sla1, and Rvs167 — involved in cortical actin assembly; Las17, the yeast homolog of human Wiskott-Aldrich Syndrome protein (WASP), which binds to and activates the Arp2/3 actin nucleation complex (Colwill et al., 1999; Evangelista et al., 2000; Lechler and Li, 1997; Madania et al., 1999; Winter et al., 1999); Acf2, a protein required for Las17-dependent reconstitution of actin assembly *in vitro* (Lechler and Li, 1997); and several SH3 proteins of uncharacterized function: Bbc1, Yfr024c, Ypr154w, Ygr136w, and Ysc84.

Step 3: To derive a second protein-protein interaction network for comparison with the predicted phage-display network, a series of two-hybrid screens was conducted by the Fields lab (Uetz et al., 2000) with 18 different SH3 domain proteins as well as several proline-rich targets (Bbc1, Bni1, Las17, and Vrp1) as bait. Many of these proteins or protein domains were screened against both a genome-wide array of yeast Gal4 activation domain-open reading frame fusions and conventional two-hybrid libraries. In addition, the Boone lab directly assayed for two-hybrid interactions between

the SH3 domains and several proline-rich targets. Most of the resulting interactions (Figure 25) have not been reported previously. For example, only seven of the interactions within this network were identified by previous large-scale two-hybrid screens (Drees et al., 2001; Ito et al., 2001; Uetz et al., 2000), indicating that these screens were far from saturating and suggesting that thousands of two-hybrid interactions remain to be identified for the yeast proteome.

Figure 25: Two-Hybrid SH3 Domain Protein-Protein Interaction Network

Two-hybrid results, based largely on screens with SH3 domains as bait, generated a network containing 233 interactions and 145 proteins. Proteins are colored according to their *k*-core value (see Figure 23). The largest core of the two-hybrid network is a single 4-core (blue nodes). Interactions common to the phage-display network are highlighted in red.

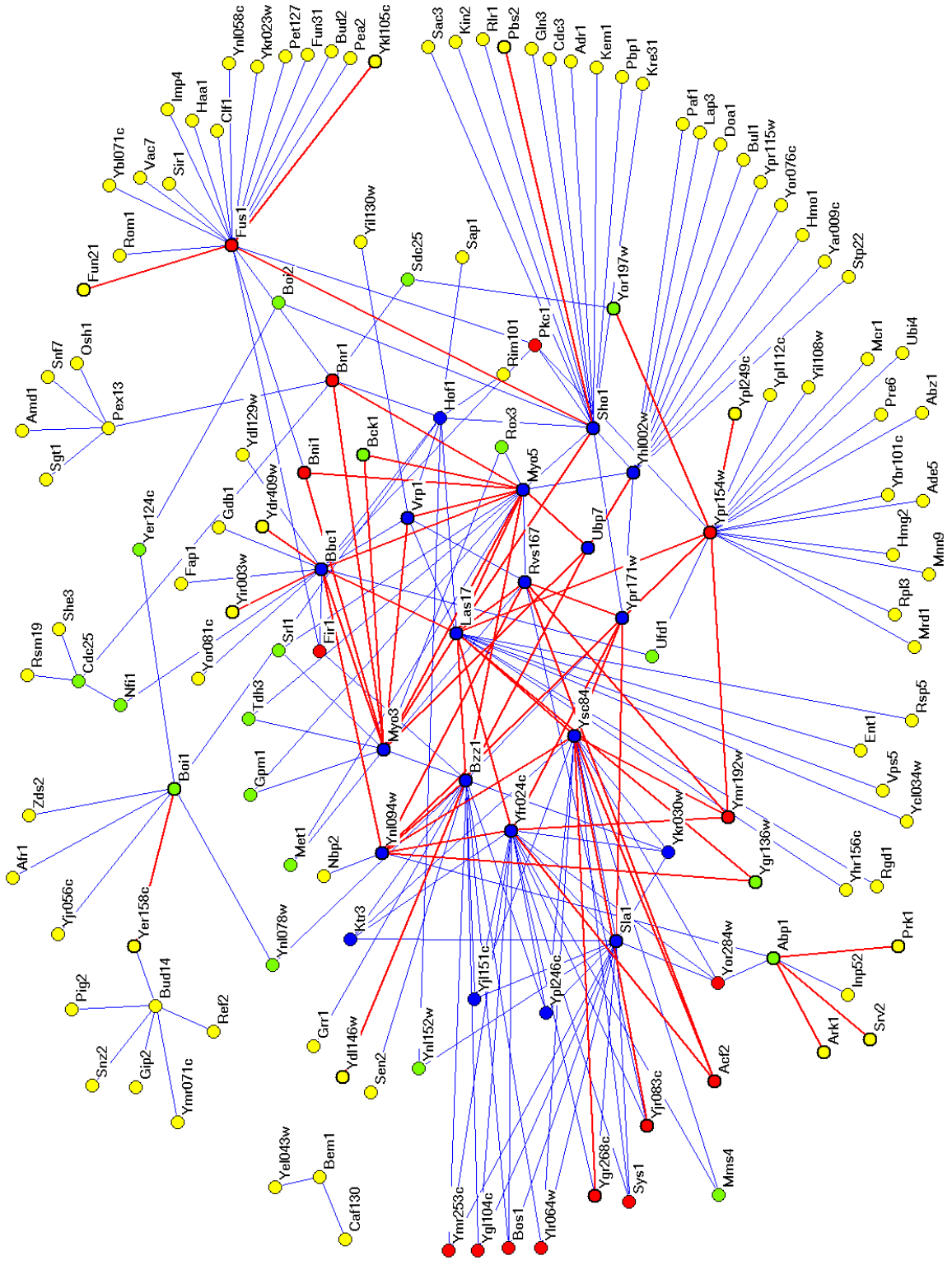

Ynl058c
Ykr023w
Pet127
Fun31
Bud2
Pea2
Ykl105c
Sac3
Kin2
Rlr1
Pbs2
Gln3
Cdc3
Adr1
Kem1
Pbp1
Kre31
Imp4
Haa1
Clf1
Ybl071c
Vac7
Sir1
Paf1
Lap3
Doa1
Bul1
Ypr115w
Yor076c
Hmo1
Yar009c
Stp22
Rom1
Fus1
Yil130w
Sdc25
Sap1
Fun21
Boi2
Yor197w
Pkc1
Rim101
Bnr1
Hof1
Sho1
Snf7
Osh1
Amd1
Pex13
Sgt1
Ydl129w
Bni1
Bck1
Rox3
Myo5
Vrp1
Ydr409w
Gdb1
Yer124c
Bbc1
Fap1
Yir003w
Yor081c
Rsm19
She3
Cdc25
Nfi1
Srl1
Fir1
Las17
Rvs167
Ubp7
Ypr171w
Ufd1
Ypl249c
Ypl112c
Yil108w
Mcr1
Ubi4
Pre6
Abz1
Ybr101c
Ade5
Hmg2
Mnn9
Rpl3
Mrd1
Ypr154w
Tdh3
Gpm1
Myo3
Bzz1
Yfr024c
Yse84
Ent1
Rsp5
Vps5
Ycl034w
Ymr192w
Ykr030w
Ygr136w
Yhr156c
Rgd1
Zds2
Afr1
Yjr056c
Met1
Nbp2
Ynl094w
Ynl078w
Yer158c
Bud14
Pig2
Snz2
Gip2
Ymr071c
Ref2
Ktr3
Sla1
Girl
Ydl146w
Sen2
Ynl152w
Yjl151c
Ypl246c
Yor284w
Abp1
Prk1
Inp52
Srv2
Ark1
Acf2
Yjr083c
Mms4
Yel043w
Bem1
Caf130
Ymr253c
Ygl104c
Bas1
Ylr064w
Ygr268c
Sys1

Figure 25

Step 4: The common elements of the phage-display and two-hybrid interaction networks were determined by finding the intersection of the data sets, where the elements of the data sets are binary protein-protein interactions and the interaction comparisons were considered reflexive (i.e., A-B = B-A). Only a subset of the interactions within the two networks is expected to overlap. In particular, the phage-display and two-hybrid analysis should identify different sets of false-positive interactions, excluding them from the overlap network. The phage-display analysis should identify a subset of the natural interactions mediated directly by short peptides, missing those that require either longer peptides or further stabilization by noncontiguous residues; moreover, some of the ligands predicted by phage display may not be surface-exposed within a folded protein. In the case of the two-hybrid interactions, there is a recognized potential for false-positives, in part due to over-expression and nuclear targeting of the fusion proteins, and because the screens of yeast proteins were conducted in yeast cells, the interactions may not be direct. Further, some of the SH3 domains screened by two-hybrid interactions were not included in the phage-display network and vice versa because some bait proteins yielded results only in one assay. In total, 59 interactions in the phage-display network were also found in the two-hybrid network (Figure 26). To determine the significance of this overlap, random phage-display networks were created by keeping the SH3-containing proteins and the number of interactions they participate in as a constant and randomly picking interacting partners from the yeast proteome. In 1,000 random networks with an average of 206 proteins (SD = 4.05), the average overlap was 0.84 interactions (SD = 1.01). Thus, the phage-display analysis was highly enriched for interactions common to the two-hybrid network. Further, the overlap network was enriched for literature-validated interactions, over threefold compared with the two-hybrid network and over fivefold compared with the phage-display network, suggesting that most of these SH3 domain interactions are likely to be physiologically relevant. YPD (Costanzo et al., 2001) contained 89 interactions that involved one of the SH3 domain proteins that were studied. Of these interactions, 17 were found in the phage-display network (394 interactions in total), 16 in the two-hybrid network (233 interactions in total), and 13 in the overlap network (59 interactions in total). Thus, the overlap network maintains most of the literature-validated interactions of the phage-

display and two-hybrid network.

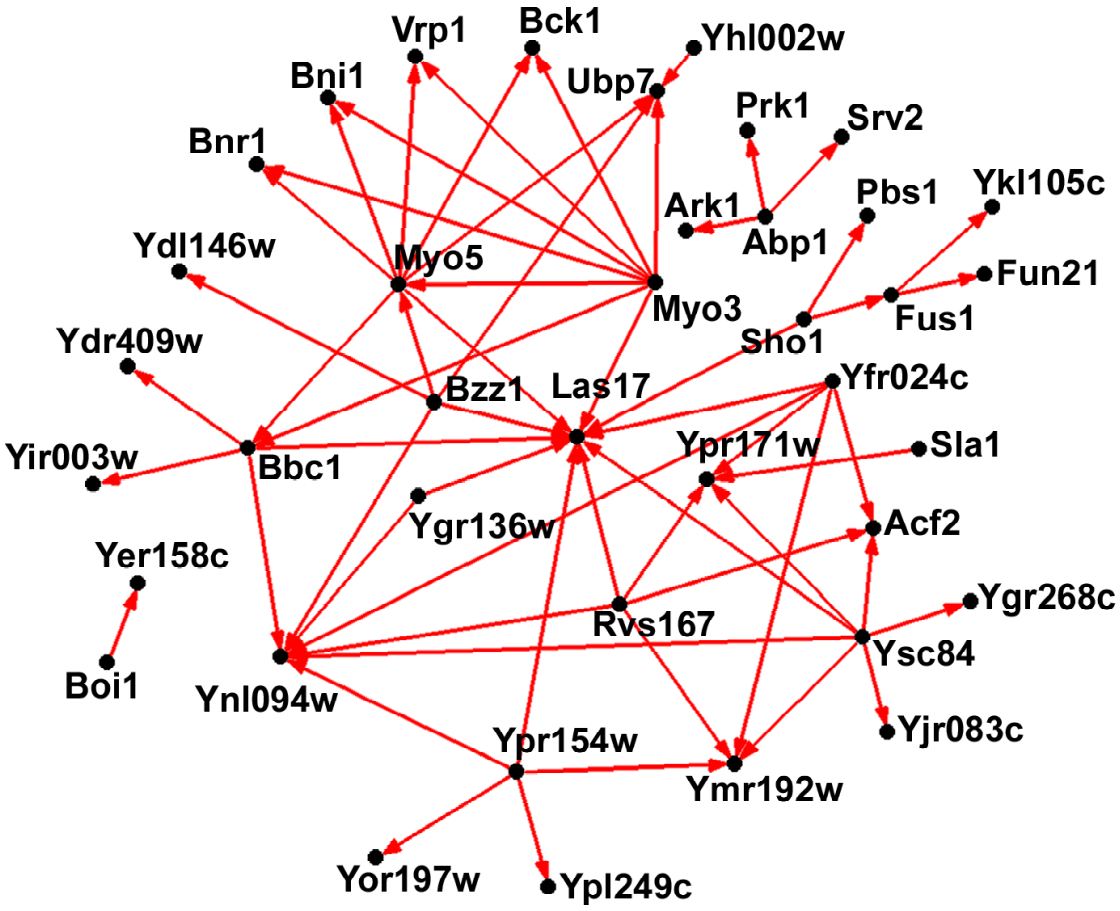


Figure 26: Overlap of the Protein-Protein Interaction Networks Derived from Phage-Display and Two-Hybrid Analysis

An expanded view of the common elements of the phage-display and two-hybrid protein-protein interaction networks, 59 interactions, and 39 proteins, is shown. All of these interactions are predicted to be mediated directly by SH3 domains. The arrows point from an SH3 domain protein to the target protein.

To examine the *in vivo* relevance of some of the interactions predicted by this strategy (Figure 26), the Boone lab focused on further analysis of the WASP homolog Las17, which localizes to cortical actin patches and interacts directly with several proteins involved in actin assembly. The network overlap predicts that the SH3 domains of 10 proteins may bind to a central proline-rich region of Las17, including three known binding partners Myo3, Myo5, and Rvs167 (Colwill et al., 1999; Evangelista et al., 2000; Winter et al., 1999); proteins identified previously by two-hybrid screens Yfr024c, Ygr136w, Ypr154w, and Ysc84 (Drees et al., 2001; Ito et al., 2001; Madania et al., 1999; Uetz et al., 2000); and previously unidentified partners Bbc1, Bzz1, and Sho1. This extensive set of interactions appears to be specific for Las17 because other actin-assembly proteins with proline-rich regions (Bni1, Bnr1, and Vrp1) were predicted to bind to only the SH3 domains of Myo3 and Myo5. The Las17 interactions appear to occur *in vivo*, because Myc epitope–tagged versions of six predicted binding partners co-immunoprecipitated with hemagglutinin (HA) epitope–tagged Las17 (Las17-HA) when expressed at normal amounts in yeast. In the case of the Bzz1-Las17 interaction, genetic and localization experiments by the Boone lab further confirmed its physiological relevance. Thus, at least nine different SH3 proteins associate with Las17 *in vivo*. Most of these proteins are highly conserved, suggesting that analogous complexes may occur for WASP-like proteins of higher eukaryotes.

The motifs derived from the phage-display experiments also predict the region of the target protein that binds the SH3 domain (Figure 27). To test this prediction, the Cesareni lab displayed five Las17 proline-rich peptide fragments as fusions to the D capsid protein on bacteriophage lambda and analyzed the binding of these fragments to a panel of SH3 domains in an enzyme-linked immunosorbent assay (ELISA). Apart from Myo3, whose best-predicted target, in the Las17-5 fragment, was not confirmed experimentally, the phage-display ligand algorithm consistently predicted the Las17 fragment that showed the strongest binding (Figure 27). These findings indicate that Las17 contains multiple binding sites of comparable affinity for several SH3 domains and suggest that Las17 may form one or more complexes containing multiple SH3 domain proteins.

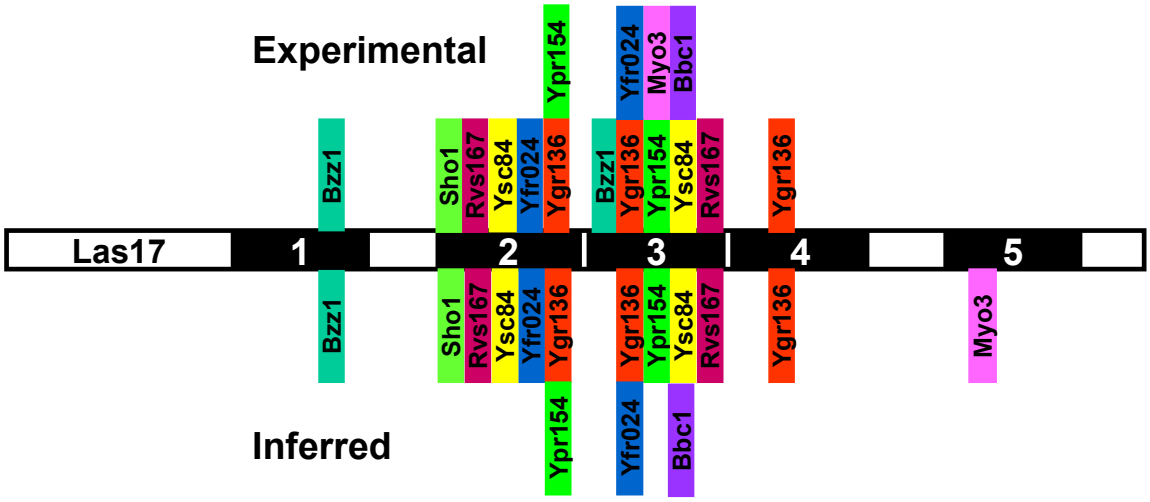


Figure 27: Schematic Representation of Potential Complexes Formed by SH3 Domain Interactions with Specific Proline-Rich Peptides of Las17

Five different proline-rich Las17 peptide fragments were displayed by fusion to the D capsid protein of bacteriophage lambda, and their reactivity with SH3 domains was tested by ELISA assay. The positive interactions observed in the ELISA experiments are shown in the upper part of the figure, whereas the interactions inferred by phage display are shown in the lower part. The fragment boundaries are Las17-1 (153-190), Las17-2 (306-336), Las17-3 (339-366), Las17-4 (374-403), and Las17-5 (423-476), respectively. For the Myo3/Myo5 paralog pair, only Myo3 was tested by ELISA assay.

Conclusion

The strategy described here has several features that make it particularly effective in the identification of relevant protein-protein interaction networks. First, both phage-display and two-hybrid analysis take full advantage of genomic information. Second, the two approaches are highly orthogonal in their respective strengths and weaknesses. Phage display uses *in vitro* binding and short synthetic peptides and predicts physical binding sites, whereas two-hybrid analysis uses *in vivo* binding and native proteins or protein domains, but may not produce direct interactions. Third, this method predicts precise binding sites. Fourth, the combined strategy is rapid and general. It can be implemented readily for other peptide recognition modules, apart from those that bind to ligands with cell type specific modifications, and other organisms with a sequenced genome.

Future Directions

Future experiments of this type may be able to achieve better results by optimizing certain steps. For example, some false positives undoubtedly arise when a predicted ligand peptide is buried in the core of the protein. To improve this aspect of the prediction, surface accessibility prediction, using PHDacc (Rost et al., 1994) for example, or from homology models (Pieper et al., 2002), could be performed to rule out buried pattern matches. To improve the proteome scanning stage used to predict the protein interaction network from the phage display data, a specificity and sensitivity analysis could be performed to assess what PSSM score threshold would keep the largest amount of physiologically relevant interactions (true positives) and disregard as many false positive interactions as possible. In this case, false positives can be defined operationally as those not identified within the literature or the yeast two-hybrid network. Thus, the optimization could be based on maximizing overlap with the yeast two-hybrid network or a set of confirmed interactions from a literature-based benchmark.

The overlap step could be improved in a number of ways. While the reasons for the false-positive and false-negative rate of yeast two-hybrid seem satisfyingly orthogonal to the phage display predicted network, other protein interaction experimental methods, such as co-immunoprecipitation coupled with mass spectrometry (Gavin et al., 2002; Ho et al., 2002), should be evaluated. The current network representation, with a single node representing a protein and a single edge representing an interaction, could be much improved by making it probabilistic. Attaching a probability value as a weight on the edges of the network could enter into the overlap calculation for a more realistic model. For instance, a weight value on an edge could be high if the interaction has been found by many different reliable methods, in multiple labs and published in high-impact journals. These highly probable edges could appear in the weighted combination of networks, which would include the 'textbook' interactions even if they were not included in all input networks. A review by Gerstein et al. (Gerstein et al., 2002) addresses some of these points in more detail. A better visualization tool that could draw networks with probabilistic information and allow one to examine parameter changes, for example in the PSSM score threshold discussed above, in real-time would compliment these method improvements and facilitate evaluation the results.

Many of the future improvements discussed so far depend on the availability of a literature-based benchmark, a manually curated collection of high-quality, expert validated interactions. Sources of more stringently validated interactions are MIPS (Mewes et al., 2002), YPD (Costanzo et al., 2001) and PreBIND (Ho et al., 2002). Collecting these together in a non-redundant set creates a benchmark of over 3,300 protein-protein interactions for yeast. Because some experimental methods are more likely to yield physiologically relevant information, for example if the interaction was detected for full length proteins expressed a native levels, the literature benchmark could also include a reliability score for each record.

A set of over 15,000 unique protein interactions collected for yeast from the literature and all available large-scale studies, contained 519 interactions involving 364 proteins where one interaction partner has an SH3 domain. Because many of these proteins are highly conserved, it will be interesting to observe how the connectivity of this network is organized in other organisms. The prospects for applying this interaction

network mapping approach in other organisms are reasonable as *Caenorhabditis elegans* only has 99 SH3 domains in 77 SH3 domain containing proteins according to SMART (Letunic et al., 2002), whereas the current mouse proteome has on the order of 327 SH3 domains in 172 proteins. Mapping protein binding module mediated interaction networks across organisms will provide a powerful data set to study the specificity of domain-mediated interactions, the evolution of complexity and the biology that these interactions dictate.

Chapter 5 – Integrated Experimental Protein Interaction Data Suggests a Large Nucleolar Complex in *Saccharomyces cerevisiae*

The majority of the work presented in this chapter has been published as follows (reprinted with permission, copyright Nature Journals):

Bader, G.D., Hogue, C.W.V.
Analyzing yeast protein-protein interaction data obtained from different sources.
Nature Biotechnology. 2002 Oct;20(10):991-7.

Abstract

Two recent high-throughput mass spectrometry (HT-MS) based protein interaction data sets are compared from budding yeast to each other and to other interaction data sets. The intersection of both HT-MS data sets reveals 198 interactions among 222 proteins, many of which reflect large multiprotein complexes. For interaction experiments that generate topology free networks, direct pairwise bait protein to associated protein "spoke" modeled interactions are roughly three times more accurate compared to the literature than an all proteins connected to all "matrix" model. The pool of all published protein interaction information from Saccharomyces cerevisiae is now 15,143 interactions among 4,825 yeast proteins, and power law scaling supports an estimate of 20,000 specific protein interactions in yeast. Notably, a large previously unsuspected nucleolar complex of 148 proteins, including 39 proteins of unknown function is identified. This complex consists of a network of subcomplexes, which appear to reflect the microscopic ultrastructural and functional organization of the nucleolus. The analysis suggests that existing large-scale protein interaction data sets are non-saturating and that integrating many different experimental data sets yields a clearer biological view than any single method alone.

Introduction

As proteomics technologies such as mass spectrometry and yeast two-hybrid become more sensitive and robust, they are becoming more automated and high-throughput. These experimental systems (Fields, 2001) are currently providing a wealth of data on gene function via molecular interactions and post-translational protein modifications. Protein-protein interactions mediate many aspects of cellular behaviour (Pawson et al., 2001) and are the basis for assemblies of molecular machines such as RNA polymerase II. Estimates of the number of protein interactions range from two to ten per protein (Marcotte et al., 1999). Thus, given the size of the Human proteome and taking into account splice variants of genes, cellular protein interactions will eventually

comprise more information than the Human Genome Project. Storing and analyzing this data represents a major bioinformatics challenge.

Two recent high-throughput analyses of protein complex composition in the budding yeast *Saccharomyces cerevisiae* by Gavin et al. (Gavin et al., 2002) and Ho et al. (Ho et al., 2002) have generated an unprecedented amount of protein interaction information. Both methods use tagged proteins as baits for high affinity capture of complexes whose protein components are subsequently identified using mass spectrometry (MS) (Pandey and Mann, 2000). Ho et al. use overexpressed bait proteins in a mild, single step purification protocol based on the FLAG epitope tag and ultra-sensitive LC-MS/MS for protein identification termed HMS-PCI (high-throughput mass spectrometric protein complex identification). Gavin et al. use a more stringent two-step purification based on the tandem-affinity purification (TAP) tag using native bait protein expression and less precise peptide mass fingerprinting by MALDI-TOF MS for identification.

There are clear advantages and disadvantages to each approach (von Mering et al., 2002). As each study detected interactions covering approximately 25% of the predicted yeast proteome, each represents a partial analysis of protein-protein interaction space. Together, the two HT-MS data sets provide functional information for 2,283 yeast proteins.

The Biomolecular Interaction Network Database (BIND) (Bader et al., 2001) and its associated bioinformatics infrastructure is used to compare and analyse current large-scale protein interaction data sets. BIND was designed to collect diverse experimental data on molecular interactions, complexes and pathways in a machine-readable format. The HMS-PCI and TAP data sets was first compared to each other and then a global analysis of all current electronically accessible knowledge of experimentally determined yeast protein interaction datasets was undertaken, including large-scale two hybrid screens (Drees et al., 2001; Fromont-Racine et al., 2000; Ito et al., 2001; Tong et al., 2002; Uetz et al., 2000). Gene Ontology (GO) (Dwight et al., 2002; The Gene Ontology Consortium, 2000) derived annotation for *S. cerevisiae* proteins was applied to examine functional connections in the genome-scale experiments. A recently described method

based on k-cores (Tong et al., 2002) to find and visualize molecular complexes is used to reveal new functional connections within the nucleolus.

Results

Modeling Biochemical Complexes as Binary Interactions

The purification processes used in the FLAG and TAP tag based experiments isolate complexes of proteins that are sufficiently self-assembled around the tagged bait protein to withstand the purification protocol. Not all proteins in any given complex will interact directly with the bait protein, because interactions may be bridged by other molecules in the mixture (e.g. RNA, proteins), or interact with the bait at the same time (e.g. if the bait protein is involved in multiple physiologically relevant complexes). Consequently, in a computational analysis, the bait and associated proteins must be considered a population of biomolecular complexes of unknown topology. While it is relatively straightforward to compare this information to known complexes in databases, most protein association information has been recorded as pairwise protein interactions resulting from experimental methods ranging from yeast two-hybrid screens to biochemical purification protocols, such as co-immunoprecipitation. To successfully compare multi-protein complexes to previously determined protein interaction data sets, two models that represent complexes of unknown topology as collections of hypothetical pairwise interactions are compared.

The first model, termed “spoke”, assumes that the bait interacts directly with each one of the proteins in the population of complexes, like spokes of a wheel, as shown here for a single purification.

Population of complexes: $C = \{b, c, d, e\}$ (b=bait)
Spoke model hypothetical interactions: $i_S = \{b\text{-}c, b\text{-}d, b\text{-}e\}$

This model excludes consideration of any homodimer formation or higher-ordered self-oligomerization of any protein in the set. The spoke representation yields

fewer interactions than may actually be present, and may misrepresent indirect interactions. Both Gavin et al. (Gavin et al., 2002) and Ho et al. (Ho et al., 2002) implicitly used the spoke model when determining criteria for filtering promiscuously binding proteins based on frequency of occurrence. Spoke model representation is useful to reduce complexity in data visualization.

The second model, termed "matrix", assumes that any two proteins within the population of complexes have a pairwise interaction as shown below:

Population of complexes; $C = \{b, c, d, e\}$
Matrix model hypothetical interactions; $i_M = \{b\text{-}b, b\text{-}c, b\text{-}d, b\text{-}e, c\text{-}c, c\text{-}d, c\text{-}e, d\text{-}d, d\text{-}e, e\text{-}e\}$

This model contains all possible true interactions within the experimental data but necessarily has a large number of false interactions as well, a problem that grows quadratically with the number of subunits in the complex. Further, matrix topologies are physically implausible for larger multiple sub-unit complexes because of probable steric clash. Both Gavin et al. and Ho et al. used a matrix model to determine their maximum data set overlap with previous large-scale yeast two-hybrid (LS-Y2H) data sets.

A recent analysis of large-scale protein interaction data sets (von Mering et al., 2002) used the matrix model to represent and compare HMS-PCI and TAP data and to derive measures of accuracy. The matrix model amplifies the effect of non-specific interacting proteins by connecting them to all other associated proteins in the complex. The functional distribution of interactions for the spoke modeled HMS-PCI and TAP data sets more closely resembles that for literature and LS-Y2H interactions than matrix modeled data does (Figure 28). The spoke modeled HMS-PCI and TAP data sets have similar interaction density patterns along the diagonal of the function interaction matrix, however, TAP has less inter-functional group interaction density (below the diagonal), possibly signifying less non-specific interactions between proteins in this set. While information is discarded in the spoke model, this may be an appropriate trade-off since spoke data is roughly three times more accurate compared to our literature benchmark than matrix representation (See Table 10, accuracy = size of literature benchmark overlap/size of data set). If a matrix representation is used, it may be useful to weight the

direct bait protein to associated protein (spoke) interactions with a higher significance score than other matrix interactions.

Caution is urged when interpreting these diagrams as assessments of interaction data set reliability, as many modular proteins have multiple annotations. A set of functional annotation terms can be chosen to maximize or minimize interaction density along the diagonal of the functional matrix graphs. Thus, while interesting, these graphs cannot provide a complete view of most large-scale data sets, and conclusions drawn from methodological comparisons will be questionable until the *Saccharomyces cerevisiae* proteome is fully mapped and annotated using multiple methods.

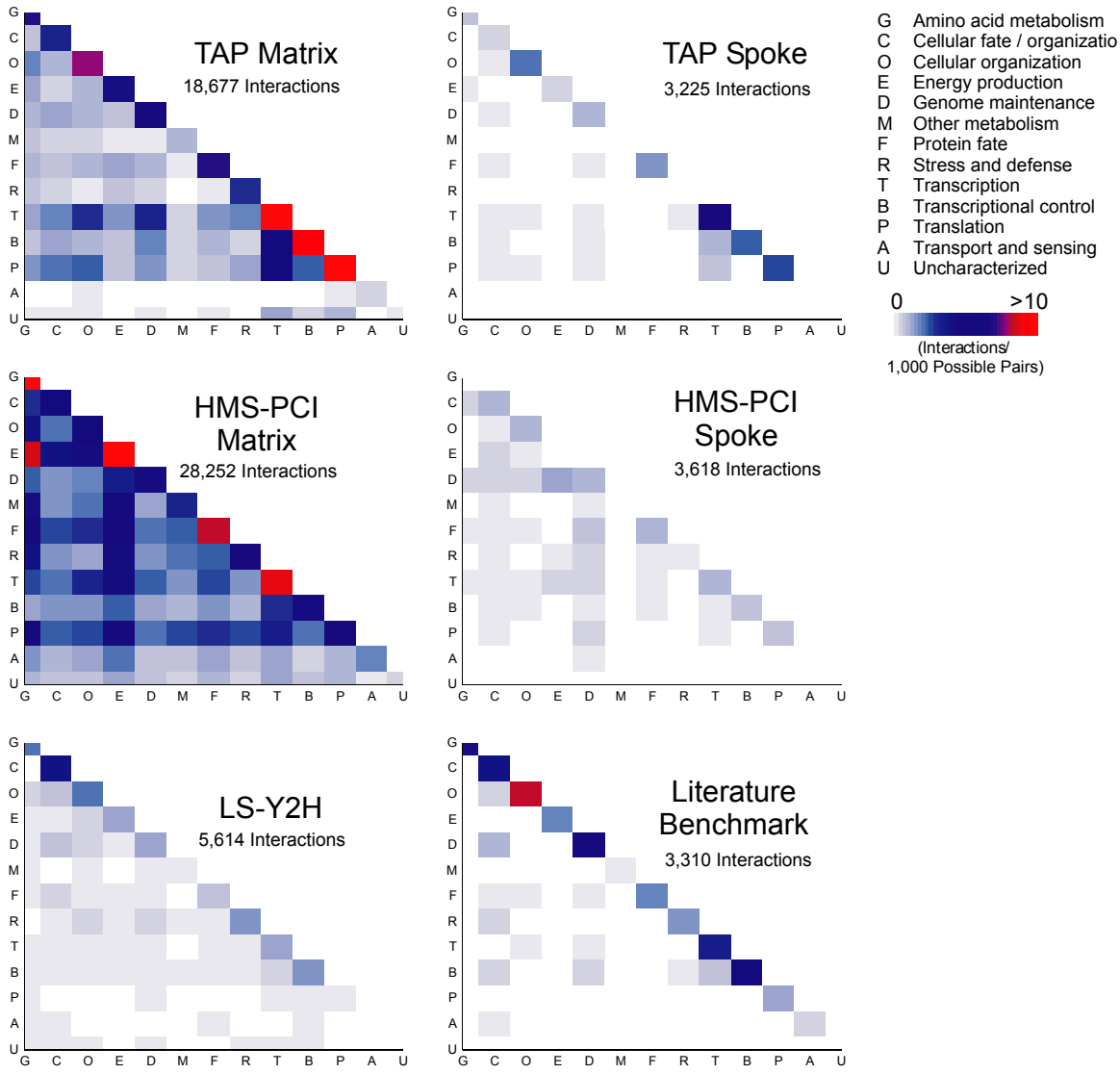


Figure 28: Functional Annotation Matrices Showing the Distribution of Interactions of Six Data Sets. Annotation is as per von Mering et al. (von Mering et al., 2002) to aid comparison. The HMS-PCI matrix interaction set is corrected compared to the von Mering version as it was derived from original immunoprecipitation (IP) data, whereas the published HMS-PCI data collapsed multiple IPs into one protein set. Definition of the functional annotation matrix is from (Ge et al., 2001).

Comparison of HMS-PCI and TAP Overall Data Sets

The overall networks of the two HT-MS data sets are remarkably different in connectivity despite being similar in size. The HMS-PCI data set appears much more interconnected whereas the TAP data set comprises more clusters of protein complexes which are sparsely connected (Figure 31). Increased regulatory network proteins may create a higher level of connectivity between well-known protein complexes. To assess whether the HMS-PCI and TAP data sets are different in this respect, a high-level ratio of regulatory to housekeeping protein GO annotation was computed. The regulatory category contains processes that include the word response (e.g. stress response), control (e.g. cell shape and cell size control) and cycle (e.g. cell cycle), processes (e.g. mating, budding) that are not involved in typical housekeeping roles and any process having to do mainly with protein level regulation and cell signaling (e.g. protein degradation, de/phosphorylation) (Table 7). For the yeast proteome, this ratio was 0.45, while for TAP it was 0.43 and for HMS-PCI it was 0.77. Thus there is a higher level of regulatory proteins in HMS-PCI than in the proteome and in the TAP data set. This may partially explain the higher level of connectivity in the HMS-PCI data set. However, there are still large fractions of unknown and unannotated proteins and what the true fraction is for any of these data sets cannot be determined.

GO Biological Process	Regulatory (r) or housekeeping (h)
cell adhesion	r
aging	r
autophagy	h
budding	r
cell cycle	r
cell growth and/or maintenance	h
chromosome organization and biogenesis	h
carbohydrate metabolism	h
cell organization and biogenesis	h
cell shape and cell size control	r
DNA damage response and repair	r
DNA metabolism	h
mating	r
general metabolism	h
mating-type determination	r
nucleolar and ribosome biogenesis	h
nutritional response pathway	r
protein biosynthesis	h
protein degradation	r
protein amino acid phosphorylation/dephosphorylation	r
protein metabolism and modification	h
protein transport	h
DNA recombination	h
DNA replication	h
RNA localization and processing	h
transcription	h
signal transduction	r
sporulation	r
stress response	r
transport	h
unknown	

Table 7: Definition of Regulatory and Housekeeping Biological Process Annotation Sets

Comparison of HMS-PCI and TAP HT-MS Common Baits

Of approximately 6,300 proteins ostensibly encoded by the yeast genome, Ho et al. selected 725 baits and Gavin et al. chose 1,739 baits; of these, 68% (493/725) and 26% (454/1,739) yielded detectably associated proteins (Table 8). These may be considered method efficiency ratios and may reflect differences in the bait expression systems selected. Only 115 baits were common to both studies and of these 81 were associated with identifiable proteins in both datasets. Seven common baits did not associate with any proteins in either experiment and 27 had partners in one method but not in the other. To evaluate the biological relevance of these two methods, the 115 common purifications from each method were compared to a literature benchmark consisting of 3,310 non high-throughput published interactions, which are presumed to be real, garnered from MIPS (Mewes et al., 2000), YPD (Costanzo et al., 2001) and PreBIND (Ho et al., 2002) encompassing 1,762 proteins. The PreBIND set encompasses the known PubMed literature concerning all HMS-PCI baits, thus can be considered comprehensive for the limited common bait subset. The TAP (628 interactions, 522 proteins) and HMS-PCI (875 interactions, 651 proteins) spoke model data sets from common baits contained 87 and 66 benchmark interactions involving 116 and 94 proteins, respectively, while TAP (4,916 interactions, 522 proteins) and HMS-PCI (7,618 interactions, 651 proteins) matrix model sets from common baits had 264 and 193 benchmark interactions involving 216 and 118 proteins, respectively. Thus, the TAP method is approximately 30% better at finding previously published interactions, at least for the limited intersection set. Interestingly, the HMS-PCI method finds 32% more unknown or unannotated proteins than TAP for the set of proteins associated with common baits.

	TAP - Gavin *et al.*	**HMS-PCI - Ho *et al.***
Overall Method	TAP/MALDI-TOF MS	FLAG/LC-MS/MS
Tag Size	>200 amino acids, C-terminal	8 amino acids, C-terminal
Bait Expression	Endogenous promoter	Ectopic; GAL1 or *tet* promoter
Cloning	Homologous Recombination, chromosome based	Gateway™ Recombination, vector based
Attempted Number of Genes Tagged	1,739	725
Culture volume	>2L	500ml
Affinity Isolation method	Tandem Affinity Purification	Immunoprecipitation
Unique Bait Proteins Detected as Expressed	1,167	600
Affinity Isolation attempts	588	1,558 (repeated baits)
Mass Spectrometry	MALDI-TOF MS	LC-MS/MS
Protein ID Method	Peptide Fragment Mass	MS/MS, Sequence
MS samples	20,946	15,683
Protein Ids	16,830	35,000
Baits with Hits	454	493
Filtered, Unique Proteins	1,363	1,578
Reproducibility	13 repeat attempts, 70%	64/86 immunoblots, 74%
Annotation	YPD	GO
Contaminant Frequency Cutoff	3.5% determined empirically	3.0% determined analytically
Total Proteins Excluded	66	434* control lanes + filtered at 3.0% + ribosomal set
Common Baits	115	115
Common Baits with hits	95	94

Table 8: Overall Comparison of TAP and HMS-PCI Methods

The asterisk indicated that non-redundant from GAL1 or *tet* methods combined.

Comparison of Common Hits

Given that each data set encompassed 25% of the yeast proteome, the two data sets show little overall overlap, despite approximately 70% internal reproducibility within each data set (Gavin et al., 2002; Ho et al., 2002). In part, this minimal overlap reflects bait selection by different functional criteria and differing expression systems effects. The intersection of the two data sets using the spoke data representation model contains only 198 associations among 222 proteins (Figure 29). This subset is probably the most reliable data in the two experimental sets, as it was independently found by both methods. The two largest common networks in the intersection are comprised of nucleolar proteins, including yeast orthologues of novel proteins recently detected in purified human nucleolar preparations (Andersen et al., 2002; Harnpicharnchai et al., 2001). One nucleolar network in the intersection set contains six essential proteins of unknown function: Ydr449c, Yjl069c, Yjl109c, Ygr090w, Ylr222c and Ylr409c (Figure 29). A number of other smaller complexes are observed, many with known function. These include components of the proteasome regulatory particle, polyadenylation and elongation factors, chromosomal segregation, mitotic exit complexes and proteins involved in mRNA splicing, vesicle trafficking, glucose repression and cytoskeleton rearrangement (Figure 29).

Figure 29: Overlap of the Spoke Models of TAP and HMS-PCI

There are 222 proteins and 310 arrows representing 198 protein associations. Arrows represent spoke interactions and point from bait to associated protein. Arrows are colored according to which study the interaction was found: Red, HMS-PCI; Blue, TAP; Cyan, Both HMS-PCI and TAP. Proteins are represented as nodes, are labeled with the common *Saccharomyces cerevisiae* gene name and are colored by GO derived cellular localization annotation: Yellow, Nucleolus; Red, Bud; Orange, Nucleus; Green, Membrane; Purple, Intracellular; Black, Unknown or unannotated.

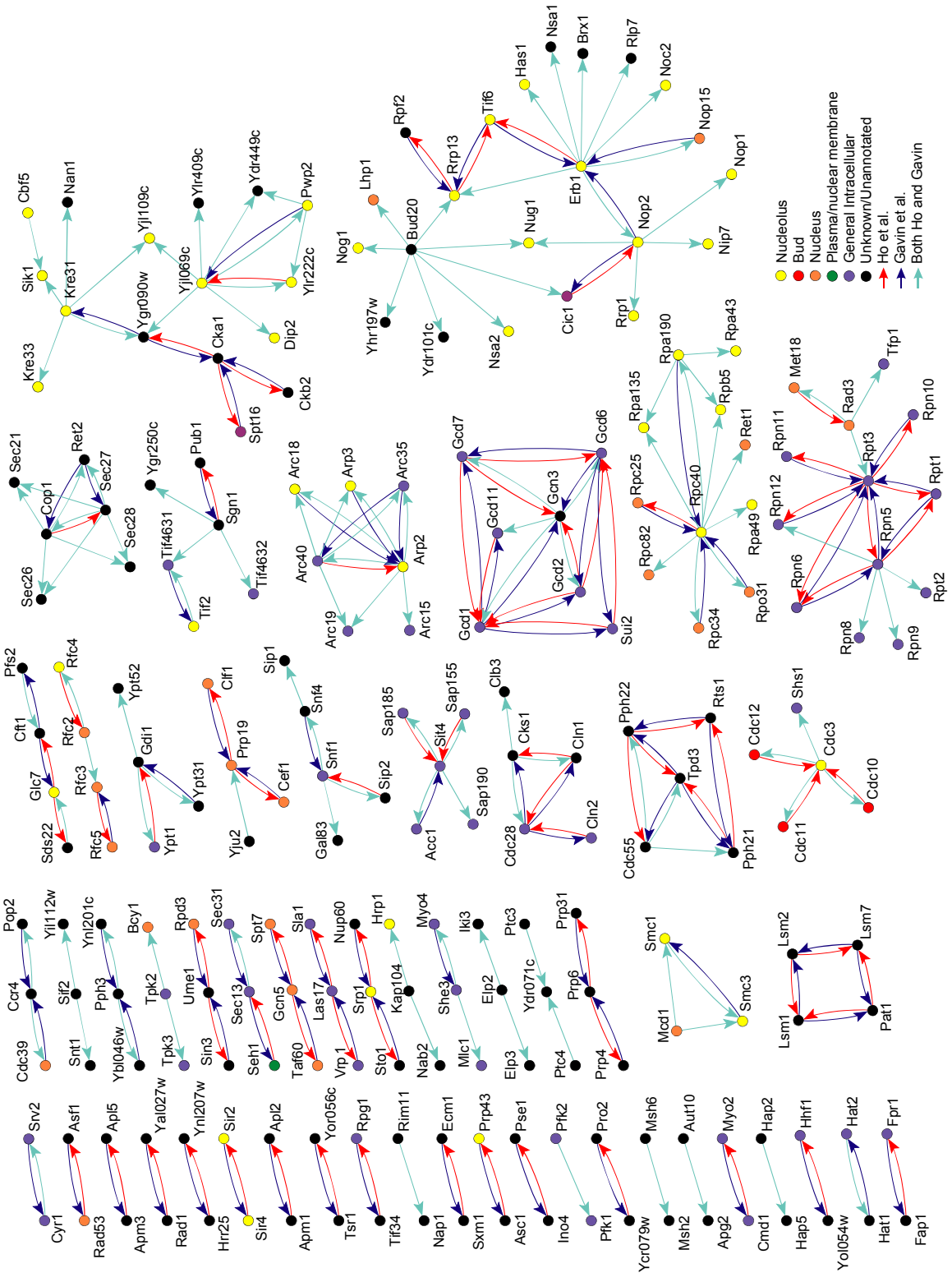

Nucleolus
Bud
Nucleus
Plasma/nuclear membrane
General Intracellular
Unknown/Unannotated
Ho et al.
Gavin et al.
Both Ho and Gavin
Cbf5
Nan1
Sik1
Kre31
Yjl109c
Yir409c
Ydr449c
Pwp2
Kre33
Ygr090w
Yjl069c
Ylr222c
Cka1
Dip2
Ckb2
Spt16
Nog1
Lhp1
Rpf2
Tif6
Has1
Nsa1
Brx1
Rlp7
Noc2
Nop15
Nop1
Bud20
Rrp13
Etb1
Nug1
Nop2
Nip7
Yhr197w
Ydr101c
Nsa2
Cic1
Rrp1
Sec21
Ret2
Cop1
Sec27
Sec28
Sec26
Ygr250c
Pub1
Sgn1
Tif4631
Tif4632
Tif2
Arc18
Arp3
Arc35
Arc40
Arp2
Arc19
Arc15
Gcd7
Gcd6
Gcn3
Gcd11
Gcd2
Gcd1
Sui2
Rpa190
Rpa43
Rpa135
Rpb5
Rpc25
Ret1
Rpc40
Rpc82
Rpa49
Rpc34
Rpo31
Met18
Tfp1
Rad3
Rpn10
Rpn11
Rpt3
Rpt1
Rpn12
Rpn5
Rpn6
Rpt2
Rpn8
Rpn9
Pfs2
Rfc4
Cft1
Rfc2
Glc7
Rfc3
Sds22
Rfc5
Ypt52
Gdi1
Ypt31
Ypt1
Clf1
Prp19
Cef1
Yju2
Sip1
Snt4
Snf1
Sip2
Gal83
Sap185
Sap155
Sit4
Sap190
Acc1
Clb3
Cks1
Cln1
Cln2
Cdc28
Pph22
Rts1
Tpd3
Cdc55
Pph21
Cdc12
Shs1
Cdc3
Cdc10
Cdc11
Pop2
Yil112w
Ccr4
Ynl201c
Sif2
Pph3
Cdc39
Snt1
Ybl046w
Bcy1
Tpk2
Tpk3
Rpd3
Ume1
Sin3
Sec31
Sec13
Seh1
Spt7
Gcn5
Taf60
Sla1
Las17
Nup60
Vrp1
Hrp1
Srp1
Sto1
Myo4
Kap104
She3
Nab2
Iki3
Elp2
Mic1
Ptc3
Ydr071c
Elp3
Prp31
Prp6
Ptc4
Prp4
Smc1
Smc3
Mcd1
Lsm2
Lsm7
Lsm1
Pat1
Srv2
Asf1
Apl5
Yal027w
Ynl207w
Sir2
Apl2
Yor056c
Rpg1
Rim11
Ecm1
Prp43
Pse1
Pfk2
Pro2
Msh6
Aut10
Myo2
Hap2
Hhf1
Hat2
Fpr1
Cyr1
Rad53
Apm3
Rad1
Hrr25
Sir4
Apm1
Tsr1
Tif34
Nap1
Sxm1
Asc1
Ino4
Pfk1
Ycr079w
Msh2
Apg2
Cmd1
Hap5
Yol054w
Hat1
Fap1

Figure 29

Functional Bias Exists in the Data Sets

Various subsets of the experimental results were examined to see if they were enriched in proteins of specific biological function (functional bias) according to yeast functional annotation terms derived from the Gene Ontology (GO) (Dwight et al., 2002; The Gene Ontology Consortium, 2000). In general, Ho et al. focused on regulatory pathways in cell cycle control, DNA damage response and repair, signal transduction and protein phosphorylation/ dephosphorylation. In contrast, the Gavin et al. expressed baits with associated proteins, were enriched in general metabolism, nucleolar and ribosome biogenesis, protein metabolism and transcription. The HMS-PCI data set had more membrane-localized proteins, but otherwise subcellular compartments were evenly represented in both bait selection sets.

No significant functional bias was found in baits that yielded associated proteins versus those that did not. However, by examining the set of all identified proteins (1,579 from HMS-PCI and 1,363 from TAP) as well as the set of only associated proteins (1,317 from HMS-PCI and 1,179 from TAP), it is evident that the functional bias mirrors the choice of baits, as might be expected from previous results showing that proteins of like function in yeast associate (Schwikowski et al., 2000). The only exception to this correlation is that metabolic proteins are over-represented in the HMS-PCI interaction set as compared to the bait set. This may reflect the propensity of the more sensitive LC-MS/MS method to detect low levels of non-specifically associated background proteins. It may be that contaminant frequency filter cut-offs need be adjusted after examining the comparison of these two data sets (Table 8). Interestingly, HMS-PCI and TAP data sets respectively contain 41% and 35% proteins of unknown and/or unannotated GO biological process. Thus, HT-MS methods may help to provide functional connections for the large unannotated portion of the yeast proteome.

Assuming that baits should generally pull down proteins of like function, it is expected that the distribution of function in the set of proteins associated with the 115 common baits will be similar in each experiment. Cell cycle and unknown proteins are heavily represented in the set of 115 common baits. In the set of proteins interacting with

the common baits, HMS-PCI contained more general metabolism, transport, signal transduction and proteins of unknown function while TAP comprised more DNA damage response and repair, nucleolar and ribosome biogenesis, transcription, RNA localization and processing as well as more nuclear and nucleolar localized proteins. Functional bias of the protein exclusion list does not explain this bias, thus it most likely relates to biological sample handling, such as cell disruption techniques.

Integration and Analysis of All Yeast Interaction Data

To assess the proteome coverage provided by all HT-MS and LS-Y2H studies to date, the spoke and matrix models of the HMS-PCI and TAP data sets were combined and compared to a compiled data set of interactions from multiple LS-Y2H experiments (Drees et al., 2001; Fromont-Racine et al., 2000; Ito et al., 2001; Tong et al., 2002; Uetz et al., 2000). 173 interactions were found between 265 proteins common to LS-Y2H (5,614 interactions, 3,652 proteins) and spoke MS (6,645 interactions, 2,283 proteins) and 304 interactions between 388 proteins common to LS-Y2H and matrix HT-MS (44,680 interactions, 2,283 proteins). All machine-readable data from various data sets (Costanzo et al., 2001; Drees et al., 2001; Fromont-Racine et al., 2000; Gavin et al., 2002; Ho et al., 2002; Ito et al., 2001; Mewes et al., 2000; Tong et al., 2002; Uetz et al., 2000) was collected and integrated to form a non-redundant set of 15,143 experimentally determined yeast protein interactions encompassing 4,825 proteins, or about 76% of the proteome (Table 9). The largest component of this integrated network contains 15,059 interactions among 4,689 proteins, leaving only 136 proteins not part of the main group. A full NxN comparison among selected large-scale individual data sets is shown in Table 10. The combined HT-MS matrix data set only overlaps 33% with the MIPS+PreBIND+YPD literature benchmark, leaving 67% of previously found protein interactions involving proteins in the combined HT-MS data set. It is concluded from this analysis that even with the advent of recent HT-MS studies, the detectable protein interaction space in the yeast system is far from saturated.

	Proteins	Interactions	Homodimers
Ho spoke	1,578	3,618	0
Ho matrix	1,578	28,252	1,578
Gavin spoke	1,363	3,225	0
Gavin matrix	1,363	18,677	1,363
Uetz	1,001	946	43
Ito full	3,274	4,468	82
Ito core	796	805	52
PreBIND	859	1,196	0
MIPS	964	1,353	51
YPD	1,538	2,205	283
MIPS+PB+YPD	1,762	3,310	303

Table 9: Properties of Large Yeast Interaction Data Sets

Proteins\ Interactions\ Homodimers	**MIPS+ PreBIND +YPD**	**YPD**	**MIPS**	**Pre BIND**	**Ito core**	**Ito full**	**Uetz**	**Gavin matrix**	**Gavin spoke**	**Ho matrix**
Ho spoke	265\ 210\ 0	230\ 168\ 0	161\ 119\ 0	169\ 113\ 0	71\ 41\ 0	109\ 64\ 0	88\ 55\ 0	333\ 366\ 0	222\ 198\ 0	1,578\ 3,618\ 0
Ho matrix	448\ 480\ 135	385\ 357\ 126	226\ 202\ 21	246\ 192\ 0	101\ 69\ 13	162\ 117\ 22	120\ 86\ 12	658\ 2,230\ 658	362\ 549\ 0	
Gavin spoke	361\ 333\ 0	276\ 198\ 0	249\ 230\ 0	163\ 117\ 0	71\ 40\ 0	97\ 55\ 0	78\ 47\ 0	1,363\ 3,225\ 0		
Gavin matrix	537\ 691\ 121	452\ 418\ 111	319\ 412\ 23	227\ 188\ 0	118\ 73\ 5	182\ 122\ 15	134\ 91\ 9			
Uetz	168\ 106\ 3	142\ 86\ 3	117\ 70\ 1	77\ 47\ 0	201\ 133\ 10	276\ 187\ 15				
Ito full	205\ 135\ 10	175\ 112\ 10	114\ 69\ 1	94\ 54\ 0	796\ 804\ 52					
Ito core	127\ 82\ 7	109\ 68\ 7	76\ 46\ 1	61\ 35\ 0						
PreBIND	859\ 1,196\ 0	579\ 554\ 0	442\ 402\ 0							
MIPS	964\ 1,353\ 51	803\ 834\ 31								
YPD	1,538\ 2,205\ 283									

Table 10: Large Yeast Interaction Data Set Cross Comparison

Size of each data set is given in Table 9.

The large integrated data set contains a higher percentage of proteins of unknown function and localization than the proteome. Of the approximately 1,500 predicted ORFs not identified by any protein interaction method, 75% are of unknown biological process and 80% have no localization GO annotation. These ORFs may be present in extremely low abundance in the cell or may only be expressed during specific developmental stages (e.g. spore formation).

As described previously by Barabasi *et al.* (Jeong et al., 2000; Jeong et al., 2001), the integrated network follows a power law node connectivity distribution. Within this distribution, essential proteins show a higher level of connectivity (10.7 average connections) than non-essential proteins (5.0 average connections), as shown in Figure 30. Furthermore, by scaling the power-law connectivity distribution of the integrated data set (4,825 proteins), defined above, to the yeast proteome (6,334 proteins (Chervitz et al., 1999; Pruitt and Maglott, 2001)), it can be estimated that there exist on the order of 20,000 protein interactions in yeast, a lower estimate than that provided by von Mering et al. (von Mering et al., 2002)

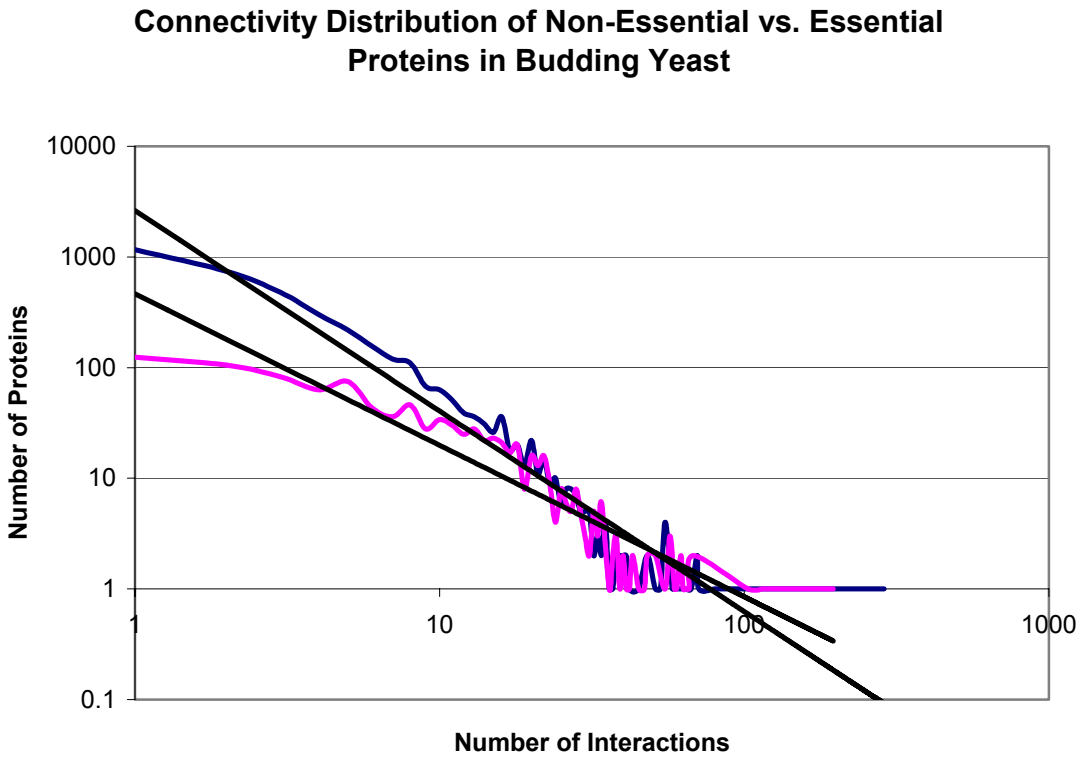


Figure 30: Comparing the Connectivity of Essential and Non-Essential Proteins

Blue diamonds represent non-essential proteins, pink squares represent essential proteins. The lines of best fit follow power laws and have R^2 values of 0.90 and 0.85, respectively. Essential proteins are generally more highly connected in protein interaction networks than non-essential proteins.

A Novel Nucleolar Network

Using a method of complex detection in interaction networks based on finding k-cores, as previously described (Tong et al., 2002), it was determined that both HT-MS data sets contain a dense, previously unsuspected, nucleolar network. A k-core of a network, or graph, is a subgraph where all proteins are connected to at least k other proteins in the subgraph, where k is 0,1,2,3... The k-core method was applied to the integrated yeast interaction network without HT-MS data (Figure 31A), to the HMS-PCI (Figure 31B) and TAP (Figure 31C) HT-MS data sets alone and to the fully the integrated network including all HT-MS data (Figure 31D). The nucleolar network emerges as the data set size is increased. Notably, only a few nucleolar proteins are present in the highly connected regions of the network before HT-MS data inclusion (Figure 31A). In contrast, both the individual HMS-PCI and TAP data sets contain highly connected networks involving nucleolar proteins. Many of the proteins in the nucleolar network are orthologues of human proteins recently found in highly purified human nucleoli (Andersen et al., 2002; Harnpicharnchai et al., 2001).

Interestingly, three of the sub-complexes that are visually apparent in Figure 31D correspond to the known substructure of the nucleolus as determined by electron microscopy (Olson et al., 2000). The fibrillar component (FC) involved in pre-rRNA transcription corresponds to a sub-complex of proteins with likely transcriptional functions, labeled SAGA (Figure 31D). All 14 known components of the SAGA complex are visible in Figure 31D, although two other proteins are also highly connected to SAGA, Taf145 and Spt15. Taf145 and Spt15 are known to participate in the RNA polymerase II general transcription factor complex with other SAGA components. The dense fibrillar component (DFC) is the site of rRNA processing and corresponds to the complex of proteins labeled “rRNA splicing/modification”. Known nucleolar links with snRNA associated proteins are visible in the many links between the nucleolar complex and RNA modification complexes (e.g. U4/U6 snRNP, U4/U6.U5 tri-snRNP complex, U2 snRNP and U1 snRNP complexes). All 9 known components of polyadenylation factor I (PFI) are clustered in Figure 31D along with Rna14 and Ref2, known to be associated with PFI, and Pti1, a protein of unknown function that seems to be a

previously unknown component of PFI. The granular component (GC) involved in assembling pre-ribosomal proteins, corresponding to the protein cluster labeled "nucleolus". Consistent with recent findings of nucleolar functional links to cell cycle control (Visintin and Amon, 2000), the anaphase promoting complex (APC) is seen connecting to the nucleolus, SAGA, and the proteasome (Cdc23 interacts with Spt2, Ada2 and Rpt1, Cdc16 interacts with Mus81 and Rpt1). All 11 known components of APC are visible in Figure 31D. Of the 18 known 19S proteasome regulatory particle (PRP) components, the 9-core in Figure 31D misses Rpn1, Rpn2, Rpn4 and Rpn7. These are connected to the 19S PRP in the underlying data set, but not by 9 interactions and so do not appear in the 9-core. Interestingly, Ecm29, Hsm3, Rad23, Ubp6 and Ygl004c appear highly connected with the 19S PRP. Ubp6 and Rad23 are known to be associated with elements of the proteasome, but Ecm29, Hsm3, and Ygl004c, a WD40 repeat containing protein, are not although their high connectivity suggests that they may be components of PRP. While Jsn1 is not known to be part of any complex, it has been shown to interact with over 160 proteins almost exclusively in HT-Y2H screens. Jsn1 has been shown to bind to SAGA, APC, protein components of the proteasome, nucleolus and the region on the Figure 31D labeled rRNA splicing/modification, although these interactions may be mediated by at least one RNA bridging molecule, since Jsn1 has been predicted to bind RNA. Thus, as illustrated by identification of a large nucleolar complex, sufficient non-directed coverage of protein interactions can reveal large-scale functional domains, without a priori knowledge of the functional annotation in the integrated data set.

Figure 31: Visual Representation of Molecular Complexes in Protein Interaction Networks Found Using the K-Core Method

While there are higher k-cores in these sets, a k-core level was chosen that represents as many nucleolar annotated proteins as possible without becoming too large. **A**) 6-core of the integrated yeast protein interaction network before addition of HT-MS data. **B**) 6-core of the HMS-PCI data set. **C**) 6-core of the TAP data set. **D**) 9-core of the integrated yeast data set after addition of HT-MS data. The complex connectivity surrounding the nucleolus is clearer and more complete in the fully integrated data set in panel D indicating that data integration is necessary for better understanding of a biological system. APC - anaphase-promoting complex, SAGA - SAGA (Spt-Ada-Gcn5-acetyltransferase) transcriptional activator-histone acetyltransferase complex, DDR - DNA Damage Response, TRAPP - Transport Protein Particle complex. 19S regulatory subunit of the proteasome is labeled 'proteasome'. Proteins are colored according to GO cellular component although nucleolar localized annotation was supplemented with yeast orthologues of human proteins recently found to be in the human nucleolus (Andersen et al., 2002). In 1,000 randomly permuted networks from A, B, C, D, the mean highest k-core was 5 (SD=0), 5.85 (SD=0.36), 5 (SD=0) and 7 (SD=0), respectively. Thus, the high k-core numbers in A, C and D are highly unlikely to occur by chance.

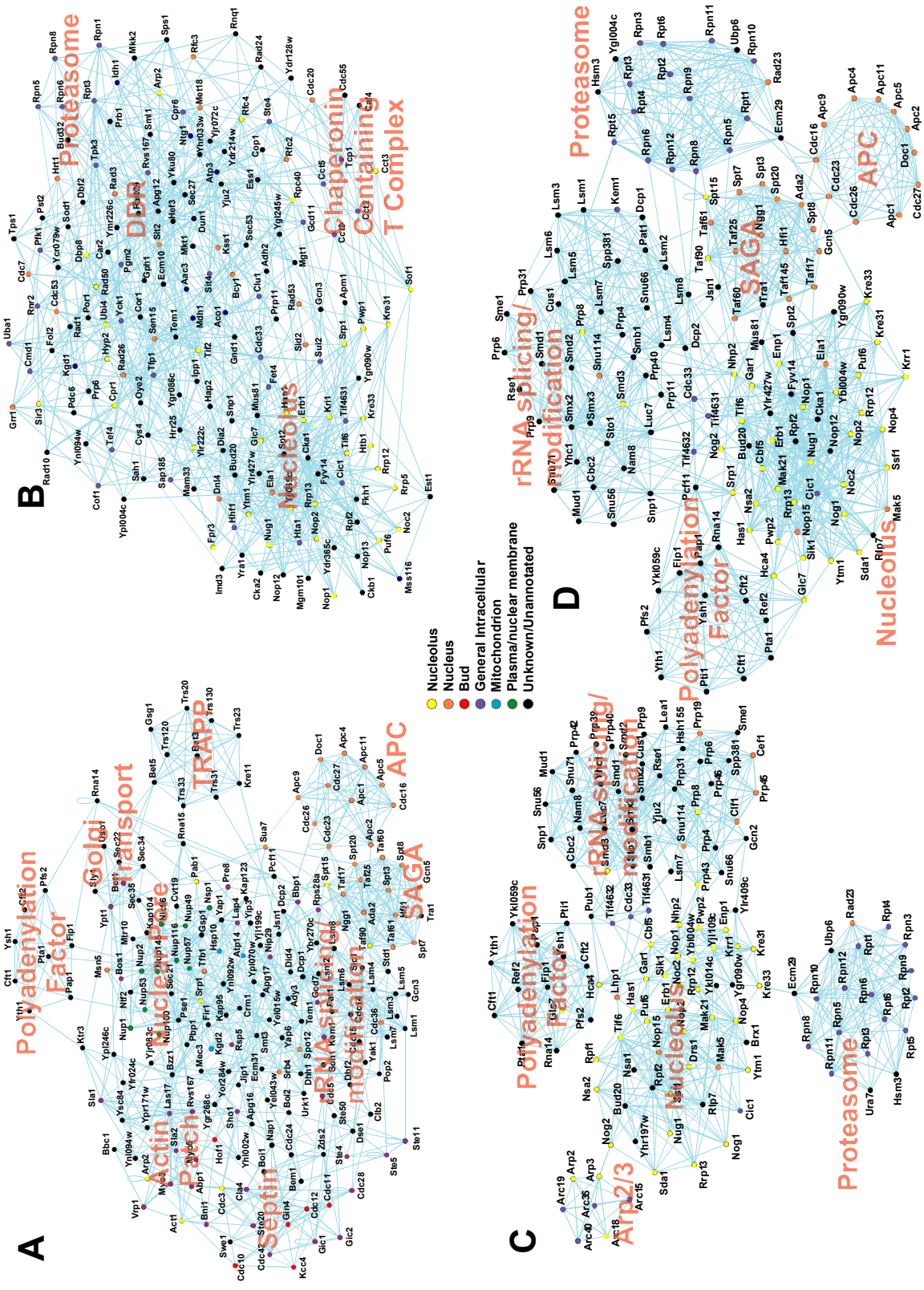

A
Polyadenylation Factor
Golgi Transport
Actin Patch
Nuclear pore
TRAPP
APC
SAGA
Septin
rRNA modification
B
Proteasome
DNA Repair
Chaperonin Containing T Complex
Mediator
C
Polyadenylation Factor
rRNA modification
Arp2/3
Nucleosome
Proteasome
D
Proteasome
rRNA splicing/ modification
SAGA
APC
Polyadenylation Factor
Nucleolus
Nucleolus
Nucleus
Bud
General Intracellular
Mitochondrion
Plasma/nuclear membrane
Unknown/Unannotated

Figure 31

Conclusion

Large-scale experiments have the potential to discover previously unknown functional connections among components of the cell, and hence promise to rapidly expand our knowledge of biology. However, data quality plays an extremely important role in this knowledge expansion. Large-scale techniques so far do not show enough internal consistency to warrant complete acceptance of the resulting data. This indicates that each screen will have to be performed multiple times before achieving a high enough data quality for a particular method. While it is relatively straightforward to systematically identify stable multi-protein complexes, or "cellular machines", detecting transient regulatory interactions, often involved in signaling pathways, metabolons and hyperstructures (Norris et al., 1999) is still difficult. Considering these factors, it is important to concurrently develop computational systems, such as BIND (Bader et al., 2001; Xenarios et al., 2002), that can both integrate, visualize and mine available molecular interaction data sets in order to speed the emergence of a clear view of protein complexes and associated regulatory interactions.

Experimental Protocol

Data Sources

All protein interaction data sets from MIPS (Mewes et al., 2000), Gene Ontology and PreBIND (http://bioinfo.mshri.on.ca/prebind/) were collected as described previously (Ho et al., 2002). The YPD protein interaction data are from March 2001 and can be requested from Proteome, Inc. (http://www.proteome.com). Other interaction data sets are from BIND (http://www.bind.ca). Yeast homologues of human nucleolar proteins are from the supplementary material made available by Andersen *et al.* (Andersen et al., 2002) A BIND yeast import utility was developed to integrate data from SGD (Chervitz et al., 1999), RefSeq (Pruitt and Maglott, 2001), Gene Registry (http://genome-www.stanford.edu/Saccharomyces/registry.html), the list of essential genes from the yeast deletion consortium (Winzeler et al., 1999) and GO terms (Dwight et al., 2002; The

Gene Ontology Consortium, 2000). This database ensures proper yeast gene name matching among the multiple data sets that may use different names for the same genes. The yeast proteome used here is as defined by SGD and accessed via RefSeq and contains 6,334 ORFs including the mitochondrial chromosome. Before performing comparisons, the various data sets were entered into BIND. Pairwise protein interaction data was entered as BIND interaction records.

BIND Data Model Format of MS Data

As BIND was conceived as a comprehensive archive of experimental molecular assembly information, representing affinity purification experimental data was part of the original design. Purification results that have been processed to remove promiscuously binding proteins have been entered into BIND, according to the data specification (Bader and Hogue, 2000), as complex records that group affinity associated proteins together.

The format for a single pull-down complex for the primary experimental data from both the Gavin *et al.* and Ho *et al.* data deposited into BIND is as follows:

BIND Complex Record: $C = \{i_1,i_2,i_3,i_4\}$
BIND Interaction Records: $i_1 = \{b\text{-}u\}$, $i_2 = \{u\text{-}c\}$, $i_3 = \{u\text{-}e\}$, $i_4 = \{u\text{-}f\}$
Where *u* is an unknown protein and *b* is the bait.

The BIND specification allows an interaction record to be defined as molecule A interacting with an unknown partner specifically for the purpose of representing complexes of unknown topology. Note that Gavin *et al.* refer to a "complex" as a manually compiled collection of proteins possibly from more than one purification, not the primary experimental results.

Either the matrix or the spoke model may be applied to the primary interaction data in BIND to generate the hypothetical pairwise interactions for analysis purposes. Further, the information may be updated in the future as the exact topology of the complex and exact pairwise interactions become known.

Visualization and Network Analysis

Visualization, network analysis and k-core finding were performed using the Pajek program for large network analysis (Batagelj and Mrvar, 1998) (http://vlado.fmf.uni-lj.si/pub/networks/pajek/) as described previously (Ho et al., 2002; Tong et al., 2002). Power law analysis was also accomplished as previously described (Ho et al., 2002). The connectivity distribution of the integrated yeast protein interaction network follows a power law with equation $y = 3{,}987x^{-1.8}$ with an R^2 value of 0.89. As can be seen from the R^2 value, the power law fit is not perfect. There is likely much noise in the integrated data set from various experiments. The true connectivity distribution will only be known when there is a perfectly known biological interaction network.

Chapter 6 – An Automated Method for Finding Molecular Complexes in Large Protein Interaction Networks

All of the work presented in this chapter will appear in the following publication:

Bader, G.D., Hogue, C.W.V.
An Automated Method for Finding Molecular Complexes in Large Protein Interaction Networks
(Submitted - June 2002)

Abstract

Recent advances in proteomics technologies such as two-hybrid, phage display and mass spectrometry have enabled us to create a detailed map of biomolecular interaction networks. Initial mapping efforts have already produced a wealth of data. As the size of the interaction set increases, databases and computational methods will be required to store, visualize and analyze the information in order to effectively aid in knowledge discovery. This paper describes a graph theoretic clustering algorithm that detects densely connected regions in large protein-protein interaction networks that may represent molecular complexes. The method is based on vertex weighting by local neighborhood density and outward traversal from a locally dense seed protein to isolate the dense regions according to given parameters. The algorithm has the advantage over other graph clustering methods of having a directed mode that allows fine-tuning of clusters of interest without considering the rest of the network and allows examination of cluster interconnectivity, which is relevant for protein networks. The "Molecular Complex Detection" (MCODE) algorithm has been implemented and evaluated using protein interaction and complex information from the yeast Saccharomyces cerevisiae. This is the first report of a predictive algorithm to find protein complexes in heterogeneous protein interaction data. Dense regions of protein interaction networks can be found, based solely on connectivity data, many of which correspond to known protein complexes. The algorithm is not affected by a known high rate of false positives in data from high-throughput interaction techniques. The program is available from ftp://ftp.mshri.on.ca/pub/BIND/Tools/MCODE.

Background

Recent papers published in *Science*, *Nature* among others describe large-scale proteomics experiments that have generated large data sets of protein-protein interactions and molecular complexes (Drees et al., 2001; Fields, 2001; Fromont-Racine et al., 2000; Gavin et al., 2002; Ho et al., 2002; Ito et al., 2001; Uetz et al., 2000). Protein structure (Christendat et al., 2000) and gene expression data (Kim et al., 2001) is also

accumulating at a rapid rate (Christendat et al., 2000; Drees et al., 2001; Fromont-Racine et al., 2000; Gavin et al., 2002; Ho et al., 2002; Ito et al., 2001; Kim et al., 2001). Bioinformatics systems for storage, management, visualization and analysis of this new wealth of data must keep pace. We previously published a simple graph theory method that identified a functional protein complex around the yeast protein Las17 that is involved in actin cytoskeleton rearrangement (Tong et al., 2002). Here we extend the method to better apply it to the accumulating information in protein networks.

Currently, most proteomics data is available for the model organism *Saccharomyces cerevisiae*, by virtue of the availability of a defined and relatively stable proteome, full genome clone libraries (Winzeler et al., 1999), established molecular biology experimental techniques and an assortment of well designed genomics databases (Chervitz et al., 1999; Costanzo et al., 2001; Mewes et al., 2000). Using the Biomolecular Interaction Network Database (BIND - http://www.bind.ca) (Bader et al., 2001) as an integration platform, 15,143 yeast protein-protein interactions were collected among 4,825 proteins (about 75% of the yeast proteome). Much larger data sets than this will eventually be available for other well studied model organisms as well as for the human proteome. These complex data sets present a formidable challenge for computational biology to develop automated data mining analyses for knowledge discovery.

Here the first report of an algorithm is presented to identify molecular complexes in a protein interaction network derived from heterogeneous experimental sources. Based on a previous observation that highly interconnected, or dense, regions of the network represent complexes (Tong et al., 2002), the "Molecular Complex Detection" (MCODE) algorithm has been implemented and evaluated on a yeast protein interaction compilation using known molecular complex data from a recent systematic mass spectrometry study of the proteome (Gavin et al., 2002) and from the MIPS database (Mewes et al., 2000).

Predicting molecular complexes from protein interaction data is important because it provides another level of functional annotation above other guilt-by-association methods. Since sub-units of a molecular complex generally function towards the same biological goal, prediction of an unknown protein as part of a complex allows increased confidence in the annotation of that protein.

MCODE also makes the visualization of large networks manageable by extracting the dense regions around a protein of interest. This is important, as it is now obvious that the current visualization tools present on many interaction databases (Bader et al., 2001), originally based on the Sun Microsystems graph layout Java applet do not scale well to large networks.

Algorithm

A network of interacting molecules can be intuitively modeled as a graph, where vertices are molecules and edges are molecular interactions. If temporal pathway or cell signaling information is known, it is possible to create a directed graph with arcs representing direction of chemical action or direction of information flow, otherwise an undirected graph is used. Using this graph representation of a biological system allows graph theoretic methods to be applied to aid in analysis and solve biological problems. This graph theory approach has been used by other biomolecular interaction database projects such as DIP (Xenarios et al., 2002), CSNDB (Takai-Igarashi et al., 1998), TRANSPATH (Wingender et al., 2000), EcoCyc (Karp et al., 2000) and WIT (Overbeek et al., 2000) and is discussed by Wagner and Fell (Wagner and Fell, 2001).

There is no standard graph theory definition of density, but definitions are normally based on the connectivity level of a graph. Density of a graph, G=(V,E), with number of vertices, |V|, and number of edges, |E|, is defined here as |E| divided by the theoretical maximum number of edges possible for the graph, $|E|_{max}$. For a graph with loops (an edge connecting back to its originating vertex), $|E|_{max} = |V| (|V|+1)/2$ and for a graph with no loops, $|E|_{max} = |V| (|V|-1)/2$. So, density of G, $D_G=|E|/|E|_{max}$ and is thus a real number ranging from 0.0 to 1.0. Algorithms for finding clusters, or locally dense regions, of a graph are an ongoing research topic in computer science and are often based on network flow theory/minimum cut (Flake et al., 2002; Goldberg, 1984). To find locally dense regions of a graph, MCODE instead uses a vertex-weighting scheme based on the clustering coefficient, which measures ‘cliquishness’ of the neighborhood of a vertex (Watts and Strogatz, 1998). A clique is defined as a maximally connected graph.

The MCODE algorithm operates in three stages, vertex weighting, complex prediction and optionally post-processing to filter or add proteins in the resulting complexes by certain connectivity criteria. The first stage, vertex weighting, weights all vertices based on their local network density using the highest k-core of the vertex neighborhood. A *k*-core, or *k*-connected region of a graph, contains vertices with at least *k* edges to other vertices in the core. The highest *k*-core is the central most densely connected region of a graph. The term core-clustering coefficient of a vertex, *v*, is defined here to be the density of the highest *k*-core of the immediate neighborhood of *v* (vertices connected directly to *v*) including *v*. This value is related to the clustering coefficient (Watts and Strogatz, 1998) of a vertex, *v*, which is the immediate neighborhood density of *v*. The core clustering coefficient is used here instead of the normal clustering coefficient because it amplifies the weighting of heavily interconnected graph regions while removing possible noise that can occur in a biomolecular interaction network, known to be scale-free (Albert et al., 2000; Barabasi and Albert, 1999; Fell and Wagner, 2000; Ho et al., 2002; Jeong et al., 2000; Wagner and Fell, 2001). A scale-free network has a vertex connectivity distribution that follows a power law, with relatively few highly connected vertices (high degree) and many vertices having a low degree. A given highly connected vertex, *v*, in a dense region of a graph may be connected to many vertices of degree one. These low degree vertices do not interconnect within the neighborhood of *v* and thus would reduce the normal clustering coefficient, but not the core-clustering coefficient. The final weight given to a vertex is the product of the vertex core clustering coefficient and the highest *k*-core level, k_{max}, of the immediate neighborhood of the vertex. This weighting scheme further boosts the weight of densely connected vertices. While this specific weighting function is based on local network density, it is partially empirical. Other possible functions are not evaluated here.

The second stage, molecular complex prediction, takes as input the vertex weighted graph, seeds a complex with the highest weighted vertex and recursively moves outward from the seed vertex, including vertices in the complex whose weight is above a given threshold, which is a given percentage away from the weight of the seed vertex. If a vertex is included, its neighbors are recursively checked in the same manner to see if they are part of the complex. A vertex is not checked more than once, since complexes

cannot overlap in this stage of the algorithm (see below for a possible overlap condition). This process stops once no more vertices can be added to the complex based on the given threshold and is repeated for the next highest unseen weighted vertex in the network. In this way, the densest regions of the network are identified. The vertex weight threshold parameter defines the density of the resulting complex. A threshold that is closer to the weight of the seed vertex identifies a smaller, denser network region around the seed vertex.

The third stage is post-processing. Complexes are filtered if they do not contain at least a two-core. The algorithm may be run with the 'fluff' option, which increases the size of the complex according to a given 'fluff' parameter between 0.0 and 1.0. For every vertex in the complex, *v*, its neighbors are added to the complex if they have not yet been seen and if their normal clustering coefficient is higher than the given fluff parameter. Vertices that are added by the fluff parameter are not marked as seen, so there can be overlap among predicted complexes with the fluff parameter set. If the algorithm is run using the 'haircut' option, the resulting complexes are two-cored, thereby removing the vertices that are singly connected to the core complex. If both options are specified, fluff is run first, then haircut.

Resulting complexes from the algorithm are scored and ranked. The complex score is empirically defined as the product of the complex subgraph, C=(V,E), density and the number of vertices in the complex subgraph (D_C x |V|). This ranks larger more dense complexes higher in the results.

MCODE may also be run in a directed mode where a seed vertex is specified as a parameter. In this mode, MCODE only runs once to predict the single complex that the specified seed is a part of. Typically, when analyzing complexes in a given network, one would find all complexes present (undirected mode) and then switch to the directed mode for the complexes of interest. The directed mode allows one to experiment with MCODE parameters to fine tune the size of the resulting complex according to existing biological knowledge of the system. In directed mode, MCODE will first pre-process the input network to ignore all vertices with higher vertex weight than the seed vertex. If this were not done, MCODE would preferentially branch out to denser regions of the graph, which could belong to a separate, but denser complex. Thus, a seed vertex for directed mode

should always be the highest density vertex among the suspected complex. There is an option to turn this pre-processing step off, which will allow seeded complexes to branch out into denser regions of the graph, if desired.

The time complexity of the entire algorithm is polynomial $O(mn^4)$ where n is the number of vertices and m is the number of edges in the input graph, G. This comes from the vertex-weighting step. Finding a k-core in a graph proceeds by progressively removing vertices of degree < k until all remaining vertices are connected to each other by degree k or more, and is thus $O(n^2)$. The highest k-core is found by trying to find k-cores from one up until all vertices have been found and cannot go beyond a number of steps equal to the highest degree in the graph. Thus, the highest k-core step is $O(n^3)$. The inner loop of the algorithm only operates twice for every edge in the input graph, thus is $O(2mn^3)$. The outer loop operates once on all vertices in the input graph, thus the entire time complexity of the weighting stage is $O(n2mn^3) = O(mn^4)$. The complex prediction stage is $O(n)$ and the optional post-processing step can be up to $O(cs^2)$, where c is the number of complexes that were found in the previous step and s is the number of vertices in the largest complex - $O(cs^2)$ to find the 2-core once for each complex.

Even though the fastest min-cut graph clustering algorithms are faster, at $O(n^2 \log n)$ (Hartuv and Shamir, 1999), MCODE has a number of advantages. Since weighting is done once and comprises most of the time complexity, many algorithm parameters can be tried, in $O(n)$, once weighting is complete. This is useful when evaluating many different parameters. MCODE is relatively easy to implement and since it is local density based, has the advantage of a directed mode and a complex connectivity mode. These two modes are generally not useful in typical clustering applications, but are useful for examining molecular interaction networks. Additionally, only those proteins above a given local density threshold are assigned to complexes. This is in contrast to many clustering applications that force all data points to be part of clusters, whether they truly should be part of a cluster or not.

Pseudocode

Stage 1: Vertex Weighting

```
procedure MCODE-VERTEX-WEIGHTING
        input: graph: G = (V,E)
        for all v in G do
                N = find neighbors of v to depth 1
                for all w in N do
                        K = Get highest k-core graph
                        k = Get highest k-core number
                        d = Get density of K
                        Set weight of v = k x d
                end for
        end for
end procedure
```

Stage 2: Molecular Complex Prediction

```
procedure MCODE-FIND-COMPLEX
        input: graph: G = (V,E); vertex weights: W;
                vertex weight percentage: d; seed vertex: s
        if s already seen then return
        for all v neighbors of s do
                if weight of v > (weight of s)(1 - d) then add v to complex C
                call: MCODE-FIND-COMPLEX (G, W, d, v)
        end for
end procedure

procedure MCODE-FIND-COMPLEXES
        input: graph: G = (V,E); vertex weights: W;
                vertex weight percentage: d
        for all v in G do
                if not already seen v then call: MCODE-FIND-COMPLEX(G, W, d, v)
        end for
end procedure
```

Stage 3: Post-Processing (optional)

```
procedure MCODE-FLUFF-COMPLEX
        input: graph: G = (V,E); vertex weights: W;
                fluff density threshold: d; complex graph: C = (U,F)
        for all u in C do
                if weight of u > d then add u to complex C
```

```
        end for
end procedure


procedure MCODE-POST-PROCESS
        input: graph: G = (V,E); vertex weights: W; haircut flag: h; fluff flag: f;
                fluff density threshold: t; set of predicted complex graphs: C
        for all c in C do
                if c not 2-core then filter
                if h is TRUE then 2-core complex
                if f is TRUE then call: MCODE-FLUFF-COMPLEX(G, W, t, c)
        end for
end procedure
```

Overall Process:

```
procedure MCODE
        input: graph: G = (V,E); vertex weight percentage: d;
                haircut flag: h; fluff flag: f; fluff density threshold: t;
                set of predicted complex graphs: C
        call: W = MCODE-VERTEX-WEIGHTING (G)
        call: C = MCODE-FIND-COMPLEXES (G, W, d)
        call: MCODE-POST-PROCESS (G, W, h, f, t, C)
end procedure
```

Implementation

MCODE has been implemented in ANSI C using the cross-platform NCBI Toolkit (http://www.ncbi.nlm.nih.gov/IEB) and the BIND graph library in the SLRI Toolkit (http://sourceforge.net/projects/slritools). Both of these source code libraries are freely available. The actual MCODE source code is not yet freely available. The MCODE program has been compiled and tested on UNIX, Mac OS X and Windows. Because a yeast gene name dictionary is used to recognize input and generate output, the MCODE executable currently only works for yeast proteins in a user friendly manner. The algorithm, however is completely general, via the graph theory abstraction, to any graph and thus to any biomolecular interaction network. MCODE binaries are available from ftp://ftp.mshri.on.ca/pub/BIND/Tools/MCODE.

Results

Evaluation of MCODE

The evaluation of MCODE requires a set of experimentally determined biomolecular interactions and a set of associated experimentally determined molecular complexes. Currently, the largest source for such data is for proteins from the budding yeast, *Saccharomyces cerevisiae*. Recently, a large-scale mass spectrometry study by Gavin et al. (Gavin et al., 2002) provided a large data set of protein interactions with manually annotated molecular complexes. Also available are the protein interaction and complex tables of MIPS (Mewes et al., 2000) and YPD (Costanzo et al., 2001). MCODE was used to automatically predict protein complexes in our collected protein-protein interaction data sets. Resulting complexes were then matched to known molecular complexes from Gavin et al. (the Gavin benchmark) and the MIPS benchmark using an overlap score. Parameter optimization was then used to maximize the biological relevance of predicted complexes according to the given benchmarks. YPD was not used as a current version could not be acquired.

To ensure that MCODE is not unduly affected by the expected high false-positive rate in large-scale interaction data sets, large-scale and literature derived MCODE predictions were compared. MCODE was then used to predict complexes in the entire set of machine readable protein-protein interactions that were collected for yeast. Complexes of interest were then further examined using the directed mode and complex connectivity mode of MCODE.

Evaluation of MCODE Using the Gavin *et al.* Data Set of Protein Interactions and Complexes

In this study, using all forms of protein interaction data available was desired, which requires mixing of different types of experiments, such as yeast two-hybrid and co-immunoprecipitation. Two-hybrid results are inherently pairwise, whereas copurification

results are sets of one or more proteins. For a copurification result, only a set of size 2 can be directly considered a pairwise interaction, otherwise it must be modeled as a set of hypothetical interactions. Biochemical copurifications can be thought of as populations of complexes with some underlying pairwise protein interaction topology that is unknown from the experiment. In the general case of the purification used by Gavin et al., one affinity tagged protein was used as bait to pull associated proteins out of a yeast cell lysate. The two extreme cases for the topology underlying the population of complexes from a single purification experiment are a minimally connected 'spoke' model, where the data are modeled as direct bait-associated protein pairwise interactions, and a maximally connected 'matrix' model, where the data are modeled as all proteins connected to all others in the set (See Chapter 5 – Integrated Experimental Protein Interaction Data Suggests a Large Nucleolar Complex in Saccharomyces cerevisiae). The real topology of the set of proteins must lie somewhere between these two extremes.

Gavin et al. raw data from 588 biochemical purifications were represented using the spoke model, described above, to get 3,225 hypothetical protein-protein interactions among 1,363 proteins for input to MCODE. A list of 232 manually annotated protein complexes based on the original purification data reported by Gavin et al. was filtered to remove five reported 'complexes' each composed of a single protein and six complexes of two or three proteins that were already in the data set as part of a larger complex. This yielded a filtered set of 221 complexes that were used to evaluate MCODE, although some of these complexes have significant overlap to other complexes in the set.

To evaluate which parameter choice would allow automatic prediction of protein complexes from the spoke modeled Gavin et al. interaction set that best matched the manually annotated complexes, MCODE was run using all four possible combinations of the two Boolean parameters over a full range of 20 vertex weight percentage (VWP) and fluff parameters (0 to 0.95 in 0.05 increments). During this parameter optimization process, MCODE was limited to find complexes of size two or higher.

A scoring scheme was developed to determine how effectively an MCODE predicted complex matched a complex from the benchmark set of complexes. In this case, the benchmark complex set was the Gavin et al. hand-annotated complex set. The overlap score was defined as $\omega = i^2/a*b$, where i is the size of the intersection set of a

predicted complex with a known complex, *a* is the size of the predicted complex and *b* is the size of the known complex. A protein is part of the intersection set only if it is present in both predicted and known complexes. Thus, a predicted complex that has no proteins in a known complex has $\omega = 0$ and a predicted complex that perfectly matches a known complex has $\omega = 1$. Also, predicted complexes that fully overlap, but are much larger or much smaller than any known complexes will get a low ω. The overlap score of a predicted complex vs. a benchmark complex is then a measure of biological significance of the prediction, assuming that the benchmark set of complexes is biologically relevant. The best parameter choice for MCODE on this protein interaction data set is one that predicts a set of complexes that match the largest number of benchmark complexes above a threshold ω. Since there is overlap in the Gavin benchmark complex database, a predicted complex may match more than one known complex with a high ω.

For each of the 840 parameter combinations tested during the parameter optimization stage, the number of MCODE predicted complexes was plotted against the number of matched known complexes over a range of ω thresholds from 'no threshold' to 0.1 to 0.9 (in 0.1 increments). If no ω threshold is used, a predicted complex only needs at least one protein in common with a known complex to be considered a match. If predicted and known complexes are only counted as a match when their ω is above a specific threshold, the number of matched complexes declines with increasing ω threshold, as shown in Figure 32. Interestingly, the average and maximum number of matched known complexes drops more quickly from zero until a ω threshold of 0.2 than from 0.2 to 0.9 indicating that many predicted complexes only have one or a few proteins that overlap with known complexes. A ω threshold of 0.2 to 0.3 thus seems to filter out most predicted complexes that have insignificant overlap with known complexes. Figure 33 shows the range of number of complexes predicted and number of known complexes matched for the 0.2 ω threshold over all tried MCODE parameters. A y=x line is also plotted to show that data points tend to be skewed towards a higher number of matched known complexes than predicted complexes because of the redundancy in the Gavin complex benchmark. Data points closest to the upper right portion of the graph

maximize both number of matched known complexes and number of predicted complexes. MCODE parameter combinations that result in these data points therefore optimize MCODE on this data set (according to the overlap score threshold). This result shows that the number of predicted complexes should be similar to the number of matched known complexes for a parameter choice to be reasonable, although the number of matched known complexes may be larger because of some commonality among complexes in the benchmark set. The parameter combination corresponding to the best data point (63,88) at an overlap score threshold of 0.2 is haircut=FALSE, fluff=TRUE, VWP=0.05 and a fluff density threshold between 0 and 0.1. These parameter optimization results for MCODE over this data set were stable over a range of ω thresholds up to 0.5. Above 0.5, the result was not stable as there were generally too few predicted complexes with high overlap scores (Figure 32).

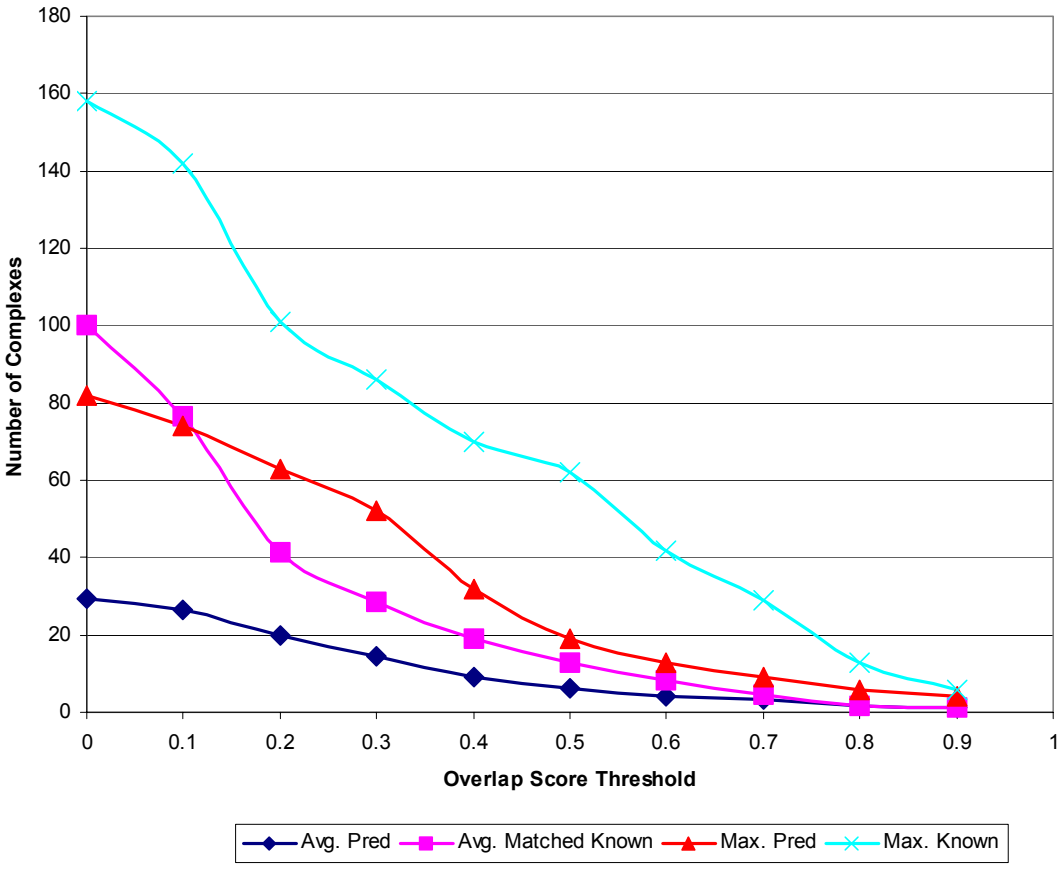


Figure 32: Effect of Overlap Score Threshold on Number of Predicted and Matched Known Complexes

Average and maximum number of predicted and matched known complexes seen during MCODE parameter optimization (840 parameter combinations) plotted as a function of overlap score threshold. As the stringency for the closeness that a predicted complex must match a known complex is increased (increase in overlap score), fewer predicted complexes match known complexes.

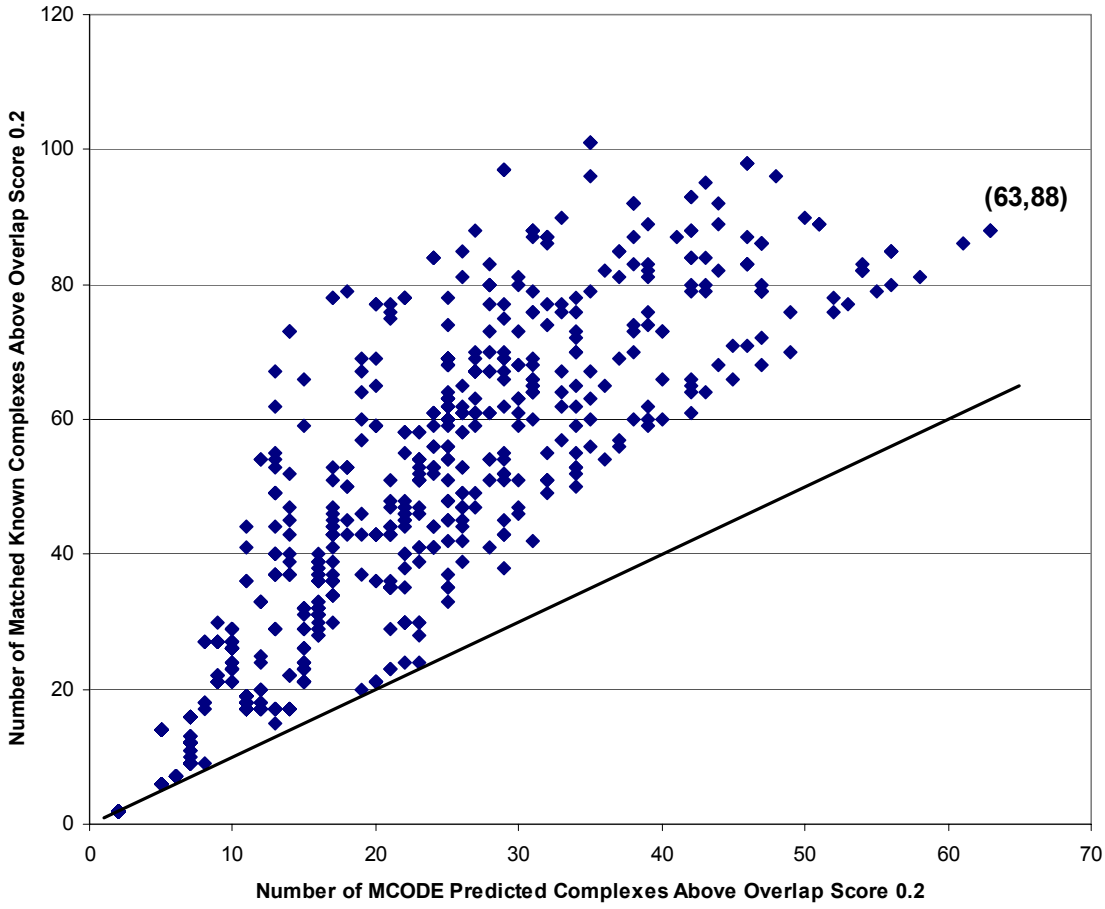


Figure 33: Number of Predicted and Matched Known Complexes at Overlap Score Threshold of 0.2

Number of known complexes matched to MCODE predicted complexes plotted against number of MCODE predicted complexes, both with an overlap score above 0.2.

A specificity versus sensitivity analysis (Baldi et al., 2000) was also performed. Defining the number of true positives (TP) as the number of MCODE predicted complexes with ω over a threshold value and the number of false positives (FP) as the total number of predicted MCODE complexes minus TP. The number of false negatives (FN) equals the number of known benchmark complexes not matched by predicted complexes. Sensitivity was defined as [TP/(TP+FN)] and specificity was defined as [TP/(TP+FP)]. The MCODE parameter choice that optimizes both specificity and sensitivity is the same as from the above analysis.

MCODE predicted complexes only matched 88 of the 221 complexes in the known data set indicating that MCODE could not recapitulate the majority of the Gavin complex benchmark solely using protein connectivity information. This is not surprising, since many of the hand-annotated complexes were created directly from single co-immunoprecipitation results, which are not highly interconnected in the spoke model. For example, CDC3 was used as a bait to co-immunoprecipitate CDC10, CDC11, CDC12 and YDL225W. A complex was annotated as containing these five proteins, but only CDC3 was used as bait. If more elements of a complex are used as baits, the proteins become more interconnected and more readily predicted by MCODE. A good example of this is the Arp2/3 complex, which is highly conserved in eukaryotes and is involved in actin cytoskeleton rearrangement. The structure of this complex is known by X-ray crystallography (Robinson et al., 2001) thus actual protein-protein interactions from the structure can be matched up to the co-immunoprecipitation results. MCODE predicted all seven components of the Arp2/3 complex crystal structure and five extra proteins using the optimized parameters. Six out of the seven Arp2/3 subunits were used as baits by Gavin et al. and the resulting benchmark complex included the five extra proteins that MCODE also predicted (Nog2, Pfk1, Prt1, Cct8 and Cct5) that are not in the crystal structure. Cct5 and Cct8 are known to be involved in actin assembly, but Nog2, Pfk1 and Prt1 are not. These extra proteins likely represent non-specific binding in the experimental approach. These two cases are shown diagrammatically in Figure 34. Interestingly, using the haircut parameter would remove all five extra proteins that are not in the crystal structure, leaving only the seven that are present. This shows that while the parameter optimization allows maximum matching of the hand-annotated known

complexes, these complexes may not all be physiologically relevant and thus another parameter set may better predict 'real' complexes.

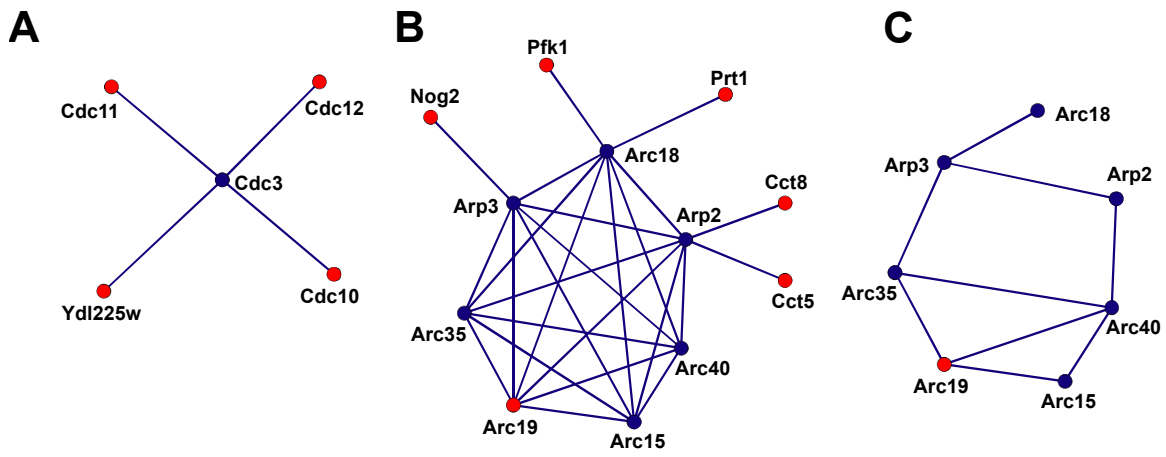


Figure 34: Examples of Gavin *et al.* Annotated Complexes Missed and Hit by MCODE

Protein complexes are represented as graphs using the spoke model. Vertices represent proteins and edges represent experimentally determined interactions. Blue vertices are baits in the Gavin et al. study. A) A CDC3 complex hand-annotated by Gavin et al. that was missed by MCODE because of a lack of connectivity information among sub-components. This complex annotation was the result of a single co-immunoprecipitation experiment. B) The Arp2/3 complex as annotated by Gavin et al. and as found by MCODE with parameters optimized to the data set. Note the five extra proteins that have minimal connectivity to main cluster. C) The protein connection map seen from the crystal structure of the Arp2/3 complex. The crystal structure is from *Bos taurus* (cow), but is assumed to be very similar to yeast based on very high similarity between cow and yeast Arp2/3 subunits.

To explore the effect of certain MCODE parameters on resulting predicted complexes, various features of these complexes were examined while changing specific parameters and keeping all else constant. Linearly increasing the VWP parameter increased the size of the predicted complexes exponentially while reducing the number of complexes predicted in a linear fashion. Figure 35 shows this effect with both fluff and haircut parameters turned off. At high VWP values, very large complexes were predicted and these encompassed most of the data set, thus were not very useful.

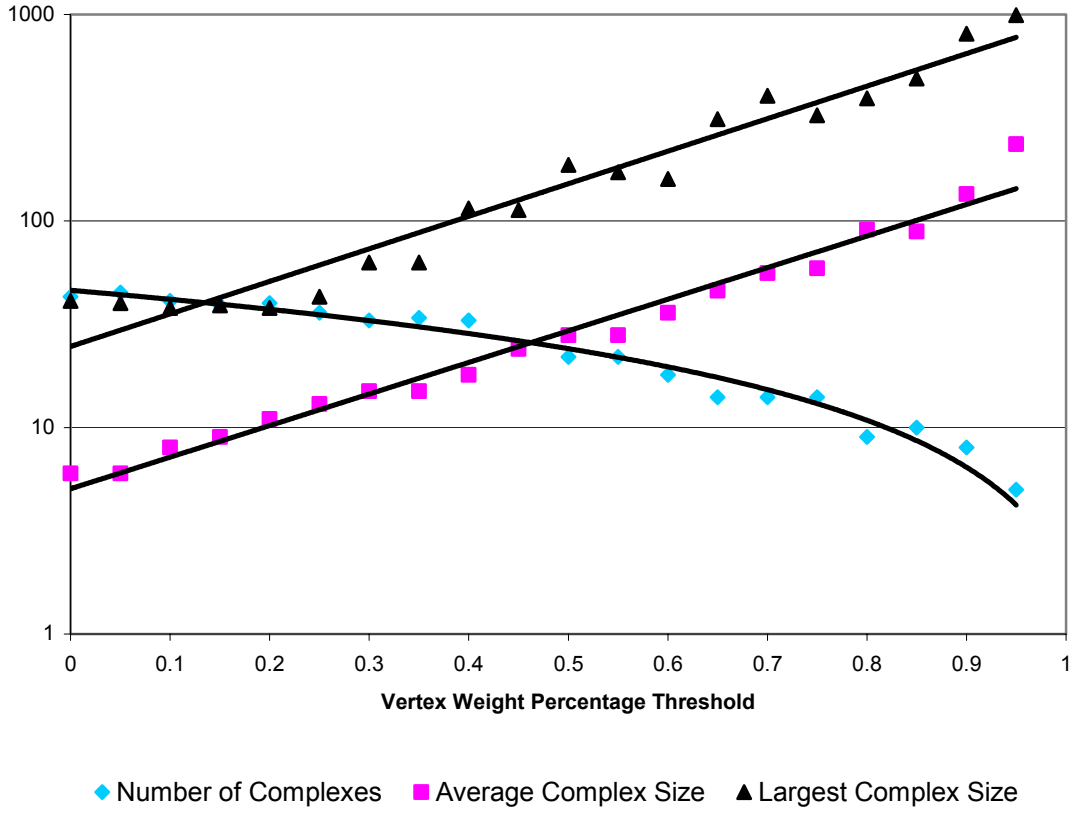


Figure 35: Effect of Vertex Weight Percentage Parameter on Predicted Complex Size

As the vertex weight percentage parameter of MCODE is increased, the number of predicted complexes steadily decreases and the average and largest size of predicted complexes increases exponentially. The y-axis follows a logarithmic scale.

Because using haircut=TRUE would have led MCODE to predict the Arp2/3 complex perfectly (according to the crystal structure as discussed above), the haircut parameter was examined to see if it has any general effect on the number of matched predicted complexes. Setting haircut=TRUE had no significant effect on the number of complexes predicted, but generally reduced the number of matched known complexes at low ω thresholds (0 to 0.1) compared to haircut=FALSE. At higher ω thresholds, haircut=TRUE had no significant effect. Since the haircut=TRUE option removes less-connected proteins on the fringe of a predicted complex and this reduces the number of predicted complexes with low overlap scores, these fringe proteins likely contribute to low-level overlap (<0.2 ω) of the known complexes.

The effect of changing the fluff density threshold when setting fluff=TRUE on the number of matched benchmark complexes was also investigated. Linearly increasing the fluff density threshold in the MCODE post-processing step linearly decreased the number of matched complexes above an overlap score of 0.2.

Evaluation of MCODE Using MIPS Data Set of Protein Interactions and Complexes

Since the Gavin et al. data set was developed by only one group using a single experimental method, it may not accurately represent protein complex knowledge for yeast. The MIPS protein complex catalogue (http://mips.gsf.de/proj/yeast/catalogues/complexes/) is a well-curated set of 260 protein complexes for yeast that was compiled from the literature and is thus a more realistic data set comprised of varied experiments from many labs using different techniques. After filtering away 50 'complexes' each composed of a single protein and 2 highly similar complexes, 208 complexes were left in the MIPS known set. This set did not include information from the recent large-scale mass spectrometry studies (Gavin et al., 2002; Ho et al., 2002).

MCODE was run again with a full combination of parameters, this time over a set of 9088 protein-protein interactions among 4379 proteins which did not include the recent large-scale mass spectrometry studies but included all interactions from the MIPS, YPD and PreBIND databases as well as from the majority of large-scale yeast two-hybrid experiments to date (Drees et al., 2001; Ito et al., 2001; Mayes et al., 1999; Tong et al.,

2002; Uetz et al., 2000). This interaction set is termed 'Pre HTMS'. All of the interactions in this set were published before the last update specified on the MIPS protein complex catalogue and many are included in the MIPS protein interaction table, thus it was assumed that the MIPS complex catalogue took into account the information in the known interaction table. Protein complexes found by MCODE in this set were compared to the MIPS protein complex catalogue to evaluate how well MCODE performed at locating protein complexes *ab initio*.

The same evaluation of MCODE that was done using the Gavin et al. data set was performed with the MIPS data set. From this analysis, including specificity versus sensitivity plots, the MIPS complex benchmark optimized parameters were haircut=TRUE, fluff=TRUE, VWP=0.1 and a fluff density threshold of 0.2. This result was stable up to a ω threshold of 0.6 after which it was difficult to evaluate the results as there were generally too few predicted complexes above the high ω thresholds. This parameter combination led MCODE to predict 166 complexes of which 52 matched 64 MIPS complexes with a ω of at least 0.2. Examining the ω distribution for this parameter set reveals that, even though this prediction is optimized, most of the predicted complexes don't show overlap to those in the known MIPS set (Figure 36). This might signify that either the MIPS complex catalogue is not complete, that there is not enough data in the dataset which MCODE was run on, or a human annotated definition of a complex does not perfectly match with a graph density based definition.

The effect of the VWP parameter on complex size and of the haircut and fluff parameters on number of matched complexes was very similar to that seen when evaluating MCODE on the Gavin complex benchmark.

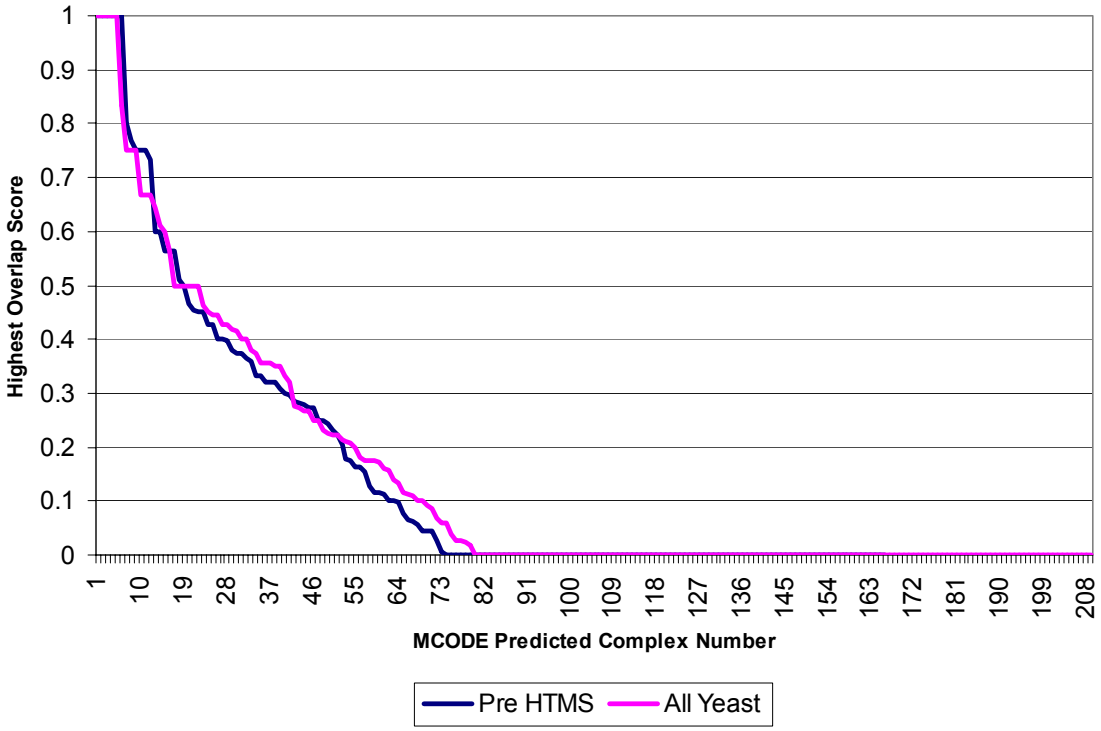


Figure 36: Overlap Score Distributions of Pre HTMS and AllYeast interaction sets with MIPS Complex Benchmark Optimized MCODE Parameter Sets

The number of MCODE predicted complexes in the pre-large scale mass spectrometry (Pre HTMS) and AllYeast protein-protein interaction sets with a given overlap score threshold compared to the MIPS benchmark complex set is shown. The majority of predicted complexes have an overlap score of zero meaning that they had no overlap with the catalogue of known MIPS protein complexes.

Effect of Data Set Properties on MCODE

Since many large-scale protein interaction data sets from yeast are known to contain a high level of false positives (von Mering et al., 2002), the effect these might have on MCODE predictions was examined. Sensitivity vs. specificity was plotted for MCODE predictions, with parameters chosen to maximize these values at ω threshold of 0.2 against the MIPS and Gavin complex benchmarks for the various data sets (Figure 37).

MCODE predictions on the high-throughput data sets, termed 'Gavin Spoke', 'Y2H' and 'HTP only' (see Methods), are about as specific as the literature derived interaction data set, but not as sensitive (Figure 37A). MCODE predictions on interaction data sets containing the literature derived benchmark, labelled 'Benchmark', 'Pre HTMS' and 'AllYeast', are generally more sensitive and specific than those containing just the large-scale interaction sets. This shows that the addition of large-scale experimentally derived interactions that are known to contain a high number of false positives do not unduly affect the prediction of complexes by MCODE.

It can be seen from Figure 37B that the Gavin complex benchmark set is biased towards the Gavin et al. spoke modeled interaction data. This is expected and is the main reason why the less biased MIPS complex set is used throughout this work as a benchmark instead of the Gavin set.

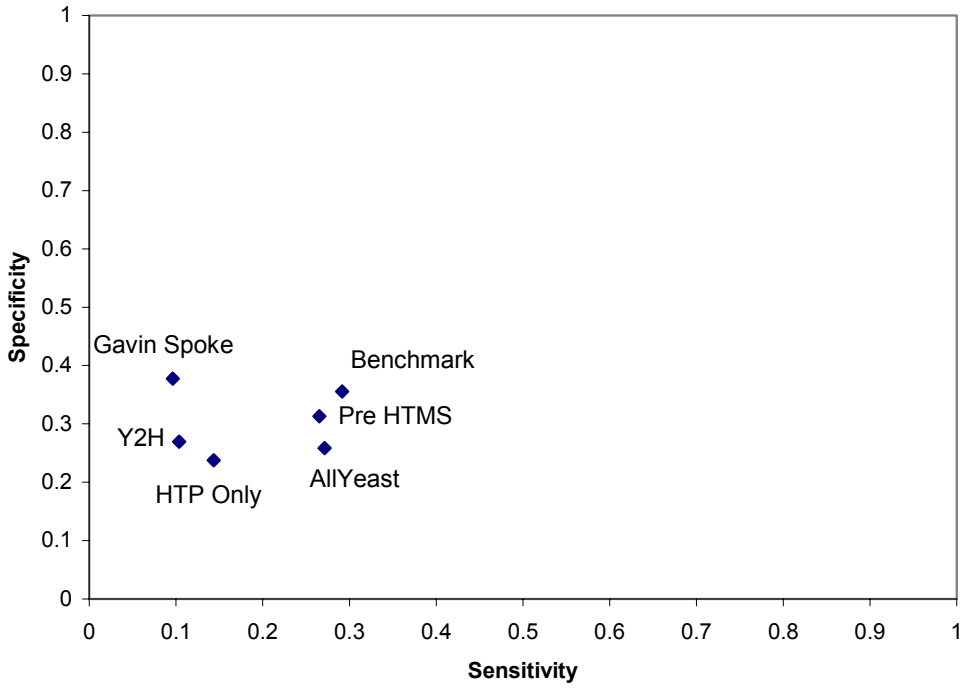
Since the result of a co-immunoprecipitation experiment is a set of proteins, which are modeled as binary interactions using the spoke method, it was useful to evaluate whether this affects complex prediction compared to an experimental system that generates purely binary interaction results, such as yeast two-hybrid. As can be seen in Table 11, MCODE does find known complexes in the 'Y2H' set of only yeast two-hybrid results, thus this set does contain dense regions that are known protein complexes. This being said, the Y2H set is the least dense of all data sets examined here so is expected to have less dense regions of the network and thus less MCODE predictable complexes per number of proteins present in the set. MCODE predicts a similar amount of complexes as well as finding a similar amount of known complexes in the Y2H and

Gavin Spoke data sets indicating that these data sets are not significantly different from each other in the amount of dense network regions that they contain, even though they are different sizes. Taken together, the latter results and those in Figure 37B show that the spoke model is a reasonable representation of the Gavin et al tandem affinity purification data.

Figure 37: Sensitivity vs. Specificity Plots of MCODE Results Among Various Data Sets

Specificity is plotted versus sensitivity of the best MCODE results at an overlap score above 0.2 against both the MIPS (Panel A) and Gavin (Panel B) complex benchmarks. Panel A shows that there is no large inherent difference among interaction data sets resulting from significantly different experimental methods. Panel B shows that the Gavin benchmark is expectedly biased towards the Gavin interaction data set and thus should not be used as a general benchmark.

A


Specificity
Sensitivity
Gavin Spoke
Benchmark
Pre HTMS
Y2H
AllYeast
HTP Only

B

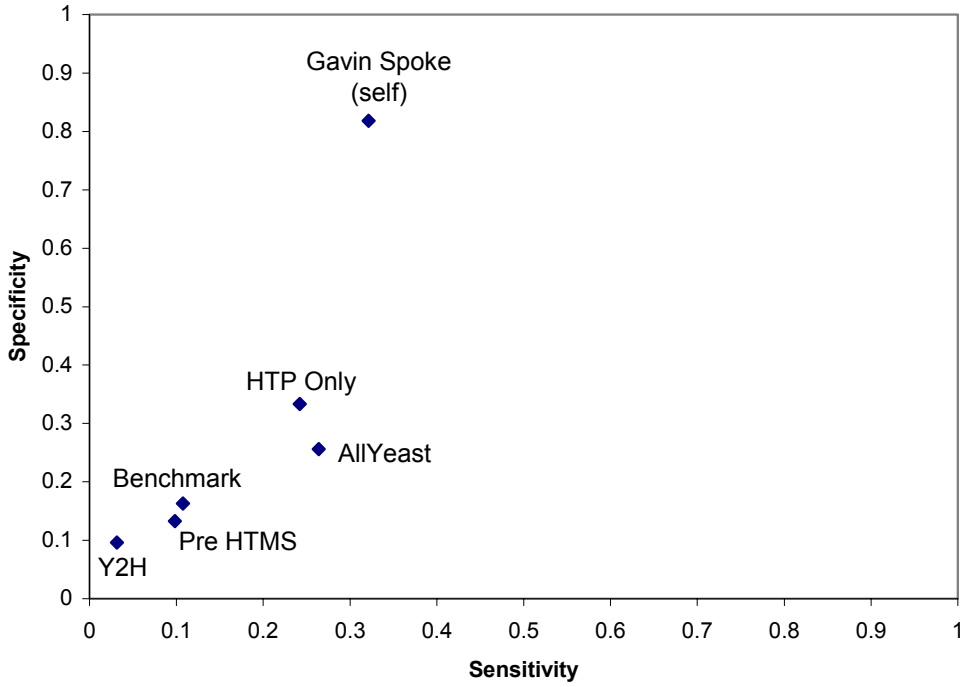

Specificity
Sensitivity
Gavin Spoke
(self)
HTP Only
AllYeast
Benchmark
Pre HTMS
Y2H

Figure 37

Data Set	Number of Proteins	Number of Interact-ions	Number of Predicted Complexes	MCODE Complexes Predicted Above ω=0.2	Matched Benchmark Complexes	Complex Benchmark	Best MCODE Parameters
Gavin Spoke	1363	3225	77	63	88	Gavin	hFfT\0.05\0.05
Gavin Spoke	1363	3225	53	20	20	MIPS	hTfT\0.1\0.35
Pre HTMS	4379	9088	158	21	28	Gavin	hTfT\0\0.2\
Pre HTMS	4379	9088	166	52	64	MIPS	hTfT\0.1\0.2
AllYeast	4825	15143	209	52	76	Gavin	hFfT\0\0.1
AllYeast	4825	15143	209	54	63	MIPS	hTfT\0\0.1
Benchmark	1762	3310	141	23	30	Gavin	hTfT\0\0.3
Benchmark	1762	3310	163	58	67	MIPS	hTfT\0.1\0.05
HTP Only	4557	12249	138	46	77	Gavin	hTfT\0.05\0.1
HTP Only	4557	12249	122	29	35	MIPS	hTfT\0.05\0.15
Y2H	3847	6133	73	7	7	Gavin	hTfT\0.2\0.1
Y2H	3847	6133	78	21	26	MIPS	hTfT\0\0.1

Table 11: Summary of MCODE Results with Best Parameters on Various Data Sets

Statistics and a summary of results are shown for the various data sets used to evaluate MCODE. 'Gavin Spoke' is the Gavin et al. data set represented as binary interactions using the spoke model; 'Pre HTMS' is the set of all yeast interaction not including the recent high-throughput mass spectrometry studies (Gavin et al., 2002; Ho et al., 2002); 'AllYeast' is the set of all yeast interactions that were collected for yeast; 'Benchmark' is a set of interactions found in the literature from YPD, MIPS and PreBIND; 'HTP Only' is the combination of all large-scale and high-throughput yeast two-hybrid and mass spectrometry data sets; 'Y2H' is the set of all yeast two-hybrid results from large-scale and literature sources. See Methods for full explanation of data sets. The 'Best MCODE Parameters' are formatted as haircut True of False, fluff True or False\VWP\Fluff Density Threshold Parameter.

Predicting Complexes in the Yeast Interactome

Given that MCODE performed reasonably well on test data, it was decided to predict complexes in a much larger network. All machine-readable protein-protein interaction data from various data sets (Costanzo et al., 2001; Drees et al., 2001; Fromont-Racine et al., 2000; Gavin et al., 2002; Ho et al., 2002; Ito et al., 2001; Mewes et al., 2000; Tong et al., 2002; Uetz et al., 2000) were collected and integrated to form a non-redundant set of 15,143 experimentally determined yeast protein interactions encompassing 4,825 proteins, or approximately three quarters of the proteome. This set was termed 'AllYeast'. MCODE was parameter optimized, as above, using the MIPS benchmark. The best resulting parameter set was haircut=TRUE, fluff=TRUE, VWP=0 and a fluff density threshold of 0.1. With these parameters, MCODE predicted 209 complexes, of which 54 matched 63 MIPS benchmark complexes above an overlap score of 0.2. Complexes found in this manner should be further studied using MCODE in directed mode by specifying a seed vertex and trying different parameters to examine how large a complex can get before seemingly biologically irrelevant proteins are added (see below).

Figure 36 shows that even when a large set of interactions is used as input to MCODE, most of the MCODE predicted complexes do not match well with known complexes in MIPS. The complex size distribution of MCODE predicted complexes matches the shape of the MIPS set, but the MCODE complexes are on average larger (Average MIPS size=6.0, Average MCODE Predicted size=9.7). The average number of YPD and GO functional annotation terms per protein in an MCODE predicted complex is similar to that of MIPS complexes (Table 12). This seems to indicate that MCODE is predicting complexes that are functionally relevant. Also, closer examination of the top, middle and bottom five scoring MCODE complexes shows that MCODE can predict biologically relevant complexes (Table 13).

Many of the 209 predicted complexes are of size 2 (9 predicted complexes) or 3 (54 predicted complexes). Complexes of this size may not be significant since it is easy to create high density subgraphs of size 2 or 3, but becomes combinatorially more

difficult to randomly create high density subgraphs as the size of the subgraph increases. To examine the relevance of these small predicted complexes of size 2 or 3, the sensitivity and specificity of the optimized MCODE predictions against the MIPS complex benchmark was calculated while disregarding the small complexes. First, complexes of size 2, then of size 3, were removed from the optimized MCODE predicted complex set. The specificity only slightly increased when small predicted complexes were removed (0.26 to 0.27 for size 2 removal and 0.26 to 0.28 for size 3 removal). Sensitivity however, decreased, slightly for size 2 removal (0.27 to 0.26) and significantly (0.27 to 0.2) for size 3 removal. Thus, predicted complexes of size 3 are more significant than those of size 2, although both sets overlap the MIPS benchmark. In light of these results, small complexes have been reported as predictions. Also, because MCODE found these small complexes in regions of high local density, they may be good cores for further examination with MCODE in directed mode, especially since the haircut option was turned on here.

Data Set	YPD Functions	YPD Roles	GO Components	GO Processes
MCODE on All Yeast Interactions	0.58	0.89	0.39	0.59
MIPS Complex Database	0.50	0.75	0.39	0.48
MCODE Random Model (100 AllYeast network permutations)	0.72	1.24	0.52	0.85

Table 12: Average Number of YPD and GO Annotation Terms in Complex Sets

The average number of YPD and GO functional annotation terms per protein in an MCODE predicted complex is shown for MCODE predicted complexes on the AllYeast set, the MIPS complex database and the MCODE random model. A lower number indicates that the complexes from a set contain more functionally related proteins (or unannotated proteins). In the cases of multiple annotation, all terms are taken into account. Even though there are multiple annotation terms per protein and a variable amount of unannotated proteins per complex, these numbers should perform well in relative comparisons based on the assumption that the distribution of the latter two factors is similar in each data set.

Complex Rank	Score	Proteins	Interactions	Density	Cell Role	Cell Localization
1	10.04	46	236	0.22	RNA processing/ modification and protein degradation (26S Proteasome)	Nuclear
Protein names		Dbf2,Ecm29,Gcn4,Hsm3,Hyp2,Lhs1,Mkt1,Nas6,Pre1,Pre2,Pre4,Pre5,Pre6, Pre7,Pre8,Pre9,Pup3,Rad23,Rad24,Rad50,Rfc3,Rfc4,Rpn1,Rpn10,Rpn11,Rp n12,Rpn13,Rpn3,Rpn4,Rpn5,Rpn6,Rpn7,Rpn8,Rpn9,Rpt1,Rpt2,Rpt3,Rpt4,R pt5,Rpt6,Scl1,Ubp6,Ura7,Ygl004c,Yku70,Ypl070w				
2	9	19	90	0.51	RNA processing/modification	Nuclear
Protein names		Cft1,Cft2,Fip1,Fir1,Hca4,Mpe1,Pap1,Pcf11,Pfs2,Pta1,Pti1,Ref2,Rna14,Ssu72, Uba2,Ufd1,Yor179c,Ysh1,Yth1				
3	7.72	56	220	0.14	Pol II transcription	Nuclear
Protein names		Ada2,Adr1,Ahc1,Cdc23,Cdc36,Epl1,Esa1,Fet4,Fun19,Gal4,Gcn5,Hac1,Hfi1, Hhf2,Hht1,Hht2,Ire1,Luc7,Med7,Myo4,Ngg1,Pcf11,Pdr1,Prp40,Rna14,Rpb2,R po21,Sap185,Sgf29,Sgf73,Spt15,Spt20,Spt3,Spt7,Spt8,Srb6,Swi5,Taf1,Taf10 ,Taf11,Taf12,Taf13,Taf14,Taf2,Taf3,Taf5,Taf6,Taf7,Taf8,Taf9,Tra1,Ubp8,Yap 1,Yap6,Ybr270c,Yng2				
4	7.58	18	72	0.44	Cell cycle control, protein degradation, mitosis (Anaphase Promoting Complex)	Nuclear
Protein names		Apc1,Apc11,Apc2,Apc4,Apc5,Apc9,Cdc16,Cdc23,Cdc26,Cdc27,Dmc1,Doc1, Leu3,Rpt1,Sic1,Spc29,Spt2,Ybr270c				
5	7	15	56	0.52	Vesicular transport (TRAPP Complex)	Golgi
Protein names		Bet1,Bet3,Bet5,Fks1,Gsg1,Gyp6,Kre11,Sec22,Trs120,Trs130,Trs20,Trs23, Trs31,Trs33,Uso1				
102	3	3	3	1	RNA splicing	Nuclear
Protein names		Msl5,Mud2,Smy2				
103	3	3	3	1	Signal transduction, Cell cycle control, DNA repair, DNA synthesis	Nuclear
Protein names		Ptc2,Rad53,Ydr071c				
104	3	3	3	1	Cell cycle control, mating response	Uknown
Protein names		Far3,Vps64,Ynl127w				
105	3	3	3	1	Chromatin/chromosome structure	Nuclear
Protein names		Gbp2,Hpr1,Mft1				
106	3	3	3	1	Pol II transcription	Nuclear
Protein names		Ctk1,Ctk2,Ctk3				
205	2	3	4	1	Vesicular transport	ER
Protein names		Rim20,Snf7,Vps4				
206	2	3	4	1	Protein translocation	Cytoplasmic
Protein names		Srp14,Srp21,Srp54				

207	2	3	4	1	Protein translocation	Cytoplasmic
Protein names		Srp54,Srp68,Srp72				
208	2	3	4	1	Energy generation	Mitochondrial
Protein names		Atp1,Atp11,Atp2				
209	2	4	5	0.67	Nuclear-cytoplasmic and vesicular transport	Varied
Protein names		Kap123,Nup145,Sec7,Slc1				

Table 13: Statistics for Top, Middle and Bottom Five Scoring Optimized MCODE Predicted Complexes Found in All Known Yeast Protein Interaction Data Set

Score is defined as the product of the complex subgraph density and the number of vertices (proteins) in the complex subgraph (D_C x |V|). This ranks larger more dense complexes higher in the results. Density is calculated using the loop formula if homodimers exist in the complex, otherwise the 'no loop' formula is used.

Complexes that are larger and denser are ranked higher by MCODE and these generally correspond to known complexes. Interestingly, some MCODE complexes contain unknown proteins that are highly connected to known complex subunits. For example, the second highest ranked MCODE complex is involved in RNA processing/modification and contains the known polyadenylation factor I complex (Cft1, Cft2, Fip1, Pap1, Pfs2, Pta1, Ysh1, Yth1 and Ykl059c). Seven other proteins involved in mainly RNA processing/modification (Fir1, Hca4, Pcf11, Pti1, Ref2, Rna14, Ssu72) and protein degradation (Uba2 and Ufd1) are highly connected within this predicted complex. Two unknown proteins Pti1 and Yor179c are highly connected to RNA processing/modification proteins and are therefore likely involved in the same process (Figure 38). Pti1 may be an unknown component of the polyadenylation factor I complex. The 23rd highest ranked predicted complex is interesting in that it is involved in cell polarity and cytokinesis and contains two proteins of unknown function, Yhr033w and Yal027w. Yal027w interacts with two kinases, Gin4 and Kcc4, which in turn interact with the components of the Septin complex (Cdc3, Cdc10, Cdc11 and Cdc12) (Figure 39).

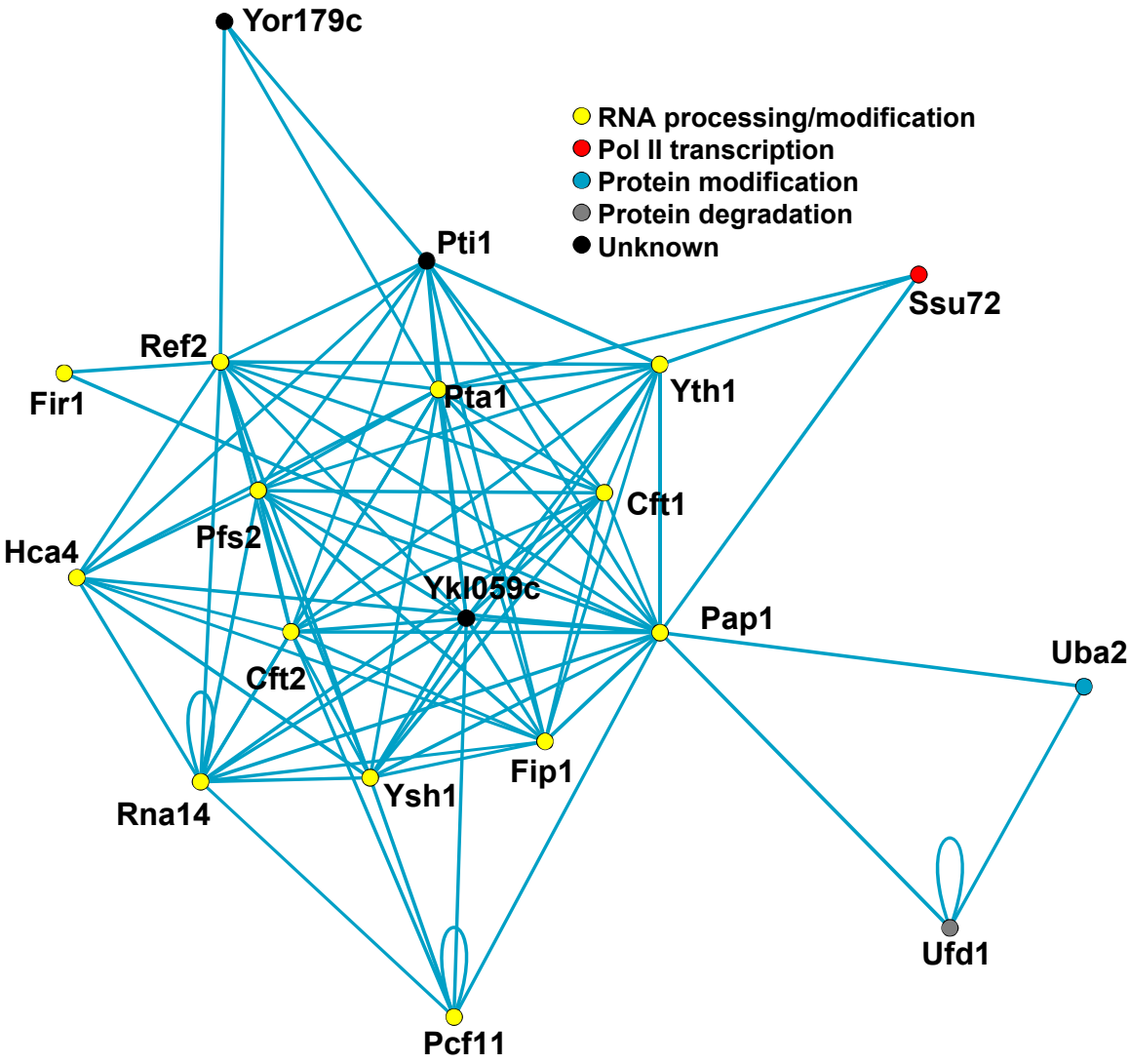


Figure 38: The Second Highest Ranked MCODE Predicted Complex is Involved in RNA Processing and Modification

This complex incorporates the known polyadenylation factor I complex (Cft1, Cft2, Fip1, Pap1, Pfs2, Pta1, Ysh1, Yth1 and Ykl059c) and contains other proteins highly connected to this complex, some of unknown function. The fact that the unknown proteins (Yor179c and Pti1) connect more to known RNA processing/modification proteins than to other proteins in the larger data set likely indicates that these proteins function in RNA processing/modification. This complex was most highly ranked by MCODE from the predicted complexes in the AllYeast interaction set.

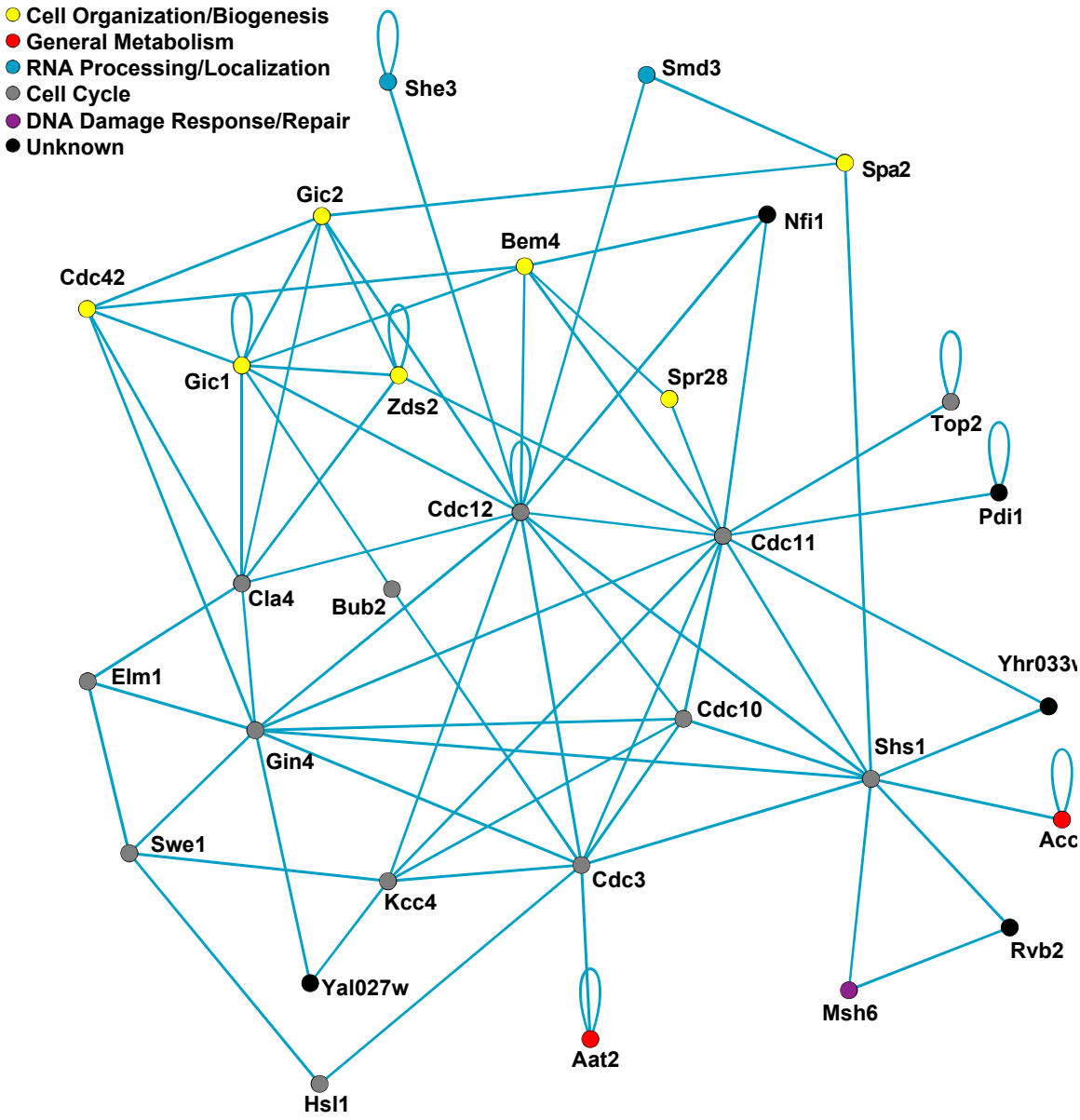


Figure 39: An MCODE Predicted Complex Involved in Cytokinesis

This predicted complex incorporates the known Septin complex (Cdc3, Cdc10, Cdc11 and Cdc12) involved in cytokinesis and other cytokinesis related proteins. The Yal027w protein is of unknown function, but likely functions in cell cycle control according to this figure, possibly of cytokinesis. This complex was ranked 23rd by MCODE from the predicted complexes in the AllYeast interaction set.

Significance of MCODE Predictions

Recent research on modeling complex systems (Albert et al., 2000; Wagner and Fell, 2001; Watts and Strogatz, 1998) has found that networks such as the world wide web, metabolic networks (Jeong et al., 2000) and protein-protein interaction networks (Jeong et al., 2001) are scale-free. That is, the connectivity distribution of the vertices of the graph follows a power law, with many vertices of low degree and few vertices of high degree. Scale-free networks are known to have large clustering coefficients, or clustered regions of the graph. In biological networks, at least in yeast, these clustered regions seem to correspond to molecular complexes and these subgraphs are what MCODE is designed to find.

To test the significance of clustered regions in biological networks, 100 random permutations of the large set of all 15,143 yeast interactions were made. The random networks have the same number of edges and vertices as the original network and follow a power-law connectivity distribution. Running MCODE with the same parameters as the original network (haircut=TRUE, fluff=TRUE, VWP=0 and a fluff density threshold of 0.1) on the 100 random networks resulted in an average of 27.4 (SD=4.4) complexes per network. The size distribution of complexes found by MCODE did not match that of the complexes found in the original network, as some complexes found in the random networks were composed of >1500 proteins. One random network that had an approximately average number of predicted complexes (27) was parameter optimized using the MIPS benchmark to see how parameter choice affects the size distribution and number of predicted complexes. Parameters of haircut=TRUE, fluff=TRUE, VWP=0.1 and a fluff density threshold of zero produced the maximal number of 81 complexes for this network, but these complexes were composed of on average 27 proteins (without counting an outlier complex of size 1961), which is much larger than normal (e.g. larger than the MIPS set average of 6.0). None of these predicted complexes matched any MIPS complexes above an overlap score of 0.1. Also, the random network complexes had a much higher average number of YPD and GO annotation terms per protein per complex than for MIPS or MCODE on the original network (Table 12). This indicates,

as expected, that the random network complexes are composed of a higher level of unrelated proteins than complexes in the original network. Thus, the number, size and functional composition of complexes that MCODE predicts in the large set of all yeast interactions are highly unlikely to occur by chance.

Directed Mode of MCODE

To simulate an obvious example where the directed mode of MCODE would be useful, MCODE was run with relaxed parameters (haircut=TRUE, fluff=TRUE, VWP=0.05 and a fluff density threshold of 0.2) compared to the best parameters on the AllYeast network. The resulting fourth highest ranked complex, when visualized, shows two clustered components and represents two protein complexes, the proteasome and an RNA processing complex, both found in the nucleus (Figure 40). This is an example of where a lower VWP parameter would have been superior since it would have divided this large complex into two more functionally related complexes. The highest weighted vertices in the center of each of the two dense regions in Figure 40 are the Rpt1 and Lsm4 proteins. MCODE was run in directed mode starting with these two proteins over a range of VWP parameters from 0 to 0.2, at 0.05 increments. For Lsm4, the parameter set of haircut=TRUE, fluff=FALSE, VWP=0 was used to find a core complex, which contained 9 proteins fully connected to each other (Dcp1, Kem1, Lsm2, Lsm3, Lsm4, Lsm5, Lsm6, Lsm7 and Pat1). Above this VWP parameter, the core complex branched out into proteasome subunit proteins, which are not part of the Lsm complex (see Figure 41A). Using this VWP parameter, combinations of haircut and fluff parameters were used to further expand the core complex. This process was stopped when the predicted complexes began to include proteins of sufficiently different known biological function to the seed vertex. Proteins, such as Vam6 and Yor320c were included in the complex at moderate fluff parameters (0.4-0.6), but not at higher fluff parameters, and these are known to be localized in membranes outside of the nucleus, thus are likely not functionally related to the Lsm complex proteins. Therefore, the 9 proteins listed above

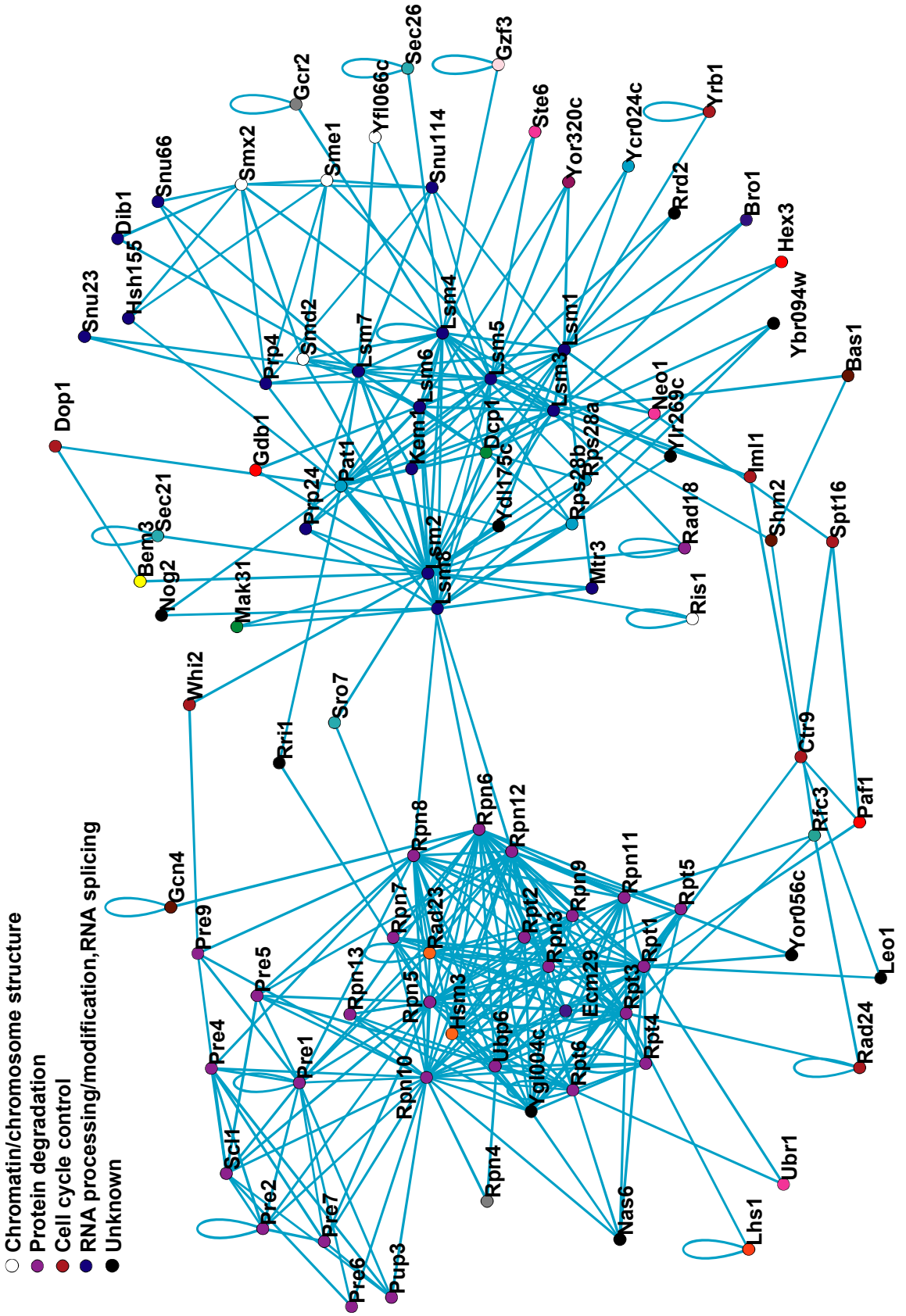
were decided to be the final complex (Figure 41B). This is intuitive because of their maximal density (a 9-clique).

Using this same method of known biological role "titration" on Rpt1 found a complex of 34 proteins (Gal4, Gcn4, Hsm3, Lhs1, Nas6, Pre1, Pre2, Pre3, Pre4, Pre5, Pre6, Pre7, Pre9, Pup3, Rpn10, Rpn11, Rpn13, Rpn3, Rpn5, Rpn6, Rpn7, Rpn8, Rpn9, Rpt1, Rpt2, Rpt3, Rpt4, Rpt6, Rri1, Scl1, Sts1, Ubp6, Ydr179c, Ygl004c) and 160 interactions using the parameter set haircut=TRUE, fluff=TRUE, VWP=0.2 and a fluff density threshold of 0.3. Two regions of density can be seen here corresponding to the two known subunits of the 26S proteasome. The 20S proteolytic subunit of the proteasome is comprised of 15 proteins (Pre1 to Pre10, Pup1, Pup2, Pup3, Scl1 and Ump1) of which Pre7, Pre8, Pre10, Pup1, Pup2 and Ump1 are not found with MCODE. The 19S regulatory subunit of the proteasome is known to have 21 subunits (Nas6, Rpn1 to Rpn13, Rpt1 to Rpt6 and Ubp6) of which Rpn1, Rpn2, Rpn4, Rpn12 and Rpt5 are not found with MCODE. Known complex components not found by MCODE are not present at a high enough local density regions of the interaction network, possibly because not enough experiments involving these proteins are present in our data set. Figure 41C shows the final Rpt1 seeded complex. Of note, Ygl004c is unknown and binds to almost every Rpt and Rpn protein in the complex although all of these interactions were from a single immunoprecipitation experiment (Ho et al., 2002). As well, Rri1 and Ydr179c have unknown function and both bind to each other and to Rpn5. Thus one would predict that these three unknown proteins function with or as part of the 26S proteasome. The protein Hsm3 binds to eight other 19S subunits and is involved in DNA mismatch repair pathways, but is not known to be part of the proteasome, although all of these Hsm3 interactions are from a particular large-scale experiment (Gavin et al., 2002). Interestingly, Gal4, a transcription factor involved in galactose metabolism, is found to be part of the proteasome complex. While this metabolic functionality seems unrelated to protein degradation, it has recently been shown that the binding is physiologically relevant (Gonzalez et al., 2002). These cases illustrate the possible unreliability of both functional annotation and interaction data, but also that seemingly unrelated proteins should not be immediately discounted if found to be part of a complex by MCODE.

Of note, the known topology of the 26S proteasome (Bochtler et al., 1999) compares favorably with the complex visualization of Figure 41C without taking into account stoichiometry. Thus, if enough interactions are known, visualizing complexes may reveal the rough structural outline of large complexes. This should be expected when dealing with actual physical protein-protein interactions since there are few allowed topologies for large complexes considering the specific set of defining interactions and steric clashes between protein subunits.

Figure 40: An MCODE Predicted Complex That is Too Large (Relaxed Parameters)

An example of a predicted complex that incorporates two complexes, proteasome (left) and an RNA processing complex (right). These should probably be predicted as separate complexes as can be seen by the clear distinction of biological role annotation on one side of this layout compared to the other (purple versus blue). This figure, however, shows the large amount of overall connectivity between these two complexes. This complex was ranked fourth by MCODE from the predicted complexes in the AllYeast interaction set with slightly relaxed parameters compared to the optimized prediction.


Chromatin/chromosome structure
Protein degradation
Cell cycle control
RNA processing/modification,RNA splicing
Unknown
Gcr2
Sec26
Gzf3
Yfl066c
Sme1
Smx2
Snu66
Snu114
Ste6
Yor320c
Ycr024c
Yrb1
Dib1
Rrd2
Bro1
Hsh155
Hex3
Snu23
Ybr094w
Smd2
Lsm4
Lsm7
Lsm1
Rrp4
Lsm6
Lsm5
Lsm3
Bas1
Dop1
Neo1
Lsm1
Dcp1
Yir269c
Gdb1
Pat1
Rps28b
Rps28a
Iml1
Ydl175c
Rrp24
Sec21
Lsm2
Rad18
Shm2
Spt16
Bem3
Lsm8
Mtr3
Nog2
Mak31
Ris1
Whi2
Sro7
Rri1
Ctr9
Rpn6
Rpn8
Rpn12
Rfc3
Paf1
Gcn4
Rpn11
Rpn7
Rpn9
Rpt5
Rad23
Rpt2
Rpn3
Rpt1
Yor056c
Pre9
Pre5
Rpn13
Rpn5
Hsm3
Ecm29
Rpt3
Leo1
Pre4
Pre1
Ubp6
Rpt6
Rpt4
Rad24
Rpn10
Ygl004c
Scl1
Ubr1
Rpn4
Pre2
Pre7
Nas6
Lhs1
Pre6
Pup3

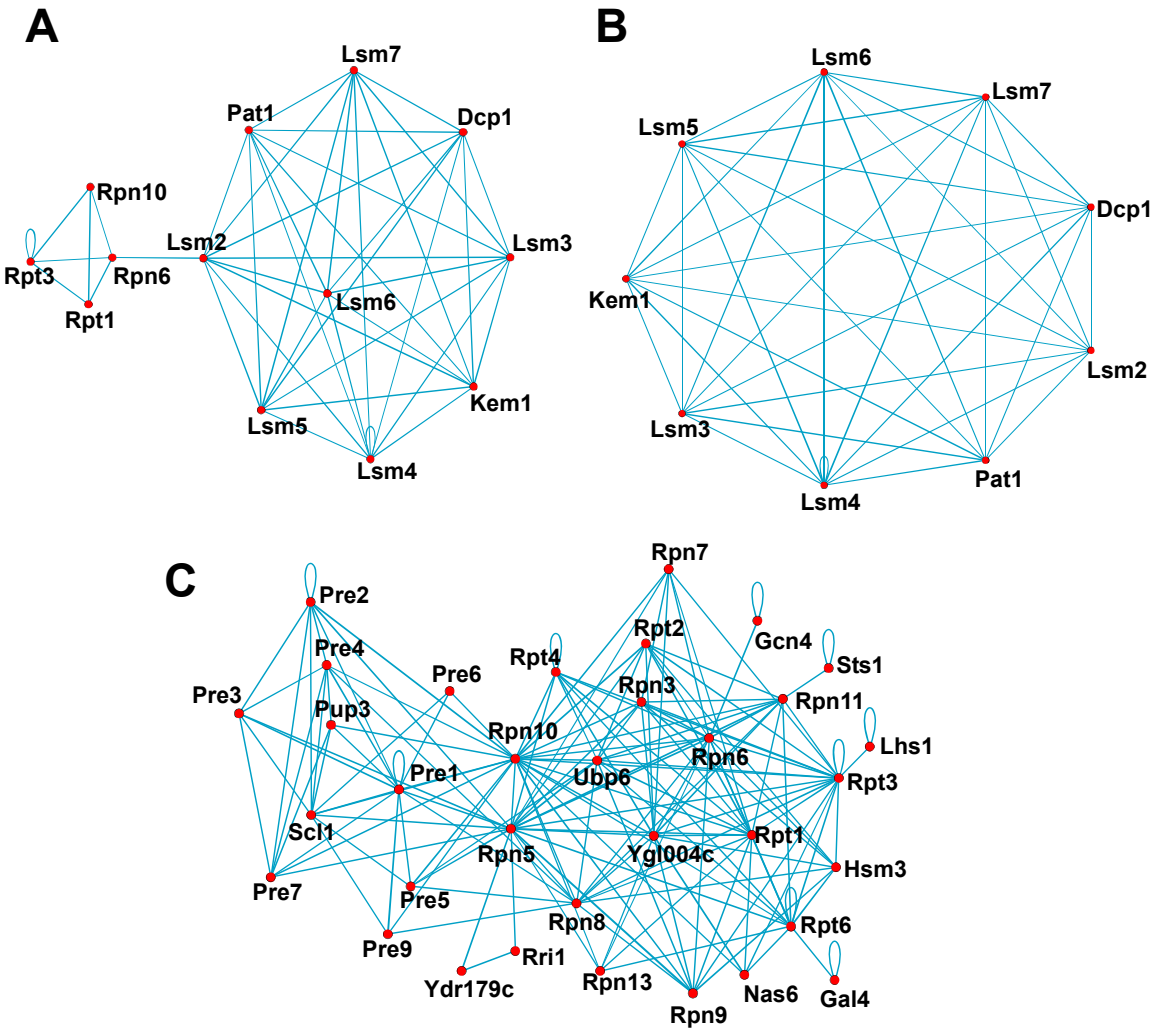


Figure 41: MCODE in Directed Mode

MCODE was used in directed mode to further study the complex in Figure 40 by using seed vertices from high density regions of the two parts of this complex. A) The result of examining the Lsm complex using MCODE parameters that are too relaxed (haircut=TRUE, fluff=FALSE, VWP=0.05). B) The final Lsm complex using MCODE parameters of haircut=TRUE, fluff=FALSE and VWP=0 seeded with Lsm4. C) The final 26S proteasome complex seeded with Rpt1 using MCODE parameters haircut=TRUE, fluff=TRUE and VWP=0.2. Visible here are two regions of density in this complex corresponding to the 20S proteolytic subunit (left side - mainly Pre proteins) and the 19S regulatory subunit (right side - mainly Rpt and Rpn proteins).

Complex Connectivity

MCODE may also be used to examine the connectivity and relationships between molecular complexes. Once a complex is known using the directed mode, the MCODE parameters can be relaxed to allow branching out into other complexes. The MCODE directed mode preprocessing step must also be turned off to allow MCODE to branch into other connected complexes, which may reside in denser regions of the graph than the seed vertex. As an example, this was done with the Lsm4 seeded complex (Figure 42). MCODE parameters were relaxed to haircut=TRUE, fluff=FALSE, VWP=0.2 although they could be further relaxed for greater extension out into the network.

Figure 42: Examining Complex Connectivity with MCODE

The complexes shown here are known to be nuclear localized and are involved in protein degradation (19S proteasome subunit), mRNA processing (Lsm complex and mRNA Cleavage/Polyadenylation complex), cell cycle (anaphase promoting complex) and transcription (SAGA transcriptional activation complex).

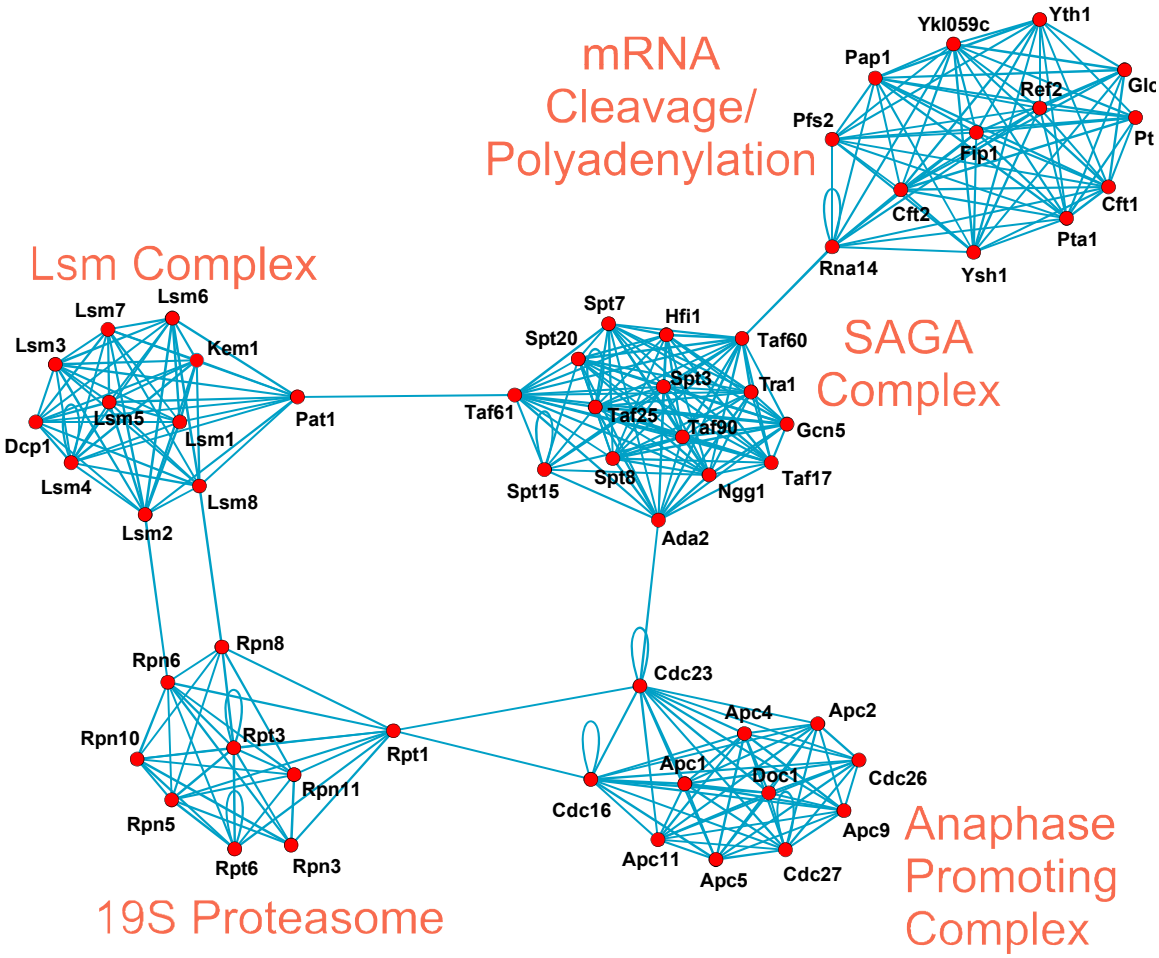

mRNA
Cleavage/
Polyadenylation
Ykl059c
Yth1
Pap1
Ref2
Glc
Pfs2
Pt
Fip1
Cft2
Cft1
Pta1
Rna14
Ysh1
Lsm Complex
Lsm7
Lsm6
Lsm3
Kem1
Lsm5
Pat1
Dcp1
Lsm1
Lsm4
Lsm8
Lsm2
Spt7
Hfi1
Spt20
Taf60
SAGA
Complex
Spt3
Tra1
Taf61
Taf25
Taf90
Gcn5
Spt15
Spt8
Ngg1
Taf17
Ada2
Rpn8
Rpn6
Cdc23
Apc4
Apc2
Rpn10
Rpt3
Rpt1
Apc1
Doc1
Cdc26
Rpn11
Cdc16
Rpn5
Apc9
Apc11
Apc5
Cdc27
Rpt6
Rpn3
Anaphase
Promoting
Complex
19S Proteasome

Figure 42

Discussion

This method represents an initial step in taking advantage of the protein function data being generated by many large-scale protein interaction studies. As the experimental methods are further developed, an increasing amount of data will be produced which will require computational methods for efficient interpretation. The algorithm described here allows the automated prediction of protein complexes from qualitative protein-protein interaction data and is thus able to help predict the function of unknown proteins and aid in the understanding of the functional connectivity of molecular complexes in the cell. The general nature of this method may allow complex prediction for molecules other than proteins as well, for example metabolic complexes that include small molecules.

MCODE cannot stand alone in this task; it must be combined with a graph visualization system to ease the understanding of the relationships among molecules in the data set. The Pajek program for large network analysis (Batagelj and Mrvar, 1998) is used with the Kamada-Kawai graph layout algorithm (Kamada and Kawai, 1989). Kamada-Kawai models the edges in the graph as springs, randomly places the vertices in a high energy state and then attempts to minimize the energy of the system over a number of time steps. The result is that the Euclidean distance, here in a plane, is close to the graph-theoretic or path distance between the vertices. The vertices are visually clustered based on connectivity. Biologically, this visualization can allow one to see the rough structural outline of large complexes, if enough interactions are known, as evidenced in the proteasome complex analysis above (Figure 41C).

It is important to note and understand the limitations of the current experimental methods (e.g. yeast two-hybrid and co-immunoprecipitation) and the protein interaction networks that these techniques generate when analyzing the resulting data. One common class of false-positive interactions arising from many different kinds of experimental methods is that of indirect interactions. For instance, an interaction may be seen between two proteins using a specific experimental method, but in reality, those proteins do not physically bind each other, and one or more other molecules that are generally part of the same complex mediate the observed interaction. As can be seen for the Arp2/3 complex

shown in Figure 34B,C, when pairwise interactions between all combinations of proteins in a complex are studied, this creates a very dense graph. Interestingly, this false-positive effect is normally considered a disadvantage, but is an advantage with MCODE as it increases the density in the region of the graph containing a complex, which can then be more easily predicted.

Apart from the experimental factors that lead to false-positive and false-negative interactions, representational limitations also exist computationally. Temporal and spatial information is not currently described in interaction networks. A complex found by the MCODE approach may not actually exist even though all of the component proteins bind each other *in vitro*. Those proteins may never be present at the same time and place. For example, molecular complexes that perform different functions sometimes have common subunits as with the three types of eukaryotic RNA polymerases.

Complex stoichiometry, another important aspect of biological data, is not represented either. While it is possible to include full stoichiometry in a graph representation of a biomolecular interaction network, many experimental methods do not provide this information, so a homo-multimeric complex is normally represented as a simple homodimer. When an experiment does provide stoichiometry information, it is not stored in most current databases, such as MIPS and YPD. Thus, one is forced to return to the primary literature to extract the data, an extremely time-consuming task for large data sets.

Some quantitative and statistical information is present when integrating results of large-scale approaches and this is not used in our current graph model. For instance, the number of different types of experiments that find the same interaction, the quality of the experiment, the date the experiment was conducted (newer methods may be superior in certain aspects) and other factors that pertain to the reliability of the interaction could all be considered to determine a reliability index or p-value on edges in the graph. For instance, one may wish to rank results published in high-impact journals above other journals and rank classical purification methods above high-throughput yeast two-hybrid techniques when determining the quality of the interaction data. It may also be possible to weight vertices on the graph by other quality criteria, such as whether a protein is

hypothetical from a gene prediction or not or whether a protein is expressed at a particular time and place in the cell. For example, if one were interested in a certain stage of the cell cycle, proteins that are known to be absent at that stage could be reduced in weight (VWP in the case of MCODE) compared to proteins that are present. It should be noted that any weighting scheme that tries to assess the quality of an interaction might make false assumptions that would prevent the discovery of new and interesting data.

This paper shows that the structure of a biological network can define complexes, which can be seen as dense regions. This may be attributed to indirect interactions accumulating in the literature. Thus, interaction data taken out of context may be erroneous. For instance, if one has a collection of protein interactions from various different experiments done at different times in different labs from a specific complex that form a clique, and if one chooses an interaction from this clique, then how can one verify if it is indirect or not. We would only begin to know if we had a very detailed description of the experiment from the original papers where we could tell the amount of work and quality of work that went into measuring each interaction. Thus with only a qualitative view of interactions, in reference to Dobzhansky (Dobzhansky, 1973), nothing in the biomolecular interaction network would make sense except in light of molecular complexes and the functional connections between them. If one had a highly detailed representation of each interaction including time, place, experimental condition, number of experiments, binding sites, chemical actions and chemical state information, one would be able to computationally delve into molecular complexes to resolve topology, structure, function and mechanism down to the atomic level. This information would also help to judge the biological relevance of an interaction. Thus, we require databases like BIND (Bader et al., 2001) to store this information. The integration of known qualitative and quantitative molecular interaction data in a machine-readable format should allow increasingly accurate protein interaction, molecular complex and pathway prediction, including actual binding site and mechanism information in a sequence and structural context.

Based on the scale-free network analysis, it would seem that real biological networks are organized differently than random models of scale-free networks in that they have higher clustering coefficients around specific regions (complexes) and the

vertices in these regions are related to each other, by biological function. Thus, attempts to model biological networks and their evolution in a global way solely using the statistics of scale-free networks may not work, rather modeling should take into account as much extant biological knowledge as possible.

Future work on MCODE could include researching different, possibly adaptive, vertex scoring functions to take into account, for example, the local density of the network past the immediate neighborhood of a vertex and the inclusion of functional annotation and p-values on edges. Time, space and stoichiometry should also be represented on networks and in visualization systems. The process of 'functional annotation titration' in the directed mode of MCODE could be automated.

Conclusions

MCODE effectively finds densely connected regions of a molecular interaction network, many of which correspond to known molecular complexes, based solely on connectivity data. Given that this approach to analyzing protein interaction networks performs well using minimal qualitative information implies that large amounts of available knowledge is buried in large protein interaction networks. More accurate data mining algorithms and systems models could be constructed to understand and predict interactions, complexes and pathways by taking into account more existing biological knowledge. Structured molecular interaction data resources such as BIND will be vital in creating these resources.

Materials and Methods

Data Sources

All protein interaction data sets from MIPS (Mewes et al., 2000), Gene Ontology (Dwight et al., 2002; The Gene Ontology Consortium, 2000) and PreBIND (http://bioinfo.mshri.on.ca/prebind/) were collected as described previously (Ho et al.,

2002). The YPD protein interaction data are from March 2001 and were originally requested from Proteome, Inc. (http://www.proteome.com). Other interaction data sets are from BIND (http://www.bind.ca). A BIND yeast import utility was developed to integrate data from SGD (Chervitz et al., 1999), RefSeq (Pruitt and Maglott, 2001), Gene Registry (http://genome-www.stanford.edu/Saccharomyces/registry.html), the list of essential genes from the yeast deletion consortium (Winzeler et al., 1999) and GO terms (Dwight et al., 2002; The Gene Ontology Consortium, 2000). This database ensures proper matching of yeast gene names among the multiple data sets that may use different names for the same genes. The yeast proteome used here is defined by SGD and RefSeq and contains 6,334 ORFs including the mitochondrial chromosome. Before performing comparisons, the various interaction data sets were entered into a local instance of BIND as pairwise protein interaction records. The MIPS complex catalogue was downloaded in February 2002.

The protein interaction data sets used here were composed as follows. 'Gavin Spoke' is the spoke model of the raw purifications from Gavin et al. (Gavin et al., 2002). 'Y2H' is all known large-scale (Drees et al., 2001; Fromont-Racine et al., 2000; Ito et al., 2001; Tong et al., 2002; Uetz et al., 2000) combined with normal yeast two-hybrid results from MIPS. 'HTP Only' is only high-throughput or large-scale data (Drees et al., 2001; Fromont-Racine et al., 2000; Gavin et al., 2002; Ho et al., 2002; Ito et al., 2001; Tong et al., 2002; Uetz et al., 2000). The 'Benchmark' set was constructed from MIPS, YPD and PreBIND as previously described (Ho et al., 2002). 'Pre HTMS' was composed of all yeast sets except the recent large-scale mass spectrometry data sets (Gavin et al., 2002; Ho et al., 2002). 'AllYeast' was the combination of all above data sets. All data sets are non-redundant.

Network Visualization

Visualization of networks was performed using the Pajek program for large network analysis (Batagelj and Mrvar, 1998) (http://vlado.fmf.uni-lj.si/pub/networks/pajek/) as described previously (Ho et al., 2002; Tong et al., 2002)

using the Kamada-Kawai graph layout algorithm followed by manual node adjustments and was formatted using CorelDraw 10. Power law analysis was also accomplished as previously described (Ho et al., 2002).

References

1. Aasland R, Abrams C, Ampe C, Ball LJ, Bedford MT, Cesareni G *et al*.: **Normalization of nomenclature for peptide motifs as ligands of modular protein domains.** *FEBS Lett* 2002, **513:** 141-144.

2. Albert R, Jeong H, Barabasi AL: **Error and attack tolerance of complex networks.** *Nature* 2000, **406:** 378-382.

3. Altschul SF, Madden TL, Schaffer AA, Zhang J, Zhang Z, Miller W *et al*.: **Gapped BLAST and PSI-BLAST: a new generation of protein database search programs.** *Nucleic Acids Res* 1997, **25:** 3389-3402.

4. Andersen JS, Lyon CE, Fox AH, Leung AK, Lam YW, Steen H *et al*.: **Directed proteomic analysis of the human nucleolus.** *Curr Biol* 2002, **12:** 1-11.

5. Bader GD, Donaldson I, Wolting C, Ouellette BF, Pawson T, Hogue CW: **BIND-The biomolecular interaction network database.** *Nucleic Acids Res* 2001, **29:** 242-245.

6. Bader GD, Hogue CW: **BIND-a data specification for storing and describing biomolecular interactions, molecular complexes and pathways.** *Bioinformatics* 2000, **16:** 465-477.

7. Bairoch A: **The ENZYME database in 2000.** *Nucleic Acids Res* 2000, **28:** 304-305.

8. Bairoch A, Apweiler R: **The SWISS-PROT protein sequence database and its supplement TrEMBL in 2000.** *Nucleic Acids Res* 2000, **28:** 45-48.

9. Baisnee PF, Pollastri G, Pecout Y, Nowick J, Baldi P. ICBS: A Database of Protein-Protein Interactions Mediated by Interchain Beta-Sheet Formation. 10th International Conference on Intelligent Systems for Molecular Biology (ISMB) . 2002.
   Ref Type: Abstract

10. Baldi P, Brunak S, Chauvin Y, Andersen CA, Nielsen H: **Assessing the accuracy of prediction algorithms for classification: an overview.** *Bioinformatics* 2000, **16:** 412-424.

11. Ball CA, Dolinski K, Dwight SS, Harris MA, Issel-Tarver L, Kasarskis A *et al*.: **Integrating functional genomic information into the Saccharomyces genome database.** *Nucleic Acids Res* 2000, **28:** 77-80.

12. Barabasi AL, Albert R: **Emergence of scaling in random networks.** *Science* 1999, **286:** 509-512.

13. Baranov PV, Kubarenko AV, Gurvich OL, Shamolina TA, Brimacombe R: **The Database of Ribosomal Cross-links: an update.** *Nucleic Acids Res* 1999, **27:** 184-185.

14. Barstead R: **Genome-wide RNAi.** *Curr Opin Chem Biol* 2001, **5:** 63-66.

15. Batagelj V, Mrvar A: **Pajek - Program for Large Network Analysis.** *Connections* 1998, **2:** 47-57.

16. Bender A, Pringle JR: **Use of a screen for synthetic lethal and multicopy suppressee mutants to identify two new genes involved in morphogenesis in Saccharomyces cerevisiae.** *Mol Cell Biol* 1991, **11:** 1295-1305.

17. Benson DA, Karsch-Mizrachi I, Lipman DJ, Ostell J, Rapp BA, Wheeler DL: **GenBank.** *Nucleic Acids Res* 2002, **30:** 17-20.

18. Berman HM, Westbrook J, Feng Z, Gilliland G, Bhat TN, Weissig H *et al*.: **The Protein Data Bank.** *Nucleic Acids Res* 2000, **28:** 235-242.

19. Bernstein FC, Koetzle TF, Williams GJ, Meyer EFJ, Brice MD, Rodgers JR *et al*.: **The protein data bank: a computer-based archival file for macromolecular structures.** *Arch Biochem Biophys* 1978, **185:** 584-591.

20. Blackstock WP, Weir MP: **Proteomics: quantitative and physical mapping of cellular proteins.** *Trends Biotechnol* 1999, **17:** 121-127.

21. Blythe MJ, Doytchinova IA, Flower DR: **JenPep: a database of quantitative functional peptide data for immunology.** *Bioinformatics* 2002, **18:** 434-439.

22. Bochtler M, Ditzel L, Groll M, Hartmann C, Huber R: **The proteasome.** *Annu Rev Biophys Biomol Struct* 1999, **28:** 295-317.

23. Brusic V, Rudy G, Harrison LC: **MHCPEP, a database of MHC-binding peptides: update 1997.** *Nucleic Acids Res* 1998, **26:** 368-371.

24. Cassman M, Hunter T, Pawson T: **Proteins suggest form of their own database.** *Nature* 2000, **403:** 591-592.

25. Castagnetto JM, Hennessy SW, Roberts VA, Getzoff ED, Tainer JA, Pique ME: **MDB: the Metalloprotein Database and Browser at The Scripps Research Institute.** *Nucleic Acids Res* 2002, **30:** 379-382.

26. Chen X, Lin Y, Gilson MK: **The binding database: Overview and user's guide.** *Biopolymers* 2001a, **61:** 127-141.

27. Chen X, Lin Y, Liu M, Gilson MK: **The Binding Database: data management and interface design.** *Bioinformatics* 2002, **18:** 130-139.

28. Chen X, Liu M, Gilson MK: **BindingDB: a web-accessible molecular recognition database.** *Comb Chem High Throughput Screen* 2001b, **4:** 719-725.

29. Chervitz SA, Aravind L, Sherlock G, Ball CA, Koonin EV, Dwight SS *et al*.: **Comparison of the complete protein sets of worm and yeast: orthology and divergence.** *Science* 1998, **282:** 2022-2028.

30. Chervitz SA, Hester ET, Ball CA, Dolinski K, Dwight SS, Harris MA *et al*.: **Using the Saccharomyces Genome Database (SGD) for analysis of protein similarities and structure.** *Nucleic Acids Res* 1999, **27:** 74-78.

31. Christendat D, Yee A, Dharamsi A, Kluger Y, Savchenko A, Cort JR *et al*.: **Structural proteomics of an archaeon.** *Nat Struct Biol* 2000, **7:** 903-909.

32. Colwill K, Field D, Moore L, Friesen J, Andrews B: **In vivo analysis of the domains of yeast Rvs167p suggests Rvs167p function is mediated through multiple protein interactions.** *Genetics* 1999, **152:** 881-893.

33. Costanzo MC, Crawford ME, Hirschman JE, Kranz JE, Olsen P, Robertson LS *et al*.: **YPD, PombePD and WormPD: model organism volumes of the BioKnowledge library, an integrated resource for protein information.** *Nucleic Acids Res* 2001, **29:** 75-79.

34. Costanzo MC, Hogan JD, Cusick ME, Davis BP, Fancher AM, Hodges PE *et al*.: **The yeast proteome database (YPD) and Caenorhabditis elegans proteome database (WormPD): comprehensive resources for the organization and comparison of model organism protein information.** *Nucleic Acids Res* 2000, **28:** 73-76.

35. Crasto C, Marenco L, Miller P, Shepherd G: **Olfactory Receptor Database: a metadata-driven automated population from sources of gene and protein sequences.** *Nucleic Acids Res* 2002, **30:** 354-360.

36. DDBJ/EMBL/GenBank: *The DDBJ/EMBL/GenBank Feature Table Definition Version 2.1*. 1997.

37. Demir E, Babur O, Dogrusoz U, Gursoy A, Nisanci G, Cetin-Atalay R *et al*.: **PATIKA: an integrated visual environment for collaborative construction and analysis of cellular pathways.** *Bioinformatics* 2002, **18:** 996-1003.

38. Dobzhansky T: **Nothing in Biology Makes Sense Except in the Light of Evolution.** *American Biology Teacher* 1973, **35:** 125-129.

39. Drees BL, Sundin B, Brazeau E, Caviston JP, Chen GC, Guo W *et al*.: **A protein interaction map for cell polarity development.** *J Cell Biol* 2001, **154:** 549-571.

40. Dutt MJ, Lee KH: **Proteomic analysis.** *Curr Opin Biotechnol* 2000, **11:** 176-179.

41. Dwight SS, Harris MA, Dolinski K, Ball CA, Binkley G, Christie KR *et al.*: **Saccharomyces Genome Database (SGD) provides secondary gene annotation using the Gene Ontology (GO).** *Nucleic Acids Res* 2002, **30:** 69-72.

42. Eeckman FH, Durbin R: **ACeDB and macace.** *Methods Cell Biol* 1995, **48:** 583-605.

43. Eilbeck K, Brass A, Paton N, Hodgman C. INTERACT: an object oriented protein-protein interaction database. Ismb. 87-94. 1999.
Ref Type: Abstract

44. Eisen MB, Spellman PT, Brown PO, Botstein D: **Cluster analysis and display of genome-wide expression patterns.** *Proc Natl Acad Sci U S A* 1998, **95:** 14863-14868.

45. Eisenberg D, Marcotte EM, Xenarios I, Yeates TO: **Protein function in the post-genomic era.** *Nature* 2000, **405:** 823-826.

46. Ellis LB, Hershberger CD, Bryan EM, Wackett LP: **The University of Minnesota Biocatalysis/Biodegradation Database: emphasizing enzymes.** *Nucleic Acids Res* 2001, **29:** 340-343.

47. Evangelista M, Klebl BM, Tong AH, Webb BA, Leeuw T, Leberer E *et al.*: **A role for myosin-I in actin assembly through interactions with Vrp1p, Bee1p, and the Arp2/3 complex.** *J Cell Biol* 2000, **148:** 353-362.

48. Feldman HJ, Hogue CW: **Probabilistic sampling of protein conformations: new hope for brute force?** *Proteins* 2002, **46:** 8-23.

49. Fell DA, Wagner A: **The small world of metabolism.** *Nat Biotechnol* 2000, **18:** 1121-1122.

50. Ficarro SB, McCleland ML, Stukenberg PT, Burke DJ, Ross MM, Shabanowitz J *et al.*: **Phosphoproteome analysis by mass spectrometry and its application to Saccharomyces cerevisiae.** *Nat Biotechnol* 2002, **20:** 301-305.

51. Fields S: **Proteomics. Proteomics in genomeland.** *Science* 2001, **291:** 1221-1224.

52. Fields S, Song O: **A novel genetic system to detect protein-protein interactions.** *Nature* 1989, **340:** 245-246.

53. Flake GW, Lawrence S, Giles CL, Coetzee FM: **Self-Organization of the Web and Identification of Communities.** *IEEE Computer* 2002, **35:** 66-71.

54. Fromont-Racine M, Mayes AE, Brunet-Simon A, Rain JC, Colley A, Dix I *et al.*: **Genome-wide protein interaction screens reveal functional networks involving Sm-like proteins.** *Yeast* 2000, **17:** 95-110.

55. Fromont-Racine M, Rain JC, Legrain P: **Toward a functional analysis of the yeast genome through exhaustive two- hybrid screens.** *Nat Genet* 1997, **16:** 277-282.

56. Gasteiger J: **Chemical Information in 3D-Space.** *J Chem Inf Comput Sci* 1996, **36:** 1030-1037.

57. Gavin AC, Bosche M, Krause R, Grandi P, Marzioch M, Bauer A *et al*.: **Functional organization of the yeast proteome by systematic analysis of protein complexes.** *Nature* 2002, **415:** 141-147.

58. Ge H, Liu Z, Church GM, Vidal M: **Correlation between transcriptome and interactome mapping data from Saccharomyces cerevisiae.** *Nat Genet* 2001, **29:** 482-486.

59. Gerstein M, Lan N, Jansen R: **Proteomics. Integrating interactomes.** *Science* 2002, **295:** 284-287.

60. Ghosh D: **Object-oriented transcription factors database (ooTFD).** *Nucleic Acids Res* 2000, **28:** 308-310.

61. Goldberg AV: **Finding a Maximum Density Subgraph.** *Technical Report UCB/CSD University of California, Berkeley, CA* 1984, **84**.

62. Gonzalez F, Delahodde A, Kodadek T, Johnston SA: **Recruitment of a 19S proteasome subcomplex to an activated promoter.** *Science* 2002, **296:** 548-550.

63. Gough NR, Ray LB: **Mapping cellular signaling.** *Sci STKE* 2002, **2002:** EG8.

64. Guarente L: **Synthetic enhancement in gene interaction: a genetic tool come of age.** *Trends Genet* 1993, **9:** 362-366.

65. Hamosh A, Scott AF, Amberger J, Bocchini C, Valle D, McKusick VA: **Online Mendelian Inheritance in Man (OMIM), a knowledgebase of human genes and genetic disorders.** *Nucleic Acids Res* 2002, **30:** 52-55.

66. Harnpicharnchai P, Jakovljevic J, Horsey E, Miles T, Roman J, Rout M *et al*.: **Composition and functional characterization of yeast 66S ribosome assembly intermediates.** *Mol Cell* 2001, **8:** 505-515.

67. Hartman JL, Garvik B, Hartwell L: **Principles for the buffering of genetic variation.** *Science* 2001, **291:** 1001-1004.

68. Hartuv E, Shamir R: **A clustering algorithm based on graph connectivity.** *Information processing letters* 1999, **76:** 175-181.

69. Hendlich M: **Databases for protein-ligand complexes.** *Acta Crystallogr D Biol Crystallogr* 1998, **54:** 1178-1182.

70. Henikoff JG, Greene EA, Pietrokovski S, Henikoff S: **Increased coverage of protein families with the blocks database servers.** *Nucleic Acids Res* 2000, **28:** 228-230.

71. Higgins DG, Thompson JD, Gibson TJ: **Using CLUSTAL for multiple sequence alignments.** *Methods Enzymol* 1996, **266:** 383-402.

72. Ho Y, Gruhler A, Heilbut A, Bader GD, Moore L, Adams SL *et al*.: **Systematic identification of protein complexes in Saccharomyces cerevisiae by mass spectrometry.** *Nature* 2002, **415:** 180-183.

73. Hofmann K, Bucher P, Falquet L, Bairoch A: **The PROSITE database, its status in 1999.** *Nucleic Acids Res* 1999, **27:** 215-219.

74. Hogue CW: **Cn3D: a new generation of three-dimensional molecular structure viewer.** *Trends Biochem Sci* 1997, **22:** 314-316.

75. Hogue CW, Ohkawa H, Bryant SH: **A dynamic look at structures: WWW-Entrez and the Molecular Modeling Database.** *Trends Biochem Sci* 1996, **21:** 226-229.

76. Ideker T, Thorsson V, Ranish JA, Christmas R, Buhler J, Eng JK *et al*.: **Integrated genomic and proteomic analyses of a systematically perturbed metabolic network.** *Science* 2001, **292:** 929-934.

77. Igarashi T, Kaminuma T: **Development of a cell signaling networks database.** *Pac Symp Biocomput* 1997, 187-197.

78. Ito T, Chiba T, Ozawa R, Yoshida M, Hattori M, Sakaki Y: **A comprehensive two-hybrid analysis to explore the yeast protein interactome.** *Proc Natl Acad Sci U S A* 2001, **98:** 4569-4574.

79. Ito T, Tashiro K, Muta S, Ozawa R, Chiba T, Nishizawa M *et al*.: **Toward a protein-protein interaction map of the budding yeast: A comprehensive system to examine two-hybrid interactions in all possible combinations between the yeast proteins.** *Proc Natl Acad Sci U S A* 2000, **97:** 1143-1147.

80. Jeong H, Mason SP, Barabasi AL, Oltvai ZN: **Lethality and centrality in protein networks.** *Nature* 2001, **411:** 41-42.

81. Jeong H, Tombor B, Albert R, Oltvai ZN, Barabasi AL: **The large-scale organization of metabolic networks.** *Nature* 2000, **407:** 651-654.

82. Jones S, Thornton JM: **Principles of protein-protein interactions.** *Proc Natl Acad Sci U S A* 1996, **93:** 13-20.

83. Kamada T, Kawai S: **An algorithm for drawing general indirect graphs.** *Information processing letters* 1989, **31:** 7-15.

84. Kanehisa M, Goto S, Kawashima S, Nakaya A: **The KEGG databases at GenomeNet.** *Nucleic Acids Res* 2002, **30:** 42-46.

85. Kans JA, Ouellette BF: **Submitting DNA Sequences to the Databases.** In *Bioinformatics*. Edited by Baxevanis AD, Ouellette BF. Toronto: John Wiley & Sons; 1998:319-353.

86. Karp PD: **An ontology for biological function based on molecular interactions.** *Bioinformatics* 2000, **16:** 269-285.

87. Karp PD, Riley M, Paley SM, Pellegrini-Toole A: **The MetaCyc Database.** *Nucleic Acids Res* 2002a, **30:** 59-61.

88. Karp PD, Riley M, Paley SM, Pellegrini-Toole A, Krummenacker M: **Eco Cyc: encyclopedia of Escherichia coli genes and metabolism.** *Nucleic Acids Res* 1999, **27:** 55-58.

89. Karp PD, Riley M, Saier M, Paulsen IT, Collado-Vides J, Paley SM *et al*.: **The EcoCyc Database.** *Nucleic Acids Res* 2002b, **30:** 56-58.

90. Karp PD, Riley M, Saier M, Paulsen IT, Paley SM, Pellegrini-Toole A: **The EcoCyc and MetaCyc databases.** *Nucleic Acids Res* 2000, **28:** 56-59.

91. Kel-Margoulis OV, Romashchenko AG, Kolchanov NA, Wingender E, Kel AE: **COMPEL: a database on composite regulatory elements providing combinatorial transcriptional regulation.** *Nucleic Acids Res* 2000, **28:** 311-315.

92. Kim SK, Lund J, Kiraly M, Duke K, Jiang M, Stuart JM *et al*.: **A gene expression map for Caenorhabditis elegans.** *Science* 2001, **293:** 2087-2092.

93. Kohn KW: **Molecular interaction map of the mammalian cell cycle control and DNA repair systems.** *Mol Biol Cell* 1999, **10:** 2703-2734.

94. Kolchanov NA, Ignatieva EV, Ananko EA, Podkolodnaya OA, Stepanenko IL, Merkulova TI *et al*.: **Transcription Regulatory Regions Database (TRRD): its status in 2002.** *Nucleic Acids Res* 2002, **30:** 312-317.

95. Kolpakov FA, Ananko EA: **Interactive data input into the GeneNet database.** *Bioinformatics* 1999, **15:** 713-714.

96. Kolpakov FA, Ananko EA, Kolesov GB, Kolchanov NA: **GeneNet: a gene network database and its automated visualization.** *Bioinformatics* 1998, **14:** 529-537.

97. Korber B, Brander C, Haynes B, Koup R, Moore J, Walker B: *HIV Molecular Immunology Database 1998*. Theoretical Biology and Biophysics Group, Los Alamos National Laboratory, Los Alamos, NM; 1998.

98. Kreegipuu A, Blom N, Brunak S: **PhosphoBase, a database of phosphorylation sites: release 2.0.** *Nucleic Acids Res* 1999, **27:** 237-239.

99. Lander ES, Linton LM, Birren B, Nusbaum C, Zody MC, Baldwin J *et al*.: **Initial sequencing and analysis of the human genome.** *Nature* 2001, **409:** 860-921.

100. Lechler T, Li R: **In vitro reconstitution of cortical actin assembly sites in budding yeast.** *J Cell Biol* 1997, **138:** 95-103.

101. Letunic I, Goodstadt L, Dickens NJ, Doerks T, Schultz J, Mott R *et al*.: **Recent improvements to the SMART domain-based sequence annotation resource.** *Nucleic Acids Res* 2002, **30:** 242-244.

102. Lila T, Drubin DG: **Evidence for physical and functional interactions among two Saccharomyces cerevisiae SH3 domain proteins, an adenylyl cyclase-associated protein and the actin cytoskeleton.** *Mol Biol Cell* 1997, **8:** 367-385.

103. Lockhart DJ, Winzeler EA: **Genomics, gene expression and DNA arrays.** *Nature* 2000, **405:** 827-836.

104. Madania A, Dumoulin P, Grava S, Kitamoto H, Scharer-Brodbeck C, Soulard A *et al*.: **The Saccharomyces cerevisiae homologue of human Wiskott-Aldrich syndrome protein Las17p interacts with the Arp2/3 complex.** *Mol Biol Cell* 1999, **10:** 3521-3538.

105. Maeda T, Takekawa M, Saito H: **Activation of yeast PBS2 MAPKK by MAPKKKs or by binding of an SH3- containing osmosensor.** *Science* 1995, **269:** 554-558.

106. Mann M, Hendrickson RC, Pandey A: **Analysis of proteins and proteomes by mass spectrometry.** *Annu Rev Biochem* 2001, **70:** 437-473.

107. Marcotte EM, Pellegrini M, Ng HL, Rice DW, Yeates TO, Eisenberg D: **Detecting protein function and protein-protein interactions from genome sequences.** *Science* 1999, **285:** 751-753.

108. Mayer BJ: **SH3 domains: complexity in moderation.** *J Cell Sci* 2001, **114:** 1253-1263.

109. Mayes AE, Verdone L, Legrain P, Beggs JD: **Characterization of Sm-like proteins in yeast and their association with U6 snRNA.** *EMBO J* 1999, **18:** 4321-4331.

110. Mendelsohn AR, Brent R: **Protein interaction methods--toward an endgame.** *Science* 1999, **284:** 1948-1950.

111. Mewes HW, Frishman D, Gruber C, Geier B, Haase D, Kaps A *et al*.: **MIPS: a database for genomes and protein sequences.** *Nucleic Acids Res* 2000, **28:** 37-40.

112. Mewes HW, Frishman D, Guldener U, Mannhaupt G, Mayer K, Mokrejs M *et al*.: **MIPS: a database for genomes and protein sequences.** *Nucleic Acids Res* 2002, **30:** 31-34.

113. Michalickova K, Bader GD, Dumontier M, Lieu H, Betel D, Isserlin R *et al*.: **SeqHound: biological sequence and structure database as a platform for bioinformatics**
research (in press). *BMC Bioinformatics* 2002, **3**.

114. Mohr E, Horn F, Janody F, Sanchez C, Pillet V, Bellon B *et al*.: **FlyNets and GIF-DB, two internet databases for molecular interactions in Drosophila melanogaster.** *Nucleic Acids Res* 1998, **26:** 89-93.

115. Moran MF, Koch CA, Anderson D, Ellis C, England L, Martin GS *et al*.: **Src homology region 2 domains direct protein-protein interactions in signal transduction.** *Proc Natl Acad Sci U S A* 1990, **87:** 8622-8626.

116. Mullen JR, Kaliraman V, Ibrahim SS, Brill SJ: **Requirement for three novel protein complexes in the absence of the Sgs1 DNA helicase in Saccharomyces cerevisiae.** *Genetics* 2001, **157:** 103-118.

117. Nayal M, Hitz BC, Honig B: **GRASS: a server for the graphical representation and analysis of structures.** *Protein Sci* 1999, **8:** 676-679.

118. Neubauer G, Gottschalk A, Fabrizio P, Seraphin B, Luhrmann R, Mann M: **Identification of the proteins of the yeast U1 small nuclear ribonucleoprotein complex by mass spectrometry.** *Proc Natl Acad Sci U S A* 1997, **94:** 385-390.

119. Norris V, Alexandre S, Bouligand Y, Cellier D, Demarty M, Grehan G *et al*.: **Hypothesis: hyperstructures regulate bacterial structure and the cell cycle.** *Biochimie* 1999, **81:** 915-920.

120. Object Management Group: *CORBA Architecture and Specifications*. OMG Publications; 1996.

121. Olson MO, Dundr M, Szebeni A: **The nucleolus: an old factory with unexpected capabilities.** *Trends Cell Biol* 2000, **10:** 189-196.

122. Ostell J, Kans JA: **The NCBI Data Model.** In *Bioinformatics, a Practical Guide to the Analysis of Genes and Proteins*. Edited by Baxevanis AD, Ouellette BF. John Wiley & Sons; 1998:121-144.

123. Overbeek R, Larsen N, Pusch GD, D'Souza M, Selkov EJ, Kyrpides N *et al*.: **WIT: integrated system for high-throughput genome sequence analysis and metabolic reconstruction.** *Nucleic Acids Res* 2000, **28:** 123-125.

124. Pandey A, Mann M: **Proteomics to study genes and genomes.** *Nature* 2000, **405:** 837-846.

125. Paoluzi S, Castagnoli L, Lauro I, Salcini AE, Coda L, Fre' S *et al*.: **Recognition specificity of individual EH domains of mammals and yeast.** *EMBO J* 1998, **17:** 6541-6550.

126. Pawson T: **Protein modules and signalling networks.** *Nature* 1995, **373:** 573-580.

127. Pawson T, Gish GD, Nash P: **SH2 domains, interaction modules and cellular wiring.** *Trends Cell Biol* 2001, **11:** 504-511.

128. Pawson T, Nash P: **Protein-protein interactions define specificity in signal transduction.** *Genes Dev* 2000, **14:** 1027-1047.

129. Pawson T, Scott JD: **Signaling through scaffold, anchoring, and adaptor proteins.** *Science* 1997, **278:** 2075-2080.

130. Perler FB: **InBase, the Intein Database.** *Nucleic Acids Res* 2000, **28:** 344-345.

131. Pieper U, Eswar N, Stuart AC, Ilyin VA, Sali A: **MODBASE, a database of annotated comparative protein structure models.** *Nucleic Acids Res* 2002, **30:** 255-259.

132. Ponomarenko JV, Orlova GV, Frolov AS, Gelfand MS, Ponomarenko MP: **SELEX_DB: a database on in vitro selected oligomers adapted for recognizing natural sites and for analyzing both SNPs and site-directed mutagenesis data.** *Nucleic Acids Res* 2002, **30:** 195-199.

133. Pruitt KD, Maglott DR: **RefSeq and LocusLink: NCBI gene-centered resources.** *Nucleic Acids Res* 2001, **29:** 137-140.

134. Rammensee H, Bachmann J, Emmerich NP, Bachor OA, Stevanovic S: **SYFPEITHI: database for MHC ligands and peptide motifs.** *Immunogenetics* 1999, **50:** 213-219.

135. Ren R, Mayer BJ, Cicchetti P, Baltimore D: **Identification of a ten-amino acid proline-rich SH3 binding site.** *Science* 1993, **259:** 1157-1161.

136. Roberts RJ, Macelis D: **REBASE--restriction enzymes and methylases.** *Nucleic Acids Res* 2001, **29:** 268-269.

137. Robinson RC, Turbedsky K, Kaiser DA, Marchand JB, Higgs HN, Choe S *et al*.: **Crystal structure of Arp2/3 complex.** *Science* 2001, **294:** 1679-1684.

138. Robison K, McGuire AM, Church GM: **A comprehensive library of DNA-binding site matrices for 55 proteins applied to the complete Escherichia coli K-12 genome.** *J Mol Biol* 1998, **284:** 241-254.

139. Rost B, Sander C, Schneider R: **PHD--an automatic mail server for protein secondary structure prediction.** *Comput Appl Biosci* 1994, **10:** 53-60.

140. Salama JJ, Donaldson I, Hogue CW: **Automatic annotation of BIND molecular interactions from three- dimensional structures.** *Biopolymers* 2002, **61:** 111-120.

141. Salcini AE, Confalonieri S, Doria M, Santolini E, Tassi E, Minenkova O *et al*.: **Binding specificity and in vivo targets of the EH domain, a novel protein-protein interaction module.** *Genes Dev* 1997, **11:** 2239-2249.

142. Salcini AE, McGlade J, Pelicci G, Nicoletti I, Pawson T, Pelicci PG: **Formation of Shc-Grb2 complexes is necessary to induce neoplastic transformation by overexpression of Shc proteins.** *Oncogene* 1994, **9:** 2827-2836.

143. Salgado H, Santos-Zavaleta A, Gama-Castro S, Millan-Zarate D, Diaz-Peredo E, Sanchez-Solano F *et al*.: **RegulonDB (version 3.2): transcriptional regulation and operon organization in Escherichia coli K-12.** *Nucleic Acids Res* 2001, **29:** 72-74.

144. Sanchez C, Lachaize C, Janody F, Bellon B, Roder L, Euzenat J *et al*.: **Grasping at molecular interactions and genetic networks in Drosophila melanogaster using FlyNets, an Internet database.** *Nucleic Acids Res* 1999, **27:** 89-94.

145. Schaff J, Loew LM: **The virtual cell.** *Pac Symp Biocomput* 1999, 228-239.

146. Schomburg I, Chang A, Hofmann O, Ebeling C, Ehrentreich F, Schomburg D: **BRENDA: a resource for enzyme data and metabolic information.** *Trends Biochem Sci* 2002a, **27:** 54-56.

147. Schomburg I, Chang A, Schomburg D: **BRENDA, enzyme data and metabolic information.** *Nucleic Acids Res* 2002b, **30:** 47-49.

148. Schonbach C, Koh JL, Sheng X, Wong L, Brusic V: **FIMM, a database of functional molecular immunology.** *Nucleic Acids Res* 2000, **28:** 222-224.

149. Schuler GD, Epstein JA, Ohkawa H, Kans JA: **Entrez: molecular biology database and retrieval system.** *Methods Enzymol* 1996, **266:** 141-162.

150. Schwikowski B, Uetz P, Fields S: **A network of protein-protein interactions in yeast.** *Nat Biotechnol* 2000, **18:** 1257-1261.

151. Selkov E, Basmanova S, Gaasterland T, Goryanin I, Gretchkin Y, Maltsev N *et al*.: **The metabolic pathway collection from EMP: the enzymes and metabolic pathways database.** *Nucleic Acids Res* 1996, **24:** 26-28.

152. Serov VN, Spirov AV, Samsonova MG: **Graphical interface to the genetic network database GeNet.** *Bioinformatics* 1998, **14:** 546-547.

153. Spirov AV, Bowler T, Reinitz J: **HOX Pro: a specialized database for clusters and networks of homeobox genes.** *Nucleic Acids Res* 2000, **28:** 337-340.

154. Stein L: **Creating a bioinformatics nation.** *Nature* 2002, **417:** 119-120.

155. Stoesser G, Baker W, van den BA, Camon E, Garcia-Pastor M, Kanz C *et al*.: **The EMBL Nucleotide Sequence Database.** *Nucleic Acids Res* 2002, **30:** 21-26.

156. Szymanski M, Barciszewski J: **Aminoacyl-tRNA synthetases database Y2K.** *Nucleic Acids Res* 2000, **28:** 326-328.

157. Takai-Igarashi T, Nadaoka Y, Kaminuma T: **A database for cell signaling networks.** *J Comput Biol* 1998, **5:** 747-754.

158. Tateno Y, Imanishi T, Miyazaki S, Fukami-Kobayashi K, Saitou N, Sugawara H *et al*.: **DNA Data Bank of Japan (DDBJ) for genome scale research in life science.** *Nucleic Acids Res* 2002, **30:** 27-30.

159. The Gene Ontology Consortium: **Gene ontology: tool for the unification of biology.** *Nat Genet* 2000, **25:** 25-29.

160. Thorn KS, Bogan AA: **ASEdb: a database of alanine mutations and their effects on the free energy of binding in protein interactions.** *Bioinformatics* 2001, **17:** 284-285.

161. Tong AH, Drees B, Nardelli G, Bader GD, Brannetti B, Castagnoli L *et al*.: **A combined experimental and computational strategy to define protein interaction networks for peptide recognition modules.** *Science* 2002, **295:** 321-324.

162. Tong AH, Evangelista M, Parsons AB, Xu H, Bader GD, Page N *et al*.: **Systematic genetic analysis with ordered arrays of yeast deletion mutants.** *Science* 2001, **294:** 2364-2368.

163. Uetz P, Giot L, Cagney G, Mansfield TA, Judson RS, Knight JR *et al*.: **A comprehensive analysis of protein-protein interactions in Saccharomyces cerevisiae.** *Nature* 2000, **403:** 623-627.

164. van Helden J, Naim A, Mancuso R, Eldridge M, Wernisch L, Gilbert D *et al*.: **Representing and analysing molecular and cellular function using the computer.** *Biol Chem* 2000, **381:** 921-935.

165. Venter JC, Adams MD, Myers EW, Li PW, Mural RJ, Sutton GG *et al*.: **The sequence of the human genome.** *Science* 2001, **291:** 1304-1351.

166. Visintin R, Amon A: **The nucleolus: the magician's hat for cell cycle tricks.** *Curr Opin Cell Biol* 2000, **12:** 752.

167. von Mering C, Krause R, Snel B, Cornell M, Oliver SG, Fields S *et al*.: **Comparative assessment of large-scale data sets of protein-protein interactions.** *Nature* 2002, **417:** 399-403.

168. Wagner A: **Robustness against mutations in genetic networks of yeast.** *Nat Genet* 2000, **24:** 355-361.

169. Wagner A, Fell DA: **The small world inside large metabolic networks.** *Proc R Soc Lond B Biol Sci* 2001, **268:** 1803-1810.

170. Wang Y, Addess KJ, Geer L, Madej T, Marchler-Bauer A, Zimmerman D *et al*.: **MMDB: 3D structure data in Entrez.** *Nucleic Acids Res* 2000, **28:** 243-245.

171. Wang Y, Anderson JB, Chen J, Geer LY, He S, Hurwitz DI *et al*.: **MMDB: Entrez's 3D-structure database.** *Nucleic Acids Res* 2002, **30:** 249-252.

172. Watts DJ, Strogatz SH: **Collective dynamics of 'small-world' networks.** *Nature* 1998, **393:** 440-442.

173. Weininger D: **SMILES, a Chemical Language and Information System.** *J Chem Inf Comput Sci* 1988, **28:** 31-36.

174. Westbrook J, Feng Z, Jain S, Bhat TN, Thanki N, Ravichandran V *et al*.: **The Protein Data Bank: unifying the archive.** *Nucleic Acids Res* 2002, **30:** 245-248.

175. Wheeler DL, Chappey C, Lash AE, Leipe DD, Madden TL, Schuler GD *et al*.: **Database resources of the National Center for Biotechnology Information.** *Nucleic Acids Res* 2000, **28:** 10-14.

176. White D, Batagelj V, Mrvar A: **Analyzing Large Kinship and Marriage Networks.** *Social Science Computer Review* 1999, **17:** 245-274.

177. Wingender E, Chen X, Fricke E, Geffers R, Hehl R, Liebich I *et al*.: **The TRANSFAC system on gene expression regulation.** *Nucleic Acids Res* 2001, **29:** 281-283.

178. Wingender E, Chen X, Hehl R, Karas H, Liebich I, Matys V *et al*.: **TRANSFAC: an integrated system for gene expression regulation.** *Nucleic Acids Res* 2000, **28:** 316-319.

179. Winter D, Lechler T, Li R: **Activation of the yeast Arp2/3 complex by Bee1p, a WASP-family protein.** *Curr Biol* 1999, **9:** 501-504.

180. Winzeler EA, Shoemaker DD, Astromoff A, Liang H, Anderson K, Andre B *et al*.: **Functional characterization of the S. cerevisiae genome by gene deletion and parallel analysis.** *Science* 1999, **285:** 901-906.

181. Xenarios I, Rice DW, Salwinski L, Baron MK, Marcotte EM, Eisenberg D: **DIP: the database of interacting proteins.** *Nucleic Acids Res* 2000, **28:** 289-291.

182. Xenarios I, Salwinski L, Duan XJ, Higney P, Kim SM, Eisenberg D: **DIP, the Database of Interacting Proteins: a research tool for studying cellular networks of protein interactions.** *Nucleic Acids Res* 2002, **30:** 303-305.

183. Yates JR: **Mass spectrometry. From genomics to proteomics.** *Trends Genet* 2000, **16:** 5-8.

184. Zanzoni A, Montecchi-Palazzi L, Quondam M, Ausiello G, Helmer-Citterich M, Cesareni G: **MINT: a Molecular INTeraction database.** *FEBS Lett* 2002, **513:** 135-140.

185. Zhu H, Bilgin M, Bangham R, Hall D, Casamayor A, Bertone P *et al*.: **Global analysis of protein activities using proteome chips.** *Science* 2001, **293:** 2101-2105.

Appendices

Appendix A: The BIND Data Specification in ASN.1

```
-- $Id: bind.asn,v 3.0 2002/07/10 03:29:33 gbader Exp $
-- ********************************************************************************
--
--  Biomolecular Interaction Network Database (BIND)
--  Data Specification
--
--  Interaction, Molecular Complex, Biological Pathway Data Structures
--
--
--  Authors: Gary D. Bader,       Christopher W.V. Hogue
--           bader@mshri.on.ca    hogue@mshri.on.ca
--
--           Ian Donaldson
--           ian.donaldson@utoronto.ca
--
--  Publication to cite:
--   Gary D. Bader and Christopher W. V. Hogue
--   BIND - a data specification for storing and describing biomolecular
--   interactions, molecular complexes and pathways
--   Bioinformatics May 2000 16: 465-477.
--
--  Thanks to SLRI staff, especially Ian Donaldson for invaluable discussion.
--
--  Hogue Lab - University of Toronto Biochemistry Department and the
--  Samuel Lunenfeld Research Institute, Mount Sinai Hospital
--  http://bioinfo.mshri.on.ca  hogue@mshri.on.ca
--
--  REVISIONS
--  Revision 0.1 -  Oct. 21, 1998
--  Revision 0.5 -  Feb. 2,  1999 (BIND web based data entry prototype)
--  Revision 0.6 -  Feb. 26, 1999 (Feedback from Biophysical Soc. Conf.)
--  Revision 0.8 -  May  3,  1999
--  Revision 0.9 -  May  31, 1999
--  Revision 1.0 -  June 7,  1999 (comments only added to 0.9)
--  Revision 1.1 -  Dec. 23, 1999 Internal revision (not for public release)
--  Revision 2.0 -  Jan. 31, 2000 (Minor changes from 1.1)
--  Revision 2.1 -  Nov. 7,  2000 (Added genetic interactions)
--  Revision 2.16 - Nov. 14, 2001 (Cumulative minor changes)
--  Revision 3.0 - Jul. 10, 2002 (Cumulative minor changes)
--
--  ftp://bioinfo.mshri.on.ca/pub/BIND/Spec/bind.asn for the latest revision.
--
--
--  NOTE:  This specification is in a variant of ASN.1 1990 that may not
--         be compatible with newer ASN.1 tools.  This specification also
--         depends on public domain specifications available from the
--         U.S. National Center for Biotechnology Information (NCBI)
--         ftp://ncbi.nlm.nih.gov/toolbox/ncbi_tools/
--         http://www.ncbi.nlm.nih.gov/Toolbox/
--
--
--  Copyright Notice:
--
--  Copyright 2001 Mount Sinai Hospital (MSH)
--
--  This program is free software; you can redistribute it and/or
--  modify it under the terms of the GNU General Public License as
--  published by the Free Software Foundation; either version 2 of
--  the License, or any later version.
--
--  This program is distributed in the hope that it will be useful,
--  but WITHOUT ANY WARRANTY; without even the implied warranty of
--  MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
--
--  See the GNU General Public License for more details.
--
--  You should have received a copy of the GNU General Public License
--  along with this program; if not, write to the
--         Free Software Foundation, Inc.,
--         59 Temple Place, Suite 330, Boston, MA
--         02111-1307  USA
--  or visit http://www.gnu.org/copyleft/gpl.html
--
--  SPECIAL EXCEPTIONS
--
```

```
--  As a special exception, Mount Sinai Hospital gives permission to
--  link this program with the following non-GPL programs or libraries,
--  and distribute the resulting executable, without including the source
--  code for these in the source distribution:
--
--  a) CodeBase 6.5 or greater from Sequiter Software Inc.
--
--  b) The C or C++ interface to Oracle 8.x or greater, from
--     Oracle Corporation or IBM DB2 UDB.
--
-- ********************************************************************************

BIND DEFINITIONS ::=
BEGIN

EXPORTS BIND-Submit,
        BIND-Interaction, BIND-Interaction-set,
        BIND-Pathway, BIND-Pathway-set,
        BIND-Molecular-Complex, BIND-Complex-set,
        BIND-cellstage, BIND-object, BIND-object-type-id, BIND-place-set,
        BIND-condition-set, BIND-loc,
        BIND-action-set, BIND-state-set,
        RealVal-Units, Interaction-id,
        Molecular-Complex-id, Pathway-id,
        BIND-bid;

IMPORTS Date, Int-fuzz, User-object, Dbtag FROM NCBI-General
        Author FROM NCBI-Biblio
        Bioseq FROM NCBI-Sequence
        Pub FROM NCBI-Pub
        BioSource FROM NCBI-BioSource
        Seq-loc, Seq-id FROM NCBI-Seqloc
        Biostruc FROM MMDB
        Biostruc-feature-set FROM MMDB-Features
        BIND-profile FROM BIND-Profile;


-- /!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\
-- ****************************************
-- * Data Submission and Database exchange *
-- ****************************************

-- ********************************************************************************
-- General data exchange
--
-- This object is used to submit all information to BIND.
--
-- Field description for BIND-Submit
-- *********************************
-- date         = date of creation of this set
-- database     = description of database where this data originated
-- sub          = person who is responsible for this data submission
-- sub-id       = BIND submit ID for this submission
-- acc-nums     = list of BIND accession numbers that this submission contains
-- interactions = a collection of interaction records
-- complexes    = a collection of molecular complex records
-- pathways     = a collection of pathway records
-- ********************************************************************************


BIND-bid ::= INTEGER  --reserved for future use

BIND-Submit ::= SEQUENCE {
        date Date,
        database BIND-Database-site OPTIONAL,
        sub BIND-Submitter,
        sub-id BIND-Submit-id OPTIONAL,
        acc-nums SEQUENCE OF BIND-accession-number OPTIONAL,
        interactions BIND-Interaction-set OPTIONAL,
        complexes BIND-Complex-set OPTIONAL,
        pathways BIND-Pathway-set OPTIONAL
        }

BIND-Submit-id ::= INTEGER

BIND-accession-number ::= CHOICE {
        interaction Interaction-id,
        complex Molecular-Complex-id,
        pathway Pathway-id
        }
```

```
-- ************************
-- * Database description *
-- ************************

-- ********************************************************************************
-- Description of a database site
--
-- Field description for BIND-Database-site
-- ****************************************
-- descr        = text description of this database
--                (e.g. C. elegans interaction database)
-- country      = country where this database is based. Use full name.
--                (e.g. Canada)
-- homepage-url = Internet Universal Resource Locator for the database web
--                site (e.g. http://bioinfo.mshri.on.ca)
-- reference    = a Medline reference for this database
-- ********************************************************************************


BIND-Database-site ::= SEQUENCE {
        descr VisibleString,
        country VisibleString,
        homepage-url VisibleString OPTIONAL,
        reference BIND-pub-set OPTIONAL
        }


-- ************************
-- * Database description *
-- ************************

-- ********************************************************************************
-- Description of a record collection
--
-- Field description for BIND-Rec-coll-descr
-- *****************************************
-- descr = text description of this record collection
--         (e.g. BIND, Hogue Lab Interactions)
-- db    = database where this record originated (for use in a data warehouse)
-- ********************************************************************************


BIND-Rec-coll-descr ::= SEQUENCE {
        descr VisibleString,
        db BIND-Database-site OPTIONAL
        }


-- *************
-- * Submitter *
-- *************

-- ********************************************************************************
-- Description of a submitter (Adaptation of NCBI Submit-Block)
--
-- Field description for BIND-Submitter
-- ************************************
-- contact = submitter contact information
-- hup     = hold this submission until published
-- subtype = submission type
-- tool    = tool used to submit record (e.g. BIND Web Data Entry version 1.0)
-- ********************************************************************************


BIND-Submitter ::= SEQUENCE {
        contact BIND-Contact-info,
        hup BOOLEAN DEFAULT FALSE,
        subtype ENUMERATED {
                not-specified (0),
                new (1),                -- new data
                update (2),             -- update by author
                revision (3),           -- 3rd party (non-author) update
                import (4),             -- import into the database
                export (5),             -- export from the database
                other (255) },
        tool BIND-Submission-tool OPTIONAL
        }


-- ********************************************************************************
```

```
-- Structured submission tool description
-- ****************************************************************************


BIND-Submission-tool ::= SEQUENCE {
        name VisibleString,
        version VisibleString,
        descr VisibleString OPTIONAL
        }


-- ***********************
-- * Contact Information *
-- ***********************

-- ****************************************************************************
-- Contact information (Adaptation of NCBI Contact-info)
--
-- Field description for BIND-Contact-info
-- ***************************************
-- first-name     = First name of submitter
-- middle-initial = Middle initial of submitter
-- last-name      = Last name of submitter
-- address        = Street address of submitter
-- room           = Room number
-- dept           = Department
-- institute      = Institute if this is different than organization
--                  (e.g. research institute)
-- organization   = Organization (e.g. University of Toronto)
-- city           = City
-- pcode          = Zip or postal code
-- country        = Country
-- phone          = Phone number (with area code)
-- fax            = Fax number (with area code)
-- email          = E-mail address
-- userid         = User ID number
-- password       = User password
-- other          = any other contact information
-- ****************************************************************************


BIND-Contact-info ::= SEQUENCE {
        first-name VisibleString OPTIONAL,
        middle-initial VisibleString OPTIONAL,
        last-name VisibleString OPTIONAL,
        address SEQUENCE OF VisibleString OPTIONAL,
        room VisibleString OPTIONAL,
        dept VisibleString OPTIONAL,
        institute VisibleString OPTIONAL,
        organization VisibleString OPTIONAL,
        city VisibleString OPTIONAL,
        pcode VisibleString OPTIONAL,
        country VisibleString OPTIONAL,
        phone VisibleString OPTIONAL,
        fax VisibleString OPTIONAL,
        email VisibleString OPTIONAL,
        userid INTEGER OPTIONAL,
        other SEQUENCE OF Dbtag OPTIONAL,
        password VisibleString OPTIONAL
        }


-- ***************
-- * Publications *
-- ***************

-- ****************************************************************************
-- A set of publications
--
-- Field description for BIND-pub-set
-- **********************************
-- disputed = TRUE if a BIND-pub-object in this set contains a dispute flag
-- pubs     = a sequence of BIND-pub-objects
-- evidence = unpublished data/evidence for use in a private satellite
--            database
-- ****************************************************************************


BIND-pub-set ::= SEQUENCE {
        disputed BOOLEAN DEFAULT FALSE,
        pubs SEQUENCE OF BIND-pub-object,
```

```
        evidence SEQUENCE OF BIND-evidence-object OPTIONAL
        }


-- ****************************************************************************
-- A publication
--
-- Field description for BIND-pub-object
-- *************************************
-- descr   = text description of this object
-- opinion = does this publication support or dispute the data?
-- quality = stores quality of information measure that may be in publication
--           (IMPORTANT: This is not a user based quality assessment)
-- pub     = full NCBI publication reference
-- extref  = external reference(s) to an e.g. publication database
-- ****************************************************************************


BIND-pub-object ::= SEQUENCE {
        descr VisibleString OPTIONAL,
        opinion ENUMERATED {
                none (0),
                support (1),
                dispute (2)
                },
        pub Pub,
        quality BIND-quality OPTIONAL,
        extref SEQUENCE OF BIND-other-db OPTIONAL
        }


-- ****************************************************************************
-- A piece of user defined evidence
--
-- Field description for BIND-evidence-object
-- ******************************************
-- descr         = text description of this object
-- opinion       = does this evidence support or dispute the data?
-- quality       = stores quality of information measure that may be in
--                 publication
--                 (IMPORTANT: This is not a user based quality assessment,
--                  only used if quality assessment is in a publication.)
-- user-evidence = user defined evidence (e.g. gel picture)
-- extref        = external reference(s) to an e.g. evidence database (LIMS)
-- ****************************************************************************


BIND-evidence-object ::= SEQUENCE {
        descr VisibleString OPTIONAL,
        opinion ENUMERATED {
                none (0),
                support (1),
                dispute (2)
                },
        user-evidence User-object,
        quality BIND-quality OPTIONAL,
        extref SEQUENCE OF BIND-other-db OPTIONAL
        }


-- ****************************************************************************
-- A quality assessment from an experimenter roughly mapped to a percentage.
-- (IMPORTANT: This is not a user based quality assessment, only used if
-- quality assessment is in a publication.)
--
-- Field description for BIND-quality
-- **********************************
-- quality = quality assessment normalized to a percentage. Higher percentage
--           means better quality.
--           (e.g. rating of A,B,C,D maps to 100%,75%,50%,25%)
-- descr   = A description of quality assessment system.
-- ****************************************************************************


BIND-quality ::= SEQUENCE {
        quality-pct INTEGER,
        descr VisibleString
        }


-- *****************
```

```
-- * Record Update *
-- *****************

-- ********************************************************************************
-- An update for a record
--
-- Field description for BIND-update
-- *********************************
-- date  = date of this update
-- descr = text description of update (this can store any update information
--         up to the entire previous version of the record in ASN.1)
-- ********************************************************************************


BIND-update-object ::= SEQUENCE {
        date Date,
        descr VisibleString
        }

BIND-update-list ::= SEQUENCE {
        updates SEQUENCE OF BIND-update-object
        }


-- ********************************************************************************
-- An author (can't use Auth-list from NCBI-Biblio because it is not exported)
--
-- Field description for BIND-author
-- *********************************
-- auth = an author
-- ********************************************************************************


BIND-author ::= SEQUENCE {
        auth Author
        }


-- ********************************************************************************
-- Cell cycle stage
--
-- Field description for BIND-cellstage
-- ************************************
-- phase = phase of cell cycle
-- descr = text description of cell stage (e.g. if 'other' is specified)
-- ********************************************************************************


BIND-cellstage ::= SEQUENCE {
        phase INTEGER {
                not-specified (0),
                constitutive (1),
                interphase (2),
                division (3),
                g1 (4),
                s (5),
                g2 (6),
                mitosis (7),
                prophase (8),
                prometaphase (9),
                metaphase (10),
                anaphase (11),
                telophase (12),
                cytokinesis (13),
                meiosis (14),
                prophase1 (15),
                leptotene (16),
                zygotene (17),
                pachytene (18),
                diplotene (19),
                diakinesis (20),
                metaphase1(21),
                anaphase1 (22),
                telophase1 (23),
                meiotic-cytokinesis (24),
                prophase2 (25),
                metaphase2 (26),
                anaphase2 (27),
                telophase2 (28),
                meiotic-cytokinesis2 (29),
                other (255)
```

```
            },
        descr VisibleString OPTIONAL
        }


-- ********************************************************************************
-- A Real Number
--
-- Field description for RealVal-Units
-- ***********************************
-- scaled-integer-value * 10^(scale-factor)
-- units = string value of the units involved (e.g. ml, M, etc.)
-- ********************************************************************************


RealVal-Units ::= SEQUENCE {
        scale-factor          INTEGER,
        scaled-integer-value  INTEGER,
        units VisibleString   OPTIONAL
        }


-- ********************************************************************************
-- A Fuzzy Real Number
--
-- Modeled after NCBI Int-fuzz
--
-- Field description for RealFuzzVal-Units
-- ***************************************
-- p-m   = plus or minus a fixed amount
-- range = max to min
-- alt   = set of alternate numbers
-- ********************************************************************************


RealFuzzVal-Units ::= CHOICE {
        p-m RealVal-Units,
        range RealFuzzVal-Range,
        alt SEQUENCE OF RealVal-Units
        }

RealFuzzVal-Range ::= SEQUENCE {
        max RealVal-Units,
        min RealVal-Units
        }


-- ********************************************************************************
-- A possibly fuzzy integer
--
-- Field description for BIND-int-fuzz
-- ***********************************
-- num      = an integer
-- num-fuzz = a fuzzy integer
-- ********************************************************************************


BIND-int-fuzz ::= CHOICE {
        num INTEGER,
        num-fuzz Int-fuzz
        }


-- ********************************************************************************
-- A generalized value
--
-- Field description for BIND-param
-- ********************************
-- descr = description of this parameter
-- value = the actual value
-- ********************************************************************************


BIND-param ::= SEQUENCE {
        descr VisibleString,
        value BIND-param-val
        }

BIND-param-val ::= CHOICE {
        string VisibleString,
        real RealVal-Units,
```

```
        realfuz RealFuzzVal-Units
        }


-- /!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\
-- ***************
-- * Interaction *
-- ***************

-- ********************************************************************************
-- A set of interactions
--
-- Field description for BIND-Interaction-set
-- ******************************************
-- date         = date this set of records was collected
-- database     = name and description of database that this set originates
-- interactions = set of interaction records
-- ********************************************************************************


BIND-Interaction-set ::= SEQUENCE {
        date Date OPTIONAL,
        database BIND-Database-site OPTIONAL,
        interactions SEQUENCE OF BIND-Interaction
        }


-- ********************************************************************************
-- A BIND-Interaction record can store all of the details of an interaction
-- between any two genes, molecules (or atoms).
--
-- Generally, BIND-Interaction stores a physical interaction between two
-- objects.  However, if gene is chosen for one of the BIND-object types,
-- the interaction will be less precise and will be considered a functional
-- relationship instead of a physical interaction.  Gene can represent the
-- actual genetic sequence as DNA (or RNA in e.g. some virii) or any
-- intermediate or final product of the gene such as mRNA, tRNA, protein or
-- post-translationally modified protein.
--
-- Field description for BIND-Interaction
-- **************************************
-- date     = date of record entry
-- updates  = a list of updates for this record
-- iid      = interaction accession number
-- a        = molecule A interacts with...
-- b        = molecule B
-- descr    = description of interaction
-- source   = empirical evidence references (publications)
-- authors  = person(s) who authored this record
-- division = interaction is part of a record collection/division
--            (i.e. a satellite BIND database, HTI)
-- priv     = TRUE if this interaction is private
-- extref   = external reference(s) for e.g. another interaction database
-- ********************************************************************************
-- NOTE: In the context of this data specification, the 'priv' flag means:
--       -Do not export this record.
--       -In a public database, this record is not available to be publicly
--        retrieved.
--       -In a private database, this record can be retrieved, but it will
--        not be exported.
-- ********************************************************************************


BIND-Interaction ::= SEQUENCE {
        date Date,
        updates SEQUENCE OF BIND-update-object OPTIONAL,
        iid Interaction-id,
        a BIND-object,
        b BIND-object,
        descr BIND-descr,
        source BIND-pub-set,
        authors SEQUENCE OF BIND-author OPTIONAL,
        division BIND-Rec-coll-descr OPTIONAL,
        priv BOOLEAN DEFAULT FALSE,
        extref SEQUENCE OF BIND-other-db OPTIONAL
        }

Interaction-id ::= INTEGER


-- ***********************
```

```
-- * Biomolecular Object *
-- ***********************

-- ********************************************************************************
-- Any chemical object
--
-- Field description for BIND-object
-- *********************************
-- short-label = short label of this object (e.g. ATP, S4, HSP70)
-- other-names = list of short-label synonyms for this object
-- id = the type of chemical object and a pointer to a record in a database
--      of the object type (e.g. protein database)
-- origin = material source (biological or chemical origin)
-- cell-stage = description of cell cycle stages this object is specific to
-- place = the cellular location of this molecule
-- seq = space for sequence, if it is not in a public database
--       ALSO, this can be a consensus sequence for binding of this object
--       (e.g. transcription factor binding to DNA)
-- struc = space for complete structure, if not in public database
--         (This should not be used to store a structure that is already in
--         the MMDB)
-- descr = text description of this object
-- user-id = OBSOLETE, use extref
-- extref  = user defined use (e.g. can be used to reference records in an
--           optional relational BIND-object table)
-- ********************************************************************************


BIND-object ::= SEQUENCE {
        short-label VisibleString,
        other-names SEQUENCE OF VisibleString OPTIONAL,
        id BIND-object-type-id,
        origin BIND-object-origin,
        cell-stage SEQUENCE OF BIND-cellstage OPTIONAL,
        place BIND-place-set OPTIONAL,
        seq Bioseq OPTIONAL,
        struc Biostruc OPTIONAL,
        descr VisibleString OPTIONAL,
        user-id INTEGER OPTIONAL,     --OBSOLETE: Feb.13.2002, use extref
        extref SEQUENCE OF BIND-other-db OPTIONAL
        }


-- ********************************************************************************
-- An ID for a chemical object
--
-- Field description for BIND-object-type-id
-- *****************************************
-- not-specified  = the type of this object is not specified
-- protein        = the object is a protein - reference protein sequence
-- dna            = the object is DNA - reference DNA sequence
-- rna            = the object is RNA - reference RNA sequence
-- small-molecule = the object is a small molecule (e.g proton to penicillin)
--                  reference a small molecule database like LIGAND
-- complex        = the object is a molecular complex - reference molecular
--                  complex in BIND
-- gene           = the object is a gene - reference DNA sequence. While the
--                  DNA is referenced, this object can actually represent
--                  DNA, RNA, protein or other modified gene product. It is
--                  a 'fuzzy' representation.
-- photon         = the object is light - record properties of light
-- ********************************************************************************


BIND-object-type-id ::= CHOICE {
        not-specified NULL,
        protein BIND-id,
        dna BIND-id,
        rna BIND-id,
        small-molecule BIND-small-molecule-id,
        complex Molecular-Complex-id,
        gene BIND-id,
        photon BIND-photon
        }

BIND-object-origin ::= CHOICE {
        not-specified NULL,
        org BioSource,
        chem BIND-chemsource
        }
```

```
-- ********************************************************************************
-- Summary description of a chemical compound
--
-- Field description for BIND-chemsource
-- ************************************
-- names            = chemical compound name and any synonyms
-- smiles-string    = standard smiles-string for this compound
--   References for SMILES language:
--     D. Weininger, SMILES, a Chemical Language and Information System.
--     1. Introduction to Methodology and Encoding Rules,
--     J. Chem. Inf. Comput. Sci. 1988, 28, 31-36.
--   Web sites:
--     http://www.daylight.com/dayhtml/smiles/smiles-intro.html
--     http://www2.ccc.uni-erlangen.de/services/smiles.html
-- molecular-weight = molecular weight of this compound in g/mol
-- chemical-formula = chemical formula of the compound (e.g.C3H7NO2)
-- cas-number       = Chemical Abstracts Service (http://www.cas.org/)
--                    database number for this compound (e.g. 56-41-7)
-- nat-prod       = biological source information if this is a natural product
-- ********************************************************************************


BIND-chemsource ::= SEQUENCE {
        names SEQUENCE OF VisibleString,
        smiles-string VisibleString OPTIONAL,
        chemical-formula VisibleString OPTIONAL,
        molecular-weight RealVal-Units OPTIONAL,
        cas-number VisibleString OPTIONAL,
        nat-prod BioSource OPTIONAL
        }


-- ***************
-- * Identifiers *
-- ***************

-- ********************************************************************************
-- General sequence or domain identifier
--
-- Field description for BIND-id
-- *****************************
-- gi = NCBI integer accession number (optional only for sequence data with
--      no NCBI database identifier).
--      NOTE: gi is stored so that a BIND-object refers to a constant sequence
--      molecule.  This is necessary to maintain data integrity of Seq-loc's
--      also stored in the BIND database.
-- di    = domain accession number (from the domain split database)
-- other = open field for other possible NCBI defined pointers
--         (if possible, equivalent GenBank accession number to this
--          gi should be stored here as well)
--         Any database pointer to a sequence may be put in here. e.g. PIR
--
-- NOTE: There is a field for gi in a Seq-id, but it should not be used
-- in this object. Private databases should use the Seq-id.general field.
-- ********************************************************************************


BIND-id ::= SEQUENCE {
        gi Geninfo-id OPTIONAL,
        di Domain-id OPTIONAL,
        other SEQUENCE OF Seq-id OPTIONAL
        }

Geninfo-id ::= INTEGER

Domain-id ::= INTEGER


-- ********************************************************************************
-- Pointer to a small molecule database
--
-- Field description for BIND-small-molecule-id
-- ********************************************
-- internal = id number of an internally kept record of a chemical compound
--            Primary key in the BIND small molecule database.
-- other-db = generic pointer to any other database (e.g. Japanese LIGAND db)
--      Contains the name of the database, an integer pointer and/or a string
--      pointer.
-- ********************************************************************************
```

```
BIND-small-molecule-id ::= CHOICE {
        internal Internal-small-molecule-id,
        other-db BIND-other-db
        }

Internal-small-molecule-id ::= INTEGER

BIND-other-db ::= SEQUENCE {
        dbname VisibleString,
        intp INTEGER OPTIONAL,
        strp VisibleString OPTIONAL
        }

-- ********************************************************************************
-- Description of electro-magnetic radiation (light)
--
-- Field description for BIND-photon
-- *********************************
-- wavelength = wavelength for this light (can be fuzzy)
-- intensity  = intensity for this light (can be fuzzy)
-- ********************************************************************************

BIND-photon ::= SEQUENCE {
        wavelength RealFuzzVal-Units,
        intensity RealFuzzVal-Units
        }

-- ***********************************************
-- * Interaction Description (in BIND-Interaction) *
-- ***********************************************

-- ********************************************************************************
-- Full description of an interaction
--
-- Field description for BIND-descr
-- ********************************
-- simple-descr   = text description of this interaction
-- place          = description of cellular place of interaction
-- cond           = binding conditions/experimental conditions
-- cons           = conserved sequence comment
-- binding-sites  = location of binding sites on molecule A and B
-- action = list of chemical actions that can occur in this interaction
-- state  = list of chemical states of molecule A and B as well as required
--          state for interaction to occur
-- intramolecular = only relevant if molecule A and B refer to the same
--                  molecule. TRUE if the interaction is intramolecular
-- ********************************************************************************


BIND-descr ::= SEQUENCE {
        simple-descr VisibleString OPTIONAL,
        place BIND-place-set OPTIONAL,
        cond BIND-condition-set OPTIONAL,
        cons BIND-cons-seq-set OPTIONAL,
        binding-sites BIND-loc OPTIONAL,
        action BIND-action-set OPTIONAL,
        state BIND-state-descr OPTIONAL,
        intramolecular BOOLEAN DEFAULT FALSE
        }


-- *********************************************
-- * Cellular Interaction Place (in BIND-descr) *
-- *********************************************


-- ********************************************************************************
-- A set of cellular locations.
--
-- It is assumed that adjacent cellular locations listed in this set
-- represents spanning across adjacent sub-cellular compartments
--
-- Field description for BIND-place-set
-- ************************************
-- max-bpid = the highest bpid used in this set
-- places   = set of BIND-place objects
-- ********************************************************************************


BIND-place-set ::= SEQUENCE {
```

```
        max-bpid BIND-place-id,
        places SEQUENCE OF BIND-place
        }


-- ********************************************************************************
-- Location of interaction with respect to the cell
--
-- Field description for BIND-place
-- ********************************
-- bpid       = internal BIND place ID number (0..n within interaction record)
-- gen-place  = general cellular locations where this interaction takes place
--              (computer readable)
-- spec-place = specific text locations of the interaction
--              (human readable)
-- source     = empirical evidence references (publications)
-- descr      = text description (e.g. method of finding interaction place)
-- ********************************************************************************


BIND-place ::= SEQUENCE {
        bpid BIND-place-id,
        gen-place BIND-gen-place-set,
        spec-place BIND-spec-place-set OPTIONAL,
        source BIND-pub-set OPTIONAL,
        descr VisibleString OPTIONAL
        }

BIND-place-id ::= INTEGER


-- ********************************************************************************
-- Unique reference to a cellular place
--
-- Field description for BIND-place-ref
-- ************************************
-- from-iid = interaction that contains the place
-- place    = BIND-place-id (bpid) of the place
-- ********************************************************************************


BIND-place-ref ::= SEQUENCE {
        from-iid Interaction-id,
        place BIND-place-id
        }


-- ********************************************************************************
-- General start and end places for an interaction
--
-- Field description for BIND-gen-place-set
-- ****************************************
-- start = general place in the cell where this interaction takes place
-- end   = general place in the cell where this interaction ends
--         (e.g. for translocation)
-- descr = text description (e.g. mechanism of translocation)
-- ********************************************************************************


BIND-gen-place-set ::= SEQUENCE {
        start BIND-gen-place,
        end BIND-gen-place OPTIONAL,
        descr VisibleString OPTIONAL
        }


-- ********************************************************************************
-- General cellular place where this interaction takes place
--
-- This object is meant to be computer readable for e.g. a pathway
-- drawing program.  Further cell locations are not listed because
-- there are too many in biology.
--
-- Field description for BIND-gen-place
-- ************************************
-- A listing of general cell places
-- other = provide further description in BIND-spec-place
-- ********************************************************************************


BIND-gen-place ::= CHOICE {
```

```
        not-specified NULL,
        extracellular NULL,
        cytoplasm NULL,
        cell-wall BIND-membrane,
        outer-membrane BIND-membrane,
        cytoplasmic-membrane BIND-membrane,
        organelle-unknown BIND-membrane,
        organelle-other BIND-membrane,
        nucleus BIND-membrane,              --OBSOLETE Dec2001. Use nucleus-dmo
        nuc-outer-membrane BIND-membrane,   --OBSOLETE Dec2001. Use nucleus-dmo
        nuc-inner-membrane BIND-membrane,   --OBSOLETE Dec2001. Use nucleus-dmo
        nuclear-pore BIND-localize,
        nucleolus BIND-localize,
        chromatin BIND-localize,
        er-general BIND-membrane,
        er-smooth BIND-membrane,
        er-rough BIND-membrane,
        golgi BIND-membrane,
        golgi-stack BIND-membrane,
        cis-golgi BIND-membrane,
        medial-golgi BIND-membrane,
        trans-golgi BIND-membrane,
        vacuole BIND-membrane,
        vesicle BIND-membrane,
        lysosome BIND-membrane,
        peroxisome BIND-membrane,
        endosome BIND-membrane,
        mitochondrion BIND-dmo,
        chloroplast BIND-chlor,
        plastid BIND-dmo,
        centrosome BIND-localize,
        centriole BIND-localize,
        cytoskeleton BIND-localize,
        ribosome BIND-localize,
        flagella BIND-cilflag,
        cilia BIND-cilflag,
        other NULL,
        nucleus-dmo BIND-dmo
        }

-- ********************************************************************************
-- Description of a location in a lipid bilayer membrane
--
-- Field description for BIND-membrane
-- ***********************************
-- not-specified = somewhere in membrane
-- outer-surface = on the outer surface of the membrane
-- within        = within the bilayer
-- inner-surface = on the inner surface of the membrane
-- lumen         = in the lumen that the membrane surrounds
-- ********************************************************************************


BIND-membrane ::= ENUMERATED {
        not-specified (0),
        outer-surface (1),
        within (2),
        inner-surface (3),
        lumen (4)
        }

-- ********************************************************************************
-- Description of a location in a double membrane organelle
--
-- Field description for BIND-dmo
-- ******************************
-- general          = generally part of the organelle, outer or inner membrane
--                    localization not known
-- outer-membrane   = found in the outer membrane
-- inner-membrane   = found in the inner membrane
-- general-membrane = found in the membrane fraction, outer or inner membrane
--                    not known
-- ********************************************************************************


BIND-dmo ::= CHOICE {
        general NULL,
        outer-membrane BIND-membrane,
        inner-membrane BIND-membrane,
```

```
        general-membrane BIND-membrane
        }


-- ********************************************************************************
-- Description of a location in a chloroplast
--
-- Field description for BIND-chlor
-- ********************************
-- general          = generally part of the organelle, outer or inner membrane
--                    localization not known
-- outer-membrane   = found in the outer membrane
-- inner-membrane   = found in the inner membrane
-- grana            = found in the grana
-- thylakoid        = found in the thylakoid
-- general-membrane = found in the membrane fraction, outer or inner membrane
--                    not known
-- ********************************************************************************


BIND-chlor ::= CHOICE {
        general NULL,
        outer-membrane BIND-membrane,
        inner-membrane BIND-membrane,
        grana BIND-membrane,
        thylakoid BIND-membrane,
        general-membrane BIND-membrane
        }


-- ********************************************************************************
-- Description of a location in a non membrane surrounded cell component
--
-- Field description for BIND-localize
-- ***********************************
-- not-specified = somewhere in the component
-- component     = part of the component
-- peripherally-associated = associated with the surface of the component
-- other         = other (should have a description in BIND-gen-place-set2)
-- ********************************************************************************


BIND-localize ::= ENUMERATED {
        not-specified (0),
        component (1),
        peripherally-associated (2),
        other (255)
        }


-- ********************************************************************************
-- Description of a location in a cilium or flagellum
--
-- Field description for BIND-cilflag
-- **********************************
-- general  = generally part of the organelle
-- membrane = part of the membrane surrounding the cilium or flagellum
-- inside   = inside the plasma membrane
-- ********************************************************************************


BIND-cilflag ::= CHOICE {
        general NULL,
        membrane BIND-membrane,
        inside BIND-localize
        }


-- ********************************************************************************
-- Specific start and end places for an interaction
-- (Human readable)
--
-- Field description for BIND-spec-place-set
-- *****************************************
-- start  = specific location where this interaction takes place
--         (e.g. trans golgi, basal membrane, inner mitochondrial space, etc.)
-- end    = specific location where this interaction ends
-- ********************************************************************************


BIND-spec-place-set ::= SEQUENCE {
```

```
        start BIND-spec-place,
        end BIND-spec-place OPTIONAL
        }


-- ****************************************************************************
-- Specific place of an interaction
-- (Human readable)
--
-- Field description for BIND-spec-place
-- ************************************
-- descr    = description of this place
-- other-db = reference in a cellular location database
-- ****************************************************************************


BIND-spec-place ::= SEQUENCE {
        descr VisibleString,
        other-db BIND-other-db OPTIONAL
        }


-- ****************************************
-- * Interaction conditions (in BIND-descr) *
-- ****************************************

-- ****************************************************************************
-- A set of experimental conditions.
--
-- Field description for BIND-conditions-set
-- *****************************************
-- max-icid   = the highest icid used in this set
-- conditions = set of BIND-condition objects
-- ****************************************************************************


BIND-condition-set ::= SEQUENCE {
        max-icid Internal-conditions-id,
        conditions SEQUENCE OF BIND-condition
        }


-- ****************************************************************************
-- An experimental condition that has been used to observe
-- this interaction.  Interaction conclusion must be reproducible
-- using this information.
--
-- Field description for BIND-condition
-- ************************************
-- icid        = internal condition id (0..n within interaction record)
-- general     = list of possible general experimental conditions
-- system      = experimental system used
-- exp-form-a  = experimental form of biomolecule A used if different from
--               actual biomolecule. (e.g. HIS tagged sequence, ATP analogue)
-- exp-form-b  = experimental form of biomolecule B used if different from
--               actual biomolecule.
-- site        = site(s) on molecule A or B that this experiment detects or
--               are involved in this experiment
-- descr       = text description (e.g. if 'other' is specified
--               in conditions or system)
-- other-db    = reference to an experimental method database
-- source      = empirical evidence
-- genetic-exp = genetic experiment description
-- action      = chemical action(s) that this experiment detects
-- state       = chemical state(s) that this experiment detecs
-- negative-result = TRUE if this experiment is a negative result e.g. a
--                   mutation in a specific residue described in the
--                   experimental form within this BIND-condition prevents
--                   the interaction from being seen.  Should only be TRUE
--                   when the experiment is changing molecule A or B so that
--                   they do not interact to show the importance of the
--                   changed form on the interaction.
-- bait-condition  = flag to mark if A or B is a 'bait' in this experiment.
--                   e.g. in a co-immunoprecipitation experiment, A is an
--                   epitope tagged protein that is used to purify interactors
--                   from cell lysate.  This flag could be used in a
--                   visualization system to draw an arrow from bait to hit.
-- ****************************************************************************


BIND-condition ::= SEQUENCE {
```

```
        icid Internal-conditions-id,
        general ENUMERATED {
                in-vitro (0),
                in-vivo (1),
                in-situ (2),
                in-silico (3),
                other (255)
                },
        system BIND-experimental-system,
        exp-form-a BIND-experimental-form OPTIONAL,
        exp-form-b BIND-experimental-form OPTIONAL,
        site SEQUENCE OF BIND-loc-site-ref OPTIONAL,
        descr VisibleString OPTIONAL,
        other-db BIND-other-db OPTIONAL,
        source BIND-pub-set OPTIONAL,
        genetic-exp BIND-genetic-experiment OPTIONAL,
        action SEQUENCE OF BIND-action-ref OPTIONAL,
        state SEQUENCE OF BIND-state-ref OPTIONAL,
        negative-result BOOLEAN DEFAULT FALSE,
        bait-condition ENUMERATED {
                        a-is-bait (0),
                        b-is-bait (1),
                        not-applicable (3)
                } DEFAULT not-applicable
        }

Internal-conditions-id ::= INTEGER


-- ********************************************************************************
-- Unique reference to an experimental condition
--
-- Field description for BIND-condition-ref
-- ****************************************
-- from-iid  = interaction that contains the place
-- condition = Internal-conditions-id (icid) of the experimental condition
-- ********************************************************************************


BIND-condition-ref ::= SEQUENCE {
        from-iid Interaction-id,
        condition Internal-conditions-id
        }


-- ********************************************************************************
-- A list of experimental systems
-- ********************************************************************************


BIND-experimental-system ::= INTEGER {
        not-specified (0),
        alanine-scanning (1),
        affinity-chromatography (2),
        atomic-force-microscopy (3),
        autoradiography (4),
        competition-binding (5),
        cross-linking (6),
        deuterium-hydrogen-exchange (7),
        electron-microscopy (8),
        electron-spin-resonance (9),
        elisa (10),
        equilibrium-dialysis (11),
        fluorescence-anisotropy (12),
        footprinting (13),
        gel-retardation-assays (14),
        gel-filtration-chromatography (15),
        hybridization (16),
        immunoblotting (17),
        immunoprecipitation (18),
        immunostaining (19),
        interaction-adhesion-assay (20),
        light-scattering (21),
        mass-spectrometry (22),
        membrane-filtration (23),
        monoclonal-antibody-blockade (24),
        nuclear-translocation-assay (25),
        phage-display (26),
        reconstitution (27),
        resonance-energy-transfer (28),
        site-directed-mutagenesis (29),
```

```
        sucrose-gradient-sedimentation (30),
        surface-plasmon-resonance-chip (31),
        transient-coexpression (32),
        three-dimensional-structure (33),
        two-hybrid-test (34),
        allele-specific-complementation (35),
        far-western (36),
        colocalization (37),
        other (255)
        }

-- ********************************************************************************
-- Description of the experimental form of a biomolecule used.
--
-- Field description for BIND-experimental-form
-- ********************************************
-- object  = experimental form is a biomolecule
--           (e.g. HIS tagged sequence, ATP analogue)
-- profile = experimental form is a profile (in silico experiments)
-- gene    = experimental form of a gene (if A or B is a gene)
-- ********************************************************************************


BIND-experimental-form ::= CHOICE {
        object BIND-object,
        profile BIND-profile,
        gene BIND-genotype
        }


-- ********************************************************************************
-- Description of the experimental form of a gene.
--
-- A collection of all of the alleles of this gene that are present on
-- chromosomal and extra-chromosomal genetic elements. Genotype is the
-- allelic composition of the gene of interest.
--
-- Field description for BIND-genotype
-- ***********************************
-- tot-copy-num = the total copy number of all the alleles of this gene in
--                the organism used in this experiment
-- alleles      = the sequence of alleles of this gene in the organism used
--                in this experiment (generally only the alleles that are
--                interesting from the point of view of this experiment -
--                other alleles may be put in the background field)
-- expression   = the phenotype expression(s) of this collection of alleles
-- background   = genetic background of this genotype
-- ********************************************************************************


BIND-genotype ::= SEQUENCE {
        tot-copy-num BIND-allele-copy-num OPTIONAL,
        alleles SEQUENCE OF BIND-allele,
        expression SEQUENCE OF BIND-phenotype OPTIONAL,
        background BIND-genetic-background OPTIONAL
        }


-- ********************************************************************************
-- Description of the copy number of an allele.
--
-- Field description for BIND-allele-copy-num
-- ******************************************
-- high         = high allele copy number
-- high-ex      = high, copy number is known
-- single       = single allele copy
-- wild-type    = wild-type allele copy number
-- wild-type-ex = wild-type, copy number is known
-- reduced      = reduced allele copy number
-- reduced-ex   = reduced, copy number is known
-- ********************************************************************************


BIND-allele-copy-num ::= CHOICE {
        high NULL,
        high-ex BIND-int-fuzz,
        single NULL,
        wild-type NULL,
        wild-type-ex BIND-int-fuzz,
        reduced NULL,
```

```
        reduced-ex BIND-int-fuzz
        }


-- ********************************************************************************
-- Description of an allele - a form of a gene.
--
-- Field description for BIND-allele
-- *********************************
-- id               = a reference to the DNA genome sequence "gene"
-- names            = the name(s) of this allele
-- form             = what is the experimental form of this allele
-- copy-num         = how many copies of this experimental form are present
-- biosource        = the type of genetic element that contains this allele
--                    and where it came from. E.g. from a chromosome or from
--                    a plasmid from a specific strain.
-- descr            = optional text description
-- source           = empirical evidence references
-- ********************************************************************************


BIND-allele ::= SEQUENCE {
        id BIND-id,
        names SEQUENCE OF VisibleString OPTIONAL,
        form BIND-allele-exp-obj-choice,
        copy-num BIND-allele-copy-num,
        biosource BioSource,
        descr VisibleString OPTIONAL,
        source BIND-pub-set OPTIONAL
        }


-- ********************************************************************************
-- The choice for experimental form of an allele
--
-- Field description for BIND-allele-exp-object-choice
-- ***************************************************
-- genomic  = the allele is not changed from the sequenced genome
--            (referred to by BioSource)
-- deletion = the allele has been deleted
-- mutation = the allele has been mutated from the genome sequence, explicit
--            description of new allele sequence is attached.
-- ********************************************************************************


BIND-allele-exp-obj-choice ::= CHOICE {
        not-specified NULL,
        genomic NULL,
        knock-out NULL,
        mutation BIND-object
        }


-- ********************************************************************************
-- Type and some results from a genetic experiment.
--
-- Field description for BIND-genetic-experiment
-- *********************************************
-- result-phenotype = the resulting phenotype
-- type             = the type of genetic interaction experiment
-- dep-changes      = this experimental result depends on these changes to the
--                    background (e.g. gene disruptions of A and B show a
--                    synthetic lethality only when C is knocked out)
--                    These are changes OTHER than those described for
--                    experimental form of A and B.
-- descr            = optional text description
-- molecule-present = a molecule that was present during this experiment
--                    (e.g. a drug or aptamer)
-- environment      = description of the environmental conditions
--                    (e.g. temperature)
-- ********************************************************************************


BIND-genetic-experiment ::= SEQUENCE {
        result SEQUENCE OF BIND-genetic-exp-result,
        type BIND-genetic-exp-system,
        dep-changes SEQUENCE OF BIND-allele-change OPTIONAL,
        descr VisibleString OPTIONAL,
        molecule-present SEQUENCE OF BIND-object OPTIONAL,
        environment SEQUENCE OF BIND-param OPTIONAL
        }
```

```
-- ********************************************************************************
-- Result of a genetic experiment.
--
-- Field description for BIND-genetic-exp-result
-- *********************************************
-- phenotype  = the resulting phenotype
-- relation   = the phenotypic relation between A, B and AB in this experiment
--              if known
-- background = the organism and strain background resulting from the
--              experiment (background of the progeny)
-- ********************************************************************************


BIND-genetic-exp-result ::= SEQUENCE {
        phenotype BIND-phenotype,
        relation BIND-genetic-relation OPTIONAL,
        background BIND-genetic-background
        }


-- ********************************************************************************
-- A genetic relation.  The relation between experimental form of A, B and the
-- result of genetic experiment AB.
--
-- Field description for BIND-genetic-relation
-- *******************************************
-- a-wild-type  = TRUE if A is wild-type
-- b-wild-type  = TRUE if B is wild-type
-- a-eq-b       = TRUE if phenotype of A is the same as that of B
-- ab-phenotype = the phenotype of the result of the genetic experiment
-- ********************************************************************************


BIND-genetic-relation ::= SEQUENCE {
        a-wild-type BOOLEAN,
        b-wild-type BOOLEAN,
        a-eq-b BOOLEAN,
        ab-phenotype BIND-genetic-ab-phenotype
        }

BIND-genetic-ab-phenotype ::= CHOICE {
        wild-type NULL,
        a-type NULL,
        b-type NULL,
        novel BIND-synthetic-phenotype
        }


-- ********************************************************************************
-- Description of a synthetic phenotype
--
-- Field description for BIND-synthetic-phenotype
-- **********************************************
-- modulation-pct-a = modulation percentage of phenotype for A
-- modulation-pct-b = modulation percentage of phenotype for B
--                    NOTE: 0% of phenotype is no parent phenotype, above 100%
--                    is an enhanced phenotype (more than parent phenotype)
-- mix              = a mix of A and B phenotypes
-- novel            = a completely new phenotype (see phenotype in
--                    BIND-genetic-exp-result)
-- ********************************************************************************


BIND-synthetic-phenotype ::= CHOICE {
        modulation-pct-a INTEGER,
        modulation-pct-b INTEGER,
        mix BIND-mixed-phenotype,
        novel NULL
        }


-- ********************************************************************************
-- Description of a mixed phenotype
--
-- Field description for BIND-mixed-phenotype
-- ******************************************
-- a-pct = percentage of phenotype for A
-- b-pct = percentage of phenotype for B
-- ********************************************************************************
```

```
BIND-mixed-phenotype ::= SEQUENCE {
        a-pct INTEGER,
        b-pct INTEGER
        }


-- ********************************************************************************
-- Enumerated list of possible genetic experiments
--
-- These experiments involve a genetic cross between two parents, each
-- containing a set of alleles in their genotype.  The offspring have a
-- known mixture of alleles from the parents which can confer a phenotype
-- change.  Synthetic phenotypes occur e.g. when a double mutant phenotype
-- is more than the sum phenotypes of each single mutant.
--
-- Field description for BIND-genetic-exp-system
-- *********************************************
-- synthetic-lethal        = lethal phenotype observed in synthetic experiment
-- synthetic-growth-defect = negative phenotype change observed
-- synthetic-enhancement   = positive phenotype change observed
-- suppression             = a phenotype is suppressed when an allele is added
-- epistasis               = mutant gene causing phenotype is acting upstream
--                           in the genetic network
-- non-complementation     = two mutations fail to complement but are in
--                           different genes.
-- ********************************************************************************


BIND-genetic-exp-system ::= INTEGER {
        not-specified (0),
        synthetic-lethal (1),
        synthetic-growth-defect (2),
        synthetic-enhancement (3),
        suppression (4),
        epistasis (5),
        non-complementation (6),
        other (255)
        }


-- ********************************************************************************
-- The background of a genetic experiment. E.g. organism and strain info.
--
-- Field description for BIND-genetic-background
-- *********************************************
-- org     = the organism and strain of this background
-- ploidy  = the ploidy number of the organism (haploid=1, diploid=2, etc.)
-- ploidy-diff = records if certain chromosomes are present in different
--               copy numbers than the organism ploidy (e.g. trisomy 21)
-- changes = the changes in this background from the genome referenced by
--           'org BioSource' in this object (e.g. his-)
--           Either a simple description or an explicit standard form.
--           NOTE: these changes are not required for the genetic interaction,
--           they merely document changes from a common background.
-- ********************************************************************************


BIND-genetic-background ::= SEQUENCE {
        org BioSource,
        ploidy BIND-int-fuzz,
        ploidy-diff BIND-ploidy-diff OPTIONAL,
        changes BIND-genetic-background-choice
        }

BIND-ploidy-diff ::= SEQUENCE {
        chromosome INTEGER,
        copy-num BIND-int-fuzz
        }

BIND-genetic-background-choice ::= CHOICE {
        descr VisibleString,
        standard SEQUENCE OF BIND-allele-change
}


-- ********************************************************************************
-- Description of an allele change from wild-type (the genome form)
--
-- Field description for BIND-allele-change
```

```
-- ****************************************
-- old-form = a reference to the allele in the BioSource that is changed
-- new-form = a description of the changed allele
-- descr    = optional text description
-- ****************************************************************************


BIND-allele-change ::= SEQUENCE {
        old-form BIND-allele,
        new-form BIND-allele,
        descr VisibleString OPTIONAL
        }


-- ************************************************************
-- * Interaction conserved sequence comment (in BIND-descr) *
-- ************************************************************

-- ****************************************************************************
-- Conserved sequence comment set
--
-- Only relevant for biological sequences.
-- (e.g. Derived from multiple alignment information)
--
-- Field description for BIND-cons-seq-set
-- ****************************************
-- a = conserved sequence comment for molecule A
-- b = conserved sequence comment for molecule B
-- ****************************************************************************


BIND-cons-seq-set ::= SEQUENCE {
        a BIND-conserved-seq OPTIONAL,
        b BIND-conserved-seq OPTIONAL
        }


-- ****************************************************************************
-- Conserved sequence comment
--
-- Alignment data is not stored here, only the conclusion from it.
--
-- Field description for BIND-conserved-seq
-- ****************************************
-- seq-el   = sequence elements that have been shown to be conserved
-- descr    = text description (e.g. method of determining conserved sequence)
-- other-db = reference to a conserved sequence database (e.g. BLOCKS)
-- source   = empirical evidence
-- ****************************************************************************


BIND-conserved-seq ::= SEQUENCE {
        seq-el Seq-loc,
        descr VisibleString OPTIONAL,
        other-db SEQUENCE OF BIND-other-db OPTIONAL,
        source BIND-pub-set OPTIONAL
        }


-- ********************************************************************
-- * Binding location on molecules in an interaction (in BIND-descr) *
-- ********************************************************************

-- ****************************************************************************
-- Binding location on a BIND-object
--
-- Field description for BIND-loc
-- ******************************
-- detailed = atomic level detail of interaction sites
-- general  = sequence element level description of interaction sites
-- source   = empirical evidence
-- ****************************************************************************


BIND-loc ::= SEQUENCE {
        detailed Biostruc OPTIONAL,
        general BIND-loc-gen OPTIONAL,
        source BIND-pub-set OPTIONAL
        }
```

```
-- ********************************************************************************
-- General binding location on a BIND-object
--
-- Field description for BIND-loc-gen
-- ***********************************
-- a-sites = list of binding sites on molecule A
-- b-sites = list of binding sites on molecule B
-- bound   = list of sequence elements from A and B that are bound together
-- ********************************************************************************


BIND-loc-gen ::= SEQUENCE {
        a-sites BIND-loc-site-set OPTIONAL,
        b-sites BIND-loc-site-set OPTIONAL,
        bound SEQUENCE OF BIND-loc-pair OPTIONAL
        }


-- ********************************************************************************
-- A set of BIND-loc-site objects
-- ********************************************************************************


BIND-loc-site-set ::= SEQUENCE {
        max-slid BIND-Seq-loc-id,
        sites SEQUENCE OF BIND-loc-site
        }


-- ********************************************************************************
-- A graph describing which sites on A bind to which sites on B
--      BIND-loc-site objects are nodes (vertices) in the graph
--      BIND-loc-pair objects are edges in the graph
--
-- Field description for BIND-loc-site
-- ***********************************
-- slid     = internal ID of this sequence element
--              (0..n within interaction record)
-- site     = a sequence element (point or interval)
-- sub-unit = if molecule A or B is a molecular complex, specifies which
--              sub-unit the site is on.
-- descr    = description of this binding site
-- source   = empirical evidence
-- ********************************************************************************


BIND-loc-site ::= SEQUENCE {
        slid BIND-Seq-loc-id,
        site Seq-loc,
        sub-unit BIND-complex-subunit OPTIONAL,
        descr VisibleString OPTIONAL,
        source BIND-pub-set OPTIONAL
        }


-- ********************************************************************************
-- A pair of binding sites that are bound to each other
--
-- Field description for BIND-loc-pair
-- ***********************************
-- a-slid = the Seq-loc pointed to by this ID is connected to...
-- b-slid = the Seq-loc pointed to by this ID
-- source = empirical evidence
-- ********************************************************************************


BIND-loc-pair ::= SEQUENCE {
        a-slid BIND-Seq-loc-id,
        b-slid BIND-Seq-loc-id,
        source BIND-pub-set OPTIONAL
        }

BIND-Seq-loc-id ::= INTEGER


-- ********************************************************************************
-- Unique reference to a site on a biomolecule
--
-- Field description for BIND-loc-site-ref
-- ***************************************
-- from-iid = interaction that contains the binding site
```

```
-- molecule = the molecule in the interaction (from-iid) that this site is on
--            (A or B)
-- site     = BIND-Seq-loc-id (slid) of the binding site
-- ********************************************************************************


BIND-loc-site-ref ::= SEQUENCE {
        from-iid Interaction-id,
        molecule ENUMERATED {
                a (1),
                b (2)
                },
        site BIND-Seq-loc-id
        }


-- ***********************************************
-- * Interaction chemical action (in BIND-descr) *
-- ***********************************************

-- ********************************************************************************
-- A set of chemical actions
--
-- Chemical actions mediated by a molecule (A or B) in the
-- interaction (a set because a kinase may phosphorylate a protein multiple
-- times)
--
-- Field description for BIND-action-set
-- *************************************
-- max-iaid = the highest iaid used in this set
-- actions  = set of BIND-action objects
-- ********************************************************************************


BIND-action-set ::= SEQUENCE {
        max-iaid Internal-action-id,
        actions SEQUENCE OF BIND-action
        }


-- ********************************************************************************
-- A chemical action
--
-- Field description for BIND-action
-- *********************************
-- iaid      = internal action id (unique identifier for this action in a set)
--             (0..n within interaction record)
-- descr     = text description (e.g. if 'other' is specified for type)
-- direction = direction of chemical action
-- type      = type of chemical action
-- result    = the product(s) of this chemical action
--  NOTE    this field holds the exact chemical form that is produced, and is
--          used by reference by the next interaction acting on the "product".
--          For a biopolymer this holds the atoms&bonds representation of the
--          molecule.
-- diff       = the atomic level detail of differences created by this action
-- signal     = more general kinetics, signal transduction
-- kinetics   = chemical action kinetics
-- conditions = link to experimental conditions used to observe this action,
--              e.g. if there were multiple experimental conditions stored in
--               this interaction record and this action was only seen using
--               some of them.
-- sub-unit-a  = if molecule A is a molecular complex, specifies the sub-unit
--               to which this chemical action applies
-- sub-unit-b  = if molecule B is a molecular complex, specifies the sub-unit
--               to which this chemical action applies
-- action-site = sites on molecule A and B that are involved in the action
-- other-db    = reference to a database of chemical actions
-- source      = empirical evidence
-- ********************************************************************************


BIND-action ::= SEQUENCE {
        iaid Internal-action-id,
        descr VisibleString OPTIONAL,
        direction BIND-direction,
        type BIND-action-type,
        result SEQUENCE OF BIND-result-object OPTIONAL,
        diff Biostruc-feature-set OPTIONAL,
        signal BIND-signal OPTIONAL,
        kinetics BIND-kinetics OPTIONAL,
```

```
        conditions SEQUENCE OF BIND-condition-ref OPTIONAL,
        sub-unit-a BIND-complex-subunit OPTIONAL,
        sub-unit-b BIND-complex-subunit OPTIONAL,
        action-sites SEQUENCE OF BIND-action-site OPTIONAL,
        other-db BIND-other-db OPTIONAL,
        source BIND-pub-set OPTIONAL
        }

Internal-action-id ::= INTEGER

BIND-result-object ::= SEQUENCE {
        irid Internal-result-id,
        object BIND-object
        }

Internal-result-id ::= INTEGER


-- ********************************************************************************
-- Unique reference to a chemical action or a chemical action result object
--
-- Field description for BIND-action-ref
-- ************************************
-- from-iid = interaction that contains the chemical action
-- action   = internal action ID number of the chemical action
-- irid     = reference to chemical result from action
--              (0..n within interaction record)
-- ********************************************************************************


BIND-action-ref ::= SEQUENCE {
        from-iid Interaction-id,
        action Internal-action-id,
        irid Internal-result-id OPTIONAL
        }


-- ********************************************************************************
-- A direction
-- ********************************************************************************


BIND-direction ::= ENUMERATED {
        none (0),
        a-to-a (1),
        a-to-b (2),
        b-to-b (3),
        b-to-a (4),
        other (255)
        }


-- ********************************************************************************
-- The type of action and object of that action
--
-- Action type        object of that action
-- add                BIND-object or NULL
-- remove             BIND-object or NULL
-- cut-seq            Seq-loc or NULL
--
-- Field description for BIND-action-type
-- **************************************
-- -not-specified = action is not-specified (unknown)
-- -none = no chemical action, but e.g. kinetics information needs to be
--         stored (action is known to be nothing)
-- -add = add an object (e.g. phosphate) to an object
-- -remove = remove an object (e.g. phosphate) from an object
-- -bond-break = non-sequence cut action - e.g. small molecule hydrolysis
-- -cut-seq = cut a sequence, location may be specified after which
--            (right-side) the cut is made.
--            (e.g. restriction enzyme)
-- -change-conformation = a change in conformation of a molecule
--         (e.g. hck protein -> phosphorylation causes conformational change)
-- -change-configuration = a change in configuration of a molecule
--         (e.g. by an epimerase or isomerase)
-- -change-other = another type of change (e.g. metal ion exchange)
-- -other = another action
--
-- Field description for BIND-action-object
-- ****************************************
-- none     = no action object
```

```
-- object   = any BIND-object that is added or removed (e.g. phosphate)
-- location = location where a sequence was cut
-- ********************************************************************************


BIND-action-type ::= CHOICE {
        not-specified NULL,
        none NULL,
        add BIND-action-object,
        remove BIND-action-object,
        bond-break NULL,
        cut-seq BIND-action-object,
        change-conformation NULL,
        change-configuration NULL,
        change-other NULL,
        other NULL
        }

BIND-action-object ::= CHOICE {
        none NULL,
        object BIND-object,
        location Seq-loc
        }


-- ********************************************************************************
-- A chemical signal description
--
-- A more general notion of kinetics describing signal transduction.
--
-- Field description for BIND-signal
-- *********************************
-- action    = signal modification
-- direction = direction of signal
-- factor    = the factor of the amplification or the repression
-- descr     = text description (e.g. if 'other' is specified)
-- source    = empirical evidence
-- ********************************************************************************


BIND-signal ::= SEQUENCE {
        action ENUMERATED {
                none (0),
                amplify (1),
                repress (2),
                other (255)
                },
        direction BIND-direction,
        factor RealVal-Units OPTIONAL,
        descr VisibleString OPTIONAL,
        source BIND-pub-set OPTIONAL
        }


-- ********************************************************************************
-- Chemical kinetics and thermodynamics data
--
-- Field description for BIND-kinetics
-- ***********************************
-- descr           = optional text description of this object
-- kf              = forward reaction rate
-- kr              = reverse reaction rate
-- kd              = dissociation constant of interaction
-- ka              = association constant of interaction
-- keq             = equilibrium constant of interaction
-- km              = Michaelis-Menten constant
-- vmax            = max. velocity of reaction
-- rxn-order       = reaction order
-- conc-a          = concentration of molecule A
-- conc-b          = concentration of molecule B
-- conc-a-bound    = concentration of molecule A that is bound
-- conc-b-bound    = concentration of molecule B that is bound
-- conc-a-unbound  = concentration of molecule A that is not bound
-- conc-b-unbound  = concentration of molecule B that is not bound
-- enz-activity-amp-factor = amplification factor for enzyme kinetic activity
--                           (what factor is enzyme activity changed?)
-- temp            = temperature of the interaction system (observed)
-- ph              = pH of the interaction system
-- half-life-a     = 1/2 life for molecule A
-- half-life-b     = 1/2 life for molecule B
-- buffer          = buffer text description
```

```
-- delta-g         = delta G (delta Gibbs free energy)
-- delta-s         = delta S (delta entropy)
-- delta-h         = delta H (delta enthalpy)
-- heat-capacity-a = heat capacity of molecule A
-- heat-capacity-b = heat capacity of molecule B
-- other           = any other related values (e.g. k1, k2...)
-- source          = empirical evidence
-- ********************************************************************************


BIND-kinetics ::= SEQUENCE {
        descr VisibleString OPTIONAL,
        kf RealVal-Units OPTIONAL,
        kr RealVal-Units OPTIONAL,
        kd RealVal-Units OPTIONAL,
        ka RealVal-Units OPTIONAL,
        keq RealVal-Units OPTIONAL,
        km RealVal-Units OPTIONAL,
        vmax RealVal-Units OPTIONAL,
        rxn-order RealVal-Units OPTIONAL,
        conc-a RealVal-Units OPTIONAL,
        conc-b RealVal-Units OPTIONAL,
        conc-a-bound RealVal-Units OPTIONAL,
        conc-b-bound RealVal-Units OPTIONAL,
        conc-a-unbound RealVal-Units OPTIONAL,
        conc-b-unbound RealVal-Units OPTIONAL,
        enz-activity-amp-factor RealVal-Units OPTIONAL,
        temp RealVal-Units OPTIONAL,
        ph RealVal-Units OPTIONAL,
        half-life-a RealVal-Units OPTIONAL,
        half-life-b RealVal-Units OPTIONAL,
        buffer VisibleString OPTIONAL,
        delta-g RealVal-Units OPTIONAL,
        delta-s RealVal-Units OPTIONAL,
        delta-h RealVal-Units OPTIONAL,
        heat-capacity-a RealVal-Units OPTIONAL,
        heat-capacity-b RealVal-Units OPTIONAL,
        other SEQUENCE OF BIND-kinetics-other OPTIONAL,
        source BIND-pub-set OPTIONAL
        }

BIND-kinetics-other ::= SEQUENCE {
        descr VisibleString,
        value RealVal-Units
        }


-- ********************************************************************************
-- Sites on molecule A and B that are involved in the action.
--
-- The action may be 'performed' by the site or be affected by the action.
--
-- Field description for BIND-action-site
-- ************************************
-- a = site on molecule A
-- b = site on molecule B
-- ********************************************************************************


BIND-action-site ::= SEQUENCE {
        a BIND-action-site-ref,
        b BIND-action-site-ref
        }


-- ********************************************************************************
-- A site on a molecule, by reference or by value
--
-- Field description for BIND-action-site-ref
-- ******************************************
-- slid = A reference to a predefined binding site stored in the BIND-loc-gen
--        object for this interaction
-- site = A description of a site on a molecule if one can not be referenced
--        from the BIND-loc-gen object
-- ********************************************************************************


BIND-action-site-ref ::= CHOICE {
        slid BIND-Seq-loc-id,
        site BIND-loc-site-set
        }
```

```
-- ************************************************************************
-- * Interaction - chemical state for molecule A and/or B (in BIND-descr) *
-- ************************************************************************

-- ************************************************************************
-- Chemical state and required chemical state for molecules A and B
--
-- The chemical state in the BIND-state-descr is "the chemistry" of A or B
-- in this particular molecular interaction.  The chemistry is referred to by
-- reference, typically to another interaction record's
-- interaction:action:result which encodes a BIND-object that is the
-- "bio-processed" form of A or B used in this interaction.
--
-- Field description for BIND-state-descr
-- **************************************
-- a = list of possible chemical states for A that can undergo this
--     interaction
-- a-required-state = the state that A in the above list of possible states
--                    is required to assume before interaction takes place.
-- b = list of possible chemical states for B that can undergo this
--     interaction
-- b-required-state = the state that B in the above list of possible states
--                    is required to assume before interaction takes place.
-- NOTE: multiple required states are only used if molecule A or B is a
--       molecular complex and the state of more than one sub-unit needs
--       to be denoted as required.
-- ************************************************************************


BIND-state-descr ::= SEQUENCE {
        a BIND-state-set OPTIONAL,
        a-required-state SEQUENCE OF BIND-required-state OPTIONAL,
        b BIND-state-set OPTIONAL,
        b-required-state SEQUENCE OF BIND-required-state OPTIONAL
        }


-- ************************************************************************
-- A set of chemical states
--
-- e.g. multiple phosphorylations on a protein; all of which may be active
--      in this interaction record.
--
-- Field description for BIND-state-set
-- ************************************
-- max-isid = highest Internal-state-id used in this set
-- states   = list of possible chemical states
-- ************************************************************************


BIND-state-set ::= SEQUENCE {
        max-isid Internal-state-id,
        states SEQUENCE OF BIND-state
        }

Internal-state-id ::= INTEGER


-- ************************************************************************
-- Unique reference to a chemical state
--
-- Field description for BIND-state-ref
-- ************************************
-- from-iid = interaction that contains the chemical state
-- molecule = the molecule in the interaction (from-iid) that is in this state
--            (A, B)
-- state    = Internal-state-id (isid) of the chemical state
-- ************************************************************************


BIND-state-ref ::= SEQUENCE {
        from-iid Interaction-id,
        molecule ENUMERATED {
                a (1),
                b (2),
                other (255)
                },
        state Internal-state-id
        }
```

```
-- ********************************************************************************
-- A chemical state
--
-- Points to the chemistry of a molecule, if known, by reference to an
-- interaction:action with an explicit 'result' field.
-- This allows conversion of a sequence to chemistry with modifications -
-- can describe a protein that has been phosphorylated at a certain residue,
--
-- Here we can exactly state the chemistry of a molecule as it is found in
-- the cell, even though the top BIND-object may only refer to the GI.
--
-- Field description for BIND-state
-- ********************************
-- isid = Internal-state-id (unique for each state in a BIND-state-set)
--        (0..n within interaction record)
-- activity-level = general activity of molecule
-- cause = sequence of actions from this or other Interactions that bring
--         about this state (and the resulting object that is this state, if
--         result objects have been specified in chemical-action, via an
--         optional IRID in the BIND-action-ref).
--         References an external interaction:action uniquely.
--         The "cause" is really the Interaction:action pair elsewhere
--         in the database that is the most recent step in the biochemical
--         conversion that forms the biochemical entity in A or B.
--         Action and state are peer BIND-descr tags, this allows
--         a reference to causal 'action' within the chemical state.
-- descr    = text description (e.g. method used to determine this state)
-- source   = empirical evidence for this state
-- sub-unit = if molecule A or B is a molecular complex, specifies the
--            sub-unit to which this state applies
-- ********************************************************************************


BIND-state ::= SEQUENCE {
        isid Internal-state-id,
        activity-level ENUMERATED {
                not-specified (0),
                inactive (1),
                very-low (2),
                low (3),
                medium (4),
                medium-high (5),
                high (6),
                very-high (7),
                extreme (8),
                active (9),
                other (255)
                },
        cause SEQUENCE OF BIND-action-ref OPTIONAL,
        descr VisibleString OPTIONAL,
        source BIND-pub-set OPTIONAL,
        sub-unit BIND-complex-subunit OPTIONAL
        }


-- ********************************************************************************
-- A required chemical state for interaction to take place
--
-- The state in the state set that is required for the interaction to take
-- place. Uniquely locates a chemical state within this interaction record
-- by Internal-state-id.
--
-- Field description for BIND-required-state
-- *****************************************
-- isid   = Internal-state-id of the required state.  Points to
--          one chemical state in the BIND-state-set in the same record
-- descr  = description of state requirement
-- source = empirical evidence
-- ********************************************************************************


BIND-required-state ::= SEQUENCE {
        isid Internal-state-id,
        descr VisibleString OPTIONAL,
        source BIND-pub-set OPTIONAL
        }


-- /!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\
```

```
-- **********************
-- * Molecular-Complex *
-- **********************

-- ********************************************************************************
-- A set of Molecular Complexes
--
-- Field description for BIND-Complex-set
-- ***************************************
-- date      = date this set of records was collected
-- database  = name and description of database that this set comes from
-- complexes = set of molecular complex records
-- ********************************************************************************


BIND-Complex-set ::= SEQUENCE {
        date Date OPTIONAL,
        database BIND-Database-site OPTIONAL,
        complexes SEQUENCE OF BIND-Molecular-Complex
        }


-- ********************************************************************************
-- A molecular complex record
--
-- A collection of one or more interactions that form a complex.
-- i.e. two or more BIND-objects that operate as a unit.  It is a
-- useful shorthand when defining BIND pathways.
--
-- A molecular complex can also be defined if the interactions in it are not
-- completely known. Create interactions with molecule A as the sub-unit of
-- the complex and molecule B as 'not-specified' for all of the known
-- sub-units.
--
-- Field description for BIND-Molecular-Complex
-- ********************************************
-- date              = date of record entry
-- updates           = a list of updates for the record
-- mcid              = molecular complex accession number.
-- descr             = text description of complex (e.g. ribosome)
-- sub-num           = total number of sub-units in this complex
-- sub-units         = collection of BIND-objects in the complex (meant to be
--                     a non-redundant list)
-- interaction-list  = list of interactions in this complex
-- ordered           = TRUE if order of interactions is known and
--                     interaction-list is ordered in this way
-- complex-topology  = a connectivity graph of the complex topology
--                     with BIND-objects as nodes
-- source            = empirical evidence references
-- authors           = person(s) who authored this record
-- division          = interaction is part of a record collection/division
--                     (i.e. a satellite BIND database)
-- priv              = TRUE if this complex is private
-- sub-unit-type     = number and type of subunits in this complex
-- extref            = external reference(s) to an e.g. other complex database
-- ********************************************************************************


BIND-Molecular-Complex ::= SEQUENCE {
        date Date,
        updates SEQUENCE OF BIND-update-object OPTIONAL,
        mcid Molecular-Complex-id,
        descr VisibleString OPTIONAL,
        sub-num BIND-mol-sub-num,
        sub-units SEQUENCE OF BIND-mol-object,
        interaction-list SEQUENCE OF Interaction-id,
        ordered BOOLEAN DEFAULT FALSE,
        complex-topology SEQUENCE OF BIND-mol-object-pair OPTIONAL,
        source BIND-pub-set,
        authors SEQUENCE OF BIND-author OPTIONAL,
        division BIND-Rec-coll-descr OPTIONAL,
        priv BOOLEAN DEFAULT FALSE,
        sub-unit-type SEQUENCE OF BIND-mol-sub-unit-type OPTIONAL,
        extref SEQUENCE OF BIND-other-db OPTIONAL
        }

Molecular-Complex-id ::= INTEGER


-- ********************************************************************************
-- Uniquely specify a molecule within a complex (a sub-unit)
```

```
--
-- Field description for BIND-complex-subunit
-- ******************************************
-- mcid  = molecular complex ID of the complex that contains the sub-unit
-- bmoid = BIND molecular complex object ID of the sub-unit
-- ********************************************************************************


BIND-complex-subunit ::= SEQUENCE {
        mcid Molecular-Complex-id,
        bmoid BIND-mol-object-id
        }


-- ********************************************************************************
-- Copy number of a sub unit in a Molecular Complex
--
-- This number can be an integer or a fuzzy integer.
--
-- Field description for BIND-mol-sub-num
-- **************************************
-- num      = integer number of sub-units
-- num-fuzz = fuzzy integer number of sub-units (e.g. microtubule, virus)
-- ********************************************************************************


BIND-mol-sub-num ::= CHOICE {
        num INTEGER,
        num-fuzz Int-fuzz
        }


-- ********************************************************************************
-- A subunit of a molecular complex.
-- Each sub-unit must be its own BIND-complex-subunit, even repeated sub-units
--
-- Field description for BIND-mol-object
-- *************************************
-- bmoid    = internal ID BIND-object
--              (0..n within complex record)
-- sub-unit = a sub-unit in a molecular complex
-- num      = OBSOLETE: moved to BIND-mol-sub-unit-type.stoichiometry
-- state    = which BIND-object in BIND is this sub-unit
-- ********************************************************************************


BIND-mol-object ::= SEQUENCE {
        bmoid BIND-mol-object-id,
        sub-unit BIND-object,
        num BIND-mol-sub-num OPTIONAL,       --OBSOLETE, do not use (Dec.01.2001)
        state BIND-mol-object-source OPTIONAL
        }


-- ********************************************************************************
-- The source of a molecular complex sub-unit within BIND
--
-- Field description for BIND-mol-object-source
-- ********************************************
-- a      = if this molecular complex is from object A of the given IID
-- b      = if this molecular complex is from object B of the given IID
-- result = if this molecular complex is from a chemical action result
-- ********************************************************************************


BIND-mol-object-source ::= CHOICE {
        a Interaction-id,
        b Interaction-id,
        result BIND-action-ref
        }


-- ********************************************************************************
-- Description of a type of sub-unit and its stoichiometry.  E.g. complex
-- of 8 subunits is A4B3C1
-- (e.g. colour on the molecular complex topology graph)
--
-- Field description for BIND-mol-sub-unit-type
-- ********************************************
-- descr         = description of this type of sub-unit
-- stoichiometry = stoichiometry of this sub-unit
```

```
-- ********************************************************************************


BIND-mol-sub-unit-type ::= SEQUENCE {
        descr VisibleString,
        stoichiometry BIND-mol-stoich OPTIONAL
        }


-- ********************************************************************************
-- Stoichiometry of a type of sub-unit of a molecular complex.
--
-- Field description for BIND-mol-stoich
-- *************************************
-- num    = stoichiometry (number of this type of sub-unit)
-- bmoids = which sub-units in the complex are of this type. 2 CASES:
--          Exact stoichiometry is known (num.num chosen):
--            All sub-units must be represented under
--            BIND-Molecular-Complex.sub-units
--          Fuzzy stoichiometry is known (num.num-fuzz chosen):
--            Not all sub-units have to be represented under
--            BIND-Molecular-Complex.sub-units, but the ones that are must be
--            referenced in the bmoids field.
-- ********************************************************************************


BIND-mol-stoich ::= CHOICE {
        num BIND-mol-sub-num,
        bmoids SEQUENCE OF BIND-mol-object-id
        }


-- ********************************************************************************
-- The connections in a molecular complex topology map between molecular
-- complex subunits.  BIND-mol-object-id's are used to reference
-- BIND-mol-object subunits.
-- Note: when using BIND-mol-object-pair to describe the complex topology,
--       each sub-unit in the complex must be explicitly described in its
--       own BIND-mol-object (even the repeated sub-units).
--
-- Field description for BIND-mol-object-pair
-- ******************************************
-- a-bmoid    = the sub-unit pointed to by this ID is connected to...
-- b-bmoid    = the sub-unit pointed to by this ID
-- iid        = the interaction-id that this link may refer to.
-- instance fields are OBSOLETE: moved to BIND-mol-sub-unit-type
-- ********************************************************************************


BIND-mol-object-pair ::= SEQUENCE {
        a-bmoid BIND-mol-object-id,
        b-bmoid BIND-mol-object-id,
        a-instance INTEGER OPTIONAL,  --OBSOLETE, do not use (Dec.01.2001)
        b-instance INTEGER OPTIONAL,  --OBSOLETE, do not use (Dec.01.2001)
        iid Interaction-id OPTIONAL
        }

BIND-mol-object-id ::= INTEGER


-- /!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\/!\
-- **********************************
-- * Biomolecular chemical pathway *
-- **********************************

-- ********************************************************************************
-- A set of Pathways
--
-- Field description for BIND-Pathway-set
-- **************************************
-- date     = date this set of records was collected
-- database = name and description of database that this set comes from
-- pathways = set of pathway records
-- ********************************************************************************


BIND-Pathway-set ::= SEQUENCE {
        date Date OPTIONAL,
        database BIND-Database-site OPTIONAL,
        pathways SEQUENCE OF BIND-Pathway
        }
```

```
-- ********************************************************************************
-- A pathway record.
--
-- A collection of one or more interactions that form a pathway.
-- i.e. Two or more BIND-objects that are generally free from each
-- other, but form a network of interactions.
--
-- Field description for BIND-pathway
-- **********************************
-- date     = date of record entry
-- updates  = a list of updates for the record
-- pid      = pathway accession number
-- pathway  = a collection of interactions and signal modification objects
-- descr    = description of a pathway
-- source   = empirical evidence references
-- authors  = person(s) who authored this record
-- division = interaction is part of a record collection/division
--            (i.e. a satellite BIND database)
-- priv     = TRUE if this pathway is private
-- extref   = external reference(s) to an e.g. other pathway database
-- ********************************************************************************


BIND-Pathway ::= SEQUENCE {
        date Date,
        updates SEQUENCE OF BIND-update-object OPTIONAL,
        pid Pathway-id,
        pathway SEQUENCE OF Interaction-id,
        descr BIND-path-descr,
        source BIND-pub-set,
        authors SEQUENCE OF BIND-author OPTIONAL,
        division BIND-Rec-coll-descr OPTIONAL,
        priv BOOLEAN DEFAULT FALSE,
        extref SEQUENCE OF BIND-other-db OPTIONAL
        }

Pathway-id ::= INTEGER


-- ********************************************************************************
-- Pathway description
--
-- Field description for BIND-path-descr
-- *************************************
-- descr              = text description of pathway
--         (e.g. lipid biosynthesis, bacterial chemotaxis, Ras pathway, etc.)
-- cell-cycle         = stage of a cell cycle that this pathway is in effect
-- pathological-state = disease manifestation if this pathway is present
-- pathway-actions    = list of chemical actions that occur in the pathway.
--                      Specified actions can only come from interactions in
--                      this pathway.
-- phenotype          = the normal phenotype of this pathway
-- ********************************************************************************


BIND-path-descr ::= SEQUENCE {
        descr VisibleString OPTIONAL,
        cell-cycle SEQUENCE OF BIND-cellstage OPTIONAL,
        pathological-state SEQUENCE OF BIND-pathol-state OPTIONAL,
        pathway-actions SEQUENCE OF BIND-action-ref OPTIONAL,
        phenotype SEQUENCE OF BIND-phenotype OPTIONAL
        }


-- ********************************************************************************
-- Pathological state
--
-- Description of a disease that is caused by a change in an interaction in a
-- 'physiologically normal' pathway.
--
-- Field description for BIND-pathol-state
-- ***************************************
-- pathway-iid  = interaction in the physiologically normal pathway
-- cause        = change to the interaction that cause the pathological state
--                Choice of interaction was ablated or replaced by another
--                interaction
-- pathol-state = names(s) of the pathological state
-- database     = database(s) that contain information about this disease
--                (e.g. OMIM at http://www.ncbi.nlm.nih.gov/Omim/)
```

```
-- phenotype    = phenotype of this disease state
-- descr        = description of the pathological state
-- source       = empirical evidence references
-- ********************************************************************************


BIND-pathol-state ::= SEQUENCE {
        pathway-iid Interaction-id,
        cause BIND-pathol-state-cause,
        pathol-state SEQUENCE OF VisibleString,
        database SEQUENCE OF BIND-other-db OPTIONAL,
        phenotype SEQUENCE OF BIND-phenotype OPTIONAL,
        descr VisibleString OPTIONAL,
        source BIND-pub-set OPTIONAL
        }

BIND-pathol-state-cause ::= CHOICE {
        destroyed NULL,
        replaced-by Interaction-id
        }


-- ********************************************************************************
-- Phenotype
--
-- Field description for BIND-phenotype
-- ************************************
-- trait     = trait (e.g. colour)
-- name      = name of phenotype (e.g. red)
-- wild-type = TRUE if this phenotype is wild-type with respect to the genome
-- descr     = optional text description
-- db-links  = links to other databases
--             E.g.
--             GO - (http://www.geneontology.org) can be referenced in the
--                  form GO:0003684 damaged DNA binding
--             DGAP - (http://dgap.harvard.edu)
--
--             E.g. GO can be referenced using a BIND-other-db object like so:
--             BIND-other-db ::= SEQUENCE {
--                dbname "GO",
--                intp 0003684       (use INTEGER field wherever possible)
--             }
--
-- source    = empirical evidence references
-- ********************************************************************************


BIND-phenotype ::= SEQUENCE {
        trait VisibleString OPTIONAL,
        name VisibleString,
        wild-type ENUMERATED {
                not-specified (0),
                true (1),
                false (2)
                },
        descr VisibleString OPTIONAL,
        db-links SEQUENCE OF BIND-other-db OPTIONAL,
        source BIND-pub-set OPTIONAL
}

END
```

```
-- $Id: bindprof.asn,v 1.2 2002/02/19 16:07:59 gbader Exp $
-- ****************************************************************************
--
--  Biomolecular Interaction Network Database (BIND)
--  Data Specification
--
--  Sequence Profile Data Structures
--
--
--  Authors: Gary D. Bader,       Christopher W.V. Hogue
--           bader@mshri.on.ca    hogue@mshri.on.ca
--
--  Thanks to SLRI staff, especially Ian Donaldson for invaluable discussion.
--
--  Hogue Lab - University of Toronto Biochemistry Department and the
--  Samuel Lunenfeld Research Institute, Mount Sinai Hospital
--  http://bioinfo.mshri.on.ca  hogue@mshri.on.ca
--
--  REVISIONS
--  Revision 1.0 - March 29, 2000
--
--  ftp://bioinfo.mshri.on.ca/pub/BIND/Spec/bindprof.asn for latest revision.
--
--
--  NOTE:  This specification is in a variant of ASN.1 1990 that may not
--         be compatible with newer ASN.1 tools.  This specification also
--         depends on public domain specifications available from the
--         U.S. National Center for Biotechnology Information (NCBI)
--         ftp://ncbi.nlm.nih.gov/toolbox/ncbi_tools/
--         http://www.ncbi.nlm.nih.gov/Toolbox/
--
--
--  Copyright Notice:
--
--  Copyright 2000 Mount Sinai Hospital (MSH)
--
--  This program is free software; you can redistribute it and/or
--  modify it under the terms of the GNU General Public License as
--  published by the Free Software Foundation; either version 2 of
--  the License, or any later version.
--
--  This program is distributed in the hope that it will be useful,
--  but WITHOUT ANY WARRANTY; without even the implied warranty of
--  MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
--
--  See the GNU General Public License for more details.
--
--  You should have received a copy of the GNU General Public License
--  along with this program; if not, write to the
--         Free Software Foundation, Inc.,
--         59 Temple Place, Suite 330, Boston, MA
--         02111-1307  USA
--  or visit http://www.gnu.org/copyleft/gpl.html
--
--  SPECIAL EXCEPTIONS
--
--  As a special exception, Mount Sinai Hospital gives permission to
--  link this program with the following non-GPL programs or libraries,
--  and distribute the resulting executable, without including the source
--  code for these in the source distribution:
--
--  a) CodeBase 6.5 or greater from Sequiter Software Inc.
--
--  b) The C or C++ interface to Oracle 8.x or greater, from
--     Oracle Corporation or IBM DB2 UDB.
--
-- ****************************************************************************

BIND-Profile DEFINITIONS ::=
BEGIN

EXPORTS BIND-profile;

IMPORTS RealVal-Units FROM BIND;


-- ****************************************************************************
-- A profile (or position specific score matrix) for a sequence.
--
-- This profile structure is an ASN.1 version of the PROSITE profile
-- data structure.  The PROSITE homepage is http://www.expasy.ch/prosite/
```

```
--
-- For a well written, full description of the original data structure,
-- by Phillipp Bucher, see the document http://www.expasy.ch/txt/profile.txt
-- ********************************************************************************


BIND-profile ::= SEQUENCE {
        general-spec BIND-p-gs,
        disjoint BIND-p-disjoint,
        norm SEQUENCE OF BIND-p-norm OPTIONAL,
        cut-off SEQUENCE OF BIND-p-cutoff OPTIONAL,
        defaults SEQUENCE OF BIND-p-defaults OPTIONAL,
        im SEQUENCE OF BIND-p-im
        }

-- General specifications for the profile

BIND-p-gs ::= SEQUENCE {
        alphabet VisibleString,
        length INTEGER OPTIONAL,
        topology ENUMERATED {
                linear (1),
                circular (2)
                } DEFAULT linear,
        begin INTEGER OPTIONAL,
        end INTEGER OPTIONAL,
        log-base RealVal-Units OPTIONAL,
        p0 RealVal-Units OPTIONAL,
        random-model SEQUENCE OF RealVal-Units OPTIONAL
        }

-- Disjointednes definition for multiple profile-sequence alignments
-- One globally optimal alignment can be specified

BIND-p-disjoint ::= SEQUENCE {
        definition ENUMERATED {
                unique (1),    --zero parameters
                protect (2),   --2 int parameters
                other (255)    --provide a name
                },
        parameters SEQUENCE OF BIND-p-param OPTIONAL,
        other-name VisibleString OPTIONAL
        }

-- Profile parameter

BIND-p-param ::= SEQUENCE {
        param BIND-p-param-val,
        descr VisibleString OPTIONAL
        }

-- Profile parameter value

BIND-p-param-val ::= CHOICE {
        integer INTEGER,
        real RealVal-Units,
        low-value NULL
        }

-- Score normalization instructions

BIND-p-norm ::= SEQUENCE {
        function ENUMERATED {
                linear (1),    --2 real parameters
                gle-zscore (2),        --5 real parameters
                other (255)    --provide a name
                },
        other-name VisibleString OPTIONAL,
        parameters SEQUENCE OF BIND-p-param OPTIONAL,
        mode-nr INTEGER OPTIONAL,
        priority INTEGER OPTIONAL,
        text VisibleString OPTIONAL
        }

-- Recommended cut-off value(s) for scores in the profile

BIND-p-cutoff ::= SEQUENCE {
        rscore INTEGER,
        level INTEGER OPTIONAL,
        text VisibleString OPTIONAL,
        norm SEQUENCE OF BIND-p-co-norm OPTIONAL
```

```
        }

-- Cut-off value in normalized score units

BIND-p-co-norm ::= SEQUENCE {
        nscore RealVal-Units,
        mode-nr INTEGER
        }

-- Defaults for position specific profile parameters

BIND-p-defaults ::= SEQUENCE {
        sy-i VisibleString DEFAULT "-",       --one character only
        sy-m VisibleString DEFAULT "X",       --one character only
        params-i SEQUENCE OF BIND-p-score-i,
        params-m SEQUENCE OF BIND-p-score-m
        }

-- Position specific profile scores for insert positions

BIND-p-score-i ::= SEQUENCE {
        name ENUMERATED {
                b0 (1),     -- initiation scores
                b1 (2),
                e0 (3),     -- termination scores
                e1 (4),
                bm (5),     -- state transition scores from e.g.
                bi (6),     --  state B to state M
                bd (7),
                be (8),
                mm (9),
                mi (10),
                md (11),
                me (12),
                im (13),
                ii (14),
                id (15),
                ie (16),
                dm (17),
                di (18),
                dd (19),
                de (20),
                i (21),     -- insert extension scores
                i0 (22)
                },
        value BIND-p-param-val
        }

-- Position specific profile scores for match positions

BIND-p-score-m ::= SEQUENCE {
        name ENUMERATED {
                m (1),      -- match extension scores
                m0 (2),
                d (3)       -- deletion extension score
                },
        value BIND-p-param-val
        }

-- The profile matrix itself

BIND-p-im ::= SEQUENCE {
        type-params BIND-p-im-type,
        sy VisibleString OPTIONAL     --one character only
        }

BIND-p-im-type ::= CHOICE {
        i SEQUENCE OF BIND-p-score-i,
        m SEQUENCE OF BIND-p-score-m
        }


END
```

COPYRIGHT RELEASE AUTHORIZATIONS