VERSION 13.0

User's Guide

PATHWAY TOOLS

Pathway Tools User's Guide, Version 13.0

Copyright © 1996, 1999-2009 SRI International, 1997-1999 DoubleTwist, Inc.
All rights reserved. Printed in the U.S.A.

We gratefully acknowledge contributions to Pathway Tools, used by permission, from:
Jeremy Zucker, Harvard Medical School
The Laboratory of Christos Ouzounis, European Bioinformatics Institute

DoubleTwist is a registered trademark of DoubleTwist, Inc.
Medline is a registered trademark of the National Library of Medicine.
Oracle is a registered trademark of Oracle Corporation.
MySQL ® is a registered trademark of MySQL AB in the United States, the European Union and other countries.
The Generic Frame Protocol is Copyright © 1996, The Board of Trustees of the Leland Stanford Junior University and SRI International. All Rights Reserved.
The Pathway Tools Software is Copyright © 1997-1999 DoubleTwist, Inc., SRI International 1996, 1999-2007. All Rights Reserved.
The EcoCyc Database is Copyright © SRI International 1996, 1999-2009, Marine Biological Laboratory 1996-2001, DoubleTwist Inc. 1997-1999. All Rights Reserved.
The MetaCyc Database is Copyright © SRI International 1999-2009, Marine Biological Laboratory 1998-2001, DoubleTwist Inc. 1998, 1999. All Rights Reserved.
Allegro Common Lisp is Copyright © 1985-2009, Franz Inc. All Rights Reserved.
All other trademarks are property of their respective owners.

Any rights not expressly granted herein are reserved.

This product may include data from BIND (`http://blueprint.org/bind/bind.php`) to which the following two notices apply:
(1) Bader GD, Betel D, Hogue CW. (2003) BIND: the Biomolecular Interaction Network Database. Nucleic Acids Res. 31(1):248-50 PMID: 12519993
(2) This data 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.

The dictionary used by the Pathway Tools spell checker was provided courtesy of www.biology-online.org.

SRI International
333 Ravenswood Ave.
Menlo Park, CA 94025 U.S.A.
biocyc-support@ai.sri.com

Contents

Contents **v**

Preface **xvii**

1 Introduction to the Pathway Tools Software **1**

1.1 Databases and Tools 1

1.1.1 Databases 2

1.1.2 Pathway Tools 3

2 Invoking Pathway Tools **5**

2.1 Pathway Tools Init File 6

2.1.1 Parameters for Running in Web Server Mode 6

2.1.2 Additional Parameters for Running in Web Server Mode 6

2.1.3 Parameters Used when Accessing PGDBs within a MySQL or Oracle Server 8

2.1.4 Parameters Associated with User Accounts Defined for a Pathway Tools Web Server 8

2.2 X-Windows Basics 9

2.3 Running Pathway Tools from the Command Line 9

2.3.1 Command Line Arguments 11

3 Pathway/Genome Navigator **17**

3.1 Using the Mouse To Navigate and Issue Commands 17

3.2 Menus and Dialogs 18

3.2.1 Menu Bar . . . 18
3.2.2 Single-Choice Menus . . . 18
3.2.3 Multiple-Choice Menus . . . 18
3.2.4 Dialogs . . . 18
3.2.5 Aborting Out of Menus and Dialogs . . . 19
3.3 Organism Summary Display . . . 19
3.4 Single Organism Display . . . 20
3.5 Notion of Current Organism . . . 21
3.6 Examples . . . 22
3.7 Query Facilities . . . 23
3.7.1 Direct Queries . . . 24
3.7.2 Indirect Queries: Navigation . . . 25
3.7.3 History List . . . 26
3.7.4 Programmatic Queries . . . 26
3.8 Object Displays and Queries . . . 27
3.8.1 Shared Display Characteristics . . . 28
3.8.2 Systems Level Overviews . . . 31
3.8.3 Pathways . . . 54
3.8.4 Reactions . . . 58
3.8.5 Proteins . . . 59
3.8.6 RNAs . . . 62
3.8.7 Genes . . . 62
3.8.8 Compounds . . . 64
3.8.9 Transcription Units . . . 65
3.8.10 Genome Browser . . . 66
3.8.11 Displaying External Tracks on the Genome Browser . . . 69
3.8.12 Comparative Genome Browser . . . 72
3.9 Advanced Queries using the BioVelo querying language . . . 73

3.10 Miscellaneous Commands and Tools . . . 73
3.10.1 Home . . . 73
3.10.2 Back . . . 73
3.10.3 Forward . . . 73
3.10.4 History . . . 74
3.10.5 Next Answer . . . 74
3.10.6 Clone . . . 74
3.10.7 Print . . . 74
3.10.8 Tools Menu . . . 75
3.10.9 Help . . . 80
3.10.10 Exiting Pathway Tools . . . 80
3.10.11 User Preferences . . . 80
3.10.12 Keyboard Shortcuts . . . 84
3.10.13 Tips . . . 85
3.11 Comparative Operations . . . 86
3.11.1 Global Comparative Analyses . . . 86
3.11.2 Sequential Comparison . . . 89
3.11.3 Parallel Comparison . . . 91

4 The Import/Export Facility 97
4.1 Pathway Import/Export . . . 98
4.2 SBML Export . . . 99
4.3 Genbank Format Export . . . 99
4.4 Linking Table Export . . . 100
4.5 Full Flat File Dump . . . 100
4.6 Frame Import/Export . . . 100
4.6.1 Frame Export . . . 101
4.6.2 Frame Import . . . 103
4.6.3 Supported file formats for frame import and export: . . . 106

4.7 Importing citations from PubMed . . . 107

5 Database Sharing Via the PGDB Registry 109

5.1 Downloading PGDBs from the Registry . . . 110

5.2 Publishing PGDBs in the Registry . . . 110

5.2.1 Details of what happens during each step: . . . 111

5.2.2 Preliminary Step: Setting Preferences . . . 112

5.2.3 Registering a PGDB . . . 114

5.2.4 About Click-Through Licenses . . . 116

6 PathoLogic: Automated Creation of Pathway/Genome Databases 117

6.1 Overview of PathoLogic Execution . . . 118

6.1.1 Database Generation Perspective . . . 118

6.1.2 PathoLogic Operation . . . 119

6.2 PathoLogic Input File Formats . . . 120

6.2.1 File genetic-elements.dat . . . 120

6.2.2 The PathoLogic File Format . . . 121

6.2.3 GenBank File Format . . . 124

6.2.4 Specifying Gene Ontology Terms . . . 127

6.2.5 Directory Structure for a PGDB . . . 128

6.3 Creating a Pathway/Genome Database . . . 129

6.3.1 Invoke PathoLogic . . . 129

6.3.2 Create New Organism . . . 130

6.3.3 Create genetic-elements.dat File . . . 132

6.3.4 Specify Reference PGDB . . . 132

6.3.5 Trial Parse . . . 134

6.3.6 Build Pathway/Genome Database . . . 136

6.3.7 Metabolic Pathway Prediction . . . 137

6.3.8 Update Build for New Annotation . . . 143

6.3.9 Adding or Replacing a Sequence File . . . 145
6.3.10 PGDB Housekeeping Tasks . . . 145
6.4 Refining the PGDB . . . 148
6.4.1 Refine: Assign Probable Enzymes . . . 148
6.4.2 Refine: Rerun Name Matcher . . . 152
6.4.3 Refine: Rescore Pathways . . . 152
6.4.4 Refine: Create Protein Complexes . . . 153
6.4.5 Refine: Assign Modified Proteins . . . 156
6.4.6 Refine: Predict Transcription Units (Operons) . . . 157
6.4.7 Refine: Transport Inference Parser . . . 160
6.4.8 Refine: Pathway Hole Filler . . . 173
6.4.9 Refine: Update Cellular Overview . . . 185
6.4.10 Refine: Run Consistency Checker . . . 185
6.5 Output from PathoLogic . . . 185
6.5.1 Pathway/Genome Database . . . 185
6.5.2 Pathway Predictor Summary Pages . . . 185
6.5.3 Interpretation of the PathoLogic Summary Pages . . . 186
6.6 PathoLogic Batch Mode . . . 189
6.6.1 File organism-params.dat . . . 190
6.7 Ongoing PGDB Curation . . . 192
6.8 Updating for New Versions of MetaCyc . . . 192
6.9 Suggested PGDB Release Procedures . . . 196

7 Editing Pathway/Genome Databases 199
7.1 Overview of the Editors . . . 199
7.1.1 Right-Click on Object Handles to Edit Existing Objects . . . 200
7.1.2 Available Pathway/Genome Editors . . . 200
7.1.3 Saving Edits . . . 204
7.2 Frame Data Model . . . 204

7.2.1 Frames . . . 205
7.2.2 Relationship of Frames to Pathway Tools Displays . . . 207
7.3 Editors . . . 207
7.3.1 Gene Editor . . . 208
7.3.2 Editing Transcriptional Regulatory Information . . . 209
7.3.3 Intron Editor . . . 213
7.3.4 RNA Editor . . . 214
7.3.5 Protein Editors . . . 214
7.3.6 Pathway Editors . . . 218
7.3.7 Reaction Editor . . . 227
7.3.8 Chemical Compound Editor . . . 234
7.3.9 Marvin Compound Structure Editor . . . 234
7.3.10 JME Compound Structure Editor . . . 238
7.3.11 Compound Duplicate Checker . . . 240
7.3.12 MDL Molfile Import/Export . . . 243
7.3.13 Regulation Editor . . . 243
7.3.14 Publication Editor . . . 244
7.4 Editing Examples . . . 245
7.4.1 Changing a Gene's Functional Annotation . . . 245
7.4.2 Changing the Annotation for an Enzyme Gene . . . 246
7.4.3 Entering a New Pathway . . . 247
7.5 Advanced Editing Topics . . . 249
7.5.1 Saving DB Updates . . . 249
7.5.2 Editing Restrictions . . . 253
7.5.3 Right-Button Menu . . . 254
7.5.4 Object Names Visible to PGDB Users . . . 255
7.5.5 Frame Name Conventions . . . 256
7.5.6 Special Formatting of Text . . . 258

7.5.7 Citations . . . 259
7.5.8 Frame References . . . 260
7.5.9 Creating Links Between a PGDB and External Databases . . . 260
7.5.10 Modified Proteins . . . 265
7.5.11 Bulk Creation of New Frames . . . 265
7.5.12 When Pathway Tools Makes Recommendations . . . 266
7.5.13 Curator Crediting . . . 266
7.6 Curation Tutorial . . . 269
7.6.1 Introduction . . . 269
7.6.2 Before You Create a Pathway . . . 270
7.6.3 Composing a Pathway . . . 274
7.6.4 Exporting a pathway . . . 291

8 Pathway Tools Web Server Operation 295
8.1 Operational Procedures . . . 297
8.2 Customizing Pathway Tools Web Server Pages . . . 298
8.2.1 JavaScript Customization . . . 300
8.2.2 CSS Style Sheet Customization . . . 300
8.2.3 Adding, Removing, Modifying the Top Menu Bar . . . 301
8.3 Template Files and HTML Virtual Inclusion . . . 302
8.4 Setting Up BLAST Access . . . 305
8.4.1 Web Server Log File . . . 305
8.5 Creating Links to Pathway Tools Pages . . . 306
8.5.1 Omics Viewer . . . 306
8.5.2 Pathway Images . . . 307
8.6 Web Accounts . . . 308
8.6.1 How To Use The System for Everyday Users . . . 309
8.7 Installation and Operation of Web Accounts System By System Administrator 311
8.7.1 Configure the Pathway Tools Initialization File . . . 311

8.7.2 Disabling Web Accounts . . . 314
8.7.3 Background: Which files are required to be included? . . . 314
8.7.4 Technical Details . . . 317
8.7.5 How to Debug an Installation . . . 317
8.7.6 Default HTML directory not set properly . . . 317
8.7.7 Summary . . . 319
8.8 Troubleshooting . . . 319

9 Troubleshooting **321**
9.1 Frequently Asked Questions . . . 321
9.2 Reporting Problems . . . 321

10 Guide to the Pathway Tools Schema **323**
10.1 Slots Valid in Multiple Classes . . . 324
10.1.1 Common-Name . . . 325
10.1.2 Synonyms . . . 325
10.1.3 Abbrev-Name . . . 325
10.1.4 Names . . . 326
10.1.5 Comment . . . 326
10.1.6 Citations . . . 326
10.2 Class Compounds . . . 327
10.2.1 Appears-In-Left-Side-Of, Appears-In-Right-Side-Of . . . 327
10.2.2 Aromatic-Rings . . . 327
10.2.3 Atom-Charges . . . 327
10.2.4 Charge . . . 327
10.2.5 Chemical-Formula . . . 328
10.2.6 Display-Coords-2D . . . 328
10.2.7 Gibbs-0 . . . 328
10.2.8 Molecular-Weight . . . 328

10.2.9 N-Name, N-1-Name, N+1-Name . . . 328
10.2.10 Regulates . . . 329
10.2.11 Smiles . . . 329
10.2.12 Structure-Atoms . . . 329
10.2.13 Structure-Bonds . . . 329
10.3 Class DNA-Binding-Sites . . . 329
10.3.1 Abs-Center-Pos . . . 329
10.3.2 Involved-in-Regulation . . . 330
10.3.3 Site-Length . . . 330
10.4 Class Enzymatic Reactions . . . 330
10.4.1 Enzyme . . . 330
10.4.2 Required-Protein-Complex . . . 330
10.4.3 Reaction . . . 330
10.4.4 Regulated-By . . . 331
10.4.5 Cofactors, Prosthetic-Groups . . . 331
10.4.6 Alternative-Substrates, Alternative-Cofactors . . . 331
10.4.7 Reaction-Direction . . . 332
10.5 Class Genes . . . 332
10.5.1 Left-End-Position, Right-End-Position . . . 333
10.5.2 Centisome-Position . . . 333
10.5.3 Transcription-Direction . . . 333
10.5.4 Product . . . 333
10.5.5 Interrupted? . . . 333
10.6 Class Organisms . . . 334
10.6.1 PGDB-Authors . . . 334
10.6.2 PGDB-Copyright . . . 334
10.6.3 PGDB-Footer-Citation . . . 335
10.6.4 PGDB-Home-Page . . . 335

10.6.5 PGDB-Name . . . 335
10.6.6 PGDB-Unique-ID . . . 335
10.6.7 Strain-Name . . . 335
10.6.8 Contact-Email . . . 335
10.6.9 Genome . . . 336
10.7 Class Pathways . . . 336
10.7.1 Pathway-Interactions . . . 336
10.7.2 Predecessors . . . 336
10.7.3 Reaction-List . . . 337
10.7.4 Hypothetical-Reactions . . . 337
10.7.5 Assume-Unique-Enzymes . . . 337
10.7.6 Enzyme-Use . . . 338
10.7.7 Enzymes-Not-Used . . . 338
10.7.8 Primaries . . . 339
10.7.9 Species . . . 339
10.7.10 Disable Display . . . 339
10.7.11 Super-Pathways . . . 339
10.7.12 Sub-Pathways . . . 339
10.7.13 Pathway-Links . . . 340
10.7.14 Polymerization-Links . . . 340
10.7.15 Class-Instance-Links . . . 340
10.7.16 Layout-Advice . . . 340
10.8 Class Polypeptides . . . 341
10.8.1 Gene . . . 341
10.8.2 Features . . . 341
10.8.3 Splice-Form-Introns . . . 341
10.9 Class Promoters . . . 341
10.9.1 Absolute-Plus-1-Pos . . . 342

10.9.2 Binds-Sigma-Factor . . . 342
10.9.3 Component-Of . . . 342
10.9.4 Minus-35-Left, Minus-35-Right, Minus-10-Left, Minus-10-Right . . . 342
10.10 Class Complexes . . . 342
10.10.1 Components . . . 343
10.11 Class Proteins . . . 343
10.11.1 Component-Of . . . 343
10.11.2 DNA-Footprint-Size . . . 343
10.11.3 GO-Terms . . . 343
10.11.4 Locations . . . 344
10.11.5 Modified-Form . . . 344
10.11.6 Molecular-Weight-KD . . . 344
10.11.7 Molecular-Weight-Seq . . . 344
10.11.8 Molecular-Weight-Exp . . . 344
10.11.9 pI . . . 344
10.11.10 Regulates . . . 344
10.11.11 Species . . . 345
10.11.12 Unmodified-Form . . . 345
10.12 Class Protein-Features . . . 345
10.12.1 Attached-Group . . . 345
10.12.2 Feature-Of . . . 346
10.12.3 Left-End-Position . . . 346
10.12.4 Possible-Feature-States . . . 346
10.12.5 Residue-Number . . . 346
10.12.6 Right-End-Position . . . 346
10.13 Class Reactions . . . 347
10.13.1 EC-Number . . . 348
10.13.2 Official-EC? . . . 348

10.13.3 Left, Right . . . 348
10.13.4 Substrates . . . 348
10.13.5 Enzymes-Not-Used . . . 349
10.13.6 DeltaG0 . . . 349
10.13.7 Spontaneous? . . . 349
10.13.8 Species . . . 349
10.13.9 Cannot-Balance? . . . 349
10.14Class Transcription-Units . . . 350
10.14.1 Components . . . 350
10.14.2 Extent-Unknown? . . . 350
10.15Class tRNAs . . . 350
10.15.1 Anticodon . . . 350
10.15.2 Codons . . . 350
10.16Class Regulation . . . 351
10.16.1 Associated-Binding-Site . . . 351
10.16.2 Mechanism . . . 352
10.16.3 Mode . . . 352
10.16.4 Regulated-Entity . . . 352
10.16.5 Regulator . . . 352
10.16.6 Antiterminator-Start-Pos, Antiterminator-End-Pos, Anti-Antiterm-Start-Pos, Anti-Antiterm-End-Pos, Pause-Start-Pos, Pause-End-Pos . 353

Bibliography **355**

Index **357**

Preface

This document will familiarize you with the Pathway Tools software and the Pathway/Genome Databases (PGDBs) that are available from SRI International. Pathway Tools operates across one or more PGDBs, including the EcoCyc *E. coli* DB and the MetaCyc DB. Each database describes the biochemical pathways and genomes of a single organism. All PGDBs are managed by an object-oriented database system called 'Ocelot'.

Pathway Tools contains several component software modules including:

- Pathway/Genome Navigator, the primary tool by which users visualize, query, and analyze the information contained within a PGDB.
- PathoLogic, which creates new PGDBs from annotated genomes. PathoLogic contains several computational inference tools including a predictor for metabolic pathways and for operons.
- Pathway/Genome Editors, which modify the information contained within PGDBs, such as allowing users to add in-house proprietary data about a pathway or gene of interest. The Editors include a general tool for database browsing and editing, called the GKB Editor.

Pathway Tools is most commonly provided in two different configurations. Both configurations are available for the Macintosh, for the PC running Windows, and for the PC running Linux. On all platforms the software can run as both an application on the computer's desktop, and as a Web server for the intranet or the Internet.

BioCyc Configuration: Includes the Pathway/Genome Navigator and one or more PGDBs, such as EcoCyc and MetaCyc. In this configuration, PGDBs are physically included inside the binary executable program, and are available for read-only access. In addition, PGDBs that have been created using the full Pathway Tools configuration can be imported into the BioCyc configuration.

Full Pathway Tools Configuration: Includes the Navigator, PathoLogic, and Editors as well as one or more PGDBs. This configuration allows the creation of new PGDBs,

which can be stored either as disk files or in an Oracle or MySQL database server. Newly created PGDBs can be updated by users. Use of an Oracle or MySQL database is recommended if multiple users will be updating a given PGDB.

The PGDBs present in your installation of Pathway Tools will depend on exactly which PGDBs your organization has ordered. Other PGDBs are available in addition to EcoCyc and MetaCyc. The data content of different PGDBs varies. The EcoCyc DB generally contains more extensively curated data than do other PGDBs. For example, EcoCyc contains extensive data on operons, promoters, transcription factors, transporters, and paralogous gene groups, which other PGDBs tend to lack.

In this user's guide:

Chapter 1 provides an overview of the pathway bioinformatics tools and databases: what they are, the information they contain, and some of the things they help you to do.

Chapter 2 describes how to invoke Pathway Tools and how to access the Pathway Tools Web server.

Chapter 3 outlines the Pathway/Genome Navigator. It begins by providing some background information on using the mouse and some examples, to get you started quickly. It then outlines how to use the Navigator to select one or more databases and, subsequently, query, visualize, and analyze the information contained within these databases.

Chapter 4 covers the Pathway Tools import/export facilities.

Chapter 5 covers the PGDB Sharing System, which allows you to share PGDBs you have created with other Pathway Tools users, or to access PGDBs created by other Pathway Tools users.

Chapter 6 describes how to use PathoLogic to create new Pathway/ Genome Databases.

Chapter 7 describes how to use the Pathway/Genome Editors to interactively update Pathway/Genome Databases.

Chapter 8 describes the optional Web Accounts system available to the Pathway Tools Web server.

Chapter 9 outlines what to do if you encounter problems with the functionality of any of the tools or with information contained within any of the provided databases.

Additional Pathway Tools Publications and Web Sites

For more detailed information regarding different aspects of this software, consult the following resources (predominantly, EcoCyc-related publications).

1. SRI Web site containing a variety of information on the Pathway Tools and its component software systems, including
2. Pathway Tools information site containing samples of Pathway Tools file formats, example Lisp queries to PGDBs, and more, at
   `http://bioinformatics.ai.sri.com/ptools/`
3. Generic Frame Protocol (the API for Pathway Tools PGDBs) specification at
   `http://www.ai.sri.com/~gfp/spec/paper/paper.html`
4. GKB Editor User's Guide at
   `http://www.ai.sri.com/~gkb/user-man.html`
5. Scope and contents of the EcoCyc data, discussed in [11].
6. Ontology (representations) used in the Pathway Tools, discussed in [5], [10].
7. Pathway/Genome Navigator, described in [5].
8. Possible computational uses of the EcoCyc database, discussed in [7].
9. Ocelot database management system used in Pathway Tools, described in [5].

Many of these publications are available through the World Wide Web at
`http://www.ai.sri.com/pkarp/bio.html#SelPubs`

Chapter 1

Introduction to the Pathway Tools Software

Bioinformatics has been largely sequence centric and focused on building databases to store the large amounts of sequence data being generated by the Human Genome Project and microbial genome initiatives. In addition, this discipline has involved the development of tools for visualizing, querying, and analyzing sequence data. Most bioinformatics analyses focus largely on identifying genes and predicting the function of encoded gene products. However, the focus is moving from the consideration of individual genes toward examining how sets of genes and their corresponding gene products interact with each other, as well as other molecular players, to form the pathways that are the biochemical basis of cellular behavior.

Pathway bioinformatics involves the creation of databases and algorithms for storing and computing biochemical pathways. Pathway databases typically encode interactions between gene products and other compounds to organize reactions into pathways. In addition, pathway bioinformatics involves the development of tools to query, visualize, and analyze this information. One of the primary advantages of pathway bioinformatics is that it provides descriptions of how the molecular components encoded in a genome interact with each other and with other molecular players to form the biochemical basis of cellular function.

1.1 Databases and Tools

Figure 1.1 provides an overview of the Pathway Tools software. The software components operate on one or more Pathway/Genome Databases (PGDBs) (see the central rectangle). A PGDB is a collection of information that describes some of the biochemical pathways and genes of (typically) a single organism. The pathways might include, for example,

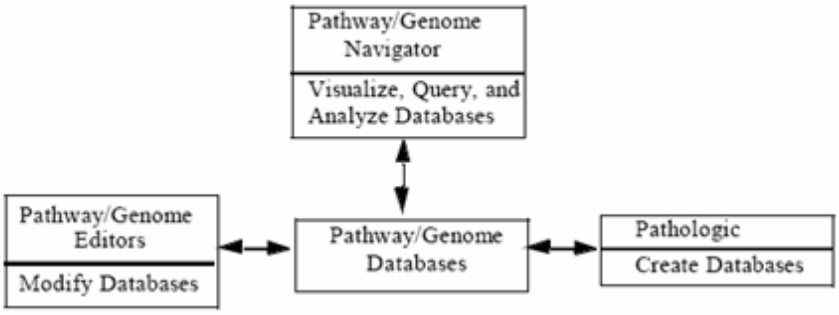


Figure 1.1: Pathway Tools and related databases

signal transduction, transport, and metabolic pathways. The genes may constitute an entire genome, or a subset of genes of interest, for example, those genes that encode a set of pathways of interest.

1.1.1 Databases

The schema of a PGDB describes pathways in terms of five biological entities (see A):

- Union of described pathways (metabolic overview)
- Individual pathways
- Reactions that compose these pathways
- Compounds that participate in these reactions
- Enzymes (a subset of the gene products) that catalyze these reactions

Genomes are described in terms of three biological entities (see B):

- Genomic maps of sequenced genetic element(s)
- Their constituent genes
- Corresponding gene products

The gene products that are enzymes provide the primary link between the representation of the genomes and that of the metabolic pathways (see C). However, PGDBs typically describe all known or predicted gene products of the corresponding organism, not just

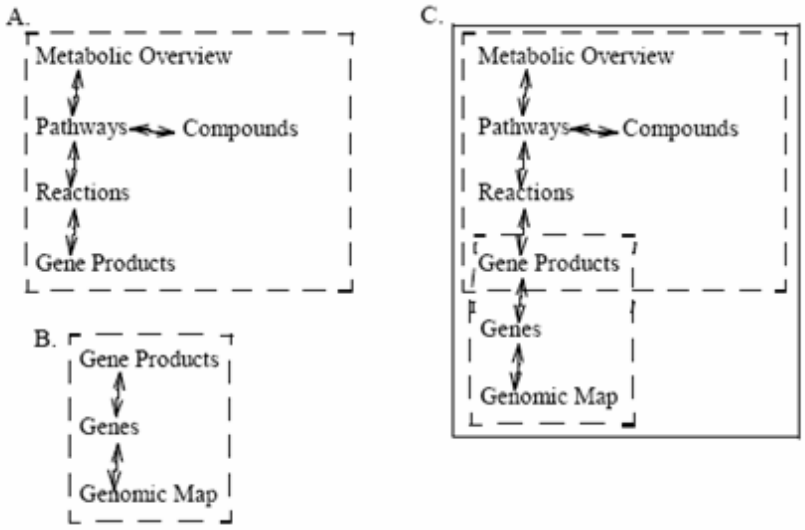


Figure 1.2: Representation of pathways and genomes

those genes whose products are known or predicted to be enzymes — that is, the space of enzymes is a subset of the space of gene products. Incorporated pathways include those of biosynthesis, degradation, energy production, and intermediary metabolism, for compounds such as amino acids, carbohydrates, fatty acids, nucleotides, and enzyme cofactors. Different PGDBs vary as to the degree to which they contain transmembrane transport, genetic regulatory, or signal transduction pathways (with the exception of the EcoCyc *E. coli* database, which contains all the preceding pathway types).

1.1.2 Pathway Tools

The primary tool with which the user may visualize, query, and analyze the information contained within databases is the Pathway/Genome Navigator. PathoLogic allows you to computationally predict the pathways of an organism from its annotated genome. The predicted pathways and the description of the genome are then combined to create a new PGDB. In addition, with specialized, intuitive, and interactive editing tools that are not included in the configuration documented here, you can modify existing databases; for example, update information about a particular gene, open a reading frame (ORF), or enter information about a newly discovered pathway. You can also use these editing tools to link any piece of information within a given database to one or more pieces of information contained within a Web-accessible external database, for example, a gene expression relational database management system.

Chapter 2

Invoking Pathway Tools

This chapter provides the information necessary to get you started using Pathway Tools. It is assumed that Pathway Tools and one or more accompanying database(s) have been successfully installed on your machine and are ready to run.

Pathway Tools can be operated in two different modes. Each of these modes offers some software capabilities that the other mode does not provide — for details see `http://biocyc.org/desktop-vs-web-mode.shtml`. For example, Web mode provides many comparative genomics operations that desktop mode does not provide.

Desktop mode: The user interacts with the Pathway Tools through the UNIX X Window System (X-Windows) or through Microsoft Windows. All the Pathway Tools software components are available in desktop mode: the Navigator, Editing Tools, and PathoLogic.

Web mode: Pathway Tools operates as a Web server that can support simultaneous queries from many different users through the network. Only the Navigator operates in Web mode, but not any of the editing tools nor PathoLogic.

For technical reasons, when the Pathway Tools are running in Web mode under Unix, they nonetheless require access to the X-Windows system, meaning that a proper UNIX X-Windows environment must be established, even though users do not interact with the Pathway Tools through that environment. Therefore, we begin with a brief overview of how to define the X-Windows environment for use by Pathway Tools.

Starting with Pathway Tools version 10.0, the most important configuration and initialization parameters can now be set in an Init File, which will simplify future upgrades. This file and its parameters are discussed in Section 2.1.

Microsoft Windows users should read the installation guide for information on starting and stopping Pathway Tools. Please see `http://bioinformatics.ai.sri.com/`

ptools/installation-guide/released/index.html.

Sections 8.1, 8.2, and 8.4 are for all users. Otherwise, the remainder of this chapter is for UNIX users.

2.1 Pathway Tools Init File

After a successful installation of Pathway Tools, a file with initialization parameters will have been written, retaining values from a prior installation, if possible. The resulting file can be found at ptools-local/ptools-init.dat. It can be customized and edited further. The syntax and the parameters in the file are described here. In UNIX, many of these parameters can be overridden by the "Command Line Arguments" described in Section 2.3.1.

Each parameter needs to be specified on its own line, with the name of the parameter followed by a separator consisting of one space character, which is followed by the value, which is usually a text string or an integer. Lines beginning with the hash character # are ignored and used for comments.

2.1.1 Parameters for Running in Web Server Mode

Setting these parameters is MANDATORY if you run Pathway Tools in Web server mode.

WWW-Server-Hostname Hostname of the machine on which Pathway Tools is running, which Pathway Tools will insert in URLs within the WWW pages that it generates.

WWW-Server-Port Port number on which Pathway Tools should listen for Web requests, and which Pathway Tools will insert in URLs within the Web pages that it generates. The standard port for Web servers is 80, but this may require root privileges for starting up the Pathway Tools Web server.

2.1.2 Additional Parameters for Running in Web Server Mode

Setting these parameters is OPTIONAL if you run Pathway Tools in Web server mode.

Support-Email-Address If you wish for users of your Web site to direct technical-support emails to you instead of the makers of BioCyc and Pathway Tools, specify your support email address and it will be inserted into your Web site.

WWW-Publish Specify which organisms should be published by the Web server. Users may publish all PGDBs on a site by using the `all` value. Possible values are `all`, or a list of organism IDs delimited by +'s. Example of the latter: `ECOLI+META+VCHO`.

WWW-Server-Proxy-Port Needed only for a Proxy Web Server setup. This is the port number on which another Web server, such as Apache, will listen for requests to forward them to the Pathway Tools Web server. When using this parameter, the other Web server needs to be configured to actually forward the requests.

WWW-Html-Root-Dir The root directory from which the static pages are served. The directory specified should not end with a slash or backslash. If this paramater is not specified, its value default to the subdirectory `htdocs` under the directory used to install Pathway Tools.

WWW-Server-User If a WWW-Server-Port smaller than 1024 is specified, on UNIX, the root user needs to start Pathway Tools. After the Web server starts listening, it will switch to the user account indicated here, to reduce security risks.

WWW-Site-Name A string giving the name by which the current Pathway Tools based Web site is known, e.g., "BioCyc" or "MouseCyc", or whatever name should be used in Web pages generated by Pathway Tools.

WWW-Use-Gzip Set this to "Y" (no quotes) to have the web server use `gzip` to compress CSS (.css), JavaScript (.js), and HTML (.html) files served by the web server. This applies only for non-Windows systems (e.g., Linux). The files are compressed and thereafter served compressed without using `gzip`. That is, the file system works as a cache for compressed files. A file is re-compressed if the original is more recent than the compressed version. You can therefore change any of the static files while the server is running and it will be re-compressed if the server needs it. A value of "N" says not to compress while the server is running (the default). Note that you can compress all the CSS, JavaScript, and HTML files with `gzip` offline and turn off this option. The compressed files would still be served. In particular, it is possible to serve compressed files under Windows, but you need to compress the files offline.

WWW-Quick-Search-Textfield-Label If WWW-Quick-Search-Textfield-Label is set to a given string, then that string will appear to the left of the Quick Search text box at the bottom of every Web server Web page. The default is to have it display, "Quick Search". The default value for this parameter is `Quick_Search`. If you want the name to contain a space, use the underscore character '_' to represent it.

WWW-Show-Organism-Summary-Link Set to "Y" to have the Query Page display the "Summary page for dataset" link. A value of "N" suppresses it. The default setting is "Y".

WWW-Show-Update-History-Link Set to ″Y″ to have the Query Page display the ″History of updates for this dataset″ link. A value of ″N″ suppresses it. The default setting is ″Y″.

WWW-Show-Diagram/Omics-Viewer-Links Set to ″Y″ to have the Query Page display the ″Cellular Overview Diagram / Omics Viewer″ link. A value of ″N″ suppresses it. The default setting is ″Y″.

2.1.3 Parameters Used when Accessing PGDBs within a MySQL or Oracle Server

Setting these parameters is MANDATORY if you want to access PGDBs from a MySQL or Oracle server.

RDBMS-Server-Hostname Hostname of the machine on which the MySQL or Oracle server is running.

RDBMS-Server-Port Port number on which either the MySQL or Oracle server is listening. The MySQL standard is 3306; The Oracle standard is 1521.

RDBMS-Database-Name Name of database within the MySQL or Oracle server in which PGDBs should be stored.

RDBMS-Username Username used to log in to the MySQL or Oracle server.

RDBMS-Password Password used to log in to the MySQL or Oracle server.

2.1.4 Parameters Associated with User Accounts Defined for a Pathway Tools Web Server

Optionally, a Pathway Tools based Web server can contain user accounts whose data are stored within a MySQL server. The following parameters configure that optional accounts system.

User-Account-RDBMS-Server-Hostname Hostname of the machine on which the MySQL User Account server is running.

User-Account-RDBMS-Server-Port Port number on which the MySQL User Account server is listening. The MySQL standard is 3306.

User-Account-RDBMS-Database-Name Name of database within the MySQL User Account server in which user information should be stored.

User-Account-RDBMS-Username Username used to log in to the MySQL User Account server.

User-Account-RDBMS-Password Password used to log into the MySQL User Account server.

2.2 X-Windows Basics

X-Windows is the name of the windowing system on UNIX computers. If you are physically at the computer where Pathway Tools will run, you normally do not need to know anything about X-Windows: it will simply work like it is supposed to. But, to be able to use X-Windows when remotely logging in to a UNIX computer (named, say, `myptoolsserver`), your local computer needs to be able to forward X-Windows.

To remotely log in from another UNIX computer (or from the optional X11 application on a Mac OS X computer) to `myptoolsserver`, use the following UNIX command, which both logs you in and accesses X-Windows:

```
ssh -X myptoolsserver
```

To remotely log in from a Microsoft Windows computer to `myptoolsserver` you will need to install and configure a third-party X Server such as xFree86 or Hummingbird Exceed. Be sure to configure it to forward X-Windows.

2.3 Running Pathway Tools from the Command Line

To start Pathway Tools for X-Windows operation on a UNIX-like computer operating system, such as Linux or Mac OS X, type

`pathway-tools` *`args`*

The text *`args`* consists of zero or more command-line arguments as defined under "Command Line Arguments" in Section 2.3.1.

There are a few ways to allow convenient access to the Pathway Tools program from within a command-line environment:

- The `pathway-tools` command must be copied to the UNIX `/usr/local/bin` directory.
- The directory `aic-export/pathway-tools/ptools/<version>/`, which contains the Pathway Tools executable file, must be on the user's `PATH` environment variable.

- A shell script invoking `pathway-tools` correctly can be written and placed within a directory contained in the `PATH` environment variable. An example of such a script can be found at the following web page, at step 16: http://bioinformatics.ai.sri.com/ptools/installation-guide/released/unix.html.

For both X-Windows and Web server operation, a proper X-Windows environment must have been established, as described in the previous section.

To exit from the Pathway Tools, click on the **Exit** command in the **File** menu of the Navigator window.

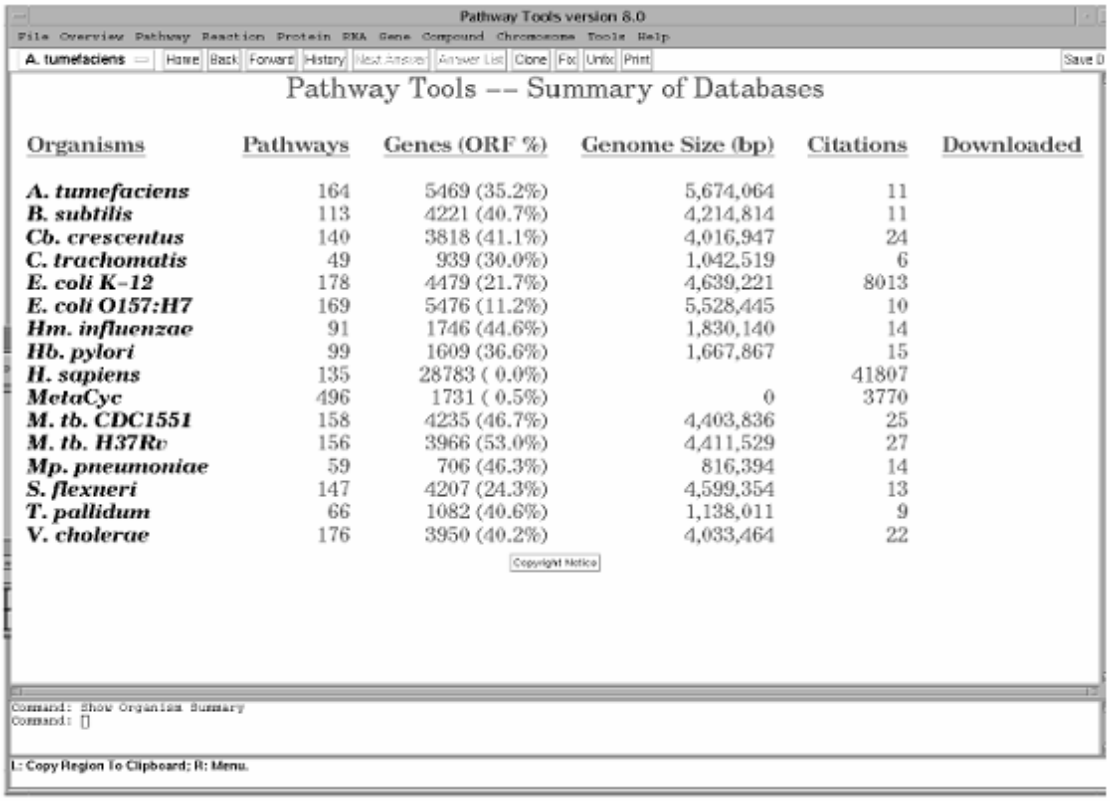


Figure 2.1: Navigator displaying Summary of Databases

When you invoke Pathway Tools for X-Windows operation, it will create one large window for the Pathway/Genome Navigator that is divided into several regions, called panes (see Figure 2.1).

Different information is presented in the different panes. The main pane initially contains a listing of available databases, in this case 16 databases. The horizontal pane at the bottom will print the names of commands that you select by clicking the mouse in the command menu. At times, informational messages will be printed in this pane. In the

course of use, pop-up windows will also appear periodically with messages or to request information from you. At times, the main pane will contain command buttons.

2.3.1 Command Line Arguments

Starting with Pathway Tools version 10.0, the most important configuration and initialization parameters can now be set in an Init File, which is discussed in Section 2.1. However, most of those parameters can be overridden by their corresponding command line argument.

Valid command line arguments for UNIX Pathway Tools are

`-api` This argument sets up the Pathway Tools to accept external queries from a package such as **PerlCyc** or **JavaCyc**. The PerlCyc module allows users to write programs in Perl that query a PGDB located on a Pathway Tools server running on the same machine as the Perl program. Communication is by UNIX file sockets. Consequently, this functionality is available only under UNIX. The PerlCyc module is not included in the Pathway Tools distribution. For more information about these API packages, see `http://bioinformatics.ai.sri.com/ptools/ptools-resources.html`.

`-dbdef` ***filepath*** Instructs Pathway Tools to load definitions from *filepath* of external databases to which database links are defined in one or more PGDBs. For more information on bulk loading of links to external databases, please see Section 7.5.9.

`-eval` ***expression*** The *expression* argument, which must be a valid Lisp expression, is executed by the Lisp interpreter. Once execution completes, Pathway Tools will start the Navigator in desktop mode or in web mode, depending on the other command-line options given. Example:

```
% pathway-tools -eval '(print-frame (quote trp))'
```

`-id` Instructs the Pathway Tools to print identifying version information for itself, and then exit.

`-linkdef` ***filepath*** Instructs Pathway Tools to load from *filepath* database links to be defined in one or more PGDBs. Note that the links are not saved unless you explicitly perform a save operation later in the session. For more information on bulk loading of links to external databases, please see Section 7.5.9.

`-lisp` Instructs Pathway Tools to start the Lisp interpreter instead of the Pathway/Genome Navigator interface, so that the user may enter Lisp expressions, such

as queries to Pathway Tools data. To exit from the Lisp interpreter back to UNIX, type `(exit)`. This option is not valid during Web operation.

`-load` ***filepath*** Instructs Pathway Tools to cause Lisp to open and execute the Lisp expressions stored in *filepath*. Once execution completes, Pathway Tools will start the Navigator in desktop mode or in web mode, depending on the other command-line options given. An example loadable file:

```
;; Most files should begin by specifying that they will be interpreted
;; within the Lisp package called ecocyc.
(in-package :ecocyc)
;; Select MetaCyc as the current PGDB
(select-organism :org-id 'meta)
;; Add a new synonym to the frame for L-tryptophan and then print the
;; frame
(add-slot-value 'trp 'synonyms "L-tryptohan xyz")
(print-frame 'trp)
```

`-org` ***orgid*** Instructs Pathway Tools to select the organism whose organism ID is `orgid` as the current organism. In Web server mode, this organism will be the default organism selected on the Pathway Tools Query Page.

`-patch` Instructs the Pathway Tools to download and install all the latest patches from the public Pathway Tools patch Web site, and then exit.

`-patho` ***directory*** **or `-patho -f`** ***filepath*** It is possible to run PathoLogic in Batch Mode, where all predicted results are automatically accepted. This command-line option is followed by the directory where the organism's data files can be found, or by `-f directory-file`, where `directory-file` is a file that lists all of the directories that PathoLogic should search to find input files (one directory per PGDB). Please see Section 6.6 for more information on PathoLogic Batch Mode.

`-rdbmstest` When Pathway Tools is configured for operation with a MySQL or an Oracle database, this option attempts to connect to the RDBMS server, then lists those PGDBs that are present, and then exits. This option is used for verifying that a proper connection to the RDBMS can be established at a very basic level.

The following UNIX command line arguments are applicable to Pathway Tools only when operating in Web mode:

`-acl-file` ***filepath*** User access rights to PGDBs can be controlled in a fine-grained fashion using Access Control Lists (ACLs). Once passwords are enabled, all published PGDBs are by default restricted to access by authorized users only. If an ACL

file is used, each PGDB can have its own access policy. A PGDB can be open (available to all, with no authorization control), or authorized (available only to all users specified in the `-passwd-file` option), or limited to specific authorized users.

This feature is controlled using an ACL file located in the place given by the *filepath* argument. The ACL file has the following format:

```
organism-id    :auth
organism-id    :open
organism-id    user1 user2 user3 ...
```

Examples:

```
human     :auth
```

This line would indicate that all authorized users can access the "human" PGDB.

```
agro     :open
```

This line indicates that anyone, whether authorized or not, can access the "agro" PGDB.

```
ecoli     joe fred tom
```

This line indicates that only users joe, fred and tom can access the "ecoli" PGDB.

Note that these restrictions prevent any access to the PGDB at all, unless the user is in the authorized group of users. That is, multi-organism queries are also checked to make sure the user is only allowed to access that set of organisms for which he is authorized.

To facilitate the entry of long lists of users, the ~ character can serve as a line-continuation character.

Example:

```
ecoli       ~
      joe   ~
      fred  ~
      tom
```

would have the same effect as the line

```
ecoli     joe fred tom
```

`-allow-webcrawlers` The Web server mode activates Web crawler detection by default. When too many requests come in from the net, which originate from the same IP number within a certain time period, such an IP number will be blocked from further access, until the Pathway Tools server is rebooted. The assumption is that aggressive Webcrawlers (from spammers) are taxing the performance from

the Web server substantially, making access for *bona fide* users more sluggish. However, under certain circumstances, it may be desirable to bypass the Web crawler detection, which can be achieved by supplying this argument, when calling the `pathway-tools` script.

-background-color ***color*** This will modify the way that images are generated by the Pathway Tools server in web mode, such that the colors used in the images will show up well against the specified background. It can take as arguments black, white, gray, or blue. This option does not actually modify the background color of the web pages that Pathway Tools serves in web mode. You would modify your CSS settings to achieve that change; see Section 8.2 for more information).

-blast Restore access to the BLAST program from the Web query page if it was removed by virtue of the `-no-blast` or `-gene-link-db` arguments being supplied.

-email ***email-address*** Specifies the email address to which technical and content-related support questions should be addressed. This address will appear on Pathway Tools generated Web pages.

-gene-link-db ***db*** When a user site sets up a Pathway Tools database in conjunction with a previously existing database of genes for an organism, it is sometimes useful to have references to genes in Pathway Tools Web pages link directly to gene pages in the user's pre-existing database, rather than to the gene pages generated by the Pathway/Genome Navigator. To accomplish this, the user must create a Database frame in the PGDB that contains information necessary for linking to the desired external database, and each gene frame in the PGDB must contain a link to the corresponding object in that database. For more information, please see Section 7.5.9. Then, if the Database frame ID is supplied as the value of this command line argument, pages generated by the Pathway Tools Web server will substitute links to the external database anywhere it would normally link to a gene page.

-google-text-search By default, if your Web server is not served from port `80`, the Pathway Tools query page will not contain a full-text-search query box, powered by Google™. If your Web server is indexed by Google nevertheless, and you want the search box to appear, supply this argument.

-no-blast Ordinarily the Web server's Query Page contains an option to invoke the BLAST program. If access to BLAST from this page is not desired (e.g., if the BLAST program is not installed, or if such functionality is available elsewhere on a Web site), then supplying the `-no-blast` option causes it to be removed from the query page. Note that BLAST is disabled by default when the `-gene-link-db` argument is supplied.

-no-google-text-search By default, if your Web server is served from port `80`, the Pathway Tools query page will contain a full-text-search query box, powered by

Google™. This capability is useful only if the site can be indexed by Google. If your Web server is running on an internal network or cannot be indexed by Google for other reasons, supply this argument to remove the search box.

-passwd-file ***filepath*** This option allows control over which users can access a given PGDB. The *filepath* argument gives the location of the password file in the file system. Once password authentication is turned on, all PGDBs will only be available to authorized users by default. This can be tuned using access control lists, described below for the `-acl-file` option. Note that the user names defined via this mechanism are completely orthogonal to any user accounts defined via the Web Accounts system defined in Chapter **??**.

The format of the password file is simply a series of lines in plain text of the form

```
user1:password1
user2:password2
```

etc.

-port ***NNN*** This option is valid only when Pathway Tools is operating in Web server mode and is typically used in conjunction with the `-user` option. It specifies in *NNN* the TCP/IP port on which the Pathway Tools Web server will listen for requests. By default, port `1555` is used on UNIX (Linux), and port `80` is used on Microsoft Windows. See the `-user` argument for UNIX security information.

-proxy-port ***NNN*** This argument is valid only when Pathway Tools is operating in Web server mode. It specifies in *NNN* the TCP/IP port on which another Web server, such as Apache, will listen for requests and forward them to the port on which the Pathway Tools Web server listens for requests. When using this argument, you also have to configure the other Web server, such as Apache, to actually forward the requests. Typical use of this command line argument is to specify `-proxy-port 80` to work around firewall restrictions; most firewalls do allow traffic to port 80.

-user ***username*** This option is valid only when Pathway Tools is operating in Web server mode and is typically used in conjunction with the `-port` option. It specifies in *username* the UNIX account that Pathway Tools should use to process Web server requests. UNIX allows only `root` accounts to listen on TCP/IP ports numbered up to 1024, so if you specify `-port 80`, for example, then only `root` can start Pathway Tools, but Pathway Tools will switch to running as user *username* right after starting to listen on port `80`. This option is neither available nor necessary on Microsoft Windows.

-www Instructs Pathway Tools to operate in Web server mode.

-www-publish ***pubspec*** This argument is applicable when Pathway Tools is running in Web server mode, and affects which of the available PGDBs are visible through the Web server.

Example:

Consider a user who wishes to publish all available PGDBs. In this case, use command:

```
pathway-tools -www -www-publish all
```

The full set of possible `pubspec` arguments is

all — make all PGDBs visible

orgA+orgB+...+orgX — make the specified set of organisms visible. Each **orgX** is an organism ID.

public — make public PGDBs visible (excludes the formerly restricted PGDB MetaCyc)

Chapter 3

Pathway/Genome Navigator

The preceding chapter describes how to start Pathway Tools, including the Navigator. This chapter describes the basics of how to work with the Navigator and then leads you through several examples, which you should attempt to execute as you read. We will focus on navigating the information contained within EcoCyc, *E. coli* database.

3.1 Using the Mouse To Navigate and Issue Commands

User interaction with the Navigator is primarily mouse oriented. In general, the left mouse button is used to invoke specific commands and for hypertext navigation. Whereas the right button is used to bring up menus of additional operations, where available. The middle button has some specialized uses that will be described later. Items that can be clicked on include elements of the command menu, buttons within display panes, pop-up menus, and textual and graphical elements in display windows. You can tell when an item is mouse sensitive: if you move the mouse pointer over a sensitive item, a bounding rectangle appears around the item. In some cases, a mouse-documentation line also flashes at the bottom of the window to explain what will happen if you click on the item. For example, move the mouse cursor over the **Reaction Mode** command of the command menu, and note that a rectangle comes up and surrounds these words. Note also that the following optional mouse operations appear in the bottom pane, **L: Retrieve reactions; R: Menu**. This indicates that use of the left mouse button allows you to retrieve one or more reactions from the current database (see below), while use of the right button will bring up a menu that indicates the available options for the current cursor position. At times a tooltip may appear giving additional information about the screen item the mouse cursor is over.

The Navigator may appear to be unresponsive for several reasons (see "Frequently Asked Questions" in Chapter 5). The most common reason occurs periodically and depends

on how much memory is available for operation of Pathway Tools. At roughly half-hour intervals, the Navigator pauses to perform a memory management function called 'garbage collection' that takes approximately 30 seconds to complete.

3.2 Menus and Dialogs

The Pathway/Genome Navigator has a Menu Bar, Single-Choice Menus, Multiple-Choice Menus, and Dialogs, described below. You may encounter Single-Choice Menus, Multiple-Choice Menus, and Dialogs while using the Menu Bar or by right-clicking on the title of a biological object display.

3.2.1 Menu Bar

The Menu Bar at the top of the Navigator window contains a pull-down menu for each major biological object type — Metabolic **Overview**, **Pathway**, **Reaction**, **Protein**, **RNA**, **Gene**, **Compound**, **Chromosome** — as well as **File**, **Tools**, and **Help** menus containing general commands. For example, clicking on the **Reaction** menu will reveal a list of query options that apply specifically to reactions.

3.2.2 Single-Choice Menus

This type of pop-up menu lets you select a single item from a list. As soon as you left-click on an item, the selection is made and the menu vanishes.

3.2.3 Multiple-Choice Menus

This type of pop-up menu allows you to select *one or more* items from a list. Click on as many items as you want; when you have finished selecting, click on the **OK** button at the bottom. If you change your mind and want to de-select an item before you have clicked **OK**, simply click on the item a second time.

3.2.4 Dialogs

A Dialog allows you to answer several questions in one window. Alternative answers are presented, with default answers in boldface. Click on a different answer to override the default. When all questions are answered, click **OK** at the bottom of the menu.

3.2.5 Aborting Out of Menus and Dialogs

If the Pathway/Genome Navigator is requesting input from you via a Menu or a Dialog and you want to abort out of it, try one of the following actions:

Click Cancel or No Select.

- Type ^Z (i.e., hold down the control key and press the `Z` key while the control key is still held down).
- Click the top of the Menu or Dialog.
- Click outside the Menu or Dialog.

3.3 Organism Summary Display

When you invoke the Navigator, the main window contains the Organism Summary Display (see Figure 3.1).

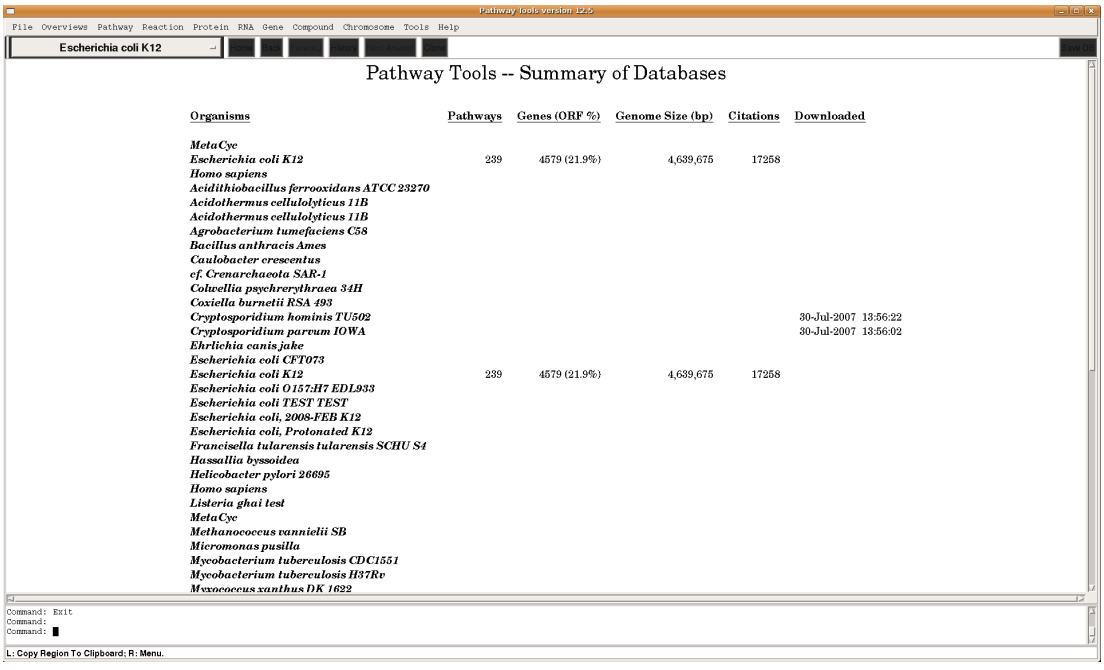


Figure 3.1: Organism Summary Display

At the top of the Organism Summary Display is the Pathway Tools banner, followed by a list of available databases. The first column lists the names of organisms for which

databases are available. To the right are additional columns containing statistics on the database content. From left to right, each row gives the number of pathways (computationally predicted and user created), the number of genes (and the percentage of these that represent open reading frames, or ORFs), the genome size in base pairs, the number of literature citations, and a download date, if available. At the top of the command menu, the Current Organism is listed. By default, the Current Organism is Escherichia coli K12. At the bottom of the page is the **Copyright Notice** command.

To return to the Organism Summary Display from any Navigator window, use the command **File** → **Summarize Databases** or the **Home** button.

3.4 Single Organism Display

The Single Organism Display summarizes the content of the database for an individual organism (see Figure 3.2).

Figure 3.2: Single Organism Display

The Single Organism Display for an organism is accessible from the Organism Summary Display by left-clicking the name of an organism in the **Organisms** column. The Single Organism Display for the selected organism then replaces the Organism Summary Display in the main display window. In addition, this organism becomes the Current Organism.

The top of the Single Organism Display gives the organism name and the specific subspecies and strain whose genome and predicted metabolic pathways are available as a database. Directly below is the **Summarize Pathway Evidence** command (available from the Single Organism Display of PGDBs, but not for EcoCyc). Clicking on this option takes you to a series of HTML pages that outline the genomic evidence for the metabolic pathways of this organism. The **Replicon** column lists sequenced replicons (chromosomes and plasmids) of the organism; for example, in the case of *S. cerevisiae* (yeast), all 16 chromosomes and the mitochondrial genome are listed. For each such element, summary information on its genes is provided in additional columns, that is, the number of mapped genes, and a breakdown of this number between protein coding and RNA coding genes. The size (in base pairs) is also listed. Clicking on an element name takes you to a window display of the genomic map of that element (see "Genome Browser" in Section 3.8.10). For example, clicking on Chromosome XVI of the *S. cerevisiae* summary display takes you to a display of the genomic map for chromosome XVI.

In multiple organism PGDBs, like MetaCyc, pathways are annotated with the identifiers of organisms in which those pathways have been experimentally elucidated. By clicking on the name of a taxonomic group, an organism display page is generated for that taxon that lists pathway information associated with the taxon. The top of the organism display page is the same as for Single Organism Display. This page presents the synonyms, the rank in NCBI Taxonomy and the taxonomic lineage. For each taxon the list of pathways that were experimentally observed for it or one of its children is also presented. For example, for taxon Bacteria the list of all bacterial pathways appears, and for *E. coli* K-12 the list of pathways experimentally observed in this organism are presented. Additionally, the list of pathways for which this taxon is in the taxonomic range (that is, pathways that were deemed by the PGDB curators that are possible to appear in organisms placed in certain branches of the NCBI Taxonomy) is presented at the bottom of the page.

The taxonomic lineage also appears in the organism display page for single organism PGDB. There is a display page for each of the taxa in the lineage but the pathway information is not presented since it is considered redundant.

3.5 Notion of Current Organism

Pathway Tools always keeps track of what it considers to be the current organism for processing. The identity of the current organism is important in two respects. First, all

command-mode queries are directed against the database for this organism and, second, for comparative analyses this database serves as the reference for comparison with one or more other user-specified databases. However, when you follow hypertext links within the Navigator, through clicking on the name of an enzyme within a pathway drawing, the next object displayed, the enzyme, is always from the same organism as the organism containing the previous object. This organism is not necessarily the current organism.

When you first enter the Navigator, by default the current organism is *E. coli*. This means that all queries will be directed against the database for this organism. To change the current organism, select a database for a different organism from the Organism Summary Display (Section 3.3) or go to the top of the command menu where the name of the current organism is listed within the Organism Selector. Left-click once on this name to bring up a listing of all available databases and left-click once on the name of an organism to make it the new current organism. The last organism selected in this way or via the All Organisms display is the current organism. Its name is displayed in the Organism Selector at the top of the command menu. An asterisk ("*") next to the name of an organism in the Organism Selector means that the DB for that organism has unsaved changes.

To explore the pathway/genome information space of a given organism, it must be the current organism. To follow the examples in the rest of this chapter, set *E. coli* as the current organism. All the Navigator commands outlined below for navigating, querying, visualizing, and analyzing EcoCyc may also be used to explore any one of the other available databases, except MetaCyc (which does not contain information on objects that are not relevant, for example, replicons).

If your installation includes one or more organism PGDBs that were created using an older version of Pathway Tools (e.g., because you previously created them using an older version of PathoLogic or because you downloaded them from the PGDB Registry), their schema may be out of date with respect to the currently installed software. The first time you select such an organism, the software will automatically detect that its schema is out of date and offer to upgrade and save the PGDB for you. If you do not upgrade, you will not be able to access the PGDB.

3.6 Examples

In the following examples it is assumed that the main Navigator window is now visible on your monitor and that EcoCyc is current. These examples are designed to introduce you to some basic Navigator features that allow you to retrieve and view information from EcoCyc.

Example 1
Invoke the menu command **Reaction** → **Search by EC#**, which allows you to look up a reaction by its Enzyme Commission number. A pop-up window appears: type "5.3.1.9"

and click **OK**. (If nothing happens when you type, you may have to move the mouse pointer into the pop-up window, and/or click on its title bar).

A display window for the reaction catalyzing the first step of glycolysis will appear. A number of items in this display are mouse sensitive, including the name of the enzyme that catalyzes the reaction, the gene that encodes this enzyme, the pathways containing the reaction, and the compounds in the reaction equation. Click on any of these items to see displays of the respective objects, such as the glycolysis I pathway.

Example 2
Invoke the command **Pathway** → **Search by Class**. A pop-up menu gives a classification hierarchy for metabolic pathways.

Select a class, for example, **Amino Acids** → **Individual amino acids** (under **Biosynthesis**). A new pop-up menu lists a number of individual pathways for amino acid biosynthesis (the instances of this class). Click on one of the pathways.

A drawing of the selected pathway appears. Virtually every item in the drawing is mouse sensitive. For example, you can click on a reaction arrow to see a display of that reaction, and you can click on an enzyme name to see a display of that enzyme.

Example 3
Invoke the command **Protein** → **Search by Substring** . In the pop-up menu, type "pyruvate" and click **OK**. A second pop-up menu lists all proteins (including enzymes) contained in EcoCyc whose name (or one of its alternate names) contains the substring pyruvate. Click on any protein name, and then click on **Use these values** at the bottom of the menu, to display that protein.

Scroll downward in the pane for this protein (scroll by clicking on the small arrow at the bottom of the scroll bar). Click on one of the gene names. When the gene display appears, click on the map position of the gene (assuming your gene has a map position).

When the genome browser appears, you can zoom in on a region of interest by using the navigation bar (see "Genome Browser" in Section 3.8.10). You can obtain detailed information on a gene by left-clicking on a gene name.

3.7 Query Facilities

When searching for a particular piece of information about *E. coli* metabolism, we can usually try two different strategies to find that information: a direct and an indirect approach.

Using the direct approach, we issue queries for the entity we seek. For example, imagine that we seek information on the *hisA* gene, such as its map position and the name of the

enzyme it encodes. The Navigator allows you to call up an information window for a gene by its common name (or by other names by which that gene is known).

Using the indirect approach, we bring up the information window for an object by first issuing a direct query for a related object, and by then navigating to the object of real interest. For example, imagine that we had forgotten the name of the *hisA* gene, but we knew it encoded the enzyme that catalyzed the last step in the biosynthesis of histidine. We could use a direct query to display the biosynthetic pathway for histidine, and then click on the name of the enzyme catalyzing the last reaction in the pathway. The resulting information window for that enzyme names the gene (*hisA*) encoding the enzyme. Clicking on the gene name displays the information window for *hisA*.

In summary, by using the Pathway/Genome Navigator we can traverse many paths through EcoCyc to arrive at the same information.

3.7.1 Direct Queries

To query a given type of object, you must select from the menu associated with that class of object. The object type menus are

1. Overview
2. Pathway
3. Reaction
4. Protein
5. RNA
6. Gene
7. Compound
8. Chromosome

Each menu contains a set of type-specific predefined queries. For example, from the compound menu, you can query compounds by any of these criteria:

1. Exact compound name or ID
2. Substring within a compound name
3. Chemical substructure specified using the SMILES chemical notation [18] (all compounds containing that substructure will be returned as the result of the query)

4. Compound class chosen from a menu-based classification hierarchy of compounds (such as amino acids, carbohydrates, and nucleotides)

5. Advanced search that includes some combination of name, molecular weight, chemical formula, and substructure

Note that most commands that allow you to query objects by their exact name allow you to enter in several names within one pop-up window, separated by commas, for example, "hisA, hisB, hisC". The exception is the compound menu, because many compounds have commas within their names.

3.7.1.1 Queries with Multiple Answers

Some queries return more than one object as the result. For example, most modes allow you to query objects by a substring search, such as searching for all proteins with "pyruvate" in their names. In this instance, the Navigator creates a menu that lists the proteins satisfying the query. The menu allows you to select one, some, or all of the proteins — click on individual protein names to select them, or click on **Select All** to select all of them (clicking on an already-selected name will de-select it). When you have finished selecting, click on **Use these values**.

The Navigator immediately displays one or more of the proteins you selected (see "Pane Layout" in Section 3.10.11.1 to find out how to divide the main display window into multiple panes; see "History and Answer Lists" in Section 3.10.11.8 for information on how to control the number of objects displayed by **Next Answer**). It remembers the others on a list called the Answer List.

When you want to see the next protein on the Answer List, click on **Next Answer** in the command menu. For more information about the Answer List, see Section 3.10.11.8.

3.7.2 Indirect Queries: Navigation

The information window for each object usually lists a number of related objects. For example, a gene display shows the product of that gene; if the product is an enzyme, then the reaction(s) catalyzed by the enzyme is listed, as are the pathways in which that reaction occurs. Similarly, a compound display lists the reaction(s) that produce and consume that compound, and the pathway(s) in which the compound is found.

Each of these related objects is "live" in the sense that clicking on the object displays an information window for that object. Objects are color coded by type to make their relationships more evident, and to make it more obvious which visual elements within a

complex display are mouse sensitive. A bold-face font is used instead for mouse-sensitive objects, when monochrome monitors are used.

On occasion, nothing happens when you click on a related object, most likely because EcoCyc has no information about that object, although that incompleteness should be remedied in a future version.

3.7.3 History List

The Navigator keeps a *History List* of the last few objects that you have displayed. You can return to a previously displayed object by clicking on the **Back** button in the command menu. For example, if there are three items on the history list, clicking on **Back** three times returns you to the display you started with. Clicking on **Forward** moves you through the history list in the other direction. To select one or more arbitrary objects from the history list, click on the **History** button in the command menu. You are presented with a multiple choice menu of every item on the history list. To see the current history list, select **Tools** → **History** → **Show on Console**.

By default, the history list contains as many as fifty items. This length can be changed (see "History and Answer Lists" in Section 3.10.11.8).

When you exit the Navigator, your history list is saved in a file called `.ecocyc-history` in your home directory. This feature enables you to easily start up again at the same place where you left off.

3.7.4 Programmatic Queries

The examples below of different complex queries against Pathway/Genome Databases are written in the Common Lisp programming language. To write programmatic queries, you must understand a number of aspects of Pathway/Genome Database schemas, such as class and slot names. The current schema is described in the Appendix.

The Preface lists a variety of additional reference sources relevant to writing Lisp queries, including a longer set of example queries that are available through the SRI Web site.

One convenient way to examine the answer to a query is to put the result of the query on the Answer List of the Pathway/Genome Navigator, and to look at each answer using the **Next Answer** command. The first query below shows how to do so.

```
;; Find genes located between 20 and 30 centisomes on the map
(loop for gene in (get-class-all-instances '|Genes|)
      for pos = (get-slot-value gene 'centisome-position)
      when (and pos
                (> pos 20)
```

```
                (< pos 30))
      collect gene)


;; The preceding query returns a list of genes. To run the Pathway/
;; Genome Navigator with those genes on the Answer list,
;; evaluate the following. The "*" means "the result returned by
;; the last expression evaluated".
(eco :answer-list *)


;; Find reactions involving pyruvate as a substrate
(loop for rxn in (get-class-all-instances '|Reactions|)
      when (member 'pyruvate (get-slot-values rxn 'substrates))
      collect rxn)


;; Find all genes whose products catalyze a reaction involving
;; pyruvate as a substrate
(loop for rxn in (get-class-all-instances '|Reactions|)
      for genes = (genes-of-reaction rxn)
      when (member-slot-value-p rxn 'substrates 'pyruvate)
      append genes)


;; Find all enzymes that use pyridoxal phosphate as a cofactor
;; or prosthetic group
(loop for protein in (get-class-all-instances '|Proteins|)
      for enzrxn = (get-slot-value protein 'enzymatic-reaction)
      when (and enzrxn
                (or (member-slot-value-p enzrxn
                          'cofactors 'pyridoxal_phosphate)
                    (member-slot-value-p enzrxn
                          'prosthetic-groups 'pyridoxal_phosphate)
                    ))
      collect protein)
```

3.8 Object Displays and Queries

Specific queries can be issued in each command mode and result in different object displays.

3.8.1 Shared Display Characteristics

Certain aspects of the object displays are shared by most or all of the different object classes.

3.8.1.1 Gene-Reaction Schematic

The many-to-many relationships among genes, enzymes, and reactions can be complex. An enzyme comprising several subunits might catalyze more than one reaction, and a given reaction might be catalyzed by multiple enzymes. The *Gene-Reaction Schematic* depicts the relationships among a set of genes, enzymes, and reactions (see Figure 3.3 and `http://ecocyc.org/new-image?type=REACTION-IN-PATHWAY\&object=DAHPSYN-RXN`).

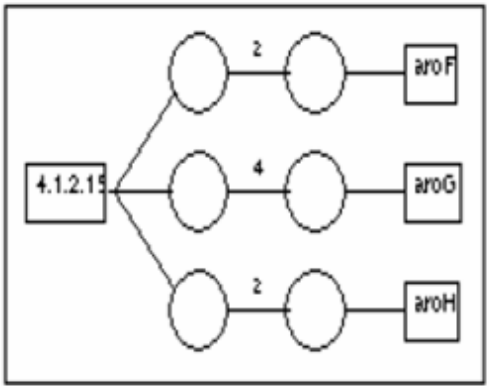


Figure 3.3: Gene-reaction schematic

It is drawn in reaction displays, protein displays, and gene displays. It is generated by starting with the object that is the focus of the current display (which is highlighted in the schematic), and then recursively traversing database relationships from that object to related objects, such as from a gene to its product, or from a reaction to the enzyme(s) that catalyzes it. The schematic summarizes these complex relationships succinctly, and also constitutes a navigational aid: click on an object in the schematic to cause the Navigator to display that object.

In gene-reaction schematics, the boxes to the left represent reactions, the boxes on the right represent genes, and the circles in the middle represent proteins. The lines indicate relationships among these objects. For example, the schematic in means that the aroF gene encodes a polypeptide (the circle to the left of the box for the aroF gene) that forms a homodimer (the next circle to the left — the 2 indicates two copies) that in turn catalyzes

reaction 4.1.2.15. The situation for the aroG gene is similar except that its product forms a homotetramer. Gene-reaction schematics that depict heteromultimers show more than one gene-product circle connected to a single circle for a protein complex. Gene-reaction schematics also include modified forms of a protein (or tRNA) when relevant. For example, the schematic for the acyl carrier protein shows a yellow circle for the unmodified form of the protein, and it shows 19 orange circles, which represent different modified forms of the protein.

Gene-reaction schematics should *not* be confused with pathway diagrams; although both diagrams are graphs, they mean different things.

3.8.1.2 Citations and Comments

Citations and comments are found in all types of objects within organism databases. Comments authored by database curators are found in several locations in object displays. In some cases, citations are presented in the same manner as are comments. Consider a line from an enzyme display such as

```
Inhibitors (allosteric) [3]: NADH [4], succinate [2,Comment1]
```

This line identifies NADH and succinate as allosteric inhibitors of the enzyme being shown. The **[4]** indicates that EcoCyc contains a citation that pertains to the fact that NADH is an allosteric inhibitor of the enzyme. When the mouse is over the "4", the pointer documentation window at the bottom of the screen shows the start of the citation information. Clicking on the "4" navigates to where the full citation information is displayed in the References section at the bottom of the page. Clicking on the full citation information displays the citation in a pop-up window or, if the citation is available through PubMed, in a Web browser window. Analogously, a comment and citation pertain to the role of succinate as activator or inhibitor: **[2,Comment1]**. Passing the mouse over the "Comment1" shows the start of the comment in the pointer documentation window, and clicking on it brings up the full comment in a pop-up window. The **[3]** is a more general citation about the inhibition of the enzyme that does not pertain precisely to NADH or succinate alone.

Citations of general relevance to an object (as opposed to citations that pertain to a particular property or data value) are shown on a separate line in the object display window, such as

```
Citations: [1,2,3,4]
```

Note that citation indicators can be either numeric or mnemonic. To choose the style of citation indicator you prefer, use the command **Tools** → **Preferences** → **Citation Reference Style**.

3.8.1.3 Database Links

Objects within EcoCyc contain links to a variety of other databases. For example, some *E. coli* polypeptides are linked to both SwissProt and PDB entries; some *E. coli* genes are linked to entries in the Coli Genetic Stock Center database. These databases are not a part of or provided with Pathway Tools and may require a license from their owners or distributors to access their contents.

In general, databases contain two types of links: *unification* and *relationship* [4]. Unification links are links to descriptions of the *same object* in another database, and are displayed in a line that lists one or more links, where each link displays the name of the other database, and the unique identifier of the target object in that database, such as

```
Unification links: SwissProt:P34554
```

Relationship links refer to a *related object* in the remote database. Because a number of possible relationships might exist between the source object in EcoCyc and the target object in the other database, the line describing relationship links displays the name of the foreign database, the ID of the target object, and the name of the relationship to the target object. For example, some *E. coli* polypeptides are linked to PDB homologs if PDB does not describe the structure of this exact *E. coli* protein:

```
Relationship links: PDB:Homolog:P34554
```

When you click on the target identifier, your web browser will display the target object in a web page.

3.8.1.4 Classes

EcoCyc contains a number of taxonomic hierarchies. Reactions are classified according to the Enzyme Nomenclature system [17]. Genes are classified according to a system devised by Riley [13]. Compounds and pathways are also classified. All the classification systems are multilevel and involve a number of instances and classes. Examples of instances would be the reactions in the Current Organism, and examples of classes would be the reaction types defined by the Enzyme Nomenclature system. Many of these classes in turn are assigned to superclasses (which may recursively be assigned to additional superclasses).

Each object display shows the parent class(es) of that object. Clicking on a parent class displays the parent class. The display window for a class lists both its parent classes and its instances, if any. You can also click on these parent classes, or instances, to display them. In this manner, the user can navigate the *E. coli* taxonomic hierarchies.

3.8.2 Systems Level Overviews

Pathway Tools offers three different global displays that provide genome-scale depictions of information within a PGDB. The Cellular Overview is an overview of the biochemical processes in the cell, focusing on metabolic pathways and reactions, and transport. The Regulatory Overview (which is currently available only for EcoCyc) is a whole-organism view of the transcriptional regulatory network. The Genome Overview depicts all genes on all chromosomes.

In all three overviews, individual objects or sets of objects can be queried and highlighted. All three overviews allow omics data to be superimposed on them, and work in the desktop mode of Pathway Tools. Only the Cellular Overview currently works through the Web mode of Pathway Tools, but support for Web operation will be added to the other tools in a future release.

3.8.2.1 Commands Applicable to All Three Overviews

- **Omics Viewer: Overlay Experimental Data from:** Experimental data, such as gene expression, proteomics, reaction flux or metabolomics data, is read from a file, and reactions and/or compounds are colored according to the experimental values (absolute or relative) associated with the corresponding genes, proteins, reactions, or compounds. See Section 3.8.2.5 on viewing experimental data using the Omics Viewer. The two options are
  - Text File
  - SAM Output File (see Sub-section 3.8.2.7)
- **Clear All Highlighting:** Clear any highlighting previously applied to objects in the Overview.

3.8.2.2 The Cellular Overview

The Cellular Overview diagram is a representation of all metabolic pathways and reactions, signaling pathways, membrane proteins, and transporters defined for the current organism. In this diagram, each icon (e.g., circle, square, ellipse) represents a single metabolite. The shape of the icon encodes the chemical class of the metabolite, as listed in Table 3.8.2.2.

The shading of the icon indicates the phosphorylation state of the compound: shaded compounds are phosphorylated; unshaded compounds are unphosphorylated.

Each thick line in the Overview diagram represents a single reaction. Neither the icons nor the lines are unique in the sense that a given metabolite or a given reaction may occur

Icon Shape	Compound Class
Square	carbohydrate
Triangle	amino acid
upside down triangle	cofactor
ellipse (horizontal)	purine
ellipse (vertical)	pyrimidine
Diamond	protein
T-shape	tRNA
Circle	all other compound types

Table 3.1: Compound shapes used in the Overview Diagram

in more than one position in the diagram. If there are any thin gray reaction lines, these represent reactions for which no enzymes have been identified in the PGDB, in other words, pathway holes.

The "barbells" along the right side of the diagram represent individual reactions that have not been assigned to a particular pathway. They are presented as single reactions because their direction and role are determined by the metabolic condition of the cell. The barbell region also contains some reactions of macromolecule metabolism, such as DNA metabolism. In the region to the left of the barbells, the glycolysis and the TCA cycle pathways in the middle separate predominately catabolic pathways on the right from pathways of anabolism and intermediary metabolism on the left. The existence of anaplerotic pathways prevents rigid classification. The majority of the metabolic pathways operate in the downward direction. Pathways are generally grouped by class, and the extent of a class is indicated by background shading.

The border drawn around the Overview depicts the cytoplasmic membrane, and contains embedded transport proteins. Transported substrates use the same shape codings as for metabolic substrates. Where possible, transporters are positioned in the membrane so as to be near some of the metabolic reactions into which their substrates feed.

In the the Overview for Gram-negative bacteria, such as in EcoCyc, both the inner and outer membranes are shown. Periplasmic reactions and proteins are depicted in the space between the two membranes at the right of the diagram.

You can interrogate the Overview in several ways. To identify a compound within the Overview, move the mouse pointer over a geometric figure in the diagram — the Navigator will print the name of the metabolite and the name of the containing pathway in a tooltip. To identify a reaction, move the mouse pointer over a thick line — the Navigator will print the equation of that reaction, and the name of the containing pathway. If the mouse pointer is moved over a shaded but blank region, the Navigator will print the name of the pathway class.

If you left-click on a compound or a reaction in the Overview, that object is displayed in its own display window.

Conversely, if you are looking at another display that contains a link to a compound, reaction or pathway, you can right-click over the link to bring up a short menu of operations. One of those operations is "Show compound/reaction/pathway in overview". Selecting that menu item causes the Overview diagram to be drawn, with the designated entity highlighted.

3.8.2.2.1 Cellular Overview Commands

Show Cellular Overview Draw the Overview diagram for the current organism.

Show Key Displays a pop-up window containing a key for the Overview diagram that explains what compound classes are denoted by each node shape, and explains what the different highlighting colors represent.

Show/Hide Transport Links Toggle whether or not faint lines are shown that connect transported substrates to the pathway(s) in which they participate. By default, these links are hidden.

Highlight You can request that some entity be highlighted in the Overview. You can request that a compound, a reaction, a pathway, an enzyme (i.e., the reaction(s) catalyzed by the enzyme) or a gene (i.e., the reaction(s) catalyzed by the product of a given gene) be highlighted. The object to be highlighted can be specified in a number of ways (e.g., by name, by substring, by EC number) using a set of cascading menus that reproduce many of the query capabilities present in the other command modes. Specifically, the highlighting commands are

Species Comparison (see "Comparative Operations" in Section 3.11)

Pathway
- By Name or Frame ID
- By Substring
- By Class
- All by Class (colors all pathways according to their role, e.g., all amino-acid biosynthetic pathways are in one color)
- By Genome Clustering (colors pathways according to the clustering within the genome of the genes that encode their enzymes — the accompanying pop-up window describes the color scheme in more detail)

Reaction(s)
- By Enzyme Name
- By Enzyme Substring
- By EC Number (e.g., reaction with EC number 1.2.3.4)

- Using EC Class Hierarchy (e.g., all reactions in class 1.2)
- All by Top-Level EC Class (colors the entire diagram to reflect the chemical type of each reaction)
- From File of Reaction Designators (takes as input a file containing EC numbers, one per line)
- All without EC Numbers (of which there are many in EcoCyc, because the Enzyme Commission has yet to assign EC numbers to many reactions)
- By Pathway (e.g., all reactions that occur in the TCA cycle)
- By Substrates (this option allows you to specify a full or partial list of the reactants and products of the reaction)
- By Effects of Compound(s) on Enzyme Activity (this option allows you to highlight reactions according to modulation of the enzyme(s) that catalyze the reaction, e.g., to highlight all reactions whose enzyme is activated by ADP)
- By Enzyme Cellular Location (highlights reactions whose enzymes are known to be located in a user-selected cellular location)
- All with Multiple Isozymes
- All in Multiple Pathways (highlights reactions that occur in more than one pathway)

Gene
- By Name or Frame ID
- By Substring
- By MultiFun Class
- By Regulon (allows you to select a transcription factor, and highlights all reactions whose genes are in operons that are regulated by that transcription factor)
- Gene List From File (the file should contain a list of gene names, one per line)
- All by Replicon (colors reactions according to the replicon — chromosome or plasmid — on which their genes are located)

Compound(s)
- By Name or Frame ID
- By Substring
- By SMILES Structure
- Using Class Hierarchy (e.g., all amino acids)

Answer List Highlights items from the current Answer List that appear in the Overview.

Undo Reverts highlights to the last item or set of items that were highlighted in the Overview.

Redo Highlights the last item or set of items that had just been removed via the Undo command.

Save to File Saves a given pattern of overview highlighting on to a file.

Load from File Inputs a file created using the Save to File command. This restores the specific pattern of overview highlighting previously saved to this file.

Metabolite Tracing Metabolic transformations can be traced forward and backward from a given metabolite of interest, with the resulting paths painted onto the Overview. This command will generate a Metabolite Tracing dialog box, which can then be used to choose which metabolite to trace, how far to trace it, and other options, as described in much more detail in section .

Print as Poster The Cellular Overview diagram can be printed as a poster at 400% magnification. This command does not send output directly to a poster printer, but rather generates a file in postscript (.ps) format, which can then be sent to a poster printer or a commercial printing service, or converted to some other format as desired. The poster includes pathway, metabolite, and enzyme/transporter labels and can be customized in several ways. By default, each pathway class is colored a different color, but users can specify that their own custom highlights (e.g., some Omics dataset) be used instead. Users can supply their own poster title, explanatory text and copyright notice. The abovementioned customizations all use the Cellular Overview Diagram as it appears when viewed using the software. Alternatively, users can regenerate the diagram from scratch in order to change either what kind of information is displayed (e.g., to omit enzyme names, or to include EC numbers), font sizes (bear in mind that increasing the font size will cause the entire diagram to become larger), or the aspect ratio.

Update Regenerate the Cellular Overview diagram to reflect any changes in the database. This command is disabled if the database cannot be modified. Note that this operation can take quite a long time typically an hour or so depending on the speed of the computer and the number of pathways in the database.

Additional operations are accessible through the right mouse button. Right-clicking on a compound gives you the choice of accessing either a menu for the compound or a menu for the pathway. Right-clicking on a reaction gives you the choice of accessing a menu for the reaction, for the pathway, or for any of the compounds involved in the reaction (including those that are not displayed in the overview because they are side compounds). Right-clicking on any object also gives you the opportunity to zoom in or out of the overview. You may choose from one of several predefined magnification levels or you may specify your own. As the zoom level increases, more details become visible. For example, at 120%, reaction direction arrows become visible. At 200% magnification, pathway and metabolite labels become visible. At 300% magnification, enzyme labels appear. All these labels become larger and more readable at a magnification of 400%. Beyond 400%, the diagram will become larger, but no further information will be available.

The compound, reaction, and pathway menus are all described below.

Right-Button Compound menu:

Display compound information in main display Displays the selected compound in the main display window.

Display compound information in pop-up window Displays the selected compound in a new pop-up window.

Highlight all reactions of this compound Highlights all reactions that contain this compound as either a main substrate or a side substrate. This command is a more complete way of finding reactions of a compound than is the next command.

Highlight this compound everywhere it appears as a main Highlights all occurrences of this compound as a main substrate only.

Display all connections for this compound Highlights all occurrences of this compound as a main substrate, and draws dim lines connecting the clicked-on compound to all other occurrences of that compound, including situations in which the compound is a side substrate of a reaction (in this latter case, there is no icon for the selected compound, so the line is drawn to some other main compound icon in the relevant reaction). The lines are removed when the highlight operation is cleared or undone. These new lines are drawn slightly brighter than those used to link identical compounds to show flow of material between pathways.

Show Invokes the usual right-button Show sub-menu for the compound.

Edit Invokes the usual right-button Edit sub-menu for the compound.

Zoom Allows you to select the magnification level of the display.

Right-Button Reaction menu:

Display reaction information in main display Displays the selected reaction in the main window.

Display reaction information in pop-up window Displays the selected reaction in a new pop-up window.

Highlight this reaction everywhere it appears Highlights all occurrences of the selected reaction in the Overview.

Show enzymes and genes of this reaction in listener window Prints the names of the enzymes that catalyze this reaction, and the genes that encode those enzymes.

Display all connections for substrates of this reaction Draws dim lines connecting the substrates of the clicked-on reaction to all other occurrences of those compounds.

Display all connections for reactants of this reaction Draws dim lines connecting the reactants of the clicked-on reaction to all other occurrences of those compounds.

Display all connections for products of this reaction Draws dim lines connecting the products of the clicked-on reaction to all other occurrences of those compounds.

Highlight reactions involving genes in same operon/regulon Highlights in one color all genes that are in the same operon as the gene whose enzyme catalyzes the selected reaction; highlights in a second color all genes that are in the same regulon as the gene whose enzyme catalyzes the selected reaction. If the selected gene is in more than one regulon, you are asked to select the transcription factor defining the regulon of interest (a regulon is defined as the set of operons regulated by a specified transcription factor).

Show Invokes the usual right-button Show sub-menu for the reaction.

Edit Invokes the usual right-button Edit sub-menu for the reaction.

Zoom Allows you to select the magnification level of the display.

Right-Button Pathway menu:

Display pathway information in main display Displays the pathway containing the selected compound or reaction in the main window.

Display pathway information in pop-up window Displays the pathway containing the selected compound or reaction in a new pop-up window.

Highlight this pathway Highlights the pathway containing the selected compound or reaction.

Display all connections for compounds in this pathway Draws faint lines connecting most compounds in the pathway to all other occurrences of those compounds, including situations in which the compound is a side substrate of a reaction (in this latter case, there is no icon for the selected compound, so the line is drawn to some other main compound icon in the relevant reaction). Compounds in the pathway for which these lines are not drawn are either common small molecules (e.g., phosphate, water) or compounds whose function in the pathway is purely to act as a donor or acceptor of some small moiety (e.g., ATP as a donor of phosphate, glutamine as a donor of nitrogen, acetyl-CoA as a donor of CoA). The lines are removed when the highlight operation is cleared or undone. Lines from main compounds

in the pathway are drawn slightly brighter than those used to link identical compounds to show flow of material between pathways; lines from side compounds in the pathway are drawn slightly dimmer.

Show indexShow (Overview Submenu): Invokes the usual right-button Show sub-menu for the pathway.

Edit : Invokes the usual right-button Edit sub-menu for the pathway.

Zoom Allows you to select the magnification level of the display.

3.8.2.2.2 Displaying Reactions Corresponding to a Set of Genes To highlight a set of reactions corresponding to some gene set (such as reactions catalyzed by a set of essential genes or knockout genes), select menu **Overview** → **Highlight** → **Gene** → **Gene List from File**. This menu selection highlights all metabolic reactions catalyzed by the product of a gene listed in the file.

3.8.2.3 The Regulatory Overview

The regulatory overview initially shows the transcriptional regulatory network for the currently selected organism, in one window. Each icon in the diagram, such as a plus sign, depicts one gene. Not all genes in the genome are shown in the diagram. The genes shown are regulators (transcription factors and sigma factors) and all other genes for which the PGDB encodes regulatory information for the gene.

Every gene is displayed using an icon (shape) from . The genes are partitioned into three groups, each group being laid out on a separate ellipse. The two inner ellipses contain all the genes that regulate at least one gene. The inner-most ellipse contains the genes that regulate the most. Typically, about 15% of the regulators are in the inner-most ellipse. The outer most ellipse contains genes that are regulated but that do not regulate.

The outer-most ellipse is further partitioned into groups of genes such that all genes within one group are regulated by the same set of genes. We call these groups **multi-regulons**. Some groups are shown as a line or as a line leading to a triangle, perpendicular to the outer-most ellipse (see Figure 3.4). For example, a large blue multi-regulon is shown in the upper right corner of Figure 3.4. Note that although all genes within a multi-regulon respond to the same set of regulator genes, different genes in the group may be controlled in different ways. For example, consider a multi-regulon comprised of genes A and B, that are regulated by genes X and Y. Genes X and Y might both activate the transcription of A, but they might both inhibit transcription of B.

Arrows (edges) in the diagram depict regulatory relationships between genes. For example, an arrow pointing from gene X to gene A indicates that X regulates the transcription

Icon Shape	Gene Type
Square	A sigma factor
Plus Sign (+)	Has only activators
Minus Sign (-)	Has only inhibitors
Circle	None of the above

Table 3.2: Gene icons used by the Regulatory Overview

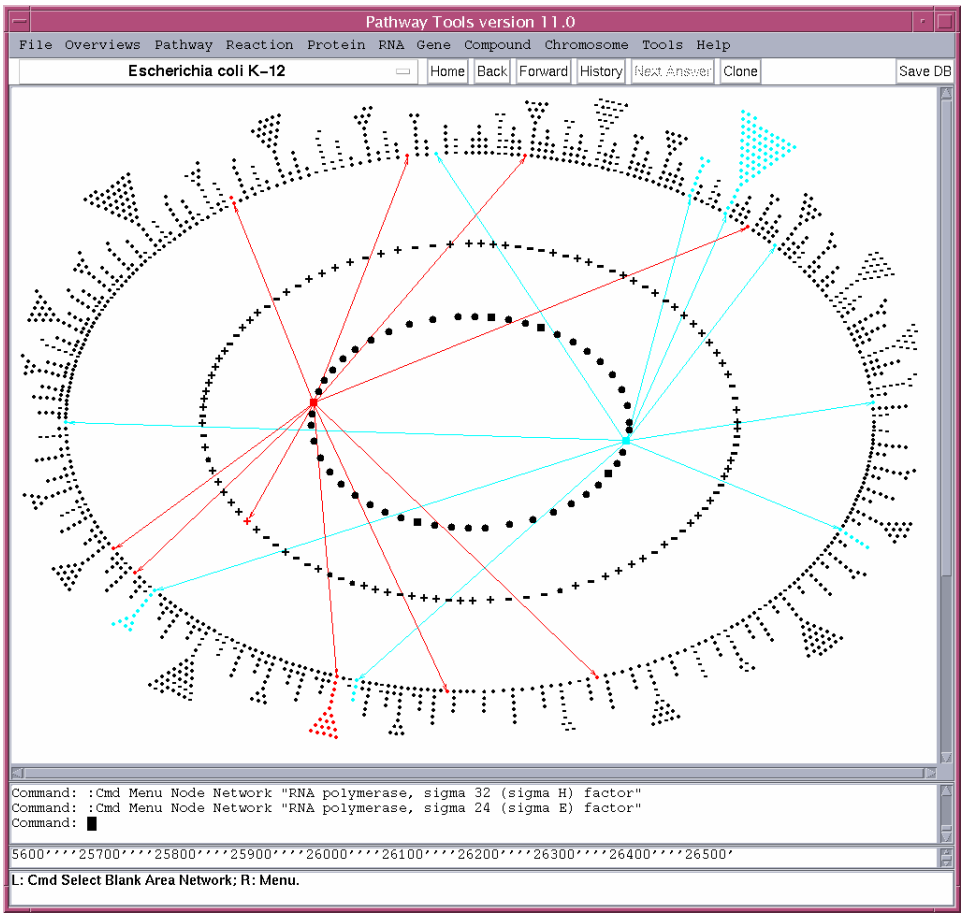


Figure 3.4: Complete Regulatory Overview with two highlighted nodes

of A. Initially, the regulatory relationships between genes are not displayed, that is, no arrows are shown between the gene icons. This behavior can be globally changed by using the command **Preferences for Regulatory Overview** (see Section 3.8.2.3.1). An arrow cannot go from an outer ellipse to an inner ellipse since the outer ellipse does not contain regulator genes. Gene names are not displayed next to the icons unless the preferences are changed or the icons are large enough. The icon sizes depend on the size (in pixels) of the pane (limited by the resolution of your computer screen monitor) and the number of

genes displayed in an ellipse.

When mousing over a gene icon, the horizontal pane at the bottom shows data related to this gene, in particular its name, product, regulators, and regulatees. A tooltip is also displayed next to the mouse pointer with the same data.

Right-clicking a gene icon opens up a menu from which a highlighting operation can be done on this gene and its relationships to other genes. You can highlight direct regulators and/or regulatees of a gene, as well as indirect regulators and/or regulatees. Each highlighting operation is done using a different color.

You can display a subset of the genes and their regulatory relationships by first highlighting them (see command **Highlight Genes and Regulatory Relationships** below or by right-clicking a gene as described above) and then using the command **Redisplay Highlighted Genes Only**. A different layout can also be used for this smaller overview.

There are actually two possible layouts: nested ellipses and straight rows. The preferred one can be chosen by using the command **Preferences for Regulatory Overview** under the menu bar **Overviews**. By default, when you start Pathway Tools for the first time, the nested ellipse layout is used for the complete overview whereas the straight rows layout is used when redisplaying the highlighted genes only.

The straight rows layout put in the top row the genes that directly regulate the most number of genes. The bottom row contains all the genes that do not regulate any genes. The intermediate rows always contain genes that regulate, but they control less genes then the top row. All the genes in one row do not directly regulate each other.

More precisely, the rows are created in the following way. The gene that directly regulate the largest number of genes is located first in the top row on the left. If the next largest gene regulator has no direct relationship with this one, it is located right next to it, and so on, until no regulators can be put on the top row. The next lower row restarts with the largest gene regulator left over, etc., until all regulators are laid out. This will typically create several rows of regulators before the bottom is reached. The bottom row lay out the genes that do not regulate any genes. The multi-regulon clusters are scattered on this bottom row to avoid icons to overlap. Note that all genes that are not regulated by some genes in this incomplete overview will go in one cluster. This is different then the complete overview where a cluster of genes is regulated by the same group of genes.

3.8.2.3.1 Regulatory Overview Commands Further operations pertaining to the regulatory overview can be found under the **Overviews** menu-bar. The commands are

Show Complete Regulatory Overview : Creates the initial regulatory overview display of the currently selected organism (see Figure 3.4). This command can also be used to redraw the full regulatory overview if a smaller regulatory network is displayed.

Highlight Genes and Regulatory Relationships : A sub-menu with four different search commands provided to select a set of genes to highlight with a color automatically selected by Pathway Tools. You can search by gene name, frame ID, gene substring names, gene ontology, or gene MultiFun. The highlighted genes will be the genes found in the regulatory overview. After selecting the genes to highlight from the search menu, another menu will open to ask if their regulatory relationships (arrows) should be drawn between them. Before answering the question, you can see the selected gene icons flashing. This give you the opportunity to locate, in the overview, the selected genes. Their name will also be displayed next to the gene icon in the overview. Once you answer the question by clicking one of the buttons, flashing will stop, the names will be erased if the icons are too closed to each other, and the relationships between them (i.e. the arrows) will be shown and highlighted in a specific color if you requested to do so. Otherwise, only the selected genes are highlighted.

Redisplay Highlighted Genes Only : The overview is redisplayed with only the highlighted genes. In this mode the diagram is no longer a global overview, but shows a subset of the full network. In most cases, the size of the gene icons will be increased since fewer genes will have to be displayed. It is most often in this reduced layout that gene names can be displayed next to the icons without overlapping each other (see Figure 3.5). The layout is in general different: the genes are displayed in rows instead of ellipses. (See the command Preferences for Regulatory Overview, below, to display the gene names.)

1. Zoom Regulatory Overview: You can change the scale of the diagram by zooming in or out.

2. Preferences for Regulatory Overview: This command allows you to control the appearance of the Regulatory Overview for the current Pathway Tools session. You can alter settings for displaying gene names, relationships between genes (i.e., arrows between gene icons), the basic geometric figure (e.g., ellipse, rounded rectangle) for the three groups of genes, and so on. There are two possible layouts: straight rows and nested ellipses. These can be applied for the complete overview or when only highlighted genes are displayed.

3. Save Current Regulatory Overview to File: This command saves the current Regulatory Overview, which could be a partial overview as displayed by the command Redisplay Highlighted Genes Only, and its current highlights.

4. Load Regulatory Overview from File: Redisplay a previously saved regulatory overview.

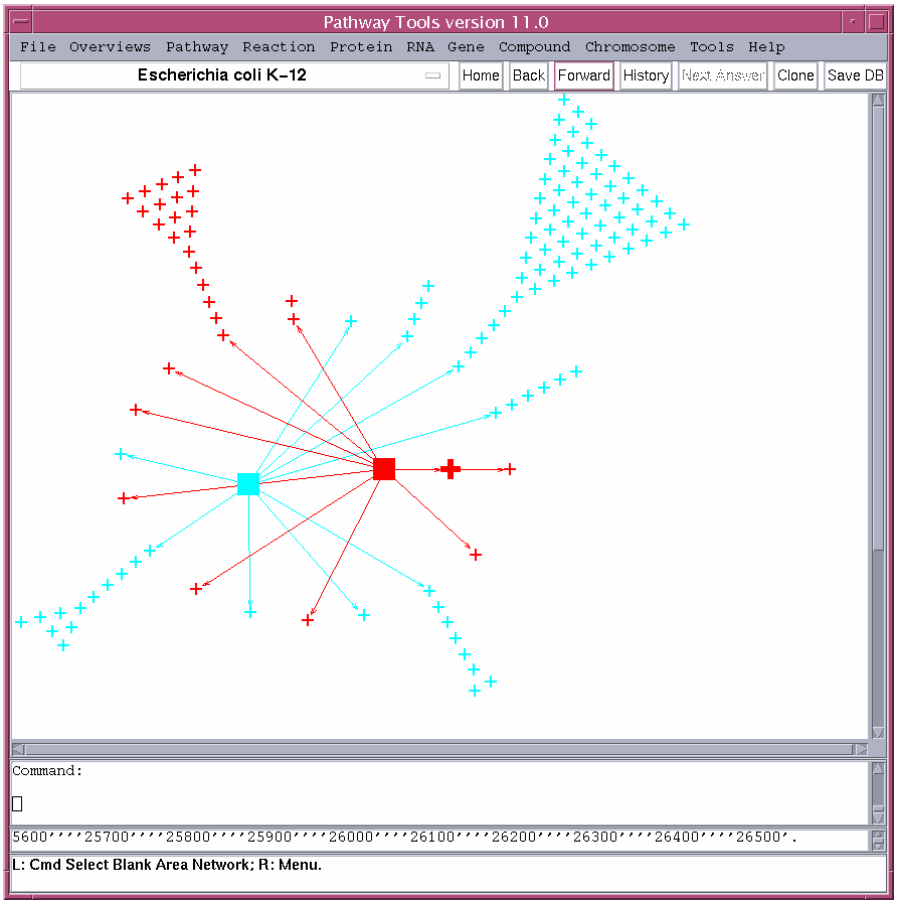


Figure 3.5: Regulatory Overview: Redisplay highlighted genes only

3.8.2.4 The Genome Overview

The Genome Overview shows in one screen all the genes in an organism's genome, as well as additional information about their transcription units and products. The Genome Overview has several key differences from the Genome Browser. Unlike the Genome Browser, the Overview is not to scale, nor does it reflect spacing between genes. Conversely, the Genome Overview shows all of the genome of an organism at once, even if that genome is split across multiple chromosomes or plasmids. Each individual replicon (chromosome, plasmid) is displayed on the page with an appropriate label identifying it.

The Genome Overview uses similar iconography to the Genome Browser, showing the direction of gene transcription with a sloping line on the top (ORF) or on the bottom (RNA-coding gene) of the gene. Lines underneath genes indicate the extent of transcription units, and are particularly useful for identifying multiple promoters. Transcription units are also indicated by shared gene color, although this coloring is replaced if you choose to map expression data onto the Overview using the Genome Omics Viewer func-

tion.

You can identify a gene in the overview by clicking on it to go to its gene page, or by mousing over it to display its gene name, product, and distance from neighboring genes at the bottom of the screen.

3.8.2.4.1 Genome Overview Commands

Show Genome Overview Draw the Genome Overview diagram for the current organism.

3.8.2.5 The Omics Viewers: Using Overviews to View Experimental Data

The Pathway Tools Omics Viewer uses the Cellular, Regulatory and Genome Overviews to illustrate the results of high-throughput experiments in a global metabolic and genomic context. The Genome and Regulatory Omics Viewers can map any dataset that focuses on genes (such as a gene expression study) onto the full genome of the organism, using a spectrum of colors to display the numerical values associated with each gene. The Cellular Omics Viewer can be used to illustrate an even wider range of high-throughput experimental results in a global metabolic pathway context. Genes (in the case of a gene expression experiment) and proteins (in the case of a proteomics experiment) that are involved in metabolism are mapped to reaction steps in the Cellular Overview, and the range of data values in a given experimental dataset is mapped to a spectrum of colors. Reaction steps in the Cellular Overview are colored according to the corresponding data value. Similarly, for metabolomics experiments, compound nodes are colored according to the data value for the corresponding compound. This facility enables the user to see instantly which pathways are active or inactive under some set of experimental conditions.

The Genome Omics Viewer can be used for

Microarray Gene Expression Data Genes are color-coded according to the relative or absolute expression level of the gene. shows a representative segment of the Genome Browser displaying results from a gene expression dataset.

Other Experimental Data Any experiment, high-throughput or otherwise, in which data values are assigned to genes can be viewed using the Genome Omics Viewer. One such possible use the mapping of a set of ESTs that have been assigned to genes onto a sequenced genome, thus offering a view of how much of, and which parts of, the genome are covered by that EST set.

The Cellular Omics Viewer can be used for

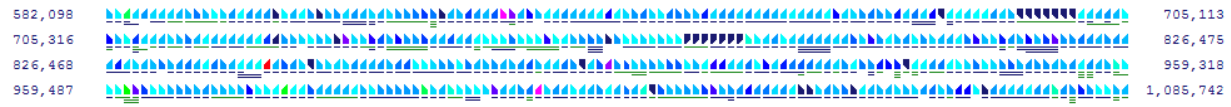


Figure 3.6: The Genome Omics Viewer displays gene expression results

Microarray Gene Expression Data Reaction lines (and protein icons, where present) are color-coded according to the relative or absolute expression level of the gene that codes for the enzyme that catalyzes that reaction step. The Cellular Omics Viewer allows a scientist to interpret the results of gene-expression experiments in a pathway context. shows a portion of the result from display of a gene expression dataset in the Cellular Omics Viewer. Note that this is the exact same file displayed in .

Proteomics Data Reaction lines (and protein icons, where present) are color-coded according to the concentration of the enzyme that catalyzes that reaction step.

Metabolomics Data: Compound icons are color-coded according to the concentration of the compound.

Reaction Flux Data Reaction lines are color-coded according to reaction flux values.

Other Experimental Data Any experiment, high-throughput or otherwise, in which data values are assigned to genes, proteins, reactions or metabolites can be viewed in a pathway context using the Omics Viewer.

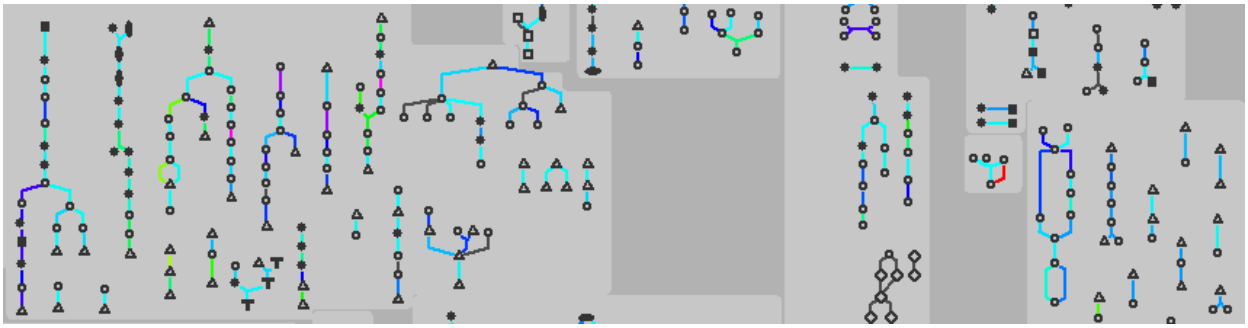


Figure 3.7: The Cellular Omics Viewer displays many kinds of high-throughput data

The Omics Viewer can show absolute data values (such as the concentration of a metabolite or protein, or the absolute expression level of a gene), or it can be used to compare two sets of experimental data by computing a ratio and mapping the ratios onto a color spectrum. Multiple sets of experimental data can be superimposed on the same overview diagram so that users can, for example, combine gene expression and metabolomics in the same figure, or view the results of two different microarray experiments together. When

combining multiple datasets, users should be careful to assign color schemes that avoid ambiguity.

The superposition of multiple sets of experimental data on the metabolic overview can also be animated to show, for example, how gene expression levels of enzymes change with time over the course of an experiment. The animation can be exported to HTML so that it can be published online.

After displaying Omics data on the Cellular Overview, navigating to any pathway display will show the Omics data superimposed on the individual pathway. If a particular reaction step has multiple isozymes then, rather than just choosing one value as is done on the Cellular Overview, all values are shown. Some colors may not show up as well against the pathway background (white by default) as they do against the gray Omics Viewer background — if this is a problem, either customize your colors to choose those that show up well against white, or use the Preferences menu to choose a gray background instead. To remove Omics data from individual pathway displays, select **Overview** → **Highlight** → **Clear All**.

3.8.2.6 Omics Dataset File Format

Experimental data is imported from a file that is provided by the user and is stored on the user's computer. Each line of the file contains data for a single gene, protein, reaction or metabolite, and is of the form:

```
name-or-ID data-column_1 ... data-column_N
```

Columns are separated by the **Tab** (`\t`) character. The first column contains the name, a known synonym for the name, or the unique identifier of a gene, protein, reaction or metabolite. Gene IDs from sequencing projects (such as the *E. coli* B-numbers) are generally acceptable and unambiguous. For protein or reaction data, EC numbers may be used.

The numbers in the data columns can represent either absolute or relative data values. If the numbers represent absolute numbers, then the software can compute relative values from two data columns. An entry (a row of data for an entity) may contain any number of data columns, but only the specified (one or two for a single experiment display, more for an animation) data columns can be visualized at one time. Reusing the same file in consecutive viewings improves display performance, as the file needs to be read only once.

Lines that start with either `#` or `;` are taken to be comments and are ignored by the program. The software uses the first row of data (i.e., the first line that is not a comment line) to determine the number of data columns to process. For example, if the first row contains five columns, only the first five columns of each subsequent row will be processed. Thus,

even if not all fields for the first row contain data, you must make sure that it contains the appropriate number of **Tab** characters.

3.8.2.7 Using Gene Expression Data from a SAM Spreadsheet

The Omics Viewer can import gene expression data from a spreadsheet generated by the SAM (Significance Analysis of Microarrays) Microsoft Excel plug-in (see `http://www-stat.stanford.edu/~tibs/SAM/`). This package combines multiple expression experiments to produce a list of statistically significant positively and negatively regulated genes. The Omics Viewer displays the positively regulated genes in one color, and the negatively regulated genes in another color. In order to import data generated using SAM, the SAM results sheet must be saved from Microsoft Excel in text format (not in Excel format). You can then select the SAM option for input to the Omics Viewer and supply the filename.

3.8.2.8 Color Scales

By default, the color scale used depends on the type and range of the data. Thus, for example, a particular color may correspond to one gene expression level for one dataset, and a different gene expression level for another dataset, depending on the range of values in each dataset. Alternatively, you can specify a color scheme to use, either by simply specifying a maximum color cutoff, or by supplying a list of cutoff values and their corresponding colors. By default, we use the spectrum from yellow to red, with yellow representing the lowest values or ratios in the dataset, and red representing the highest. Reactions for which no data was provided are shown in the Overview Key, which pops up automatically when experimental data is displayed (if it is not already visible).

For absolute data values, the spectrum is mapped evenly along a log scale between the log of the smallest value in the dataset to the log of the largest value in the dataset. The key shows actual values, however, not their logs, unless the supplied data is in log form to begin with. For relative data values, the ratio is fixed at 1 (or 0, for log values) at the center of the scale.

In many cases in the Cellular Omics Viewer, several genes or enzymes, each with its own expression level or concentration, will map to a single reaction, because the reaction might be catalyzed by an enzyme complex made of several gene products, or the reaction might be catalyzed by several isozymes, each with its own gene or genes. Since a reaction can be shown in only a single color, the following method is used to choose which of several values to display. For absolute data values, the maximum value is displayed; for relative data values, the value whose log has the greatest deviation from zero is displayed, under the assumption that the user is primarily interested in identifying the genes whose expression levels differ most between the two datasets.

3.8.2.9 Usage

To create an experiment view, select **Overview** → **Omics Viewer: Overlay Experimental Data from**, and specify whether data will be from a text file or from a SAM output file (see Sub-section 3.8.2.7). A dialog box pops up with the following fields (assuming data is from a text file — only a filename is needed for SAM output):

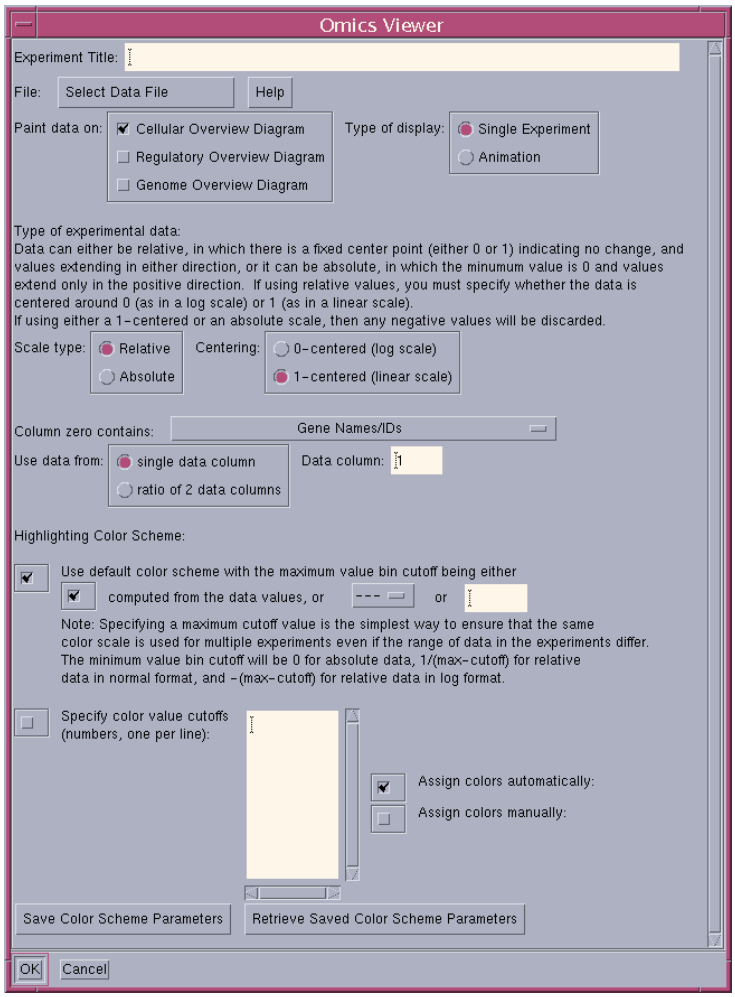


Figure 3.8: Omics Viewer input dialog

Experiment Title Enter a few words to describe the data being viewed. The title is displayed in the appropriate section of the Overview Key. This field is optional.

File Click on the button to select a file containing the experimental data. The **Help** button describes the required file format.

Reload? This option is available only if the specified file has already been loaded (i.e., if you previously displayed data from the same file). If not selected, the previously cached data is used. Select this option if the file has changed since it was last read.

Paint data on Here, you may choose to paint your data onto one or more of the Cellular, Regulatory, and Genome Overviews. Note that although many data types can be painted on the Cellular Overview, at the moment only gene datasets can be painted on the Genome or Regulatory Overviews. If you choose to paint onto both Overviews, a separate window will be generated for each simultaneously.

Type of display Select either **Single Experiment** or **Animation**.

Scale type Select either **Relative** or **Absolute**. Some of the later options in the dialog box, as well as the color scale used, will depend on this choice.

Centering Choose a zero-centered scale when the data values are logs, and a 1-centered scale when the data values are linear. Also choose the zero-centered option if the scale comprising the data is centered at zero (as opposed to 1) for other reasons, for example, when displaying reaction flux data in which the sign of each datum indicates reaction direction. If a 1-centered scale is used, any zero or negative values in the data are ignored.

Column zero contains The data file format requires that the first column (column zero) contain either gene, protein, reaction, or compound names or IDs. Specify which of these your data file contains. If your data file contains multiple types of data, you may select the last option, which allows column zero to contain any of the above types of data. Be careful when mixing different data types in the same file. Data values for one type of object may not be directly comparable to data values for a different object type. In addition, names that are unambiguous when looking only for a gene may become ambiguous when the object can be either a gene, protein, or compound.

Superimpose on previous display? This item is available only when the Omics Viewer is already being used to display another dataset. If this box is checked, then the new experiment view will be superimposed on the old one. If this box is unchecked (the default), then the previous display will be cleared before the new data is shown. [Note that this option has been disabled in the current version because of the reorganization of the Omics Viewer — we hope to bring it back in a future version of the software.]

Use data from This item is available only when relative data is displayed. If the ratios themselves appear in a column in the data file, select **single data column**. If you want the ratios computed from columns of absolute data values, select **ratio of 2 data columns**.

Use expression data from data column When using a single column, enter the column number containing the data to be viewed. The column containing the gene or protein names is considered to be column 0, so the data column numbering starts at 1. When displaying an animation, you should enter a list of columns, one on each line. Alternatively, you can enter a range — for example, 1-10 — to indicate that all columns in that range should be used.

Compute relative data values as the ratio of two data columns Enter the two column numbers containing the data to be compared. When displaying an animation, for the denominator you can specify either a single column, which will serve as the denominator for each numerator column, or you can specify a different denominator column for each numerator column.

Highlighting Color Scheme Here, you can elect to use either the default color scheme or one you specify.

Default color scheme with the maximum bin cutoff computed from the data This option is the simplest to specify. It is useful when you do not know much about the range of values in your data file or you want to see the color spectrum divided evenly across the full range of the data. Since this option will produce a different color scheme for every experiment, it is not useful for comparing figures generated across multiple experiments (though every time point in a single animation will use the same color scheme, of course).

Default color scheme with a specified maximum value bin cutoff All values above the maximum cutoff (or below the corresponding minimum cutoff) are displayed in the same color, and the full color spectrum is divided evenly over the space between the maximum and minimum cutoffs. You can specify a maximum cutoff either by selecting one from the menu of commonly used possibilities or by typing in a value yourself. If you specify the same maximum cutoff for multiple experiments, all will be displayed using the same color scheme.

Specify color value cutoffs, assign colors automatically To exercise more control over the display, you can provide the full list of cutoff values. This is useful, for example, if you are interested in grouping the data into only a few broad categories (e.g., 2x over/under-expressed, 10x over/under-expressed) and are not interested in finer gradations. Enter a list of numbers, one per line (e.g., 10, 2, 0.5, 0.1). Remember that if your data file has log values, the cutoffs should be log values as well (e.g., 0.3 rather than 2 to achieve a 2x cutoff).

Specify color value cutoffs, assign colors manually This option gives you maximum control over the appearance of the resulting display. After providing the value cutoff numbers, click on **Assign colors manually**. A color selector box appears containing a scale computed from your supplied cutoff values, and a color spectrum. Assign colors to cutoff bins by clicking on the button

corresponding to a bin and then clicking on the desired color. Click **OK** when done.

Save/Retrieve Color Scheme Once you have created a color scheme, you can elect to save it to a file, so you can later retrieve and reuse it. Use the **Save Color Scheme Parameters** and **Retrieve Saved Color Scheme Parameters** buttons for these purposes.

There will be a pause while the data is read and processed. Once processing is complete, a new window pops up for each selected overview, and an additional report window pops up containing a few statistics about the data and reporting any problem rows in the data file. Note that in whole genome microarray experiments, typically a large fraction of a given genome does not code for enzymes and therefore will not have any corresponding reaction in the Cellular Overview. Statistics are provided both for the dataset as a whole and for the subset of data shown painted onto the Overview. For an animation, a new window pops up containing the animation display, along with buttons to start, stop, and step through the animation. No reports are generated for animations.

The Overview Key, in addition to the legend for mapping colors to data values, also includes a histogram showing the distribution of data values across the range, by color. In the default coloring scheme, the range is broken down into 50 subranges. Histogram bars to the left of the central axis count the genes or other entities that actually appear in the Overview. Bars to the right of the central axis count the genes in the remainder of the dataset (i.e., those not in the Overview).

For gene expression experiments in the Cellular Omics Viewer, to see exactly what expression values correspond to a colored reaction, middle-click on the reaction. The expression value is displayed in parentheses after each gene name in the listener window. This capability is particularly useful when a reaction is catalyzed by several isozymes, to see the expression level for each individual isozyme. In the Genome Omics Viewer, mousing over a given gene will display at the bottom of the window its expression value, followed by its name, product, and the distance between it and its nearest neighbors.

3.8.2.10 Metabolite Tracing Using the Cellular Overview

The Metabolite Tracing facility permits users to trace the path of a metabolite through the metabolic network. Since the metabolic network is highly interconnected, there will typically be many such paths. Rather than attempting to trace all of them at once, this facility stops at branch points to allow the user to select which one or more paths should be followed.

Invoking the command **Overview→ Metabolite Tracing** brings up the metabolite tracing control panel. To start, the user should enter a starting metabolite and specify whether the path should be traced forward or backward. A forward trace direction searches for

paths that consume the starting metabolite. A backward trace direction searches for paths that produce the starting metabolite. Note that since many reactions are reversible, those reactions will appear in both types of path.

Once a trace has been started, the software will highlight steps proceeding from the starting metabolite until it reaches a branch point. Candidate branches will be highlighted and a selectable list of branch end compounds will be displayed in the control panel. Users can select a branch to follow either by choosing the corresponding compound in the control panel or by clicking on the corresponding reaction edge in the overview display. There is a notion of the current path, which has been selected by the user and is highlighted in one color, and the set of candidate next steps to choose from, which are highlighted in another color. The user can also choose whether branch points that were not followed are to be highlighted in a third color or not at all. At any point, the user can search for a particular metabolite in the network that has been explored so far. This will cause the path to that metabolite to become the chosen path.

Because a particular trace may contain segments from multiple pathways, it can be difficult to follow visually. Thus, the user can also choose to show just the currently selected path as a single pathway in a popup window by invoking the corresponding command from the control panel **View** menu. Further help and description of other metabolite tracing options can be obtained by invoking the control panel **Help** command.

3.8.2.11 Reachability Analysis

Given a starting set of metabolites (called the nutrients), the Reachability Analysis tool determines which reactions can fire, and which other metabolites are produced as a result of this, in an automated and iterative manner. This tool allows checking whether the reaction network is complete enough to produce a set of target metabolites, which could include a set of key intermediate building blocks that are essential for considering the cell to be alive.

The algorithm underlying reachability analysis is called "forward propagation". The following definition was published in [15]:

"The forward propagation problem: given a set of input nutrients, what compounds will be produced by the SMM (Small Molecule Metabolism) when it metabolizes those nutrients? We seek a qualitative prediction of what compounds will be produced, but not the quantities of those compounds."

The inputs are:

- a set S of starting metabolites, called nutrients.
- a set B of bootstrap metabolites, called auxiliary nutrients (see below).

- optionally, a set T of target metabolites that are the goals to be produced.

The forward propagation algorithm used by the Reachability Analysis tool is as follows:

A set S of starting metabolites is defined by the user. The set of reactions available for metabolism is determined from the PGDB, consisting of all reactions that are assigned to metabolic pathways; plus reactions that stand alone, but which use only small-molecule metabolites. The reactions that came from pathways may use some macromolecular substrates, such as proteins that are modified.

During forward propagation, each reaction is checked for whether it can be "fired". A given reaction will fire when all its reactant (input) metabolites can be found in either the set S of starting metabolites or the growing set P of metabolites that are being produced by reactions that have already fired. Once a reaction has fired, its product (output) metabolites will be added to the set P. The reaction firing is iterated until no further reactions can be fired, and no further metabolites can be added to P.

Reaction directionality is taken into account, such that a unidirectional reaction can only fire if the reactants are present. Reversible reactions can fire in either direction.

The primary output of the algorithm is the set P of produced metabolites, and a set F of all reactions that fired. The secondary output is the set PT of target molecules that could be found in P, and the set UPT of target molecules that could not be found in P.

Bootstrap metabolites: Circularities can occur in a metabolic network, whereby some metabolites are needed for their own synthesis. For example, although glycolysis produces ATP, it requires an early input of ATP before any ATP is produced. To model this situation in forward propagation, ATP must be specified as an initial bootstrap metabolite. To this set of metabolites, other metabolites may need to be added, when it is known that these metabolites are essential for the cell, but for which the biosynthetic pathways may not yet have been completely elucidated. So this can be used to bypass known gaps in the reaction network.

The Reachability Analysis tool is invoked by the menu item **Tools→Reachability Analysis...**, which brings up a panel that allows setting up computational reachability experiments, recording their results in files, and examining past results.

At the top of the panel is a box that lists previously saved files, if there are any, and it gives a quick summary of the results contained in the file, in terms of how many reactions fired, and how many of the target metabolites could be produced. Most of the files are read-only, so past results can be re-examined, but not accidentally modified. Selecting the "New - Unsaved" item allows creating a new experiment setup from scratch. Selecting an existing file, and then clicking on the **Duplicate File** button allows creating a new file, which has the setup contents filled in, duplicating those contents from the selected file. However, the results are left blank, the idea being that some modification to the setup will be made, and then the reachability analysis will be run on the modified setup.

It is also the case that the very last (bottom) file is still editable, when it is selected. It will be frozen in read-only mode once a newer file is created.

Setting up a reachability experiment consists of defining 3 sets of metabolites. The **Growth Medium** lists the nutrients that comprise the set S of starting metabolites. The **Biomass Composition** lists the set T of target metabolites. And the **Auxiliary Nutrients** lists the set B of bootstrap metabolites.

The first two of these sets are stored in the PGDB in frames of the corresponding classes. The panel allows selection of these frames, when they already exist. A selected frame can be edited by clicking on the **Edit/Create New...** button. When the selector says "Not Specified Yet", then clicking the button will allow creation of a new frame.

The editing panels for these two classes of frames both contain list-based sections allowing the sets of metabolites to be specified, with buttons to add metabolites by exact or by substring match, or to delete a metabolite from the list. Additionally, for each metabolite, a comment can be added in the text box beneath the list, which can be used for justifying the presence of the metabolite.

The **Growth Medium Editor** allows specifying two lists of metabolites, the Major Nutrients and the Trace Nutrients, which is a distinction made in the traditional literature on growth media. However, for the reachability analysis, the union of those two nutrient sets is used. The **Biomass Composition Editor** contains just one list, the Essential Compounds.

The third set of metabolites, the Auxiliary Nutrients, can be specified directly in the main panel, using the same type of list-based dialog. This information is stored in the file for the experiment, and not in the PGDB. The rationale is that definitions for growth media are rarely modified, whereas the auxiliary nutrients are likely to be actively tinkered with to make the reachability work adequately.

Once these three sets of metabolites have been defined, the **Invoke Reachability Analysis** button can be clicked. After this finishes, a timestamp is shown, the results are saved in the file carrying this timestamp, and the quick summary of reactions fired and metabolites produced is updated.

After results have been produced, clicking the **Display on Cellular Overview** button will paint the results on the overview in the Navigator window. The mouse-over tooltips for reactions in the overview show information about whether the reaction fired, and which metabolites are present or absent, to help understand why certain parts of the reaction network may not have fired.

Please note the following current limitations:

Correspondance between compound classes and instances: The Pathway Tools schema tries to faithfully model reactions in the EC system of enzyme classification, where numerous reaction equations are written in terms of compound classes, to capture the broad

substrate specificity that some enzymes exhibit. An example compound class is: *cis,trans*-polyisoprenyl*n*-PP , which stands for several metabolites that would have varying number of *n* subunits.

Other enzymes are very specific, however, regarding the exact metabolite they require, out of the larger pool of similar metabolites represented by a compound class. For example, several enzymes involved in fatty acid elongation in *E. coli* process 3-hydroxy-acyl-ACP intermediates of varying chain lengths, whereas a few enzymes involved in lipid-A synthesis require the one instance of chain length 14, called (R)-3-hydroxymyristoyl-ACP.

Pathway Tools can infer some of these class-instance correspondances automatically, but not all of them. In the Reachability Analysis tool, when a reaction equation is written in terms of a compound class on the reactant and product sides, an inference mechanism generates additional reaction equations on the fly that are written in terms of instances occurring in the classes of the equation, if the instances are the only ones that mass-balance the reaction equation. Some class-instance correspondances can not be found automatically, therefore some chains of reactions may not fire that really should have, according to biochemistry.

3.8.3 Pathways

The Pathway/Genome Navigator automatically produces diagrams of biochemical pathways that are familiar to biochemists — the drawings that mimic those found in biochemistry textbooks. Because the power of computer graphics exceeds that of the printed page in several respects, the computer-generated diagrams provide more flexibility than those found in textbooks by allowing pathways to be expanded, contracted, and combined, and by adding additional information (e.g., regarding regulation).

Pathway diagrams show a set of interconnected chemical reactions, the enzymes that catalyze those reactions, and the reacting substrates, as in .

Substrates are drawn in two ways. The compounds that are shared between subsequent reactions lie along the backbone of the pathway; they are called main compounds or simply *mains*. Side compounds are drawn adjacent to the reaction arrow, with a curved arrow showing whether they are consumed or produced by the reaction. Enzyme names are drawn on the other side of the reaction arrow.

Pathway diagrams may include arrows showing regulatory interactions among the substrates and enzymes of a pathway. A dashed arrow leads from a substrate to a "+" or "–" sign adjacent to the enzyme whose activity the substrate modulates. The "+" or "–" indicates whether the effect on enzyme activity is positive or negative. Some enzymes are modulated by additional compounds that are not substrates in the reaction. Mousing over the "+" or "–" displays a tooltip that lists all compounds that activate and inhibit the enzyme, respectively, either by direct enzyme modulation or by activating or repressing

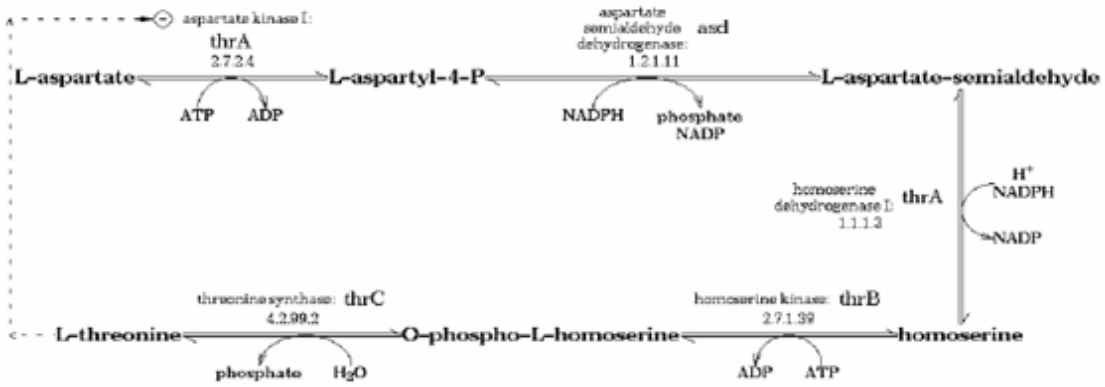


Figure 3.9: EcoCyc pathway: threonine biosynthesis

transcription of the gene that codes for the enzyme. The display page for the enzyme shows a more detailed breakdown of the activators and inhibitors into different classes (e.g., allosteric, competitive). Click on the enzyme name to see that page.

One type of relationship among pathways is shown within a pathway display under the headings "Superpathways" and "Subpathways". EcoCyc contains superpathways that are defined as connected aggregations of smaller pathways. For example, when the pathway for tryptophan biosynthesis is displayed, the Superpathway subheading lists a superpathway called "superpathway of phenylalanine, tyrosine, and tryptophan biosynthesis". Clicking on the name of this superpathway navigates to it to show the synthesis of all three aromatic amino acids from chorismate.

All pathway drawings are computed automatically using pathway-layout algorithms devised by SRI's Bioinformatics Research Group. Although SRI continually improves these algorithms to produce more intuitive and informative displays, be aware that the algorithms sometimes produce unintuitive results.

A somewhat novel aspect of the pathway displays is the use of branching reaction arrows to represent complex relationships among reactions, as shown in Figure 3.10 . For example, (a) depicts a situation in which one reaction converts two reactants A and B to the product C; two different reactions produce A and B, as shown by the in-pointing reactions. Five different reactions transform A to B in situation (f). The situation in (g) involves two reactions that convert A to X, but only one of those reactions involves the reactant B.

Another aspect of our pathway displays that is not typical of textbook pathway drawings is the depiction of polymerization steps. A dashed line indicates that two compound names are in certain situations meant to represent the same species. For example, most textbooks depict saturated fatty acid elongation as a spiral, where each turn of the spiral

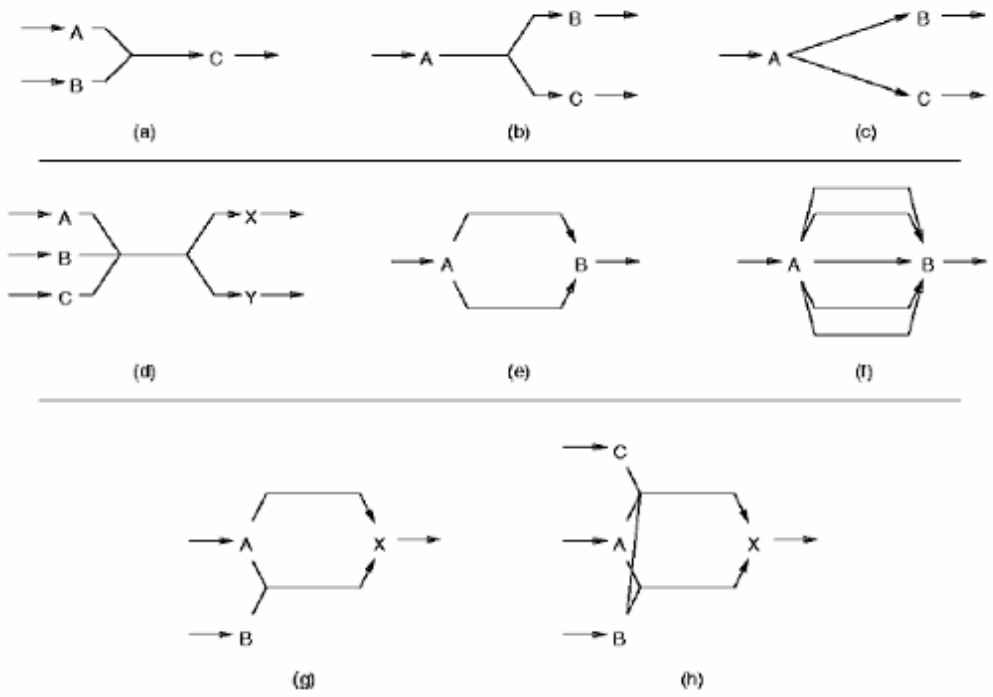


Figure 3.10: Examples of branching reaction arrows

adds two carbons to the backbone. Our representation shows the pathway as a cycle, using generic rather than specific names for the compounds involved. At the "beginning" of the cycle is acyl(n)-ACP, which undergoes several reactions producing acyl(n+2)-ACP. A dashed line is drawn between these two names to indicate that the (n+2) species becomes the (n) species for the next iteration of the cycle. We also use the dashed line when showing equivalence between a specific name for a compound (such as a starting or ending compound for a series of polymerization reactions) and the generic form. Using this scheme, we can compactly represent polymerization pathways as cycles of generic compounds, with specific compounds as inputs and/or outputs.

A small circle at the bottom of the pathway display depicts the positions of the genes that encode the enzymes within the current pathway on the *E. coli* genomic map. When you move the mouse over a given gene, its name and map position are printed at the bottom of the display window, and all reactions in the pathway involving the gene's enzyme are highlighted; clicking on the gene displays a window for that gene. Also included is a graph showing which transcription factors affect transcription of the genes in the pathway.

3.8.3.1 Pathway Commands

Search by Name or Frame ID (see "Direct Queries" in Section 3.7.1 for a general description of the query by name command)

Search by Substring (similar to other substring queries)

Search by Class You choose one or more pathways by first selecting one class from a menu of pathway classes, and then one or more pathways from a menu of all *E. coli* database pathways in that class.

Search by Organism In multiple organisms PGDBs, like MetaCyc, you query pathways according to the species in which they occur. You are first asked to select one or more species using the NCBI Taxonomy Browser restricted to all species defined in the PGDB. You can select one or more pathways from a menu of all pathways that are known to occur in those species. This command is enabled only for multiple organisms PGDBs.

Search by Substrates You can search for pathways according to the compounds that participate in their component reactions. As many as 20 compounds can be specified and each compound can be constrained as an input to the entire pathway, a net product of the entire pathway, a reactant in any component reaction, or a product of any component reaction. Pathways are retrieved only if *all* specified criteria are met.

3.8.3.2 Command Buttons

[More Detail / Less Detail]The Pathway/Genome Navigator can customize pathway drawings in a variety of respects by filtering more or less information from the drawings. For example, EC numbers and gene names can be displayed or hidden. Compound structures can be drawn and pathways can be illustrated as a skeletal overview that is restricted to compounds at the exterior of the pathway and at branch points. Although you can specify preferences (see " in Section) to provide fine control over pathway drawings, these command buttons provide a fast and easy way to increase or decrease the amount of detail shown in a pathway drawing.

Enzyme View]MetaCyc contains enzymes from many different organisms. By default, a pathway display in MetaCyc includes associated enzymes from any of the species for which the pathway is listed as having data available. Alternatively, your can choose to display enzymes from only a single organism by selecting that organism for the Enzyme View.

3.8.4 Reactions

A reaction display shows the class(es) containing the reaction within the classification of reactions. It shows the enzyme(s) that catalyze the reaction, the gene(s) that codes for the enzymes, and the pathway that contains the reaction. The displays show the EC number for the reaction and the reaction equation. Note that there exists a one-to-one mapping between EC numbers and reactions, but not between EC numbers and enzymes [17], which is why we label reactions, and not enzymes, with the EC number. The standard change in Gibbs free energy of the reaction is listed when known.

The direction in which the reaction is drawn depends on the user's preference settings. The default behavior is for the reaction to be drawn in the direction in which the reaction is defined by the Enzyme Commission, or the direction in which the reaction is stored in the database, for reactions that do not have assigned EC numbers. The alternative behavior is for the reaction to be drawn in the direction in which it occurs in a pathway.

Links to the ENZYME and LIGAND databases by EC number are shown.

Many of the preceding items are mouse sensitive. For example, if you click on the name of an enzyme, gene, substrate, or pathway, the Navigator displays that object.

3.8.4.1 Reaction Menu

Search by Name or Frame ID (see "Direct Queries" in Section 3.7.1 for a general description of the query by name command) Reactions typically do not have names, although in some cases the name of the enzyme that catalyzes a reaction can be used to retrieve a reaction. This command is most useful for calling up a specific reaction by its frame ID.

Search by Substring

Search by EC # Allows you to call up a reaction by its Enzyme Commission number.

Search by Class You can choose one of the reaction classes defined by the Enzyme Nomenclature committee [17] from a menu of all such classes. You are presented with a menu of all reactions within that class; your selected reaction is displayed.

Search by Pathway You choose a pathway by first selecting from a menu of pathway classes, and then from a menu of all pathways in that class; a third menu lists all reactions within that pathway. Your selected reaction is displayed.

Search by Substrates You specify one or more desired reactants and/or products, and you can choose from a list of reactions meeting these criteria. Because all reactions are considered reversible for these purposes, there is no real distinction between reactants and products. However, if the "Constrain compounds to specified sides?"

box is checked (the default), then, in order for a reaction to meet the criteria, compounds specified in the reactants section must all be on the same side of the reaction, and on the opposite side to all compounds specified in the products section. If this box is unchecked, there is no distinction between compounds specified as reactants and compounds specified as products—any reaction that contains all specified compounds as either reactants or products meets the criteria.

3.8.5 Proteins

Protein displays are fairly complicated because of the many-to-many relationship between enzymes and reactions (one enzyme can catalyze multiple reactions, and one reaction may be catalyzed by multiple enzymes). Furthermore, each catalytic activity of an enzyme may be influenced by different sets of cofactors, activators, and inhibitors. Also, many genes can code for subunits of a protein complex. The protein display is usually divided into sections to address these complexities. (See [10] for details of our representations of enzymes and activators.)

The first section of the display lists general properties of the protein, such as synonyms, molecular weight, pI, cellular location, and subunit structure. If the protein is itself a substrate in one or more biochemical reactions, those reactions are listed and sorted by the pathways in which they occur. If EcoCyc records that the protein is modified by some chemical group, a drawing of the protein coupled to the appropriate structure is shown.

Subsequent sections of the display describe each catalytic activity of the protein, if it is an enzyme (see Figure 3.11 and Figure 3.12).

Each activity section lists a reaction catalyzed by the enzyme, and the enzyme name (and synonyms) for that activity. The substrate specificity of the enzyme is described in some cases by listing alternative compounds that the enzyme will accept for a specified substrate. The cofactor(s) and prosthetic groups required by the enzyme are listed next, along with any known alternative compounds for a specified cofactor. Activators and inhibitors of the enzyme are listed, qualified as to the mechanism of action, when known. In addition, this section indicates which of the listed activators and inhibitors are known to be of physiological relevance, as opposed to whether the effects are known purely because of *in vitro* studies.

The direction in which the reaction is drawn (i.e., $A + B \rightleftharpoons C + D$ vs. $C + D \rightleftharpoons A + B$) depends on the user's preference settings. The default behavior is for the reaction to be drawn in the direction defined by the Enzyme Commission. Conversely, for reactions that do not have assigned EC numbers, the reaction is drawn based on the manner it was stored in the database. The alternative behavior is for the reaction to be drawn in the direction in which it occurs in a pathway, if known, or the direction in which the database indicates that the current enzyme tends to catalyze the reaction, if specified.

Additional sections of the display list each component (subunit) of the protein; when those components are polypeptides, the gene encoding the polypeptide is shown. These sections, and the first section, also list the molecular weight and pI of the subunits and the protein complex, when known.

If the protein is a transcription factor, the list of known transcription units (operons) controlled by the transcription factor is displayed in the protein display. See "Transcription Units" in Section 3.8.9 for more information about interpreting displays of transcription units.

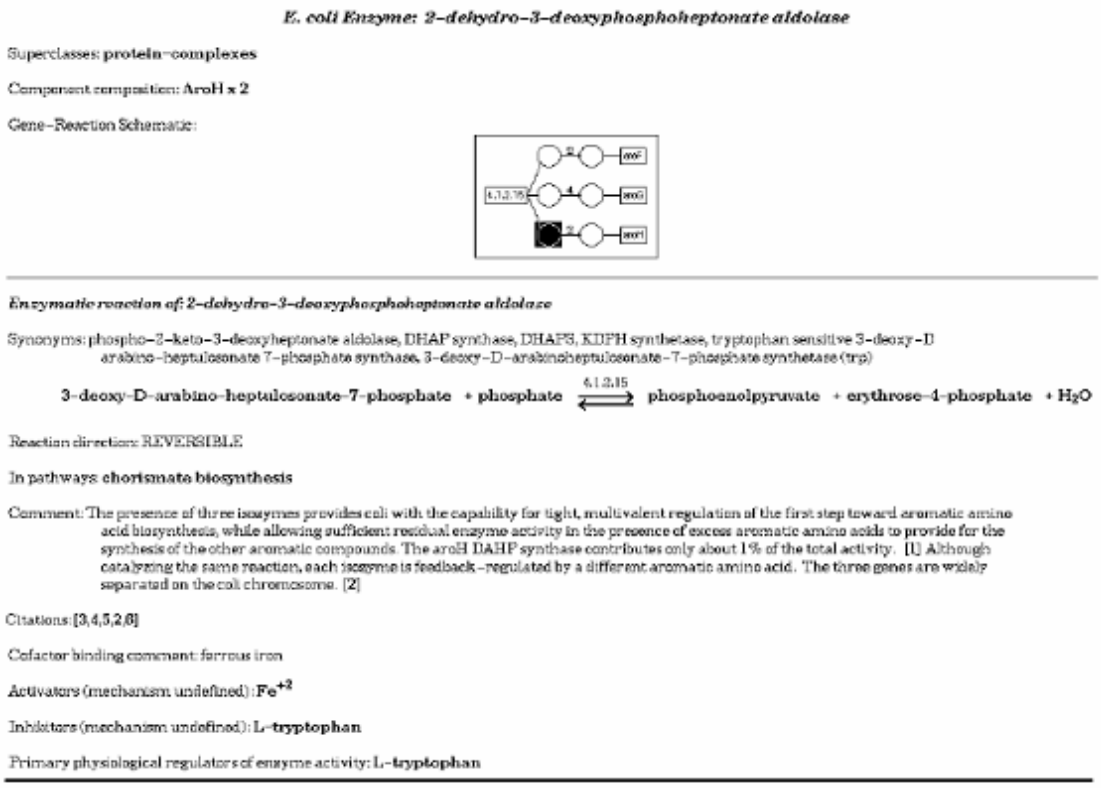


Figure 3.11: Upper portion of protein display window for 2-dehydro-3-deoxyphosphoheptonate aldolase

3.8.5.1 Protein Menu

Search by Name or Frame ID (see "Direct Queries" in Section 3.7.1 for a general description of the query by name command)

Search by Substring The program prompts you for one or more substrings, and then searches for proteins whose common name or synonyms contain all the substrings you entered.

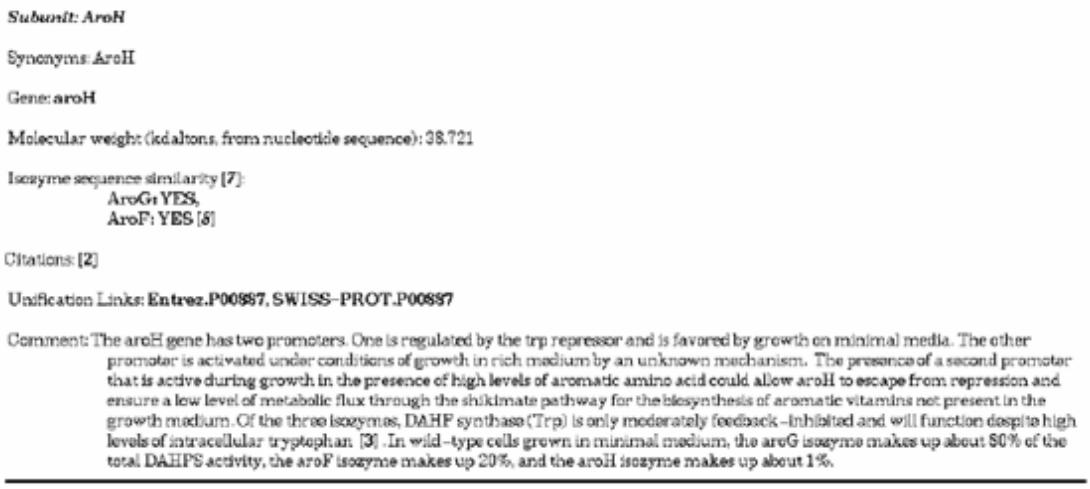


Figure 3.12: Lower portion of protein display window for 2-dehydro-3-deoxyphosphoheptonate aldolase

Search by Pathway You choose a pathway by first selecting from a menu of pathway classes, and then from a menu of all *E. coli* database pathways in that class; a third menu lists all enzymes within that pathway. The chosen enzyme is displayed.

Search by Organism In multiple organism PGDBs, such as MetaCyc, you can query proteins according to the species in which they occur. You are first asked to select one or more species using the NCBI Taxonomy Browser restricted to all species defined in the PGDB. You can select one or more proteins from a menu of all proteins that are known to occur in those species. This command is enabled only for for multiple organism PGDBs.

Search by UniProt Acc You query proteins by their UniProt Accession Number; of the form P12345 (see `http://www.uniprot.org/docs/userman.htm#AC_line` for the exact specification).

Search by GO Term Select a Gene Ontology term either by browsing the GO hierarchy, by searching for a particular GO term ID, keyword, or some combination. Despite the name, GO actually describes gene products rather than genes themselves (though the distinction tends to be meaningful only when multiple splice forms are involved). Thus, all the proteins associated with the selected GO terms will be retrieved. The first matching protein is displayed, and all others are placed on the Answer List.

Search by MultiFun Term MultiFun is an alternative scheme for classifying genes and their products. Browse and select a MultiFun class to see matching proteins. The first matching protein is displayed, and all others are placed on the answer list.

Search by Weight, pI You query proteins by molecular weight and pI value. Proteins are retrieved only if they match all specified criteria. Empty fields are treated as unspecified and are ignored.

Search for Enzyme by Modulation You query enzymes according to the activators, inhibitors, or cofactors that modulate their activity. The dialog box for this command allows you to select the type of modulation (e.g., activation, inhibition), and then allows you to select the compound of interest.

With a button labeled **Protein Sequence** within the protein display, you can retrieve the amino-acid sequence of a polypeptide. The amino-acid sequence is computed on demand by translating the nucleotide sequence stored for each gene. This button appears multiple times within the protein display for a heteromultimer, to allow you to retrieve sequences for each polypeptide chain within the multimer.

3.8.6 RNAs

RNA displays contain comments and citations for an RNA, a button to display the RNA sequence, and a link to the gene that encodes it.

3.8.6.1 RNA Menu

The RNA menu items are analogous to the menu items by the same name in the other object type menus.

- Search by Name or Frame ID
- Search by Substring
- Search by Class

3.8.7 Genes

The gene display lists information for a gene such as its map position on its chromosome, the functional class(es) assigned by Riley [14], Gene Ontology terms, and the direction of transcription. The gene product is listed (when known); when the product is an enzyme, the display shows the equation(s) of the reaction(s) catalyzed by the enzyme, and the pathways that contain those reactions.

A button located near the gene map position brings up the genome browser zoomed and centered on the region containing the gene (see Section 3.8.10 for a description of the Genome Browser).

If this gene is known to be interrupted (by a stop codon), the display prints a message to this effect.

The PGDBs for some organisms contain definitions of paralogous groupings of genes within the organism, which are usually computed using sequence-clustering methods. If a gene is known to be a member of one or more paralogous groups (genes containing multiple domains can be part of multiple groups), a message to this effect is printed, along with the name(s) of the paralogous group(s). Clicking on the name of a group displays all genes in that group; the genes within that list are themselves clickable. The display of a paralogous gene group also shows the chromosomal locations of all genes within the group.

The bottom of the gene display shows the local context of the gene in its chromosomal location. The display includes the upstream and downstream open reading frame, the transcription unit, transcription start sites, transcription factor binding sites, and applicable terminators. If the transcription unit(s) containing the gene is known, it is displayed below the local context display (see "Transcription Units" in Section 3.8.9 for more information about interpreting displays of transcription units).

In eukaryotic organisms, if the gene contains introns, a graphic shows their locations. Alternative splice forms are displayed.

3.8.7.1 Gene Commands

Search by Name or Frame ID (see Section 3.7.1, for a general description of the query by name command)

Search by Substring The program prompts you for one or more substrings, and then searches for genes whose common name or synonyms contain all the substrings you enter.

Search by Gene Ontology Term Brings up the Gene Ontology Browser allowing navigation to a gene based on its annotations.

Search by MultiFun class Choose a functional class from a menu of MultiFun gene classes, and then choose one or more genes from a list of all the genes in that class.

Three buttons at the top of the gene display provide access to sequence information:

Nucleotide Sequence Retrieves the nucleotide sequence of the gene (coding region).

Nucleotide Sequence Neighborhood Retrieves an arbitrary nucleotide sequence with endpoints specified by the user but defaulting to the gene boundaries.

Protein Sequence Retrieves the amino-acid sequence of the gene product. The amino-acid sequence is computed on demand by translating the nucleotide sequence stored for each gene. If the gene codes for multiple proteins (due to alternative splice forms), you must select which splice form to view.

3.8.8 Compounds

A compound display lists the common name and synonyms for a compound, plus its parent class or classes within the classification of compounds. It shows a compound's two-dimensional structure, plus its empirical formula, molecular weight, and pK_A when known. The display lists all reactions in which the compound appears, sorted by the pathways that contain each reaction.

The display of some chemical structures within compound displays uses a concept called *superatoms*, which is a hierarchical structuring of chemical structures. For example, when displaying the structure for succinyl-CoA, the structure is initially displayed with the word "CoA" in place of the structure of the CoA moiety. If you click on the word CoA, however, the full structure of that moiety is displayed.

3.8.8.1 Compound Menu

The compound menu can be used to query for small molecules such as pyruvate, D-glucose, and ATP.

Search by Name or Frame ID Usually several synonyms are recorded for each compound (and for other objects) — you can retrieve the compound by any of these known names.(see "Direct Queries" in Section 3.7.1 for a general description of the query by name command)

Search by Substring The program prompts you for one or more substrings and then searches for compounds whose common name or synonyms contain all the substrings entered by the user.

Search by SMILES substructure You are prompted to enter a structure in SMILES format, and then choose one or more compounds from a menu of compounds containing the target structure (the Navigator online help system describes the SMILES format).

Search by Ontology Browser You can browse the Compound Ontology and select one or more compound classes or compounds from the Ontology Browser. You can also search by name, frame ID or substring within the browser.

Advanced Search This command displays a dialog box that allows you to specify criteria for compound name, molecular weight, chemical formula, and substructure. Compounds are retrieved if they match all the specified criteria. Empty fields are treated as unspecified and are ignored. For the exact or partial matching of chemical formulae, it is important to enter in the chemical formula in a case-sensitive manner. In other words, "NH4" is considered different than "nh4". This is necessary because some element symbols use both upper and lower case, and there might be ambiguity if all of the adjacent letters were upper or lower case.

3.8.9 Transcription Units

Some PGDBs contain information about the clustering of genes into transcription units, the transcription factors that control a transcription unit, the location of transcription start sites, and transcription factor binding sites within a transcription unit. We define a transcription unit as the set of genes, DNA control sites, and transcription factors associated with one transcription start site. When a set of genes is transcribed from more than one transcription start site, those genes are part of more than one transcription unit.

The display of a transcription unit shows the transcription start site, transcription factor binding site(s), gene(s), transcription terminator(s) and attenuation interactions associated with that transcription unit, when known. Directly below the transcription start site, the base pair position of the transcription start site is printed, as is the direction of transcription ("+" for clockwise). The transcription factor binding sites and attenuation regulators are displayed in two different colors (which depend on the current color preferences). By default those colors are red when binding of the transcription factor results in inhibition and green when binding stimulates transcription.

The drawing of the transcription unit can be used for navigation within the PGDB. Clicking on a gene provides the gene display for that gene. Clicking on a transcription factor binding site displays the protein display for the transcription factor, which lists all other transcription units controlled by that transcription factor. Clicking on the transcription start site produces a transcription unit display window such as that shown in , which displays detailed information about each site within the transcription unit, including its nucleotide position and sequence, evidence for the site, and literature citations for the site. Each site is numbered to unambiguously identify multiple sites for the same transcription factor.

Currently, there exists no menu for querying transcription units directly — they can be found only by starting with the transcription factors that control them or the genes con-

tained within them.

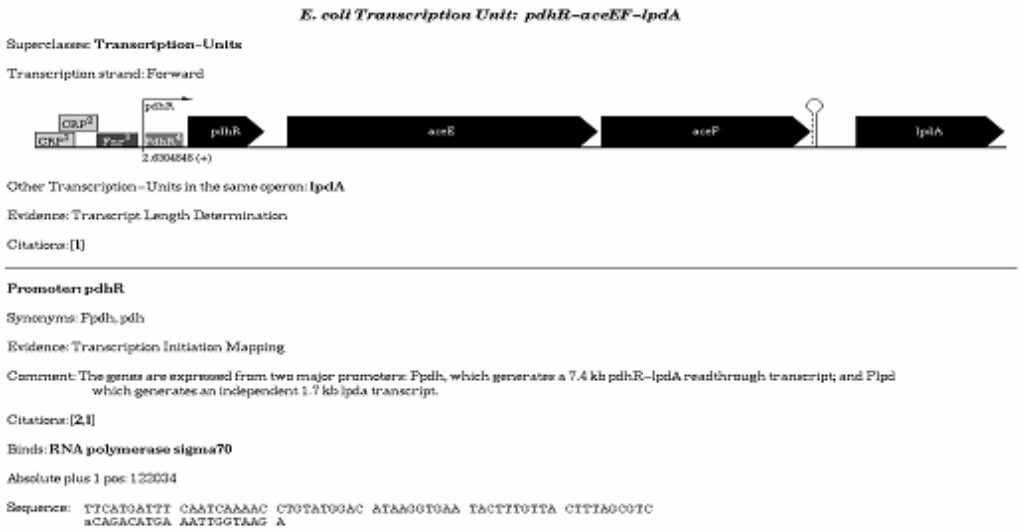


Figure 3.13: Transcription -unit display window

3.8.10 Genome Browser

The genome browser can be used to examine one replicon (chromosome or plasmid) at a time. It can be invoked by clicking on a replicon listed on the single organism display, from a gene display by clicking on the base-pair coordinates mentioned on the map position line, or from the menu item **Chromosome** → **Select & Browse Chromosome/Replicon** .

At the top of the display, the full length of the chromosome is shown at low resolution. A region of the chromosome can be selected for display at much higher magnification in the lower part of the screen. The selected region will be drawn using as many lines as will comfortably fit on the screen, often five lines. The full chromosome view at the very top indicates the magnified region by means of a red, rectangular cursor.

Selection of the magnified region can be achieved by the following methods:

Clicking on a position within the full chromosome line at the top will show the immediate neighborhood of that position. In the desktop version, one can click anywhere on the full chromosome. However, through the Web, only the tick marks are clickable. The tick marks in the magnified region can also be clicked on, to recenter the region around the selected tick mark quickly.

Start and end base-pair positions can be entered in the corresponding text entry boxes; clicking the Go button displays that region.

The region around a gene can be shown by entering the gene name in the corresponding text entry box and clicking on the Go button. The selected gene will be visually highlighted.

The panel of navigation arrows can be used for moving to a nearby region. The panel allows lateral translation to the left or right, and also serves to zoom in or out.

The magnified section indicates the transcription direction of genes by rectangular blocks with an arrow at one end, pointing from the 5′ to the 3′ end. ORFs for actual or inferred proteins have symmetrical arrowheads (with the arrow apex in the center), whereas RNA genes have an asymmetrical arrowhead (with the apex at the top edge). Phantom- and pseudo-genes are crossed out with a big, diagonal X. When a gene wraps across more than one line, a zigzag at the end of the line indicates that the gene continues on the next line. Clicking on a gene brings up the corresponding gene description page.

Gene arrows filled with solid colors have transcription unit (operon) information available. All the adjacent genes that are part of a given operon are assigned the same color. Genes that have not been assigned to any transcription unit are not colored.

Additionally, transcription-units are indicated by a gray background area behind the genes, spanning the entire region of the operon.

Moving the mouse-cursor over the genes reveals their product name and the length in base pairs of the intergenic region between the chosen gene and its neighboring genes to the left and right. If the number of base pairs carries a minus sign, the genes overlap by that many bases. As an example:

```
Gene: xdhB

Product: putative xanthine dehydrogenase subunit, FAD-binding domain

Intergenic distances (bp): xdhA< +11 xdhB -3 >xdhC
```

This means that there are 11 bp to the left of xdhB before xdhA is reached, but to the right, xdhC overlaps with xdhB by 3 bp.

If the overlap between adjacent genes is more than a small amount, the shorter gene is drawn above the longer gene to avoid visual clashes.

When zooming in to a great level of detail, transcription start sites and terminators are drawn. Transcription start sites are indicated by small arrows that point toward the 3′ end of the transcript. Moving the mouse-cursor over the transcription start sites reveals the operon they are part of. The transcription factors controlling the operon are also shown, with a plus sign meaning activation and a minus sign meaning inhibition. Clicking on a transcription start site brings up the corresponding transcription unit description page.

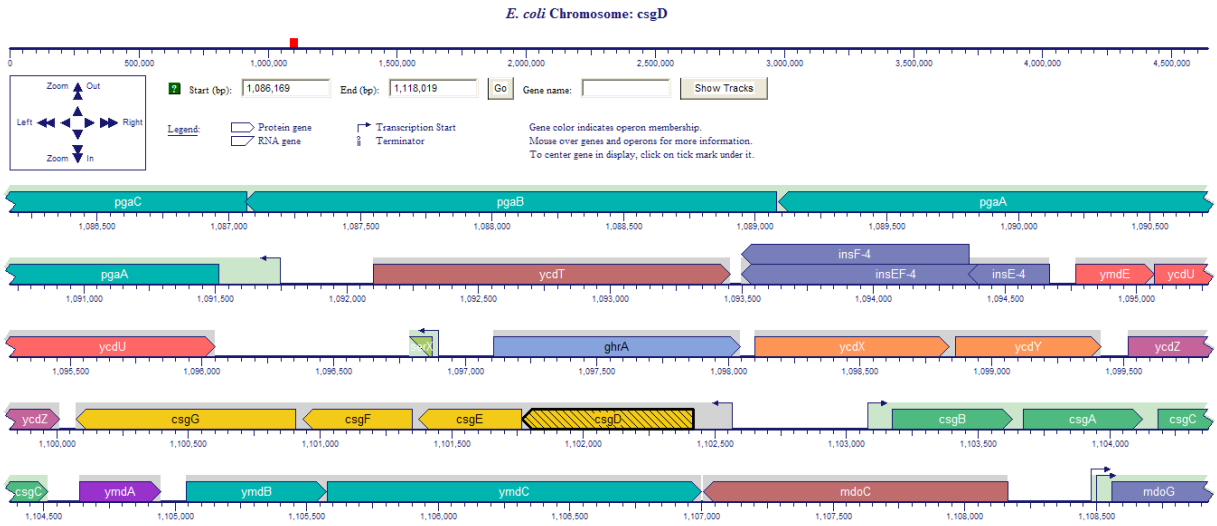


Figure 3.14: Genome browser display

3.8.10.1 Chromosome Menu

Select & Browse Chromosome/Replicon For organisms with multiple chromosomes or plasmids, you can select the current chromosome or plasmid for display. Clicking on a chromosome within an organism-summary page also selects that chromosome. This command has the keyboard shortcut Ctrl-B (standing for "Browse").

Show Sequence of a Segment of Chromosome Retrieves the nucleic acid sequence of a region of the current chromosome, or the reverse-complement of a specified region.

Print Poster Generates a postscript file of a poster-size depiction of the chromosome using the genome browser. That file may be sent to a local printer or to a commercial printing service. The user may supply various data for the poster such as a title, subtitle, and block of explanatory text to be printed on the poster.

Add External Track This allows uploading of a GFF file containing external track information. Please see Section 3.8.11 for more information.

Add or Replace Sequence File Use this command to either add a sequence file (in FASTA format) to a chromosome that previously lacked one, or to replace an existing sequence file with an updated one. Be sure to update the position of genes on the chromosome to reflect the change in the underlying sequence.

3.8.11 Displaying External Tracks on the Genome Browser

External annotations can be shown alongside the display of a chromosome region, in form of additional tracks that are uploaded by the user. The supported file format is GFF, version 2. A short description of this format can be found on the help page, reached by clicking on the green icon containing a question mark, in the genome browser.

The GFF file allows definition of segments on the chromosome that are denoted by a start and stop base-pair position. In an attribute field, a name can be assigned to the segment, and in a score field, a numerical value (such as an expression value) can be supplied. This allows a broad range of different data types to be shown in the genome browser, aligned with the genes and transcription units that a PGDB already describes. This could include alternate gene predictions, or the results of expression experiments. Each specified segment can state a source and feature value, allowing different segment types to be supplied in one file. The external track mode of the genome browser will display different combinations of source/feature values grouped together. If in these groups some of the shown segments overlap due to their base-pair positions, such segments will be displayed on separate lines, to avoid visual clashes.

To view data from such a GFF file in an external track, first go to the menu command **Chromosome** → **Select & Browse Chromosome / Replicon** and open the genome browser. Once the browser is open, click on the "Show Tracks" button to the right of the gene name dialog box. This will enter the external tracks mode, in which the magnified region will no longer wrap to fill the screen, instead making room for external tracks that will be displayed underneath. Vertical hair lines will be shown for easier visual alignment of features in external tracks with the magnified region. You can then select the menu command **Chromosome** → **Add External Track**. A dialog box will appear asking you to enter a URL that points to a GFF file, or to browse local files for a GFF file. Once you have selected a file and pressed OK, the tracks from the GFF file will display on the genome browser, as shown in .

The external tracks display will show the feature name on the left, the sequence name if one is included (the example in does not have sequence names), and the appropriate color to match the feature's score, if a score value was found in the GFF file.

Following the display of a track, you can continue to browse the genome normally, using the standard Left, Right, Zoom Out, and Zoom In controls, and the Gene Name box.

You can also display data from more than one GFF file at the same time. Load each file individually using the procedure described above. Tracks from the first file loaded will appear just below the gene line. Tracks from the second file loaded will appear below those from the first, and so on.

The horizontal bars represent the feature data found in the GFF track file. These are arranged in rows distributed vertically, so as to help prevent overlapping features from

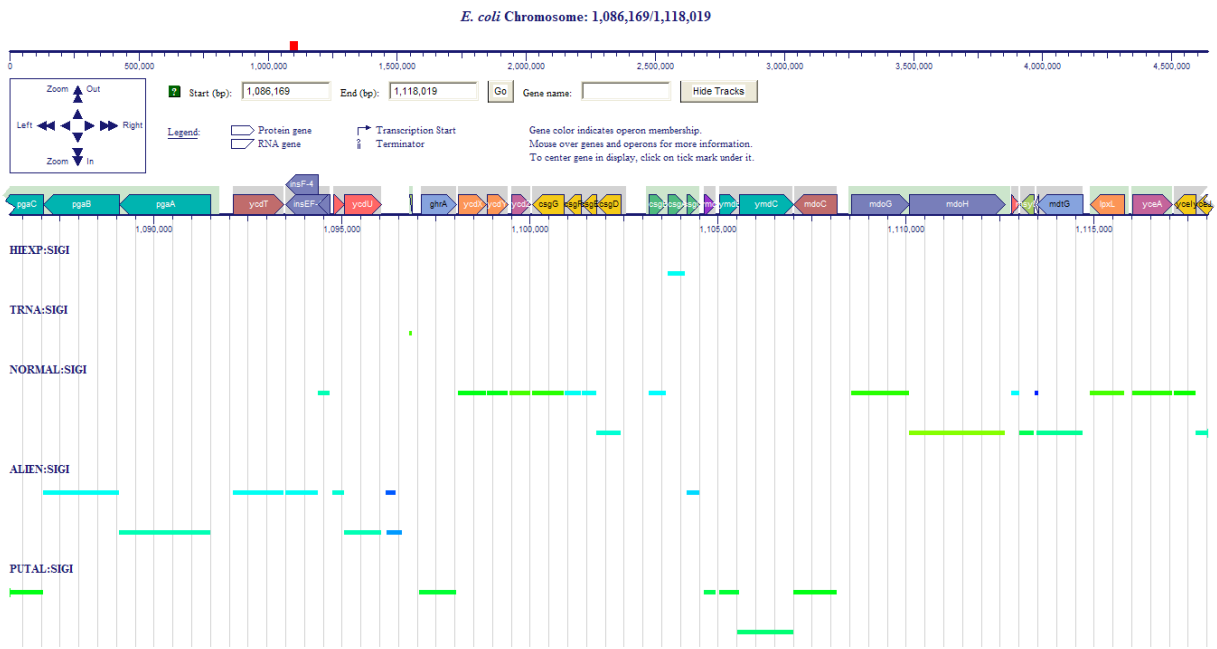


Figure 3.15: External tracks display data in its genomic context

running into each other and being indistinguishable. The number of distributed rows varies with the zoom scale you're viewing the track data at, so that features can fit; there is no other meaning to this. The length of each horizontal bar shows the extent of each individual feature reading. The color is drawn from a spectrum that shows the magnitude of its score.

In order to get a better feel for this magnitude, a graph of the same track feature data is also plotted above the horizontal bars. Here, each feature measurement is represented by a single point at its midpoint. Not only is the magnitude of the score plotted in the same color as in the horizontal bars display, but it's also simultaneously represented as the height on the graph. This offers an intuitive method of viewing trends and anomalies in the data at a glance.

In the Web version, it is possible to choose to display, or turn off the display, of either the horizontal bars or the graph plot or both, for each of multiple tracks viewed simultaneously. Reference a pull-down selector control next to the listing of the track at the bottom of the page, which switches between "Show both graph and horizontal", "Show only graph", "Show only horizontal", and "Both invisible". This allows you to stack graphs from different tracks close to each other, so that you can compare them and see fine differences between them.

It is also possible to shift the plotted range of this graph for each track file viewed. Beside the listing of the track there is also a line saying "graph Y range from [ ] to [ ]" with a Set button. Fill in the desired lower and upper Y coordinates of the range, press the Set

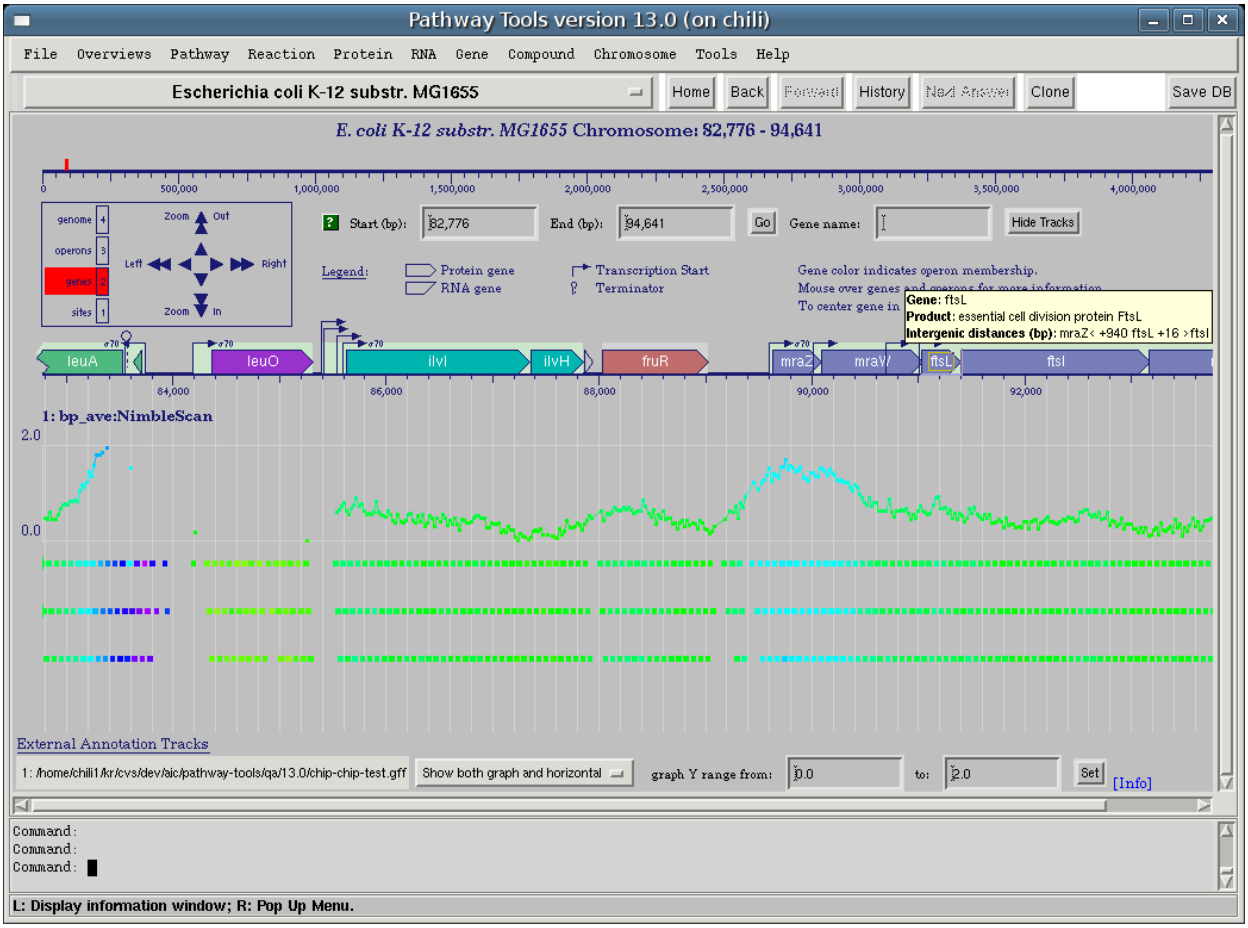


Figure 3.16: External tracks in graphing mode

button, and that particular graph will be redisplayed with that setting. Entries may be in integers or decimals. The lower range must be less than the upper range coordinate. Note that, since the graph can have an arbitrary scale, when the graph points are plotted, any points that lie outside of this range are not displayed at all.

In the Web version, it is also possible for you to upload a local GFF tracks file from your disk into the server. Select the file that you want to upload by hitting the Browse button, which will allow you to explore the files on your local disks. After you have browsed and selected which GFF file you'd like to send to the server, simply hit the Upload button. This button will then gray out and turn into an "Uploading..." button.

Depending upon your server configuration and the size of your GFF file, it can take several minutes to finish uploading. During this time, the page will not respond, and you should not click any more controls. After the file has finished successfully uploading and being parsed, it will let you know by refreshing the page. The new tracks plot will be displayed, and the button reset to Upload.

3.8.12 Comparative Genome Browser

The comparative genome browser can be used to examine several replicons (chromosomes or plasmids) simultaneously side by side. This allows easy visual comparison of related organisms to observe similarities and differences in their gene arrangements. For the alignment to work, Database Links of the relationship type "Ortholog" need to exist among genes of the organisms to be compared. Such Database Links can also be dynamically loaded from a MySQL database, with an additional setup. Implementation details are available upon request.

The comparative genome browser is usually entered from a page describing a gene. In the section that shows the Database Links, there are two buttons that can be clicked on, one called "Align in Multi-Genome Browser" and the other "Select Organisms". When the button invoking the comparative genome browser is clicked for the first time in a session, the user will be asked in a popup menu to select the set of organisms to consider in the alignment. Thereafter, the selected set of organisms is remembered by the Navigator session until the user changes the selection with the corresponding button. The selected organism set also filters the ortholog links shown on the gene page itself. When the Pathway Tools work as a Web server, the organism selection is stored as a cookie called BIOCYC-ORGIDS in the user's Web browser, for the duration of the session.

When the comparative genome browser is invoked from a gene page, that gene and its organism orchestrate the rest of the alignment. In the display, the top-most replicon is the reference, against which the comparisons are made by following the ortholog links for every gene of the top replicon in its visible section. The selected gene that is the focus of the comparison is highlighted on each replicon by a thick outline and a slanted hashed background. These selected genes are lined up at the center position of their lengths.

The navigation in the magnified region can be achieved by the following methods:

- The region around a gene can be shown, by entering the gene name in the corresponding text entry box, followed by clicking on the "Go" button. This substring search will show candidates of the lead replicon with orthology links for positioning the other replicons.
- The gene description page carries a button called "Align in Multi-Genome Browser". If there are ortholog links, these links will bring up a browser centered on the gene in question.
- The panel of navigation arrows can be used for moving to a nearby region. The panel allows lateral translation to the left or right, and also serves to zoom in or out.

Genes with solid colors have links to orthologs. All the corresponding orthologs are assigned the same color, out of a set of a dozen colors that will be reused repeatedly. Genes for which no ortholog links were found in the PGDB are not colored.

The other display features are the same as described for the regular genome browser.

3.9 Advanced Queries using the BioVelo querying language

The BioVelo language can be used to write complex and simple queries of PGDBs. It can replace, to some extent, the underlying Lisp language of Pathway Tools, to search a PGDB according to complex constraints. The complete BioVelo language documentation, as well as two GUI interfaces to write queries, is available on the Web at `http://www.biocyc.com/query.html`.

3.10 Miscellaneous Commands and Tools

Various tools not specific to any type of biological object are available from the File or Tools menu or as buttons in the main window.

3.10.1 Home

The **Home** button takes you to the Organism Summary Display (see "Organism Summary Display" in Section 3.3).

3.10.2 Back

The **Back** button returns you to the object displayed before the one you are currently viewing. You can use the Back button to go many steps backward in this manner, or you may instead use the History button (see "History" in 3.10.11.8).

3.10.3 Forward

If you have used the **Back** or **History** button to view a previously viewed object, you can advance forward to more recently viewed objects one step at a time using the **Forward** button.

3.10.4 History

The history list tracks the objects you have recently displayed. You can move forward or backward in this list one step at a time, or select a specific object from the list (see "History" in 3.10.11.8, for more details).

3.10.5 Next Answer

When a query such as a substring search returns multiple answers, the first answer is displayed and the rest are placed on the Answer List. This button displays the next object on the Answer List.

3.10.6 Clone

It is sometimes useful to capture one or more object displays for future reference, such as to allow information about two different objects to be visually compared (see "Parallel Comparison" in section 3.11.3). The **Clone** button causes a new window to be created as a copy of an existing pane in the main display window (the command prompts you to select the pane to clone, if more than one pane is visible). You can navigate in the cloned window as usual, so that several tracks of inquiry may be pursued at the same time.

3.10.7 Print

Select menu **File** → **Print** to print any Pathway/Genome Navigator pane to a printer, Postscript file or a PDF file.

All functions are implemented for all platforms but some of the functions depend on external applications that do not come bundled with pathway tools due to licence issues.

3.10.7.0.1 Linux and Mac OS Printing using printer depends on `lpr` command being available. This is the default for most Linux machines. Please contact you system administrator if you are unable to print the desired printer. Printing to a PDF file depends on Ghostscript being installed on your machine.

3.10.7.0.2 Windows Printing using printer depends on Ghostscript and GSview applications being installed on your machine, while printing to a PDF file depends only on Ghostscript. Please consult the Instalation manual on how to install these applications.

3.10.8 Tools Menu

The Tools menu provides access to the following commands.

3.10.8.1 PathoLogic

This command invokes the PathoLogic module.

3.10.8.2 Preferences

You can customize the displays produced by the Pathway/Genome Navigator (see 3.10.11 for more information).

3.10.8.3 Pane

The **Clone** command provides one way of looking at displays of more than one object at a time (see 3.10.6). A second approach is to change your preferences so that the Navigator has more than one display pane active at a time (see "Pane Layout" in section 3.10.11.1). When multiple panes are active, the Navigator normally displays the next object in the least recently used pane. However, you can use the **Fix** button to fix a given display pane so that it is not overwritten. **Fix** asks you to click on the window that you want to fix, if more than one display pane is active in the Pathway/Genome Navigator. Similarly, the **Unfix** button prompts you to click in the window that you no longer wish to remain fixed.

3.10.8.4 Instant Patch

Selecting menu item **Tools** → **Instant Patch** → **Download and Activate All Patches** downloads and applies the latest Pathway Tools software patches to both the Pathway Tools installation and the current running Pathway Tools application. There is usually no need to restart the Pathway Tools application to incorporate patches. Select **Activate Installed Patches** if you have installed a patch manually since Pathway Tools was last started and you want to load it into your running session (all locally installed patches are automatically loaded when a new session starts up).

3.10.8.5 History

The history list tracks the objects you have recently displayed. You can move forward or backward in this list one step at a time, or select a specific object from the list (see 3.10.11.8, for more details).

3.10.8.6 Answer List

When a query such as a substring search returns multiple answers, the first answer is displayed and the rest are placed on the Answer List. The **Next** command displays the next object on the Answer List, the **Select** command lets you choose lets you choose one or more objects from the Answer List and the **Show on Console** command displays the Answer List in your original terminal window.

3.10.8.7 Ontology Browser

Invokes the GKB Editor Hierarchy Viewer for examining the taxonomy of classes and instances in the PGDB. From this viewer you can perform various editing operations and can invoke other GKB Editor viewers, such as the relationships viewer.

3.10.8.8 Prepare BLAST Reference Data

If you want users to be able to BLAST a query sequence against your organism's genome (submitted through the Web Pathway Tools query page), or if you want to use the Pathway Hole Filler to search your organism's genome for missing enzymes (see 6.4.8.2.4) you must:

Make sure that the **blastall** and **formatdb** programs are installed correctly (see `http://bioinformatics.ai.sri.com/ptools/installation-guide/released/blast.html`).

Build protein and nucleotide BLAST databases for your organisms using the **Both** command in the **Prepare BLAST Reference Data** menu. The databases can be built separately using the **Nucleotide** or **Protein** commands. The resulting databases are stored in the directory `<name>cyc/version/data/`, where "`<name>`" is the abbreviation of your PGDB.

If your database was built from EST data, a BLAST protein database for your organism can be built from a set of translated EST sequences using the **Protein from ESTs** command in the **Prepare BLAST Reference Data** menu.

Upon selecting this command, you will be prompted to select the appropriate file containing the set of translated ESTs your BLAST protein database will include. This file should be composed of multiple FASTA format protein sequences, one sequence for each EST. For each sequence in the file, the protein frame ID should appear in the FASTA description line. For instance, in the CauloCyc database, the gene CC2913 encodes the protein whose ID is CC2913-MONOMER. The FASTA format sequence for this protein would appear in the file as

```
>CC2913-MONOMER
MTQYRITFEGPVILGAGLAGLTAALSATTGAAKTALVLSPTPLASGCSSAWAQGGMAAAL
SGDDSPALHAADTIAAGAGLCDPQAVDLLTREGPQAVRDLAALGAPFDRKADDGFVLSLE
AAHSAARVARVGGDGAGAAIMAAVIAAVRATPGIEVRENARARRLLQDANGRVVGVLADV
DGALVEIRSTAVILATGGVGGLYAVTTTPAQVRGEGLGLAALAGAMIADPEFVQFHPTAI
DIGRDPAPLATEALRGEGAILRNADGKAFMADYHPAKELAPRDVVARALHAERAAGRGAF
LDATAAVGAHFPHEFPAVFEACMSAGIDPRRQMIPVTPAVHYHMGGVATDLDGRASLPGL
YAAGECASTGVQGANRLASNSLLEAAVFGARAGRAAAAEGATGGPPVSLEPLPDLPDAAL
QGLRKAMSRDAGVIRDADGLTRLLGEIETLEAGHGQGPILVAARLIVTAALAREESRGGH
CRIDFPATDPVGVRTFVTLDGREPGLRYAAE
```

The file will include N FASTA format sequences, where N is the number of ESTs used to build your database.

When the **Select File Containing Protein Sequences** dialog appears, select to the file containing your translated ESTs as described above and click **OK**. The BLAST formatdb program will generate the BLAST protein database. Like the **Both**, **Protein**, and **Nucleotide** menu commands, the resulting BLAST database files are stored in the directory `<name>cyc/version/data/`, where "`<name>`" is the abbreviation of your PGDB.

Note that you can only directly access BLAST to search your organism from a running Pathway Tools Web server. In desktop mode, BLAST is used by the Pathway Hole Filler, but the BLAST program cannot be accessed directly by the user.

3.10.8.9 Browse Downloadable PGDBs

Select this command to download and install optional PGDBs from a list. Some of these databases are provided by SRI, and some are provided by third parties (see Section , if you are interested in making yours publicly available via this feature).

If you select any number of PGDBs to download, you may see a click-through license. You must agree to the terms of this license to gain access to the downloadable PGDBs.

Important Notes:

Commercial users: Your license specifies which PGDBs you may use with Pathway Tools; you may download these PGDBs only if allowed by the license.

Microsoft Windows users: Perform these steps prior to browsing or downloading PGDBs:

Download unxutils from `http://unxutils.sourceforge.net/`, unzip it, move its usr/local/wbin directory into your ptools installation's aic-export directory, and rename "wbin" to "winutils".

If you attempted to use PGDB Sharing before installing unxutils, PGDB sharing may fail, and you may need to remove tar and tar.gz files from your Windows temporary folder, the location of which depends upon which version of Windows you have as well as how Windows is configured. An example temporary folder is

```
c:\Documents and Settings\smith\Local Settings\Temp\
```

Download speed: Downloading may take hours depending on the size of the database, your Internet connection speed, the database provider's FTP server speed, and the database provider's Internet connection speed.

Omitted PGDBs: Two types of shared PGDBs are omitted from your list:

PGDB versions you have already installed

any version of a PGDB that was built into your Pathway Tools

Firewall: The PGDB sharing system uses FTP to transfer files. This requires you to be able to connect to `ftp://ftp.ai.sri.com` on port 21 as well as all ports in the range 1024 to 65535. Your computer need not LISTEN on any port. Consult your network administrator for further assistance.

3.10.8.10 Publish PGDBs

See Chapter 5 for information on using this command.

3.10.8.11 Search

This menu will allow you to search for and browse Curator, Organism, or Organization frames in the database.

3.10.8.12 Upgrade Schema of Current DB

When you upgrade to a new version of Pathway Tools, you must invoke this command for each PGDB that you have created or imported from elsewhere to support any schema or other important changes required by the software.

3.10.8.13 Consistency Checker

The Consistency Checker should be run on PGDBs at regular intervals, such as before public releases of a PGDB. This procedure will find violations of internal consistency constraints and some data and formatting errors. Consistency checks are divided

into two groups: Automatic Tasks and Manual Tasks . Automatic tasks usually do not require any user interaction — repairs are performed automatically by the software. Many times, these tasks will provide warning messages. Not all warnings will need to be fixed — some are provided merely for informational purposes and left up to the user's discretion. Manual tasks require user interaction to fix the reported violations. The user should carefully examine the output from the consistency checker to see what (if any) changes need to be made. The entire Consistency Checker session output is also saved as a text file in the `reports` directory of the organism. For example, if you are running Consistency Checker on CauloCyc, your output will be saved in `Caulocyc/version/reports/Consistency-checker-report-2006-08-21_12-21-17.txt`.

Note that if you run several consistency checker sessions in one day, each output file can be distinguished by the date and time appended to the name of each report filename.

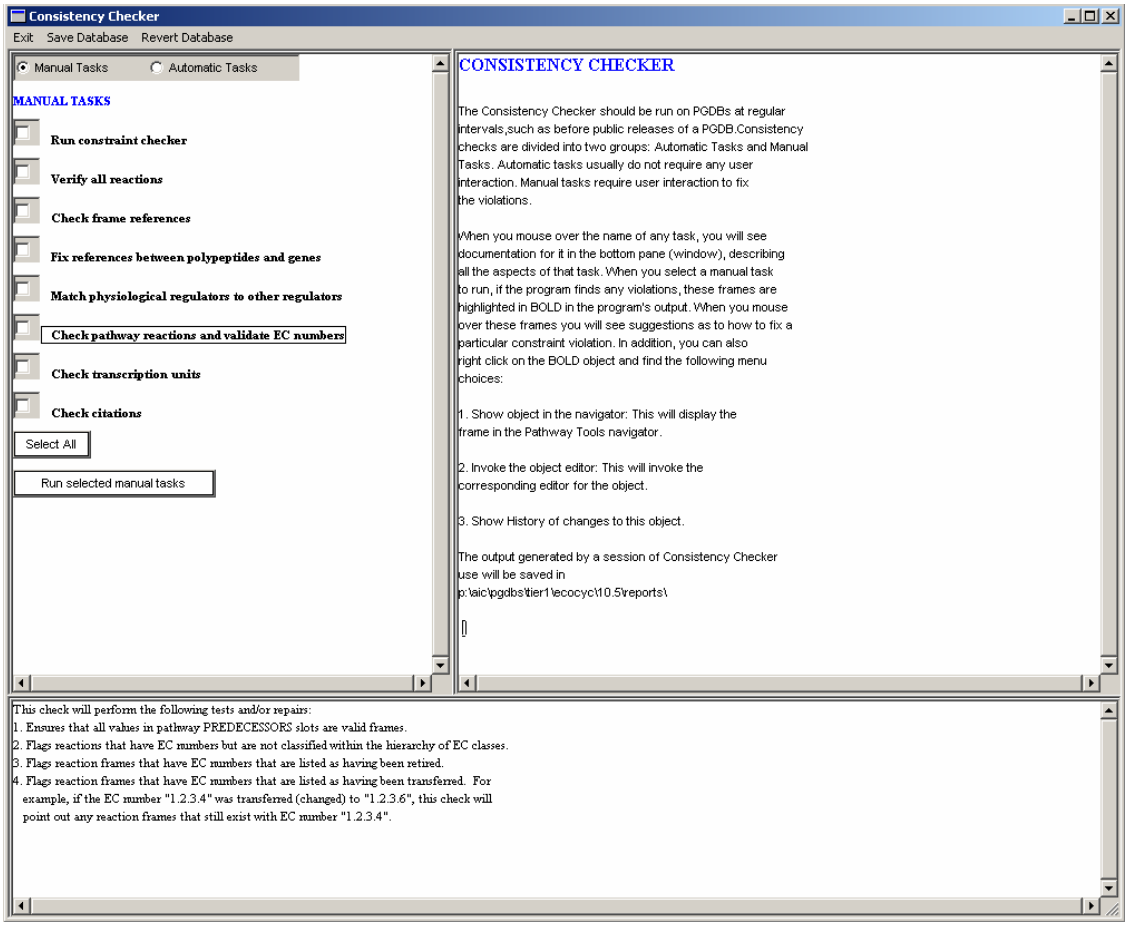


Figure 3.17: Consistency Checker graphical interface

In the Consistency Checker graphical interface (see Figure 3.17) you can select to run either the "Manual Tasks" or "Automatic Tasks" by toggling the radio buttons. When

you mouse over a task's name, you will see documentation for that particular task in the bottom window pane, which describes all aspects of that task. When you select and run a manual task, if the program finds any violations, the PGDB frames whose contents are problematic are displayed in boldface in the program's output. When you mouse over these frames, you will see suggestions as to how to fix a particular violation. In addition, for both automatic and manual tasks, you can right-click on any boldface object and find the following menu choices:

Show object in the navigator: This will display the object in the Pathway Tools navigator.

Invoke the object editor: This will invoke the corresponding editor for the object.

Show History: This will show the history of changes that were made to this object.

When you have finished going through the tasks, you must save the changes by clicking the button "**Save Database**" in order to preserve your changes. If you do not want to keep the changes that were made to the database, you can use the button "**Revert Database**" to revert all your changes and bring the database back to its original form that you started out with.

3.10.9 Help

The **Help** menu contains a small number of help topics for the Pathway/Genome Navigator.

3.10.10 Exiting Pathway Tools

Selecting menu **File** → **Exit** terminates the Pathway/Genome Navigator. If the Navigator is the only part of Pathway Tools you were running, then selecting this menu item also exits Pathway Tools.

3.10.11 User Preferences

You can customize the displays produced by the Pathway/Genome Navigator. Commands to change user preferences are found in the **Tools** → **Preferences** menu. If you change preferences during a session, then before you exit the session you will be asked if the changes should be saved. Preferences are saved in the file `.ecocyc-prefs` in your home directory. That file is loaded when the Pathway/Genome Navigator starts, so the program is automatically configured to your own preferences.

The options within the preferences menu are as follows.

3.10.11.1 Pane Layout

One to four display panes can be present simultaneously. These panes are arranged on the screen in tiled fashion, so the size of each pane depends on the total number of panes present. You can choose the number of panes from the **Pane Layout** menu. Note that some object types look better displayed on panes of certain sizes. For example, the complex graphical displays of pathways and genetic maps generally look better when only one or two display panes are present, so that the display can cover the entire screen width. Simple or primarily textual displays such as for genes or compounds (especially those with simple structures) do not suffer from being displayed on a smaller pane, and it might be advantageous to be able to display several objects at a time.

3.10.11.2 Color

Several color palette, which assign specific colors to the window background and each type of object, have been predefined and are named in the **Colors** menu. Because different monitors show colors differently, you are encouraged to try out several of these color combinations until you find one you like. Note that for monochrome monitors, only two color palette are available: **Black on White** and **White on Black**. On color monitors, palette are available with black, white, gray, or blue backgrounds.

3.10.11.3 Text Font Size

To select a font size for the Navigator display panes, go to the **Tools** → **Preferences** → **Text Font Size** menu and click on the preferred size. Text embedded in graphics is controlled not here but rather in the object displays listed below.

3.10.11.4 Citation Reference Style

You can use this preference to choose whether references in page displays show up in numeric form (e.g. [1]), in short-hand mnemonic form (e.g. [Smith95]), or in full or abbreviated APA style (e.g. [Smith & Jones, 1995]).

3.10.11.5 Overview Display

You can apply a scale factor to alter the size of the Cellular Overview. Enter an integer that will be treated as a percentage — a value of 100 is the default scale factor.

3.10.11.6 Pathway Display

This menu determines what elements are included in pathway diagrams, and how those elements are drawn. The options **None**, **Most**, and **All** for structure-drawing preferences are the same as for reaction displays, although for pathways the default is not to show structures. Most of the preference options should be self-explanatory.

The preferences available for pathway displays are

Show structures for main compounds Main compounds are those that run along the main backbone of the pathway.

Show names when structures are shown If *No* and compound structures are drawn, the compound names will be omitted.

Show side compounds Side compounds are those that do not run along the main backbone of the pathway. They can be omitted from the display completely, if only a general overview of the pathway is desired.

Show side structures Whether or not structures are drawn for side compounds.

Show enzyme names Like side compounds, enzyme names can be either omitted or included in the pathway diagram.

Font size for mains Five logical font sizes are available. Smaller font sizes mean that pathways can be displayed more compactly, whereas larger font sizes may be easier to read (especially, for example, when the display is to be converted to a slide or transparency).

Font size for sides and enzymes Side compounds and enzymes can be displayed using either the same-size font as for main compounds, or with font that is one size smaller.

Reaction arrow emphasis Two options are available for reaction arrows. The **reversibility** option draws double arrows to indicate that a particular reaction is reversible (a reaction is assumed to be reversible unless it is known to be irreversible). The **pathway-flow-direction** option draws single arrows for each reaction, in a direction to indicate the typical flow of the pathway. The former option provides more information, but the latter option may be clearer to read, particularly in complex branching pathways.

Layout for linear pathways Linear pathways are laid out in snake fashion by default, to enable as much of the pathway as possible to fit on the viewport. The other options are to draw the pathways in a single horizontal or vertical line. Displays of branched or cyclic pathways cannot be customized in this fashion.

Show pathway graph only, without title or text When this option is selected, only the pathway graph itself is displayed. This might be useful, for example, when preparing figures for publication or slides.

3.10.11.7 Reaction Display

Preferences in this menu control how the reaction participants are drawn within the reaction display. By default, most compounds are drawn with structures. However, there is a set of compounds for which we choose not to show structures. These are typically common cofactors such as ATP and NADH, whose structures, if drawn, are likely to distract the user from the principal transformation occurring. You can change this default to show either all or no structures. Be aware that structures for some compounds are not currently available in MetaCyc and other PGDBs.

You can add to and remove from the set of compounds for which structure drawing is suppressed under the **Most** option by right-clicking on the compound in question. A menu of commands appears if the drawing of a compound structure has not been suppressed. The menu contains the item **Show name only for this compound**, which turns off structure drawing for the compound on this and future reaction displays. If structure drawing is currently suppressed for the compound, the menu item reads **Show structure for this compound**.

The preferences available for reaction displays are

Show structures This determines whether structures are drawn for compounds in the reaction equation. The options are **Most**, **None,** and **All** and are described above.

Display reaction direction Reactions can be drawn either in the direction specified by the Enzyme Nomenclature Commission (option **enzyme nomenclature**, the default), or in the direction in which the reaction appears in metabolic pathways (option **direction in pathway(s)**). If a particular reaction appears in different directions in different pathways, then the reaction is always displayed in the Enzyme Nomenclature direction.

3.10.11.8 History and Answer Lists

This dialog box allows you to alter the length of the history list. A smaller history list means that fewer items are stored, but the list is faster to cycle through. To change the length of the history list, click on the number currently displayed. It disappears, and you can enter a new number. If you change your mind and do not want to enter a new number, simply press return, and the previous value will reappear.

Another part of the History/Answer List dialog box allows you to control what happens when the **Next Answer** command is invoked. By default, each time **Next Answer** is invoked, new objects are displayed in all the unfixed display panes in the main window. However, you can change the default so that only one new object at a time is displayed.

3.10.11.9 Database Sharing

The Database Sharing preferences allow you to set the parameters that are used by Pathway Tools for connecting to our Pathway / Genome Database Registry. Please see Section 5.2.2 for more information.

3.10.11.10 General

The General Preferences option allows you to set the Author frame ID which will be credited with your changes for certain entities in the PGDB, such as proteins and pathways. It will default to the user name from the operating system. Please see Section 7.6.2.2 for more information.

3.10.11.11 Reverting and Saving User Preferences

Select **Tools** → **Preferences** → **Restore Saved Preferences** to revert to the set of preferences previously stored in your `.ecocyc-prefs` file. Unless you have saved a new set of preferences during the current session, these were the preferences in effect when you started your session.

Select **Tools** → **Preferences** → **Save** to save the current set of preferences to your `.ecocyc-prefs` file. These preferences will be loaded in your next session. Note that if you change preferences but do not save them, or if you change them again after saving them, then upon exiting you are automatically asked if you want to save the new preferences.

Select **Tools** → **Preferences** → **Restore Defaults** to revert to the "factory settings" for the preferences.

3.10.12 Keyboard Shortcuts

The following table summarizes some keyboard shortcuts that are available to allow you to move around the Navigator using just keystrokes. Note that 'Ctrl' is short for the Control key, and that all the keys in the Keystrokes column should be held down simultaneously.

Table 3.3: Table Keyboard shortcuts

Command Shortcut	**Keystrokes**
Chromosome→ Select & Browse Chromosome/Replicon	Ctrl-b
Compound→ Search by Substring	Ctrl-Shift-c
Tools→ Instant Patch→ Download and Activate All Patches	Ctrl-d
Protein→ Search by Substring	Ctrl-Shift-e
Overview→ Show Genome Overview	Ctrl-g
RNA→ Search by Substring	Ctrl-Shift-n
File→ Print	Ctrl-p
Pathway→ Search by Substring	Ctrl-Shift-p
File→ Exit	Ctrl-q
Overview→ Show Complete Regulatory Overview	Ctrl-r
Reaction→ Search by Substring	Ctrl-Shift-r
File→ Save Current PGDB	Ctrl-s
Tools→ PathoLogic	Ctrl-t
Tools→ Preferences→ Layout of Window Panes→ 1 pane	Ctrl-1 (that is a numeral one, not lower-case letter L
Tools→ Preferences→ Layout of Window Panes→ 2 panes	Ctrl-2
Tools→ Preferences→ Layout of Window Panes→ 3 panes	Ctrl-3
Tools→ Preferences→ Layout of Window Panes→ 4 panes	Ctrl-4
Next Answer	Ctrl-(right arrow key)
Go back in history list	(left arrow key)
Go forward in history list	(right arrow key)

3.10.13 Tips

Pathway Tools includes a "tip" system which is designed to help users become more familiar with many of the features of the software. Periodically, Pathway Tools will display a dialog containing useful information about some aspect of its use. Tips are normally displayed at most once per day. In desktop operation, tips are displayed at startup; in web operation, tips are displayed upon a user's first visit to the site. In some cases, Pathway Tools also displays tips that are relevant to a user's activity. For example, when the user invokes PathoLogic, an object editor, or the genome browser, Pathway Tools may display a tip relevant to that part of the software.

See Section 8.7.1.2 for further details on the behavior of tips in web mode.

3.11 Comparative Operations

Information is stored in databases using a consistent format compatible with the Pathway/Genome Navigator. Consequently, the Pathway/Genome Navigator can be used not only to navigate the information contained within a given database but to also compare the information contained in two or more databases; that is, the Navigator supports comparative analyses of the pathways and genomes of two or more databases.

This section is relevant to you only if you currently have access to more than one database. For the purposes of demonstrating some of the available comparative operations, we provide some examples centered around a number of PGDBs, some of which may not be included in your distribution of the Pathway Tools and databases. If this is the case, simply replace one or more of the named databases with one or more of your own databases (e.g., proprietary databases developed using PathoLogic).

Two distinct types of comparative analysis are supported:

- Global comparisons of the metabolic networks of a user-defined set of databases
- Comparisons of specific instances of biological objects across two or more databases

Global analyses are available from within the Overview Mode of the Current Organism with the results painted onto the Overview diagram for this organism. Comparisons of specific instances of biological objects across databases are available from within each of the seven command modes.

3.11.1 Global Comparative Analyses

The metabolic overview of the current organism can be used to highlight all reactions of the Overview shared or not shared with any or all members of a user-specified group of organism databases. This highlighting allows you to compare the metabolic overview of the current organism against those of one or more other organisms. For this reason, these comparisons are referred to as *global comparative analyses*. For example, if your interest lies in developing antimicrobial drugs, these kinds of analyses provide a convenient means of computationally predicting the spectrum (i.e., across the organisms for which databases are provided) of an antimicrobial agent designed to target a specific metabolic enzyme/reaction(s).

To perform such an analysis:

1. Select an organism as the current organism and invoke **Overview** → **Show Overview**.

2. Remove any past highlighting using the **Overview** → **Highlight** → **Clear All** command.

3. Invoke Overview → Highlight → Species Comparison.

4. Specify the nature of the comparison using the **Shared/Not-Shared** and **Any/All** options.

5. Select a set of organisms for comparison.

To select the organism to be compared against other selected organisms (the reference organism), make it the current organism. After you display the overview and invoke the **Overview** → **Highlight** → **Species Comparison** command, the **Species Comparison Dialog** box (see Figure 3.18) will appear.

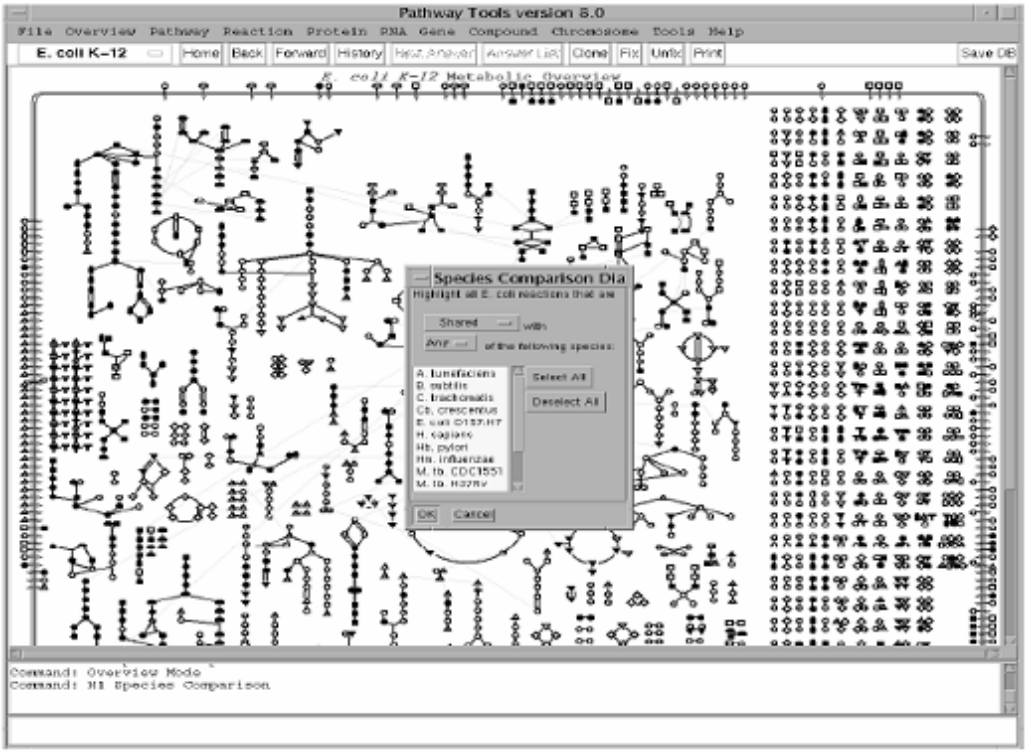


Figure 3.18: Global comparative analysis

This dialog box is used to select both the nature of the comparison and the organisms to be compared against the Current Organism. Specify the type of comparison to be performed by selecting the **Shared** or **Not-Shared** option. By default the **Shared** function will be used. This option defines the nature of reactions highlighted, that is, those shared among or not shared among the Current Organism and any (one or more) or all of the

members of a user-defined set of organisms. Next, specify how the comparison is to be performed by selecting the **Any** or **All** option. Selecting **Any** means that reactions shared between the reference organism and *any one or more* or *all* of the organisms in the user-specified list will be highlighted. Selecting **All** imposes a stricter criterion; only those reactions shared between the reference organism and all listed organisms are highlighted. The final step is to define the set of databases to be used for comparison. Go to the box within the **Species Comparison Dialog Box** that lists the available databases (except the reference organism). Select one or more of these databases by left-clicking on the **Select All** button. Alternatively, you can select a subset of this list. To create such a subset, select each desired database by left-clicking on its name, which will then become shaded. To de-select a database, left-click once on its shaded name (it will become unshaded). To de-select all selected organisms, left-click on the **Deselect All** button. Once a set of one or more organisms has been selected, left-click on the **OK** button.

Depending on the number of databases selected, the exact nature of the comparison, and the type of machine on which you are running, the comparison may take as long as several minutes. The clock face icon indicates that the analysis is under way. Once the analysis is finished, the results are painted onto the metabolic overview of the reference organism. All reactions that satisfy the specified conditions are highlighted in a given color, and the number of highlighted reactions is listed in the Lisp command pane.

Additional species comparisons can be performed with the results superimposed upon the existing highlighting using a different color. Any number of comparisons can be superimposed. Specific colors are used for successive comparisons. When a Gray/Black Background is selected in the User Preferences (see "User Preferences" in 3.10.11) the colors yellow, green, cyan, orange, magenta, Navaho-white, red, purple, spring-green, and deep-pink are used in succession. Alternatively, with a White Background selected in the User Preferences (see Section) the colors green, orange, magenta, turquoise, hot-pink, blue-violet, light-sky-blue, red, and fire-brick are used. Once all colors have been used, the Navigator recycles through the given list of colors in the order stated above. Any overlap between sets of highlighted reactions is shown in white. To identify the specific combination of overlapped colors that resulted in a given reaction being highlighted white, move the mouse cursor over the reaction; the stoichiometric equation for this reaction and the name(s) of overview pathway(s) in which it occurs will appear in the bottom pane. A set of colored bars appears to the left of the reaction equation. These denote the overlapped highlight colors for this reaction. Invoking the **Overview** → **Show Key** command brings up the general key for the Metabolic Overview. At the bottom of this is a specific key for the highlight colors painted onto the Overview. For each color, the nature of the corresponding analysis is summarized.

The key is updated dynamically with successive highlights. However, if you undo the previous highlight (see below) the corresponding key entry is removed and will not be recovered should you choose to Redo the last highlight (see next paragraph). The **Overview** → **Highlight** → **Undo** command removes the highlighting introduced by the last species

comparison performed. It can be used consecutively to remove superimposed highlighting colors. The **Overview** → **Highlight** → **Redo** command re-highlights only the last set of reactions unhighlighted. However, as noted above, the color key is not updated to reflect this. The **Overview** → **Highlight** → **Clear All** command removes existing highlighting.

You can unhighlight the results of a previous comparison before your perform another one. Alternatively, you can clear the overview of all previous highlightings by using the **Clear All** command. Use multiple highlighting to perform complex comparative analyses such as those relevant to designing a desired spectrum for an antimicrobial drug. For example, you could identify all reactions that are shared by all members of a target set of organisms (e.g., those that commonly cause a specific infectious disease) but that are not predicted to occur in another set of organisms (e.g., members of normal gut flora).

3.11.2 Sequential Comparison

When browsing through a PGDB, you might find that more detailed information about a given biological object may be available from EcoCyc than from other PGDBs. This occurs because the information content of the other PGDBs is largely computationally derived, while that for *E. coli* is experimentally derived. It is therefore sometimes informative to navigate from an object display to the comparable EcoCyc display. It should not be assumed *a priori* that objects with identical names in the computationally derived and EcoCyc's are functionally identical since, for example, "equivalent" enzymes may differ across organisms in terms of their regulation and subunit composition. However, unless reliable experimental data is available about the object in the computationally derived database (consult the primary literature to find such data), the *E. coli* homolog can serve as a working "model" for the computationally derived object (since it will generally contain more information).

3.11.2.1 Current Organism-Dependent Comparisons

One method that may be used to navigate from a computationally derived database object display to that for the E. coli homolog involves changing the current organism from one of the computationally derived databases to E. coli and then using the command mode query facilities to find the relevant homolog (assuming one exists). When you search, you can find homologs by using substrings of enzyme and pathway names. Be cautious when using this approach to find E. coli gene homologs because, in general, a consistent vocabulary has not been used for naming genes across organisms. The primary disadvantage of this approach is that it may require numerous mouse operations and by the time you navigate to the relevant *E. coli* display, you could have forgotten pertinent details of the other display. This can be ameliorated by either printing the individual window displays

or making notes about each display for subsequent comparison. Alternatively, as outlined below, you can set display command options so that relevant object displays can be placed side by side for direct visual comparison.

3.11.2.2 Current Organism-Independent Comparisons

A second method for navigating from a computationally derived database object display to that for an *E. coli* homolog requires the use of pop-up windows. This option is available from compound, reaction, protein, and gene display windows. Specifically, when viewing a display for the given object within a specific organism, the comparable object, if present in another organism, can be accessed by first moving the mouse cursor over the object name in the display title (a rectangle should appear around it) and right-clicking the mouse button. This brings up a pop-up menu from which the **Show in Organism** option can be selected to display a list of available organism databases. Selecting a given database results in the display window being replaced with a comparable display for the same object in the selected database (if the object does not exist in the selected database, then the following error message appears in the bottom pane (the listener pane): **"Object does not exist in selected organism"**). By alternately using the **Back** and **Forward** commands, you can compare the entries for the given object in the two databases. To view the same entry in a third database, go to the display window, repeat the above procedures, and select the name of a third database. Now use the **Back** and **Forward** commands to scroll between three distinct displays of the same object across the three selected databases. The above operations can be continued until displays of the given object in any number of available organisms have been accessed. Note that none of these operations changes the Current Organism. You should be cognizant of the nature of the Current Organism since all queries issued through the command menu operate upon the corresponding database.

Alternatively, from a given display window (e.g., Overview and Pathway Displays), you can directly navigate from a component of the display (e.g., a reaction or compound) to a display of that component as present in another user-specified database. For example, set the current organism to *Mycobacterium tuberculosis.* Using the **Pathway** → **Search by Substring** command, bring up a display of the glycolytic pathway in the main display window, and then right-click on any compound in the pathway to bring up a menu of options. Select the **Show in Organism** option by either left- or right-clicking on it to bring up the **Choose species** menu, which lists available databases. Left-click on the name of the organism of interest, for example: *E. coli*. This brings up a display window for the compound in the specified database (assuming such an object exists). If this compound information cannot be found, then an error message to that effect appears in the bottom pane (listener pane): **"Object does not exist in selected organism KB"**. By using a comparable set of commands, you can also select a reaction, enzyme, or gene from a pathway display and navigate directly to a display of that object in a specified database (if present).

A comparable set of operations can be used on Overview displays. For example, select *Mycobacterium tuberculosis* as the current organism and use **Overview** → **Show Overview** to bring up the Overview Display for this organism. Move the mouse cursor over a compound in the overview display. A left-click will take you to the display window for this compound. Alternatively, a right-click will bring up a menu the last two options of which are relevant to species comparison:

- Display compound information for selected species.
- Display pathway information for selected species.

Selecting either one of these options causes the **Choose species** menu to appear, listing names of available databases. Selecting a database brings up the compound/pathway display for the selected database. This operation does not change the nature of the current organism. Use the **Back** command to return to the metabolic overview for the current organism.

Alternatively, the mouse cursor may be moved over a reaction in the database for the metabolic overview. A left-click on the reaction will take you to the display window for this reaction in the current organism. Alternatively, a right-click on the reaction in the overview will bring up a menu, the last two options of which are relevant to species comparison:

- Display reaction information for selected species.
- Display pathway information for selected species.

Selection of either of these options brings up the **Choose species** menu, which lists available databases. Selecting a given database brings up the relevant reaction/pathway display. Use the **Back** command to return to the metabolic overview.

3.11.3 Parallel Comparison

The operations described above present sequential displays of specific instances of particular objects, for example, a particular pathway, as described in two or more distinct databases. For pathways and reactions, however, a table can be generated that combines information about a reaction or pathway across a user-specified set of organisms. To access this display, press the **Cross-Species Comparison** button at the top of any reaction or pathway display page. An example of a pathway comparison is shown in Figure 3.19.

Additionally, user preferences can be set to support concurrent display of multiple display windows such that you can view equivalent instances of an object across two or more organisms; that is, each window provides a display for each organism.

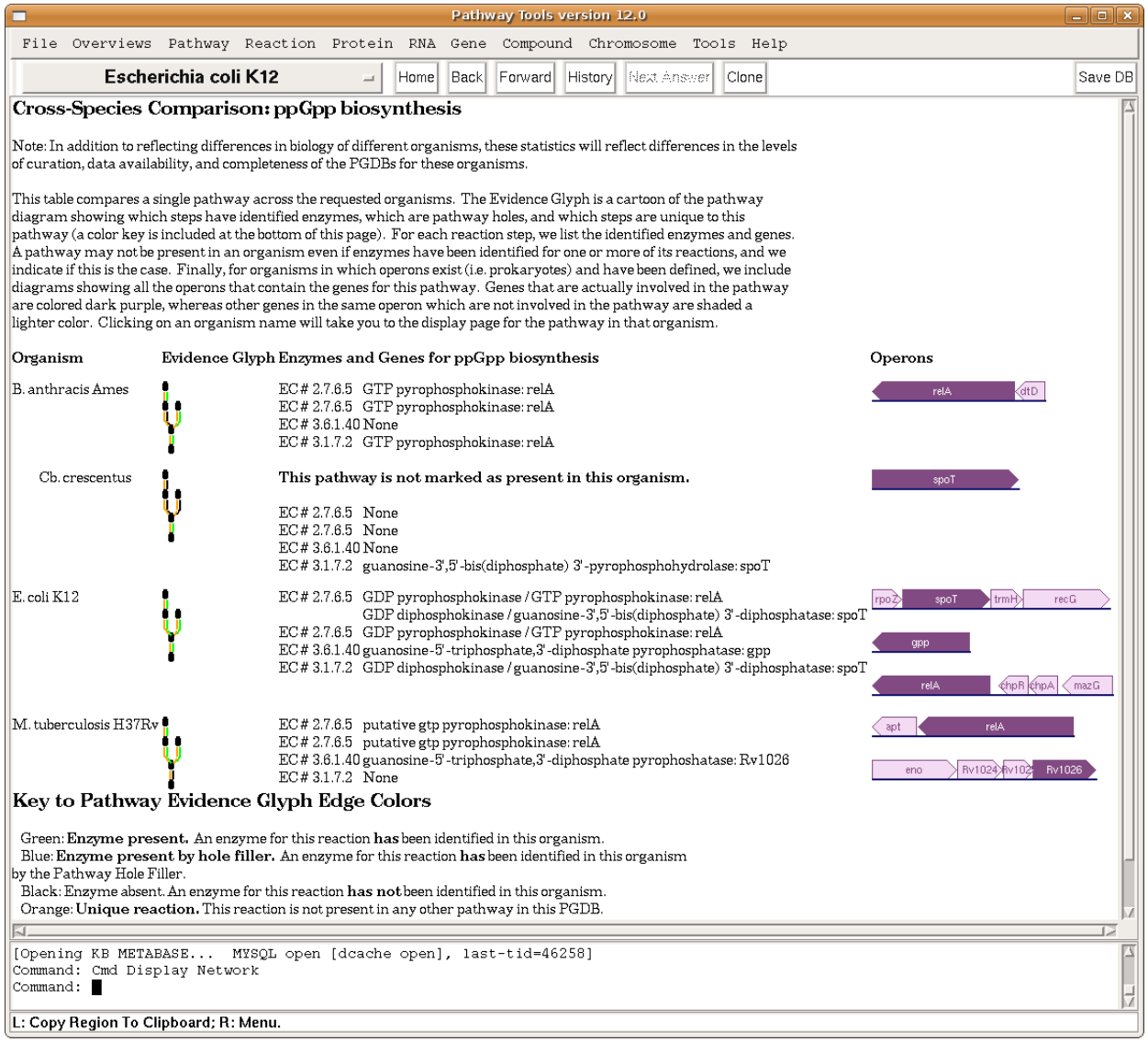


Figure 3.19: A cross-species comparison of the ppGpp biosynthesis pathway across four different organisms.

For example, to compare the alanine biosynthesis pathways of *E. coli* K-12 and *Hm. influenzae*, select **Tools** → **Preferences** → **Pane Layout** → **2 panes**. Set the Current Organism to *E. coli* K-12 and then invoke **Pathway** → **Search by Name or Frame ID** . This brings up the Dialog Box into which you should type **alanine biosynthesis-I**. A display for this pathway appears in the top window. Click on the **Fix** command button and then click on the top window. Move the cursor over the words "alanine biosynthesis-I" as they appear in the display title and right-click the mouse button. This brings up a pop-up menu from which you should select the **Show in Organism** option to bring up a list of available databases. Selecting one of these — for example, *H.influenzae* — results in

a display of the alanine biosynthesis pathway for the selected database in the main display of the bottom window. You can now visually compare the pathway across the two organisms (see Figure 3.20).

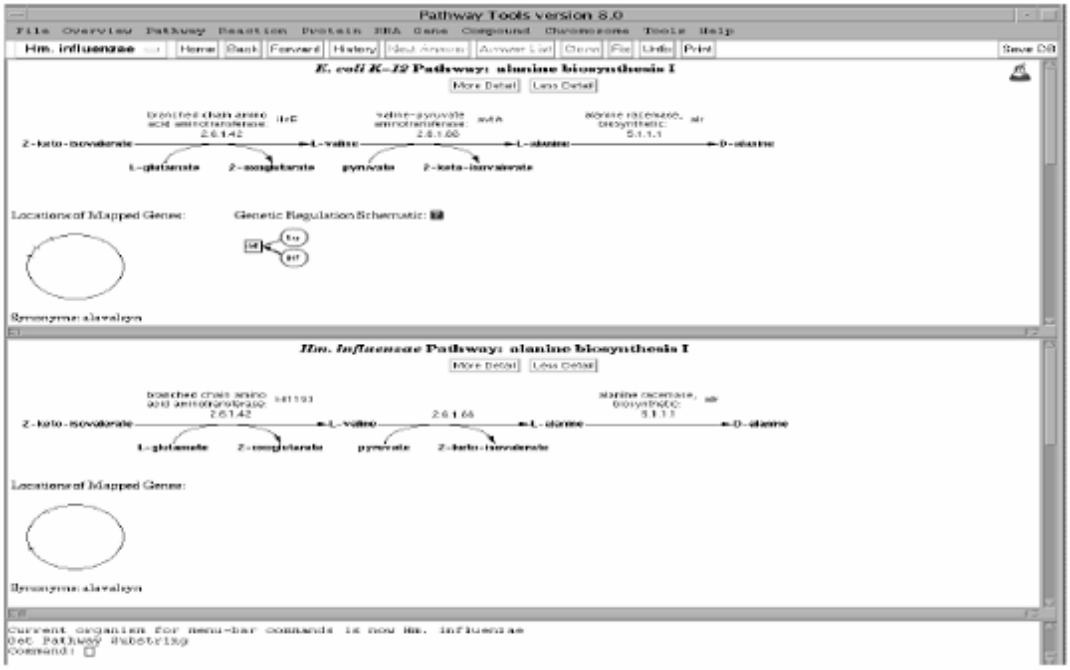


Figure 3.20: Global comparative analysis

By using a comparable set of commands, you can also compare other objects. For example, Figure 3.21 shows a comparison of the genomic maps of *E. coli* and *H.influenzae*.

The user preferences support incorporation of as many as four distinct windows in the display layout. By consecutive application of the **Fix** command, you can visually compare four instances of a given class. Note, however, that two or more metabolic overviews cannot be displayed using this approach.

To compare two metabolic-overview diagrams, use the **Clone** command (see 3.10.6). You can also use this option to compare instances of other classes, for example, pathways. Use the **Clone** command to duplicate any window display; you can clone any number of window displays. Cloned windows do not support the full range of menu options in the full Navigator window, but they do support history commands and clicking on objects within the window, such as for hypertext navigation or for editing. However, you do not have access to any of the command menu options. The history for the clone window is cleared at the time of cloning, but as you navigate between related displays, a local history is constructed for this window. Move back and forth between these displays by using the move back and forth in history commands available within the clone window.

To use the **Clone** command to compare two metabolic overviews, set the Current Organ-

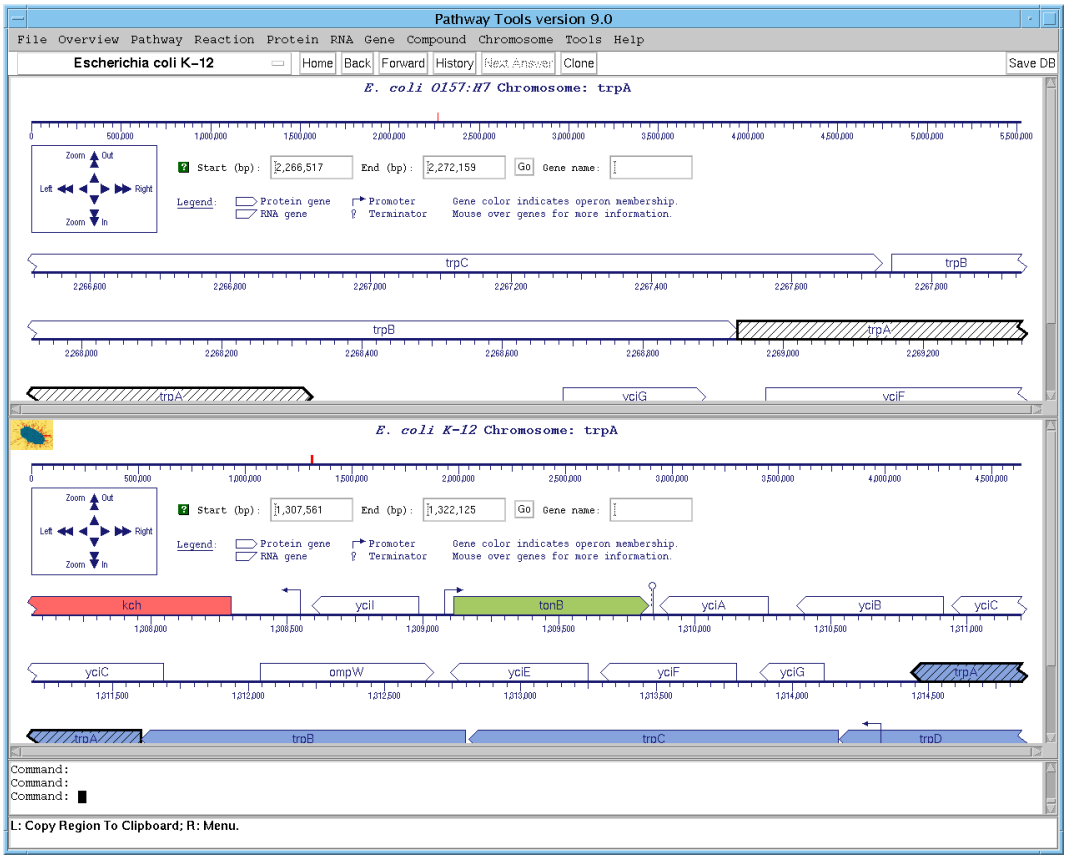


Figure 3.21: Global comparative analysis

ism to an organism of interest (e.g., *Hb. pylori*) and invoke **Overview** → **Show Overview**. The overview for this organism appears in the main display window. Click on the **Clone** button to bring up the overview display window (see Figure 3.22).

Change the current organism to that of the organism to be compared and again invoke **Overview** → **Show Overview**. The metabolic overview for the second organism appears in the main display window. Now position the clone and main display windows so that the respective overviews can be visually compared.

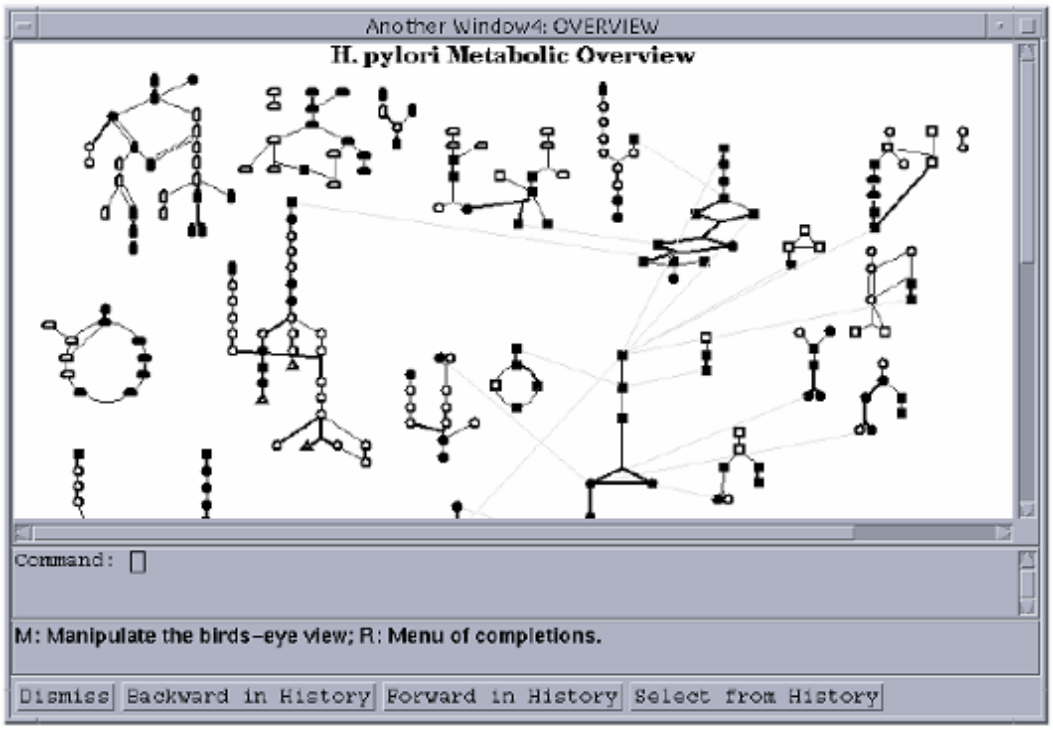


Figure 3.22: A clone window showing the Hb. pylori metabolic overview

Chapter 4

The Import/Export Facility

Although the native data storage format for PGDBs is the Ocelot Frame Representation System, there are several reasons for exchanging some or all of the data in other formats. Users might want to export Pathway Tools data to other applications. They might want to import data from other sources into PGDBs. A user might also want to export data from a PGDB, modify the data using another program (such as editing the data in a spreadsheet), and then import the data back into Pathway Tools.

This chapter surveys Pathway Tools functionality for importing and exporting data from a number of different formats. In some cases Pathway Tools contains an import tool only, in other cases it contains an export tool only (example: SBML), and in some cases it can both import and export the same format (example: column-delimited formats).

Note that none of these file formats support all of the information present in a PGDB, so some information will be lost when exporting to any of these formats.

Pathway Tools supports import and/or export in the following formats:

1. Column-delimited formats that are easy to manipulate with external spreadsheet programs.
2. Attribute-value formats that are easy to parse with external text processing tools.
3. BioPAX, which is an OWL RDF/XML-based format for exchange of pathway data. See `http://www.biopax.org/`.
4. SBML, which is an XML-based format for capturing models of biochemical reaction networks. See `http://www.sbml.org/`. Only SBML export is supported.
5. Genbank format, a standard format for exchange of gene annotations for an entire chromosome. Export is supported. Import of Genbank files when building a PGDB is supported by PathoLogic.

6. A Pathway Tools file format that allows pathways and other PGDB data to be easily exchanged between PGDBs. Import and export of this format are both supported.

7. The widely used MDL Molfile format can be used to exchange compound structures.

8. Citations can be imported via the Internet from PubMed.

Multiple files can be written through one command — see Section 4.5.

4.1 Pathway Import/Export

The pathway import/export facility allows you to export selected pathways and related objects from one PGDB to another PGDB within your Pathway Tools installation. It is also possible to export selected pathways to a file, which can then be imported into another PGDB, possibly at another site. The following scenarios illustrate situations for which this facility is useful:

1. You have created new pathways for your organism PGDB, and we would like to submit those pathways to the MetaCyc database. You would export your pathways to a file and email the file to SRI (see hrefhttp://metacyc.org/MetaCycPosting.shtmlhttp://metacyc.org/MetaCycPosting.shtml for more information).

2. You want to exchange pathways you have created with another user who has been developing a PGDB for a related organism.

3. You want to import into your PGDB individual pathways that have appeared in a new release of MetaCyc, but you do not want to run the general pathway rescoring procedure.

To export the current pathway that you are viewing to another PGDB within Pathway Tools, right-click on the Pathway handle and choose **Edit** → **Export Pathway to DB...**. This will cause a pop-up window to appear with a selection of the available PGDBs to send the current pathway.

To export the current pathway to a PGDB external to your current Pathway Tools session, you will need to export the current pathway (or a set of pathways) to a file first. To select a pathway for export, right-click on the Pathway handle and choose **Edit** → **Add Pathway to File Export List**. All pathways that you want to export to a single file should be selected in this fashion. When this is complete, select **File** → **Export** → **Selected Pathways to File...**. You will have an opportunity to edit the list of pathways to be exported and

specify the file name. In addition to the pathway frames themselves, related objects such as reactions, compounds, and publications are exported. Enzymes and genes can also be exported. The dialog lists some situations in which you may or may not want to include enzymes and genes in your export file.

To import from a file created using the above commands, select **File** → **Import** → **Pathways from File...** and supply the file name. Any frames in the export file that do not exist in the current PGDB are created. Frames that already exist in the current PGDB are generally not overwritten or modified, even if they are different in the export file (thus, this facility is not useful for exchanging updates to existing frames between PGDBs). Check any imported pathways to make sure that they look correct.

4.2 SBML Export

The **File** → **Export** → **Selected Reactions to SBML File...** command brings up a dialog panel that allows selection of a set of reactions from the current PGDB to be written to an SBML file. SBML is an XML-based format for capturing models of biochemical reaction networks. See `http://www.sbml.org/`

4.3 Genbank Format Export

The **File** → **Export** → **Selected Chromosome to Genbank File...** command allows exporting all the gene annotations of a chromosome to a Genbank file. If the PGDB has more than one chromosome/replicon, a small popup menu allows selection of the chromosome to export. Thereafter, a dialog panel allows specifying the name of the output file. A default filename is suggested, which consists of the frame ID of the selected chromosome, together with the customary ".gbk" file suffix. Pathway Tools will be busy for about a minute (for a few thousand genes), while writing out the file.

Not all the rich data in a PGDB can be represented adequately in a Genbank file, so there may be some information loss. For example, there is no obvious and standardized way to capture gene synonyms in a Genbank file. The comments for genes are written to the /note feature qualifier, but comments for the corresponding gene products (usually a protein) are not written out.

The resulting Genbank file can be read in by other, external gene annotation tools. It is know to be readable by **Artemis**, version 7.1.

Genbank files can also be read by the PathoLogic component of Pathway Tools when constructing a new PGDB. The combined read and write capability allows, for example, using the Pathway Hole Filler to improve the annotation of existing Genbank files, by

running them through Pathologic and the Pathway Hole Filler, and then exporting the resulting data again in the Genbank format.

4.4 Linking Table Export

The **File** → **Export** → **Generate Link Tables...** command allows generating a set of Tab-delimited files containing IDs and names of various objects to help with creating Database links to and from external sources. A dialog panel allows selection of a directory for these files. The suggested default location is the "data/" subdirectory. For more information about these tables, consult Section 7.5.9.

You can also bulk import links to external databases. See Section 7.5.9.4 for more information.

4.5 Full Flat File Dump

The **File** → **Export** → **Entire DB to attribute-value and BioPAX files** command allows writing out the full set of flat files for the currently selected PGDB that we also make available for download through `http://biocyc.org`. The various objects in the Pathway Tools schema are written in both column-delimited and attribute-value formats. A BioPAX file is written as well. For a detailed list of the files, their contents, and their formats, see the online description at `http://bioinformatics.ai.sri.com/ptools/flatfile-format.html`.

4.6 Frame Import/Export

The **File** → **Import** → **Frames from File...** and **File** → **Export** → **Selected Frames to File...** commands can be used to import and export collections of frames to files in two different formats: to column-oriented files in which each column is delimited by a specific character, and to an attribute-value format that resembles MEDLINE export format. Frames to be exported can be selected by means of query commands (i.e., the Answer List), or by browsing through the hierarchy of object classes (such as exporting all frames that are instances of the class `Promoters`).

One use of this system is for users to export a set of data, edit it using a program such as Excel, and re-import the edited data.

A second use case for this facility is for creation of new frames. Imagine that you have run an external promoter prediction program, and that you want to import its results into

a PGDB. First, use this facility to export one or more existing promoters to a file. Examine the format of the file carefully. Next write a program that formats your promoter data into that file format, and run the importer on that file. Be sure the file contains previously unused unique identifiers for each new promoter.

4.6.1 Frame Export

Begin by opening the Export dialog, by selecting **File** → **Export** → **Selected Frames to File**. This brings up the dialog shown in Figure 4.6.1.

Figure 4.1: File Export dialog

Because of the complexity of the export process, the dialog displays only items that are relevant to your current set of choices. In general, items further down on the window are dependent on those higher up. So, as you change your choices for items near the top, items lower down may appear or disappear. The complete list of settings for export is as follows:

1. **Source of frames to be exported:** This item allows you to choose whether exported frames will come from the Answer List or by selecting a class from the Pathway Tools class hierarchy. If the Answer List is empty then the choice is obvious and this item is grayed out.
2. **Choose Slots:** This item appears only if "Answer List" is selected as the source of frames for export. It displays a menu showing the complete list of slots for the frames on the Answer List. You must choose at least one slot from this menu in order to export from the Answer List.
3. **Browse Frames:** This item appears if "Browse" is selected as the source of frames for export. It brings up a browser window for classes and slots, shown in Figure 4.6.1. The left pane of this window displays a tree of classes, similar to the

file folder browsers found in most operating systems. Classes that have a "+" to their left have subclasses that are not yet displayed. Clicking on the "+" opens up another level of the hierarchy, displaying a list of subclasses that may themselves have subclasses. Once a level of hierarchy is displayed, the "+" symbol changes to a "-", and clicking on the "-" closes the sublevel of the hierarchy. Rolling the mouse over a class displays its name, and in some cases additional documentation in the documentation pane immediately below the "OK" and "Abort" buttons. You can select only one class at a time. Once a class has been selected, the set of slots for that class is displayed in the right pane of the browser. Roll the mouse over a slot to display documentation for the slot. Clicking on a slot toggles whether or not it is selected. You must select at least one slot in order for export to proceed. Clicking "Select All" will select all slots, while clicking "Clear" will de-select all slots. Once you have selected a class and a set of slots, click "OK" to return to the main dialog. Click "Abort" to return to the main dialog.

4. **Include:** This item appears only if "Browse" is chosen as the source of exported frames. Selecting "Direct only" limits export to frames that are direct instances of the selected class, whereas "Direct and Indirect" exports both the direct instances of the class, and all instances of classes that are descendants of the selected class. For example, if "Reactions" is the selected class, "Direct only" selects a very small number of frames, whereas "Direct and Indirect" selects a very large number of frames, because, in addition to the small number of frames that are direct instances of the "Reactions" class, "Direct and Indirect" selects all Binding Reactions, Transport Reactions, and so on. In many cases, selecting a higher-level class and "Direct only" results in no frames being selected at all, because most frames are instances of classes at the bottom level of the class hierarchy.

5. **File Format:** This item allows you to select between two export file formats, "Delimited Columns" and "Attribute-Value". See the description of Delimited Column and Attribute-Value formats below for more detail on these formats.

6. **Column Delimiter:** This item appears only if "Delimited Column" is chosen as the file format. It offers three choices: "TAB", "Comma", and "Other". Choosing "TAB" or "Comma" produces TAB-separated or comma-separated output files, respectively. "TAB" and "Comma" are the most common formats used for importing into spreadsheet programs, such as Excel. If you choose TABs or commas as column separators, avoid exporting slots that may include these characters in their values. Choosing "Other" allows you to choose any single ASCII character as a column separator. For text slots such as comments, it is generally best to choose a character that does not appear within the slot you are exporting, such as "^", as a column separator.

7. **Multiple Slot Value Delimiter:** This item appears only if "Delimited Column" is chosen as the file format. When using the Delimited Column format, slots with

multiple values are handled by concatenating the values, with a user-specified delimiter character between values. If the same character is specified when the file is re-imported into Pathway Tools, these fields are split back into multiple slot values on import. We recommend that you use a character even more uncommon than the one used for a column delimiter, because many slots have multiple values. The initial default for this item is '$'.

8. **Output file:** This item allows you to choose the name of the file in which to place exported data. If the file name is not specified as a fully qualified file path, the export file is created in the directory you were in when you started Pathway Tools. A "Browse" button allows you to choose a file name by means of a standard file browser.

9. **Include file header**: If this item is checked, the file begins with a documentation header specifying the name of the file, the name of the DB from which the data was exported, the date and time that the export operation began, the username of the user who performed the export, the names of the classes of the exported data (or top-level class only, if frames were chosen by browsing), and the names and documentation strings of all slots exported, whether or not those slots actually had any values in the frames exported.

Once all fields have been filled, click "OK" to begin the export operation. A progress bar is displayed during the export process and a window showing the number of frames exported is displayed when exporting is complete.

Exported files can be edited with a standard text editor or any program capable of decoding the export format. In particular, most spreadsheet programs should be able to import column-delimited files easily. In a column-delimited file, the first line following the documentation header contains the names of the slots for exported frames, and is interpreted as a header on import.

4.6.2 Frame Import

The **File** → **Import** → **Frames from File...** brings up a dialog similar to the one used for specifying export parameters, shown in Figure 4.6.2. As is the case for the export dialog, items that are not relevant in the context of the choices specified by items higher up in the window are hidden.

1. **File Format:** With this item, you can choose between "Delimited Columns" and "Attribute-Value" formats. See below for details of these formats.

2. **Column Delimiter:** This item appears only if "Delimited Columns" is chosen as the import format. As on the export dialog, it allows you to choose TAB, Comma, or

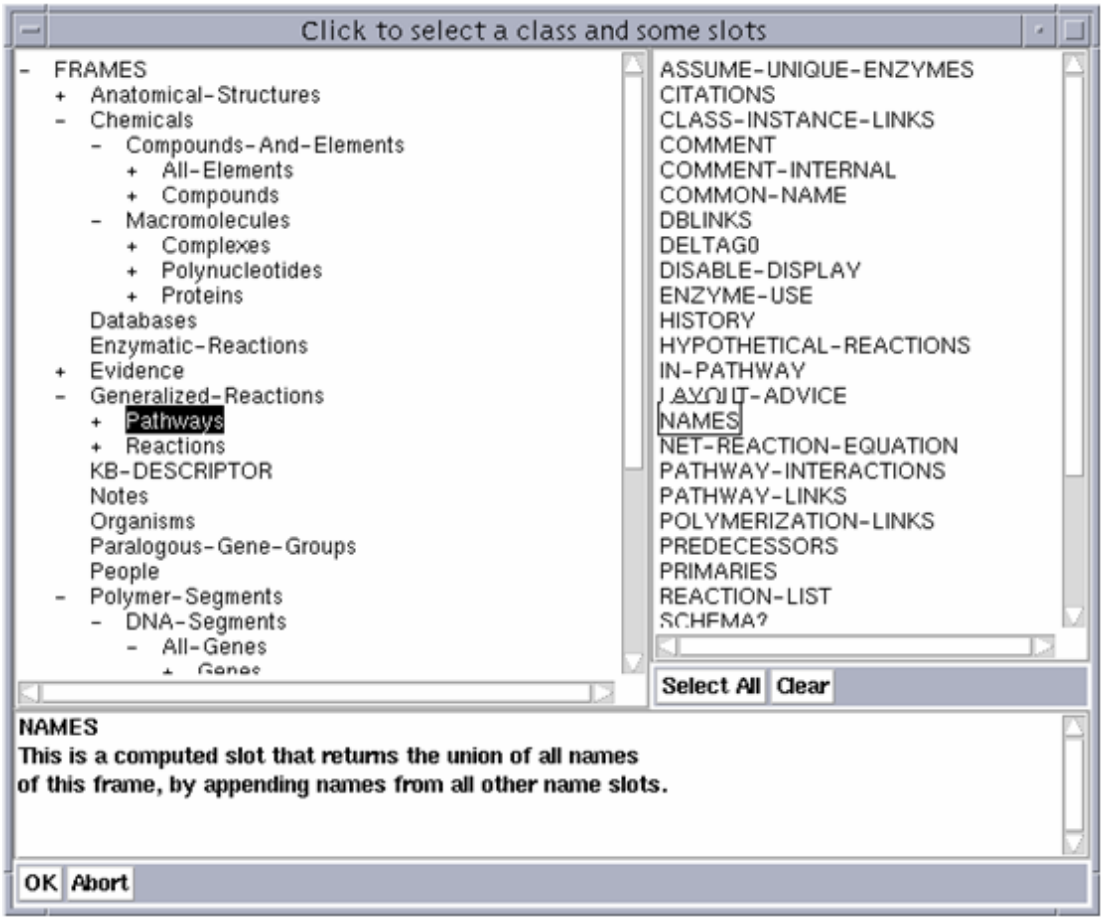


Figure 4.2: Pathway Tools class and slot browser

some other ASCII character as the column delimiter. The value of this item must match the delimiter used in the import file.

3. **Multiple Slot Value Delimiter:** This item appears only if "Delimited Columns" is chosen as the import format. Within columns values in the imported data, this character is used to split single column values in the file into multiple slot values in the DB. As is the case for the column delimiter, the character specified in this field must match the delimiter used in the import file.

4. **If object exists:** The Pathway Tools import facility allows very flexible behavior when an imported frame has the same unique identifier as an existing frame. This item offers the following choices:

   (a) **Replace entire object:** Import erases all existing slot values for the frame, including those for slots not found in the import fil**e**. The frame slots are then repopulated with values from the file. You should use this option with cau-

tion.

(b) **Update slots:** Import augments the existing slot values for the frame with values from the import file. This item causes the "If slot value exists" prompt, described below.

(c) **Log, do not import:** The name of the frame in the import file is noted in a log file, but the frame is not imported into the DB.

(d) **Ignore completely (no logging):** The frame from the import file is silently discarded, without being logged. This option can be used to verify that an import file can be parsed without errors and without affecting the DB.

(e) **Ask user each time:** This option allows different frames from the import file to be dealt with individually at the user's discretion. When a duplicate frame is encountered, a dialog appears giving the name of the frame and a menu of choices. See the description of the "If slot value exists" item for details of the choices provided. The dialog also includes a checkbox, which allows any subsequent duplicate objects to be dealt with based on the current choice, with no further user input. If "Update slots" is chosen, you may still be asked about the disposition of individual slot values.

5. **If slot value exists:** This item appears only if "Update objects" is selected as the value of "If object exists" and it controls what happens when the import process encounters slots that already have values. The possible choices for this item are

   (a) **Replace existing value:** Any existing value(s) for the slot is erased and the value(s) in the import file replaces them. We expect this to be the most commonly used option, because it ensures that edits to the export file are incorporated into the PGDB.

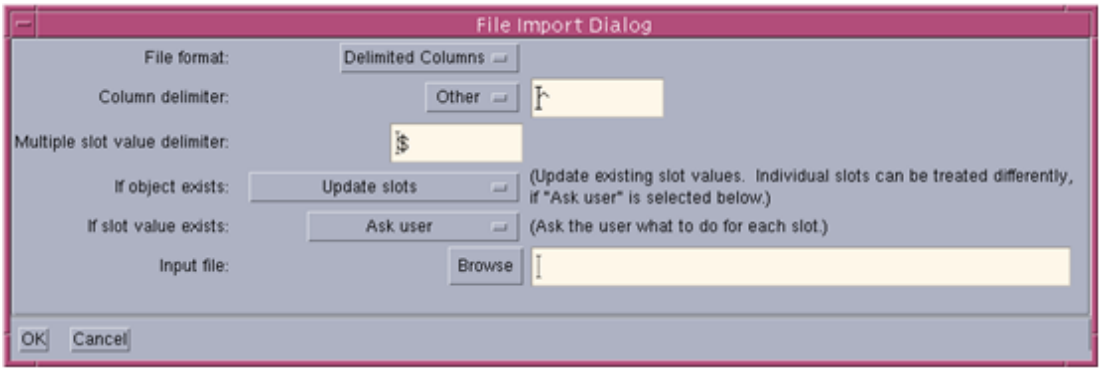


Figure 4.3: File Import dialog

1. **Add to existing values:** The existing value for the slot is augmented with values from the import file.

2. **Log and continue:** The existing slot values are left undisturbed and the values in the file are noted in a log file.

3. **Ignore completely:** The existing slot values are left undisturbed and the values in the file are discarded.

4. **Ask user:** If a slot with existing values is encountered, a dialog appears showing the existing slot values and the values found in the import file. Available choices are the same as those listed above for the "If slot value exists" item.

   The dialog also includes a checkbox, which allows any subsequent duplicate values for the slot in question to be dealt with based on the current choice, with no further user interaction. This applies only to the current slot: any other slots that have "ask user" selected will still cause the dialog to appear.

5. **Input file:** You can enter the name of the import file in the text field for this item or use the "Browse" button to induce a graphical file browser. The file name must include the file's extension. Unless the file is in the directory you were in when you started Pathway Tools, it must also include the full directory path for the file.

Once all fields have been filled, click "OK" to begin the import. A progress bar appears and tracks the import process throughout the DB update. If "ask user" was specified for dealing with either existing frames or existing slot values, you may be presented with a dialog for dealing with the frame or slot in question. In each case, the available choices are the same ones listed above for dealing with these cases. As soon as the import has completed, a window appears, summarizing the data imported. Any values in the import file that are an exact match for the existing values in the DB are ignored and do not change the DB in any way. If you suspect that the import has corrupted the PGDB, or if the import aborts before completion, you should use **File** → **Revert Current DB** to restore the PGDB to its previous state. Correct any errors in the import file and try the import again.

- **About the log file:** If "log" is chosen as the disposition of an import value that clashes with an existing frame or slot value, an entry is written to the log file. The log file is written in the same directory as the import file and its name is the same as that of the import file, with the extension ".log" appended.

4.6.3 Supported file formats for frame import and export:

Detailed documentation for Pathway Tools import and export file formats can be found at `http://bioinformatics.ai.sri.com/ptools/flatfile-format.html` . This documentation is reproduced below for convenience.

4.6.3.1 Column Delimited Format

Each line in the file represents a single frame in the DB. Columns are separated by a single character, of the user's choice. Multiple values for single slots are put into single columns. All the values of the slots are concatenated into a single string, with individual values delimited by a single user-specified character.

Following the documentation header, which is the same for both column-delimited and attribute-values formats, the first line in a column delimited file is a header line, with columns separated by the same character that delimits columns in the rest of the file. If all the frames in the file are of the same class, the first column of the header is the name of the class; otherwise, it is the string "FRAME". The last column of the header is always the string "CLASSES".

Within the portion of the file that represents individual frames, the first column is always the name of the frame, and the second column is the names of the classes from which the frame inherits, with multiple class names separated by the same character used to delimit multiple slot values. Subsequent columns are in alphabetical order of the slot name.

4.6.3.2 Attribute-value format

This format is similar to MEDLINE format. Frames are encoded as multiline entries. Each entry is separated by a blank line. Within each record, attribute names are at the beginning of each line, followed by the string "- ", followed by the value of the attribute. Attribute values can span multiple lines, in which case the second and subsequent lines of the value begin with spaces. Multiline values are reassembled into single strings containing new lines on import. Multiple slot values are handled by putting each value into a single attribute-value pair. On import, the user interface for dealing with multiple slot values is identical to that for column-delimited files: the user is queried about all the values for a single slot with a single dialog.

4.7 Importing citations from PubMed

Wherever possible, citations should be referenced by their PubMed ID in citations fields within the various editing forms and within comments (see Section 7.3.14 for more information). Usually, details of PubMed references (such as title and author information) will be downloaded from NCBI PubMed via the Internet by Pathway Tools when the user exits from an editor tool such as the protein editor.

In addition, citation details can be downloaded in bulk by invoking the command **File** → **Import** → **Citations from PubMed**. This command iterates through all objects in the

PGDB, searching for references that have not yet been imported, and retrieves details for those references. Importation from PubMed is necessary in order for a description of the referenced publication to appear at the bottom of the corresponding Navigator display.

Chapter 5

Database Sharing Via the PGDB Registry

The PGDB registry is an Internet-based mechanism for sharing PGDBs among Pathway Tools users. You can list the contents of the registry to find out what PGDBs are available for you to download into your local copy of Pathway Tools, and you can register PGDBs that you have created within the registry for sharing with other Pathway Tools users. In this chapter we use the terms "database sharing", "database publishing", and "database registering" interchangeably.

You may want to download a PGDB from the registry for use with your local copy of Pathway Tools for any of the following reasons. (a) To use the Pathway Tools comparative operations across a set of PGDBs, those PGDBs must all be resident within one copy of Pathway Tools. (b) A local copy of Pathway Tools will often execute faster than a remote Pathway Tools Web server. (c) The desktop version of Pathway Tools contains operations not present in the Web version of Pathway Tools, summarized at `http://biocyc.org/desktop-vs-web-mode.shtml`.

The registry itself is a server operating at SRI that lists a set of available PGDBs, their authors, the dates when they were deposited, and the location of a server from which they can be retrieved. That is, the registry itself does not actually contain the PGDBs; they reside at sites maintained by the users who chose to register them. The current contents of the registry can be viewed through the Web at URL `http://biocyc.org/registry.html`. In addition, Pathway Tools provides commands to list the current contents of the registry, to download a PGDB from the registry for local use, and to register a PGDB that you have created with the registry. These commands are discussed in the following sections.

5.1 Downloading PGDBs from the Registry

The Pathway Tools command **Tools-→Browse Downloadable PGDBs** will dynamically query the SRI registry server via the Internet to list the current contents of the registry, and will display the list of available PGDBs in a dialog window. The Last Updated column shows, for each PGDB, the date and time that a version of the PGDB was last deposited in the registry and remote site. Right-clicking on an organism name will display more complete information about that PGDB.

To download one or more PGDBs from the registry, select them by left-clicking on the organism names, and then click the button **Fetch & install selected PGDBs**. Pathway Tools will download an archive file for the PGDB from its remote site and unpack the archive file into a location where your Pathway Tools will find it.

To open the PGDB, close the "PGDB Registry Contents" dialog window, invoke the command **File-→Summarize Databases**, and click on your newly downloaded PGDB to open it and make it your current PGDB.

Be aware that problems could arise due to mismatches between the version of Pathway Tools used to create a PGDB and the version of Pathway Tools that you are currently running. If the version of Pathway Tools used to create a PGDB that you have downloaded is older than the version of Pathway Tools you are running, Pathway Tools will try to upgrade the PGDB schema to be compatible with the current version of Pathway Tools. If the downloaded PGDB was created with a newer version of Pathway Tools than you are running, such upgrading cannot be done, and if problems occur, you should upgrade to a newer version of Pathway Tools.

The version numbers of BioCyc PGDBs registered by SRI are the same as the versions of Pathway Tools with which they were created. However, other Pathway Tools users might not follow that convention.

5.2 Publishing PGDBs in the Registry

The process of publishing a database (i.e., creating an entry for it in the registry) involves three steps:

1. The directories and files that make up the PGDB are packaged and compressed into a single archive file.
2. The packaged file is moved to a Web server or an FTP server on a site of your choosing (usually one maintained by your organization) that allows anonymous access for retrieval of stored files.

3. The location of the packaged database, and essential information about its contents, is sent to the central SRI registry server, from which it is visible to Pathway Tools users across the Internet.

Fetching of shared databases by users who download them is done through either a standard Web server or an FTP server. If you publish one of your databases, you do not need to have Pathway Tools running on a continual basis in order for others to retrieve and install the database.

Before you publish your PGDB, ensure that the list of defined PGDB authors is as desired. If author information is not supplied when the PGDB is being registered, Pathway Tools will pop up a window asking if you want to fill it in. Also, the consistency checker should be run before the PGDB is submitted. If Pathway Tools notes that the consistency checker has not been run recently, it will pop up a window encouraging you to run it and giving you the option to do so.

As your PGDB evolves, you should plan to upload new versions of it to the registry at regular intervals using the processes described in this section.

5.2.1 Details of what happens during each step:

1. Packaging the database:

   The "tar" program is used to package the entire directory tree for the database into a single file, which is then compressed using the "gzip" program. The compressed file is then moved to the standard directory for temporary files (this is different for Windows and for UNIX/Linux—see below for details).

2. Moving the archive to a server:

   If your site configuration allows it, FTP (File Transfer Protocol) can be used to copy the packaged database from the standard temporary directory on your system to the remote server that you specified in the Database Sharing preferences dialog.

   Note: This step does not depend on an external "ftp" program, but runs entirely within Pathway Tools.

   Once the file has been copied to the remote server, the temporary file is deleted. This step is for your convenience. You can also move the packaged database manually to the location on your Web server or FTP server from which users can access it.

3. Registering:

   Pathway Tools contacts the central registry server over the Internet and transfers information about your database that will allow others to browse and download it. The information stored includes

- Database authors
- The species name and strain
- Genome
- Version
- Tier
- Copyright
- A comment for the entire database
- Contact email address
- The URL of a license agreement for the database

We have made every effort to automate the process of sharing databases. For security reasons, anonymous FTP servers are usually organized in a way that requires files that are copied to the server to be manually moved to a different directory for subsequent public retrieval. Our user interface allows you to perform each of the three required steps separately, in case a manual step is required between the packaging/storing and registering steps. Because of the wide range of platforms and network configurations in general use, we can provide only general instructions for copying files to your remote server manually, if necessary. The details of this process are dependent on the setup of your site — consult your local system administrator for more information on how to do this at your location.

5.2.2 Preliminary Step: Setting Preferences

Before actually registering your PGDB, you need to fill in the Database Sharing preferences dialog. This is available from the main command menu via **Tools** → **Preferences** → **Database Sharing**.

Note: These preferences are needed *only* for publishing your own databases, not for fetching shared databases from other sites.

The first time that you invoke the preference dialog for database sharing, an initial dialog window prompts you to enter some basic properties of the way your remote site is set up. Unless there are major changes to the way your site is administered, you should not need to run this initial setup more than once. Nevertheless, the main preferences dialog has a button that displays this dialog again, so that it is always possible to go back and change your answers to the questions in the initial dialog.

Once the two questions on the initial setup dialog have been answered, the main preferences dialog is displayed. It contains the following items:

Enable PGDB sharing functionality : This checkbox can be used to completely disable PGDB sharing, if desired. In general, enabling PGDB sharing should have little effect on the operation of Pathway Tools. The one exception to this is that displaying the summary page for all organisms' queries the server for databases that were installed, via the database sharing facility, to see if a newer version is available. If the server does not respond within a few seconds, the query is simply skipped. The summary page should display fairly quickly even if the servers for these databases are completely unavailable. If you notice a delay in displaying the summary page, temporarily disabling PGDB sharing may eliminate the delay and will have no effect on the functioning of any of your databases.

Rerun initial setup: Pushing this button will cause the initial setup dialog to appear and will allow you to change your answers to the basic setup questions.

The rest of the preferences dialog is divided into two parts. The first set of fields contains information needed for storing files on a Web server or an FTP server. If you answered "no" to the question about using a remote server for storing files in the initial settings dialog, this section will not appear at all.

Server to which files will be uploaded : If you are using Pathway Tools to upload your PGDB to an FTP server, enter the Internet name or address of the FTP server to which you will store your registered PGDBs, just as you would enter it for manually starting an FTP session.

Username for storing to the FTP server above: This is the username that will be used to log in to the FTP server. For security reasons, we do not store passwords between sessions. A pop-up dialog for entering the password will appear the first time that you store a database in each Pathway Tools session.

Directory to which archive files will be copied: Enter the full pathname for the directory within the FTP server. This is what you would give for an FTP cd command, when using FTP to transfer files by hand. In most installations, this will *not* be the same directory from which remote users will retrieve archived PGDBs.

The next set of fields contains information that will be sent to the central registry server, in order for users at other sites to connect and retrieve your databases. Regardless of how you copy files to the FTP server prior to making them available for retrieval, these values *must* be filled in before contacting the registry server for the final step in publishing your PGDB.

URL from which archived databases will be retrieved : This should be a full URL identifying the path where your PGDB archive file(s) will reside, as described above. Example:

`ftp://ftp.podunk.edu/pub/joesmith/registry/`

or

`http://www.podunk.edu/organisms/`

Once you have filled in all the fields in the preferences dialog and saved your preferences, you can continue with the process of publishing local databases.

5.2.3 Registering a PGDB

A NOTE ON LICENSING: The first time that you use any portion of the database sharing facility, whether for publishing your own databases or for retrieving databases from other sites, the click-through license agreement for the Pathway Tools Registry will appear. Carefully read the license agreement. If you are in agreement with all terms of the license agreement, click "I ACCEPT" to continue. If any of the terms of the license agreement are not acceptable, click "I DO NOT ACCEPT". The database sharing facility will not function until the license agreement has been accepted.

The main user interface for publishing your PGDBs will differ, depending on how you answered the basic setup questions. There are three possible cases, each documented separately below. Refer to the section that matches the setup of your FTP site.

Case 1: "Yes" to "use FTP for uploading", and "yes" to "is it necessary to move uploaded files before they can be retrieved".

The publishing process will have two steps: "package and upload" and "register".

Step 1:Package and Upload

Select the corresponding checkbox for the databases that you intend to publish, and when you are satisfied with your choices, click the button labeled "Package and upload selected databases". The packaging and uploading process will probably take several minutes, during which a series of messages will detail the steps being performed. When the process is complete, a "Done" message will appear in the message window.

Before continuing with the "Register" step, you must move your packaged databases to their final locations on your server, from which they can be retrieved using the Web or anonymous FTP. If you are not sure how to do this at your site, consult your local system administrator.

Step 2:Register

As noted above, you *must* have moved your files to their proper locations for retrieval by outside users before performing this step. Select the databases that you want to publish by left-clicking the corresponding checkbox in the column labeled "Select for publishing". If you require users to execute a click-through license agreement for the database, select that checkbox as well.

We strongly recommend that before completing the registration process you click the "Refresh (check server)" button. This command connects to the remote server and path that you have designated in the Database Sharing preferences dialog, and tries to obtain the file size (not the contents) of the archive file for each of the databases listed. If the file is missing, or for some reason Pathway Tools is unable to connect to the server, any error messages encountered are displayed in the message window, and the database is grayed out and will not be processed by the "Register" command. **Note:** When you click "Register", the same check is done for each database to be registered, and processing of all databases will be stopped. Running the "Refresh" command allows you to fix any problems *before* you actually try to register your databases.

If you plan to use a click-through license for any of your databases, the "Test a click-through license" button prompts for a URL and then displays the contents of that URL just as it will be displayed to users who want to install your databases. This gives you a chance to fine-tune the content and presentation of your license agreement files, before they are presented to actual users.

When you are satisfied with your choices, click the "Register selected files" button. If you have specified that a license agreement will be required for any of your databases, you will be prompted for the URL of the license agreement. Assuming that there are no problems with your Web server or FTP server, the registration process should complete very quickly.

Case 2: "No" to "use FTP for uploading".

Pathway Tools automates only the packaging of your databases, and all the work of moving the archived databases to a server is done outside of Pathway Tools.

Click on the corresponding checkbox for the databases that you intend to publish. When you are satisfied with your choices, click the "Package selected databases" button, and the packaging process will begin. Packaging normally takes a few minutes to complete, during which time a series of messages will keep you informed of the process's progress.

When packaging has completed, a message will show the location of the files containing the packaged databases. Before running the "Register" step, you must move these files into the locations that you entered in the Database Sharing preferences dialog, from which they can be retrieved by users at other locations.

Once the files have been moved to your server and are ready to be retrieved, the "Register" step is identical to that for Case 1, above. Follow the instructions for Step 2 above to complete the process of publishing your databases.

Case 3: "Yes" to "use FTP for uploading", "no" to "is it necessary to move uploaded files before they can be retrieved".

The entire publishing operation can be accomplished with a single click.

Select the databases that you want to publish by left-clicking the corresponding checkbox in the column labeled **"Select for publishing"**. If you require users to execute a click-through license agreement for the database, select that checkbox as well. When you are satisfied with your choices, click the **"Publish selected databases"** button.

The process of packaging, storing, and registering your databases will take several minutes to complete. A subwindow at the bottom of the publishing dialog will display information about the various steps involved as they are performed.

If no errors are encountered during the publishing process, a **"Done"** message will appear in the message window. Your databases will be available for immediate retrieval by other Pathway Tools users. The **"Browse Downloadable PGDBs"** command should show your databases along with those contributed by other users.

5.2.4 About Click-Through Licenses

If desired, you can require users at other sites to execute what is commonly referred to as a "click-through" license, before they are allowed to retrieve and install your databases on their system. If the user clicks on "**I DO NOT ACCEPT**", the database will not be retrieved from the server on which it is stored.

To create a click-through license, you need to put the text of your license into a file, which can be accessed by means of a standard Web URL. Pathway Tools supports a limited set of HTML tags, to allow you to add boldface, italics, and a few other formatting options. Specifically, the **B, I, H1, H2, H3,** and **H4** tags are supported. Other HTML formatting, such as table directives, will be ignored. HTML formatting cannot be extended across line breaks. If you have boldface text that extends across multiple lines, you need to add <**B**> at the beginning of each line, and <**/B**> at the end of each line. Lines separated by only a single new-line will be filled, to allow the text to fit neatly into the window that Pathway Tools puts up. Multiple new-lines separate paragraphs. Note that the filling process does not preserve word breaks across lines, so it is necessary to put a space at the beginning of each line in a paragraph. For an example of a file with this type of formatting, see the SRI click-through license at

http://bioinformatics.ai.sri.com/ptools/downloadable-database-license.html

If you view this file using the **"View -**$rightarrow$ **Source"** command available in most Web browsers, you can see the formatting tags.

Chapter 6

PathoLogic: Automated Creation of Pathway/Genome Databases

The PathoLogic component of Pathway Tools supports the creation of new pathway/genome databases (PGDBs) from the annotated genome of an organism. The program assumes that the positions of genes within the genome have already been identified and that gene functions have already been predicted. PathoLogic contains several modules that provide some combination of automated computational inference, manual review of those inferences, and manual assignments.

- The program infers metabolic pathways by analyzing the genome annotation with respect to a reference database of metabolic pathways, MetaCyc. PathoLogic also generates reports that summarize the evidence used for deducing the pathways inferred and the results of comparing the reactions carried out in the subject organism versus those in the reference database.
- The program infers which genes are likely to code for missing enzymes in the predicted metabolic pathways (pathway holes).
- The program predicts operons.
- The program guides the user in assigning monomers to multienzyme complexes.
- The program identifies transport proteins in the genome, and infers transport reactions from the free text descriptions of transporter function that are present in the genome.

We define the term *metabolic pathways* in this document to mean pathways involved in small-molecule metabolism. Therefore, the PGDB created by PathoLogic does not represent pathways involved in macromolecular metabolism, transport pathways, nor signal-transduction pathways.

The process of inferring the metabolic network of an organism from its genome, called *pathway analysis*, extends the paradigm of genome analysis by producing interpretations of a genome sequence that are biologically more informative than the annotated genome alone. For example, predictions of individual enzymatic gene functions for a given organism may be viewed from the perspective of the entire predicted metabolic network for the organism. From this "whole-organism" perspective, one may determine whether the given functional assignment fits into the predicted metabolic network in a coherent way; for example, does it help form a model of the organism's metabolism that is consistent with known physiological and biochemical properties such as experimentally isolated enzymatic activities, growth media substrate requirements, and/or substrate utilization patterns?

PathoLogic performs most of its work automatically, but some interactive assistance from the user is required. This chapter describes the steps carried out by PathoLogic and how it is used. Section 6.1 describes the events that take place during the execution of the Predictor. Section 6.2 describes PathoLogic input file formats in detail. Section 6.3 gives a step-by-step guide to how to use PathoLogic. Section 6.4 describes various refinements one can make to the database once created. Section 6.5 describes the reports generated by the Predictor. Section 6.6 describes the format of the two input files expected by the Predictor when running in automated "batch mode".

6.1 Overview of PathoLogic Execution

In defining the processing performed by PathoLogic, we begin with a rather brief technical description of what transpires at the database level. We then describe, chronologically, the phases of the Predictor execution.

6.1.1 Database Generation Perspective

PathoLogic generates a PGDB representation of the genome and metabolic pathways of the subject organism. More specifically, the principal goal of PathoLogic is to create the appropriate set of instance frames, to populate them with appropriate slot values, and to interconnect these frames in a manner that accurately reflects their semantic relationships. Two types of information are encoded in the new PGDB:

- A set of class frames that encodes the database schema. These frames are copied from MetaCyc .
- A set of instance frames that encode the chromosomes, genes, proteins, reactions, pathways, and substrates of the subject organism.

The key to understanding how PathoLogic operates is to understand how it creates, populates, and interconnects these frames. PathoLogic creates one frame for each genetic element (e.g., chromosome or plasmid) in the organism, and populates its slots with data extracted from the annotation and sequence files that PathoLogic accepts as input. It creates one frame for each gene described in the annotation file. Some of the gene information in the input file is used to populate slots in the gene frames (such as the base-pair position of the gene and the name). Other attributes populate slot values in the frame describing the protein or RNA product of the gene (e.g., the product name). PathoLogic also creates a link from the gene to the genetic element that contains it, based on the association of the gene to a file for a particular genetic element.

PathoLogic first initializes the new PGDB with the MetaCyc schema. It then reads the genetic-elements.dat file, and creates a frame for each genetic element defined in that file. Next, it reads the input file for each chromosome, and creates a gene frame for each gene in the input file. It creates one polypeptide frame for each gene whose product is a protein, and creates a frame in the rRNAs or tRNAs class for genes whose products are rRNAs or tRNAs. Then, PathoLogic creates connections between polypeptide frames and the appropriate reaction frames for those proteins that are enzymes. The connections are made based on the EC numbers and GO terms assigned to a protein in the annotation file or using a name-matching tool (for a more detailed explanation, see Section 6.3.7.1). The actual connection is made using an intermediary frame called an enzymatic reaction that describes the pairing of an enzyme and a reaction; see [11] for a detailed explanation of the role of enzymatic-reaction frames. The next step is to match the reactions now known to be catalyzed by the organism against the reactions in each MetaCyc pathway. PathoLogic initially imports every pathway containing a reaction in the subject organism (and imports all reactions and compounds in those pathways), but it then prunes out some pathways for which it decides insufficient evidence exists for their presence.

6.1.2 PathoLogic Operation

PathoLogic accepts the following inputs:

- The MetaCyc DB, which is included with Pathway Tools.
- A file called **genetic-elements.dat** which defines each genetic element in the organism, and provides pointers to each of the following files:
- A FASTA file containing the DNA sequence for each genetic element in the subject organism (i.e., each chromosome or plasmid).
- A file containing the annotation for the corresponding genetic element (e.g., features, chromosomal locations, gene functions). This file can be in GenBank format or PathoLogic format.

PathoLogic generates the following output :

- A new PGDB for the subject organism.
- A collection of reports summarizing the results of the prediction process.

The major steps in using PathoLogic are as follows:

1. Interactively enter defining information for the new PGDB, such as the name of the organism and the names of the DB authors.
2. Create the input data files.
3. Perform one or more trial parsing operations on the input data files to (a) ensure that they are in the proper format — this step might be repeated multiple times if the files are not initially in the proper format, and (b) ensure that as many enzyme names as possible are recognized by the PathoLogic enzyme name matcher — this step might be repeated multiple times after research on unrecognized enzyme names.
4. Build the new PGDB, which automatically creates the genes, proteins, reactions, and predicted pathways within the PGDB.
5. Run the operon predictor.
6. Run the pathway hole filler.
7. Define protein complexes.
8. Infer transport reactions.
9. Create the Cellular Overview diagram.

6.2 PathoLogic Input File Formats

The PathoLogic input files are described here in more detail.

6.2.1 File genetic-elements.dat

This "master file" is an index to a larger set of files provided for the organism. These files are designed to accommodate both fully assembled genomes and partially assembled genomes. Each entry in the **genetic-elements.dat** file describes one genetic element (meaning a chromosome or plasmid) or one contig. Each entry specifies the data files

provided for each genetic element or contig, and specifies properties of the genetic element or contig (for example, whether a genetic element circular or linear; to which genetic element a given contig belongs).

A sample genetic-elements.dat file is provided at `http://bioinformatics.ai.sri.com/ptools/sample-genetic-elements.dat` . The comments at the start of the sample file define the syntax of the file, the allowable fields, and the values that should be provided for each field.

For each genetic element or contig, two additional input files must be provided:

- A sequence file in FASTA format, containing the full nucleic-acid sequence of the genetic element or contig. This file is identified by the suffix fsa or fna (e.g., tpal.fna is the FASTA file containing the Treponema pallidum complete genomic sequence).
- An annotation file , describing the predicted genes for that genetic element or contig. The annotation file should be in either PathoLogic format or GenBank format. PathoLogic relies upon the file suffix to determine which format the file is in: suffix gbk indicates GenBank format (e.g., tpal.gbk is the file containing the annotation inGenBank file format), and suffix pf indicates PathoLogic format.

6.2.2 The PathoLogic File Format

PathoLogic has a simple and easy-to-parse attribute-value-based file format. Each gene record starts with a line containing the **ID** attribute, and ends with a line containing two slashes ("//"). One attribute-value pair is allowed per line, and the value is separated from the attribute name by one tab character. The location of the gene in the corresponding FASTA file is specified by the attributes **STARTBASE** and **ENDBASE**. These locations refer to the start and end of transcription for a given gene and thus indicate the direction of transcription along the chromosome. Lines starting with ";" are comments and are ignored by the parser (see examples in file format below).

The valid attributes are:

ID: The unique identifier for the gene. Should be the same as the unique ID used by the sequencing effort. If supplied, it will become the frame ID for the gene. Highly recommended.

NAME: The mnemonic shorthand name used for the gene. It will become the common name of the gene. Required.

STARTBASE: An integer. Location in the corresponding FASTA file of the translation start of the gene. The location corresponds to the first nucleotide of the initial triplet, typically ATG. Optional.

ENDBASE: An integer. Location in the corresponding FASTA file of the translation end of the gene. The location corresponds to the last nucleotide of the last translated codon. When the gene is on the minus strand, the endbase will be a smaller number than the startbase. Optional.

FUNCTION: The assigned function of the product of the gene. It will become the common name of the protein. The word ORF should be used if the function is unknown. For multifunctional proteins, supply multiple FUNCTION lines in the file. Required.

PRODUCT-TYPE: The type of gene product, as chosen from the following controlled vocabulary of terms. P = protein (or a hypothetical ORF); PSEUDO = pseudogene; TRNA = tRNA; RRNA = ribosomal RNA; MISC-RNA = some other RNA, such as can be part of various ribonuclear protein complexes. Required.

SYNONYM: Additional synonyms under which a gene may be known can be indicated here. One value per line. Optional.

EC: If the annotation effort has yielded an enzyme assignment with EC numbers, they should be indicated in the standard way of using four numbers (and/or dashes), separated by three dots. Recommended.

GO: If the annotation effort has yielded an assignment of Gene Ontology (GO) terms, they may be specified here, one per line, in the formats described in Section 6.2.4. GO term annotations with associated evidence codes or citations must be specified using this attribute; annotations lacking an evidence code and a citation may also be specified using the DBLINK attribute. Recommended.

DBLINK: Used for linking the gene object to other databases. String should be of the form <DB:Accession>. If available, the UniProt accession number that corresponds to the gene is specified here (see the PathoLogic format example below for how to specify a link). The following are some of the databases for which links may be created:

- ENTREZ
- GeneID
- CGSC
- UNIPROT (synonym: SP)
- PDB
- REFSEQ
- MetaCyc
- SWISSMODEL

- LIGAND
- ENZYME-DB
- LIGAND-MAP
- PIR
- GO
- PFAM

Links to Gene Ontology (GO) terms are processed identically to GO attributes lacking evidence codes or citations.

For a complete list of the current databases that are already defined in Pathway Tools, please see the MetaCyc External Databases web page.

To specify a link to some database not in this list, simply create a unique ID for it (like one of the above) and specify the link in the same way as to any other database. The link will be created, but will not be active until you have edited the resulting PGDB to create a description for the new database. See Section 7.5.9.3 for instructions about how to do this. Optional.

GENE-COMMENT: An additional comment may be placed here. It will be placed in the comment slot of the gene frame . Optional.

FUNCTION-COMMENT: An additional comment may be placed here. It will be placed in the comment slot of the protein frame or the enzymatic-reaction frame. If there are multiple functions for the gene product, this comment will be assigned to the immediately preceding function. Optional.

PRODUCT-ID: A unique identifier for the object that encodes the gene product. Optional.

FUNCTION-SYNONYM: Other names by which the gene product is known. If there are multiple functions for the gene product, this synonym will be assigned to the immediately preceding function. Optional.

INTRON: The start and end positions, in absolute base pair numbers with a hyphen (-) in between, of an intron in the gene. Each intron should appear on a separate line. Optional.

The attributes **FUNCTION**, **SYNONYM**, **EC**, **DBLINK**, **FUNCTION-COMMENT**, and **FUNCTION-SYNONYM** may be used several times per gene record.

Here is a short example PathoLogic Format file. For a longer example file, see `http://brg.ai.sri.com/ptools/tpal.pf`.

```
;;; The PF file format.
;;; This is a comment in front of an imaginary example record.
;;; It starts with a semicolon. Each record starts with an "ID"
;;; line, and is terminated by a "//" line.
ID      b1262
NAME    trpC
STARTBASE       1317812
ENDBASE 1316451
DBLINK  SP:P00909
PRODUCT-TYPE    P
SYNONYM foo
SYNONYM foo2
GENE-COMMENT    f453; 99 pct identical to TRPC_ECOLI SW:P00909
;;; The following shows how information about multiple functions of
;;; a protein is supplied:
FUNCTION        N-(5-phosphoribosyl) anthranilate isomerase
EC      5.3.1.24
FUNCTION-SYNONYM        phosphoribosyl anthranilate isomerase
FUNCTION-COMMENT        Amino acid biosynthesis: Tryptophan (3rd step)
FUNCTION        indole-3-glycerolphosphate synthetase
EC      4.1.1.48
FUNCTION-COMMENT        Amino acid biosynthesis: Tryptophan (4th step)
//
```

6.2.3 GenBank File Format

The information PathoLogic extracts from the GenBank file comes from its feature table , which is fully defined in the document *The DDBJ/EMBL/GenBank Feature Table Definition* [3]. To allow you to provide additional information not currently supported by GenBank format, PathoLogic supports a few additional feature qualifiers : *product_comment*, *EC_number*, and *alt_name*; see below for their descriptions.

When the annotation for a genetic element is provided in GenBank format, providing a separate sequence file for the genetic element is optional. If a separate sequence file (in FASTA format) is not provided, the sequence will be parsed from the GenBank file and stored in a FASTA file.

A common omission from GenBank files is the following line, which must precede the genes and other features in each GenBank file. Be sure to include it in yours. Note that 13 spaces follow the word "FEATURES":

```
FEATURES             Location/Qualifiers
```

In our experience, most genome centers do not prepare their GenBank files in a manner that is fully consistent with the specification of GenBank file format. For example,

they sometimes put a given piece of information in the wrong field. Here we describe the fields within the GenBank feature table that PathoLogic uses, and what information PathoLogic expects to find in those fields. We also indicate the degree to which these fields are required by PathoLogic.

The accepted features are

CDS: Gene that codes for a protein (or which represents a hypothetical ORF)

tRNA: Gene that codes for a tRNA

rRNA: Gene that codes for a ribosomal RNA

misc_RNA: Gene that codes for some other RNA, such as can be part of various ribonuclear protein complexes

The accepted qualifiers for any of the above features are

/gene: The official gene symbol used for the gene. In the absence of a gene symbol, a mnemonic shorthand name is suggested. The value of this qualifier will become the common name of the gene. Some annotation files contain the value desired here in the `/product` qualifier, enclosed in parentheses, at the end of the string (e.g., `/product="ATP-dependent protease LA (lon-1)"`). Required.

/EC_number: If the annotation effort has yielded an enzyme assignment with EC numbers, they should be indicated in the standard way of using four numbers (and/or dashes), separated by three dots. Recommended.

/product_comment: * An additional comment can be put here. It will end up in the comment slot of the protein frame. Optional.

/locus_tag: The unique identifier for this gene. Its recommended value is the unique ID assigned by the group that sequenced the genome. For example, The Institute for Genomic Research typically assigns IDs of the form "GSnnnn", where G and S are the first letters of the genus and species names, respectively, and nnnn is a number unique to each gene. An example unique ID assigned by TIGR to Helicobacter pylori is "HP0023" and to Caulobacter crescentus is "CC1087". If a value for the `/locus_tag` qualifier is supplied, it will become the frame ID for the gene. If no `/locus_tag` qualifier is supplied, Pathway Tools will use the value of the `/label` qualifier if present, and otherwise Pathway Tools will generate an arbitrary unique ID for each gene. Highly recommended.

/gene_comment: * An additional comment can be put here. It will be placed in the comment slot of the gene frame. The contents will also be parsed to identify Gene Ontology (GO) annotations in the formats described in Section 6.2.4. Optional.

/note: This feature qualifier is processed identically to the /gene_comment qualifier. Optional.

/alt_name: * Additional synonyms by which a gene may be known can be indicated here. Optional.

/product: The assigned function of this protein. It will become the common name of the protein. The string "ORF" should be used if the function is unknown. Only real functional assignments should be entered here. This means that values such as "similar to..." or "putative..." should be changed to "ORF". Required.

/pseudo: Indicates that the CDS is a pseudogene. Optional.

/db_xref: A cross-reference to another database. The string should be of the form <DB:AccessionID>. Links to the Gene Ontology (GO) database are processed as GO annotations lacking evidence codes or citations. Optional.

/go_component, /go_function, /go_process: Gene Ontology annotations, possibly including evidence codes and citations, may be specified using these feature qualifiers. For the details of the expected format, see Section 6.2.4. Recommended.

Some of the feature qualifiers above (e.g., `alt_name` and `EC_number`) may be used several times per gene record (a single value per line), if several values need to be specified. The "*" indicates that the qualifier is not an official GenBank/EMBL/DDBJ feature qualifier.

Here is an example:

```
FEATURES             Location/Qualifiers
     CDS             complement(3480761..3481768)
                     /gene="gap"
                     /EC_number="1.2.1.12"
                     /EC_number="1.2.1.13"
                     /product_comment="glycolysis"
                     /label="b1779"
                     /gene_comment="o331; 100 pct identical to
                     G3P1_ECOLI SW: P06977;
                     CG Site No. 718; alternate name gad"
                     /alt_name="gad"
                     /alt_name="foo"
                     /product="glyceraldehyde-3-phosphate
                     dehydrogenase"
                     ...
```

6.2.4 Specifying Gene Ontology Terms

Beginning in version 12.5, PathoLogic supports Gene Ontology (GO) annotations for gene products specified in either PathoLogic (.pf) or GenBank input files. GO annotations are used for two purposes in the PGDB built by PathoLogic. First, GO annotations are stored in the GO-TERMS slots of gene product frames (proteins and RNAs). Second, GO annotations from the "molecular function" aspect of the ontology are used to link gene products to the reactions they catalyze (see Section 6.3.7.1 for more details on this process).

The simplest way to specify GO term annotations is to use the DBLINK attribute (in the PathoLogic file format) or the /db_xref feature qualifier (in the GenBank file format). Simply create a database link of the form "GO:xxxxxxx", where "xxxxxxx" is replaced by the numeric identifier for the GO term. For example "GO:0000250" represents the molecular function "lanosterol synthase activity".

A limitation of the DBLINK and /db_xref formats is that they do not allow the specification of evidence codes and citations associated with GO annotations. For this reason, we also provide a GO attribute in the PathoLogic file format and support the /go_component, /go_function, and /go_process feature qualifiers in the GenBank file format. The value associated with a GO attribute or /go_* qualifier may be in either of the following formats:

- The GO term name, GO term ID, citation PubMed ID, and evidence code, separated by vertical bars. For example:
  `antimicrobial humoral response|0019730|16163390|IMP`
- The GO term name, followed by three expressions in square brackets: the GO term ID, evidence code, and citation PubMed ID. For example:
  `antimicrobial humoral response [goid 0019730] [evidence IMP] [pmid 16163390]`

In order to extract as much information as possible from a genome annotation, we also support the parsing of GO annotations from the `/note` feature qualifier of GenBank files. An example of such annotations is the following:

```
/note="similar to CCA1 (CIRCADIAN CLOCK ASSOCIATED 1),
transcription factor [Arabidopsis thaliana]
(TAIR:AT2G46830.1); similar to late elongated hypocotyl
[Castanea sativa] (GB:AAU20773.1); contains InterPro
domain Homeodomain-related; (InterPro:IPR012287); contains
InterPro domain Homeodomain-like; (InterPro:IPR009057);
contains InterPro domain Myb-like DNA-binding region,
SHAQKYF class; (InterPro:IPR006447); contains InterPro
domain Myb, DNA-binding; (InterPro:IPR001005);
go_component: nucleus [goid GO:0005634] [evidence IEA];
go_function: DNA binding [goid GO:0003677] [evidence RCA];
go_function: transcription factor activity [goid
```

```
GO:0003700] [evidence ISS] [pmid 11118137];
go_function: transcription factor activity [goid
GO:0003700] [evidence ISS] [pmid 9657154];
go_process: regulation of circadian rhythm [goid
GO:0042752] [evidence IMP] [pmid 12007421]"
```

The text of the note consists of fields separated by semicolons; some of the fields begin with `go_component:`, `go_function:`, or `go_process:`. Following these tags are GO annotations in the square-bracket format. These will be processed appropriately.

6.2.5 Directory Structure for a PGDB

During the course of PGDB construction, PathoLogic creates a directory tree within the **ptools-local** directory tree that is used by your Pathway Tools installation (see Section 2.1), to store all information about that PGDB. For example, this directory tree contains reports created by the PathoLogic parser and the PathoLogic enzyme name matcher, it contains the sequence files for the organism, and, for PGDBs stored in files, it contains a file containing the complete PGDB.

Upon completion of the build of the new PGDB, the directory tree created by PathoLogic has the following structure and files:

```
ptools-local/pgdbs/user/ORGIDcyc/
  default-version
  VERSION/
    input/
      genetic-elements.dat
      organism.dat
      organism-init.dat
      <chromosome.{fsa,fna}>
      <chromosome.{pf,gbk}>
    reports/
      name-matching-report.txt
      trial-parse-report.txt
    data/
      <orgid_CHROMOSOMEID.fsa>
      overview.graph
    kb/
      ORGIDbase.ocelot
```

where `ORGID` and `VERSION` are the Organism ID and version number you entered into the *Input Project Information Window* (see Figure 6.2). The files denoted by `<chromosome.{fsa,fna}>` and `<chromosome.{pf,gbk}>` are the sequence and annotation files, respectively. The files denoted by `<orgid_CHROMOSOMEID.fsa>` are

copies of the sequence files provided by the user in the input directory. The file `ORGIDbase.ocelot` is the file containing the complete PGDB for PGDBs stored in files.

PathoLogic creates the file `default-version`, putting inside it only the version number that you entered into the *Input Project Information Window*. If you later create any additional versions of the PGDB, PathoLogic updates the `default-version` file each time. If you decide to abandon the latest version of a database, then you can manually edit the `default-version` file to specify the version you want Pathway Tools to use.

6.3 Creating a Pathway/Genome Database

You will typically execute the PathoLogic operations described in this section in the order we describe, although there is some flexibility in this order.

6.3.1 Invoke PathoLogic

Invoke PathoLogic through the Pathway/Genome Navigator menu using the command **Tools** → **PathoLogic**. PathoLogic creates a separate new window that runs in a separate process so that PathoLogic operations can run in parallel with Navigator operations. The PathoLogic window is shown in Figure 6.1. PathoLogic commands are organized into several menus that roughly correspond to the order in which the commands are used.

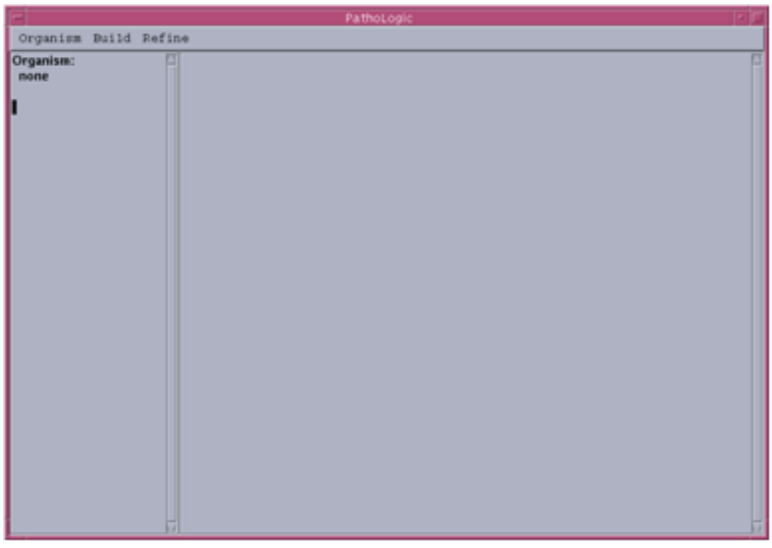


Figure 6.1: PathoLogic window

6.3.2 Create New Organism

Begin creation of a new PGDB with the command **Organism** → **Create New** . This command creates the PathoLogic Project Information dialog shown in Figure 6.2. Enter information about the organism for which you are creating a new PGDB. The Organism ID should be a short unique identifier for the PGDB itself (unique with respect to the other organism identifiers in directories **aic-export/pgdbs/biocyc/**, **ptools-local/pgdbs/user/**, and **ptools-local/pgdbs/registry/**). Most of the information requested in this dialog should be self-explanatory, but we provide descriptions of some of these fields here. Most of the information requested in this dialog can be altered at a later time (such as the PGDB authors), but this dialog is the most convenient place to enter the information.

The information that should be supplied in the fields in this dialog is as follows:

Organism/Project ID: A short mnemonic for the name of the organism that will be used as a unique identifier for the organism. This ID should be unique with respect to all other PGDBs. This ID will be used to construct the name of the directory tree containing the files for this PGDB. For example, "bsub" might be used as the identifier for B. subtilis.

Database Name: The name to use for this DB when communicating with the user. Examples: "BsubCyc", "BsubtilisDB".

Organism Taxonomic Class: The full name of the organism or the ID for the organism from the NCBI Taxonomy database, `http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?db=Taxonomy`. The NCBI Taxonomy is integrated with Pathway Tools. In order to create a PGDB you need first to select the appropriate taxon from the NCBI Taxonomy or MetaCyc. (MetaCyc contains some organisms that are not in the NCBI Taxonomy.) In the unlikely case that there is not yet a taxon for the organism then one will be created but a parent for the taxon must be selected from the NCBI Taxonomy or MetaCyc. In the even more unlikely case that there is no appropriate parent that you must select the best matching unclassified class in the NCBI Taxonomy, leaving full specification of the lineage for a later time. Selecting the most appropriate class will enable a better prediction of pathways. (See Section 6.3.7.2 for information on how taxonomic information is used in pathway prediction.) You can select a class by either entering the full name of your organism or the NCBI Taxonomy ID, or by clicking the select button that will start the NCBI Taxonomy browser (see Figure 6.3.)

Create organism: You may specify whether the organism should be created and the taxonomic class selected at the previous step should be the parent rather then the organism to be created. Use this option only if your organism does not have an NCBI Taxonomy ID. If such an ID becomes available you can add it at a later time.

Full Species Name: The species name of the organism (this and most subsequent fields are not editable if you didn't create the organism).

Abbreviated Species Name: An abbreviated name for the organism.

Subspecies: The name of the subspecies of the organism.

Strain: The name of the strain of the organism that the PGDB describes.

NCBI taxonomy ID: The ID for the organism from the NCBI Taxonomy database (`http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?db=Taxonomy`).

Rank: The NCBI Taxonomy rank; the only allowable values are species, subspecies, varietas (plants), forma (plants) and strain. The value "other" indicates a value which is not allowed. The program will not accept it.

Taxonomic lineage: The taxonomic lineage shows the classification of your organism in the NCBI Taxonomy.

Default Codon Table: Specifies which genetic code this organism uses.

Mitochondrial Codon Table: Specifies which genetic code is used in the mitochondrion of this organism.

DB Storage Type: Specifies where the new PGDB will reside. A storage type of "file" means the PGDB will reside in a flat file. This approach is much easier to use, but saving of DB updates will be slower, and multiple users cannot edit a file PGDB simultaneously. A storage type of "oracle" or "mysql" means the PGDB will reside in an Oracle or MySQL database, respectively. This approach requires configuration of the Oracle or MySQL DB (see the Pathway Tools Installation Guide at `http://bioinformatics.ai.sri.com/ptools/installation-guide/released/index.html` for details). It is always possible to transition a PGDB from one storage type to another. We recommend starting with a storage type of file and transitioning to Oracle or MySQL later if necessary.

Authors: A list of the authors of the PGDB, and their institutions.

Citations: A list of Medline UIDs for citations for the PGDB, such as the publication of the full genome sequence of the organism, if available.

Project Home Page URL: The URL of a home page for this PGDB, if any.

Project Primary Contact Email: The email address of a primary contact person for this PGDB, to whom requested DB corrections should be sent.

Copyright String: A copyright notice (in HTML format) for this PGDB, if any is desired.

Footer citation for Web pages: A string (in HTML format) specifying a publication that users of the PGDB should cite.

The data collected in this dialog is saved in two files, called **organism.dat** and **organism-init.dat**, in directory **`ptools-local/pgdbs/user/ORGIDcyc/VERSION/input/`** . To run PathoLogic, you must have write access to the **ptools-local** directory tree. Comments in these two files describe the conditions under which you may edit the files.

The remainder of this section describes the process of creating a new database.

You are required to enter values for *Organism ID* and *Organism Taxonomic Class*. The value entered for *Organism ID* is automatically converted to upper case by the Predictor and is restricted to contain any combination of alphanumeric characters and underscores ("_"). This restriction occurs because the Predictor uses the *Organism ID* to generate the name of the top level directory. It is recommended that you choose a short (single word) mnemonic name as the organism name (e.g., "ctra" for *Chlamydia trachomatis*, "bsub" for *Bacillus subtilis*). The full species name (e.g., *Homo sapiens*) is required since it is needed for the GenBank queries from the gene mode to work.

This operation may take a few minutes to finish because it is initializing the schema (class hierarchy) of the new PGDB and saving this initialized form of the PGDB.

6.3.3 Create genetic-elements.dat File

Once a new PGDB has been initialized, you must create a file called **ptools-local/pgdbs/user/ORGIDcyc/VERSION/input/genetic-elements.dat** to describe the one or more chromosomes or plasmids of this organism. PathoLogic will have created a file called **sample-genetic-elements.dat** in this same directory, which you can copy and edit. The format of this file is defined in Section 6.2.1. Creation of this file is essential because it provides pointers to all the other input files required by PathoLogic. You cannot proceed further with the operation of PathoLogic without creating this file.

6.3.4 Specify Reference PGDB

In some cases, you may have already created or downloaded a PGDB for a different, related organism. This other PGDB, if it has been curated, may contain reactions or pathways not present in MetaCyc, and you may wish to see if these additional pathways or reactions can be predicted in your new PGDB. To specify one or more PGDBs to use as references for pathway prediction in addition to MetaCyc, invoke the command **Organism → Specify Reference PGDB(s) .** You will be presented with a list of available PGDBs. Select any you wish to use as a reference and click OK. Once you save the database, this

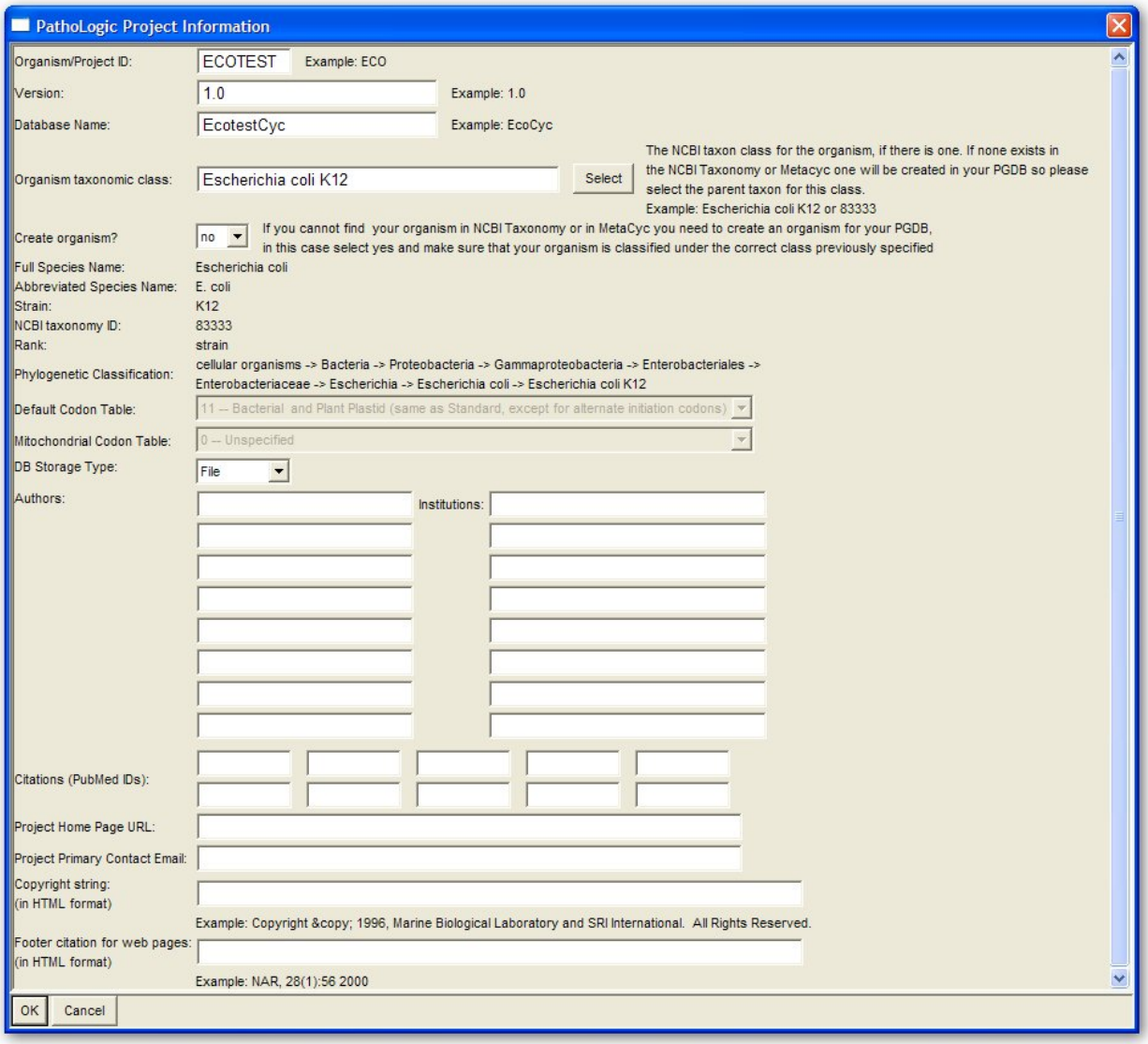


Figure 6.2: Input Project Information window (after user input)

selection will remain effective for all subsequent attempts to infer reactions and pathways until you change it again.

When a PGDB is specified as a reference, all pathways and reactions that are in that PGDB and are (a) not already in MetaCyc and (b) supported by evidence other than purely computational evidence will be imported into MetaCyc before enzyme-to-reaction mapping and pathway prediction are performed. The changes will remain a part of MetaCyc for the remainder of the user's session, but will not persist for future sessions (though they will be repeated again if further PathoLogic work is done in those sessions). In addition, any enzyme names from the reference PGDB will supplement the enzyme names in

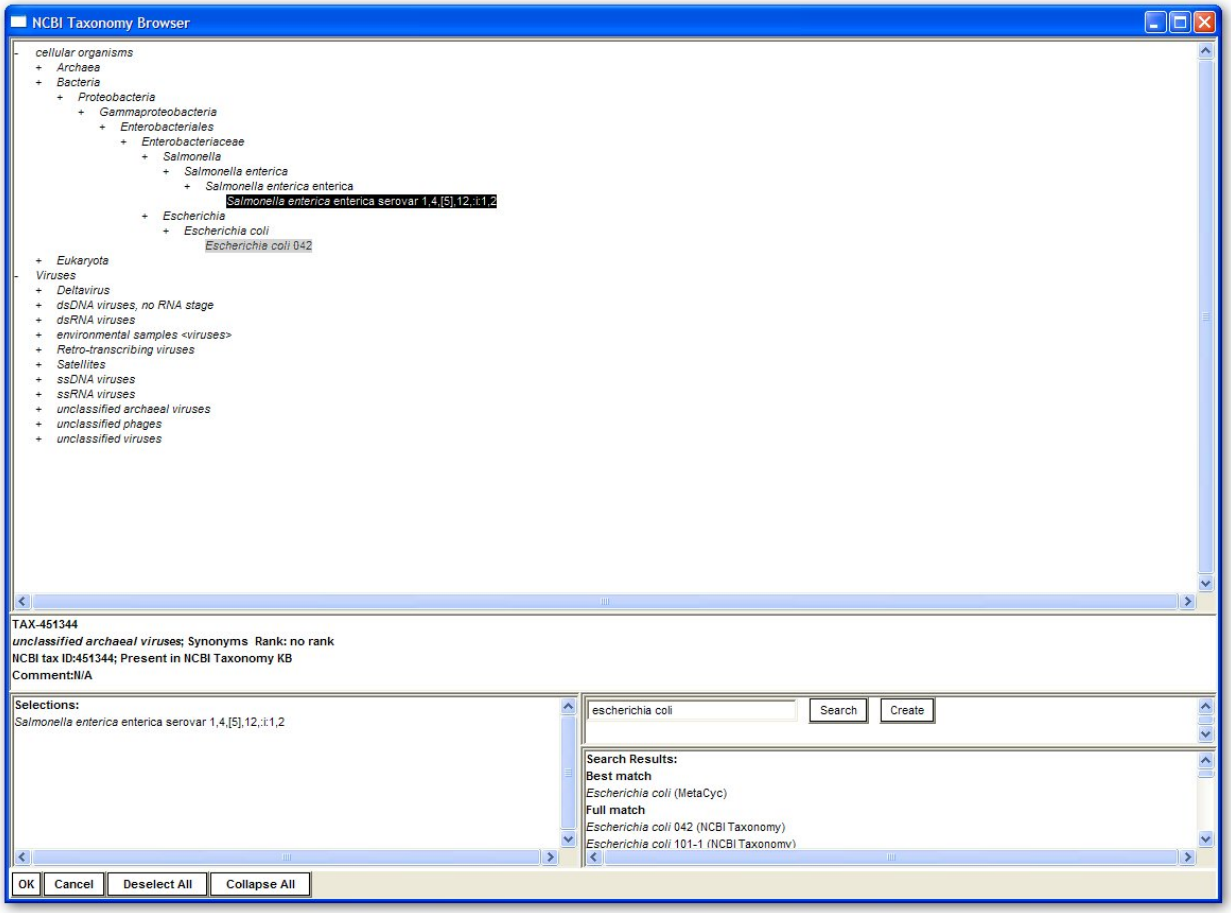


Figure 6.3: The NCBI Taxonomy Browser

MetaCyc for the purpose of matching enzyme names to reactions.

Note: This step is entirely optional — most users will find that they do not require any other reference PGDB besides MetaCyc. None of the organism PGDBs that SRI distributes as a part of Pathway Tools contain additional reactions or pathways not in MetaCyc. Users would find little benefit in specifying them as reference PGDBs. This functionality is designed for users who have access to PGDBs that have been curated outside of SRI, whose pathways have not yet been incorporated back into MetaCyc.

6.3.5 Trial Parse

The **Build** → **Trial Parse** operation allows you to test whether the input files (see Section 6.2) can be properly parsed by PathoLogic. Thus, you can detect and correct errors in the input file before the file has been used to populate the new PGDB. Malformed input files

are much easier to correct at this early stage in processing than after manual refinement of the PGDB has begun.

When this command is invoked, a dialog window is created that allows you to choose one or more files to be parsed. Click the **Parse** button to initiate parsing. You can run the parser as many times as you like. Click **Done** when you do not wish to perform any more parsing operations. The parser sends output to the main PathoLogic window that lists, for each genetic element, the number of genes and gene products found in the corresponding annotation file by the parser. Sample output is shown in Figure 6.4.

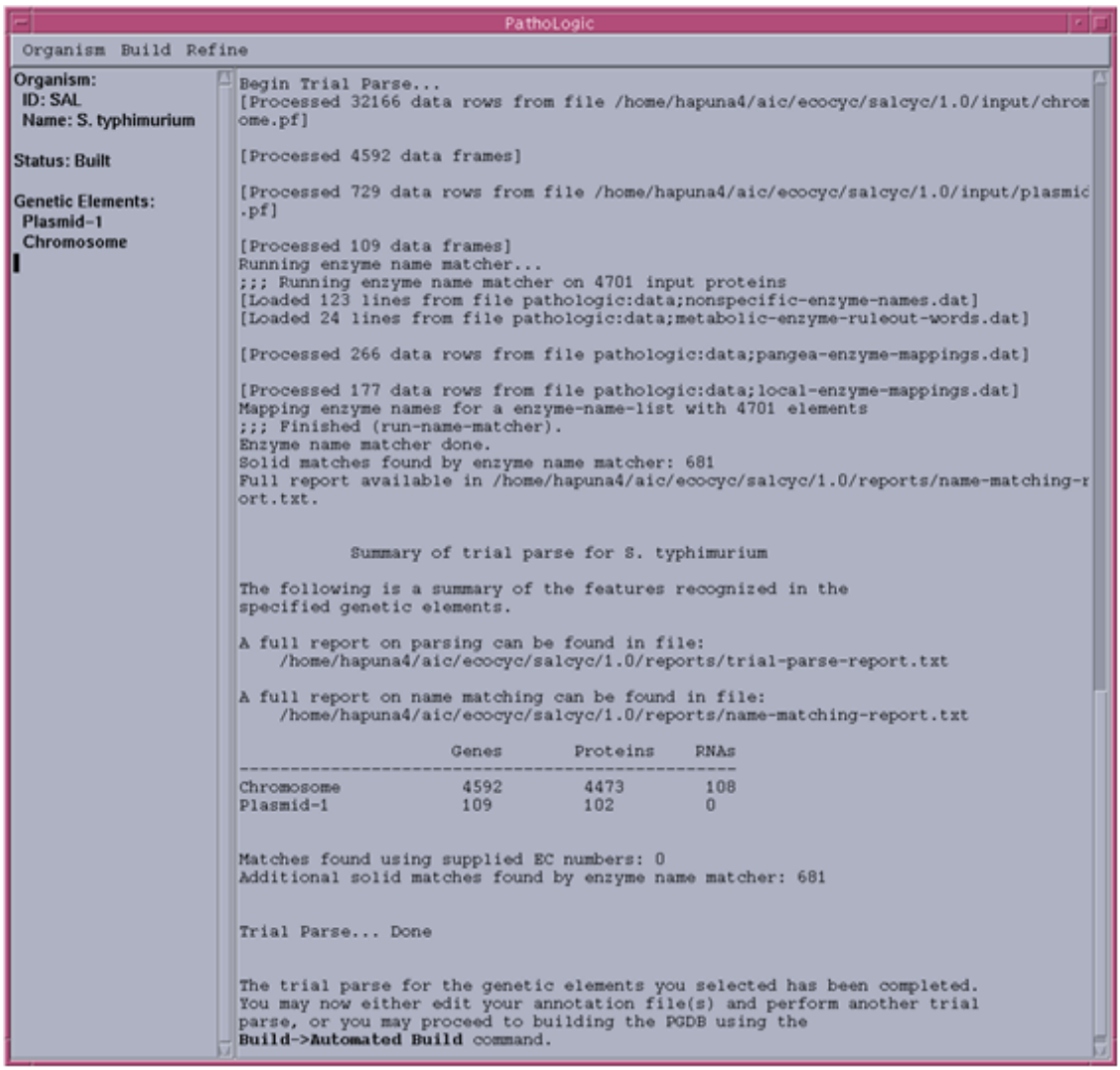


Figure 6.4: Trial parse output

A more detailed summary of the data extracted by the parser will be written to a file called `trial-parse-report.txt` whose exact location is given in the output. Carefully compare the statistics reported for each annotation file with the values you expect. Does the

number of gene-IDs found in each file match the number of genes you know to be in each file? Does the number of startbase and endbase values match the number of genes with known chromosomal locations? The number of gene-product-types should be the sum of gene-Proteins, gene-tRNAs, gene-rRNAs, gene-snRNAs, and gene-miscRNAs.

Anomalies in these statistics are probably due to formatting problems with the files. Has the tab character been used as the separator between the attribute names and values in every line? Has the correct attribute name been used in every case? If errors in the input file are found, correct the files and repeat the Trial Parse until no more anomalies are found.

6.3.6 Build Pathway/Genome Database

By the time of the automated build, you should have resolved all syntactic problems with the input files, and should have manually added as many enzyme-reaction correspondences as possible for unmatched enzyme names, as described in Section 6.3.7.1.

The command **Build** → **Automated Build** is the main phase of PathoLogic operation. During this step, PathoLogic re-parses the input files, and it creates DB objects for each of the chromosome, genes, and gene products of the subject organism. It also links the products of those genes to reaction objects for as many enzyme-reaction associations as can be inferred automatically.

Note: This step takes several minutes to complete.

The parser again generates output that lists, for each genetic element, the number of genes and gene-products found in the corresponding annotation file by the parser. A more detailed summary of the data extracted by the parser is written to a file whose name is provided.

If you do not notice any obvious errors during the automated build, then save the newly built PGDB with the command **Organism** → **Save KB**. Saving of the database will take several minutes.

Should you wish to not save the results of the build so that you can modify the input files and rerun the automated build, select **Organism** → **Revert KB**, which will result in erasing from memory everything that took place since the PGDB was initialized. The build and revert operations may be performed as often as necessary.

The next section describes the pathway prediction process that occurs during the automated build operation.

6.3.7 Metabolic Pathway Prediction

This section explains the two main aspects of metabolic pathway prediction within PathoLogic: the matching of enzymes within a genome to reactions within the reference pathway DB(s), and the prediction of metabolic pathways.

Be aware that the pathway prediction process has several limitations.

The accuracy of pathway prediction strongly depends on the accuracy of the underlying genome annotation. Any errors in the inputs (in particular, the annotated genome and the reference database) will be carried through to the subject organism pathway/genome database. Error rates in annotated genomes are difficult to quantify since the accuracy of functional assignment by homology remains to be precisely defined. In general, it is expected that some functional assignments in a genome will be incorrect. In part, this reflects the limitations of methods for prediction of function based on sequence data. If an enzyme function is predicted incorrectly, then that incorrect function may provide evidence in support of the wrong metabolic pathway. And the absence of the true function will constitute a lack of evidence for a truly present metabolic pathway. However, be aware that false-positive metabolic pathway predictions provide an opportunity for recognizing errors in the underlying genome annotation.

Note added in 2008: As the number of pathways in the MetaCyc DB has grown over time (to now more than 1,000), the sensitivity of metabolic pathway prediction has increased, but so too has the false-positive prediction rate. Thus, we recommend a careful manual review of the predicted pathways, as described in Section 6.5, to remove false-positive predictions. In 2007 SRI modified PathoLogic to perform taxonomic pruning (see Section 6.3.7.2), and that has decreased the false positive rate. However, taxonomic pruning is imperfect because the taxonomic extents of many metabolic pathways are not accurately known. Thus, it can cause pathways to be improperly removed, particularly for organisms that are taxonomically obscure. SRI is now working on a completely new pathway prediction algorithm that we expect will have much greater accuracy. We expect it to be available in 2009 or 2010.

Computationally predicted metabolic pathways will necessarily be incomplete because MetaCyc is incomplete with respect to all metabolic pathways found in nature, and because the genomes upon which the pathway predictions are based are incompletely annotated. A requirement for recognizing that a particular enzymatic step is carried out in the subject organism is that at least one enzyme that catalyzes that step must have been sequenced in the past. However, for roughly half of known EC numbers, no sequence exists in any public sequence database. Therefore, many enzymes cannot be detected by sequence-similarity searches, and thus cannot be counted during pathway analysis. These will constitute genes for which no functional assignment can be made, that is, ORFs. As additional functional assignments are made to ORFs based either on sequence analysis and/or new experimental data, annotations of genomes will become increas-

ingly complete. This completeness will help make computational pathway prediction increasingly robust and less susceptible to error.

6.3.7.1 Matching of Enzymes to Reactions

An operation of central importance for PathoLogic is the matching of enzymes defined in the input files to reactions and pathways defined in the MetaCyc DB. By establishing a correspondence between an enzyme and the reaction it catalyzes, PathoLogic infers a structured description of the function of the enzyme, and immediately is able to place the enzyme in one or more pathways, since each reaction in MetaCyc is connected directly to those pathways that contain it. PathoLogic performs this enzyme matching during both the trial parse phase and the automated build phase of its operation.

If a fully-specified Enzyme Commission (EC) number is provided for an enzyme in an input file, PathoLogic prefers to use the EC number to match the enzyme to its corresponding reaction, because EC numbers are relatively unambiguous. ("Fully-specified" means that all four components of the EC string are numbers; e.g., 1.2.3.4, rather than 1.2.3.-.) PathoLogic also makes use of annotations to terms in the molecular function aspect of the Gene Ontology (GO). If an enzyme is annotated with a a GO term whose definition has a cross-reference to one or more EC numbers or MetaCyc reactions, the enzyme will be linked to those reactions.

If no EC number is provided, then PathoLogic attempts to match using the enzyme name. **Note:** The protein sequence of the enzyme is not used for matching. Our approach is to perform matching at the functional level rather than the sequence level because we prefer to utilize existing assignments of gene function that may have been made by expert sequence analysts rather than to attempt to infer new enzyme functions for each gene in an automated fashion.

PathoLogic matches enzyme names against a dictionary of reaction names that is constructed from the following sources:

- All the names of reactions, enzymatic reactions, and mono-functional enzymes contained in the MetaCyc DB and any user-specified reference PGDBs (see Section below.) (These names include the names of all *E. coli* enzymes from the EcoCyc DB and all enzymes found in the ENZYME database [1]; those names have been imported into reaction frames in MetaCyc.)
- A file that maps enzyme names not found in MetaCyc to MetaCyc reactions. The name of this file is `aic-export/pathway-tools/pathologic/VERSION/data/enzyme-mappings.dat`.
- An optional user-provided file that maps enzyme names not found in MetaCyc to MetaCyc reactions. The name of this file is

ptools-local/local-enzyme-mappings.dat. This file will remain unchanged across version upgrades of Pathway Tools, and can thus serve as a local repository of a user's mappings that may be specific to the set of databases that will be generated. An almost empty template file can be found at aic-export/pathway-tools/pathologic/VERSION/data/local-enzyme-mappings.
This sample file has 2 example entries, which can be removed.

- An additional optional user-provided file, specific to the given database that will be predicted. The name of this file is ORGIDcyc/VERSION/input/enzyme-mappings.dat. The purpose and format of this file are the same as for the above two.

The results of the matching process are summarized in the PathoLogic window for the user. A more detailed list of all matching and nonmatching enzyme names is saved to a file called **ptools-local/pgdbs/user/ORGIDcyc/VERSION/reports/name-matching-report.txt**, which is the *Enzyme Name to Reaction Mapping Report*. The name-matching process is applied to each gene product in the input annotation file(s). The name-matching outcome for a given gene product places it into one of the following sections of the file:

Unambiguous Match: A name in the lookup table matches the gene product name string exactly, and is associated with a single reaction or a set of unambiguous reactions. A set of reactions are unambiguous if they share the same EC number or are known to be catalyzed by the same enzyme. In this case, PathoLogic automatically creates the Enzymatic-Reaction (see the Appendix for a definition of this concept) connections between the appropriate enzyme and reaction frames in the PGDB.

Ambiguous Match: The gene product name string is ambiguous because it is associated with more than one reaction in the PathoLogic lookup table, and the reactions are ambiguous (see above). Since PathoLogic cannot make a decision as to the correct enzyme name to reaction mapping, it is up to the user to resolve the ambiguity as described later in this section.

Probable Metabolic Enzyme: The gene product name string has not matched any name in our lookup table, but is considered a probable metabolic enzyme because the following conditions are true: (a) one of the words in the gene product name ends in the string "ase"; (b) the name does not contain words that would indicate it to be a nonmetabolic enzyme, such as "protein kinase", "peptidase"; and (c) the name does not match any of a list of nonspecific enzymatic activity names such as "oxidoreductase" or "transaminase".

No Match: The gene product name string has not matched any name in the PathoLogic lookup table, and is not considered a probable metabolic enzyme.

If PathoLogic cannot find an exact match to an enzyme name in the lookup table, it will try to match alternative forms of the name generated by removing parts of the name that do not describe an enzymatic activity. For example, common prefixes such as "conserved" and "hypothetical" will be removed, as will some subunit and isozyme labels.

After matching has completed, the user can use the name matching report to drive a manual phase of enzyme name matching. The Possible Metabolic Enzymes section of the report contains the gene product names most likely to be unrecognized metabolic enzymes, so the user should focus efforts on that section. The user should take care to ignore product names that are not metabolic enzymes (such as protein kinases involved in cell signaling pathways), and to ignore product names that represent general, nonspecific enzymatic activities that do not correspond to a single reaction.

For example, imagine that the enzyme name matching report contains the name "carboxymuconolactone decarboxylase". The user can follow several strategies to identify what reaction this enzyme name corresponds to:

- In the Navigator interface, select the MetaCyc DB, enter protein mode, and perform a substring query on the phrase "carbox muc lact". This query will search all MetaCyc enzymes and reactions for names that contain all three of these substrings. A variant of this name might be present in MetaCyc, allowing you to identify the reaction. When the reaction is found, right-click on the reaction to find the MetaCyc frame name for the reaction (the frame name is printed at the top of the right-button pop-up menu).
- Search for the enzyme name in SwissProt, and try to discern the reaction that the enzyme catalyzes from the SwissProt entry, based on either its EC number (if present), or the Comment section, which sometimes describes the catalytic activity of the enzyme.
- Search for the enzyme name in PubMed, and try to discern the reaction the enzyme catalyzes from the literature, and then determine whether MetaCyc describes that reaction, such as by entering compound mode, searching for one of the substrates, and then examining the list of all reactions containing that substrate.
- Search for the enzyme name in other metabolic DBs such as KEGG, and again try to discern what reaction the enzyme catalyzes, and identify the MetaCyc entry for that reaction.

When the MetaCyc reaction corresponding to that enzyme name has been identified, you can assign the reaction to the enzyme using the **Refine → Assign Probable Enzymes** command after completion of the automated build procedure. This is the simplest and, in most cases, the preferred method of ensuring that the reaction is assigned to the enzyme. Alternatively, you can inform PathoLogic of this new enzyme to reaction correspondence

by entering the enzyme name and the reaction frame name in a new line in file **ptools-local/local-enzyme-mappings.dat**. This alternative is useful if the name is likely to appear in future annotation files for other PGDBs, as the local enzyme mapping file will be reused for later PGDBs. A third approach is to change the enzyme name in the input file to a name that you know PathoLogic will recognize because that name is defined in MetaCyc, such as if there was a typographical error in the original name. For the latter two approaches, the new assignment will take effect the next time the enzyme name matching procedure is invoked (e.g., if the PGDB is rebuilt or if the **Refine → Re-Run Name Matcher** command is invoked).

If the reaction corresponding to that enzyme cannot be identified, nothing further can be done for that enzyme. If the reaction can be identified, but it does not exist in MetaCyc, you should manually create the reaction in the new PGDB after completion of the automated build process.

6.3.7.2 Assigning Evidence Scores to Predicted Metabolic Pathways

So that PathoLogic can determine whether a pathway exists in the subject organism, the pathway predictor computes a score that reflects an approximate measure of the likelihood that the pathway is present. It then combines the score with other criteria to decide whether to copy the pathway from MetaCyc (or another reference pathway PGDB) to the new PGDB.

The first step in assessing the evidence that pathway P occurs in organism O is to consider the fraction of reaction steps in P that can be catalyzed by enzymes in O. For example, the reference pathway for pyrimidine biosynthesis is composed of the reactions represented by EC numbers 6.3.5.5, 2.1.3.2, 3.5.2.3, 1.3.3.1, 4.1.1.23, and 2.4.2.10. Sequence analysis of the *H. pylori* genome predicted the existence of enzymes that catalyze all the steps in pyrimidine biosynthesis except 2.4.2.10. Therefore, we say we have evidence that five of the six steps of *E. coli* pyrimidine biosynthesis can be catalyzed by *H. pylori*.

The method also considers evidence against the occurrence of P in O. If a reaction R occurs in two pathways, P_1 and P_2, we should exercise caution in counting the occurrence of an enzyme that catalyzes R as evidence for the presence of P_1 in O, when the enzyme might in fact be present in the genome only because of its role in P_2. In particular, if every reaction step that is present in pathway P is also present in some other pathway, the presence of P in organism O should be treated as very hypothetical. You may choose to later remove from the database those pathways for which no steps unique to that pathway are present.

We use the notation $X|Y|Z$ to denote the score for a pathway P in O, meaning that P consists of X total reactions, that enzymes for Y of the reactions are present in O, and that Z of the Y reactions are used in other known pathways. For example, the score computed for *H. pylori* for the reference pathway for valine biosynthesis is 4|2|1. This means that

two of the four reactions required for this pathway can be catalyzed by *H. pylori* enzymes; one of those two enzymes catalyzes a step in a different pathway (alanine biosynthesis).

Pathways are excluded from the new PGDB based on the following conditions. These conditions are primarily designed to address the fact that many different pathways share reactions in common, and more specifically, that MetaCyc contains a number of metabolic pathways that are highly related to one another, which we term "variant pathways". For example, MetaCyc contains five variants of glycolysis that share many reactions in common. PathoLogic intelligently selects among variant pathways by preferring those pathways containing a unique enzyme — an enzyme catalyzing a reaction step found in only a single pathway. A pathway is pruned from the PGDB if one of the following conditions holds:

- No reactions contained in the pathway are known to be catalyzed by the organism ($X = 0$), OR
- None of the reactions in the pathway that are catalyzed by the organism are unique to that pathway, AND
- The pathway consisted of more than two reactions but contained evidence for only a single reaction, OR
  - The set of reactions for which there was evidence was the same as the set of reactions for which there was evidence in some other pathway, but the pathway contains additional reactions (for which there is no evidence) not contained in the other pathway, OR
  - The set of reactions for which there was evidence was a proper subset of the set of reactions for which there was evidence in some other pathway, AND
    - Both the pathway and the other pathway are members of the same variant class, OR
    - The pathway was classified as a biosynthetic pathway and was missing at least the last two steps from the end of the pathway. The rationale for this criterion is that if the final steps are missing, then the pathway is not producing its intended target, so the other pathway(s) containing the common reactions are more likely to reflect the actual role of those reactions in the organism, OR
    - The pathway was classified as a degradation pathway and was missing one or more steps from its beginning. The rationale for this criterion is that if the initial steps are missing, then the pathway is not degrading its intended substrate, so the other pathway(s) containing the common reactions are more likely to reflect the actual role of those reactions in the organism, OR

* The pathway was classified as an energy metabolism pathway and was missing at least half its reactions. The rationale for this criterion is that there is a great deal of overlap of reactions among the energy metabolism, with many reactions appearing in many pathways. Thus, many false positives are likely to be inferred from among these pathways. An arbitrary but reasonable cutoff for excluding some of these false positives is 50%.

Our approach is conservative in the sense that it attempts to bring more potential pathways to the attention of the user, to allow the user to prune out those pathways rather than having PathoLogic prune them out. That is, PathoLogic attempts to err on the side of more false-positive pathway predictions than on the side of more false-negative predictions.

Each pathway in MetaCyc is tagged with its observed taxonomic range. Assuming taxonomic pruning is enabled (the default), the above algorithm applies only to pathways whose taxonomic range includes the organism. For pathways whose taxonomic range does not include the organism, much stricter criteria must be met – there must be enzymes found for all or almost all of the steps in the pathway in order for the pathway to be predicted. This requirement greatly reduces the number of false positive pathway predictions. Taxonomic pruning can be disabled by running the **Rescore Pathways** command and requesting that it be disabled.

The scores assigned to the computationally predicted pathways of an organism should be interpreted with due caution. First, you should evaluate pathway predictions and corresponding scores by reference to relevant experimental data, for example, growth media requirements, substrate utilization patterns, experimentally isolated enzymatic activities, relevant metabolic pathway studies and gene expression analysis. For example, there may be experimentally demonstrated enzymatic activities even though the corresponding genes for these enzymes may not have been identified in the genome. This situation would result in an artificially low score for any pathway that utilized these enzymatic activities. Second, you should consider the limitations of sequence analysis as a tool for understanding metabolic function. For example, genes within a genome may have been incorrectly assigned a given enzymatic function, resulting in an artificially high score for pathways that used that enzyme.

6.3.8 Update Build for New Annotation

Sometimes, after a PGDB has been built and subject to some level of manual curation, a revised annotation or set of updates to an existing annotation becomes available. Users do not want to rebuild their PGDB again from scratch, to avoid losing their manual curation. However, it is desirable to be able to incorporate the updates into the PGDB with a minimum of effort. The command **Build** → **Update Build for New Annotation** asks

the user for one or more updated annotation files. Update files can be in PathoLogic format or in GenBank format, and there can be one file for the entire organism, or one file per genetic element. In general, if the update contains one or more new genes, you should supply one file per genetic element, so that the software will know which genetic element to assign the new genes to. The update files need contain only those genes for which information has changed. Genes that have not changed can be included or not as desired. The software will parse the update files and determine all differences between the new data and the old. These differences will be summarized in a pop-up dialog like the example shown in Figure 6.5.

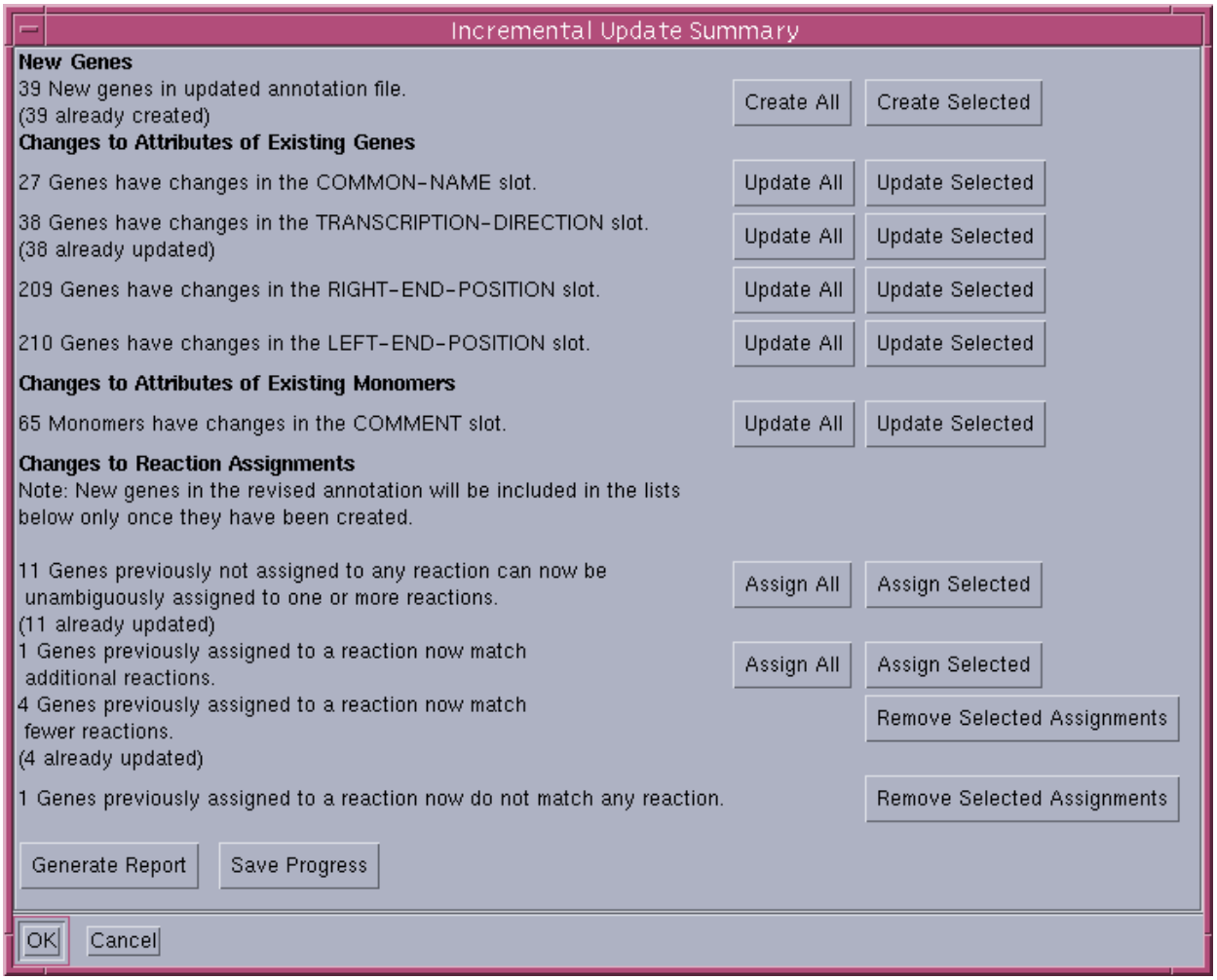


Figure 6.5: The Incremental Update Summary Display

For each set of listed changes, you can either quickly apply them all with a single button push, or you can examine the set of differences and decide in each individual case whether or not the change should be applied. The summary dialog tracks what changes have been applied and counts them as shown. At any point, you can generate a textual

report showing all differences and which have been applied. You may also save your progress and return to it later.

Two examples of dialogs for examining all the changes in a group are shown below in Figures 6.6 and 6.7.

Once all desired changes have been applied, the software invites the user to rescore pathways. This command is described in Section 6.4.3.

6.3.9 Adding or Replacing a Sequence File

An updated annotation is often accompanied by a revised genome sequence. In such cases, it is important to swap in the new sequence file alongside the revised annotation. Additionally, there are times when a PGDB is created without providing the sequence file for one or more chromosomes. When the sequence is finally available, the curator will want to add it to the PGDB so that functionality such as the genome browser or pathway hole filler can be made available. From the Navigator window, the command **Chromosome** → **Add or Replace Sequence File** allows the user to supply a new sequence file for each replicon, which is then incorporated into the PGDB.

6.3.10 PGDB Housekeeping Tasks

The following commands for managing PGDBs are found in the **Organism** menu.

6.3.10.1 Select Organism

If you are creating several new PGDBs simultaneously, you can switch between them using the command **Organism** → **Select**.

6.3.10.2 Reinitialize DB

If you want to delete a PGDB and then reinitialize it so that another automated build can be performed, use the command **Organism** → **Reinitialize DB**. This command should be used with care since it will delete not only the results of the automated build, but also any manual changes you made to the PGDB during the manual polishing phase.

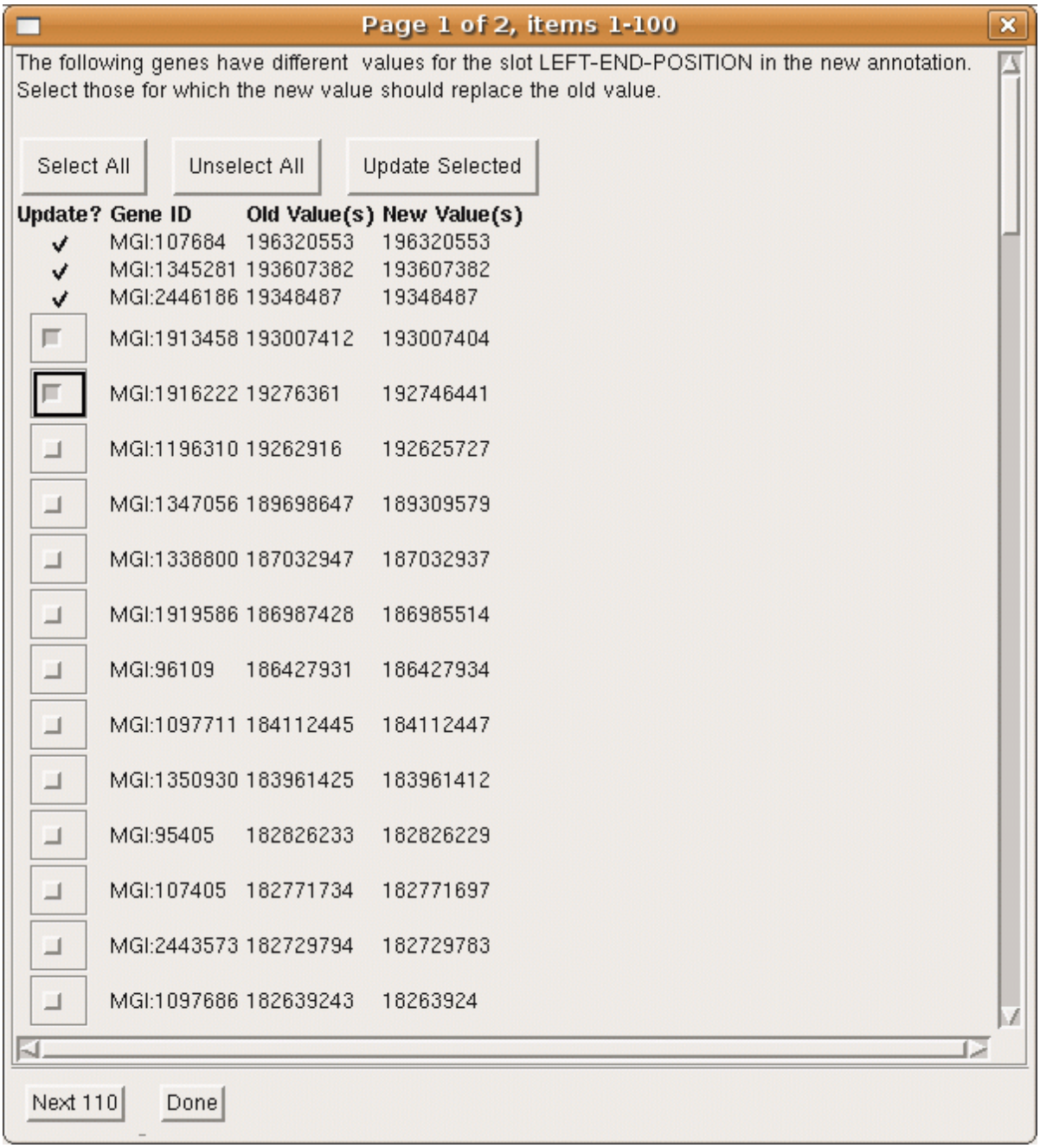


Figure 6.6: A dialog that lists all the genes whose LEFT-END-POSITION slot differs between the revised annotation file and the data in the PGDB. Users check all boxes for which the gene should be updated with the new value. In this example, the first three genes have already been updated.

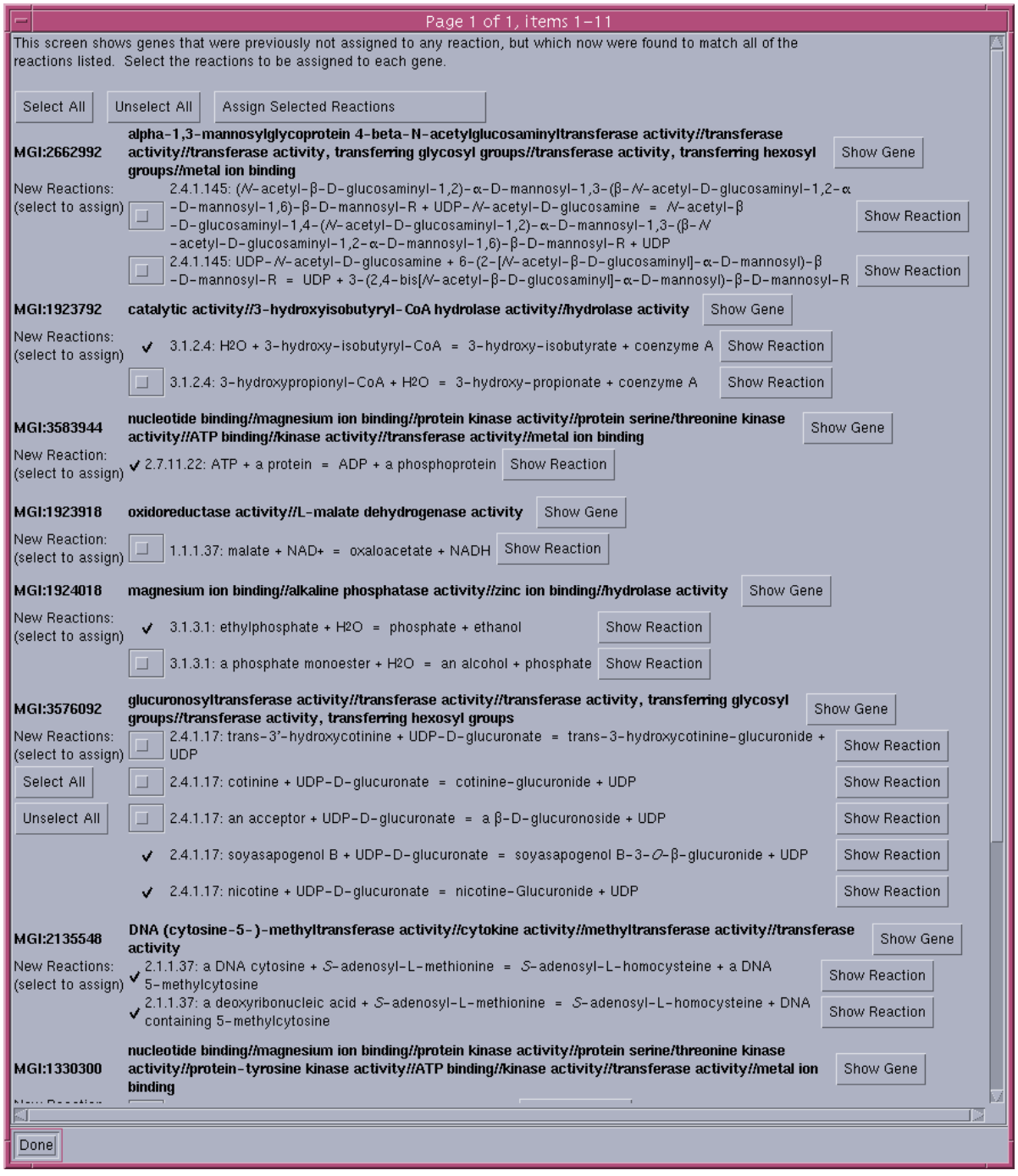


Figure 6.7: A dialog showing potential new reaction assignments resulting from revised function descriptions in the new annotation. Users should select which reactions each gene should be assigned to. The decision is aided by buttons that show either a gene or a reaction in the main Navigator window.

6.3.10.3 Convert File DB to Oracle or MySQL DB

These two commands convert a PGDB from one using a file for storage to one using an Oracle or MySQL database for storage. To use Oracle or MySQL, you must have an Oracle or MySQL RDBMS properly installed and configured at your site, and you must have created a database that has been initialized with the Pathway Tools schema. Instructions for setting up Oracle or MySQL are beyond the scope of this document. For information on installing the Pathway Tools schema, contact ptools-support@ai.sri.com.

6.3.10.4 Backup DB to File

For PGDBs that use an Oracle or MySQL database for storage, this command can be used to create a file version of the PGDB as a backup.

6.3.10.5 New Version

You may wish to freeze a stable version of a PGDB for public use while you continue to edit a development version. This command freezes the current development version by making a copy of the current version. You can designate the new version number.

6.4 Refining the PGDB

The Refinement phase of PathoLogic operation involves additional inference and manual operations that are performed after the Automated Build portion of PathoLogic has completed.

6.4.1 Refine: Assign Probable Enzymes

The command **Refine** → **Assign Probable Enzymes** is used to create additional enzyme-to-reaction assignments that the automated name matching procedure failed to find. It brings up a table of genes and their function names that were classified as probable enzymes during the name-matching process but for which matches were not found. An example table is shown in Figure 6.8. Not all the function names listed will be able to be assigned to reactions. Some may not be metabolic enzymes at all. Others may not be specific enough to be associated with a particular reaction or set of reactions. In addition to making assignments to reactions, this table allows the user to mark certain gene products as unassignable so as to avoid wasting time considering these proteins in the future.

(These proteins will continue to appear in the table for the time being, with an appropriate notation in the Status field, but the next time this command is invoked they will no longer be considered probable enzymes, so will no longer appear in the table.) You can also flag certain proteins for future consideration, and record comments describing, for example, progress made toward identifying the reaction, problems encountered, and the rationale behind particular decisions. If a comment has been recorded for a particular protein, then passing the mouse over that row in the table will cause the comment to be displayed in the lower pane. This enables the user to assess at a glance the status of any particular protein.

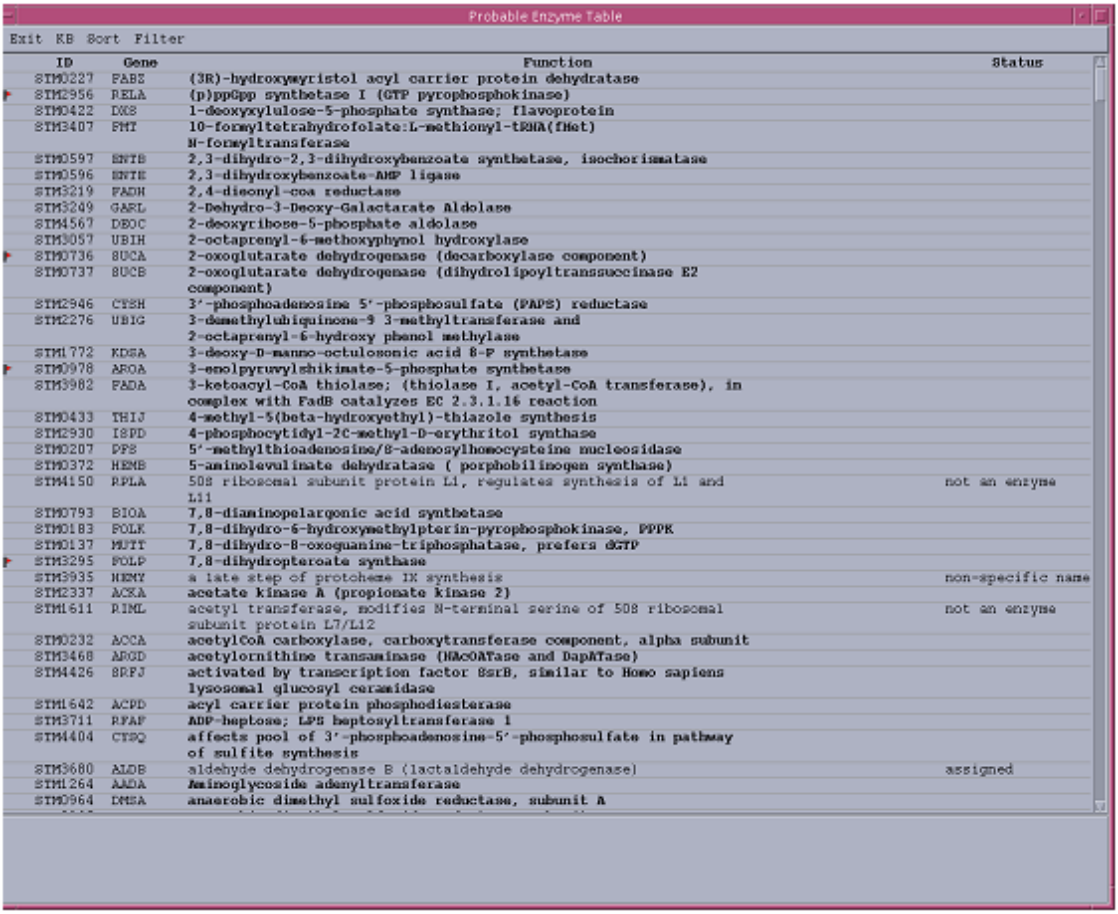


Figure 6.8: Table of unresolved probable enzymes

While working with this table, you user can save the PGDB or revert to the previously saved version at any time by invoking the **KB** → **Save** or **KB** → **Revert** commands. You can change the order in which the rows are presented by specifying a different **Sort** criterion. By default, the rows are ordered by function name, but they can also be ordered by gene ID, name, or chromosomal location. You can also choose whether or not flagged items should be listed at the beginning. If there are many proteins that have been clas-

sified as probable enzymes, the table can be very large. Using the **Filter** command, the user can choose to include, for example, only flagged items, or only function names that contain a particular substring. Alternatively, if the list of proteins in the table is not inclusive enough (e.g., if some of the proteins not identified as probable enzymes can in fact be linked to reactions), the user can use the **Filter** command to select from the full list of unassigned proteins.

To assign a protein to a reaction, or record the other types of information described above, select the row of the table corresponding to the particular function. This will bring up the Probable Enzyme Status Dialog, illustrated in Figure 6.9. This dialog lists the function name and synonyms, if any. You can edit these or supply additional synonyms by clicking on the button. Besides assigning the protein to one or more reactions, you can also enter a comment, as described in the previous paragraph, can flag (or unflag) the protein, or remove it from future consideration for one of the reasons provided.

For some proteins, including the one in Figure 6.9, a list of suggested reactions is provided. These are generally reactions whose enzyme names bear some similarity to the supplied function name (or one of its synonyms), but not enough for the name matcher to have made a definitive assignment. In cases where there are no such reactions, or too many such reactions (which is often the case when a function name is nonspecific), this option is not provided. You should conduct research, as described in Section 6.3.7.1 to attempt to discern which reaction or reactions correspond to the function name. Such reactions could be among those suggested, in which case you should select them and click the **Assign** button, or they could be completely different reactions, in which case you should click the **Assign to Reaction(s)** button. Either of these options will bring up a new dialog for assigning a protein to one or more reactions, shown in Figure 6.10. If reactions were selected from the list provided, then they will already be filled in this dialog. Otherwise, you must type in the reaction ID or EC number. If the reaction to be assigned does not yet exist in MetaCyc, you should first create it using the normal editing tools from the Pathway/Genome Navigator window, and then supply the new ID here. Multiple reactions can be assigned using a comma-separated list. Additional information such as synonyms, a comment, and citations can also be provided as part of this dialog.

Some enzymes catalyze multiple reactions. If all such reactions go by the same function name (for example, if the enzyme is nonspecific and can accept multiple substrates), then the reactions should be specified together in the above dialog, as illustrated. Often, however, each function has its own name and refers to a different catalytic activity. An example is the trpC gene product in *E. coli*, which performs both the phosphoribosyl anthranilate isomerase and the indole-3-glycerol phosphate synthase functions. The PathoLogic file format specifies that each of these functions should be specified on its own line (see Section 6.2.2), but some annotators may disregard this and supply a single function name that refers to both functions. When this happens, use the **Split Function Name into Multiple Functions** button in the Probable Enzyme Status Dialog to specify multiple function names for the protein. Each function name can then be assigned to a

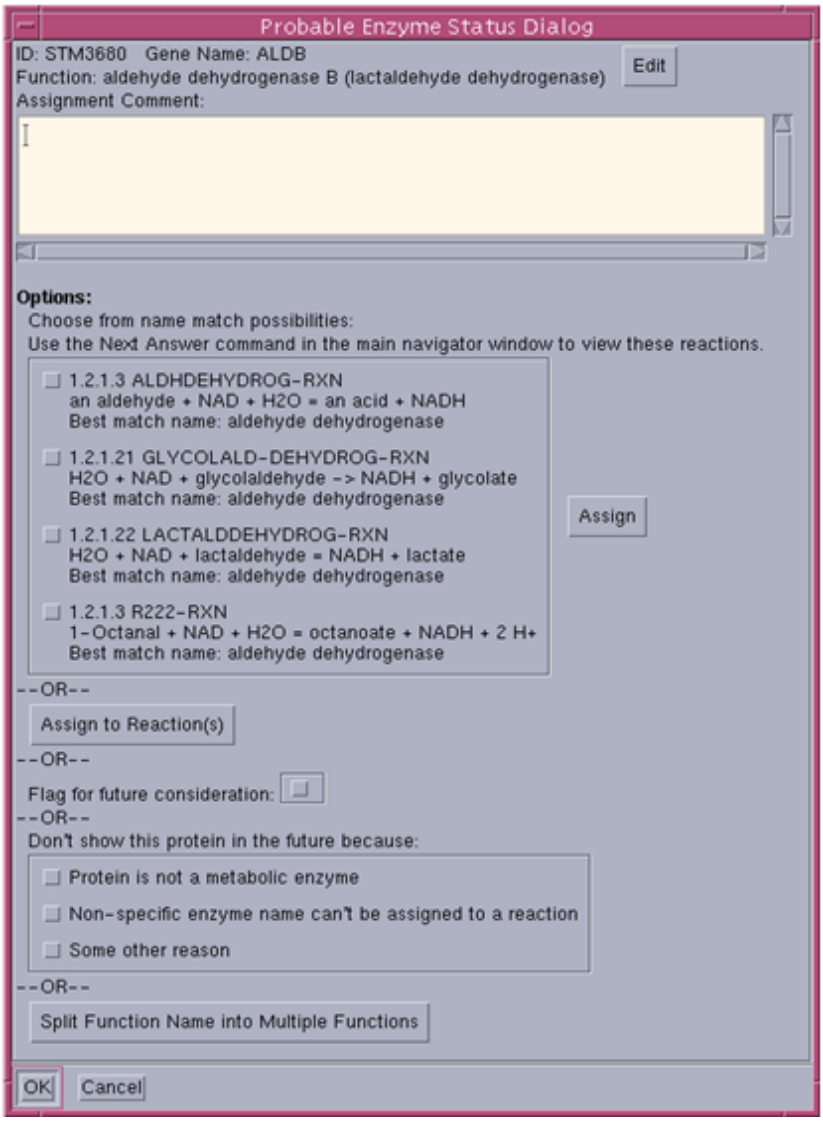


Figure 6.9: Dialog for assigning or classifying a probable enzyme

reaction individually.

As a shortcut for rapidly processing enzymes, one can right-click on a row in the probable enzyme table to access a menu containing items for:

- opening the probable enzyme dialog;
- marking an enzyme as nonspecific;
- marking a protein as not an enzyme;

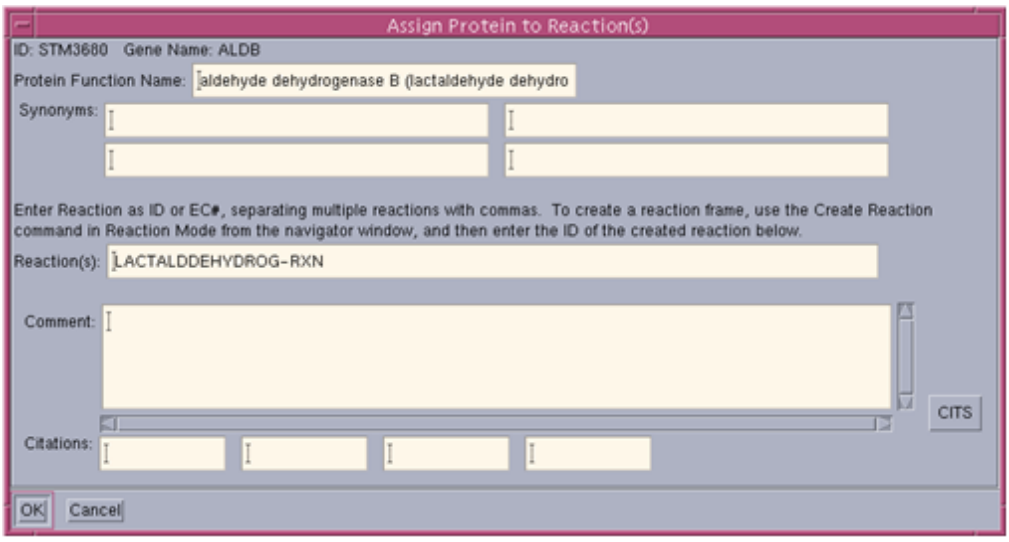


Figure 6.10: Dialog for assigning a protein to one or more reactions

- marking a protein as not to be shown in future sessions in the probable enzyme table;
- flagging or unflagging a protein for future consideration.

6.4.2 Refine: Rerun Name Matcher

If new names or synonyms have been added to MetaCyc or to one of the name files described in Section 6.3.7.1, you may rerun the name matcher after the PGDB has been built to match those additional names. This operation does not alter existing reaction assignments, but may make new assignments.

6.4.3 Refine: Rescore Pathways

The pathway scoring algorithm runs as part of the automated build procedure. However, additional manual steps, such as assigning ambiguous proteins, rerunning the name matcher, or manually adding, deleting or changing protein assignments using the editing tools, may change the set of pathways that would be inferred to be present in the organism. By default, the pathway scoring algorithm uses taxonomic pruning to limit the number of false positive pathways (see the description of the pathway scoring algorithm in Section 6.3.7.2), but the user is given an opportunity to disable taxonomic pruning if desired. Rescoring pathways imports any pathways for which there is new evidence, reimports any pathways that may have changed in MetaCyc, and deletes pathways (and,

when appropriate, their reactions and compounds) that were previously inferred but are now determined not to be present. Pathways that were previously inferred by PathoLogic but which have been manually deleted by the user will not be re-imported unless there is new or different evidence for them. Lists of newly inferred pathways and pathways that the user may now wish to delete, either because there is now insufficient evidence for them, or because they were previously deleted manually (but there is now additional evidence for them), or because they no longer exist in MetaCyc are summarized in a dialog window, shown in Figure 6.11. Users can bring up a table of all pathways in a given category and either show them in the Navigator or mark them for deletion, as desired. See Figure 6.12.

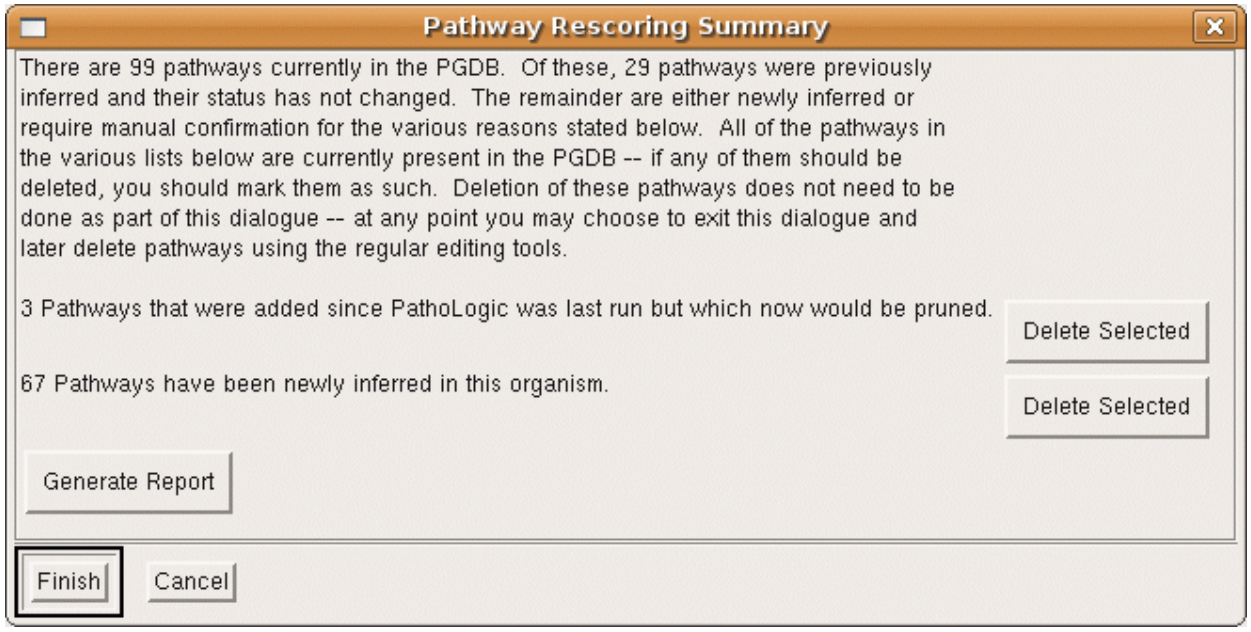


Figure 6.11: A summary of the results of the Rescore Pathways command.

6.4.4 Refine: Create Protein Complexes

A functional enzyme is often composed of several polypeptides (subunits) that aggregate to form a protein complex. To represent this biological situation faithfully, we must create a new PGDB object for the protein complex, and link that object to the PGDB object that represents the reaction that the enzyme catalyzes. However, the PGDB currently links the object for each polypeptide to the reaction it catalyzes, implying that each polypeptide is a functional isozyme. The PathoLogic protein-complex building tool (Figure 6.13), invoked by **Refine** → **Create Protein Complexes**, allows you to specify new complexes that should be built from monomer subunits, and automatically unlinks the monomers from reactions, and links the new protein complexes to appropriate reactions.

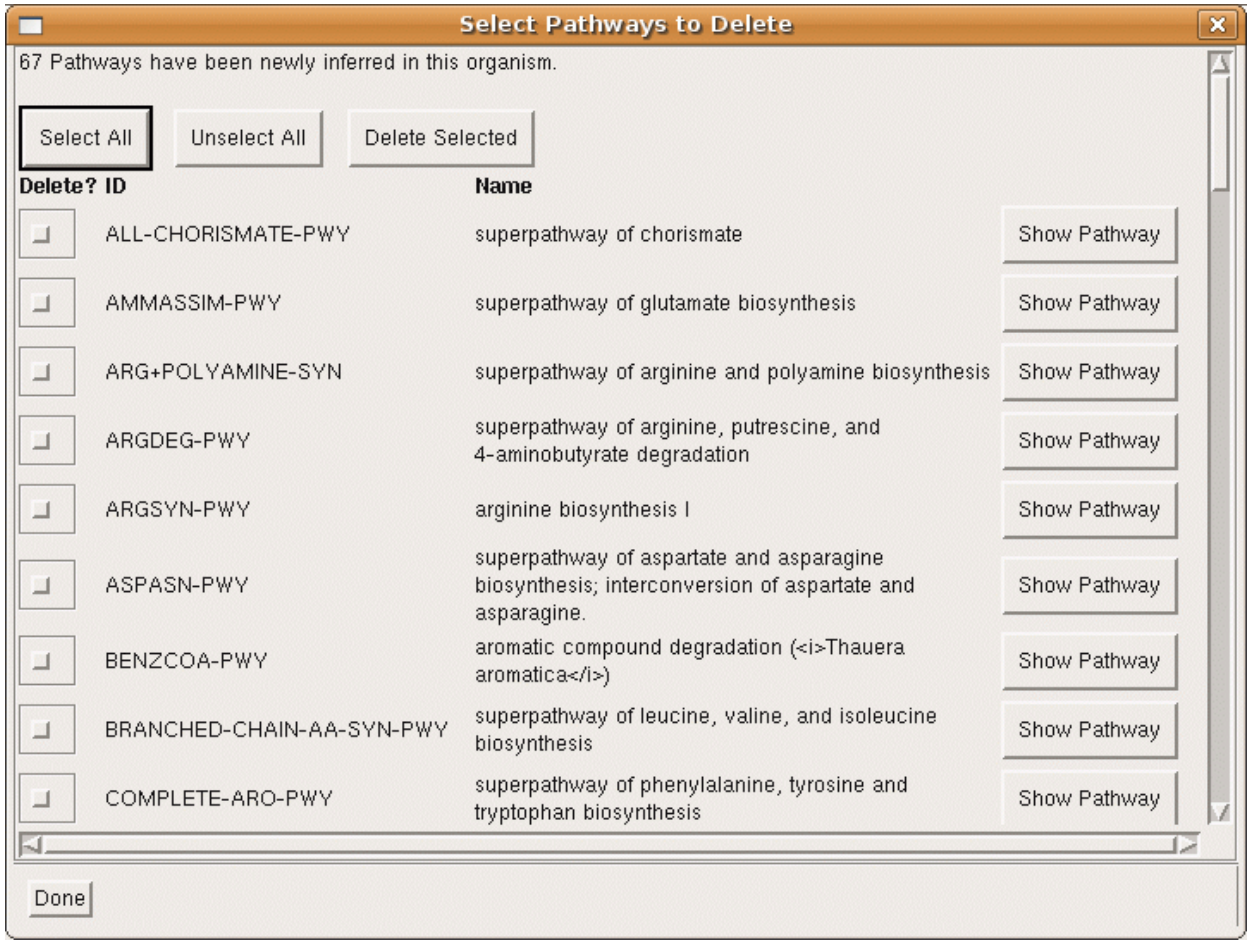


Figure 6.12: Clicking on any Delete Selected button in the Pathway Rescoring Summary will bring up a dialog like this, giving the user the opportunity to examine or delete any or all pathways in a group.

The protein-complex building tool iteratively considers every reaction in the PGDB that has more than one enzyme connected to it via an enzymatic-reaction frame, meaning that more than one protein had a name or an EC number corresponding to that reaction. These proteins must either be isozymes (separate proteins that catalyze the same activity), or subunits of an enzyme complex, or subunits of multiple isozymes for this reaction. For each such reaction, the tool groups together all polypeptides linked to that reaction and presents them to the user. You can scan the groupings to identify those polypeptides that, based on their names, appear to represent subunits of enzyme complexes. To aid you in deciding which subunits might belong within one complex, the Navigator window (which may be hidden behind other windows) displays the MetaCyc version of that reaction so that you can easily inspect the corresponding enzymes in MetaCyc.

You can avoid creating a complex for a group of monomers by clicking the **Skip** button.

Or, you can create one or more complexes. To indicate which monomers should be assigned to new protein complexes, select monomers with the mouse from one of the lists presented. You can use the **Add current history item to Complex** button to add additional monomers in the PGDB to any complex. In order to do that, open the protein to be added in the main display, then switch back to the "Group Subunits into Complexes" window and click the "Add current history item to complex" button.

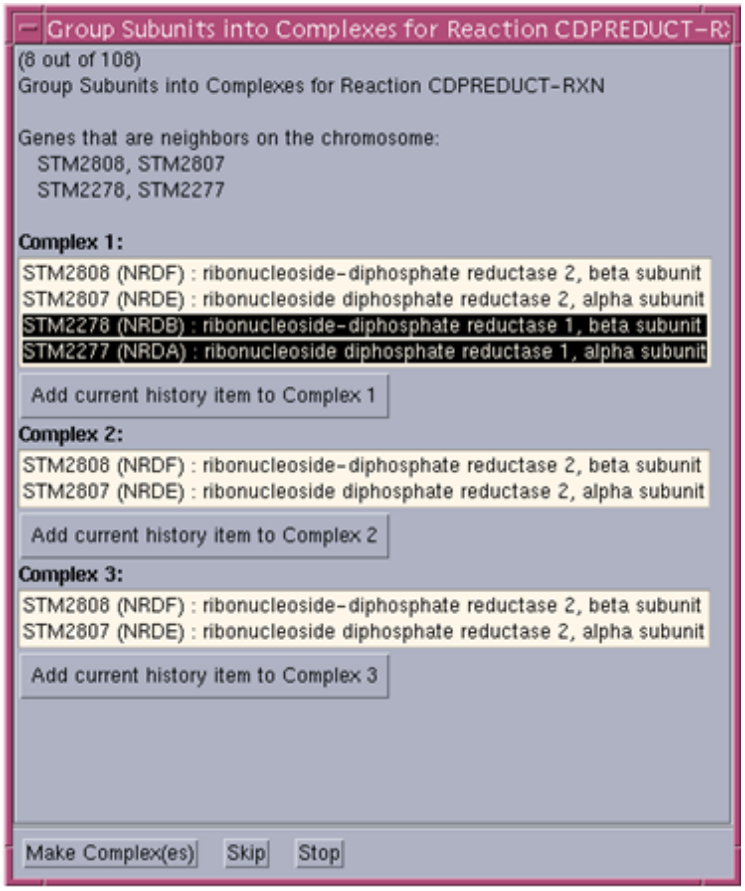


Figure 6.13: Creating protein complexes. The user will create Complex 1 containing the alpha and beta subunits of ribonucleoside-diphosphate reductase 1. In this example, the user should also create Complex 2 by selecting the alpha and beta subunits of ribonucleoside-diphosphate reductase 2 in the Complex 2 menu.

From the names of the monomers, you must make a judgment as to which subunits make up a complex. Typically, the name of the enzyme would both indicate its function and identify which subunit it is within a complex (e.g., "ribonucleoside-phosphate reductase 1, alpha subunit" and "ribonucleoside-phosphate reductase 1, beta subunit").

To create one complex, use the mouse to select the one or more monomers that belong to the first protein complex in the menu under **Complex 1**. To create a second complex,

select the one or more monomers that belong to the second complex in the menu under **Complex 2**.

Clicking on the **Make Complex(es)** button prompts a window (see Figure 6.14) with which you can specify coefficients of the subunit within the complex, in the (somewhat unlikely) case that these coefficients are known.

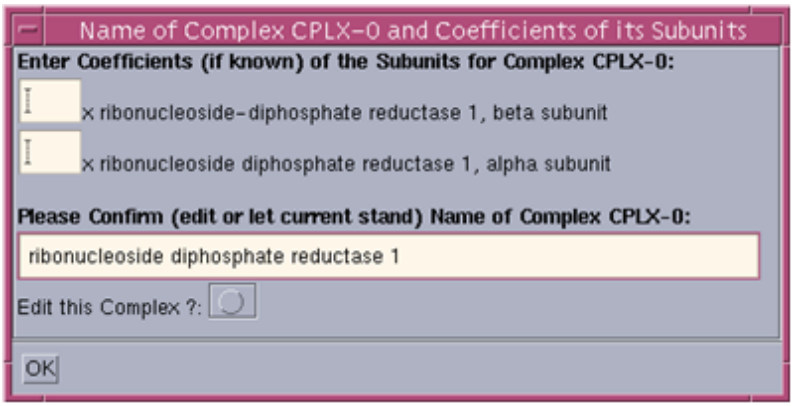


Figure 6.14: Specifying complex subunit stoichiometries

In addition, you may edit the name automatically chosen for the complex. Clicking on the **OK** button now automatically creates a PGDB object for the protein complex, and appropriately links it to other objects in the PGDB.

As shown in Figure 6.13, you have the option to go on to the next potential protein complex (thus creating no new protein complex from the current set of monomers) by clicking the Skip button . Alternatively, you can stop working on complexes (and return to them later) by clicking the Stop button.

6.4.5 Refine: Assign Modified Proteins

Some metabolic reactions use proteins as substrates; that is, proteins become modified by an enzyme-catalyzed reaction (such as phosphorylation or acetylation). The alternative chemically modified forms of a protein are represented in a PGDB as a *base form* and their *modified forms*. The base forms are the products of those genes that encode them. The modified proteins contain a slot, *Unmodified Form*, which points to (holds the ID of) the base forms. The *Modified Form* slot of the base forms gets the ID of the modified species.

When PathoLogic imports a new reaction into the PGDB that has a protein substrate (such as the Acyl Carrier Protein, or ACP), it does not know which gene product within the PGDB corresponds to ACP. The **Refine → Assign Modified Proteins** command creates a dialog with which you can indicate which protein substrates within metabolic reactions correspond to which gene products within the PGDB. PathoLogic then creates a frame for

the modified form of the protein and fills the appropriate slots for it. You are presented with a pane (see Figure 6.15) containing a list of unmodified proteins that are referenced in metabolic reactions that PathoLogic has inferred to be present in the PGDB.

For each protein substrate, PathoLogic presents a list of its best guesses as to which PGDB gene product might correspond to that protein substrate, in a selector button. There is also an empty box where you can enter a gene ID if none of the gene names in the list is correct. Pressing the **Show** button instructs the Pathway/Genome Navigator to display information about the protein substrate that is currently selected with the selector button.

If you are unsure about what gene ID to choose from the selection list and cannot find the appropriate ID by manual investigation, select *leave unconnected* from the list.

6.4.6 Refine: Predict Transcription Units (Operons)

PathoLogic can predict transcription units (TUs) on an entire genome or on genetic elements that you select. In this context, we also call them operons when comprising more than one gene. The predictor produces a set of predicted TUs that encompass all genes in the selected genetic element(s), and generates the corresponding transcription unit frames in the PGDB. Notice that the predicted TU frames will include only include the genes that compose the TU, as the predictor does not generate regulatory sites (transcription start sites, transcription factor binding sites, terminators).

The **Refine** → **Predict transcription units** command creates a dialog (Figure 6.16) that allows you to select one or more genetic elements from those in the current PGDB. You can click on **Select All** to predict TUs on the entire genome. Once genetic elements have been selected, click on **Predict TUs** to initiate the prediction process. Click **Done** when finished.

The TU predictor works by partitioning the genetic element into regions of contiguous genes that are transcribed in the same direction, called directons. A directon starts where two contiguous genes are transcribed in opposite directions (i.e., a directon boundary), and includes all contiguous genes transcribed in the same direction up to the next directon boundary (see Figure 6.17).

A special case is a single-gene directon, that is, a gene that is transcribed in one direction, whereas both its neighboring genes (up- and downstream) are transcribed in the opposite direction. Such a single-gene directon is in fact a single-gene TU. Thus, the TU predictor labels all single-gene directons as single gene TUs.

Longer directons are partitioned into contiguous gene pairs. The TU predictor then proceeds to classify each gene pair into two classes: Either the gene pair is contained within an operon (i.e., a WO, or "Within Operon" pair), or it corresponds to a boundary between adjacent TUs (i.e., a TUB or "Transcription Unit Boundary" pair). The predictor classifies

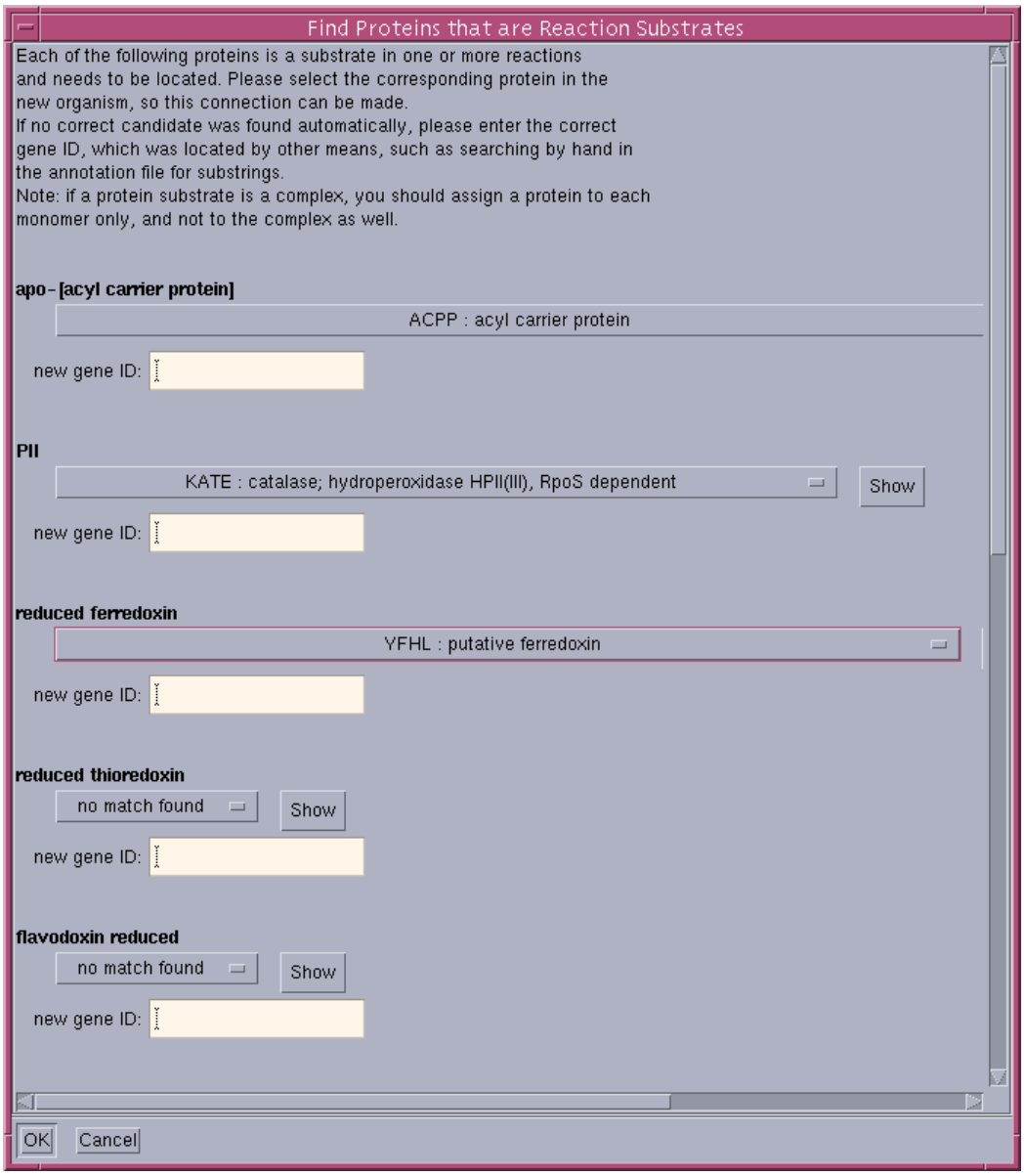


Figure 6.15: Find Proteins that are Reaction Substrates

gene pairs as WO or TUB based on six parameters:

1. Base-pair distance between the genes in a pair
2. Genes' functional classes (if known)
3. Whether or not both genes' products belong to the same protein complex
4. Whether or not both genes' products catalyze reactions in the same pathway

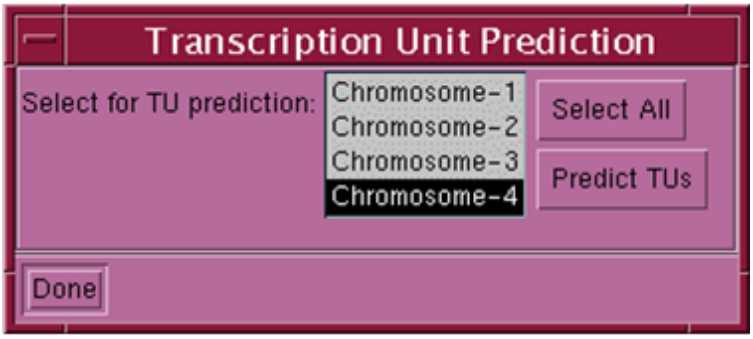


Figure 6.16: Transcription Unit Prediction dialog

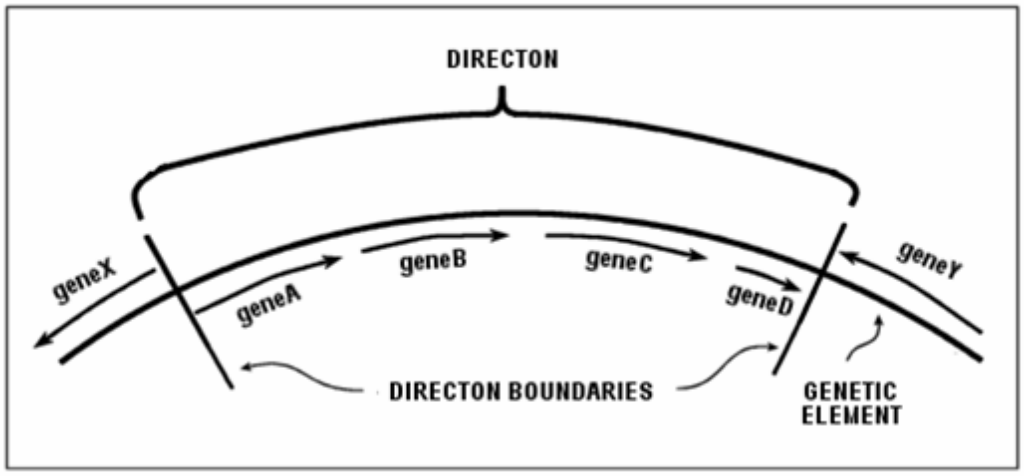


Figure 6.17: Directon. Notice the directon boundaries at the positions where transcription direction changes.

5. Whether or not the product of one gene catalyzes a reaction in a pathway and the other gene's product is a transporter for a substrate in the same pathway

6. Whether or not a gene up- or downstream from the studied gene pair (within the same directon) is related to one of the pair's genes as in 3, 4 or 5 above

Once prediction is carried out on all gene pairs within a directon, the corresponding TUs are assembled out of WO and TUB pairs. Finally, TU frames are added to the PGDB for all predicted TUs, both single-gene directons and TUs predicted within longer directons.

All predicted TU frames will have appropriate Pathway Tools evidence codes identifying them as inferred in an automated fashion (EV-AINF) without manual oversight. If the genes from a predicted TU correspond exactly to those of an existing, non-automatically inferred TU, the software will just add the appropriate AINF evidence code to the ev-

idence already in the evidence slot of the existing TU frame. A new TU frame is not generated in this last case.

The TU predictor deletes all previously predicted TUs for a genetic element, if any, before performing a new prediction. To determine whether a TU frame was predicted and should be deleted, the predictor checks the evidence slot for the EV-AINF evidence codes. If the EV-AINF evidence code for a particular TU is accompanied by other evidence codes, the TU frame is not deleted: only the AINF evidence code is removed from the TU frame evidence slot. This prevents curated TU frames from being deleted in this phase, even if they have also been predicted by the PathoLogic TU predictor.

The predictor sends output to the main PathoLogic window indicating the prediction process's status, and the name of the file where the TU prediction report is going to be saved. The report includes some global statistics on the prediction, like number of TUs and operons predicted, and prints all directons on which a prediction was carried out, including the predicted and known or curated (if any) TUs. For each predicted TU, the name of the gene products of the component genes is also printed. When curated TUs exist in the PGDB, the report also includes some statistics on how well these known TUs were predicted by PathoLogic.

6.4.7 Refine: Transport Inference Parser

The PathoLogic command **Refine → Transport Inference Parser (TIP)** identifies proteins in the current PGDB that are likely to catalyze transport reactions. Such proteins are termed probable transporters. If sufficient evidence is present, TIP infers the transport reaction for the probable transporter, and creates a corresponding reaction frame in the current PGDB. In some cases TIP will also identify multi-subunit transporters, and create protein-complex frames for those transporters in the PGDB. All probable transporters are presented for review in the Transport Inference Parser interface, which allows acceptance, rejection, or editing of inferred transporters by the user.

To identify certain transporters correctly, operons/transcription units must first have been inferred for the PGDB (bacteria only). If not, those transporters that consist of systems of several proteins will be defined incorrectly, with a transport reaction being created for each component monomer of the transport system rather than one reaction for the entire system.

TIP operates by first inferring which proteins are transporters, and then creating reactions and enzymatic reaction frames for the highly probable transporters. The TIP GUI is then presented, allowing en masse acceptance (or rejection) of these transporters, or review and editing of each individual transporter.

6.4.7.1 Transport Inference

The major computational steps performed by TIP in identifying probable transporters and constructing their reactions, enzymatic reactions, and protein complexes are as follows:

1. Find candidate transporter proteins.
2. Filter candidates.
3. Identify substrate(s).
4. Assign an energy coupling to transporter; identify any co-substrate.
5. Identify the compartment of each substrate.
6. Identify transporters that consist of subunits; group into complexes.
7. Construct full reaction including substrates, compartment assignments, and coupling.
8. Construct enzymatic reaction linking each reaction with transport protein.

Each of the steps is described below.

6.4.7.1.1 Find candidate transporter proteins TIP operates through a syntactic analysis of the common name (also referred to as annotation) field of each protein. It does not perform a sequence analysis, but attempts to understand the transport activity already assigned to proteins.

To identify candidate transporters, the common name is examined for certain words indicative of transport activity, such as "transporter", "permease", or "channel". Overly long annotations (currently those longer than 12 words) are excluded.

6.4.7.1.2 Filter candidates A candidate transporter is excluded from further consideration if its annotation exactly matches one of a number of generic annotations that contain no substrate information, such as "putative transporter".

A candidate is also excluded if it contains a counter-indicator word such as "regulator". Such counter-indicators increase the likelihood that the protein is related to a transporter rather than actually being a transporter.

6.4.7.1.3 Identify substrate(s) The annotation of each candidate transporter is searched for the names of transported substrates. This is done by looking up each word pair and individual word, as well as variants thereof, in the MetaCyc KB. When a match is found, that compound is added to the list of possible substrates for that transporter. Processing details include

- Excluding non-substrates that look like compounds such as "as", "be", "c", or "i"
- Name canonicalization, such as stripping plurals
- Handling substrates with affixes such as "-transporting" and "-specific"
- Handling special ionic forms such as "cuprous", "ferric", and "hydrogen"
- Detecting two-word substrates
- Detecting multiple substrates separated by a delimiter such as "sodium/proton"

In general, a protein can have multiple substrates. There are two cases of multiple substrates. The first is when two substrates have been identified, and one substrate is a co-substrate facilitating transport of the other, primary, substrate. In this case, the transporter will have only one primary substrate, and one reaction.

In the second case, more than one substrate has been identified, but none are co-substrates. In this case, the transporter has N substrates and N different reactions.

Note that certain multivalent elemental substrates such as iron and cobalt may be incorrectly identified as their elemental rather than their ionic form.

6.4.7.1.4 Assign an energy coupling to transporter; identify any co-substrate The energy coupling mechanism used by the transport protein is represented by a controlled vocabulary indicating the subclass of Transport-Reactions in which the reaction associated with a transporter will be assigned. The terms in that controlled vocabulary are:

UNKNOWN: Transporters of Unknown Mechanism

MECHANICAL: Mechanically driven Transporters

LIGHT: Light-driven Transporters

ELECTRON: Electron-flow-driven Transporters

DECARB: Decarboxylation-driven Transporters

PTS: PEP-dependent Transporters

ATP: ATP-driven Transporters

SECONDARY: Secondary Transporters

CHANNEL: Channel-type Facilitators

An energy coupling is assigned to each transporter as follows:

- For a small number of substrates (for example: enterobactin, glycerol) a particular coupling mechanism is inferred.
- Sometimes an annotation contains an indicator word for a coupling mechanism (for example: "ATP" "channel" "permease").
- If neither of the above is successful, the coupling mechanism is UNKNOWN.

If the energy coupling is SECONDARY, an attempt is made to determine the co-substrate that enables the transport. The co-substrate is transported in the same direction (for symporters) or in the opposite direction (for antiporters) as the primary substrate(s). The co-substrate is assigned as follows:

- If H^+ (proton) has been identified as a substrate, use it as the co-substrate; it is removed from the (primary) substrates of the transporter.
- If Na^+ has been identified as a substrate, use it as the co-substrate; it is removed from the (primary) substrates of the transporter.
- If there are exactly two substrates, choose one arbitrarily as the co-substrate; it is removed from the (primary) substrates of the transporter. The reaction that will be created is independent of this decision. An incorrect guess of the primary may, however, lead to some confusion when results are reviewed.
- If only one substrate has been identified, H^+ is assumed to be the co-substrate.

6.4.7.1.5 Identify the compartment of each substrate Currently, it is assumed that primary substrates are transported either into or out of the cytosol. It is further assumed that they cross the membrane closest to the cytosol. Simple inference is done to determine this compartment. For gram-negative bacteria, this compartment is the periplasmic space. For gram-positive bacteria, it is the extracellular space.

Representation of cellular compartments in PGDBs is currently evolving, and may change in the near future.

The default assumption is that the primary substrate is transported into the cytosol. If the indicator words "export", "uptake", or "efflux" are present in the annotation, transport in the opposite direction is inferred.

If the energy coupling of the transporter is SECONDARY, both the primary substrate and the co-substrate change compartments. If the indicator word "symport" or "symporter" is present in the annotation, it is inferred that they move in the same direction, that is, they start in the same compartment. If the indicator word "antiport" or "antiporter" is present, it is inferred that they move in opposite directions.

6.4.7.1.6 Identify transporters that consist of subunits; group into complexes Certain transporters consist of systems of proteins, rather than individual proteins. For such transporters, an attempt is made to identify the component proteins of the transporter and to group them into a complex. This complex is added to the KB, and any reactions and enzymatic reactions for the transporter are associated with the complex, rather than with its components.

Proteins are grouped into a complex if the following conditions all hold:

- The energy coupling is ATP or PTS.
- The same substrate has been identified for each.
- The genes of all such proteins share the same operon/transcription unit.

Note that if there are no operons in the organism, no subunits are identified. For that reason, this step is skipped if the organism cannot by its nature have operons. Furthermore, if operons/transcription units have not been predicted for the PGDB, many monomer proteins are likely to be erroneously identified as individual transporters. Hence it is important to run the **Refine → Predict transcription units** command, described in the preceding section, before transport prediction.

6.4.7.1.7 Construct full reaction including substrates, compartment assignments, and coupling For each primary substrate of this transporter, either import a reaction (from EcoCyc) that conforms to its substrates and compartments, or create a new one.

The full transport reaction consists of the primary substrate plus any auxiliary substrates indicated by the coupling, as well as their compartments. For example, ATP transporters have ATP and water on the left side and ADP and phosphate on the right.

If a single transport reaction from EcoCyc matches the full reaction, it is imported. If multiple reactions match, they are retained for possible import later on but not yet imported. If no imports are found, a new reaction is created.

The semantics of matching require that each substrate be present in the reaction, and that its inferred compartment is identical. Since different organisms have different cellular compartments, reactions that are analogous but that do not match exactly with respect to compartments will not be imported.

Note that EcoCyc is used as the reference PGDB for imports. Organisms with different cell compartments such as gram-positive bacteria will import no reactions; all inferred reactions will be newly created.

6.4.7.1.8 Construct enzymatic reaction linking reaction with transport protein Each reaction constructed is associated with its protein by constructing an enzymatic reaction frame that links them together. A history string and evidence code are added to track how and why it was created.

6.4.7.2 Reviewing and modifying TIP results

When TIP inference completes, all transporters that have been identified are displayed in a table, as shown in Figure 6.18. This display includes the review status of the transporter, the gene and annotation of the protein, and the inferred substrate, coupling, and reaction if present.

To view additional information about a transporter, pass the mouse over its row in the table. The protein frame ID is shown, along with the number of substrates of multi-substrate transporters and brief explanations of certain inferences.

A single transport protein may in general have several substrates, and several different transport reactions. In this case, each substrate is shown on a separate row of the table. These multi-substrate transporters have the same gene name and annotation. When you pass the mouse over its row, the explanatory text at the bottom of the table displays the number of substrates.

The initial table display shows both high- and low-probability transporters. Low-probability transporters have no reaction frames associated with them. Low-probability transporters include those for which no substrate was identified, and hence no reaction was created. If desired, these may be filtered out using the **Filter** command. Filtering by status is also supported. It is generally worthwhile to review the low-confidence transporters. Generally they are low-confidence because no substrate is identified. Sometimes this is because of a gap in our KB of chemicals, a typographical error, an alternate spelling, etc., in which case you may be able to manually identify the substrate and create a reaction with the reaction editor.

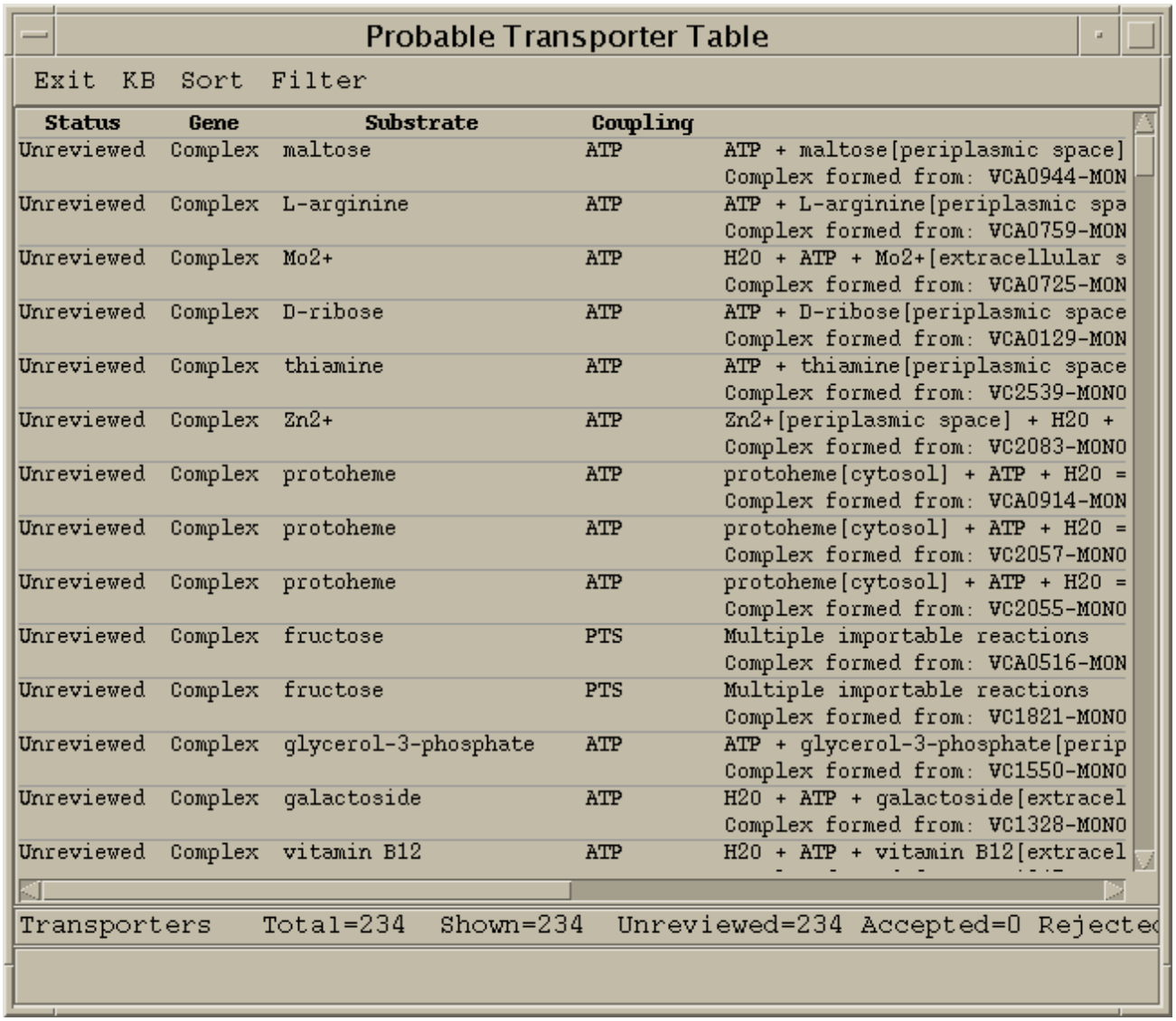


Figure 6.18: Results of transport inference, in tabular form

6.4.7.2.1 Main display The Status column indicates whether that transporter has been marked as reviewed by the user, as well as indicating if the transporter has been accepted, rejected, or whether the decision to accept or reject has been deferred.

6.4.7.2.2 Individual transporter dialog To review an individual transporter in detail, left-click on its row in the table display. A dialog box similar to the one shown in Figure 6.19 appears.

As many as four exit boxes are shown at the bottom of the dialog. If a reaction has been

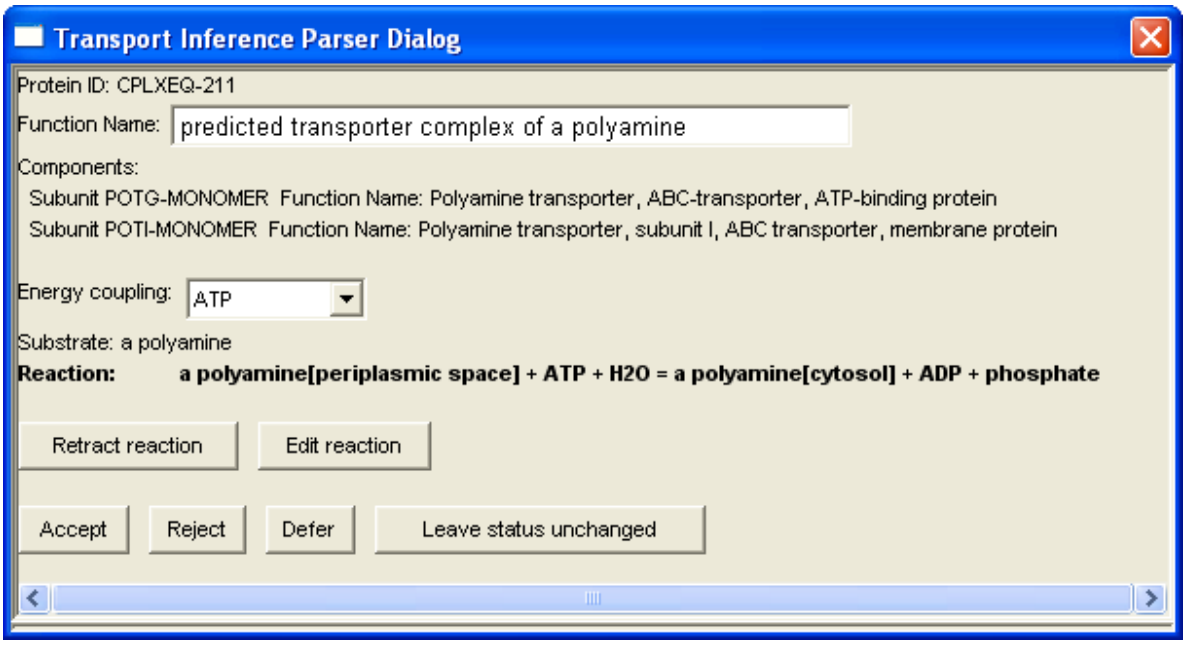


Figure 6.19: Dialog for review of an individual transporter

created for the transporter, the Accept and Reject commands are present. Clicking **Accept** causes the inferred reaction and enzymatic reaction to remain in the PGDB; if the PGDB is saved, it will be incorporated into the PGDB. Clicking **Reject** causes the reaction and enzymatic reaction to be retracted from the PGDB; regardless of whether the PGDB is saved, these will not be incorporated into the PGDB. Clicking **Defer** simply changes the status indicator in the table display; it is useful in conjunction with the **Sort** and **Filter** commands for grouping transporters. Clicking **Leave status unchanged** does simply that.

The annotation of the protein is displayed and may be edited. The inferred energy coupling is also shown, and may be changed. The assigned coupling determines the subclass of the transport reaction as noted above. The primary substrate is displayed and may not be edited. However, the **Edit reaction** button allows arbitrary changes to the reaction, including changing the primary substrate.

If a single reaction is shown, transport inference was able to determine an unambiguous reaction for the transporter, either by creating it from scratch or by importing it from EcoCyc.

If multiple imports are shown, two or more EcoCyc reactions conform to the substrates and energy coupling that have been identified. In this case, one of the reactions may be selected. When an import is selected, the **Import reaction** button appears; clicking it causes the reaction to be imported, and an enzymatic reaction associating it with the transport protein to be created.

If no reaction is shown, then either no substrate was identified, or evidence was sufficiently ambiguous to preclude inferring a reaction.

Clicking the **Edit reaction** button allows modification of the inferred reaction, or creation of a reaction if none is present. See Section 7.3.7 for details.

6.4.7.2.3 Saving and exiting

There are four commands that exit TIP:

Exit → Exit, rejecting all unresolved predictions. All frames that were created for unresolved transporters (those with status of DEFERRED or UNREVIEWED) are retracted from the PGDB. TIP is then exited without saving the PGDB. To preserve all previously ACCEPTED changes, the PGDB must be saved at a later time.

Exit → Exit, accepting all unresolved predictions. All frames that were created for unresolved transporters (those with status of DEFERRED or UNREVIEWED) are accepted into the PGDB as is. TIP is then exited without saving the PGDB. To preserve all ACCEPTED changes, the PGDB must be saved at a later time.

KB → Save KB & Exit, rejecting all unresolved predictions. All frames that were created for unresolved transporters (those with status of DEFERRED or UNREVIEWED) are retracted from the PGDB. The PGDB is then saved; this preserves all previously ACCEPTED changes.

KB → Save KB & Exit, accepting all unresolved predictions. All frames that were created for unresolved transporters (those with status of DEFERRED or UNREVIEWED) are accepted into the PGDB as is. The PGDB is saved; this preserves all ACCEPTED changes.

Each of these commands shows a summary of PGDB changes that have been made and the number of transporters that have been neither accepted nor rejected. Note that if the PGDB is not saved, either here or subsequently, no transport information will be saved to the PGDB, regardless of what has been accepted or rejected. In other words, **accepting a transporter alone is not sufficient** to update the PGDB.

6.4.7.3 Repeated Invocations of TIP

TIP inference excludes from consideration any protein that already has at least one transport reaction associated with it. Therefore, if TIP is run repeatedly, any transporters that are accepted and saved will not be re-inferred.

Note that as of this time, rejecting a transporter does not reject it permanently; subsequent invocations of TIP will re-infer it.

6.4.7.4 Suggestions for Writing Annotations

TIP relies on the annotation of the protein for most of the inferences it makes. In particular, it assumes that conventions commonly used in genome centers have been used in writing the annotation. While TIP attempts to be robust in its identification of transport proteins, substrates, etc., following certain conventions helps ensure that TIP will recognize transporters properly and create the desired reactions.

Annotation analysis it case-insensitive, so case may be mixed freely. Annotations should be no longer than twelve words. Words connected with slashes (e.g., "sodium/proton") or hyphens are counted as a single word.

The simplest annotation is

"*substrate keyword*"

where *keyword* indicates transport function. For example:

"glucose transporter"
"probable Fe+3 transport"

are simple examples that indicate transport function. Besides indicating that a protein is a transporter, certain words can indicate transport function as well as polarity — whether the transported substrate moves into or out of the cytosol. The default polarity is into the cytosol; it may be made explicit with "import", "importer", or "uptake". Transport out of the cytosol is indicated by "export", "exporter", or "efflux":

"potassium uptake protein"
"capsule polysaccharide exporter, inner-membrane protein ctrc"
"cation efflux system protein, putative"
"heme exporter protein d"

Avoid using these counter-indicator or ambiguity keywords in the annotation, which will cause TIP to ignore the protein:

"resistance"
"regulator", "regulation", "regulatory"
"repress", "repressor", "suppressor", "suppressor"
"nuclear-export"
"resembles"
"helicase"
"receptor"
"nuclear-export"

Simple annotations suffice but are noncommittal with respect to the transport coupling mechanism. If the coupling mechanism is known, it is preferable to express it in the annotation. Conventions for doing so depend on the mechanism.

Channel transporters. Indicating a channel transporter can be done with the word "channel":

"mg+2 channel"
"voltage-gated chloride channel"
"putative calcium channel"

Phosphotransferase transporters. Phosphotransferase (PTS) transporters are indicated by either "pts" or "phosphotransferase":

"sugar pts system, eiia component"
"phosphoenolpyruvate-protein sugar phosphotransferase"

ATP transporters. ATP transporters are indicated by the keywords "atp", "atp-dependent", "atp-binding", "atpase", "abc", "abc-type", "abc-2", "f-type", "v-type", or "p-type". This indicator alone is not sufficient; a separate indicator indicating transport function is also required:

"putative oligopeptide abc transporter"
"atp binding cassette Fe+2 transporter"
"atp-dependent Fe+2 transporter"
"d-allose transport atp-binding protein alsa"
"cadmium, zinc and mercury transporting atpase"
"Cu+2-transporting p-type atpase"

Secondary transporters. In general, a secondary transporter involves a cosubstrate that facilitates the transport of the primary substrate(s) as well as its polarity. In addition to import and export, polarity for secondary transporters may indicate whether the primary and cosubstrate move in the same (symport) or opposite (antiport) direction across the transport membrane.

A secondary transporter can be indicated to be simply a carrier (i.e., noncommittal about its cotransporter and polarity) with "carrier", "facilitator", or "permease":

"citrate carrier/transporter"
"glycerol uptake facilitator protein"
"2-keto-3-deoxygluconate permease"

In the absence of indicators, the cosubstrate is assumed to be a proton, and the polarity is assumed to be symport.

Symporters are indicated with "symport" or "symporter". Antiporters are indicated with "antiport", "antiporter", "exchange", or "exchanger":

"l-rhamnose:proton symport"
"probable cadaverine/lysine antiporter"
"putative na(+)/h(+) exchanger yjce"

TIP uses several rules to infer the cosubstrate from the annotation. Some genome centers join the cosubstrate to the primary substrate with a colon:

"sodium:galactoside family symporter"
"aspartate:alanine antiporter"

Others join the substrates with slashes or hyphens:

"sodium/pantothenate symporter"
"arabinose-proton symporter"

Any of these conventions is suitable for TIP. For antiporters, the substrate that is transported out of the cytosol should be last. This rule is not followed universally by genome centers; TIP assumes this as an alternative to making a completely arbitrary decision.

The primary substrate or substrates must be expressed in the annotation, or TIP will not be able to infer a transport reaction for the transporter. TIP uses the MetaCyc Chemical Compound ontology to identify substrates, so using any name or synonym from MetaCyc is generally acceptable.

Multiple substrates. Many transporters have multiple primary substrates. In these cases, one reaction per primary substrate is created by TIP. To identify multiple substrates, it is best to connect the substrates with slashes, the convention used by most genome centers:

"threonine/serine transporter"
"d-serine/d-alanine/glycine transporter"

Alternatively, the substrates may be separated with "and":

"cadmium, zinc and mercury transporting atpase"

Classes of chemicals. A substrate may be nonspecific in that it may refer to a class of chemicals rather than a single chemical:

"sugar efflux transporter"
"heme exporter protein a"
"branched-chain amino acid transport system ii carrier"
"putative cation efflux system protein cusa"

Two-word substrates. Substrates comprising two words are allowed. They may refer to either a specific chemical, or to a class of chemicals:

"ferric enterobactin transport protein fepe"
"amino acid antiporter"

Substrates of three or more words are not recognized. Note that for

"branched-chain amino acid transport system ii carrier"

the substrate identified is "amino acid".

Elements. Element names and chemical symbols (e.g., "magnesium", "Mg") are permitted in lieu of the specific valence ("Mg+2"). The latter is much preferred as it is unambiguous, but if the element name is definitive with respect to valence, it is acceptable:

"mg(2+) transport atpase, p-type 1"
"potassium efflux system kefa"
"high-affinity zinc uptake system membrane protein znub"

TIP currently does not recognize many multivalent elements like "copper", "iron", and "cobalt"; it does recognize "ferrous", "ferric", "cuprous", and "cupric":

"ferrous iron transport protein b"

Suffixes. The qualifying suffixes "-transporting", "-specific", and "-like" are permitted on substrates; some or all of multiple substrates may have a suffix:

"pts system, trehalose-specific iibc component"
"pts system, maltose and glucose-specific iiabc component (pseudogene)"
"cation-transporting p-type atpase, putative"

6.4.7.5 TIP Report

In the reports directory for the organism, a file `transport-summary.txt` is produced each time TIP is run on that organism. The file contains three lines of text for each probable transporter:

<Status> <Enzyme> <Primary Substrate> <Coupling>

<Annotation/Common Name>

<Reaction>

Each field contains the text that is shown in the corresponding field of the TIP GUI. For example, the reaction field is either the reaction text itself, "No reaction", or "Multiple importable reactions".

If a probable transporter has multiple primary substrates, a separate entry is shown for each.

The Status field is the status of the probable transporter when TIP was exited, one of ACCEPTED, REJECTED, DEFERRED, or UNREVIEWED.

6.4.8 Refine: Pathway Hole Filler

6.4.8.1 Overview of the Pathway Hole Filler Algorithm

A pathway hole occurs when, based on PathoLogic's predictions, the organism's genome appears to lack one or more enzymes required to complete a pathway. The activity of each pathway hole is known from the pathway inferred by PathoLogic. We use a set of isozyme sequences encoding the required activity in other genomes to search for candidate enzymes in the genome of interest. After identifying candidate sequences, our program uses a Bayes classifier to evaluate each candidate. Rather than evaluating the candidates based solely on their similarity to the set of search sequences, we determine the probability that the candidate has the desired function. Our classifier considers evidence from the homology search (e.g., E-values, alignment lengths, and the rank of the candidate in the BLAST output) from the pathway context of the missing reaction (e.g., is the candidate gene adjacent to a gene catalyzing an adjacent reaction in the pathway?) and operon-based data (i.e., is the candidate gene likely to appear in an operon with another gene in the pathway?).

A more detailed description of the Pathway Hole Filler is available at `http://www.biomedcentral.com/1471-2105/5/76`.

6.4.8.2 Pathway Hole Filler Operation

6.4.8.2.1 BLAST Installation Please consult `http://bioinformatics.ai.sri.com/ptools/installation-guide/released/blast.html`.

6.4.8.2.2 Overview The Pathway Hole Filler runs in two phases: training and prediction.

Training: In the training phase, the known reactions in the PGDB are used to train the Bayes classifier to be able to predict which candidates have the desired function and which do not. The training phase gathers evidence about the known reactions in the PGDB: e.g., given that gene A catalyzes the reaction adjacent to the reaction catalyzed by gene B in a pathway, what is the probability that genes A and B are neighboring genes in the genome? what is the distribution of the rank in the BLAST output of genes assigned to reactions in the PGDB? The classifier can then compare the evidence for a particular candidate sequence to the prior evidence gathered about sequences known to have the correct function, and predict the probability that that candidate has the desired function.

The PHFiller requires a list of known reactions in the PGDB from which to gather the data needed for training. Using a larger dataset for training should produce more robust results in the prediction phase; however, gathering a larger dataset will require more

time to run the PHFiller. Given a random selection of reactions, the specific reactions selected should not impact the accuracy of the prediction phase; hence, which reactions you choose are most likely less important than the number of reactions chosen. Gathering training data for 100 reactions takes about 2 hours (depending on the number of isozyme sequences available for the reactions selected) and should scale approximately linearly with the number of reactions.

Prediction: After the training data have been gathered, the program will make predictions for holes in pathways in the PGDB. The default setup groups the missing reactions by pathway and makes predictions for all pathways with holes. Since making predictions for all pathways with holes may take several hours, you may run the PHFiller in stages. To do this, select any subset of pathways with holes from the list provided by the interface. The program may also be configured to search for a list of reactions that you select, rather than searching for holes in a list of pathways.

6.4.8.2.3 Using Gene Expression Data with the Pathway Hole Filler Many studies have demonstrated the utility of gene expression data in identifying genes that operate together in a specific pathway. Thus, we have extended the PHFiller to include correlations across gene expression profiles as a source of evidence for filling pathway holes. We refer users to Karchenko et al., 2004, and Novak and Jain, 2006, for examples of the use of gene expression data in filling pathway holes.

Expression Dataset File Format

To use gene expression data with the PHFiller, you must load a data file of the appropriate format. Expression data is imported from a file that is provided by the user and is stored on the user's computer. Each line of the file contains data for a single gene, and is of the form

<geneID><data-column 1> ... <data-column N>

Columns are separated by the `Tab` character. The first column contains the unique identifier of a gene. The gene IDs from sequencing projects are generally acceptable and unambiguous.

The numbers in the data columns should be relative data values and must be supplied as such (i.e., the program will not compute relative values from the supplied data file). Each column represents the relative expression level in a particular experiment or at a particular time point. An expression profile (a row of data for a gene) must contain at least two columns, but for comparisons between profiles to be meaningful there must be adequate variation in gene expression levels over a large enough number of columns. Missing values in the expression profile are ignored.

Lines that start with either '`#`' or '`;`' are taken to be comments and are ignored by the program. The software uses the first row of data (i.e., the first line that is not a comment

line) to determine the number of data columns to process. Thus, even if not all fields for the first row contain data, you must make sure that it contains the appropriate number of `Tab` characters.

Usage

To load gene expression data for use by PHFiller, select **Refine** → **Pathway Hole Filler** → **Load gene expression profile data**. A dialog box pops up from which the user may select the appropriate file. Click on the **Select Data File** button to select a file containing the expression data. The **Help** button describes the required file format.

There may be a pause while the data is read and processed. The number of lines read successfully from the file will be reported in the PathoLogic window. A temporary file is generated on your computer and will be removed after processing is complete. Once processing is complete, you may run the PHFiller as usual.

6.4.8.2.4 Operation of the Pathway Hole Filler To identify missing enzymes in the genome predicted to fill these pathway holes, use the command **Refine** → **Pathway Hole Filler** to access the PHFiller.

The PHFiller can be operated in one of three modes:

Fully Automatic: The entire hole-filling procedure is carried out with no interaction required from the user.

Wizard: The user is guided through each step of the hole-filling procedure before the program is run.

Power User: The user must initiate each step of the hole-filling process by selecting the appropriate command from the Pathway Hole Filler menu.

In each of these modes, the program runs in three distinct steps as listed in the Pathway Hole Filler menu:

- Step I: Prepare Training Data
- Step II: Identify and Evaluate Candidates
- Step III: Choose Holes to Fill in DB

The operation of each step described below applies to both the Wizard and the user-driven modes. In Wizard mode, the interface guides you through the necessary selections for Step I and Step II, these two steps are executed, and then the Wizard prompts you to make the required selections for Step III. In the user-driven mode, each step is preceded

by dialog boxes where you must make selections. For the Fully-Automatic mode, the program uses the default options when your selections are indicated.

In the first step, "Prepare Training Data", data are extracted from the known reactions (reactions to which an enzyme has been assigned) in a PGDB. You must select from which PGDB to extract the training data (the default is the currently selected PGDB). If training data for that PGDB have been generated previously, you must decide whether to use the existing data or generate new data. Likewise, if training data have been generated for another organism, you may select that PGDB from the list and use the data to train the Bayes classifier for any other organism you wish to predict. When generating new training data, you must also decide which known reactions in the PGDB to extract data from (the default is to use all known reactions in the PGDB). The flowchart in Figure 6.20 also describes the process for preparing training data.

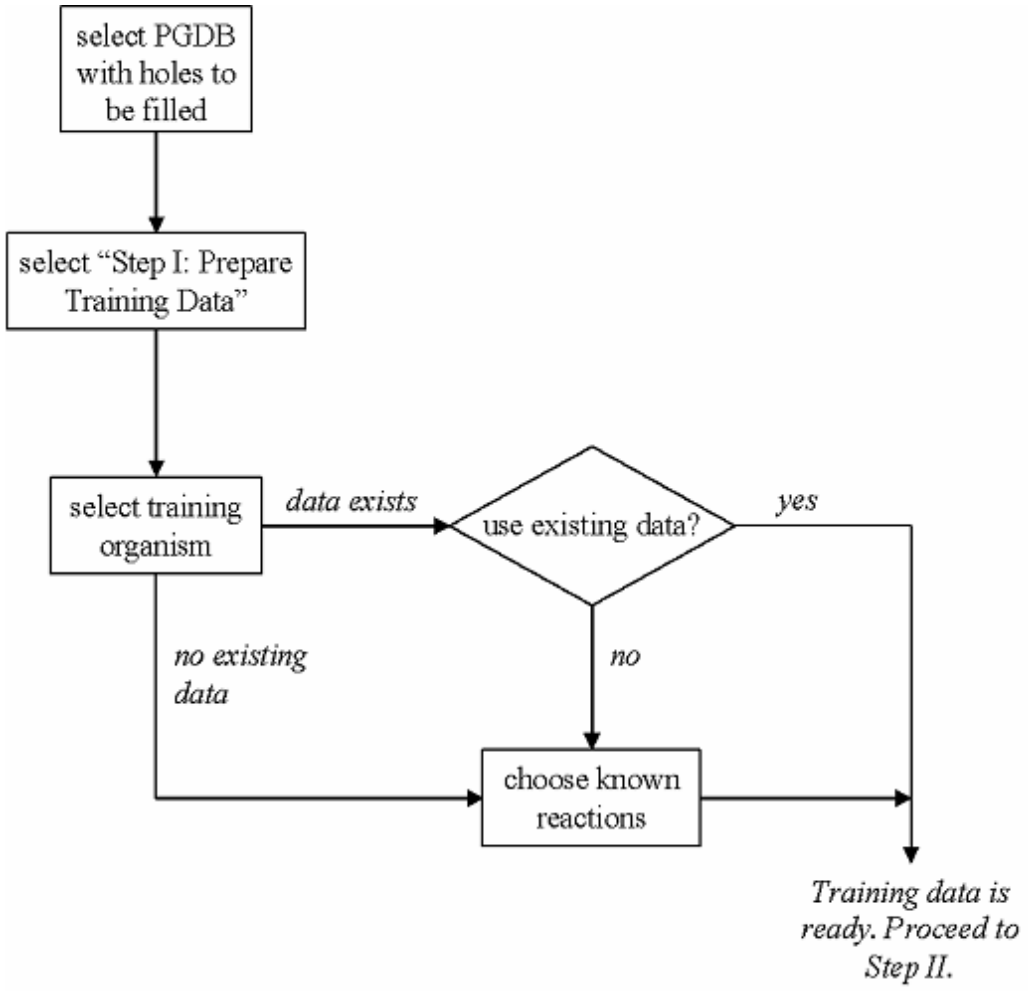


Figure 6.20: Flowchart for Step I: Prepare Training data

The second step, "Identify and Evaluate Candidates", requires that you select the set of

pathway holes that the program will attempt to fill (the default is to search for holes in all pathways in the PGDB). The program can search for holes in all pathways with holes, holes in a list of selected pathways, or a list of selected holes (reactions).

In the last step of filling pathway holes, "Choose Holes to Fill in the DB", you can specify which pathway holes to fill with which candidate identified in the second step of the process. The first window that opens after selecting Step III from the Pathway Hole Filler submenu displays a list of the top-scoring candidates for pathway holes where the probability computed for the candidate is above the user-specified cutoff (Figure 6.21). Enter any value between 0 and 1 in the box next to the button labeled "Update display for new cutoff". After you change the value and click this button, only candidates with probability greater than or equal to the specified cutoff will be displayed.

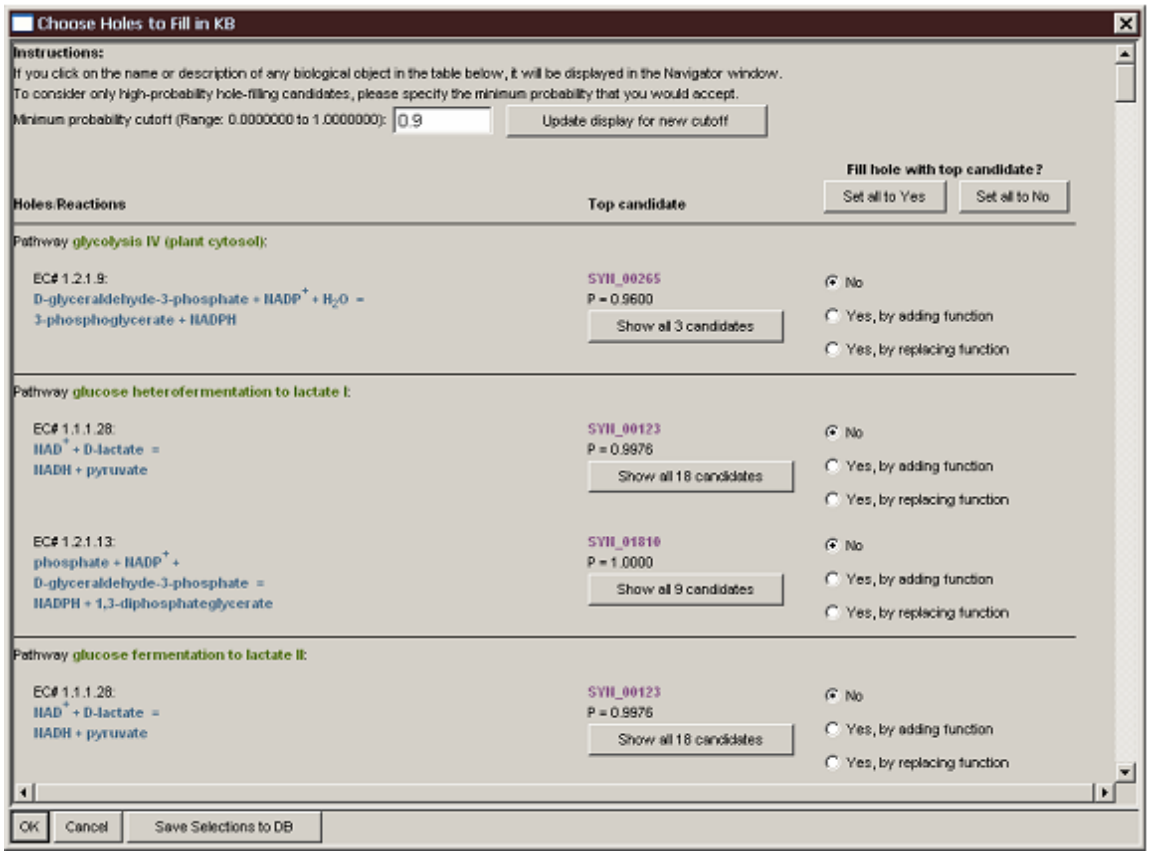


Figure 6.21: "Choose Holes to Fill in KB" page in the Pathway Hole Filler. In this case, the user selected a list of pathways for hole filling during Step II of the process. If the user had selected a list of reactions for hole filling, the pathway subheadings in the first column would be eliminated.

The candidates above the cutoff appear in a table with three columns: "Holes/Reactions", "Top candidate", and "Fill hole with top candidate?". If Step II was executed using a list of pathways with holes, all candidates for a pathway will be grouped together in the list.

For each pathway hole, the actual missing reaction and its EC number (if any) are displayed in the first column. The top-scoring candidate, its computed probability, and any existing functional annotation for the candidate appear in the second column. The last column holds a radio-button list with which you may specify the fate of the top candidate for each pathway hole. If you wish to fill a pathway hole with the top-scoring candidate, make the appropriate selection in column 3. There are three options:

No Do not fill the pathway hole with this candidate.

Yes, by adding function Fill the pathway hole with this candidate. Assign the function of the pathway hole/missing reaction to the candidate in addition to any currently known function (i.e., you believe the candidate may be a multifunctional protein).

Yes, by replacing function Fill the pathway hole with this candidate. Assign the function of the pathway hole/missing reaction to the candidate, replacing any currently known function (i.e., you believe the originally assigned function is incorrect).

The second column includes a button labeled "Show all X candidates", where X is the number of candidates identified in Step II. Clicking this button will open a second window titled "Candidates to fill pathway hole: ..." This window displays detailed evidence for each candidate for filling the pathway hole, as well as additional information describing the pathway hole and the candidate itself. Each column contains data for one candidate. Refer to Figure 6.22 and Table 6.1 for a description of each data item in the page. To fill a pathway hole with a particular candidate enzyme, select the appropriate radio button in the "Fill hole?" row. The data displayed in this page can be very helpful in choosing which pathway holes to fill.

Table 6.1: Description of fields in the "Candidates to fill pathway hole: ..." window.

Key	Description
A	The missing reaction that may be catalyzed by the candidates shown in the table.
B	A list of all pathways in which this missing reaction appears.
C	Toggles the definitions for the table rows on and off.
D	Candidate hole filler is the putative enzyme identified by the Pathway Hole Filler to potentially catalyze the reaction shown at A. This row in the table shows the protein frame name, the currently assigned function of the protein (if any), and a button to move the candidate to the last column in the table.

Table 6.1: (continued)

Key	Description
E	The "Fill hole?" row holds a radio button list where the user may specify the fate of the candidate with respect to the pathway hole. There are three choices: **No:** Do not fill the pathway hole with this candidate. **Yes, by adding function:** Fill the pathway hole with this candidate. Assign the function of the missing reaction (A) to the candidate in addition to any currently known function (i.e., you believe the candidate may be a multifunctional protein). **Yes, by replacing function:** Fill the pathway hole with this candidate. Assign the function of the missing reaction (A) to the candidate, replacing any currently known function (i.e., you believe the originally assigned function is incorrect).
F	The "Gene" row includes the name of the gene, which encodes the candidate enzyme, and the common name of the gene, if any.
G	"Probability" is the probability computed by the Pathway Hole Filler that this candidate enzyme catalyzes the missing reaction (A).
H	"Candidate's Gene-Reaction Schematic" displays the gene-reaction schematic for the candidate gene in the organism where pathway holes are being filled (i.e., the current DB). The diagram will display any functions that will be replaced if you choose to fill the pathway hole "by replacing function".
I	"Average rank" is the average rank of the candidate enzyme sequence in the BLAST output lists (e.g., if a candidate is the best hit in each search, the average rank for the candidate is 1).
J	"Best E-value" is the E-value for the best alignment of the candidate with a query sequence.
K	"Shotgun score" is the number of query sequences whose BLAST output included the candidate sequence.
L	"Average fraction aligned" is the average of each BLAST alignment length normalized by the length of the query sequence.
M	"Adjacent reactions?" shows a list of reactions adjacent to the pathway hole for which the gene encoding the product that catalyzes the reaction is adjacent to the candidate in the genome.
N	Is the candidate gene in a "direction" with another gene in the same pathway? We define a "directon" as a contiguous series of genes transcribed in the same direction.
O	The user may enter a history note to associate with the enzymatic reaction created by filling the hole.

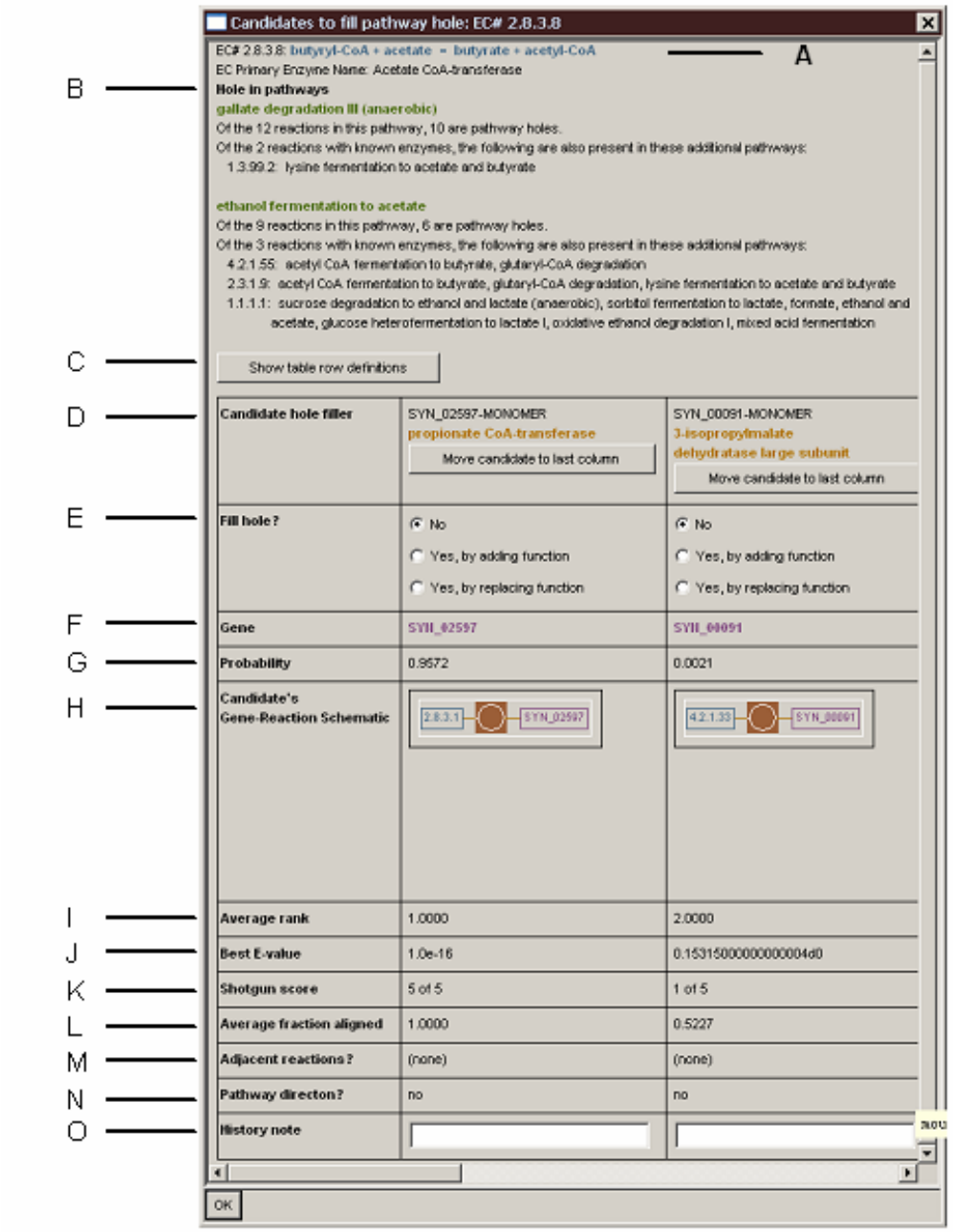


Figure 6.22: "Candidates to Fill Pathway Hole" page from the Pathway Hole Filler

Like the previous window, each candidate may be used to fill a pathway hole. The same radio button list is included in row E, "Fill hole?" of this window. The difference now is that any number of candidates may be selected to fill the pathway hole. This feature may prove to be useful, for example, for cases where multiple isozymes appear to catalyze

the same reaction, or when a reaction might be catalyzed by an enzyme complex. If the PHFiller identifies multiple subunits of the complex, you can assign each subunit to the reaction.

After making the appropriate selections in the "Candidates to fill pathway hole: ..." window, click the "OK" button in the lower left corner of the window to return to the "Choose Holes to Fill in DB" window. Any selections that were made in the previous window will be reflected in the column labeled "Top-scoring candidate". A list of genes encoding the selected candidates will be shown under the heading "Other candidates already selected". After you click the "OK" button in the "Choose Holes to Fill in DB" window, the selected changes will be made in the DB.

6.4.8.2.5 Deciding When to Fill a Pathway Hole For each pathway hole, PHFiller provides one or more candidate enzymes to fill the hole, the probability computed for each candidate, and a wealth of evidence supporting the identification of each candidate enzyme. If you choose to manually review the candidates to identify which enzyme(s) to assign to each pathway hole, the evidence provided can be used to help make the decision.

For each pathway hole, PHFiller provides a list of candidate enzymes. You can reject all candidates predicted for a particular pathway hole, or accept one or more of the candidates to fill the pathway hole. When you fill a pathway hole, the function of the pathway hole may be assigned to the candidate enzyme by choosing either **"Yes, by adding function"** (which will add the activity of the pathway hole to the candidate's existing activity) or **"Yes, by replacing function"** (which will replace the candidate's existing activity with the activity of the pathway hole).

Here, we discuss three sources of information to consider (in addition to your own knowledge of biology and metabolism) to decide whether to fill a pathway hole with a particular candidate:

1. Evidence supporting the prediction of the candidate as an enzyme catalyzing the pathway hole
2. Evidence supporting the prediction of the pathway in the organism database
3. Candidate's original assigned activity

We provide several rules of thumb to consider in the process of selecting enzymes to fill pathway holes from the list of candidates predicted by PHFiller. These "rules" should not be thought of as an algorithm for deciding which pathway holes to fill; each individual prediction has its own unique combination of evidence and background knowledge (e.g., the TCA cycle is well studied; glucose heterofermentation to lactate less so). All these

factors should be considered collectively to determine when to accept a prediction made by PHFiller.

Would the assignment of the candidate to the pathway hole be consistent with the MetaCyc Gene-Reaction Schematic?

The Gene-Reaction Schematic (Figure 3.3) depicts the relationships among a set of genes, enzymes, and reactions. At the top right of the **Candidates to fill pathway hole** page, the Gene-Reaction Schematic depicting the relationship between the pathway hole (highlighted) and any other reactions also catalyzed by the enzyme catalyzing the pathway hole reaction in the MetaCyc DB is shown.

For instance, Figure 6.23 shows the top half of the Candidates to fill pathway hole: EC 1.2.1.13. The MetaCyc Gene-Reaction Schematic indicates that the enzyme (a homotetrameric complex) catalyzing EC 1.2.1.13 also catalyzes EC 1.2.1.59 and EC 1.2.1.12. Figure 6.24 shows the first two columns in the bottom half of the same page. The candidate, SYN_01810-MONOMER, is already assigned to EC 1.2.1.12. Hence, the evidence from MetaCyc that the same enzyme that catalyzes EC 1.2.1.12 also catalyzes EC 1.2.1.13 supports the choice of SYN_01810-MONOMER to fill this pathway hole. To add the EC 1.2.1.13 function to the enzyme choose "**Yes, by adding function**" in the **Fill Hole?** row.

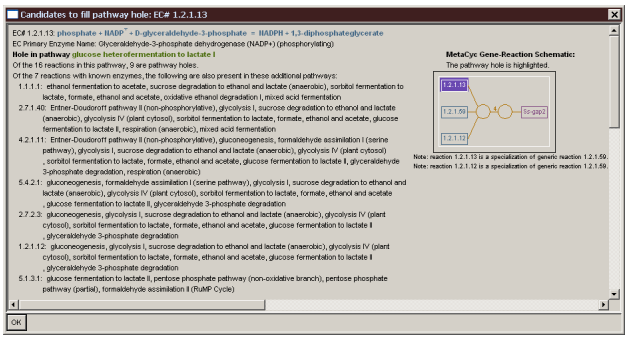


Figure 6.23: Top half of the "Candidates to Fill Pathway Hole" page from the Pathway Hole Filler

How likely is it that the predicted pathway(s) containing the hole are used by the organism?

Figure 6.23 shows that EC 1.2.1.13 appears in one pathway in this database, "glucose heterofermentation to lactate I". The pathway comprises 16 reactions; nine of these reactions are pathway holes, indicating that the pathway prediction could be a false positive. Additional information is also provided to determine how many of the seven reactions with known enzymes also appear in other pathways (with the assumption that these other pathways might better "explain" the presence of the reactions in the database). Under "Of the n reactions with known enzymes, the following are also present in these additional pathways:" Figure 6.23 shows that each of the seven reactions with known enzymes also

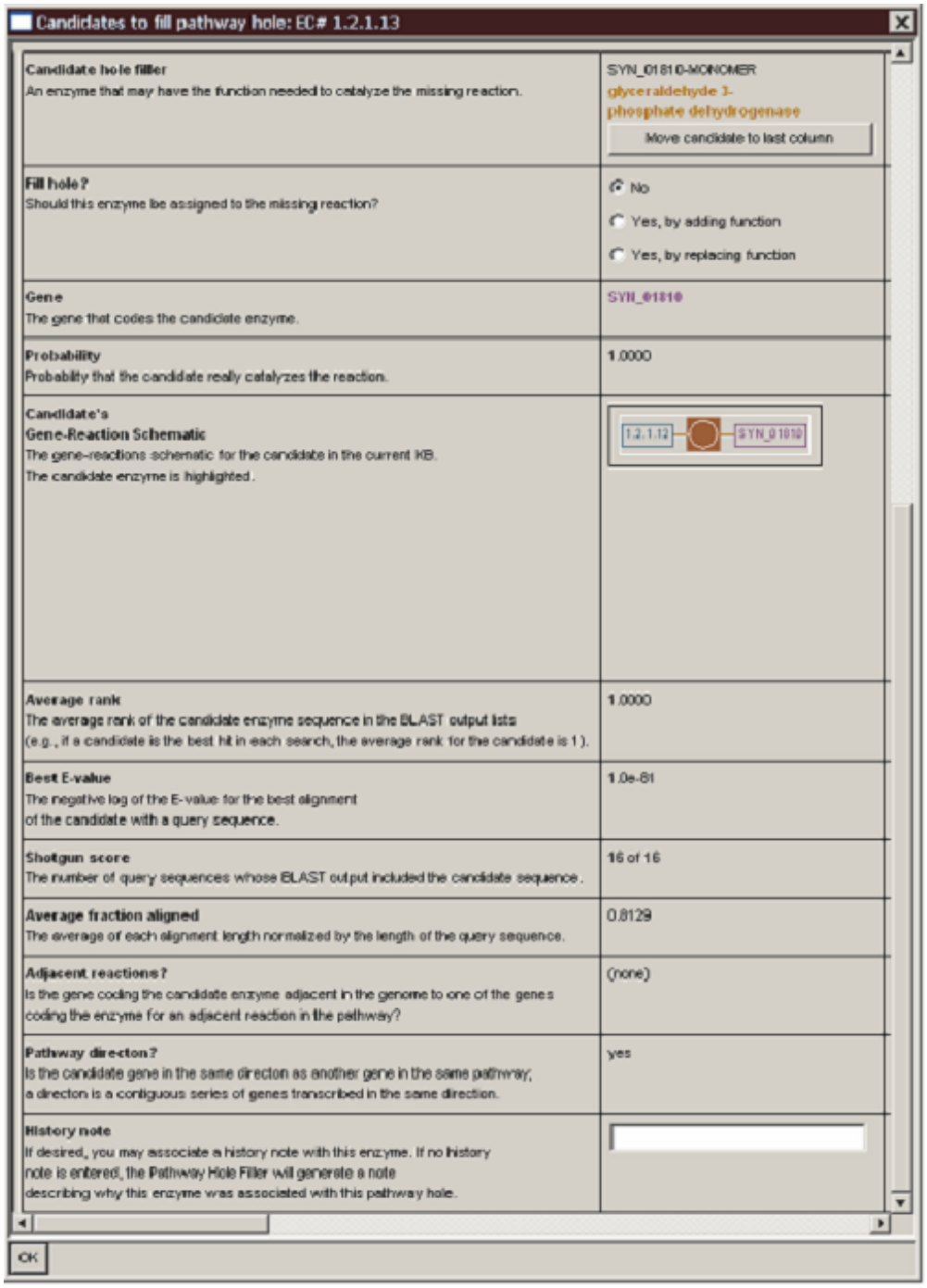


Figure 6.24: Bottom half of the "Candidates to Fill Pathway Hole" page from the Pathway Hole Filler.

appears in other pathways. Therefore, we might infer that this pathway is not present in this organism.

In this particular example, however, given the strong evidence supporting the second function for the candidate, we would recommend accepting the assignment of the candidate to the pathway hole, but would also recommend further investigation to determine if the pathway should be kept or deleted.

Is the existing activity of the candidate consistent with the pathway hole activity?

In the example presented in Figure 6.23, the correct choice of candidates to fill the pathway hole is almost obvious (notwithstanding the ambiguous prediction of the pathway itself). The evidence suggests that the highest-scoring candidate catalyzes two reactions: EC 1.2.1.12 and EC 1.2.1.13. Because we have inferred that the candidate is a multifunctional enzyme, we recommended selecting **"Yes, by adding function"** to fill the pathway hole.

But often the candidate's original annotation, which was used by PathoLogic to infer the set of reactions and pathways present in the organism, is inconsistent with the pathway hole for which the candidate has been identified. In these cases, there are several questions to pose that might help the reviewer come to a decision about whether or not to make the assignment.

Is there any evidence that the original annotation is incorrect?

In some cases, the reviewer may be able to determine that the original annotation is incorrect, thus lending further credence to the assignment of the candidate to the pathway hole. Let's say a candidate, C, for pathway hole H, was already assigned to a reaction, R, by PathoLogic based on its original annotation. If R is not part of any pathway in the database or if R is part of a pathway that is unlikely to occur in the organism (for instance, if a plant pathway is inferred for a prokaryote), the original annotation of C may be incorrect, thus increasing the belief that C actually catalyzes reaction H. In this case, the user should fill the hole, H, by selecting the "**Yes, by replacing function**" radio button.

Is the candidate's originally assigned activity in the same protein family as the enzyme activity that would catalyze the pathway hole reaction?

If the originally assigned function for the candidate protein C appears to be a member of the same protein family as an enzyme that would catalyze the pathway hole reaction, PHFiller's prediction may be a false positive. In this case, homology alone may not be sufficient to determine a likely function for protein C. The prediction might be justified, however, if another source of information supports its conclusion: Is the gene for C adjacent to a gene coding an enzyme that catalyzes an adjacent reaction in the pathway (the "**Adjacent reactions?**" row in the table), or is the gene for C in the same directon as another gene in the pathway (the "**Pathway directon?**" row in the table)? Is the original assignment for C not part of any pathway, or part of a pathway with very weak evidence? Unless pathway or operon context indicates that the candidate C may in fact catalyze the pathway hole reaction, you should probably not fill the hole with candidate C.

Is the original annotation a nonspecific function, like "hydrolase family protein" or "hypothetical protein"?

In this case, filling the pathway hole with the candidate enzyme will provide a valuable prediction of the specific function of the protein, by replacing the noninformative annotation of the protein with the reaction description (select "**Yes, by replacing function**").

6.4.9 Refine: Update Cellular Overview

The command **Refine** → **Update Overview** constructs a Cellular Overview diagram for the PGDB, taking into account any changes to reactions or pathways since the overview was last generated. The new overview diagram is saved automatically. Updating the overview diagram takes roughly one hour to complete.

6.4.10 Refine: Run Consistency Checker

The command **Refine** → **Run Consistency Checker** invokes the consistency checker graphical user interface on the current PGDB. This tool automatically corrects some data constraint violations, and flags other violations for manual review by the user. See Section 3.10.8.13 for more details on the operation of the consistency checker. We strongly recommend that the consistency checker be run soon after a PGDB is initially built, and after extensive refinements to or curation of the PGDB have been performed.

6.5 Output from PathoLogic

6.5.1 Pathway/Genome Database

The principal output as a result of running PathoLogic is the subject organism pathway/genome database.

6.5.2 Pathway Predictor Summary Pages

Following construction of a pathway/genome database for a subject organism (O), you can view a series of reports that will guide you in interpreting the results of the metabolic pathway prediction, and in manually removing false-positive pathway predictions. PathoLogic is tuned to err on the side of bringing more possible pathways to the

user's attention, which does generate more false-positive pathway predictions. The pathway summary pages are designed to provide you with information that can be used to identify and remove false-positive pathway predictions.

These reports can be viewed within the desktop software, and can be saved in HTML files that may be viewed via a Web browser. To view the reports, select the **Summary of Organisms** command in the Navigator menu. Select the newly created PGDB, click on the button **Summary of Pathway Evidence**, and select the desired report. To generate an HTML file for one of these reports, right-click on the report name in the index page, and select **Save Page as HTML**. These reports provide a convenient summary of the output created by the Predictor and are grouped into two categories:

1. Pages summarizing evidence for the presence of pathways in O
2. Page listing reactions missing from O that would complete its partial pathway.

The pathway evidence pages are grouped into five distinct sets based on pathway function and/or the $X|Y|Z$ score assigned by the Predictor to each pathway. (See Section 6.3.7.2 for the definition of this score as well as a description of how evidence is assigned.) One grouping is of pathways by functional category. Within each category pathways are ranked on the basis of assigned scores (i.e., ranked on the basis of the numerical value, Y/X, from largest to smallest). An example of this page is shown in Figure 6.25. A second grouping is of all pathways ranked by Y/X values (largest to smallest). The remaining three groupings represent a partitioning of all pathways based on pathway score, $X|Y|Z$. The first group is of pathways probably present in O, where *probably* means there is evidence for 50% or more of the steps of each pathway in this group, i.e., $Y/X > 0.5$. The second group is of pathways each of which have evidence for at least one step but less than 50% of the reactions in the pathway, that is, $0 < Y/X < 0.5$. These are pathways deemed as being possibly present in O. The last grouping is of pathways probably not present in O. All pathways in this group have assigned Y values of 0.

A "pathway glyph" is included for each pathway entry on these pages, which summarizes graphically the evidence for each step in the pathway. The glyph shows the sequence of steps in the pathway, with reaction steps drawn in different colors depending on whether or not there is evidence for catalysis of the reaction in O, and whether or not the reaction step is unique to the pathway. The meaning of the colors is explained at the end of the **Contents** section of the page.

6.5.3 Interpretation of the PathoLogic Summary Pages

For interpretation of the PathoLogic Summary Pages, we first present a set of guidelines for pathway listings and associated pathway evidence scores. They provide a framework

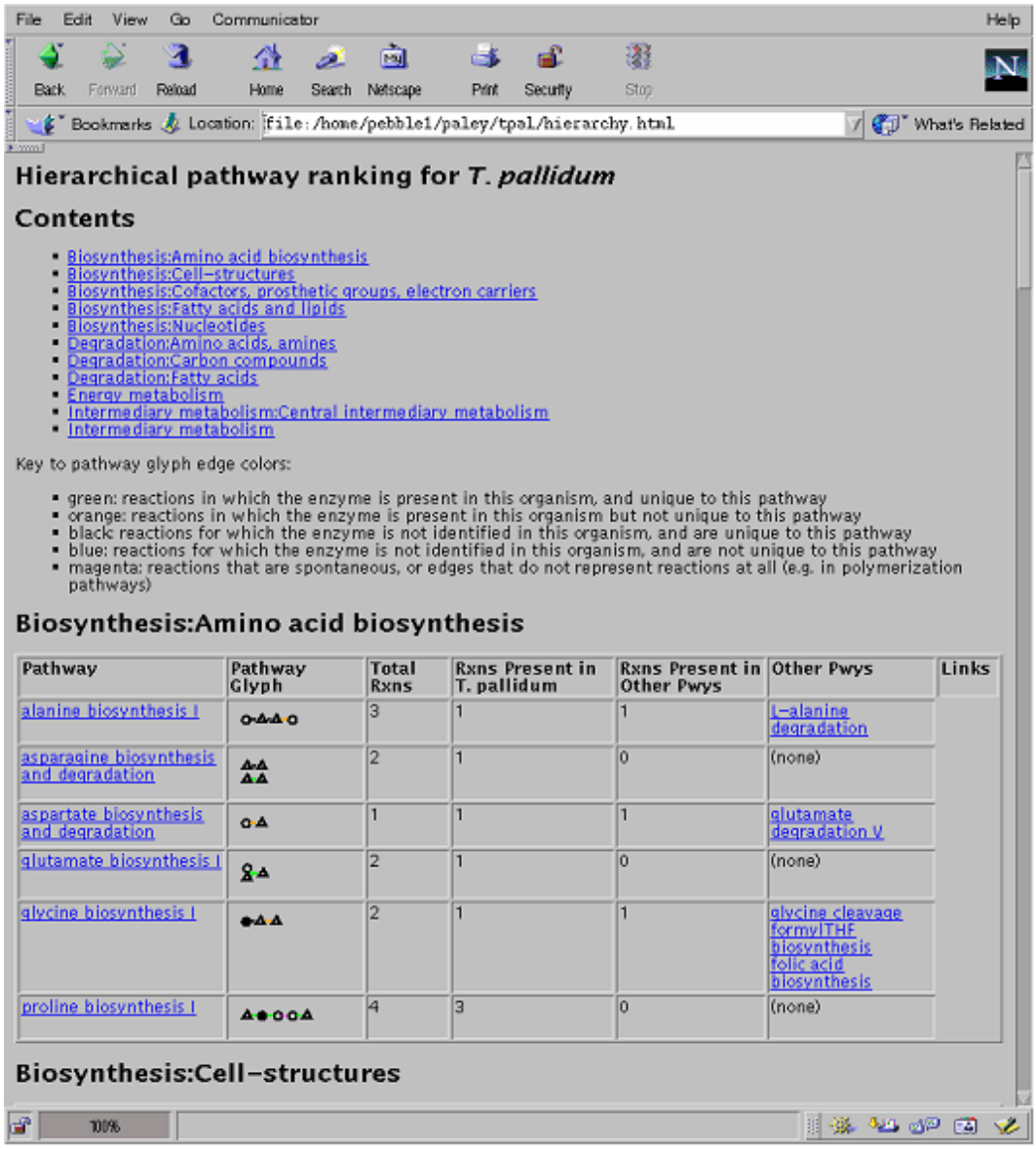


Figure 6.25: Pathway Predictor Summary Page organized by pathway functional category

for outlining issues relevant to interpretation of pathway evidence scores. We then discuss the summary page that lists reactions not predicted to be carried out in organism O that would complete its partial pathways.

It is important to understand that the pathway predictions presented in the newly created pathway/genome database are based upon the genomic sequence of a specific species and that there may be significant metabolic pathway variation between different strains of a given organism.

When the results of pathway analysis show that organism O has genes coding for enzymes that can catalyze **all** the steps of a pathway in the reference database, two interpretations are possible:

- The pathway is in fact found in O and is active.
- The pathway is not active in O for several possible reasons. For example, the genes that encode these enzymes might never be expressed, or the genes might contain mutations that render one or more enzymes within the pathway inactive, or the assignment of enzyme function through sequence analysis was incorrect.

When pathway analysis finds **partial** evidence for the presence of a pathway P in O, several interpretations are possible:

- The pathway is active in O, but genes encoding enzymes that catalyze the remaining steps in the pathway remain unidentified in the genome; the enzymes used by O may have no detectable homology to previously known enzymes that perform those functions; or there may be no known sequences for enzymes that perform those functions.
- A variant of pathway P is active in O; that variant might substitute slightly different reaction steps, catalyzed by different enzymes, for reactions in P.
- Only a segment of the full pathway P is active in O, and that segment connects with O's metabolic network in a different way than P connects with the reference metabolic network. For example, some pathways contain two or more alternative routes for accomplishing a given transformation; if only one of these routes is present in O, the transformation can still be accomplished, but the score for the pathway will be artificially low (examples of pathways containing such alternative transformations are methionine biosynthesis and purine biosynthesis).
- The pathway is not active because the genome does not in fact code for the enzymes that catalyze the remaining "missing" steps (pathway holes).
- The pathway is not active in O because most or all of the enzymes in the pathway had their functions wrongly identified by sequence analysis.

When pathway analysis finds **no** evidence for the presence of a pathway P in O, several interpretations are possible:

- The pathway is in fact not active in O.
- The pathway is in fact active in O, but all the enzymes that catalyze the steps within the pathway remain unidentified in the genome.

- A related pathway that accomplishes a similar biological function (such as catabolism of a sugar), but using a different set of reaction steps, is in fact active in O. That pathway would not be detected in O by the Predictor if that pathway was not present in the reference database.

When deciding whether a pathway was incorrectly predicted and should be deleted from the PGDB, consider these factors:

- How much overall evidence is there for the presence of the pathway, both in terms of the total number of reaction steps for which enzymes are assigned, and in terms of the number of reaction steps that are unique to the pathway and have assigned enzymes. If a reaction and its enzyme are not unique to the pathway, they may be present because of their role in another pathway.
- Is there experimental evidence for the presence of this pathway in O or in a related organism, or is the pathway far from its expected taxonomic range?
- Are other pathways or transporters present that will supply or consume the inputs to and outputs from the pathway? If not, the presence of the pathway is less likely.
- Several variants of a given pathway may have been predicted by PathoLogic because it was unable to find evidence that distinguished between them. If you are unable to find such evidence, you may wish to leave them in the PGDB until such evidence becomes available.

6.6 PathoLogic Batch Mode

Pathologic can be run in batch mode to create one or more PGDBs in an automated fashion, skipping the manual steps listed in "Refining the PGDB". Use the -patho command-line argument in either of the following forms creates a set of PGDBs; you need to specify the path to a file containing a list of param-dirs, one per line:

-patho param-dir Creates a single PGDB based on parameters specified in the following files in the directory you specify by param-dir:

- organism-params.dat (see "File organisms-params.dat" in Section 6.6.1)
- genetic-elements.dat (see "File genetic-elements.dat" in Section 6.2.1)
- and for each replicon (chromosome or plasmid) listed in genetic-elements.dat:
  - gene sequence FASTA file, <repliconid>.fna or <repliconid>.fsa
  - gene annotation file, one of the following:

1. <repliconid>.pf (see "The PathoLogic File Format" in Section 6.2.2)
2. <repliconid>.gbk (see "GenBank File Format" in Section 6.2.3)

-patho -f file Creates a set of PGDBs. You need to specify the path to a file containing a list of param-dirs, one per line.

Note: The directory or file you specify on the command line should not be the standard PGDB directory location (see Section 6.2.2) because Batch Mode needs to create the standard directories and files for you. So, you need to create the directory or file somewhere else.

6.6.1 File organism-params.dat

Each line of organism-params.dat should be of the form:

```
ATTRIBUTE       value
```

The separator character is the tab character. Case is not important, except for literal strings.

Valid attributes are:

ID: unique id—2 to 10 alphanumeric characters, no intervening spaces, must start with a letter

STORAGE: either File, Oracle, or MySQL; defaults to File

NAME: full species name

ABBREV-NAME: abbreviated species name

SUBSPECIES: subspecies name, if any

STRAIN: strain name, if any

PRIVATE?: either T or NIL, defaults to NIL, specifies whether database can be published to the general public over the Web

RANK: NCBI Taxonomy rank

DOWNLOAD-TIMESTAMP:

ORG-COUNTER: number that uniquely identifies this database

DOMAIN: the Organisms subclass Frame ID from MetaCyc or NCBI Taxonomy KB under which the instance frame for the PGDB will be created or, if CREATE? is set, the organism class for the PGDB, which will be the parent for the instance, will be created.

CREATE?: specifies whether to create a new Organism Taxonomy class for the current PGDB in the case that no appropriate Organisms subclass was found in either NCBI Taxonomy kb or MetaCyc. If CREATE? is set to t then the new frame will be created as a child of DOMAIN, which must be specified in this case

CODON-TABLE: a number specifying the codon table for the organism. A list of valid numbers can be obtained from NCBI. If not supplied, the standard codon table will be used

MITO-CODON-TABLE: a number specifying the mitochondrial codon table for the organism. A list of valid numbers can be obtained from NCBI. If not supplied, the standard codon table will be used

AUTHOR: project author—either name only, or name:institution

CITATION: medline ID

HOMEPAGE: URL for project home page

EMAIL: primary contact email address for the project

DBNAME: the "pretty name" for the project database

COPYRIGHT: copyright string, in HTML format, to appear on Web pages

FOOTER-CITATION: citation string, in HTML format, to appear on Web pages, that users of the data should cite

NCBI-TAXON-ID: ID from the NCBI organism taxonomy database

ID and one of either NAME, DOMAIN or NCBI-TAXON-ID must be specified in order retrieve the organism information and to place the organism within taxonomic hierarchy. If there is no need to create an organism class this is the only information needed about the organism since the rest will be retrieved from the NCBI Taxonomy or MetaCyc

If the NCBI Taxon ID is not in the current NCBI Taxonomy KB or the organism doesn't have one assigned and there is no frame for it in MetaCyc, you need to create the taxonomic class for your organism and to specify a parent class. To do so you need to set CREATE? and specify a DOMAIN. You will also need to specify the organism information parameters: NAME, RANK, and if available SUBSPECIES, STRAIN, CODON-TABLE and MITO-CODON-TABLE. If the organism has an NCBI Taxon ID but it is not in our database

yet assigned you can specify it using NCBI-TAXON-ID while creating the organism class. This will make it easier for our software to upgrade your PGDB.

Multiple values may be supplied for AUTHOR and CITATION, in which case each should appear on its own line. If multiple values are supplied for any other field, all but the first will be ignored. Once the file and the project are generated, the ID and STORAGE fields should **not** be edited. The other attributes may be edited so long as the correct format is maintained. Case is important only for species and strain names. Attributes other than these will be ignored.

6.7 Ongoing PGDB Curation

PGDBs need constant attention to keep the knowledge that they contain current. For example, literature searches can be conducted to identify:

- Predicted pathways for which there exists direct experimental evidence
- Experimentally isolated enzymatic activities for which no gene was identified in the genome

This information can be manually added to the relevant entries in the PGDB. Any information about an enzyme that is not linked to a gene represents an experimentally confirmed enzymatic activity for which no gene was identified in the genome. Gene-finding efforts and/or ongoing ORF annotation should focus on these enzymes as genes for these enzymes almost certainly reside in the genome.

MetaCyc does not contain all microbial pathways of small molecule metabolism. The process of PGDB construction is such that any pathways not in MetaCyc but present in the subject organism are not incorporated into the newly constructed PGDB.

Additional literature searches can reveal functions for genes of unknown functions, and transcriptional regulatory processes that can be encoded in the PGDB.

6.8 Updating for New Versions of MetaCyc

New versions of Pathway Tools are released every six months. Included in these releases are updated versions of the MetaCyc database. In addition to new pathways, reactions and compounds, the latest version of MetaCyc can also include changes to existing pathways, reactions and compounds, such as addition of structures to compounds, balancing of reactions, and fixing errors. It is recommended that these updates be periodically

propagated to any PathoLogic-generated PGDBs to keep them up-to-date with the latest knowledge. This propagation is not done automatically, however, to avoid the risk of overwriting changes that curators have made to a PGDB. Instead, a search is made for certain classes of differences between a PGDB and MetaCyc, and the user can select which changes should be made.

To begin the process, invoke the command **Tools** → **Propagate MetaCyc Data Updates** in the Navigator. Occasionally, if you have done a lot of manual editing of compounds, reactions or pathways in one of your PGDBs, you may wish to propagate changes from that PGDB to others, and if that is the case you can select the PGDB to propagate from here. Typically, however, you will want to use MetaCyc as the reference PGDB. The software will take a few minutes to compute the differences between MetaCyc (or other reference PGDB) and your PGDB, and then a dialog will appear summarizing the results. An example dialog is shown in Figure 6.26.

The dialog contains different sections for compounds, reactions and pathways. Each section lists the number of objects that differ for each attribute. For each attribute, you may choose either to propagate all the MetaCyc data, or to bring up a list of all objects with differences and select which ones to propagate individually. A sample dialog showing compound structure differences is shown in Figure 6.27. Note that for each compound it shows the object ID and common name, the structure in the current PGDB, the structure in MetaCyc, and two buttons. The button labeled **Show Compound Displays** will cause the Navigator to display the pages for the compound both in the current PGDB and in MetaCyc. The button labeled **Show Slot Differences** (which is only present for attributes such as compound structures or reaction equations that involve multiple database slots) will bring up another dialog showing the detailed differences in how the structures are represented in the databases. Select those objects that should be propagated, and click the **Update Selected** button.

Some attributes for which multiple values are permitted (such as synonyms or citations) give you the additional option of merging values from MetaCyc with the values in the current PGDB.

The dialog also lists compounds, reactions and pathways that exist in the current PGDB but not in MetaCyc. If these are objects that have been newly curated in your PGDB, that is to be expected, although we ask that once your new pathways have been sufficiently well curated you submit them for inclusion into MetaCyc (see Section 4.1). In other cases, however, these may be objects that have been deleted from MetaCyc because they were invalid or have been merged with some other object. In such cases, you may wish to delete or merge them in your PGDB also. The **Examine** button brings up a dialog listing the relevant objects (see Figure 6.28). A search is conducted for potential merge candidates for each object, both in the current PGDB and in MetaCyc, and all such candidates are listed. The **Show** button displays the corresponding object in the Navigator. In general, an object can be deleted if it does not add any useful information, for example a

Figure 6.26: Propagation Summary dialog

compound that does not participate in any reaction, or a reaction that has no enzyme (or whose enzyme is already also linked to a related reaction) and which does not participate in any pathway. Once you have indicated which objects should be deleted or merged, click the **Delete/Merge Selected** button.

This update propagation procedure only makes corrections to compounds, reactions and

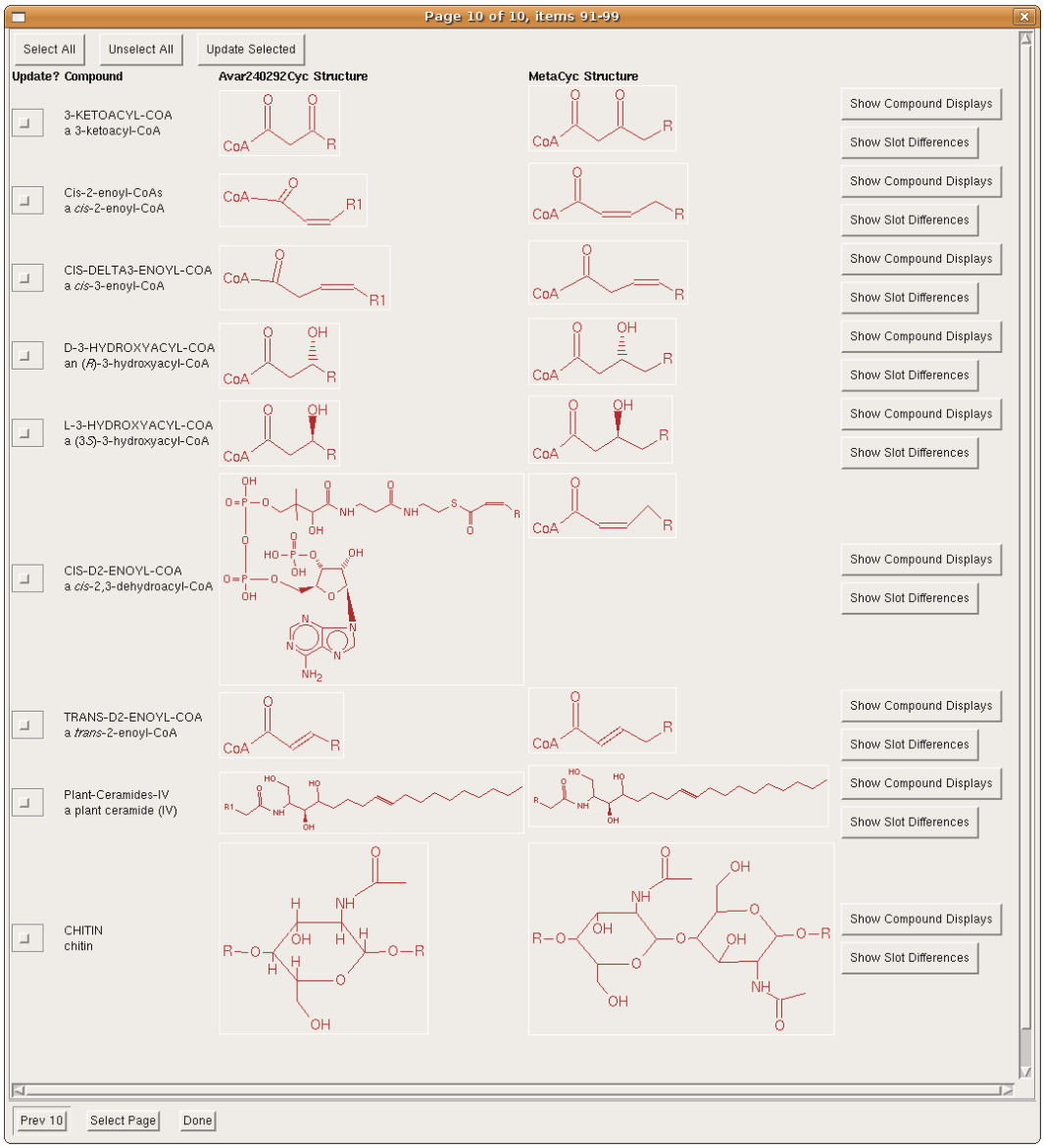


Figure 6.27: Dialog showing Compound Structure Differences

pathways that are already present in a PGDB. It does not attempt to import any new pathways from MetaCyc. If you wish to see if there is evidence for any of the new pathways in a new release of MetaCyc in your PGDB, you must use the PathoLogic command **Rescore Pathways** (see Section 6.4.3).

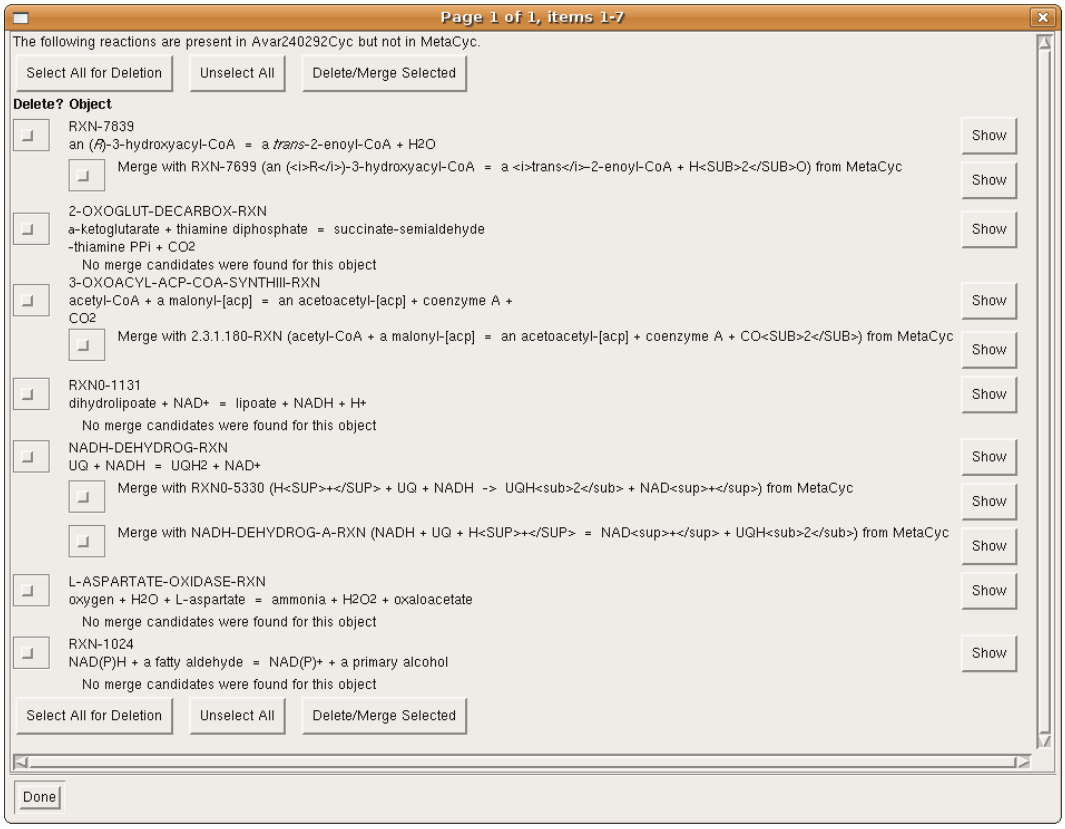


Figure 6.28: Dialog showing Reactions to Delete or Merge

6.9 Suggested PGDB Release Procedures

For groups that release their PGDBs publicly, such as on their Web sites using Pathway Tools in Web mode, we recommend the following procedure to prepare a PGDB for release. These steps are designed to ensure that a released PGDB is of high quality and is internally consistent.

- Decide on the frequency of releases for your PGDB (recommendation: two to four releases per year).
- Run the consistency checker located under the **Tools** menu of the Pathway/Genome Navigator, to check for internal errors within the PGDB. (**Note**: It is important to select and run the task called "Recompute database statistics". This tool will update the organism summary statistics needed for your release.)
- Update the Cellular Overview Diagram (see Section 6.4.9 for more details).
- Provide updates to the PGDB authors and publication information that is stored

in the organism frame for the PGDB. This can be achieved by right-clicking on the organism name and selecting **Edit** → **Frame Editor**.

- Create flat files for the PGDB through the **File** → **Export** menu. The "Entire DB to attribute-value and BioPAX files" command (found under **File** → **Export** → **Entire DB to attribute-value and BioPAX files**) writes a full set of data files, in several different formats, for the currently selected PGDB.
- Create a new version of your PGDB. Perform future editing on the new version, and release the old version to provide a stable polished PGDB to your external users.
- Deposit your PGDB in the SRI PGDB registry (see Section 5.2 for details).
- Author release notes that describe changes to your PGDB (see Section 8.2 for details).

Chapter 7

Editing Pathway/Genome Databases

The Pathway/Genome Editors hereafter referred to as simply the *editors,* are a collection of tools for interactively modifying the objects within a Pathway/Genome Database (PGDB). Each editor is oriented toward modifying a particular class of objects, such as genes, reactions, and metabolic pathways.

In this chapter:

Section 7.1 provides background information common to editing all PGDB editors.

Section 7.2 describes the object data model used by the Ocelot DBMS, which underlies all PGDBs.

Section 7.3 describes individual editors in detail.

Section 7.4 provides examples of how to use the editors to perform PGDB update tasks such as changing the functional annotation of a gene.

Section 7.5 describes advanced aspects of editing that are common to many editors, such as how to reference the biomedical literature from PGDBs, how to credit the curators of pathways and enzymes within a PGDB, and restrictions on editing of PGDBs that are designed to protect your work.

For information about strategies and policies for curation of PGDBs, refer to the Pathway Tools Curator's Guide, available at `http://bioinformatics.ai.sri.com/ptools/curatorsguide.pdf`.

7.1 Overview of the Editors

This section provides a listing of the available editors, a description of how to invoke each editor, and a brief discussion of how to save and undo changes to a PGDB. Later

sections describe each editor in more detail, and provide a more elaborate description of the Ocelot database system used by Pathway Tools.

7.1.1 Right-Click on Object Handles to Edit Existing Objects

The editors are used both to modify existing entities in PGDBs and to create new entities. All editors are launched from the Navigator window using commands listed in the next section. Before describing those commands, we describe the notion of object handles, which is needed to understand how those commands are invoked.

Whenever you see an object within the Navigator, you can launch an editor to update that object by right-clicking on the handle for the object and selecting an editor from a command menu that appears after right-clicking. Navigator displays have two types of object handles. The first type is the name of an object that appears in the title region at the top of the Navigator display window for that object. For example, if we are viewing the display for the *E. coli trpA* gene within the Navigator display (see fig. 7.1), the handle for *trpA* is the text "*trpA*" at the top of that window. You can edit the *trpA* gene by right-clicking on the title handle for *trpA*. The second type of handle consists of the names of other objects in a Navigator display. For example, the display for *trpA* contains the following handles, and by right-clicking on any of them, you can edit the respective objects:

"tryptophan synthase A protein" is the handle for the polypeptide product of trpA.

"tryptophan synthase" is the handle for protein complex in which that polypeptide is a subunit.

"indole-3-glycerol-phosphate + L-serine $\longrightarrow$ H2O + glyceraldehyde-3-phosphate + L-tryptophan" is the handle for the reaction catalyzed by that protein complex.

"L-serine" is the handle for a substrate in that reaction.

"tryptophan biosynthesis" is the handle for the pathway in which that protein complex is an enzyme.

7.1.2 Available Pathway/Genome Editors

The editors described here are available to create and update different types of objects and relationships within a PGDB.

Editing genes (Section 7.3.1) :

- To create new genes, invoke **Gene→New Gene**.

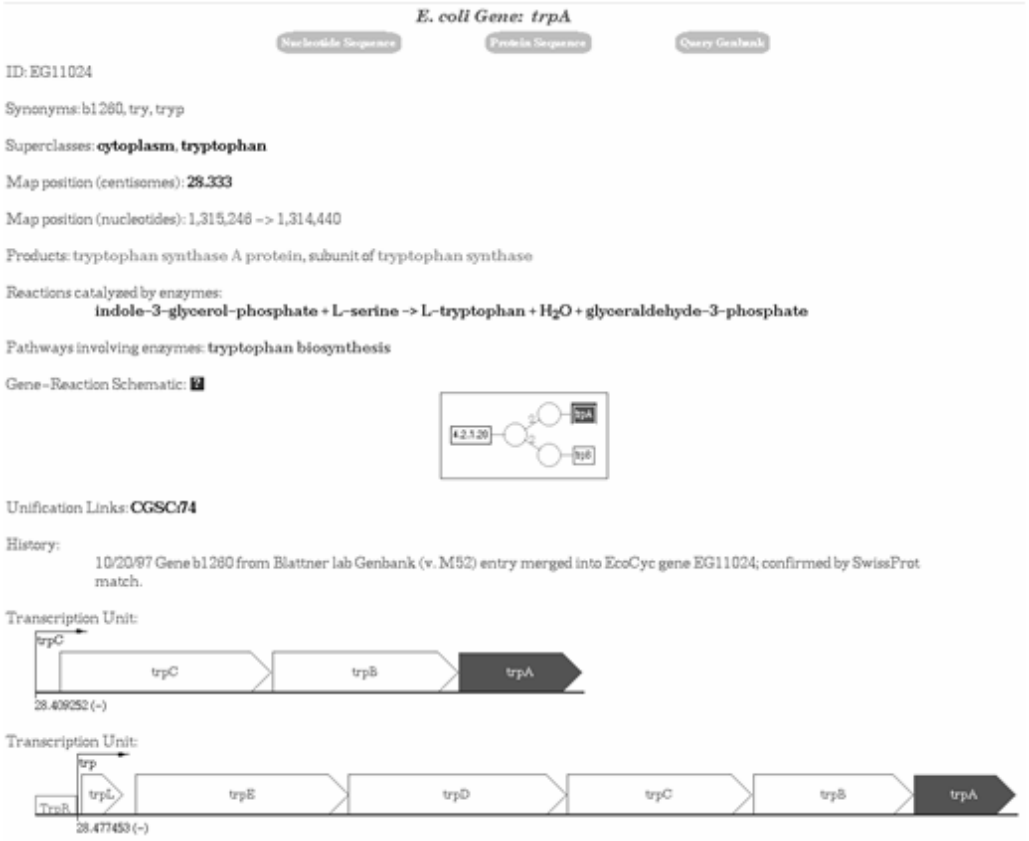


Figure 7.1: Navigator display for the gene trpA

- To modify existing genes, right-click on the gene handle and choose **Edit→Gene Editor** .

Editing intron and exon definitions (Section 7.3.3) :

- To create a new splice form for a gene, or to edit the set of introns associated with a gene or particular splice form of a gene, right-click on the gene and select **Edit→Intron Editor**.

Editing transcriptional regulatory information (Section 7.3.2) :

- To create new transcription units, **Gene→New Operon**.
- To modify existing transcription units, right-click on the transcription unit handle and choose **Edit→Transcription Unit Editor**.
- Once a transcription unit has been created, you can create and modify interactions between transcription factors and the promoter of that transcription unit as follows:

- To create or update the terminators associated with a transcription unit, right-click on the transcription-unit handle and invoke **Edit→Terminators**.
- To create a new transcription initiation regulatory interaction, right-click on the transcription unit handle and select **Edit→Create Binding Interaction**.
- To create a new transcription attenuation regulatory interaction, first create the attenuated terminator, right-click on the terminator icon and select **Edit→Add Attenuation Interaction**.
- To modify an existing regulatory interaction, right-click on the handle for the binding site or attenuation icon and choose **Edit→Regulatory Interaction Editor**.

Editing RNAs (Section 7.3.4) :

- To create new RNAs, **RNA→New**.
- To modify existing RNAs, right-click on the RNA handle and **Edit→RNA Editor**.

Editing Proteins (Section 7.3.5) :

Editing proteins has several aspects: defining the subunit structure of multimeric protein complexes, defining properties of proteins (be they monomers or multimers) and attaching proteins to reactions that they catalyze, and defining protein features such as active sites and chemical modification sites.

- To create new proteins, invoke **Protein→New.** This command will first invoke the Protein Subunit Substructure Editor to allow you to specify whether the new protein is a monomer or a multimer, and from which gene product(s) it is comprised. Once its subunit structure is specified, this command will invoke the Protein Editor to allow you to define properties of the newly created protein.
- To modify existing proteins, right-click on the protein handle and select the command for editing the desired aspect of the protein. To modify the subunit structure of a multimer, to convert a multimer to a monomer, or to convert a monomer to a multimer, select **Edit→Protein Subunit Structure Editor**. To modify properties of an existing protein, select **Edit→Protein Editor** . A button within the protein editor labeled **Edit Protein Feature(s)** allows creation and modification of features on a protein, such as phosphorylation sites and transmembrane regions.
- To associate a new or existing enzyme with a given biochemical reaction, thus specifying that the enzyme catalyzes that reaction, right-click on the reaction handle and select **Edit→Create/Add Enzyme** .

Editing pathway information (Section 7.3.6.2) : The Pathway Info Editor allows editing of aspects of a pathway other than its reaction structure, such as its comment and literature citations. The Pathway Editor allows editing of the reaction structure of a pathway.

- To create new pathways, invoke **Pathway→New**.
- For modification of a pathway, right-click on the pathway handle and choose **Edit→Pathway Editor** to edit its reaction structure, or choose **Edit→Pathway Info Editor** to modify other information about the pathway.

Editing reaction information (Section 7.3.7) :

- For creation of new reactions, **Reaction→New**.
- For modification, right-click on the reaction handle and choose **Edit→Reaction Editor**.
- To specify that a new or existing enzyme catalyzes a particular biochemical reaction, right-click on the reaction handle and select **Edit→Create/Add Enzyme**.

Editing chemical compound information :

Just as pathway structures are edited independently of textual information about pathways, chemical structures are edited independently of textual information about chemical compounds, using the following tools.

Compound Editor — edits aspects of a chemical other than its structure (Section 7.3.8)

- To create new compounds, **Compound→New**.
- For modification, right-click on the compound handle and choose **Edit→Compound Editor**.

Marvin Compound Structure Editor — first option for editing chemical structures (Section 7.3.9) — right-click on the compound handle and choose **Edit→Marvin Compound Structure Editor** .

JME Chemical Structure Editor — second option for editing chemical structures (see Section 7.3.10) — right-click on the compound handle and choose **Edit→JME Compound Structure Editor** .

Editing publication information (Section 7.3.14) : Publications can be referenced within textual comments in a PGDB object such as a protein, and within a list of citations for a PGDB object. PGDBs contain two types of literature references: references to publications that are included in the PubMed database, and references to publications that are not in PubMed.

New publication objects in a PGDB are created in two ways. First, if you reference a non-PubMed publication in the citation or comment field of any editor, and that publication does not already exist in the PGDB, you will automatically enter the Publication Editor to create it. Specifically, the software infers that you are referencing a non-PubMed publication if your reference is not of the form of a bracketed PubMed ID number, such as [78345682] If, for example, you reference [Smith95], and that publication is not already in the PGDB, you will enter the Publication Editor to create it.

The second way to create a new publication object in a PGDB is to use the **File→Import→Citations from PubMed** command to create PGDB publication objects for every PubMed citation referenced by the PGDB (see Section 4.7 for more information).

- To modify a PGDB publication , right-click on a citation handle in the References section at the bottom of any Navigator display page, e.g., the "Smith95" at the front of the reference, thus invoking the publication editor.

7.1.3 Saving Edits

Modifications that you make to a PGDB using the Editors are **not saved** until you explicitly save them using the **Save DB** button at the far right of the Navigator toolbar (this command is also accessible from the **File** menu as **File→Save Current DB**, and with the associated keyboard shortcut Ctrl-S). Until **Save DB** is executed, edits will be lost if a power failure or computer crash occurs.

You can use the menu command **File→Revert Current DB...** to discard all PGDB modifications (creation of new objects and changes to existing objects) that you have made since the last **Save DB** command was executed, if you prefer not to retain them.

7.2 Frame Data Model

To understand PGDBs, you must understand the frame data model that is used to structure information in all *Ocelot DBs* (see the Glossary for definitions of these and other terms). The frame data model is very similar to the object data model used in object-oriented databases. The frame data model comprises the following components.

7.2.1 Frames

An individual biological entity in a PGDB is represented as an *instance frame,* or *instance.* For example, the amino acid lysine and the *E. coli* gene *trpA* are instances. Instances are children of *class frames,* which represent collections of semantically related instances. Classes represent general biological concepts. In the previous examples, a class called *Amino Acids* is the parent of the instance lysine, and a class called *Genes* is the parent of the *trpA* gene instance.

Table 7.1 lists the most important classes in a PGDB, although a PGDB typically contains more than 1000 classes.

Class	**Description**
Reactions	All reactions, including enzyme and transport reactions
Pathways	All pathways, including superpathways and signal-transduction pathways
Genes	All genes
Polypeptides	All monomers
Protein Complexes	All protein complexes
Proteins	The parent of Polypeptides and Protein-Complexes

Table 7.1: Important classes in a PGDB[a]

[a]Note that class names are case sensitive.

A Database, or DB, is a collection of frames and their associated slots, values, facets, and annotations. DBs are saved permanently in Oracle or MySQL database management systems. A *knowledge base*, or KB, is a different word for a database. (A PGDB is simply a pathway-genome oriented DB.)

Each frame has a frame name (also called a frame ID, or a key) that uniquely identifies that frame within the DB.

7.2.1.1 Slots

Slots encode the attributes or properties of a frame, and also represent relationships between frames. A slot is a mapping from a frame and a slot name to a collection of zero or more values. Each value can be any Lisp object (such as a symbol, list, number, or string). Ocelot supports three collection types for slot values: sets, multisets, and lists. For example, the slot called **components** in the AROH-CPLX frame in Figure 7.2 lists the frames that represent the polypeptide subunits of the enzymatic complex. The figure shows a printed representation of two frames, AROH-MONOMER and AROH-CPLX. The former

is an instance of the class **Polypeptides,** and the latter is an instance of the class **Protein-Complexes**. For each frame we list the name of a slot within the frame, followed by a colon, followed by the zero or more values of that slot. An *annotation* records the coefficient of the AROH-MONOMER component within the ARZOH-CPLX.

```
--- Instance AROH-MONOMER ---
   Types:  Polypeptides

CITATIONS:    "[89053867]"
COMMENT:      "The aroH gene has two promoters..."
COMPONENT-OF:  AROH-CPLX
CREATION-DATE: "22-Aug-1993  14:36:13"
CREATOR:      MRILEY
GENE:         EG10080
ISOZYME-SEQUENCE-SIMILARITY: (AROG-MONOMER YES), (AROF-MONOMER YES)
MOLECULAR-WEIGHT-SEQ: 38.721
SPECIES:      "E. coli"


--- Instance AROH-CPLX ---
   Types:  Protein-Complexes


CATALYZES:    DAHPSYNTRP-ENZRXN
COMMON-NAME:  "2-dehydro-3-deoxyphosphoheptonate aldolase"
COMPONENTS:
    AROH-MONOMER
     ---COEFFICIENT: 2
CREATION-DATE: "22-Aug-1993  14:36:13"
CREATOR: MRILEY
SPECIES:      "E. coli"
SYNONYMS:     "phospho-2-keto-3-deoxyheptonate aldolase",
              "DHAP synthase", "DHAPS", "KDPH synthetase",
              "tryptophan sensitive 3-deoxy-D arabino-heptulosonate 7-phosphate synthase",
              "3-deoxy-D-arabinoheptulosonate-7-phosphate synthetase (trp)"
```

Figure 7.2: Printed representation of frames for AROH-MONOMER and AROH-CPLX

7.2.1.2 Facets

Facets encode information about slots. Facets are uniquely identified by a facet name, a slot name, and a frame. A facet has as its value a set of Lisp objects. Facets allow comments or citations to be associated with an entire slot.

7.2.1.3 Annotations

To store information about individual slot values, we use annotations. An annotation associates a labeled set of data objects with a particular slot value. Annotations are used to attach comments and citations to slot values. In addition, as shown in Figure 7.2, an annotation stores the coefficient of each subunit within a protein complex. The slot value AROH-MONOMER has an annotation whose label is **coefficient** and whose value is 2.

7.2.2 Relationship of Frames to Pathway Tools Displays

An important thing to keep in mind when modifying a PGDB is that the display windows generated by the Pathway/Genome Navigator and Pathway/Genome Editors do not literally display the exact contents of the PGDB. Every Navigator window is computed dynamically by a program that takes the information in a Pathway Tools frame (such as a pathway frame), plus information from related frames (such as the reactions within the pathway), and computationally maps the information in these frames into a drawing. These programs display some slots more or less verbatim, such as the **comment** slot — although even that slot is processed to convert citations into bracketed, clickable links. Other slots are not displayed at all. For example, when displaying a compound, the slot that stores the chemical bonds within the compound is not displayed literally, although the information it contains is used to produce a drawing of the compound structure.

Sometimes fairly complicated transformations are applied to create these drawings. For example, when displaying an enzyme, Pathway Tools displays some information from the enzyme frame itself, but it also makes use of information from the enzymatic-reaction frame(s) connected to the enzyme frame, the reaction frame (s) connected to the enzymatic reaction(s), and the compound (substrate) frames listed in the reaction. When Pathway Tools displays those related objects, the name it prints for a given frame is usually the value of the **common-name** slot for that frame, or else the frame name if that slot has no value.

Therefore, it may not always be obvious which frame you must edit to effect a given change in a display window. For example, to change the name shown for a chemical compound in the display of a pathway, you should edit the chemical compound frame, not the pathway frame. As you gain experience with the system, you will learn what editor to use to alter a given part of a Navigator display.

7.3 Editors

Each editing tool is designed for entering information about a particular data type, such as genes, pathways, or proteins. In some cases you will need to use more than one editor to

implement what may seem like a "single" change in biological knowledge. For example, if an enzymatic function is assigned to a gene whose previous function was unknown, you would use the gene editor to name the gene and describe its function, and then use the protein editor to connect the enzyme to a reaction (perhaps within a pathway), and to define properties of the enzyme such as its cofactor and inhibitors, when known.

7.3.1 Gene Editor

By a *gene* we mean a region of DNA that encodes a protein, tRNA, or other type of gene product. The gene editor is shown in Figure 7.3. Fields within the gene editor include the following:

Class: Assigns the gene to one or more classes in the MultiFun functional classification scheme developed by M. Riley and G. Serres.

Common-Name, Synonyms: The primary name for the gene, and alternative names for the gene.

Product: The name of the gene product is printed here. However, the name of the gene product is specified in the Common-Name slot of a frame that represents the gene product. If you want to edit that frame, click the **Edit** button.

Product-Types: Specifies the one (or more) broad categories of function of the gene product, when known.

Evidence: Specifies the type of evidence used to infer the function of the gene product.

Links to Databases: Defines Web-based links to information about this gene in other databases. Select a database to link to, and enter the unique identifier assigned to this gene in that database.

Gene Location: The position of the gene on its replicon (chromosome or plasmid) is specified by giving the start position and end position of the gene's coding region within the nucleotide sequence of the replicon (the former must always be the smaller number, unless the gene spans the origin of replication), and by specifying the direction of transcription for the gene. The sequence indicated by that position and direction is displayed by the editor.

Add History Note: A history note is a text comment that is marked with the name of the user who created it, and the date. We recommend that when information about a gene is revised (such as its function), you should create a history note explaining the rationale for the change. Click this button to create a new history note, which will be displayed in the window for this gene by the Navigator.

Figure 7.3: Gene Editor

7.3.2 Editing Transcriptional Regulatory Information

Definition: A *transcription unit* is a region of DNA in a prokaryotic organism that includes a single transcription start site (TSS), the transcription-factor binding site(s) that modulate the rate of transcription initiation at that transcription start site, and the gene(s) and transcription terminator(s) that are downstream of that transcription start site. There is a one-to-one relationship between a transcription unit and its transcription start site, meaning that each transcription unit must have exactly one transcription start site (unless the location of the start site has not been precisely determined). A transcription unit differs from an operon because by definition an operon must contain more than one gene, whereas a transcription unit may contain one or more genes; an operon may also include more than one transcription start sites, whereas a transcription unit must contain a single transcription start site (when known), by our definition. Therefore, our approach defines

a different transcription unit for each transcription start site in an operon containing multiple transcription start sites.

The transcription-unit editor allows you to define and modify frames that describe the complex set of molecules and interactions that define the genetic network of an organism, including

- Transcription units
- Transcription-factor binding sites
- Transcription terminators
- Molecular interactions between
  - Small-molecule ligands and transcription factors that bind to those ligands
  - Transcription factors and the DNA binding sites they recognize
  - RNA polymerases and promoter regions (including modulation of those interactions by transcription factors and their associated binding sites)

The stages in defining a genetic-network fragment centered around a single transcription unit are as follows.

7.3.2.1 Step 1: Invoke Transcription Unit Editor

Create a new transcription unit with the command **Gene→New Operon** . The transcription-unit editor, shown in Figure 7.4, allows you to enter information such as a name and synonyms for the transcription unit, an ordered list of the genes it contains, and the type of evidence used to infer the existence of the transcription unit.

With this editor, you can also enter a name and synonyms for the transcription start site associated with the transcription unit, and you can define its position as either an absolute nucleotide position on the chromosome, or as a position relative to the start codon of the first gene in the transcription unit. The editor displays the sequence corresponding to that region of the chromosome. You can also specify the type of evidence for the existence of the transcription start site, the sigma factor required for RNA polymerase to recognize this promoter, and certain numerical constants for the promoter. The editor creates a frame describing the interaction between RNA polymerase and this promoter. Once you click OK and exit, the transcription unit is created.

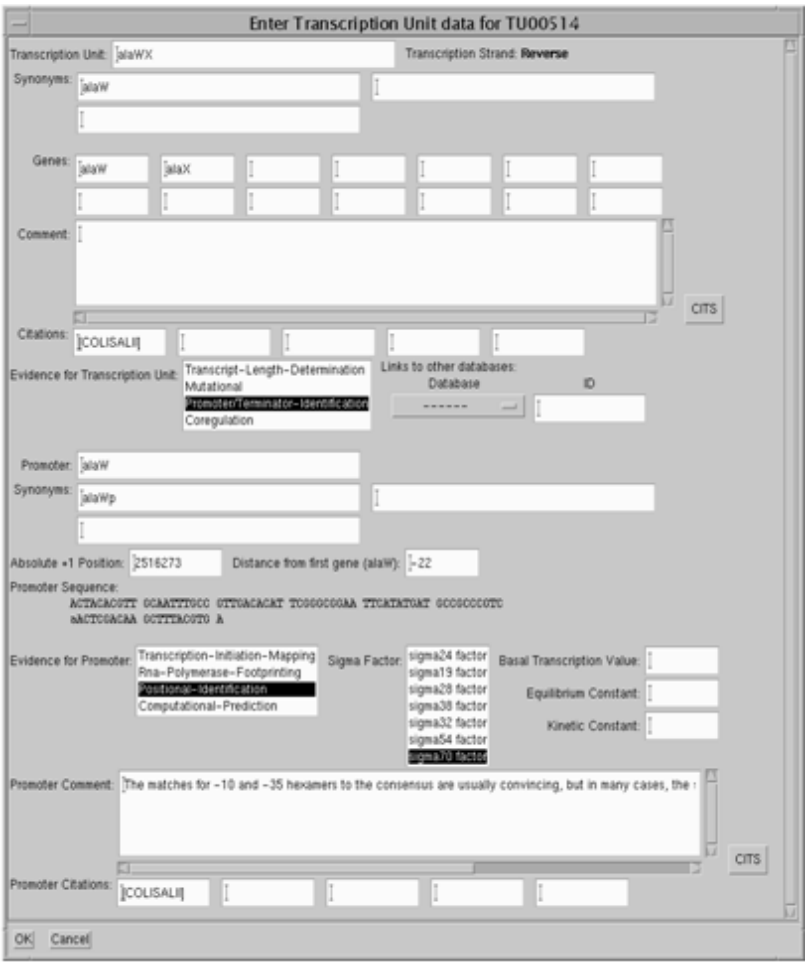


Figure 7.4: Transcription Unit Editor

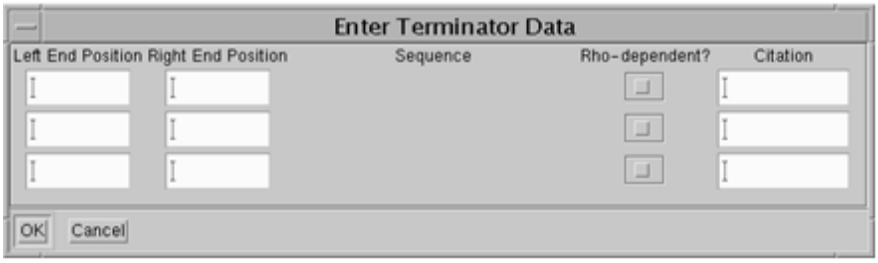


Figure 7.5: Terminator Editor

7.3.2.2 Step 2: Invoke Terminator Editor

If one or more transcription terminators are known, you can add them to this transcription unit by right-clicking the transcription unit and selecting **Edit→Terminators**, which

brings up the Terminator Editor, shown in Figure 7.5. This editor creates frames for the one or more terminators that you specify, and associates them with the transcription factor.

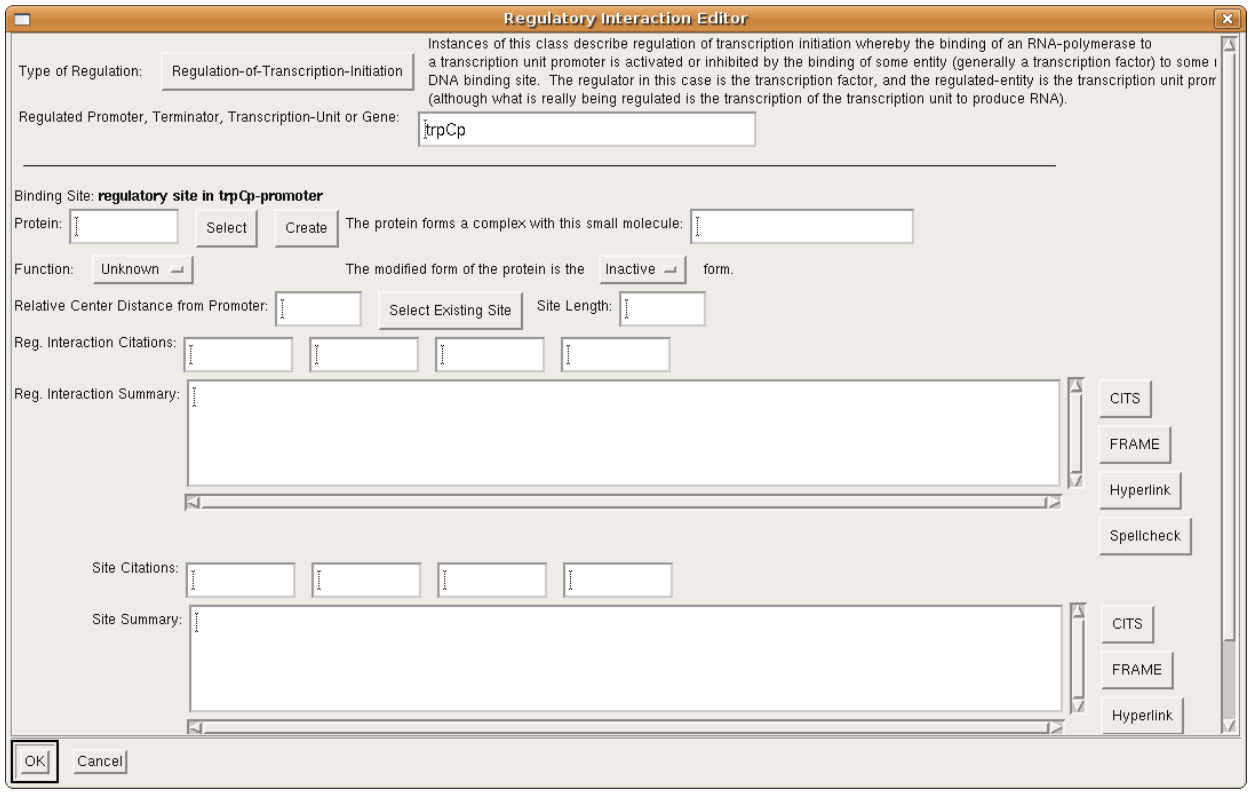


Figure 7.6: Regulatory Interaction Editor

7.3.2.3 Step 3: Invoke Regulatory Interaction Editor

To create a new transcription-factor binding site associated with this operon, right-click the Transcription Unit, and select **Edit→Create Regulatory Interaction** (note that this command will not be available unless a promoter object exists for the transcription unit. Invoking the Transcription Unit Editor will automatically create a promoter object for the transcription unit) . The Regulatory Interaction Editor (Figure 7.6) opens. There you can enter the name of the binding site and its synonyms. To specify the protein that binds to the site, you can either select a protein that is already in the database, or create one by clicking the Create button, which takes you to the Protein Editor (Section 7.3.5.2).

To create a new transcription attenuation interaction, first create the attenuated terminator. Right-click the terminator icon, and select **Edit→Add Attenuation Interaction**. The Regulatory Interaction Editor will open. The kinds of information you can enter here

will depend on the specific attenuation class you select. For example, if you specify you are creating a Ribosome-Mediated Attenuation interaction, you can choose from a list of charged tRNAs as the regulator, and may enter an anti-terminator region and a pause site. If you are creating a Protein-Mediated Attenuation interaction, then your regulator will be a protein with an optional ligand, and while you can still specify an anti-terminator region, there is no pause site.

To modify an existing regulatory interaction, right-click the handle for the binding site or attenuation icon and choose **Edit→Regulatory Interaction Editor**.

7.3.3 Intron Editor

Figure 7.7: Intron Editor

The Intron Editor is shown in Figure 7.7. You can invoke it by right-clicking on a gene and selecting menu **Edit→Intron→Editor**. It contains the following three fields:

1. **Splice Form Gene Product:** This is a menu of gene products associated with each splice form for a gene. To edit the introns for a particular splice form, select the corresponding gene product. To create a new splice form and specify its introns, select the **Create New Splice Form** option. You are then asked to enter a name for the product of the new splice form. A new polypeptide or RNA frame is created with that name, and is linked to the gene.

2. **Relative/Absolute Base Numbers:** Indicate whether you will be entering base numbers that are relative to the start of the gene or that represent absolute positions in the chromosome.

3. **Intron positions:** Enter start and end positions for each intron. Introns may be entered in any order. If an alternate splice form has a start position inside the gene, specify an intron with start position 1 (relative) and end position 1 base before the actual start of coding. Create an analogous intron if the splice form has an end position inside the gene. As intron positions are entered, some basic checks are run, and you are alerted to any possible problems that are identified. These are warnings only, and can be ignored at your own risk.

7.3.4 RNA Editor

The RNA Editor allows creation of an RNA by invoking **RNA→New**, and editing of existing RNAs by right-clicking on an RNA handle and invoking **Edit→RNA Editor**. Specify the gene coding for the RNA, and, with the Class button, the molecular type of the RNA (e.g., ribosomal RNA, transfer RNA). Note that the Pathway Tools ontology does not currently support creation of messenger RNAs within PGDBs.

The editor also allows you to enter standard information such as synonyms, a comment, and literature references.

7.3.5 Protein Editors

The *Protein Subunit Structure Editor* and the *Protein Editor* are used to define and edit a protein. You can select either of these two editors individually from the right-click edit menu for a protein. When you create a protein, you are asked to fill in both forms, in turn.

7.3.5.1 Protein Subunit Structure Editor

The Protein Subunit Structure Editor, shown in Figure 7.8, is used to specify the genes and subunits (if any) that comprise the protein. First, specify whether the protein you want to describe is active as a monomer, or as a protein complex (multimer). If it is a multimer, specify the number of distinct gene products that compose the complex. For example, a homotetramer comprises four copies of one gene product, so you would enter "1." Next, enter the name or ID of each gene product or subunit.

To create a multi-enzyme protein complex whose components are themselves protein complexes, first create the individual component enzymes and link them to the reactions they catalyze, if any. Then, when creating the multi-enzyme complex, enter the names or frame IDs for the subunit complexes in the dialog under Subunits. Do not enter any genes.

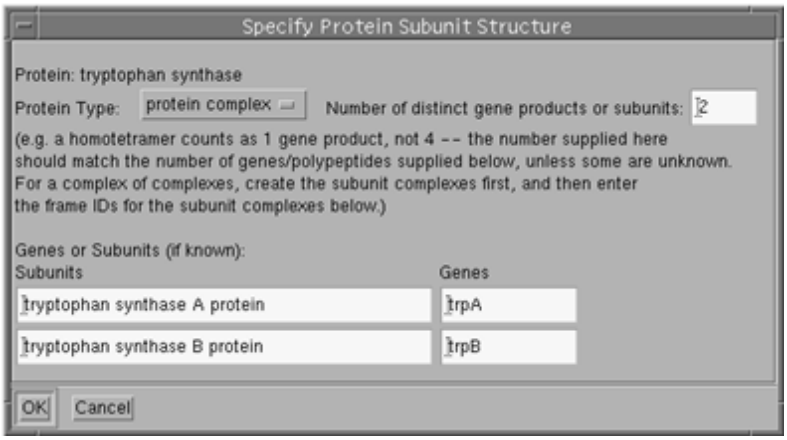


Figure 7.8: Protein Subunit Structure Editor

7.3.5.2 Protein Editor

The Protein Editor, shown in Figure 7.9, is used to specify additional information about the protein. This window is divided into several sections. The top section lets you define general information about the protein such as commentary, citations, database links, GO terms and one or more cellular locations known for the protein.

Subsequent sections allow you to specify one or more known enzymatic activities for the protein. Information you can define in each of these sections includes a name for this enzymatic activity (a multifunctional protein will have multiple names, one for each function), and the reversibility of this reaction with respect to this enzyme. For example, if the reaction is essentially irreversible in the left-to-right direction, specify it as such. If this enzyme typically catalyzes the reaction in the right-to-left direction, choose "Physiologically unidirectional, right-to-left". If the reaction is reversible, specify it as such. This entry will determine whether this step will appear in the pathway diagrams as a unidirectional or a bidirectional arrow.

For one or more modulators (activators or inhibitors), cofactors, prosthetic groups, or alternative substrates for the enzyme, for each molecule, specify the name of the molecule, its relationship to the enzyme (e.g., cofactor), and whether or not its interaction with the enzyme is physiologically relevant (e.g., a molecule that has been demonstrated to inhibit the enzyme in vitro but is not known to occur in vivo should not be marked as physiologically relevant). You can also specify one or more citations in support of this information. If you run out of slots, close the dialog window and reopen it, more empty slots will show up.

Additional sections of this window (not shown) allow you to enter information specific to each subunit of the multimer, including its copy number (coefficient) within the complex, molecular weight, and UniProt ID.

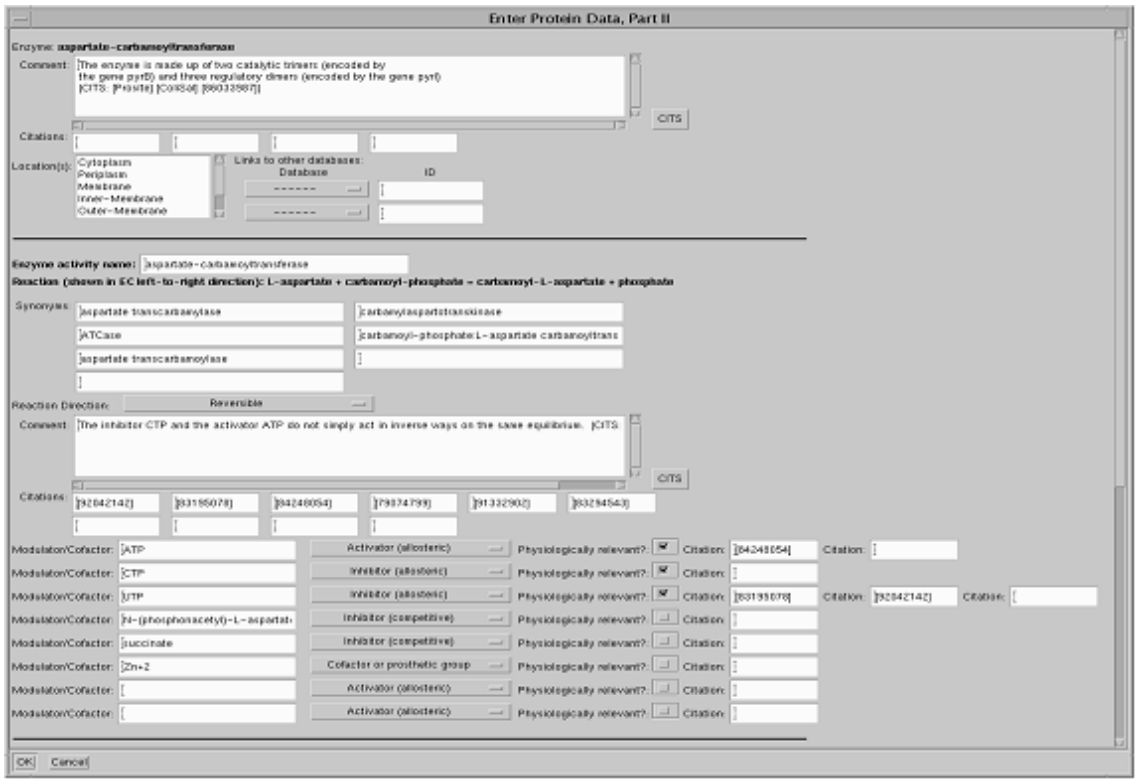


Figure 7.9: Protein Editor

7.3.5.3 Adding an Enzyme to a Reaction

In Pathway Tools, a given enzyme catalyzes a specific reaction by defining a relationship between the enzyme object and the reaction object in the PGDB. There are two different ways to define such an association.

If the reaction and the protein frame have both already been created, right-click on the protein handle in a Navigator display, and select **Edit→Add Reaction(s)**. You can then enter one or more reaction frame IDs or EC numbers. Once the reactions have been linked to the protein, the Protein Editor appears so that you can add information about directionality, activators/inhibitors/cofactors, citations, and so on.

The second way to connect an enzyme to a reaction is by right-clicking on the reaction handle and selecting **Edit→Create/Add Enzyme**. If the enzyme already exists, and you know the frame name or ID, you can enter it directly. Otherwise, select **Search by Genes or Create New Protein** to bring up the Protein Subunit Structure Editor. You can list the genes or polypeptides (or subunit complexes, for enzymes that are a complex of protein complexes) that make up the enzyme. If a protein already exists that matches what you have entered, it will be linked to the reaction. If a protein complex is specified and no such complex yet exists, it will be created. If you are not sure whether the enzyme is a

polypeptide or a protein complex, you can leave the class unspecified while you search for an existing protein. However, if you want to create a new protein, you must specify the class. Finally, the Protein Editor will pop up to allow entry of additional information about the enzyme.

7.3.5.4 Disconnecting an Enzyme from a Reaction

To disconnect an enzyme from a reaction to delete the information that the enzyme catalyzes the reaction, right-click the enzyme. In the menu that pops up, select **Edit→Remove Reaction**.

7.3.5.5 Creating Active/Inactive Forms of an Enzyme

Some proteins may be covalently modified in order to activate or inactivate them. Pathway Tools treats the activation/deactivation process as an ordinary chemical reaction, with the protein and its modified form as the reaction substrates. To create a protein activation/inactivation reaction, first read Section 7.5.10 on Modified Proteins and Section 7.3.7 to learn about the Reaction Editor. Invoke **Reaction→New**. Enter the reaction equation (e.g., XyzA + P_i = XyzA-P) and click **Show Rxn**. In general, if XyzA is recognized as a protein, and XyzA-P does not yet exist as a frame, you are asked if XyzA-P should be created as a modified form of XyzA. Once the reaction has been created, specify which form is active by linking it to a reaction, as described in Section 7.3.5.3. This same process can be used to generate active and inactive forms of a protein for transcriptional regulation, signal transduction pathways and so on.

7.3.5.6 Creating Protein Features

In the Protein Editor section for a polypeptide, the **Edit Protein Feature(s)** button allows for creating and editing of protein sequence features of various types. To create a new feature, click the button and select **Create New Feature**. You will be presented with a hierarchy of possible feature types, such as Phosphorylation-Modifications, Active Sites, and Nucleotide-Phosphate-Binding-Regions. If the exact feature type you need is not listed, you may use a nonspecific type. For example, choose **Amino-Acid-Binding-Sites** (which describe features in which a moiety binds to a single amino acid) if no subtype exists for the type of attached moiety that you are trying to describe.

After you select the feature type, the Feature Editor, shown in Figure 7.10, will appear. You may use this editor to enter descriptive information about the feature: name, comment, citations, and the attached group if the feature type requires one. The protein sequence is provided, and the feature location can be specified by either typing in the residue number or by selecting it with the mouse (a sequence search box is provided to

help in locating the feature within the sequence). You may also supply a sequence motif that may or may not be part of the actual binding or modification site but which is indicative of the feature.

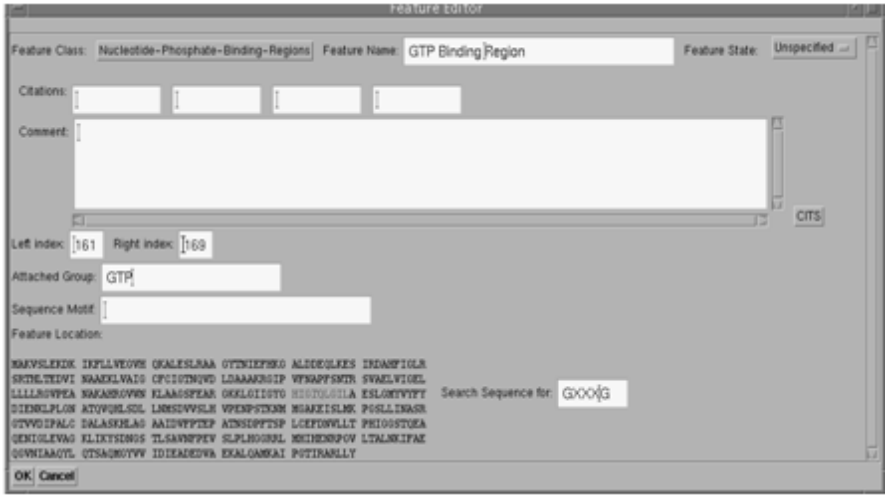


Figure 7.10: Feature Editor

7.3.6 Pathway Editors

The Pathway Editors allow you to graphically create new pathways, and to modify existing pathways. A pathway is a connected set of chemical reactions. You can manipulate the structure of a pathway by altering its component reactions: reactions can be added to or deleted from a pathway, and the connections among reactions can be altered. Reactions are connected by shared substrates. Two reactions share a substrate if a product of one reaction, $R1$, is a reactant of the next reaction in the pathway, $R2$. In this case we say that $R1$ is the *predecessor* reaction of $R2$.

Three pathway editors are provided: the *Pathway Info Editor* (which can be invoked separately from the other tools), the *Pathway Editor*, and the *Segment Editor*. With the Pathway Info Editor, you can enter non-topological information about a pathway, such as commentary, its list of synonyms, its classification in the taxonomy of pathways, and links to other databases. With the Segment Editor, you can enter a linear pathway segment — a linear sequence of reactions within a pathway. The Segment Editor is typically used to enter new pathways, or to add a new sequence of reactions to an existing pathway. With the Pathway Editor, you can add new reactions to a pathway, remove reactions from a pathway, and alter connections between individual reactions.

The Pathway Editor displays the current form of an entire pathway, and is always invoked before the Segment Editor. The Segment Editor can then be used to add new linear

segments to a (possibly empty) existing pathway. A short linear pathway can be entered using one invocation of the Segment Editor. Defining more complex pathways may take several operations:

To enter a long linear pathway (more than seven reactions), use the Segment Editor to enter several segments and then connect the segments together with the Pathway Editor.

To enter a branching pathway, use the Segment Editor to enter each linear segment, and then use the Pathway Editor to connect the segments together to form the branching structure.

To enter a circular pathway, use the Segment Editor to construct a linear form of the pathway, and then connect the ends together within the Pathway Editor.

7.3.6.1 Invoking the Pathway Editors

You can invoke the Pathway Editor either to create a new pathway or to modify an existing pathway. To create a new pathway, select **Pathway**→**New**. The Pathway Info Editor appears; when you click OK to exit, a new instance of that pathway class is created, and the Pathway Editor will pop up, letting you specify the individual reactions of the new pathway.

To modify an existing pathway, right-click on the pathway handle, and select either the **Edit**→**Pathway Info Editor** or the **Edit**→**Pathway Editor** command.

To create a super-pathway, create a new pathway as above, using the **Pathway**→**New** command. In the Pathway Editor, instead of adding a linear sequence of reactions using the Segment Editor, or individual reactions using the Reactions commands, you can add whole subpathways by using the commands in the Pathways menu. Additional reactions or connections between subpathways can be added as needed in the usual fashion.

7.3.6.2 Pathway Info Editor

With the Pathway Info Editor, you can specify the class of pathways to which the pathway belongs. Note that the general class "Pathways" is chosen by default — make sure to uncheck it when you choose a more appropriate class. You may specify the primary name and synonyms for the pathway, and give a comment and citations for the pathway. You can create links to this pathway in other pathway databases. You can also specify that one or more of the reactions in the pathway are considered hypothetical, generally because presence of the enzyme, reactants, or products has not been experimentally demonstrated.

7.3.6.3 Pathway Editor Operations

The Pathway Editor contains two main display panes (see Figure 7.11). The left pane shows reactions that have been added to the pathway but have not yet been connected to other reactions. Reactions are listed by frame ID, EC number, and reaction equation. The right pane draws the connected reactions of the pathway. With the Pathway Editor, you can add new reactions to the pathway, connect together reactions in the pathway, and delete reactions from the pathway.

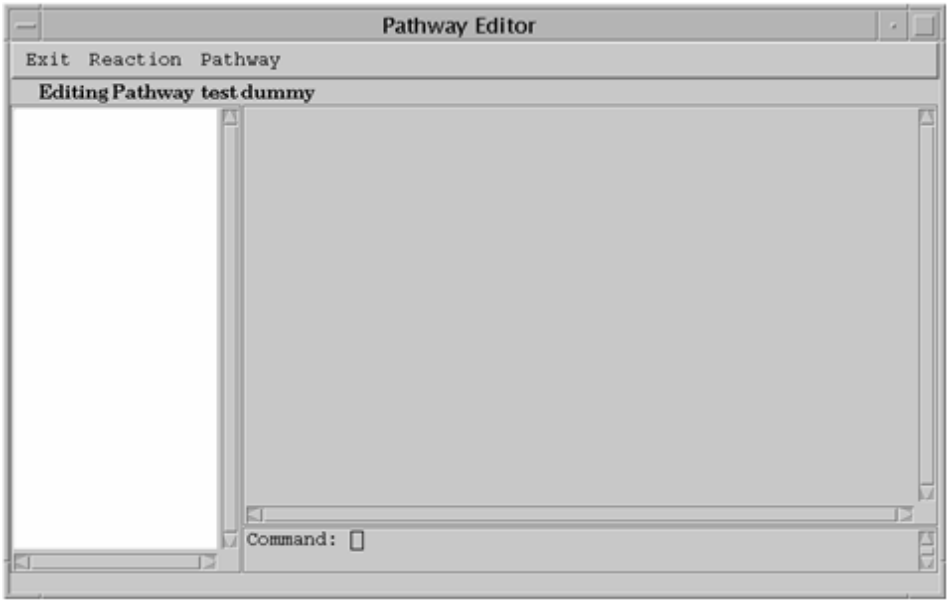


Figure 7.11: Pathway Editor

The simplest mechanism for creating pathway connections between reactions is as follows. First select a *predecessor* reaction (in either pane) by clicking on it (or on its terminal compound in the right pane). That reaction is highlighted in red; all reactions that can potentially follow it, based on substrate overlap, are highlighted in green. Click on one of the green *successor* reactions to indicate that the green reaction should directly follow the red reaction, or on the background to cancel the operation. The panes are updated to reflect the change. Until a reaction is actually linked to another reaction in the pathway, its direction is unspecified, so you may see more reactions highlighted in green than you would have expected (this can also be true if the reaction has commonly occurring substrates such as ATP).

Unconnected reactions from subpathways are displayed in the right pane. They can be manually connected to reactions in the superpathway or any subpathways(s). To keep the integrity of subpathways, reactions that originally came from subpathways are not allowed to be disconnected or deleted.

The Pathway Editor contains additional commands.

Exit Menu :

Keep Changes: Exit the Pathway Editor, transferring all changes made here back to the PGDB.

Abort Changes: Exit the Pathway Editor without transferring any of the changes made to the pathway back to the PGDB. However, if any new reaction frames were created or edited during the session, those changes will be maintained.

Reaction Menu: Many of the following commands require you to first select a reaction; those commands are also accessible by right-clicking on the reaction.

Add Reaction: Provides you with several alternative ways of specifying a reaction to add to the pathway. In a dialog window, you specify the reaction by EC number, by frame ID, according to the name of the enzyme that catalyzes the reaction, or according to the substrates of the reaction. In the latter case, the program finds the one or more reactions containing the specified substrates. If no reactions contain the specified substrates, the Reaction Editor is invoked to create a new reaction (see Section 7.3.7). The specified reaction is then added to the left pane of the Pathway Editor.

Add Reaction(s) from History: Pops up a list of all reactions on the Pathway Tools history list. Those selected are added to the pathway and appear in the left pane of the Pathway Editor. This offers a convenient way to add reactions to a pathway: find and display all desired reactions through the Navigator *before* invoking the Pathway Editor. Then the desired reactions are at the front of the history list and are easy to select.

Create New Reaction Frame: Invokes the Reaction Editor (see Section 7.3.7) to create a new reaction, which is added to the pathway.

Clone a Reaction Frame: You are prompted for the reaction frame (frame ID or EC number) that is to be copied and a new frame ID for the resulting copy. The new reaction frame is created as a copy of the specified frame and added to the pathway.

Add Connection: This command is the main focus of the Pathway Editor. It creates a connection (adjacency relationship) between two reactions in a pathway. After invoking this command, you will be instructed to click on the first (predecessor) reaction, and then the second (successor) reaction.

Delete Predecessor/Successor Link: Prompts for a reaction and pops up a list of connections between the reaction and its predecessors and successors. You can select any or all of these to delete. If any reaction becomes completely disconnected as a result of this operation, it is removed from the right pane and redrawn on the left pane.

Disconnect Reaction: Prompts for a reaction and deletes all predecessor and successor links to or from the reaction. The reaction and any others that become disconnected reappear in the left pane.

Delete Reaction from Pathway: Prompts for a reaction and removes it from the pathway. This does *not* delete the reaction frame from the PGDB.

Choose Main Compounds for Reaction: The Pathway Editor automatically infers the main and side compounds for each reaction in the pathway based on substrate overlap between reactions and other heuristics. Use this command if you want to specify different or additional main compounds for a reaction. Because reactant and product mains are specified separately, this command also provides a mechanism for specifying the direction of a reaction if the direction is ambiguous. Thus, this dialog also pops up on occasion when a reaction direction cannot be inferred directly from its predecessor/successor links.

Edit Reaction Frame: Prompts for a reaction and pops up the Frame Editor window for the reaction.

Pathway Menu :

Enter a Linear Pathway Segment: Invokes the Segment Editor to enter a linear pathway (see Section 7.3.6.4).

Guess Pathway Predecessor List: Guesses how the reactions within the pathway should be connected based on some simple substrate overlap heuristics. Any previously specified predecessor links are ignored. Although this command can save time when constructing simple pathways, it may produce incorrect results for more complicated branching pathways. Be sure to check the resulting pathway for accuracy.

Disconnect All Reactions: Removes all predecessor links between any reactions in the pathway. All reactions are moved back to the left pane, and the right pane is blank. This command is useful when a pathway is misconnected and it is easier to start rebuilding it from scratch rather than attempting to fix the links.

Invoke Relationships Editor: Brings up a window containing the Relationships Editor view of this pathway and its relationship to other frames in the DB. This editor is a very useful way to visualize the Web of slot-based relationships between all frames involved in one pathway.

Add Subpathway by Name: A dialog prompts you for a frame ID of a pathway. A menu shows the pathway with that frame ID that you can choose to add.

Add Subpathway by Substring: A dialog prompts you for a substring of the common name or synonym of the pathway. A menu shows all the relevant pathways, from which you can choose one or more to add.

Add Subpathway by Class: A cascading menu allows you to choose one or more metabolic pathways by first selecting one class from a menu of pathway classes, and then one or more pathways from a menu of all pathways in that class.

Delete Subpathway: A menu shows all the subpathways in the current superpathway. You can choose one or more subpathways to delete.

In addition, right-clicking on a compound name within the pathway display brings up a menu containing whichever of the following commands are applicable:

Add Link from/to Pathway: A pathway link indicates that a particular substrate in a pathway either comes from or feeds into some other pathway. It is displayed using an arrow connecting the pathway substrate to the name of the linked pathway. Links are one-way; because a link is created in the display of Pathway A from Compound X to Pathway B, it does not mean that a link should also appear in the display of Pathway B connecting Compound X to Pathway A. Use this command to create pathway links to or from the selected compound in the display for the current pathway. A cascading menu allows you to choose the pathway to which the link is to be created. The software itself determines the default directionality of the link, but you can change its direction by right-clicking the link arrow and selecting **Set Link Direction**. To delete a pathway link created in this fashion, right-click on the link arrow and select **Delete Link**.

Place this Compound at Cycle Top: This command is available for compounds within a cycle. Selecting this command causes the pathway to be redrawn with the indicated compound at the top of the circle.

Create Polymerization Link: This command is available when the software detects the possibility for a polymerization cycle. Invoking this command joins the selected compound to its other instantiation within the pathway by a polymerization link.

Delete Polymerization Link: Deletes a polymerization link created by the previous command.

Assign Polymerization Name: When two different instantiations of a compound exist in a pathway, connected by a polymerization link, this command can be used to assign the display name for each instantiation by selecting from a menu of possibilities. The list of possibilities is derived from the values of the slots COMMON-NAME, N-NAME, N+1-NAME, and N-1-NAME for the compound.

7.3.6.3.1 Pathway Editor Limitations In general, the order in which you specify links to generate a pathway is unimportant. However, in some complicated situations, specifying one link before another may introduce ambiguity in the partially constructed pathway. Depending on the situation specifics, one of several things might happen: the link

may be ignored, a dialog box may pop up asking you to disambiguate, or the pathway may be drawn in a bizarre arrangement. If any of these occur, try removing the offending link and adding the links in a different order. In general, links that introduce the least ambiguity should be specified first. For example, if our pathway consists of three reactions, R1: A = B, R2: B + C = D, and R3: D + C = E, and the first link we specify is from R2 to R3, then the direction of R2 remains ambiguous and we may see unpredictable results. If, however, the first link we specify is from R1 to R2, then we can infer that R2 must proceed from left to right and can safely add the link R2 to R3 later.

7.3.6.4 Segment Editor

With the Segment Editor (see Figure 7.12), you can enter a linear sequence of connected reactions in a faster manner than by adding reactions one-by-one through the Pathway Editor. Each reaction is specified using either its EC number — which when known provides a fast shorthand for the reaction — or using the reaction substrates. One segment can contain as many as seven reactions.

Each reaction is depicted in the Segment Editor by a straight arrow that connects the main substrates of the reaction, plus a curved arrow that connects the side substrates of the reaction. To specify each reaction in the sequence, either enter its EC number in the box to the left of the arrow, or enter the side and main substrates in the appropriate boxes above, below, and to the right of the arrow. Only one box is provided to enter each of the side reactants and the side products, respectively. Multiple side reactants (or products) can be entered by separating them with the "/" character.

Several types of correctness checking are performed by the Segment Editor. As soon as you enter an EC number or a substrate name, the Editor verifies whether they are defined in the PGDB or in MetaCyc. If they are unknown, pop-up windows indicate that condition. An incorrect EC number or substrate name can be altered by retyping it within its box.

In general, it is highly preferable to construct new pathways from existing reaction and compound frames in the PGDB or in MetaCyc. This approach minimizes data redundancy and data entry, minimizes typographical errors that could occur during redundant data entry, and facilitates comparative analyses by ensuring that the same object is always assigned the same unique ID in different PGDBs. Therefore, every effort should be made to find existing objects before creating new ones. If a chemical compound is not found using the first name you try, try again using other names for the compound, or using substring searches for a root of the compound name. If you still cannot find the compound within the PGDB, you should create it. Repeated searches and compound creation can be performed within the Compound Resolution Tool (Section 7.3.6.5). You can invoke that tool by clicking the **Search/Create** button in the dialog window that informs you a compound name is unknown.

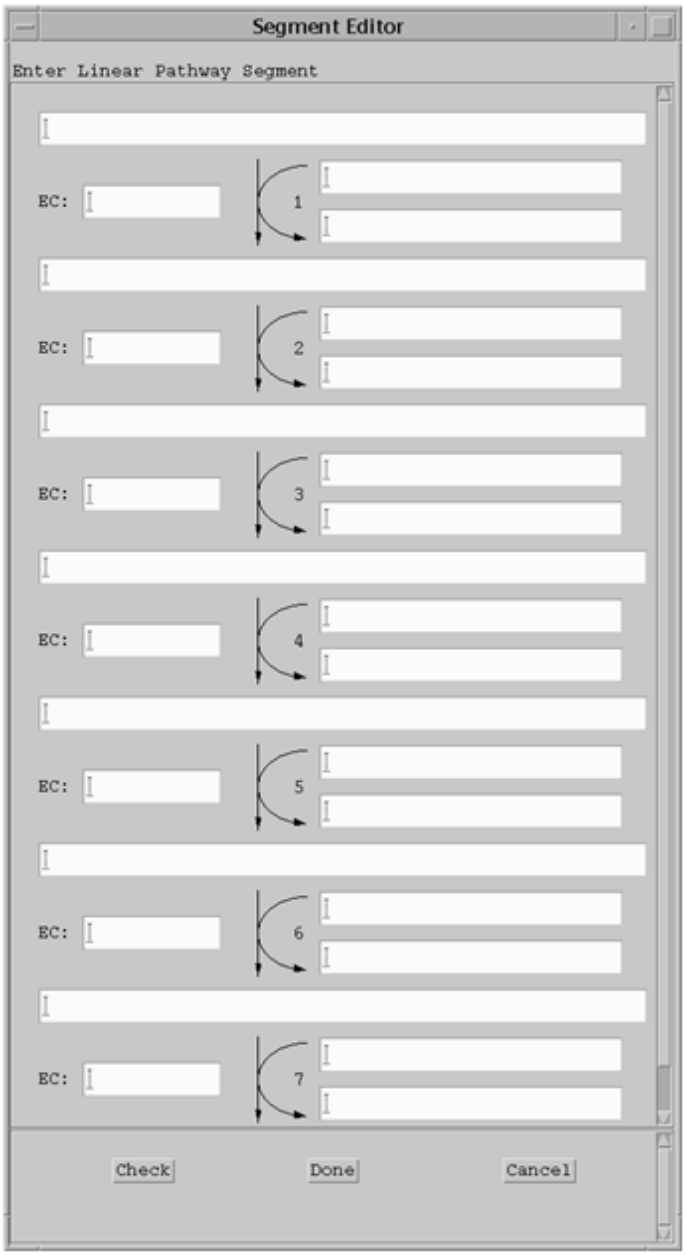


Figure 7.12: Segment Editor

Once you have successfully entered EC numbers and substrate names, you must perform a second phase of checking by clicking the **Check** button at the bottom of the Segment Editor. The checks performed in this phase include

- Checking consistency of EC number, reaction substrates, and reaction directions.
- Matching substrates for each specified reaction against reactions stored in the

PGDB. If a specified reaction is not found, the Reaction Editor (Section 7.3.7) is invoked to create it.

Reactions for which a problem is found are drawn in red; correct reactions are drawn in green. Modify red reactions to correct the indicated problem.

When all reactions are green, click the **Done** button to exit the Segment Editor. You cannot exit until you have used the **Check** button to perform checking. You can click **Cancel** at any time to abort the session, which will abort construction of the pathway segment.

7.3.6.5 Compound Resolution Tool

This tool is entered from the Pathway Segment Editor when a reaction contains compound names that could not be identified in the PGDB. A new dialog appears, presenting a choice of the names that were still unresolved or that were completely ambiguous, meaning that more than one compound contained the same string as its common name.

"Resolving a compound name" means matching it unambiguously to a compound frame in the PGDB by using successive searches to find the compound under different names or name fragments. For example, you might encounter the name "N2-succinyl-L-arginine" in an article, but a search under that name does not yield a match. You might then try searching using the substrings "succinyl" or "arginine," and find that the compound is indeed already in the PGDB under the name "N2-succinyl-arginine," and then add the synonym to the existing compound. To resolve a name, click on it, and enter an alternative synonym or substring. Click on **Exact Match** to search for a compound with exactly that name; click on **Substring Match** to search for a compound whose name contains the specified substring.

If no match was found, the string that was entered in the text line is transferred into the box that collects all the synonyms for this compound, ensuring that all the entered synonyms are retained, to save retyping them later.

If matching frames were found, they are mentioned in the Messages section of the panel. If multiple frames were found, a list pane allows selection of any one of the hits. Simultaneously, the selected compound is displayed in the lower half of the main Pathway Tools window. This should help you see whether the compound is the desired one. Two buttons exist for accepting the compound. Either the collected synonyms are transferred to the selected compound frame, or not. It can be useful to transfer the synonyms to allow future retrieval of this compound by these new names, which have previously failed, even though they ought to be valid names under which the compound should be findable.

At all times, it is possible to create a new compound frame from scratch, if you conclude that this compound truly does not exist in the PGDB. Compound creation transfers all the synonyms collected so far, so that you can avoid having to retype them. The first of

these synonyms, which was the name originally entered as part of the reaction equation, becomes the common name of the new compound frame. A pop-up window allows entering of a frame name, and the first synonym is again suggested as a suitable default.

7.3.6.6 Limitations of the Compound Resolver

Currently, compound frames that were modified by adding synonyms during compound resolution are not reverted to their unmodified state after you press the **Abort** button. Only frames that were created from scratch, both compound frames and the new reaction frame, are deleted in an abort.

The "Protein Complex?" button is not yet properly connected, so that its change does not percolate through automatically to the protein frame ID, to switch between the **MONOMER** and **CPLX** suffixes. You must type in the correct suffix. However, this button does control linking of the new frame into the right class at frame creation time (when **Done** is pressed).

7.3.7 Reaction Editor

The Reaction Editor facilitates the creation of a new reaction frame in a PGDB. Reaction frames contain links to frames for their substrates, which could be chemical compounds, proteins, or other molecules.

The literature often refers to the same chemical compound using many different chemical names. It is desirable that the PGDB not contain multiple frames that describe the same chemical compound, using different names. MetaCyc contains extensive lists of compound synonyms that will help locating an already-known compound, but in some cases, chemical names will be hard to find. The Reaction Editor contains a section for helping to locate existing substrate frames. A new substrate frame can be created, after all attempts to find an existing one have failed.

7.3.7.1 Invoking the Reaction Editor

To create a new reaction, select **Reaction→New** from the Reaction menu. An existing reaction can be edited by right-clicking on the reaction and selecting **Edit→Reaction Editor**.

At the top of the Reaction Editor, the conversion type of the reaction should be chosen first. The most common small molecule reactions will be of conversion type Chemical Reactions. The other conversion types available are

Binding Reactions: In binding reactions, no covalent modification of the substrates takes place, but the net effect is that one substrate non-covalently binds to or unbinds

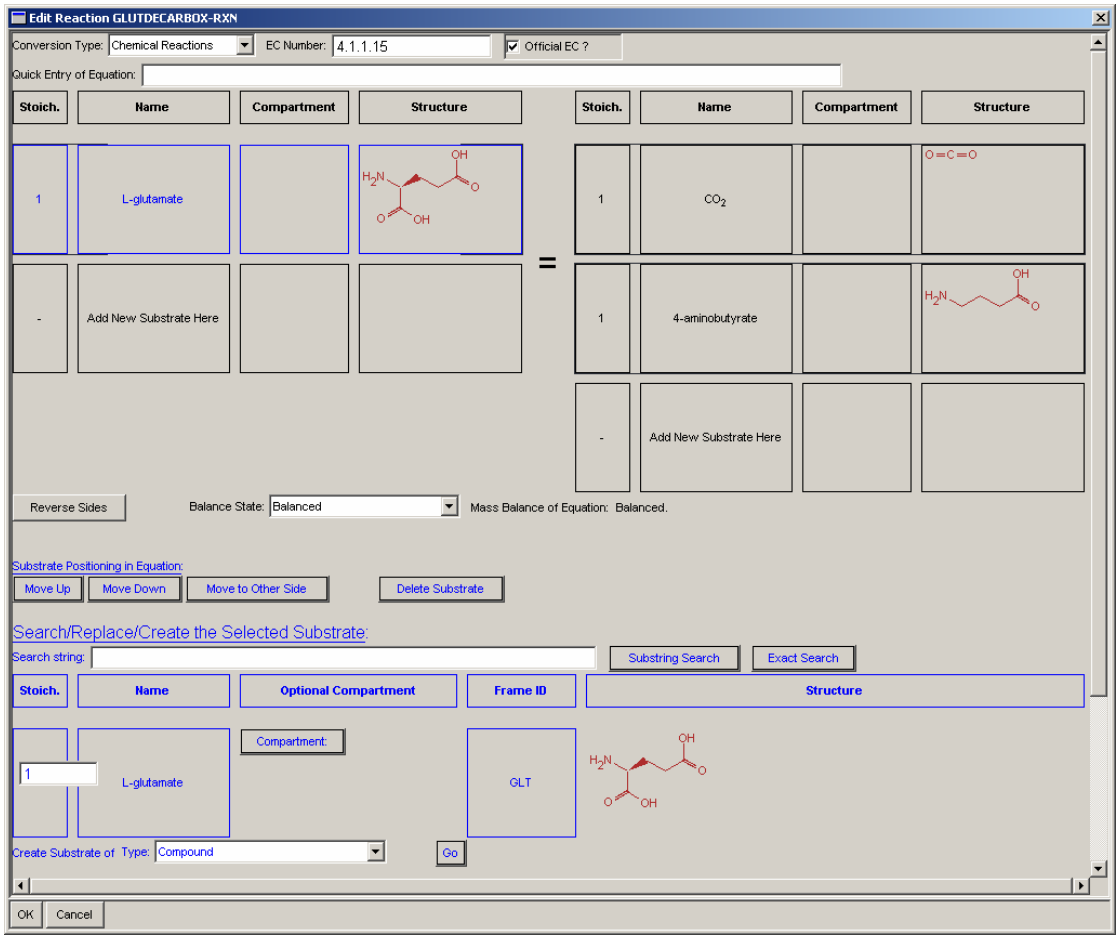


Figure 7.13: Reaction Editor

from another molecule via weak bonds (e.g., hydrogen bonds). This class includes protein complex assembly and dissociation reactions. In reactions of this class, each of the substrates maintains its original molecular identity, meaning reactions of this class affect intermolecular bonds, not intra-molecular bonds.

Transport Reactions: These are reactions in which at least one species is transported (passively or actively) across a membrane, into a different cellular compartment. The species may or may not be chemically modified in the course of the reaction.

Complex Processes: Each instance of this type represents a complex, multistep, black-box process. Details of the process may or may not actually be known, but are not described in this frame. Therefore, the entire process is considered to be a black box that we cannot see within. Examples of processes that might be represented in this manner are protein degradation, transcription, and translation. When it is not clear whether a given transition is caused by a single reaction or by a complex process, use the Unknown Conversions type.

Electron Transport Reactions: These are composite reactions that are assembled from 2 Redox Half Reactions. The resulting summary equation is shown, but its individual substrates can not be directly edited. The equation is inferred from the combination of the half reactions. Please see Section 7.3.7.4 for more details.

Redox Half Reactions: These are elementary reactions involving free electrons, capturing the redox transformation of a given substrate. They are not intended to be free-standing reactions, because free electrons do not occur in a cell on their own. Instead, the half reactions are building blocks that can be combined to form the composite Electron Transport Reactions. Please see Section 7.3.7.4 for more details.

Unknown Conversions: Instances of this type define conversions for which it is not known whether the transformation occurs as a complex process or as a single reaction step. An example is the conversion of an inactive form of a protein to an active form, when the exact chemical forms of the protein, and the nature of the transition from one form to the other are not known. For transformations that are known to be multistep, use the Complex Processes type instead.

Enter an EC number for the reaction if one has been assigned.

Next, an optional text box can be used for quick entry of a reaction equation. The text should include an equal sign ("=") to separate the left and right sides of the equation and plus signs ("+") to separate multiple substrates within a reaction side. After the equation text is entered and the cursor focus moved elsewhere (such as by clicking on another region of the editor), the Reaction Editor tries to find compound frames corresponding to each substrate name. Those substrates that are found are then shown in the table-based representation of the reaction. Substrate names that did not result in an unambiguous hit are displayed within double quotes, and should be examined and resolved by the user. The quick entry box for the equation is not updated dynamically thereafter — it is provided only for initial entry of the reaction. All further editing of the reaction should occur in the table below.

The main section of the Reaction Editor is a table that shows the substrates of the reaction, partitioned into the left and right sides. Each substrate has four columns of information that pertain to it, namely the stoichiometric coefficient of the substrate (the default is 1), the name of the substrate, the compartment (for which the default is no compartment, which denotes the cytosol), and the molecular structure (if one exists for the substrate and if it fits into the allotted table space).

At all times, an additional blank substrate is added at the bottom of both the left and right sides, which is where new substrates can be added to the reaction. One of the substrates can be chosen for editing, by left-clicking on it. The rectangular outline will be highlighted in blue to indicate which substrate is currently selected. The blue section below the substrate table is used for editing this selected substrate. This section, called

"Search/Replace/Create the Selected Substrate", allows you to modify properties of the substrate as it used in the reaction, as described further in Section .

Several buttons allow you to rearrange the order of substrates in the reaction equation. Underneath the main table, the button **Reverse Sides** allows the left and right sides of the equation to be swapped in one operation. Pertaining to the blue, selected substrate, the buttons **Move Up**, **Move Down**, and **Move to Other Side** allow the substrate to be moved accordingly. Finally, the substrate can also be deleted from the reaction. This operation will not delete the substrate from the PGDB, but will merely disconnect it from the reaction being edited.

7.3.7.2 Editing Reactions

The Reaction Editor dialog requests several pieces of information:

Conversion Type: See Section 7.3.7.

EC number: If the EC number for the reaction is known, enter it. A valid entry consists of four consecutive numbers (or dashes, standing for currently unspecified categories), which are separated by one dot each (e.g., 6.3.2.24).

Official EC?: Mark the checkbox only if the reaction is 100% identical to the official EC reaction. You can specify multiple reactions with the same EC number, but only one reaction in the DB can be the official EC reaction. This is useful when the EC reaction is a general one, and you need to specify a reaction that constitutes a special case of the general reaction.

Quick Entry of Equation: Optionally enter the reaction equation in the form A + B = C + D. The "+" signs separate individual substrate names, and the "=" sign separates the left and right sides. The operators "+" and "=" must be surrounded by spaces. Any stoichiometric coefficients should precede the substrate and should be surrounded by spaces. Specify transport reactions that involve translocation between periplasm and cytoplasm by suffixing the name of the substrate that is located within the periplasm with [periplasm]; for example, A [periplasm] + B = C + D. When displayed, such reactions show the translocation graphically.

Mass Balance of Equation: Every time the reaction is edited, the current mass balance is displayed here. If all the reaction substrates contain a chemical formula, and the reaction is unbalanced, the difference in atoms between the reactants and products is shown. A reaction may be unbalanced because of an error in the reaction equation or an error in a compound structure. It is still fairly common for reactions to be unbalanced by just a few hydrogens, because many compounds are not stored in the correct protonation state for cellular pH levels. We have an ongoing effort to improve such compound structures.

Balance State: This field can usually be ignored. It is for selecting the appropriate state regarding what is known about the reaction equation in terms of balancing all atoms on the left with those on the right side. This is important primarily if the reaction cannot be balanced, for example because the full equation is unknown at the time. The default is "State Unknown" for when a reaction is created. Upon exiting the reaction editor by the **OK** button, the value of the corresponding BALANCE-STATE slot may be set to "Balanced" by the final mass balance check that is run at exit time. However, if the reaction is not balanceable, a curator may wish to record a more detailed reason for this deficiency. If it is known that the reaction equation is incomplete (due to incomplete biochemical research), "Unbalanced Incomplete" should be selected. If the reaction uses compound classes where R-groups may stand for residues of differing mass, "Unbalanced Differing R Groups" can be chosen. Such cases can occur when dealing with polymerization pathways. If it is simply unclear why a reaction is not balanceable, and if this is unlikely to be fixable any time soon, it can be flagged with "Unbalanced Unknown".

Spontaneous?: Mark the checkbox only if the reaction occurs spontaneously under physiological conditions, that is, no enzyme is required for the reaction to occur.

Orphaned reaction?: For some curation projects, it may be desirable to mark reactions for which no gene sequence is known for any enzyme that might be catalyzing the reaction. This can help to focus experimental research aiming to find such sequences.

Citations: Enter as many as four citations, which can be any combination of PubMed IDs, citation IDs that are already in the current database, or citation IDs for citations that you are ready to put into the database.

Comment: A comment about the reaction can be entered here.

OK: After all editing is completed, click the **OK** button to execute all PGDB updates, and to exit the Reaction Editor. At that stage, a check for potentially duplicate reactions is performed, to alert you if this is the case. You can then choose to simply keep the edited reaction, or to replace it with one that is imported from MetaCyc, or to delete the edited reaction.

Cancel: Clicking this button exits the editor without causing any PGDB changes.

7.3.7.3 Search/Replace/Create the Selected Substrate

The selected substrate of the reaction equation can be edited by replacing it with a different substrate, or by creating a new substrate frame if a suitable one does not exist yet. If you search thoroughly for a substrate first, before creating one, you can avoid creating duplicate frames.

A search string can be entered into a text box, and two buttons allow triggering either a substring search or an exact search in the current PGDB, and in MetaCyc if nothing was found in the current PGDB. If a search turns up no hits, a small popup window will say so. If the search turns up exactly one hit, the selected substrate will be replaced by this new one. If there are several potential candidates, a separate dialog panel will pop up to display detailed information about the candidates such as their synonyms, molecular weights, and molecular structures, and which allows the user to select one of the candidates.

If no suitable substrate was found, it is possible to create one. First, the substrate type has to be chosen with a selector. This choice will influence the suggested classification of the substrate and will invoke the appropriate type of editor. Clicking the **Go** button will create the substrate frame and invoke the Protein Editor for proteins and protein complexes, or the Compound Editor for compounds and also for all class frames.

Allowing creation of substrate class frames is one of the new features in the Reaction Editor. More information on guidelines regarding when and how to use classes can be found in the *Curator's Guide*, but as a short summary, some reaction equations are formulated as general transformations that apply to many different substrate instances. Those instances will differ in a systematic way in one part of the molecular structure, but share the same functional groups elsewhere. In the Pathway Tools schema, we represent generic substrates as class frames, which may show a molecular structure of the shared part. For example, the hydrolysis of beta-D-glucuronosides (reaction 3.2.1.31, BETA-GLUCURONID-RXN) has the class |Beta-D-Glucuronides| on the left side and the class |Alcohols| on the right side. Both of these substrate classes contain an R-group in their molecular structures, which does not change during this reaction. The naming of class frames should follow specific guidelines that are spelled out in the *Curator's Guide.* If you choose to create a class frame, a popup text entry box will ask for the frame ID. Thereafter, the Compound Editor is invoked, to allow entry of various synonyms and other information. The class frame can also be reclassified by means of the **Class** button at the top.

In this section of the Reaction Editor, two more properties can be set for the selected substrate. The stoichiometric coefficient can be set to something other than the default, which is 1, and the cellular compartment can be chosen by clicking on the corresponding button. This will pop up the classification hierarchy of cellular spaces in the Cell Component Ontology, from which one space can be selected.

7.3.7.4 Electron Transport and Redox Half Reactions

Electron Transport Reactions (ETRs) are used to represent the electron transport processes that occur in membrane-associated enzyme complexes, involving membrane-bound electron carriers. These processes are important for energy generation by the cell, by using high-energy electrons carried in certain substrates to increase the proton gradient across

the membrane. A graphical diagram for the ETR visually conveys the key features of such processes, like the direction of the electron flow and the cell compartment locations of where the substrates are transformed.

To capture the complex processes in at least some detail, and to reuse recurring subreactions, ETRs are represented as composite reactions that are composed of Redox Half Reactions (RHRs). The substrate making the connection between 2 chosen RHRs is e^- (the free electron), which is a connection internally to the ETR, and is not visible from the outside. The ETR will have its left and right side substrates (the outside) inferred from the 2 chosen RHRs.

Thus, to construct an ETR, it is necessary to first have the needed RHRs available. They can be created and edited by selecting the Redox Half Reaction conversion type. In this mode, the Reaction Editor panel shows a text entry box at the top that is specific for RHRs: the Standard Reduction Potential. A numerical value in units of V (Volts) at standard conditions (25 degrees C, pH 7.0, 1 atm) should be entered here. The value is needed for determining the direction of electron flow correctly. An RHR needs to have the electron e^- appear on the left side, by convention, so that the reduction potential corresponds to the reduction process. In other words, an RHR should be written in the direction:

Ox. + e^- (Red.

If protons are involved in the reduction reaction, they should also be explicitly included in the equation. The substrates can have compartment information added, though the convention is that the electron e^- never has any compartment association.

In the currently supported ETRs, there is a fundamental distinction made between 2 types of RHRs: the ones involving quino-factors that are membrane-bound, and the ones that do not involve quino-factors. The quino-RHRs also do not have a compartment association for the quino-factor substrates, as they are inferred to reside in the membrane itself. However, proton substrates can have a compartment associated.

Once the RHRs for an electron transport process are available, the overall ETR can be constructed by selecting the Electron Transport Reaction conversion type. This is a special mode, which is quite different from the other modes. It offers 3 selectors to allow a choice of the subreactions, which when taken together, form the overall transformation. The resulting left and right substrate sides, which are computationally inferred, are shown in the central table region of the Reaction Editor, but these substrates are not directly editable in the usual way. Always 1 quino-RHR and 1 non-quino-RHR have to be chosen. Optionally, 1 proton transport reaction can also be included as an additional subreaction, because some electron transport complexes can extract so much energy from the transformation that they can actively pump additional protons, in a so-called "vectoral" mode.

To get started with curating ETRs, it is best to examine the existing examples provided in EcoCyc. Some examples are EC# 1.7.99.4 and EC# 1.6.5.3 .

7.3.8 Chemical Compound Editor

Chemical compounds are the small-molecule substrates for metabolic reactions, as well as activators, inhibitors, or cofactors for enzymes. To edit an existing compound, right-click on a compound handle and choose **Edit→Compound Editor**; to create a new compound, click **Compound→New**.

With the compound editor (Figure 7.14) you can specify the class of chemicals to which the compound belongs, specify the primary name and synonyms for the compound, and add a comment and citations for the compound.

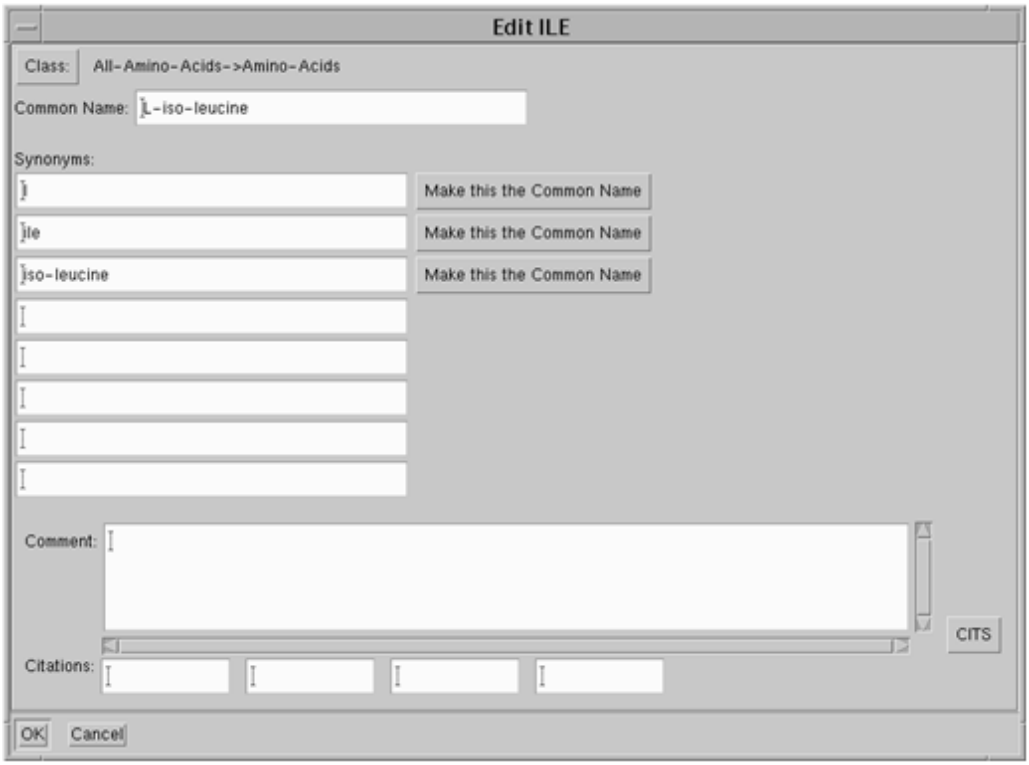


Figure 7.14: Chemical Compound Editor

7.3.9 Marvin Compound Structure Editor

The Marvin chemical structure editor is rich in features and easier to install than the JME editor described in the next section. Key advantages of Marvin are

- Support for stereochemistry with wedged bonds

- Smarter in handling the way the Navigator represents aromatic bonds, so it will not need the OpenBabel software, simplifying the installation, compared to the JME editor
- Atom aliases allow broader flexibility in naming atoms: Example aliases include R, R1, and R2. Even frame IDs for proteins, such as ACP, can be used to represent modified protein structures.
- Easier to obtain

7.3.9.1 Installation

Marvin is not developed by SRI International, and is not included in the Pathway Tools distribution. Marvin must be obtained from ChemAxon and then installed. Please consult `http://bioinformatics.ai.sri.com/ptools/installation-guide/released/marvin.html`.

7.3.9.2 Structure Editing

When the Marvin editor is invoked by right-clicking on a compound handle and selecting **Edit→Marvin Compound Structure Editor** (see Figure 7.15), a Web browser window pops up, in which the Marvin applet will run.

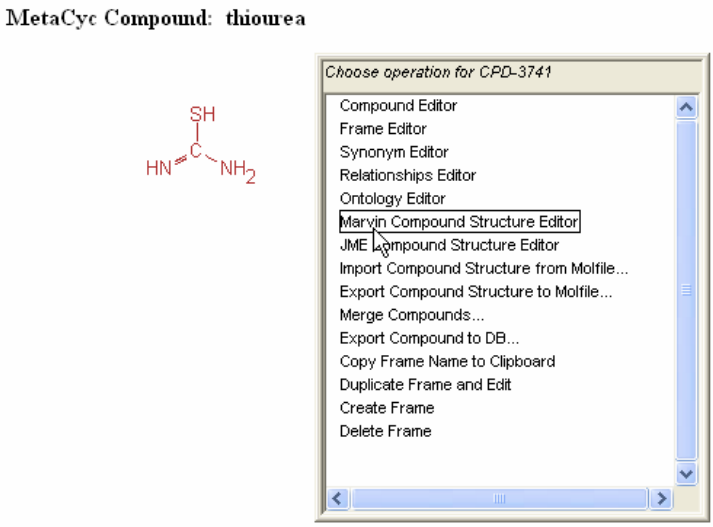


Figure 7.15: Invoking the Marvin Structure Editor

The Pathway Tools program starts a local Web server (currently on port 1557) that is used for transferring data to and from Marvin. The structure of the compound (if any) will show inside the Marvin panel, as shown in Figure 7.16.

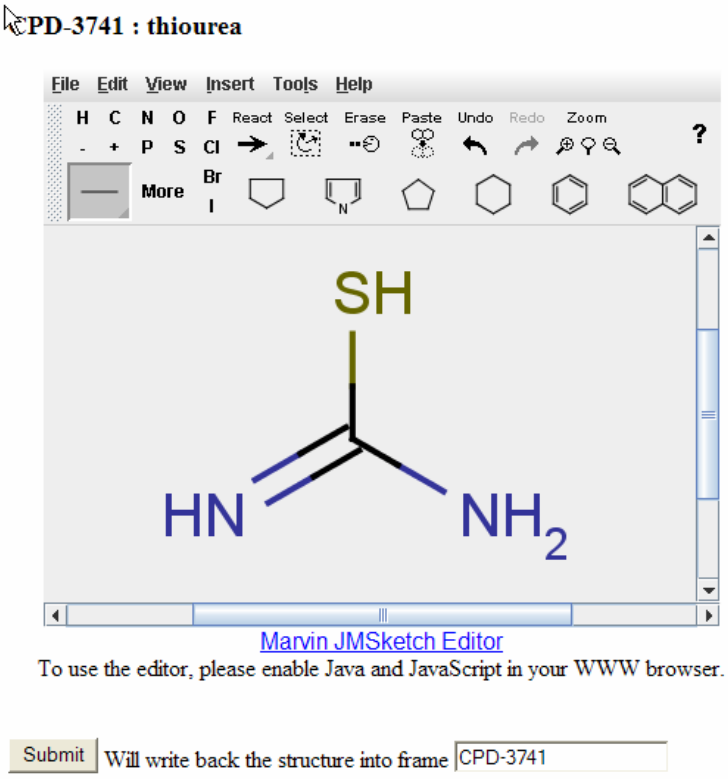


Figure 7.16: Marvin Compound Structure Editor

To add to an existing structure or to start drawing from scratch, it is necessary to first select the appropriate tool by clicking on one of the square boxes lining the drawing area at the top. This can be one of several bond types (single, double, triple, and wedged), one of several prefabricated ring systems, or an element type.

By moving the cursor in the drawing area and over existing parts of the compound structure, an attachment location can be selected. Marvin will highlight the selected location by temporarily drawing a small circle around it. If an atom is highlighted, a new bond or ring system will be attached to this atom by a left-click on the mouse. Usually, the default suggested geometry for the new bond is acceptable. If it needs to be changed, it is possible to rotate into a different geometry by not releasing the left mouse button until a satisfactory geometry has been reached.

An element can be converted to another by clicking on the desired element in the tool box and then selecting it. The tool box labeled "More" allows bringing up a large element panel with an entry box at the bottom, where atom aliases can be entered, for example,

for indicating an R-group with "R" (see Figure 7.17). Charges on an atom can be changed with the buttons labeled "+" and "-". By clicking several times on the atom, all the charges can be iterated through. A ring system can be fused alongside the highlighted bond when a prefabricated ring system was selected and is dragged over a bond.

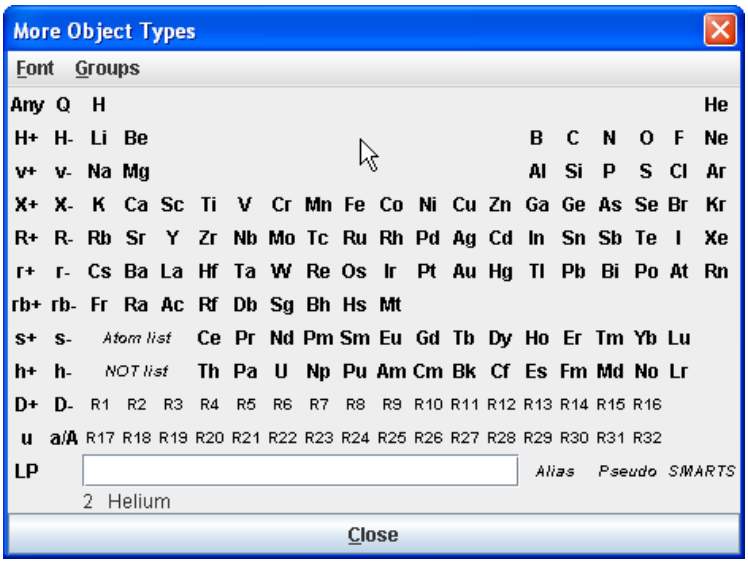


Figure 7.17: Marvin Editor: More Object Types window

Stereochemistry is supported by using the two wedged bond types for indicating whether a bond on a stereo center is pointing "up" (a solid wedge) or "down" (a hashed wedge), relative to the drawing plane. If neither wedge is chosen, the assumption is that the bond is oriented flat in the drawing plane.

If a drawing mistake is made, it can be undone by clicking the "Undo" button. It is possible to close the Web browser window entirely to start over, before submitting. Individual atoms and bonds can be deleted by clicking on the "Erase" button, and then on the parts of the compound structure that need to be deleted.

After the structure is fully edited, click the "Submit" button below the Marvin panel to cause the new structural information and coordinates to be sent to Pathway Tools. The Navigator window should now show the submitted and updated compound, and the PGDB will have been modified accordingly. The PGDB should be saved for the changes to be permanent.

It is possible to submit the structure information to a different compound frame than the one used for invoking Marvin, by editing the frame ID in the text entry box next to the "Submit" button (bottom of Figure 7.16) before actually submitting. This approach is useful when a relatively large structure needs to be added to a compound frame, which differs only slightly from some other structure. You can accomplish this by invoking Marvin on the structure that is to be copied, making the modifications, and changing the

frame ID next to the submit button to the ID of the destination frame before submitting the structure back to Pathway Tools.

7.3.9.3 Limitations

Although the interaction with Marvin is generally smooth and intuitive, there are a few known problems:

Lost aromatic heteroatom hydrogens: Differences in the way Marvin and Pathway Tools recognize aromaticity may cause implied hydrogens on aromatic heteroatoms to get lost. In some cases, Pathway Tools appears to not infer hydrogens correctly where it should. An example of such a compound is indole, which needs a hydrogen atom on its nitrogen atom. The work-around for this problem is to explicitly draw a single bond from the nitrogen, and to then change the element type at the other end of the bond to an H. Once the structure is submitted to Pathway Tools, the explicit bond to the explicit hydrogen will then be shown correctly and taken into account for computing the empirical formula and molecular weight.

No coordination bond type: A few compounds in BioCyc contain a coordination type bond to a metal atom. Marvin does not have a coordination bond type, and when such a compound is edited, the Pathway Tools coordination bonds will be converted to single bonds. To change the bond type back to a coordination bond, use the old Pathway Tools Compound Structure Editor.

7.3.10 JME Compound Structure Editor

JME is a popular Java applet for editing chemical structures. Its key advantages are

- Support for stereochemistry with wedged bonds
- Ease of use

7.3.10.1 Installation

Before this editor can be used, two pieces of external software that are not included in the Pathway Tools distribution must be obtained and installed. Please consult `http://bioinformatics.ai.sri.com/ptools/installation-guide/released/jme.html`.

7.3.10.2 Structure Editing with JME

When the JME editor is invoked by right-clicking on a compound handle and selecting **Edit→JME Compound Structure Editor**, a Web browser window pops up, in which the JME applet will run. The Pathway Tools program starts a local Web server (currently on port 1557) that is used for transferring data to and from JME. The structure of the compound (if any) will show inside the JME panel. To add to an existing structure or to start drawing from scratch, you must first select the appropriate tool by clicking on one of the square boxes lining the drawing area to the left and top. This can be one of several bond types (single, double, triple, and wedged bonds), one of several prefabricated ring systems, or an element type. By moving the cursor in the drawing area and over existing parts of the compound structure, you can select an attachment location. JME will highlight the selected location by temporarily drawing a small box around it.

You can attached a new bond or ring system to the highlighted atom by left-clicking the mouse. Usually, the default suggested geometry for the new bond is acceptable. If it needs to be changed, it is possible to rotate into a different geometry by not releasing the left mouse button until a satisfactory geometry has been reached. An element can be converted to another by selecting it and then clicking on the desired element in the tool box. Charges on an atom can be changed with the button labeled +/-. By clicking several times on the atom, all the plausible charges can be iterated through. If the cursor highlights a bond, you can switch its bond type by left-clicking the mouse, or you can fuse a ring system alongside the highlighted bond.

Stereochemistry is supported by using the two wedged bond types for indicating whether a bond on a stereo center is pointing "up" (a solid wedge) or "down" (a hashed wedge), relative to the drawing plane. If neither wedge is chosen, the assumption is that the bond is oriented flat in the drawing plane.

If a drawing mistake was made, it can be undone by clicking the UNDO button. However, this appears to be only a one-step undo. It is always possible to close the Web browser window entirely to start over. Also, the CLR button erases the entire structure from the drawing area. You can deleted individual atoms and bonds by clicking on the DEL button, and then on the parts of the compound structure that need to be deleted.

After the structure is fully edited, click on the submit button below the JME panel to send the new structural information and the coordinates back to Pathway Tools. The Navigator window should now show the submitted and updated compound, and the PGDB will have been modified accordingly. If the compound looks fine, save the PGDB so that the changes will not be lost.

It is possible to submit the structure information to a different compound frame than the one used for invoking JME, by editing the frame ID in the text entry box next to the submit button before actually submitting. This approach is useful when a relatively large structure, differing only slightly from some other structure, needs to be added to

a compound frame. To do this, invoke JME on the structure that is to be copied, make the modifications, and change the frame ID next to the submit button to the ID of the destination frame before submitting the structure back to Pathway Tools.

7.3.10.3 Limitations of JME

While the interaction with JME is generally smooth and intuitive, there currently are a few known problems:

Lost aromatic heteroatom hydrogens: Both JME and Pathway Tools recognize aromaticity and display implied hydrogens on heteroatoms. However, the information about the implied hydrogens is not sent by JME back to Pathway Tools. In some cases Pathway Tools appears to not infer hydrogens correctly where it should. An example of such a compound is indole, which needs a hydrogen atom on its nitrogen atom. The work-around for this problem is to explicitly draw a single bond from the nitrogen, and to then change the element type at the other end of the bond to an H. To do the latter, select the X element type button in JME, which will pop up a small little text entry box that would allow changing the element type to other unusual elements. By default, it is already set to H. Once the structure is submitted to Pathway Tools, the explicit bond to the explicit hydrogen will then be shown correctly and taken into account for computing the empirical formula and molecular weight.

No coordination bond type: A few compounds in BioCyc contain a coordination type bond to a metal atom. JME does not have a coordination bond type, and when such a compound is edited, the coordination bonds will be converted to single bonds. To change the bond type back to a coordination bond, use the Pathway Tools Compound Structure Editor.

Bond lengths are not adjustable: Newly drawn bonds in JME come in only one standard length. Their angles can be hand-tuned, but not their length.

Element X lock-up: It appears that after clicking on the X element type button and editing the element name from H to something else, it is no longer possible to change it to a different element. You must exit the Web browser completely (not just close the current window), before this behavior is reset.

7.3.11 Compound Duplicate Checker

The duplication of existing compounds within the database has many detrimental effects and should be avoided. A Compound Duplicate Checker program is run in the background whenever a user creates a new compound/structure or edits an existing one.

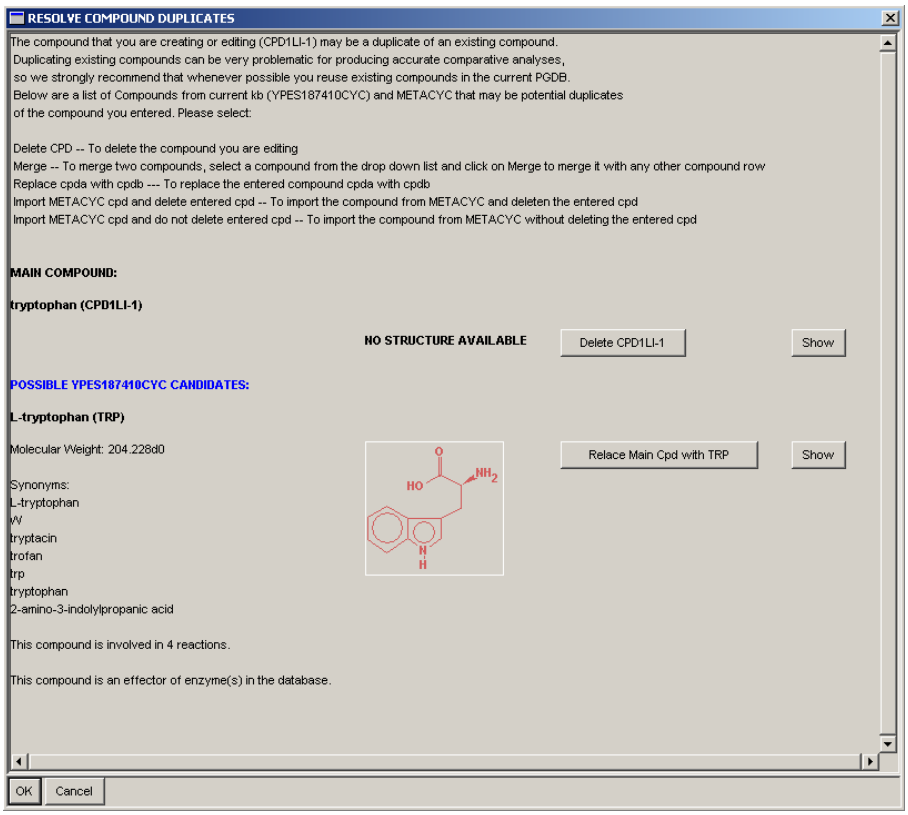


Figure 7.18: Resolve Compound Duplicate

This program will search through all the compound frames in the current database and in MetaCyc to see if it can find duplicates for the entered compound/structure. If any potential duplicates are found, the program will display these frames in a separate popup window with the following information (if available):

- Molecular Weight
- Synonyms
- Structure
- Number of reactions that use the compound (in the current database)Whether the compound is an effector of enzyme(s) in the current database

For each duplicate displayed you are given the following options:

Show: Displays the compound frame in the Pathway Tools navigator window.

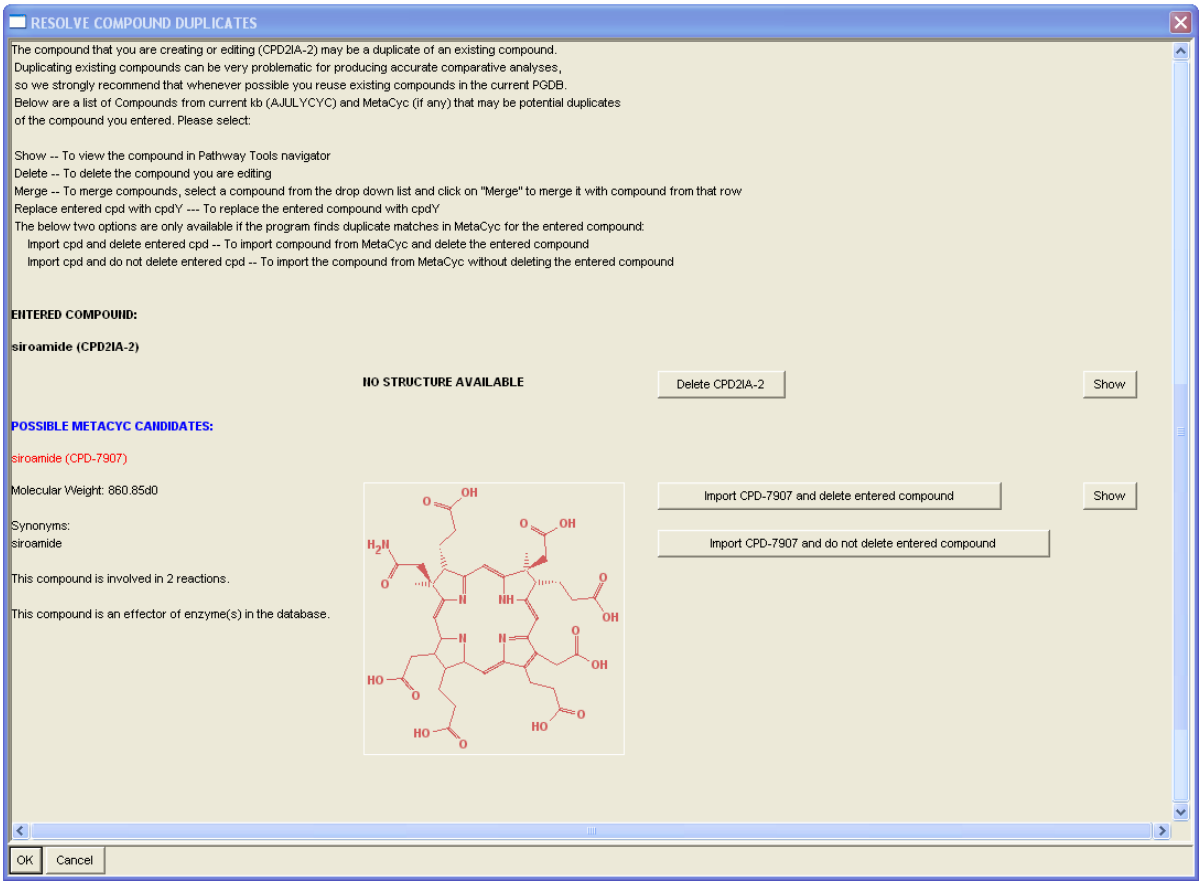


Figure 7.19: Resolve Compound Duplicate

Delete: Deletes a compound (this option may not be available if the compound is involved in any other reactions in the database or is an effector of enzymes).

Merge: Merges two compounds. To merge compounds, select a compound from the drop down menu and click on Merge to merge it with the compound from the row.

Replace Entered cpd with X: Replaces the entered compound with compound X and deletes the entered compound.

If the entered compound/structure is not found in the current database but is found in MetaCyc, the program displays the list of duplicates found in MetaCyc and gives the following additional options:

Import X from MetaCyc and delete the entered compound: Imports the compound from MetaCyc to the current database and deletes the newly entered compound.

Import X from MetaCyc without deleting the entered compound: Imports the compound from MetaCyc without deleting the newly entered compound.

Once you click "OK" to exit the Compound Duplicate Checker window, you must save the database in order to retain the changes you just made. At any point, you can click "Cancel" in the "Resolve compound duplicates" window to exit the duplicate resolver. This will automatically undo any changes you have already made within the duplicate checker.

7.3.12 MDL Molfile Import/Export

The 2D structure of a compound can be exported and imported to and from the widely used MDL Molfile file format, by right-clicking the compound handle in the Pathway/Genome Navigator window and selecting menu items **Edit→Export compound structure to molfile...** and **Edit→Import compound structure from molfile...** .

7.3.13 Regulation Editor

Regulatory interactions can take several different forms. A transcription unit can be regulated by transcription factor binding or by attenuation, an enzyme can be modulated by small molecule activators or inhibitors, and protein production or degradation can be regulated by other proteins or small molecules. In addition, it may be known that some factor regulates a gene product in some way, but there may be no information available about the nature of the regulation. Enzyme modulation interactions can be created and edited as part of the Protein Editor, and regulation of transcription initiation can be edited as described in Section . Additional regulatory interactions can be created using the command **Create→Regulatory Interaction**, accessible from the top-level **File** menu, or any right-click **Edit** menu. In the Regulatory Interaction Editor, specify the type of regulation (if no information is available, the type should just be Regulation) and the object (gene product, gene, transcription unit, reaction, etc.) that is being regulated. Depending on the selected regulation class, the software may automatically convert the regulated object to a related object of the correct type. Other available fields will depend on the particular regulation class selected, but the user will have the opportunity to indicate the regulator, whether the mode of regulation is activation, inhibition or unknown, as well as comments, citations, evidence codes, etc. Note that the mode of regulation applies to the resulting effect on the gene product or its activity – if some regulator promotes degradation of a protein, the mode should be inhibition, because it reduces the availability of the gene product, and not activation, even though it activates the degradation process.

Lists of regulators and regulatees are included in the Navigator displays for many objects. To edit or delete an existing regulatory interaction, select the appropriate command from

the right-click **Regulatory Interaction→Edit** menu.

7.3.14 Publication Editor

Figure 7.20: Publication Editor

When a PGDB frame cites a publication that does not have a PubMed ID, the reference for that publication must be defined as a publication frame within the PGDB. Every such publication frame must have a unique identifier (ID), such as SMITH95. That unique ID is used to refer to the publication frame in other editors. For example, to cite the article by Smith you can enter the text "[SMITH95]" in the Citations field within the protein editor. The publication editor can be used to create new publication frames, and to find the unique ID of publication frames when you do not know the unique ID.

It is not necessary to manually define a publication frame for publications that have a PubMed ID. Instead, use the **File→Import→Citations from PubMed** command to download the information directly from NCBI.

7.3.14.1 Creating New Publication Frames

You can create new publication frames in two ways.

1. From an editor such as the protein editor, type the new unique ID for the publication frame you want to create in a citation field (e.g. [JONES98]) and then click on a

different field. A dialog window will show up and give you two choices: (1) search for existing publications or create a new publication, or (2) return to the protein editor. If you selection option (1), the Publication Editor (Figure 7.42) will be invoked to accept the definition of the new publication. From the right-click menu of most objects, select **Edit→Create frame→Publication**. The Publication Editor pops up to accept the definition of the new publication.

2. Within the publication editor, enter the identifier that you will use to cite the publication, for example, **JONES98** (use all uppercase, with no spaces or punctuation). Then enter other information such as author, title, and source (journal, volume, and pages) in the appropriate fields, and click **OK** when done.

7.3.14.2 Finding and Editing Existing Publication Frames

If you are inside an editor such as the protein editor and you want to find the unique ID for an existing publication frame, simply enter a random citation name such as "[XYZ99]". A window will appear that gives you two search choices: (1) search for existing publications or create a new publication, or (2) return to the protein editor. If you select option (1), the Publication Editor appears. Type in the title or the surname of an author, and click **Search for Existing Publication Frame**. A menu of matching publication frames appears.

When you close the dialog, either by clicking **OK** or by selecting an existing publication from the provided menu, the ID of the publication frame is printed both to the listener pane and to the terminal window, so that you can use it in your citations.

Another way to modify an existing PGDB publication is to right-click on a citation handle in the References section at the bottom of any Navigator display page, for example, the "Smith95" at the front of the reference, to invoke the publication editor on that publication.

7.4 Editing Examples

The following examples of editing tasks illustrate the procedures involved.

7.4.1 Changing a Gene's Functional Annotation

Imagine that you have discovered a new function for a gene that is currently annotated as an ORF in the PGDB, and you want to alter the gene frame to reflect the new function you have assigned. Perform the following steps:

1. Search for the gene in the Navigator (**Gene→Search by Substring**) to display the gene (gene yaaA in B. subtilis is used here as an example).

2. Right-click on yaaA in the title bar, and select **Edit→Gene Editor** to invoke the Gene Editor.

   Imagine that the gene product is a cell-division protein that forms cytoplasmic filaments because of its similarity to the E. coli gene cafA. First, change the gene name to cafA by erasing yaaA in the Common Name field and typing "cafA" in its place.

3. To keep the name yaaA as a synonym for this gene, type "yaaA" in the first Synonyms field.

4. Change the classification for the gene by clicking the **Class** button. A window pops up showing the gene classification hierarchy. Expand the "cell processes" class and select "cell division". Other terms may be selected also. When done, click **OK**.

5. To insert a history note on this gene to record the rationale for our change, click on the Add History Note button at the bottom of the dialog box. Enter the text of the note in the pop-up window (e.g., "Function as cytoplasmic filament protein inferred due to very significant (P=10E-50) Blast hit to E. coli cafA") and click **OK**.

6. Click **OK** in the Gene Editor dialog to exit. The gene is redisplayed with its new name.

7. To alter the name assigned to the gene product, right-click on "YaaA" in the line showing the gene product, and select **Edit→Synonym Editor** to invoke the Synonym Editor. (You could have used the Synonym Editor to change the gene name, too, but could not have added the history note that way.)

8. Change "YaaA" in the Common Name field to "cytoplasmic filaments", and click **OK**.

7.4.2 Changing the Annotation for an Enzyme Gene

The preceding example was relatively simple because the gene in question coded for a structural protein rather than for an enzyme. Enzymes are more complex to encode in a PGDB because we use a combination of frames to encode the function of an enzyme.

Imagine that you want to change the annotation of the *B. subtilis* gene *yhfR* from the product phosphoglycerate mutase to the product phosphoglycerate kinase. Perform the following steps:

1. Search for the gene in the Navigator (**Gene→Search by Substring**) to display the gene yhfR.

2. Use the preceding procedure to change the gene name (if desired) and to change the protein name.

3. Remove the reaction to which this gene product is currently connected. In the gene window, right-click on the reaction equation "3-phosphoglycerate = 2-phosphoglycerate" and then select **Edit→Detach Enzyme(s)**.

4. A list of enzymes for the reaction appears. Select the appropriate one (currently named "putative phosphoglycerate mutase (glycolysis)", but if you have changed the name, the new name appears in the menu instead) and click **Use These Values**. The reaction (now with only one enzyme) is displayed in the Navigator window.

   To connect the product of yhfR to the reaction for phosphoglycerate kinase, EC# 2.7.2.3, display this reaction in the Navigator by selecting **Reaction→Search by EC#**, and typing in 2.7.2.3.

5. Right-click on the EC number in the title, and select **Edit→Create/Add Enzyme**.

6. Type in the name of the product of yhfR (since YhfR is a synonym for the protein, you can type that instead of a longer name this will also serve to disambiguate between proteins if, for example, you changed the name of the gene product to "phosphoglycerate kinase", since there was already a protein by that name in the PGDB). Click **OK**.

7. The Protein Editor appears. If you have anything to add, such as information about activators or inhibitors, or a citation or comment, you can enter it here. In this example, you should add an evidence code for this enzymatic activity. The appropriate code is EV-COMP-HINF-FN-FROM-SEQ. When you are done, click **OK**. The reaction is redisplayed, showing the link to the new enzyme.

7.4.3 Entering a New Pathway

Imagine that you want to enter the pathway for phenylalanine biosynthesis, as shown in Figure 7.21, into a PGDB.

Entering the pathway for phenylalanine biosynthesis into a PGDB is accomplished using two steps. The first step is to build the pathway out of its component reactions. Once the pathway has been constructed, the enzymes that catalyze each reaction can be linked to each reaction using the **Edit→Create/Add Enzyme**command.

1. Create the new pathway frame by selecting **Pathway→New**. The Pathway Info Editor appears so that you can classify the pathway, and enter a common-name, a comment, and perhaps literature citations for the pathway. Remember to add an evidence code with an appropriate citation. When you exit from the Pathway

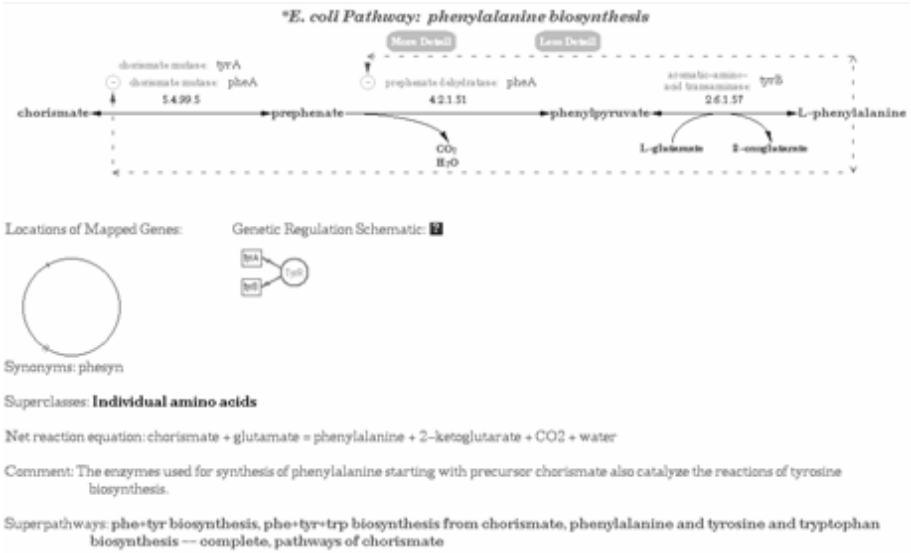


Figure 7.21: EcoCyc pathway for phenylalanine biosynthesis

Info Editor, the pathway display window for this frame is displayed, although no reactions have been added to the pathway yet, and the Pathway Editor appears.

2. From the Pathway Editor, select **Pathway→Enter a linear pathway segment** to invoke the Segment Editor. Since each reaction in this pathway has an EC number, you can simply enter the three EC numbers for these reactions in the "EC number" boxes to the left of the first three reaction arrows in the Segment Editor. Enter them in the same order that the reactions should appear in the pathway. For any reactions that did not have EC numbers, specify the reactants and products of the reaction. Click **Check** to check the pathway, and then click **Done** when the checking phase is complete.

3. You are now back in the Pathway Editor. The pathway is complete, so select **Exit→Keep changes** to exit from this editor. The pathway is redrawn by Pathway Tools.

4. Right-Click on a reaction arrow in the display and select **Edit→Create/Add Enzyme**. If you know the name or frame ID for the enzyme, and you know it already exists in the PGDB, you can enter it directly. If the enzyme is a protein complex that does not yet exist in the PGDB, you must first create it. We use a complex in the example that follows.

5. To create an enzyme that is a heterodimer, go to the Choose Protein dialog box, and select **Create New Protein**. In the window that pops up, the type protein complex should already be selected. Specify that the number of distinct gene products is 2, and type in two gene names, for example, tyrA and pheA (assuming those genes

exist in the PGDB — note that this selection is for demonstration purposes only, as it is unlikely that those two gene products form a complex in any real organism). The Choose Protein dialog box should reappear, with the text field filled in with a generated frame ID. Click **OK**.

6. The Protein Editor dialog window appears, and you can enter information about the protein complex and its catalysis of the reaction. Note that the name for the enzyme is derived from its enzyme activity name(s) (because, for example, a bi-functional enzyme has two different names for its two activities), so filling in the enzyme activity name field also updates the enzyme name. If you know the number of copies of each subunit within the complex, you can enter a coefficient in the section for each subunit. Click **OK** when you have finished.

7. The updated reaction page is displayed in the Navigator window, but you can use the Back command to return to the pathway display.

7.5 Advanced Editing Topics

This section provides more comprehensive information about editing PGDBs, such as

- How the Ocelot DBMS handles simultaneous updates to a PGDB by multiple users
- Restrictions on editing PGDBs designed to protect your work
- Full list of commands accessible through right-clicking on an object handle
- Editing commands in the Tools men.
- Encoding literature citations in a PGDB
- Rules for encoding special formatting of text, such as Greek letters
- Conventions used to name PGDB frames
- Creating URL links from PGDBs to other Web databases
- When Pathway Tools recommends a change

7.5.1 Saving DB Updates

Multiple users can update a PGDB simultaneously. The Ocelot DBMS uses an *optimistic concurrency control mechanism* to coordinate these updates, ensuring that they occur in an orderly fashion. The mechanism is optimistic in the sense that it assumes that conflicts

among updates will be infrequent. Therefore, an appropriate way to control these conflicts is to allow users to make updates at will, and to check for conflicts between updates at the time the updates are *saved*.

The Ocelot update model is based on the notion that at any given point in time, there is a *public version* of a PGDB, and that in addition, developers may have their own private versions of the PGDB in *private workspaces*. Whenever users open the PGDB stored in the Oracle or MySQL server, they are creating a new private workspace. They can make as many updates as they like to that private workspace. Those updates exist on their workstations only until they execute a **Save DB** operation (that operation is in both the Pathway Tools main menu and in the GKB Editor **DB** menu as **Save DB**). The **Save DB** operation proceeds in two phases. In the first phase, Ocelot checks whether any changes made by the user conflict with changes that may have been saved recently by other users. If no conflicts are found, Ocelot proceeds to the second phase, in which the changes are saved to the Oracle or MySQL server, and any recent changes made by other users are loaded into the user's workspace.

PGDB updates are not saved until the **Save DB** operation has completed successfully. The **Save DB** operation saves changes for the current PGDB only. If you have updated more than one PGDB, you must select each PGDB in turn and perform a **Save DB**. An asterisk ("*") next to the name of an organism in the organism-selector menu and in the Organism Summary screen indicates that unsaved changes exist for that PGDB.

Figure 7.22illustrates a scenario in which several different users are updating the DB at overlapping times.

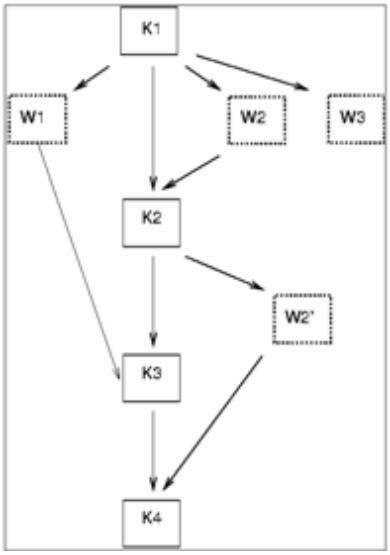


Figure 7.22: Sample set of relationships between public DBs and private workspaces

In the scenario represented in Figure 7.22, three different users open an Oracle or MySQL

PGDB at roughly the same time, and thus are all working from the same public version of the DB, called *K1*. Each user is working at a different workstation, and has private workspace, which we call *W1*, *W2*, and *W3*, respectively. Fred works in *W2* for 15 minutes or so, makes several modifications to enzymes in the glycolysis pathway, and then does a **Save DB**. Since no other user has saved any changes during Fred's session, no conflicts can occur. Therefore, Fred's changes are saved into the DB to produce a new version of the DB, *K2*.

Mary works in *W1* on the arginine biosynthesis pathway for about an hour, and then does a **Save DB**. Since she performs the save after Fred saved *W2*, Ocelot compares her changes with the changes made in *W2*. However, since Mary and Fred are working on completely different pathways, no conflicts are detected, and the changes made in *W1* are also saved successfully, resulting in a new version of the public DB called *K3*. In addition, the changes to the glycolysis enzymes made in *W2* are loaded into *W1*, and are now available to Mary.

Fred continues his work, and modifies the comments in several reactions of the glycolysis pathway. He now saves his workspace, called *W2'*. Since he saves after Mary saved *W1*, his changes are compared against Mary's changes for conflicts. Again, no conflict is found, so Fred's changes are saved successfully as a new version of the DB, called *K4*. The changes made by Mary are loaded into Fred's workspace as well. Fred might continue his work by making some changes to the glycolysis pathway itself.

However, Fred might decide that his changes are not suitable, and that he wants to undo them. To do so, he can execute the **File→ Revert Current DB** operation, which will undo all changes he has made since his last **Save DB** operation. In general, **Revert Current DB** restores the PGDB to its state as of the last **Save DB** operation, or the last **Revert Current DB,** or since a user started a session, whichever occurred most recently. **Revert Current DB** then loads in any changes that were saved by other users (in Fred's case, there were none).

Imagine that Fred continues his work for the next day in the same Pathway Tools session (i.e., without exiting out of the Pathway Tools and back to UNIX). Once a night, at 2:00 a.m., Ocelot performs a **Refresh All Open DBs** operation, *if the user's own workspace does not have any unsaved updates.* The **Refresh All Open DBs** operation loads in all recent changes that have been saved by other users to decrease the likelihood of conflicts occurring. Users can invoke the **Refresh All Open DBs** operation at any time through the **File** menu.

7.5.1.1 Checkpoints

If you want to save your DB updates, but the **Save DB** command is either undesirable (because you do not yet want your changes to be visible to other users) or unavailable (because the connection to Oracle or MySQL is temporarily unavailable), you can cap-

ture your updates in a local file by using the command **File→Checkpoint Current DB Updates to File**. Checkpointing is available only for PGDBs that are stored in Oracle or MySQL.

Pathway Tools automatically checkpoints the updates for all DBs with unsaved changes every five minutes into a file in your home directory (under Windows, in the folder `Documents and Settings/USERNAME`) called `.DBNAME-autosave.ckp`.

Both the preceding command and the auto-checkpoint capability provide ways of protecting DB updates against system crashes, and also allow you to quit from Pathway Tools, and later restart it and restore the checkpointed updates.

Checkpointed updates can be restored using the command **File→Restore Updates from Checkpoint File**. This command loads in and replays all changes stored in the checkpoint file, on top of any updates that have been made since the last checkpoint, save, or revert. To save those changes permanently to the PGDB, you must still then perform a **Save DB** operation.

7.5.1.2 Summary of File Menu Commands

The full list of database commands available through the **File** menu is as follows:

Summarize Databases: Provides a summary table for all the PGDBs in the system.

Summarize Current Organism: Summarizes the data within the currently selected PGDB.

List Unsaved Changes in Current DB: Describes the modifications you have made to the current PGDB since your last **Save DB** operation.

Revert Current DB: Takes the DB back to its state since you last saved it using **Save DB**, or since the start of your session, whichever was most recent. That is, all changes you have made since that time are discarded.

Refresh All Open DBs: Brings your version of all your currently open PGDBs up to date with respect to any transactions committed by other users. If there have been any such transactions, a window appears containing information about the transactions and the frames that have been modified. You should not invoke this command if you have modified the DB but not yet saved your changes, because some of your changes may be overwritten. The DB is automatically refreshed when you save it, and nightly (at approximately 2:00 a.m.) if you leave the Pathway Tools running overnight and you do not have any unsaved changes. You need only invoke this command yourself to refresh at other times.

Checkpoint Current DB Updates to File: Saves to a file all changes to the current DB made since the last **Save DB** or **Revert DB** operation. Those changes are not saved per-

manently in the PGDB. This operation is provided as a last-resort means of capturing changes, such as when there is a network outage or the database server is unavailable. The Checkpoint DB command is also a fast way of capturing changes if **Save DB** operations work very slowly in your environment. You could run Checkpoint DB operations frequently so you will not lose work if your computer crashes, and run slower **Save DB** operations less frequently. Checkpointing is available only for PGDBs that are stored in Oracle or MySQL.

Restore Updates from Checkpoint File: Retrieves into the current PGDB updates that were saved using the **Checkpoint Current DB Updates to File** operation. To save those updates permanently to the PGDB, you must next perform a **Save DB** operation.

Delete a DB: Deletes all traces of a PGDB from permanent storage, including the disk-based directory tree for the DB, and (for PGDBs stored in Oracle or MySQL) deletes the Oracle or MySQL data for the PGDB. Do not use this command unless you really want to remove the PGDB from Pathway Tools, such as if you intend to use PathoLogic to generate a newer version of the DB from scratch.

Save Current DB: Saves all changes for the current PGDB.

Attempt to Reconnect to Database Server: If you are editing a PGDB that resides in an Oracle or MySQL DB, this command opens a connection to the Oracle or MySQL server computer in which your DB resides. An existing connection to Oracle or MySQL can become lost for various reasons (such as loss of Internet connectivity), which will prevent you from being able to save your changes to the DB. If you successfully reopen the connection, you will be able to save. If the connection does not reopen successfully, try again later, or create a checkpoint. (Note: there is no harm in reopening a live connection.)

Print: Sends the current Navigator display window to the printer.

Import: See Section 4.1.

Export: See Section 4.1.

7.5.2 Editing Restrictions

SRI's support policy for Pathway Tools requires that you follow these restrictions on updating PGDBs:

- Do not alter PGDB schemas, such as by adding or removing classes or slots.
- Do not modify any SRI-supplied PGDBs, such as EcoCyc or MetaCyc, because any modifications you make will be lost when you upgrade Pathway Tools.

7.5.3 Right-Button Menu

The following additional commands are available via the mouse right button by clicking on an object handle. A number of these commands refer to a set of editing tools called the GKB Editor, which is not described in detail in the *Pathway Tools User's Guide.* The GKB Editor is an older editor system than the Pathway/Genome Editors, and although the GKB Editor seems less intuitive to some users, it has more power than the Pathway/Genome Editors. See URL `http://www.ai.sri.com/~gkb/user-man.html` for more information on the GKB Editor.

The right-button commands are arranged into three submenus as follows:

Edit menu:

Frame Editor: Invokes the Frame Editor (part of GKB Editor) on this frame. With this command and the other GKB Editor editing commands below, you can continue using the Pathway Tools while editing the frame. Exiting the Frame Editor causes the Pathway Tools to redisplay the edited frame to illustrate the effects of any changes.

Synonym Editor: Invokes the Synonym Editor, which provides quick access to the list of synonyms for an object. You can also alter the Common Name of the object from this editor.

Relationships Editor: Invokes the Relationships Editor (part of GKB Editor) on this frame, to view links between this frame and those to which it is related.

Ontology Editor: Invokes the hierarchical Ontology Editor (part of GKB Editor), beginning by browsing from the current frame This editor allows you to visualize the classification system (superclass-subclass hierarchy) within the Pathway Tools ontology.

Object-Type Editor: Depending on what type of PGDB object you are viewing within the Navigator, the name of one or more appropriate editors appears here (such as Gene Editor).

Export Object-Type to DB...: Allows you to export the current Object-Type instance to another Pathway / Genome Database.

Copy frame name to Clipboard: Saves the name of the selected frame into the clipboard so that you can later paste it somewhere else during editing with the GKB Editor.

Create frame: Upon right-clicking on the Create Frame command, you need to select the type of object to be created. This opens the appropriate editor (e.g., protein editor).

Delete frame: Deletes the selected frame from the DB.

Notes:

Add to history: Adds a note to the history of this frame. With this feature you can attach, to an object, multiple comments that are stamped with the current date and user.

Show:

Show frame: Displays a printed representation of the data in this frame in the terminal window in which you originally started Pathway Tools.

Show frame name: Displays the frame ID for the current frame both in the Lisp window and in the Navigator listener pane.

Show compound/reaction/pathway in overview: Displays the Overview graph, highlighting this compound, reaction, or pathway.

Show frame in other species: Displays this same biological object from another PGDB, for example, the same chemical compound or a gene with the same name (if one exists). You select the PGDB of interest in a cascading menu.

Show frame in all DBs: Displays this same biological object in all other currently open PGDBs by pushing onto the answer list all occurrences of this object in other PGDBs.

Print frame: Prints the data in this frame to a printer.

Print pathway or reaction frames: Prints the specified data to a printer.

Print pathway and its enzymes: Prints the specified data to a printer.

Show changes: Pops up a window listing the modification history of this frame. The modification history is stored as a sequential log of Generic Frame Protocol (GFP) operations that have been applied to the frame.

Refresh object display: Refreshes the display on the screen.

7.5.4 Object Names Visible to PGDB Users

Every biological entity in a PGDB is encoded as a frame (an object) in the DB. When you see an object display in the Pathway/Genome Navigator, you typically see the biological names that are stored in the slots called Common-Name and Synonyms that are defined within most frames. That is, whenever a Pathway Tools display window refers to a given object, the name that the Pathway Tools shows is computed by first attempting to retrieve the value of the Common-Name slot; if that slot is empty, then the frame name is displayed.

Conventions used by SRI in the EcoCyc and MetaCyc DBs are

1. Object names are generally all lowercase.
2. Bacterial gene names have the standard bacterial convention of "abcD".
3. Bacterial protein names are generally lowercase (such as "tryptophan synthetase"), except for protein names derived directly from the gene name, such as "TrpA".

Developers of other PGDBs can follow other conventions if they so choose — the most important thing is for all developers of a PGDB to follow a consistent set of conventions.

There are two exceptions to the previous rule that displayed object names are derived from the Common-Name slot: reactions and enzymes. Since reactions typically do not have any meaningful name, they are referred to by either the reaction equation, or the EC number. Enzymes are complicated because for a multifunctional enzyme we want to use different names to refer to the enzyme in different contexts. An enzyme that catalyzes two reactions has a different name associated with each reaction. When referring to the enzyme in the context of one of those reactions, or in the context of a pathway containing one of those reactions, we use the name associated with that reaction; that name is taken from the Common-Name slot of the enzymatic reaction frame that links the enzyme to that reaction. However, when referring to the enzyme outside the context of any particular reaction, we string together all the names associated with all the reactions. One final complication is that any names for the enzyme that are independent of the reaction(s) that it catalyzes are stored in the protein frame (e.g., names that refer to physical properties of the protein rather than the function of the protein).

7.5.5 Frame Name Conventions

Each frame has a frame name (also called an ID, or a key) that uniquely identifies that frame within the DB. Frame names must begin with a letter, and can contain numbers and hyphens (used to separate multiple words). The length of frame names is restricted to 40 characters. Instance names are written in uppercase, singular (as opposed to plural). Example instance frame names are TRPA-MONOMER, G00010, RXN0-152. Most class names are written as capitalized plurals. Example class names are Compounds, Genes, Polypeptides, Amino-Acids.

In the past, all frame names (both classes and instances) were assigned so that they conveyed some information about the biological objects that they describe. For example, the compound ketopyruvate has the frame name KETOPYRUVATE. However, we found that when applied to instances this practice leads to problems. For example, the protein frame name XYZA-MONOMER becomes obsolete if gene xyzA is renamed to qrtA. Thus the system has been changed, and currently instance objects are named systematically by the

software using a prefix that is specific for the particular type of object (compound, protein, etc) followed by a unique number. For example, maleylpyruvate has the frame name CPD-1070.

Even though the naming convention has been changed, we do keep the old names for existing objects, and we strongly recommend that frame names not be changed if they have been in use for some time, because this would result in different PGDBs referring to the same object using different frame names. Therefore, frame names should only be changed shortly after they are created. Keep in mind that you can easily alter the name that users will see by changing the value of the Common-Name slot.

By default, all new frame names are coerced to uppercase. If you are entering a class name and do not want it coerced to uppercase, surround the frame name by vertical bars when you first type it. Example: |Compounds|.

You can set a preference in the Ontology Browser (GKB Editor) under the **Preferences→Miscellaneous** menu so that this coercion to uppercase does not occur for class names.

Different conventions are used to name instances in different classes in a PGDB. The conventions include the systematic use of a prefix, followed by a set of characters that is specific to a particular PGDB (a PGDB identifier), followed by a dash and a number generated from some numeric sequence. For example, the frame name RXN0-152 is composed of the following elements: 'RXN' is the prefix for reactions, '0' is the unique PGDB identifier of the EcoCyc PGDB, and '152' is a unique numeric sequence. MetaCyc is the only PGDB that does not have a unique PGDB identifier, and thus the names of frames that were created in MetaCyc do not include that component.

The prefixes used for naming instances in different classes are as follows.

- Compounds : CPD
- Polypeptides: MONOMER
- Protein Complexes: CPLX
- Enzymatic Reactions: ENXRXN
- Reactions: RXN
- Pathways: PWY
- Genes: see Section 7.5.5.1

7.5.5.1 Naming Gene Frames

Different conventions for naming gene frames are used for different organisms. Typically, we try to use the same ID for the frame as the unique ID that has been assigned to a gene by a genome project, e.g. ECH74115_3407. However, when you create a new gene object, the software will follow the procedure outlined in Section 7.5.5, and will use the prefix G.

7.5.5.2 Naming Slots

All slot names are stored internally in uppercase. Typically, slots that take a single value have singular names, and slots that take multiple values have plural names. Examples: CATALYZES, COMPONENTS, LEFT-END-POSITION.

If you are using the Frame Editor to enter a value for a slot, and the value should be a frame name, but no frame exists with the name that you entered, one of two things will happen, depending on the slot. If the value must be a frame (based on the definition of the slot), a dialog asks you if you want to create the frame. If you create the frame, the name of the frame is the name that you originally entered, and the class under which the frame is created is determined by the slot definition. Alternatively, for some slots, such as Left, for which the value can be either a frame or a string, if the value you type does not match an existing frame it is left as a string (there may be other cases where a name is left as a symbol). Even if you later create the frame to which you were referring, the value of the slot remains a string until the Pathway Tools consistency checking code is run, at which point the correct frame name is substituted. You can tell whether a value of a slot is a frame, a string, or a symbol by looking at how it is displayed in the Frame Editor. Strings are surrounded by double quotes, symbols are displayed in a fixed-width font and are usually all uppercase, and frames are displayed in a variable-width font without quotes.

7.5.6 Special Formatting of Text

Pathway Tools provides formatting conventions that allow you to specify italic and bold type in text that is stored in the Comment slot (summaries). Most of these formatting conventions are modeled after HTML, and involve entering special markup within the comment text. The supported markup is shown in Table 7.2.

In addition, special characters such as Greek letters can be entered within comments. They are primarily used in the names of some compounds.

Example Markup	How it will Appear
Example of bold face.	Example of **bold face**.
Example of italic face.	Example of ***italic face***.
Example of <pre>typewriter face</pre>.	Example of `typewriter face`.
Example of <u>underlined text</u>.	Example of underlined text.
Example of superscript¹.	Example of superscript^{1}.
Example of subscript₁.	Example of subscript$_{1}$.
Heading Text	text is formatted as a heading
Heading Text	text is formatted as a smaller heading
Heading Text	text is formatted as a yet smaller heading
Center Text	text is centered
 	produces line break
<p>	produces paragraph break
γ	γ
Γ	Γ

Table 7.2: Formatting conventions supported by Pathway Tools.

7.5.7 Citations

Citations can be included in PGDBs in two ways. First, they can be entered in specially provided fields of the editors, in which case the citations are considered to refer to the object as a whole (such as to a pathway or protein).

Second, citations can be included within the text of a comment, as illustrated in the following example. The first line shows the syntax of the text that is to be entered into a DB within the comment field of an editor such as the Gene Editor; the second line shows what that text will look like when displayed by Pathway Tools (after new PubMed references have been imported from the PubMed database using the **File→Import→Citations from PubMed** command). Most editing forms have a button labeled CITS next to each comment field. Clicking on this button will insert the appropriate syntax at the current cursor position — you can then merely enter the PubMed or other IDs at the cursor position.

Pyruvate modulates the activity of the enzyme over a 10-fold range: |CITS: [84183611], [SMITH95]|.

Pyruvate modulates the activity of the enzyme over a 10-fold range:[Band84,Smith95].

When a user clicks on a citation displayed at the bottom of the page, the corresponding reference will be displayed by Pathway Tools or by your Web browser.

Each citation is identified by a unique identifier enclosed by square brackets. For references indexed in PubMed (the simpler case), use the PubMed ID number (e.g., 84183611) as the identifier to allow direct linking to PubMed. For references not in PubMed, you

should assign a unique ID of as many as 20 uppercase alphanumeric characters (ideally composed of the first author's last name followed by the year of publication, such as "SMITH95". If this ID is already in the database, use the format "SMITH95a"), and create a frame for each citation, as described in Section 7.3.14.

7.5.8 Frame References

As well as inserting literature citations within the text of a comment, it is possible to insert references to other frames within comment text. For example, imagine that when authoring a summary for a pathway that you want to refer to a substrate within that pathway. Although you could simply enter the name of the substrate, if you enter a frame (object) reference to the substrate, then the substrate name will be printed as the current common name of that substrate (in contrast to entering the text, which will remain static even if the substrate name changes in the future). In addition, the name will be printed in bold and will be a clickable hyperlink to the Pathway Tools page for that frame.

Most editing forms have a button labeled FRAME next to each comment field. Clicking on this button will insert the appropriate syntax for a frame reference at the current cursor position, and will prompt you to select a recently visited object from the history list as the frame reference. Should you want to reference an object that you have not visited recently, click outside the history menu to exit it, and then manually type in the PGDB ID of the object you want to reference.

An example frame reference, and its appearance within the comment, are as follows.

The first reaction in this pathway produces the compound |FRAME: GLT|.

The first reaction in this pathway produces the compound **L-glutamate**.

7.5.9 Creating Links Between a PGDB and External Databases

Pathway Tools allows you to create URL-based links from objects in a PGDB to objects in other databases outside the Pathway Tools environment that are accessible via Web queries. In addition, if you want to create links from internal databases at your organization to a PGDB, Pathway Tools can generate files called *linking tables* that specify the unique identifiers for PGDB objects, to allow you to link to them. For example, you could create links from an EcoCyc gene to homologs within a database at your site, and you could create links from those homologs to EcoCyc genes.

If your PGDB exists in conjunction with a previously existing database of genes for an organism, you may wish to have the Pathway Tools Web server send users directly to the gene pages generated by the other database, rather than to the Pathway Tools-generated gene pages. To accomplish this, you must first use one of the procedures described below

to create links from the PGDB to the other database. See the description of the **-gene-link-db** command line argument.

Links are visible in desktop views of a PGDB, and in Web pages.

The document `http://biocyc.org/linking.shtml` provides instructions on what URLs to use to link to PGDB objects.

7.5.9.1 Object Correspondence

To create links in either direction, you must first identify correspondences between PGDB objects and objects that you want to link to. To facilitate this process, Pathway Tools can generate a set of files listing the unique identifiers of PGDB objects, plus other information about those objects such as their names or EC numbers. For example, the EcoCyc genes file lists the EcoCyc ID, the common name, and the synonyms, for each EcoCyc gene. It also lists correspondences to other databases such as SwissProt. You could do a name-based search to identify correspondences between genes of interest to you, and EcoCyc genes. You could also use the SwissProt IDs to pull out sequences for each EcoCyc gene, and to compare those sequences to gene sequences of interest to you.

The linking files, and the columns provided in each file are as follows. The ORGID is a mnemonic for the name of the organism.

ORGID-gene-links.dat	ID EG# b# SP-ID CGSC-ID Name Synonyms...
ORGID-protein-links.dat	ID Gene-ID Name Synonyms...
ORGID-reaction-links.dat	ID EC#
ORGID-pathway-links.dat	ID Name Synonyms...
ORGID-compound-links.dat	ID InChi SMILES Name Synonyms...

where:

ID	=	PGDB ID
EG#	=	ID from ecogene database
b#	=	ID from Blattner GenBank entry
SP-ID	=	SwissProt ID
CGSC-ID	=	Coli Genetic Stock Center ID
EC#	=	Enzyme Commission number

7.5.9.2 Creating Links from PGDB objects

To create links from an object in a PGDB to an object in another database you must define the database itself to the PGDB, and you must define the links to the PGDB. These definitions can be created manually using the editing tools, or they can be loaded in bulk by using two different files: a database-definition file and a link-definition file. If the links were included in the original PathoLogic or GenBank format, they will be created automatically, but if the PGDB does not have a description of the external database then the links will not be live until one is created (see the next section).

For each database that you link to, you will define a unique identifier for the database, a name for the database, and a URL that can be used to query an object from that database given a unique identifier for that object.

7.5.9.3 Manual Creation of Database Links

Each of the editors for the different object types contains fields for entering links to other databases. Simply select the desired external database, and enter the ID for the object in that database. This is possible however only if information about the external database is already available in the PGDB. Many databases, such as UNIPROT, are already defined in any PGDB that you create (see MetaCyc External Databases for a listing of existing external database definitions in MetaCyc. If you want to define a new database, however, such as one that is local to your organization, you can do so using the command **File→Create→External Database Description**.

Enter an identifier, which you will use to refer to the database in link specifications (e.g., "GENEDB").

An editing form will pop up. The Direct Linking URL is the most important field to supply. Whenever a user clicks on a link, the Pathway Tools will send a Web browser to a URL that is computed by substituting the ID of a given object for the string "~A" in the value that you enter here. An example URL is: **`http://gene.pharma.com/dbquery?~A`**

If you want the database to be one of the possible selections in a particular editor, you must supply one or more Linked Object Classes. For example, if you want your database to be one of the options in the Gene Editor, you would select Genes.

If you want to edit an external database description (for example, if the linking URL has changed, or if links were added by PathoLogic, but no description has been supplied yet), simply right-click on any link to that database within an object display page, and select **Edit Remote Database Info**. See Figure 7.23 for an example of the Remote Database Editor dialog.

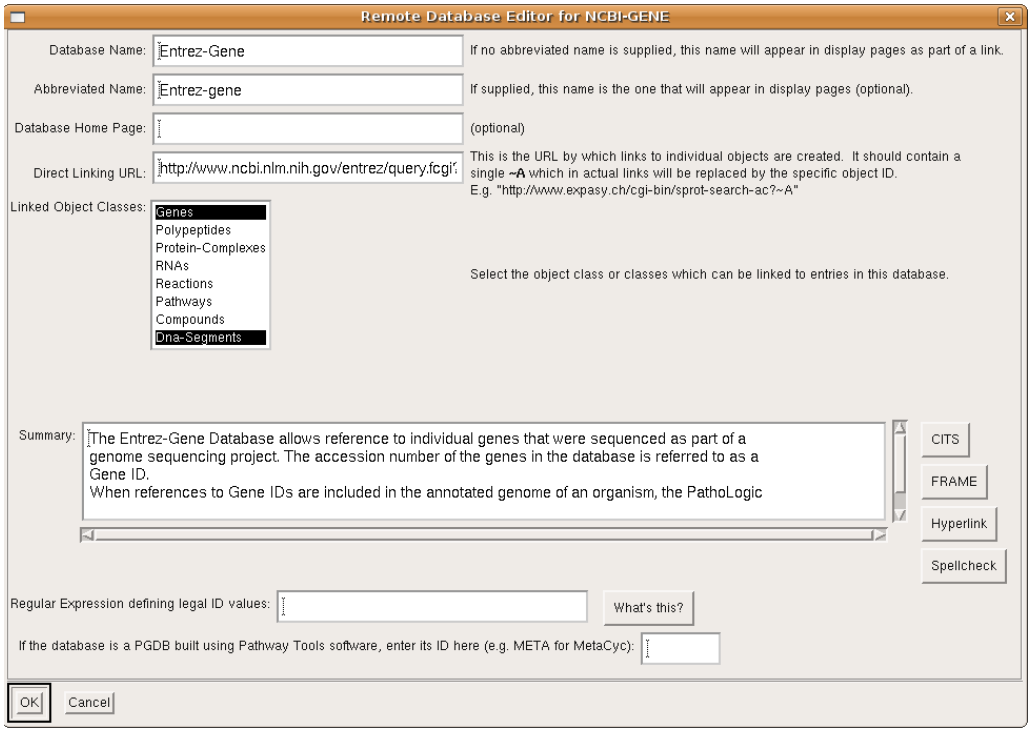


Figure 7.23: Editing an external database description.

7.5.9.4 Bulk Import of Links from a File

It is possible to import a large number of database links from a column-delimited data file (the link file) using the command **File→Import→DB Links from File**. This command will ask you to specify the link data file whose format is defined below. This approach allows

you to save the links within a PGDB. Note that this approach requires that a description of each external database referenced in the links must exist in your PGDB. Such descriptions can be created using the command **File→Create→External Database Description**.

Alternatively, you can load links into a PGDB at startup time using command-line arguments to Pathway Tools. If the links are being loaded into a PGDB that you cannot edit (such as the EcoCyc or MetaCyc PGDBs included in your software distribution), then you will have to load the links in this fashion every time you want to access them. This approach allows you to specify both a file containing links, and a file containing database descriptions. Use the command line arguments **`-dbdef`** *`database-file`* and **`-linkdef`** *`linktable-file`*.

A database-definition file defines each database that you will link PGDB objects to. An example database-definition file is as follows. Each line in the file describes one database; each column in the file is separated by a TAB.

```
# Sample database file.
DB1    ChemicalDB          http://chem.pharma.com/dbquery?~A
DB2    SequenceDB          http://seq.pharma.com/dbquery?~A
```

Imagine that DB2 is a DNA-sequence database. We want to link the EcoCyc gene whose unique identifier is EG10004 to the sequence for a homologous gene in DB2. Let us assume that the unique identifier for that homologous gene in DB2 is S9993. The preceding entry for DB2 specifies that to query object S9993, we would query `http://seq.pharma.com/dbquery?S9993`(the object identifier is substituted for the string "~A").

The link itself can be created using a link-definition file, whose format is as follows (each column in the file is separated by a TAB).

```
# Sample linktable file.
$ORGANISM ECOLI
TRP         DB1   C000111
EG10004     DB2   S9993    Homolog
```

The first non-comment line in the file must be a **$ORGANISM** specification, which indicates the PGDB in which these links are defined. You can determine the organism identifier by right-clicking on the organism's name in the Organism Summary page; the organism ID appears in the resulting menu. Subsequent lines in the file define links from one PGDB object to one object in another, Web-accessible database (one link per line).

The line beginning with **TRP** creates a unification link (a link between two pieces of the same biological object that reside in different databases). The link is from an EcoCyc object whose ID is **TRP** to an object in DB1 whose ID is **C000111**. The line beginning with **EG10004** creates a relationship link (a link between two different biological objects

based on some relationship between those objects) from **EG10004** to **S9993** in DB2; the relationship is specified as "Homolog".

Unification links and relationship links are displayed slightly differently by Pathway Tools (the word "Homolog" is printed in the second link).

7.5.10 Modified Proteins

Modified proteins are polypeptides that have one or more chemical groups attached. They can therefore be considered a kind of hybrid between a protein and a compound. We consider them instances of Polypeptides, but allow them to have values for many of the same slots as Compounds (e.g., chemical structure information). The unmodified protein is grouped together with all the modified versions under a single subclass of Polypeptides. For example, the class **Gln-B** has two instances: the unmodified protein **protein-pii** and the modified **uridylyl-pii**. The class **All-ACPs** contains the unmodified ACP-MONOMER along with instances for all the various ACP derivatives.

The unmodified protein should carry all the usual information and comments as a regular protein. It will be the only polypeptide in the class that has a link to the gene (because it is the direct gene product).

The modified protein(s) need not carry much information about the protein, but can carry additional, separate comments that apply specifically to the modified form. Each modified protein needs to link back to the unmodified form through the Unmodified-Form slot. The unmodified protein then points to all its modified versions through the Modified-Form slot. These two slots are maintained as inverse links, so the information needs to be added in one direction only.

Any protein or modified protein can be a substrate in a reaction, just like a small molecule.

7.5.11 Bulk Creation of New Frames

New frames can be created one by one using the menu-based editing tools. However, when creating a large number of frames, it can be more convenient to use scripts or enter all the data into a spreadsheet and then import it all at once. When the objects are all of the same type, the Import-Export facility is ideally suited for this. However, when creating brand new frames using this facility (as opposed to editing existing ones), it can be hard to know exactly what to enter into the spreadsheet. It is recommended that before you begin generating the spreadsheet for import, you use the Export facility to export similar data, and use it as a template. If there is no similar data in your PGDB (if, for example, you are entering data of a type that does not get created by PathoLogic), you will probably be able to find it in a curated database such as EcoCyc. Examine the exported data to determine

which slots typically contain values for your datatype. You will probably find that you will want to provide values for only a few slots— columns for other slots can be safely omitted from your spreadsheet.

When generating your own data, you must supply new frame IDs. These must of course be unique and not conflict with any frame IDs already in the PGDB. Try to follow the conventions on frame naming described in Section 7.5.5 wherever applicable. To ensure that you are not overwriting any existing frames, you can tell the import dialog to ignore or log any frames that already exist.

As an example, suppose that we want to add information about paralogs to a PGDB. Instances of the class Paralogous-Gene-Groups describe groups of paralogous genes, but this information is not predicted by PathoLogic, and cannot be added by using any of the specialized editing tools. Thus, the simplest way to get this information into a PGDB is via the Import facility. You can export the data from EcoCyc to use as a template, using the **File→Export→Selected Frames to File** command. Select the class Paralogous-Gene-Groups, and the slots GROUP-MEMBERS and COMMON-NAME (the other slots are not populated in EcoCyc). An example exported file, is shown in Figure 7.24 from inside a spreadsheet program. It is a simple matter to take the exported file, substitute the group names or numbers and gene IDs from a different PGDB, open that PGDB in the Navigator, and import the edited file, using the command **File→Import→Frames from File**.

7.5.12 When Pathway Tools Makes Recommendations

When Pathway Tools detects that you have made an allowable but not recommended edit, a window appears stating a recommended change and asking you whether you want Pathway Tools to make the recommended change for you. For example, if you specify the COMMON-NAME of an acid to be Artelinic acid, then Pathway Tools asks you to approve automatically changing the COMMON-NAME to artelinate.

If you need to disable this behavior during a Pathway Tools session, type the following to the Lisp Listener:

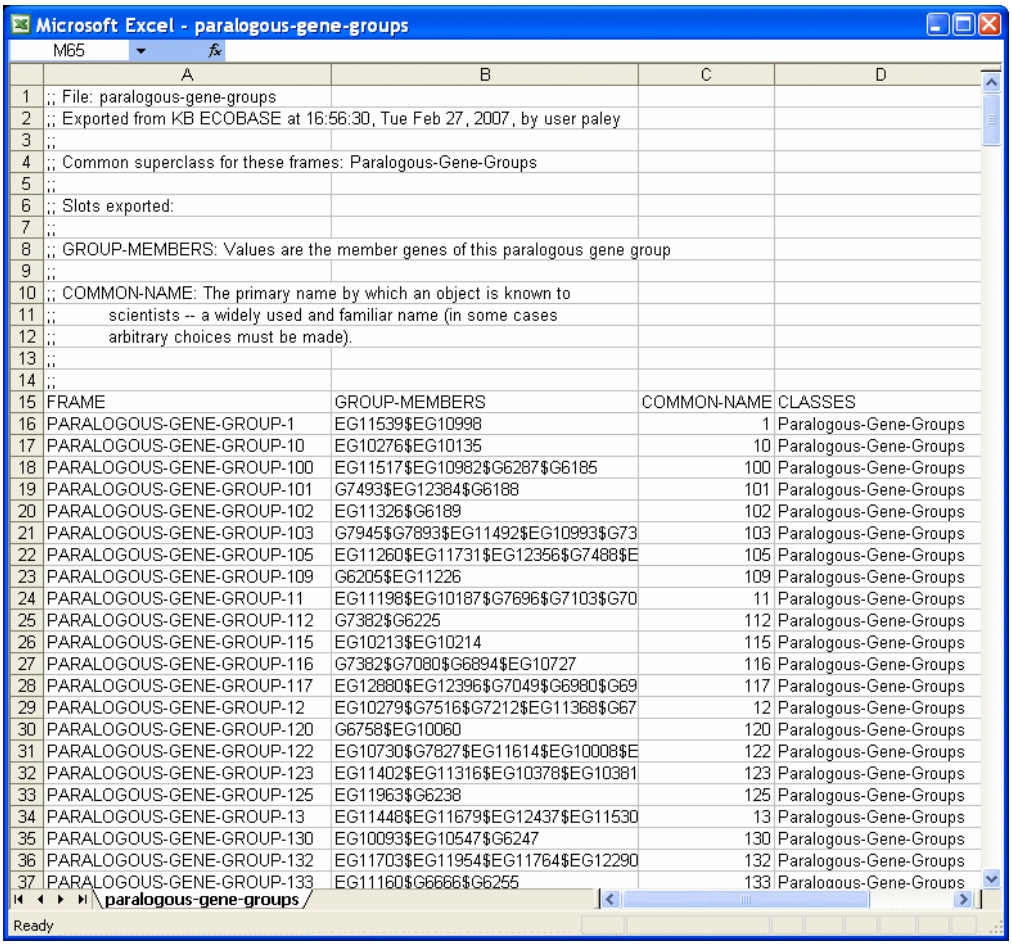
(setq *auto-recommend?* nil)

To later re-enable this behavior, type the following to the Lisp Listener:

(setq *auto-recommend?* t)

7.5.13 Curator Crediting

It is possible to attach curator crediting information to some curated objects, in particular to pathways and enzymes. That information specifies who curated information within



Microsoft Excel - paralogous-gene-groups

M65

	A	B	C	D
1	;; File: paralogous-gene-groups			
2	;; Exported from KB ECOBASE at 16:56:30, Tue Feb 27, 2007, by user paley			
3	;;			
4	;; Common superclass for these frames: Paralogous-Gene-Groups			
5	;;			
6	;; Slots exported:			
7	;;			
8	;; GROUP-MEMBERS: Values are the member genes of this paralogous gene group			
9	;;			
10	;; COMMON-NAME: The primary name by which an object is known to			
11	;; scientists -- a widely used and familiar name (in some cases			
12	;; arbitrary choices must be made).			
13	;;			
14	;;			
15	FRAME	GROUP-MEMBERS	COMMON-NAME	CLASSES
16	PARALOGOUS-GENE-GROUP-1	EG11539$EG10998	1	Paralogous-Gene-Groups
17	PARALOGOUS-GENE-GROUP-10	EG10276$EG10135	10	Paralogous-Gene-Groups
18	PARALOGOUS-GENE-GROUP-100	EG11517$EG10982$G6287$G6185	100	Paralogous-Gene-Groups
19	PARALOGOUS-GENE-GROUP-101	G7493$EG12384$G6188	101	Paralogous-Gene-Groups
20	PARALOGOUS-GENE-GROUP-102	EG11326$G6189	102	Paralogous-Gene-Groups
21	PARALOGOUS-GENE-GROUP-103	G7945$G7893$EG11492$EG10993$G73	103	Paralogous-Gene-Groups
22	PARALOGOUS-GENE-GROUP-105	EG11260$EG11731$EG12356$G7488$E	105	Paralogous-Gene-Groups
23	PARALOGOUS-GENE-GROUP-109	G6205$EG11226	109	Paralogous-Gene-Groups
24	PARALOGOUS-GENE-GROUP-11	EG11198$EG10187$G7696$G7103$G70	11	Paralogous-Gene-Groups
25	PARALOGOUS-GENE-GROUP-112	G7382$G6225	112	Paralogous-Gene-Groups
26	PARALOGOUS-GENE-GROUP-115	EG10213$EG10214	115	Paralogous-Gene-Groups
27	PARALOGOUS-GENE-GROUP-116	G7382$G7080$G6894$EG10727	116	Paralogous-Gene-Groups
28	PARALOGOUS-GENE-GROUP-117	EG12880$EG12396$G7049$G6980$G69	117	Paralogous-Gene-Groups
29	PARALOGOUS-GENE-GROUP-12	EG10279$G7516$G7212$EG11368$G67	12	Paralogous-Gene-Groups
30	PARALOGOUS-GENE-GROUP-120	G6758$EG10060	120	Paralogous-Gene-Groups
31	PARALOGOUS-GENE-GROUP-122	EG10730$G7827$EG11614$EG10008$E	122	Paralogous-Gene-Groups
32	PARALOGOUS-GENE-GROUP-123	EG11402$EG11316$EG10378$EG10381	123	Paralogous-Gene-Groups
33	PARALOGOUS-GENE-GROUP-125	EG11963$G6238	125	Paralogous-Gene-Groups
34	PARALOGOUS-GENE-GROUP-13	EG11448$EG11679$EG12437$EG11530	13	Paralogous-Gene-Groups
35	PARALOGOUS-GENE-GROUP-130	EG10093$EG10547$G6247	130	Paralogous-Gene-Groups
36	PARALOGOUS-GENE-GROUP-132	EG11703$EG11954$EG11764$EG12290	132	Paralogous-Gene-Groups
37	PARALOGOUS-GENE-GROUP-133	EG11160$G6666$G6255	133	Paralogous-Gene-Groups

paralogous-gene-groups

Ready

Figure 7.24: Sample export file for use as template when creating new frames

specific PGDB objects. When a pathway is transferred from one PGDB to another by the Pathway Import/Export facility, the curator credits are also transferred. For example, when a pathway is submitted from a PGDB created outside SRI to the MetaCyc DB, this facility identifies the curator of the pathway and its associated enzymes. The features of this curator crediting system include the following:

1. Different types of credit can be distinguished, such as authorship versus reviewing.
2. It is possible to credit at the level of either individual curators, or organizations. Some organizations may prefer to credit only by organization, that is, to not include an individual curator in the credits line.
3. Queries can retrieve all contributions made by a given curator or organization. This facility will help provide an incentive for contributions, because contributors will be able to clearly demonstrate their accomplishments.

Outside contributors who would like to submit pathways to the MetaCyc PGDB are responsible for creating their own organization and curator frames, using the editing tools described below. They should then attach credits using those frames to the pathways and enzymes, by means of the credits section in the Pathway Info Editor and Protein Editor.

7.5.13.1 Organization Editor

A graphical panel is provided for editing organization contact information. It is invoked on an existing organization frame from the right-click menu. An empty panel is brought up by the menu item called **File→Create→Organization**. You can also look up an organization from the menu item called **Tools→Search→Organizations**. This will show the home page of an existing organization, listing all the contributions if there are any.

The editing panel asks for the name, address, URL, and other contact information. It also asks for a short form of the organization's name, such as an acronym. Supplying the short form allows a more compact display of the credits.

7.5.13.2 Curator Editor

A graphical panel is provided for editing curator contact information. It is invoked on an existing curator frame from the right-click menu. An empty panel is brought up by the menu item called **File→Create→Curator**. You can also look up an curator from the menu item called **Tools→Search→Curators**. This will show the home page of an existing curator, listing the contributions if there are any.

The editing panel asks for name and email address. Especially when several curators are curating the same PGDB, it makes sense to also enter the default login account name, which if set correctly should select the appropriate curator as the default in the GUI panels that allow attaching credits You can enter the affiliation of a curator by choosing one or more organizations from the popup that appears after you click on the Select/Change button.

7.5.13.3 Attaching Credits to Pathways and Proteins

A section in the Pathway Info Editor (and in the Protein Editor) allows crediting contributors, which can be either organizations, individual curators, or a combination. To simplify this for the initial contributor(s), a default curator is preselected in the Curators GUI element, when a pathway is created. The default curator is selected by finding the curator frame that has a default login account value that matches the currently logged-in curator. If only one organization frame is referred to as this curator's affiliation, or if only

one organization frame has been defined in the PGDB, that organization frame is also selected as the default in the Organizations GUI element. Buttons in this panel quickly allow creation of additional curators or organizations. A time-stamp for the credit event is automatically assigned. There are three types of credit, one of which can be selected: Created, Reviewed, or Revised. A series of crediting events, consisting of the pathway creation, and followed by one or several review and/or revision events, can thus be recorded with this facility.

7.5.13.4 Visual Presentation of the Credits

The display page of an object that contains some values in its CREDITS slot will show a "Credits" line, displayed toward the bottom of a page, immediately above the list of literature references (if there are any).

To save space, a curator's name will be shown in the same formatting as in literature references, i.e., initials followed by the last name. The organization at which the work was performed will also be shown, but in abbreviated form, using its short name. Each curator and organization listed is hyper-linked to the corresponding home page. Several credit events are shown in chronological order. As an example, it could look like this:

```
Credits:  Created 9-Aug-05 by E.Hong, SGD
 Reviewed 8-Sep-05 by D.Kaiser, Stanford
```

7.6 Curation Tutorial

7.6.1 Introduction

This tutorial provides step-by-step instructions for the creation of a simple pathway. These steps are accompanied by screen shots, to make it easy to follow.

To keep the tutorial relatively short, not all options are covered. For more information regarding options not covered here, consult the appropriate sections of the *User's Guide*.

The easiest way to curate a new pathway is to follow this order:

1. Carefully plan the pathway, identify the different reactions that make it up, and identify the EC numbers (if applicable) of these reactions.
2. Find the individual reactions in the database, and create new reactions if necessary. You may need to create new compounds as well.
3. Compose the pathway from the individual reactions using the pathway editor.

4. Assign organisms to the new pathway (MetaCyc only) and add any commentary and citations that apply to the general pathway.
5. In MetaCyc: Create enzymes that catalyze the individual reactions. This step also includes the definition of the monomers that make up protein complexes (if applicable) and the genes encoding them. In an organism-specific PGDB: assign the appropriate enzymes, and create complexes if appropriate.
6. Curate full information for the new proteins, peptides, and genes.
7. QA: make sure that the pathway has an associated class, and that every enzyme (and the pathway) has an evidence code.

Special considerations for posting a pathway for incorporation into MetaCyc
If you would like to submit a pathway for inclusion in a future release of MetaCyc, consider the following:

- The pathway must be described in published journal articles.
- The pathway must be experimentally proven.
- Each pathway and enzyme must have a summary and literature citations.
- Indicate if you would like your name and/or affiliation to appear on the pathway and enzyme pages.

7.6.2 Before You Create a Pathway

There are a few steps you should consider before starting to curate the pathway.

7.6.2.1 Planning

This step is straightforward. While most everything can be changed at any time during the curation process, it is much easier to have everything ready when you need it. Thus, time should be spent in the planning of the pathway. Once the individual reactions have been identified, you should spend some time trying to find whether there is an appropriate EC reaction for these reactions. Pay close attention to details: for example, reactions that are very similar, and differ only in the inclusion of NADH, NADPH or NAD(P)H, may be defined as different EC reactions. The ultimate authority for EC numbers is the IUBMB Web site.

7.6.2.2 Creating Curator Frames

Before you create any new objects in the PGDB, create Organization and Curator frames for yourself. This way, every item that you create will be associated with these frames, making it easy to track your work, and providing you with the credit that you deserve. To do so, choose **File→Create→Organization** (Figure 7.25).

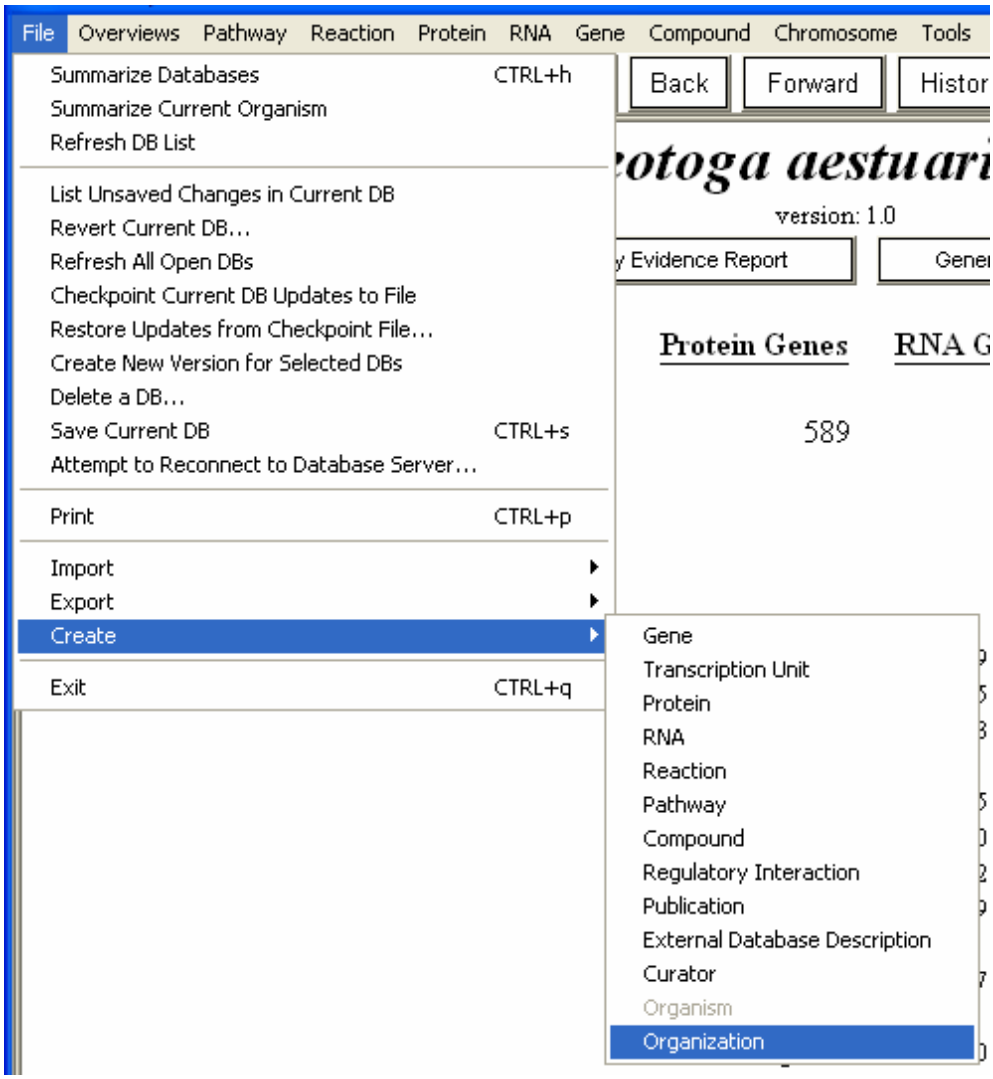


Figure 7.25: Creating a new organization frame

The Organization Editor window appears (Figure 7.26), and you can fill in the information.

Repeat the process (selecting **Create Curator**) to generate a Curator frame for yourself (Figure 7.27). Make sure you link yourself with the appropriate organization by clicking

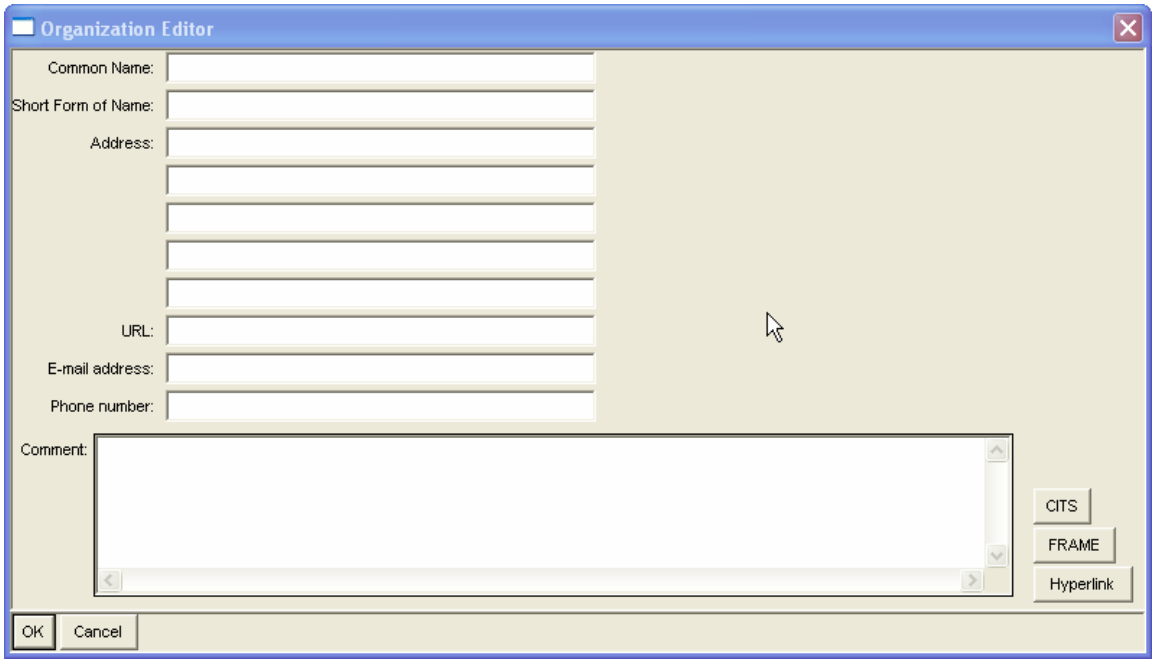


Figure 7.26: The Organization Editor

the **Affiliation** button.

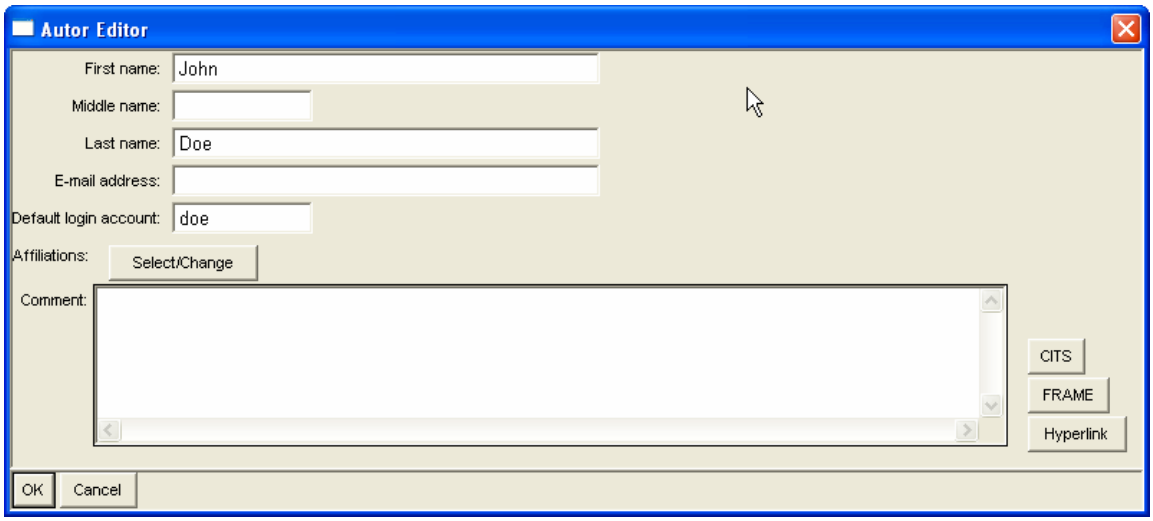


Figure 7.27: The Curator Editor

To ensure that your work is always correlated with the correct curator frame you need to make sure that this login name is always used by the system. Do this by selecting **Preferences** and then **General** from the **Tools** menu, which brings up the **General Preferences** dialog box (Figure 7.28).

Enter the same value that you used for the curator frame default login account.

This is particularly important in the Microsoft Windows environment. If this parameter is not set, the system uses the Windows login name, which may be different from the login specified for the curator frame.

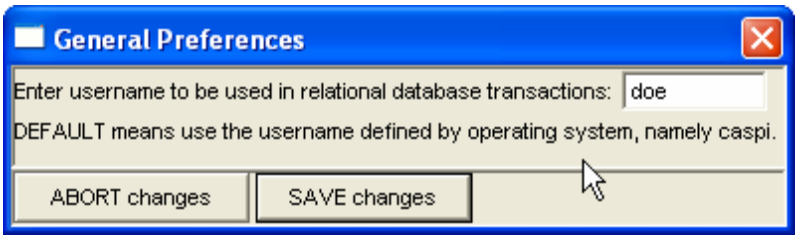


Figure 7.28: Specifying a user name in the General Preferences

7.6.2.3 Creating Compounds and Reactions

It is preferable to reusing existing reactions and compounds by importing them from MetaCyc into another PGDB, rather than creating new redundant PGDB objects. This approach minimizes data redundancy and data entry, minimizes typographical errors that could occur during redundant data entry, and facilitates comparative analyses by ensuring that the same object is always assigned the same unique ID in different PGDBs. Therefore, every effort should be made to find existing objects before creating new ones.

Most of the reactions that have EC numbers are already present in MetaCyc, and can be found by searching for the EC number (**Reaction Menu→Search by EC#**). If the reaction is present in MetaCyc but not in your PGDB, it will be automatically imported into your PGDB when you enter its EC number or reactants/products into the pathway segment editor. If you are sure that the reaction you need is not present in either database, create a new reaction. Reactions that you create should be mass balanced, if possible.

When a reaction is imported from MetaCyc, so are its reactants and products. If you have to create new compounds, try to enter the chemical structure for them and provide links to other compound databases, such as ChEBI and PubChem.

It is preferable to reusing existing reactions and compounds by importing them from MetaCyc into another PGDB, rather than creating new redundant PGDB objects. This approach minimizes data redundancy and data entry, minimizes typographical errors that could occur during redundant data entry, and facilitates comparative analyses by ensuring that the same object is always assigned the same unique ID in different PGDBs. Therefore, every effort should be made to find existing objects before creating new ones.

Most of the reactions that have EC numbers are already present in MetaCyc, and can be found by searching for the EC number (**Reaction Menu→Search by EC#**). If the reaction

is present in MetaCyc but not in your PGDB, it will be automatically imported into your PGDB when you enter its EC number or reactants/products into the pathway segment editor. If you are sure that the reaction you need is not present in either database, create a new reaction. Reactions that you create should be mass balanced, if possible.

When a reaction is imported from MetaCyc, so are its reactants and products. If you have to create new compounds, try to enter the chemical structure for them and provide links to other compound databases, such as ChEBI and PubChem.

7.6.3 Composing a Pathway

Once all the reactions have been entered, you can create the new pathway.

Start by choosing **Pathway→New**. This will open the Pathway Info Editor, as seen in Figure 7.29. Type in a name for the new pathway, and specify a class.

7.6.3.1 Specifying a Class

At the top of the window you will see two buttons — one labeled **Class**, and the other labeled **Create variant class for this pathway**. The **Class** button lets you assign the appropriate preexisting class to the new pathway. Every new pathway belongs by default to the general class "Pathways".

Clicking on the button opens the pathway class ontology browser. Click on **Pathways** to de-select it, and choose the appropriate category for the pathway, as seen in Figure 7.30. You can click on any "+" sign to expand that category to see more specific classes. You can specify any number of appropriate classes. When you are done, click **OK**.

Sometimes the new pathway may be a variant of other pathway(s) that are already in the database. The presence of multiple variants warrants the conversion of this pathway class into a "variant class". When multiple pathways reside in a variant class, each pathway page provides direct links to all other variants.

If the class that was selected is already a variant class, a message will be displayed to that effect, as seen in Figure 7.31.

If the class is not a variant class, it is simple to convert it, by clicking the **Create variant class for this pathway** button. A window appears that provides two options (see Figure 7.32).

One option is to convert the current class to a variant class, by clicking the button **Mark this class as a variant class**. The other option is to create a new, more specific variant class under the currently selected class. To do that you would need to enter a frame ID of the new class (in the form Lactose-Degradation) in the field "Variant class ID", provide

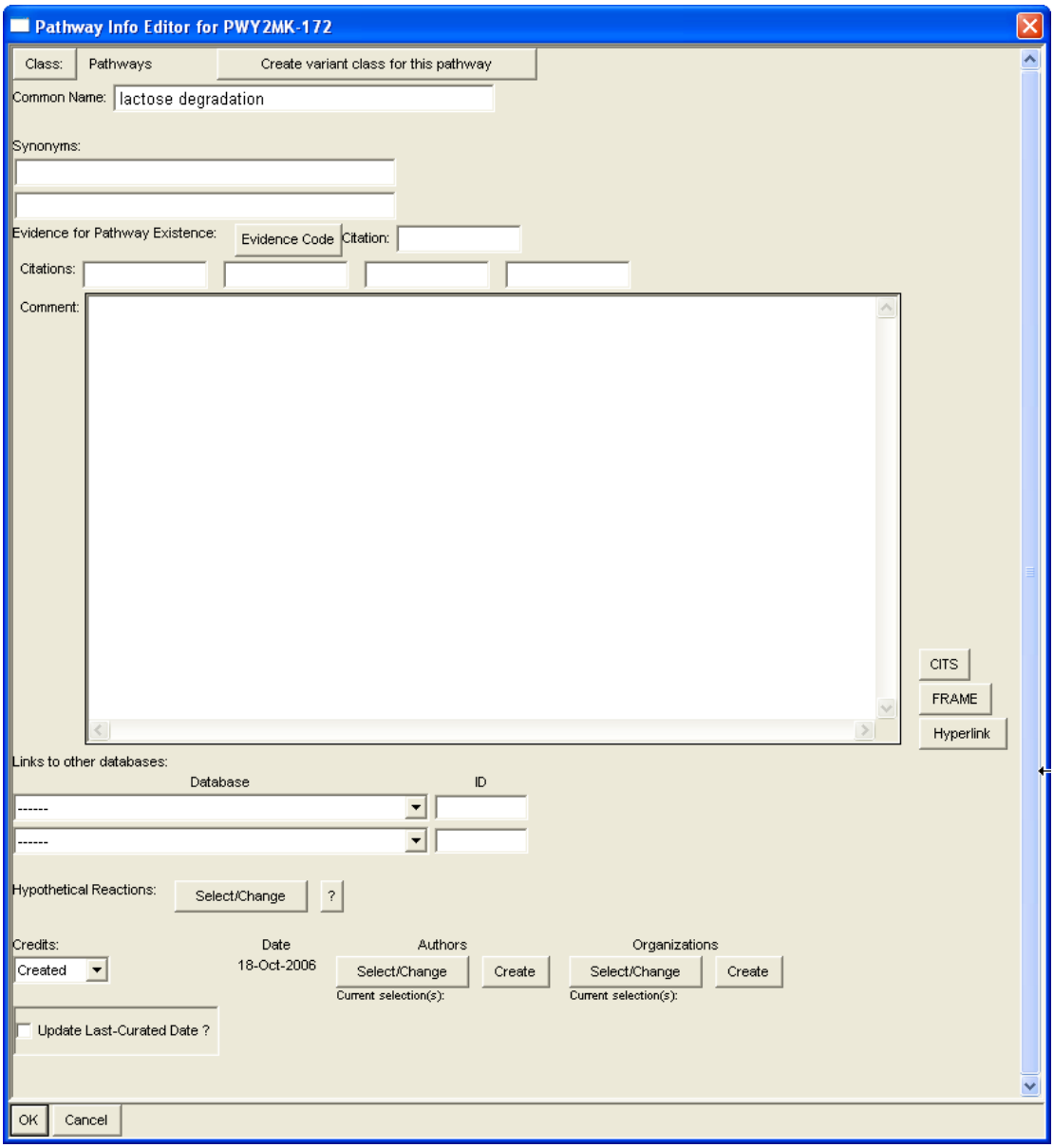


Figure 7.29: The Pathway Info Editor

a common name (in the form Lactate Degradation), and write a short description of the new class in the Documentation string that explains the type of pathways that should be contained in this class. Once you created a new variant class, you can specify the other pathways that should be moved into it.

When you are done with the class assignment, click **OK** to exit the Pathway Info Editor. As soon as you click **OK**, the Pathway Editor window will open. This editor lets you specify the reactions that participate in the pathway and connect them to each other.

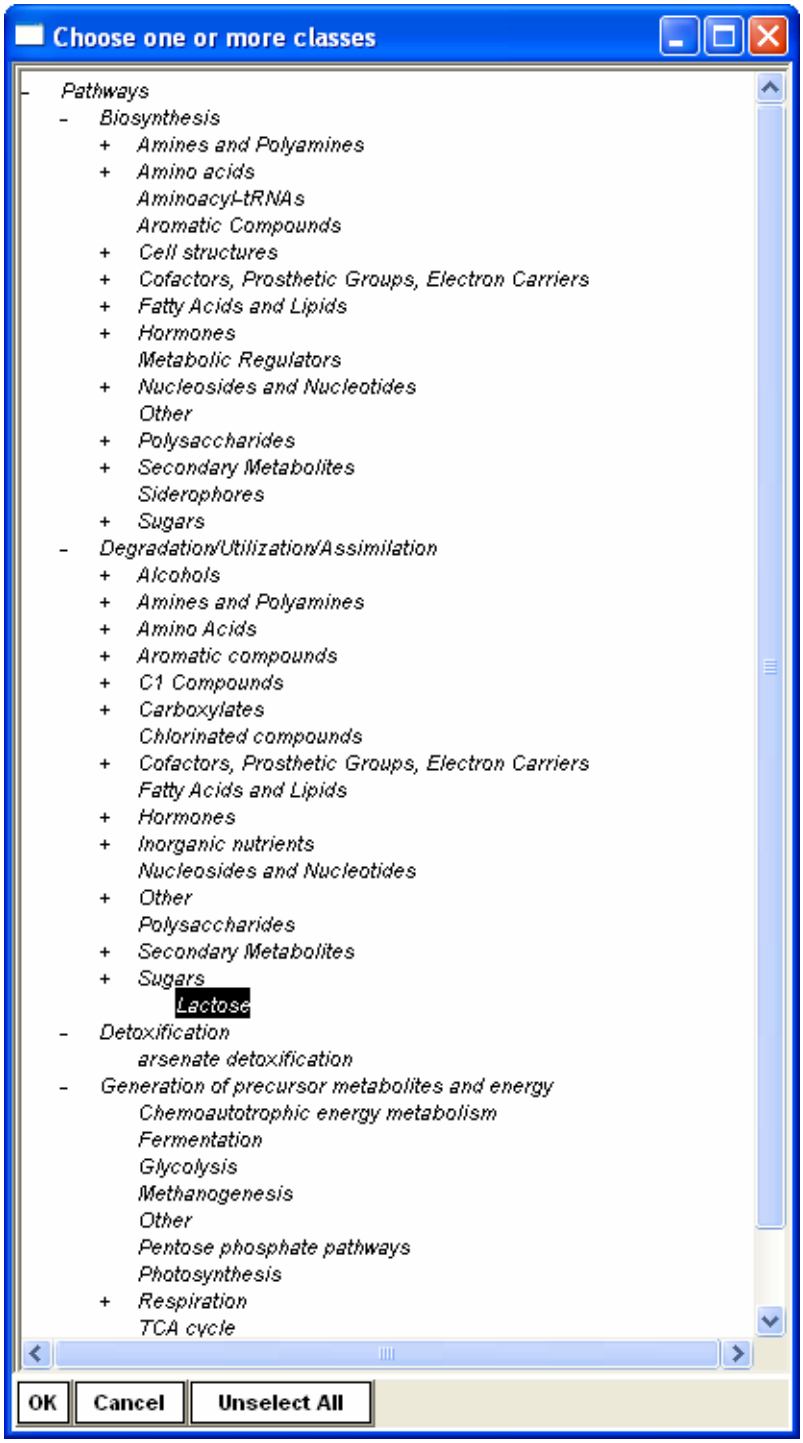


Figure 7.30: Choosing a pathway class

There are several ways to add the reactions — for example, you can search for them one by one using the **Reaction→Find and Add Reaction** option, which brings up the dialog

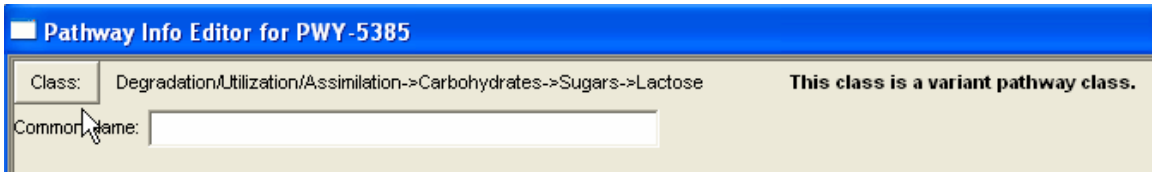


Figure 7.31: A variant class indication

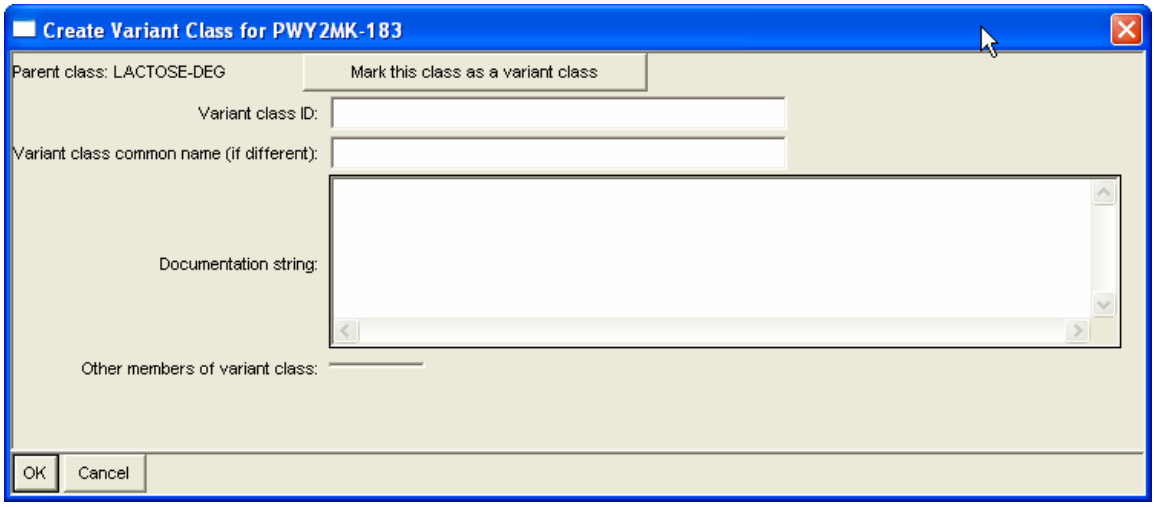


Figure 7.32: Creating a variant class

box shown in Figure 7.33. For more options, look at Sections 7.3.6.3 and 7.3.6.4.

Tip: if the reactions have been recently opened, the most convenient way is to pull them out of the "History" list, by using the item "Add reaction(s) from History" from the Reaction menu (Figure 7.34).

Tip: Once you have added all the reactions, it is a good idea to save the pathway (by exiting and choosing "keep changes", and then saving the database by clicking the "Save DB" button). This way, if anything goes wrong while connecting the reactions, you can always close the editor without saving your changes and reopen it, without having the need to import all the reactions again.

To connect two reactions you need to click (**a single click!**) on the predecessor (first) reaction. Note that the reaction will change color to red, while all the reactions that can be connected to it will change color to green (Figure 7.35). Now click on the reaction to be connected.

As soon as you click on the second reaction, the editor will merge the two reactions into a pathway and move them to the right pane (Figure 7.36). Continue connecting the rest of the reactions in the same way until the pathway is complete. You can click on a predecessor reaction in the right pane, and a successor in the left pane, as shown in Figure 7.37.

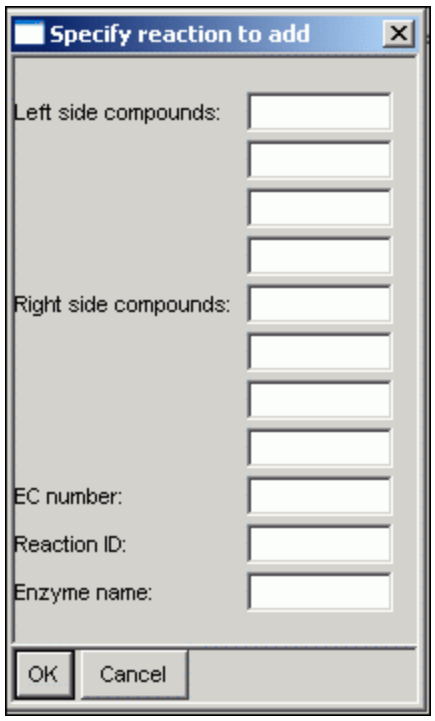


Figure 7.33: Find and Add Reaction

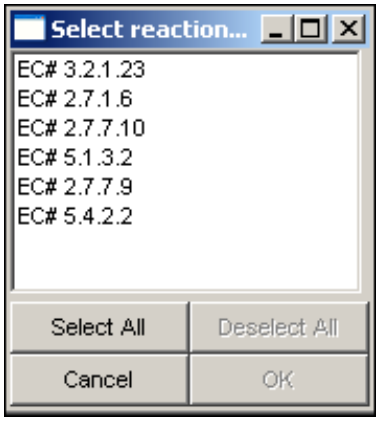


Figure 7.34: Reactions from History

If it is a circular pathway, click the last reaction, and then click on the first one to close the circle. You can right-click on any compound in the circle and choose **Place This Compound on Top** to rotate the circle to the desired position.

Some of the reactions (especially the first and last) may not use the correct main and side compounds. To fix this, right-click on the reaction, select **Choose Main Compounds for Reaction**, and specify the correct main compounds.

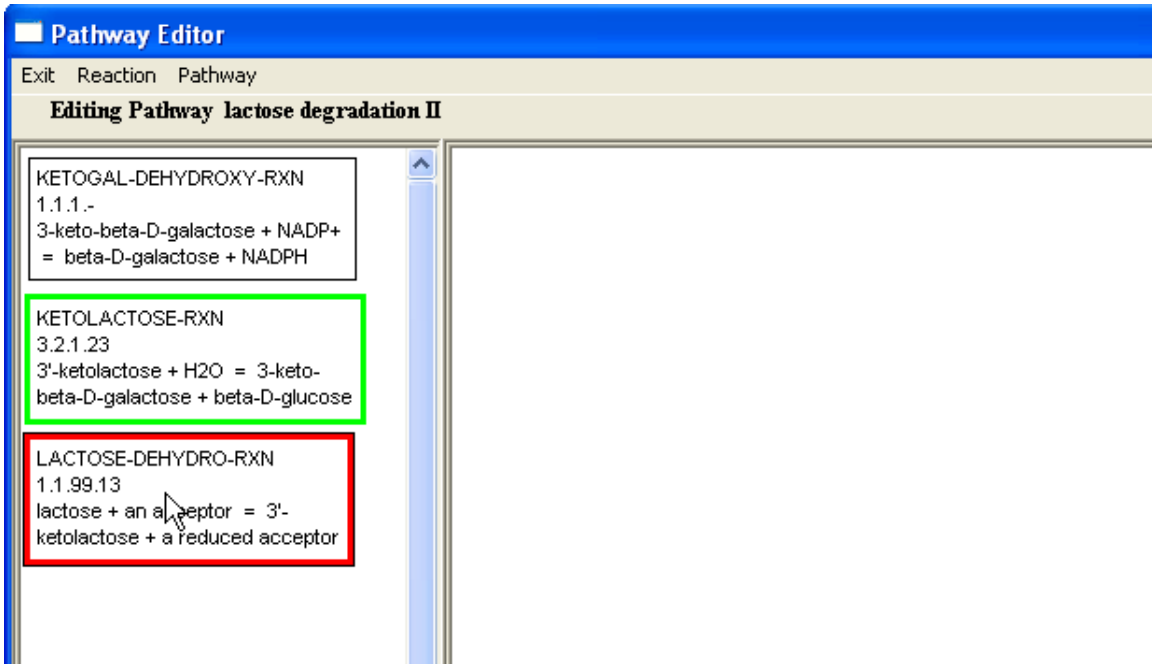


Figure 7.35: The Pathway Editor: specifying the predecessor reaction

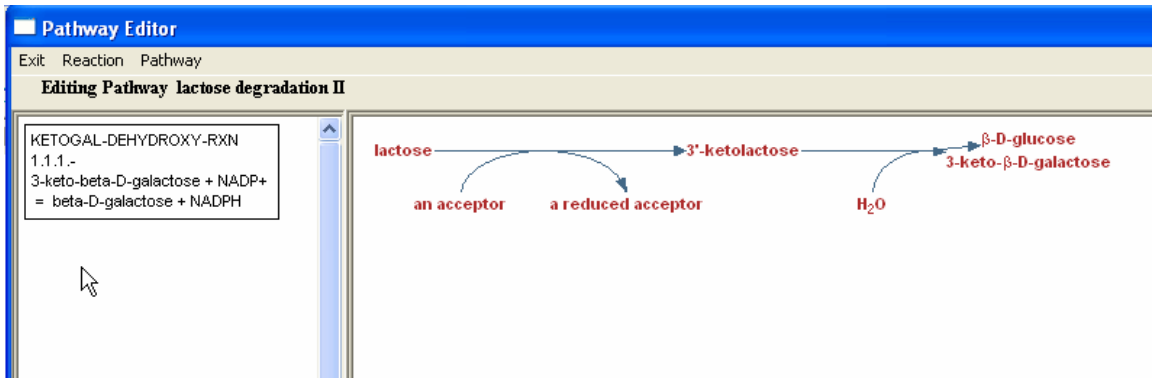


Figure 7.36: The Pathway Editor: connecting reactions

If you would like to link the pathway to other pathways, you can generate the links by right-clicking on the compound to be linked, and selecting **Add Link to/from Pathway**. This will bring up the pathway ontology browser, and would let you select the appropriate pathway. Other options include adding links to reactions, and even text strings.

When done, exit and save your changes.

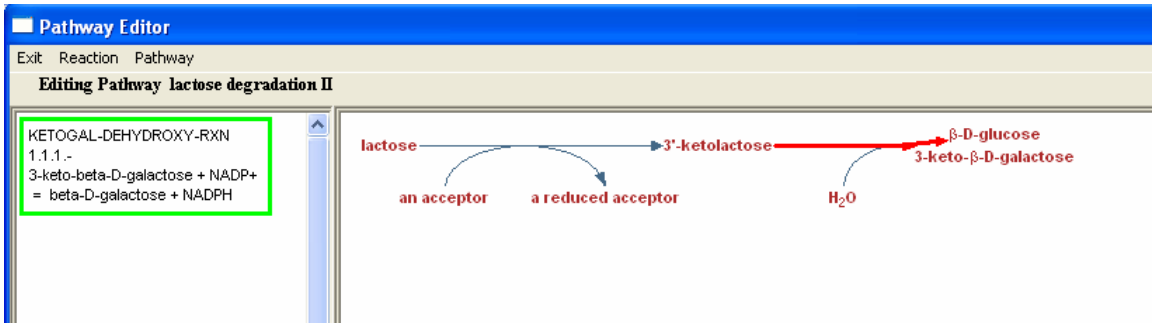


Figure 7.37: The Pathway Editor: adding the next reaction

7.6.3.2 Adding Pathway Evidence Codes and Citations

Once the pathway has been created, it is time to add evidence and citation information. Open the Pathway Info Editor again, this time by right-clicking on the pathway name, and choosing **Edit→Pathway Info Editor**.

7.6.3.2.1 Adding evidence codes Choose an appropriate publication that provides evidence for the existence of this pathway, and add the appropriate evidence code, including the reference to this publication. Clicking on the Evidence Code button brings up the evidence code browser, as seen in Figure 7.38.

Since all MetaCyc pathways must be experimentally proven, if you curate a pathway in MetaCyc at least one of the evidence codes should be of the EV-EXP-XXX type.

7.6.3.2.2 Adding citations The most convenient way to enter a citation in Pathway Tools is by using the publication's PubMed ID number. To find these numbers, search for the publication at **http://www.ncbi.nlm.nih.gov/pubmed**. Just type this number in a citation box. To embed citations within comments, use the "CITS" button. This button adds the string |CITS[ ]| to the text. You can type a Pubmed ID number between the square brackets, and the Navigator would show that citation in the form "Smith96".

After you have entered citations using PubMed ID numbers, you need to retrieve the actual citation information from the NCBI site. Do so by choosing **File→Import→Citations from PubMed** (Figure 7.39). The citations will not show up, though, until you save the database.

If you want to enter a citation for which there is no PubMed ID number, you will need to create a publication frame. To do so, start by typing a name for the new publication in the form Smith89 in any citation box (as in Figure 7.40).

As soon as you click outside the citation box a new window will appear as shown in

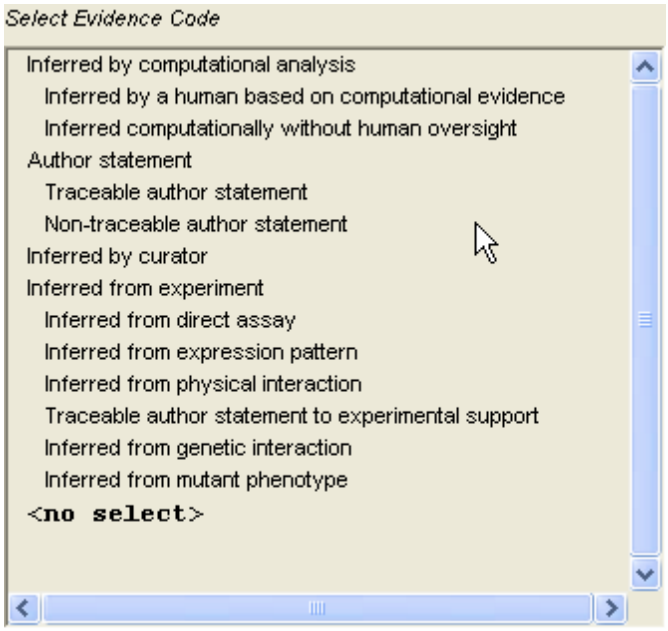


Figure 7.38: Specifying an Evidence code

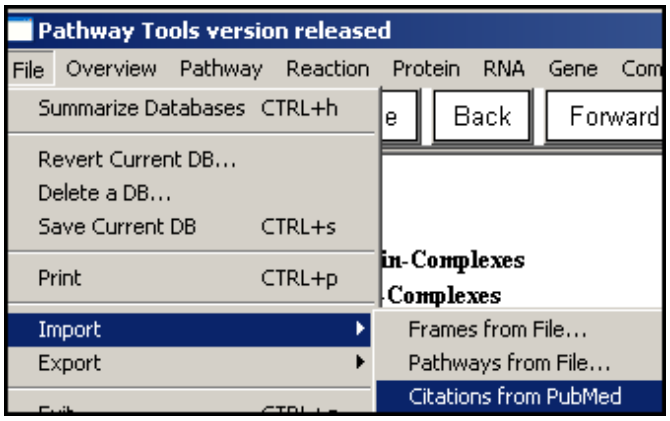


Figure 7.39: Importing PubMed Citations

Figure 7.41. Note: if no window appears, the database already has a reference with this name — in this case, you should revise the name by using the syntax Smith89a and so on.

To create the new reference click on the button **Search** or **Create Publication Frame**. The publication window appears, and you can manually enter the relevant information, as shown in Figure 7.42. Another way to get to the Publication editor is to select **File→Create→Publication**.

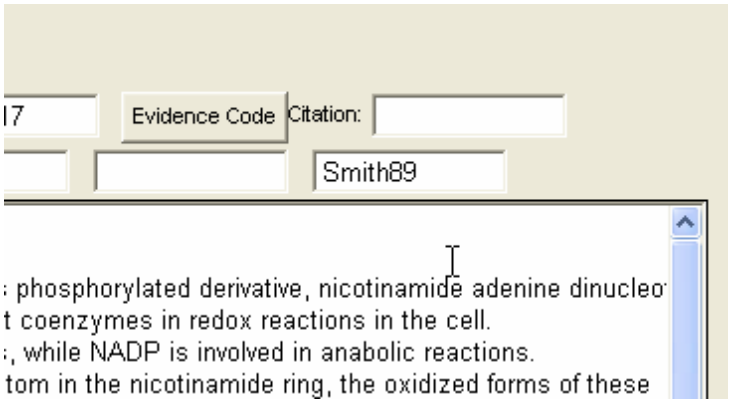


Figure 7.40: Entering a non-PubMed citation

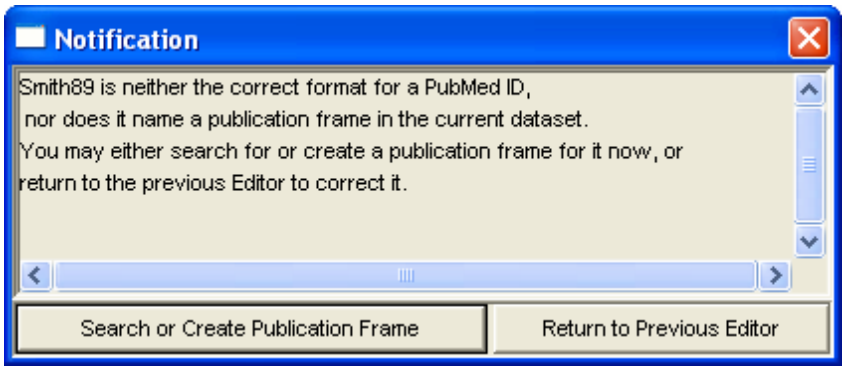


Figure 7.41: Notification about a non-existent publication frame

7.6.3.3 Adding a Summary

Add any comments that are appropriate for the pathway. Do not add comments about a specific enzyme or reaction here — those should be added directly to the object they refer to. You can refer to other objects in the database within comments by generating internal hyperlinks to these objects. To assist with this task, use the **Frame** button. This button will pop up a menu of history items that reside in the same database, from which one item can be chosen. The frame ID of the selected object will be inserted into the text, embedded inside the |FRAME: | template. For example, the compound carbon dioxide would first have to be visited, so it is entered into the history list. After selection by the FRAME button, it would insert the following construct into the text: |FRAME:CARBON-DIOXIDE|. The Navigator will show the common name of that frame (e.g., CO_2).

For example, the text shown in Figure 7.43 would appear in the Navigator as in Figure 7.44.

If the list of history items becomes so large that it is difficult to find the appropriate entry

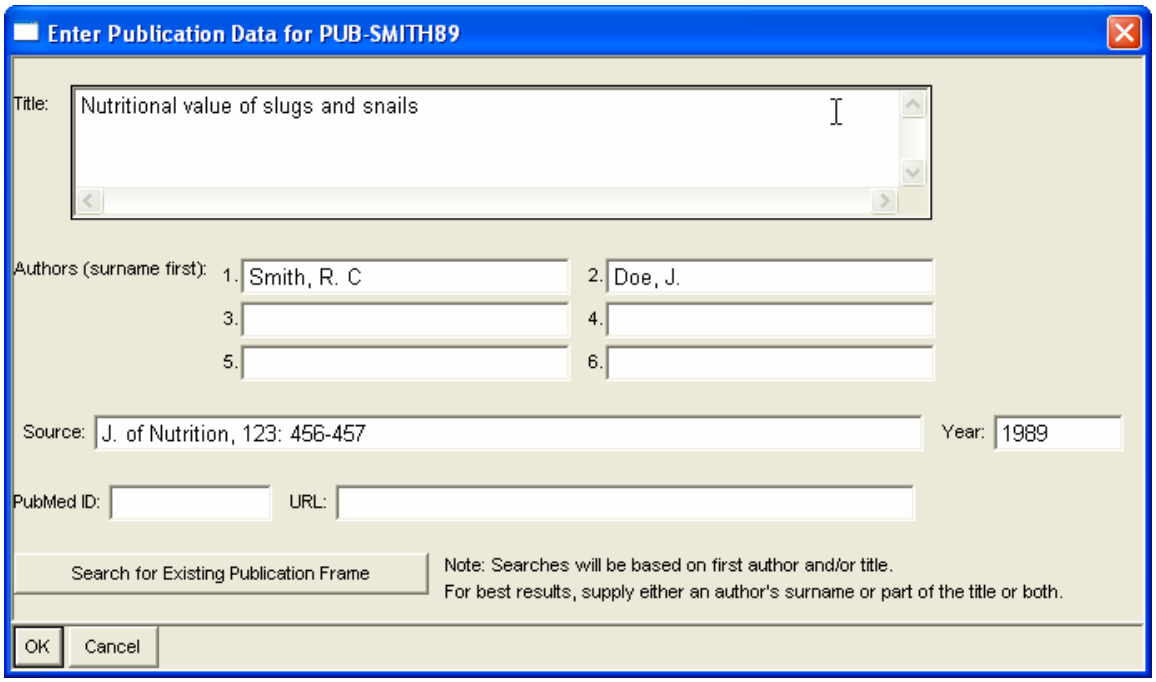


Figure 7.42: The Publication Editor

The pathways for the conversion of all of these compounds to |FRAME: CH4| follows a similar path. The first step is always the transfer of a methyl group from the substrate to |FRAME: CoM|, forming |FRAME:Me-CoM|, which is then disproportionated into |FRAME: CH4| and |FRAME: CARBON-DIOXIDE|: One in four |FRAME:Me-CoM| molecules is oxidized to |FRAME: CARBON-DIOXIDE| (see |FRAME: PWY-5209|), providing the six electrons that are required for the reduction of three |FRAME:Me-CoM| molecules to |FRAME: CH4| (see |FRAME: METHFORM-PWY|) |CITS: [Keltjens93][15882413]|.

Figure 7.43: FRAME constructs within a comment: editor view

The pathways for the conversion of all of these compounds to **methane** follows a similar path. The first step is always the transfer of a methyl group from the substrate to **coenzyme M**, forming **methyl-CoM**, which is then disproportionated into **methane** and CO_2: One in four **methyl-CoM** molecules is oxidized to CO_2 (see **methyl-coenzyme M oxidation to CO_2**), providing the six electrons that are required for the reduction of three **methyl-CoM** molecules to **methane** (see **methyl-coenzyme M reduction to methane**) **[Keltjens93, Pritchett05]** .

Figure 7.44: FRAME constructs within a comment: Navigator view

from it, you can reset the history list by opening the Tools menu and selecting “History”, and then “Clear”.

External hyperlinks to Web pages can be entered in html format, and the Hyperlink button can aid in this task by inserting the text string <A href=”http://”></A>.

For additional formatting rules (such as italics, bold, Greek characters, superscript and subscript), look at Section 3.3.2 in the Curator's Guide. When you are done, exit the Pathway Info Editor.

7.6.3.4 Creating enzymes

To add an enzyme to a reaction, right-click on the arrow symbolizing the reaction, and choose **Edit→Create/Add Enzyme**. This will bring up the dialog box shown in Figure 7.45.

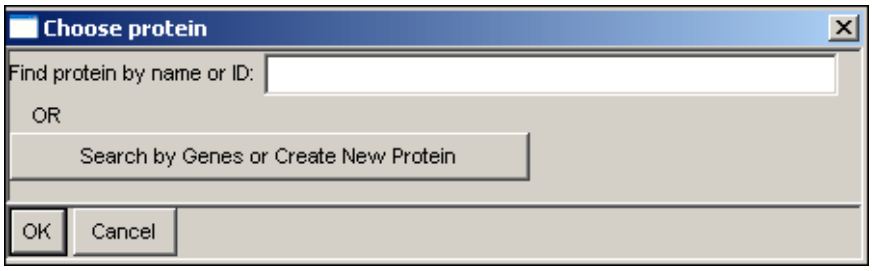


Figure 7.45: Adding an enzyme to a reaction

If the enzyme already exists in the database, type in the frame ID number for this protein. If it is a new enzyme, click on "search by Genes or Create New Protein". The "Specify Protein Subunit Structure" dialog box will appear, as in Figure 7.46.

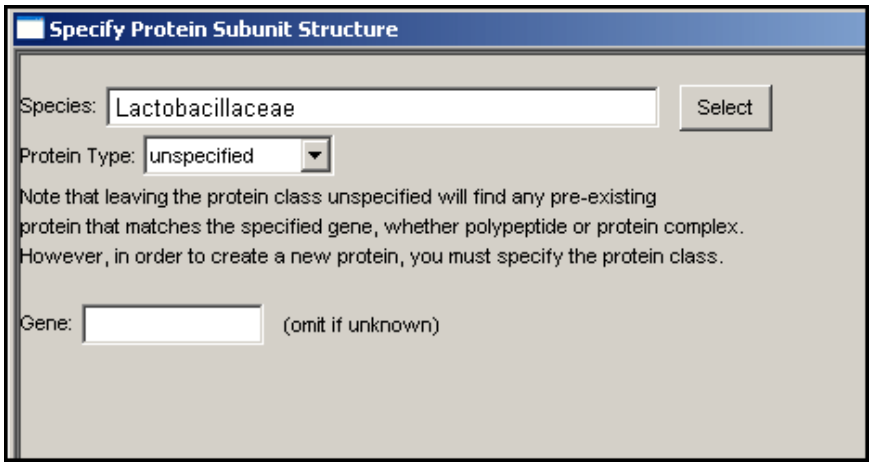


Figure 7.46: Specifying a protein unit substructure

Note that is taken from MetaCyc, where there is a need to specify an organism. If you are curating an organism-specific PGDB, naturally you would not need to select a species. Now specify the protein type. If it is a monomer, choose "Polypeptide", and if known,

specify the gene encoding that polypeptide. If it is a multimer, choose "Protein complex". The window will change accordingly, as shown in Figure 7.47. Note that in order to create a new protein, you must specify whether it is a monomer or a complex. If this information is not known, choose polypeptide, and explain it in the enzyme summary (Figure 7.56).

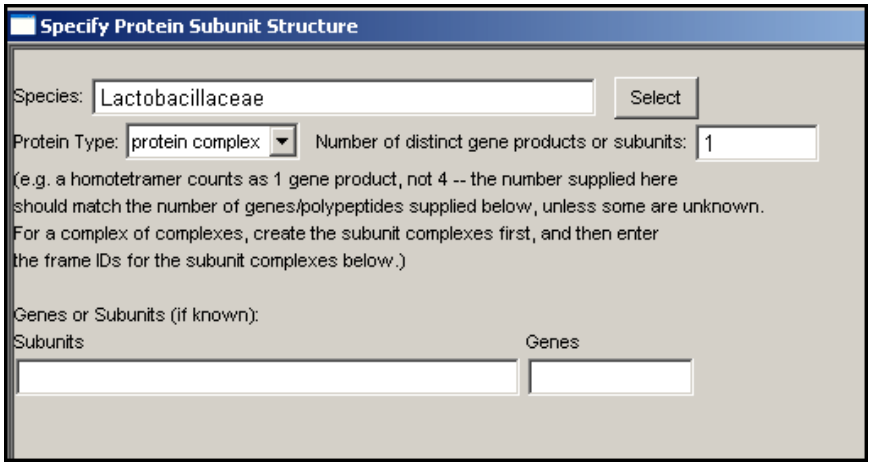


Figure 7.47: Defining a protein complex

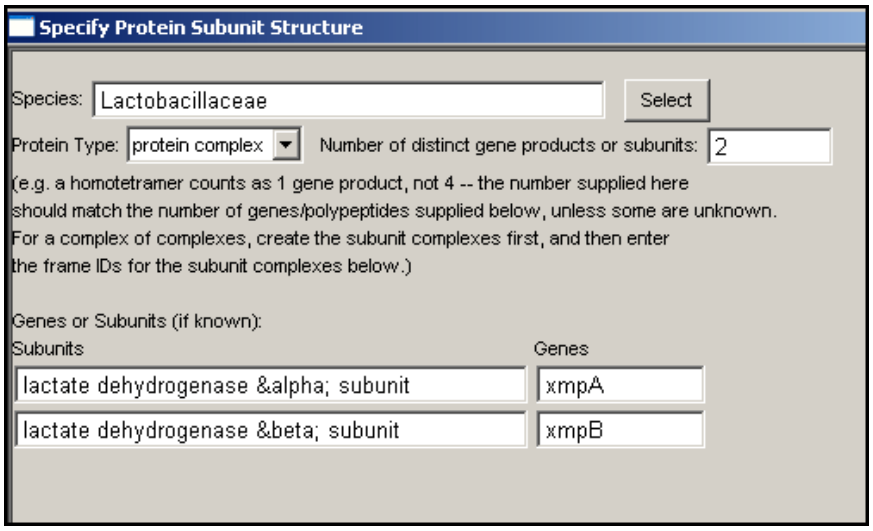


Figure 7.48: Adding a subunit to a protein complex

Specify the number of **distinct** subunits (e.g., if it is a homotrimer, then there is only one distinct subunit) and specify the gene name and subunit name for each. The window will change to add sufficient slots for specifying this information for each subunit. If you know the gene names, type them in the appropriate slots. The software will automatically fill the subunit names with the information associated with these genes. If the gene names are not known, leave them empty and fill in the subunits names manually. For subunit

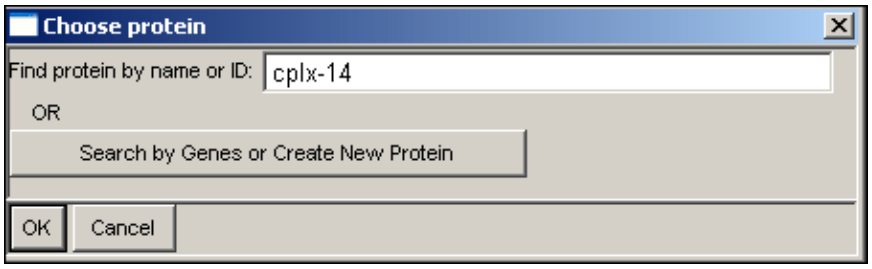


Figure 7.49: A new frame ID is assigned to the new protein complex

name, use the terminology "XXX subunit", as in "lactate dehydrogenase α subunit" or "ribulose bisphosphate carboxylase/oxygenase small subunit".

For this example, we will use a fictitious lactate dehydrogenase enzyme, composed of two α subunits and a single β subunit, encoded by the genes *xmpA* and *xmpB*, respectively (Figure 7.48).

When you click OK, the system assigns an ID number for the new enzyme complex, and shows it on the next window (Figure 7.49). It is not a bad idea to write this ID down for temporary reference. Accept it by clicking OK.

When you click OK, the system will generate frames for the genes, subunits, and the complex that you have specified during this step, and the Protein Editor opens.

7.6.3.5 Specifying Enzymatic Activity and/or Enzyme Names

Note: The protein editor window is very large, so the images shown here reflect only sections of it.

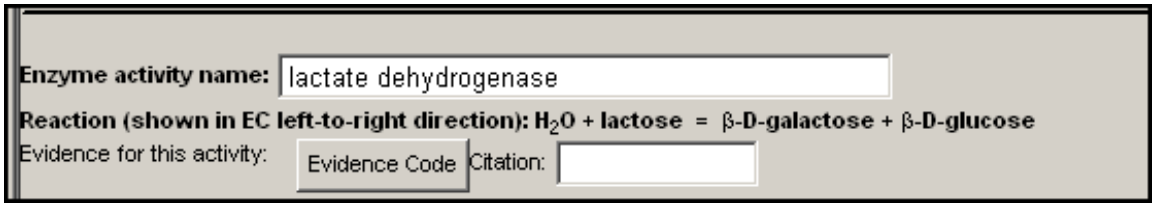


Figure 7.50: Assigning an enzymatic activity name

Scroll down to the second section of this editor. You will see a field called "Enzyme activity name" (Figure 7.50). Type the name of the enzyme activity in here. Note that the "accepted name" and "systematic name" fields of all IUBMB entries are enzymatic reaction names. Enter as many synonyms as possible — remember that when users search the database, they may be searching for the enzyme by different names than the one you chose.

Do not confuse the enzyme name with the enzymatic activity name. In many cases these two names are identical, and in such cases there is no need to specify a different enzyme name. However, in some cases the two names are different. For example, *E. coli* has two isozymes of pyruvate kinase, called pyruvate kinase I and pyruvate kinase II, respectively. For both of these the enzymatic activity name is "pyruvate kinase", even though the full enzyme names are different.

In such cases, the enzyme name, as well as all applicable synonyms, should be entered separately, by clicking on the "Edit Enzyme Name" button in the Protein Editor (Figure 7.51. This open a new window where the enzyme name can be entered, as seen in Figure 7.52.

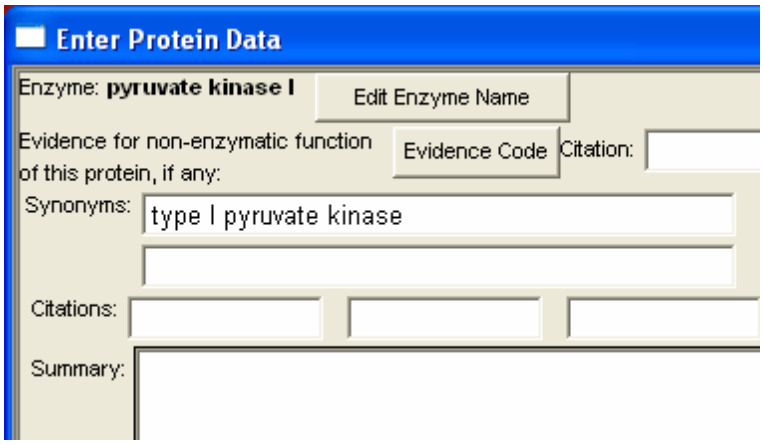


Figure 7.51: The "Edit Enzyme Name" button

Tip: This is an excellent time to close the editor and save the changes, since from here it is very easy to access all the different frames that need further curation — the enzyme, its subunits, and the genes. So click OK and **save your changes**.

7.6.3.6 Curating Full Information for Genes

When the Protein Editor closes, you will see the page that describes the new protein. This page includes the gene-reaction diagram, which in our example looks like Figure 7.53.

Gather all the information you would need for full curation of the frames you have just created.

For the gene, you would need at least the GenBank accession number and a citation for the publication that describes cloning or sequencing this gene. The easiest way to get to the Gene Editor is by right-clicking on the gene in the gene-reaction diagram, and choosing **Edit→Gene Editor**. The Gene Editor window opens, as seen in Figure 7.3. If the subunit encoded by this gene has enzymatic activity, be sure to specify that this gene encodes an

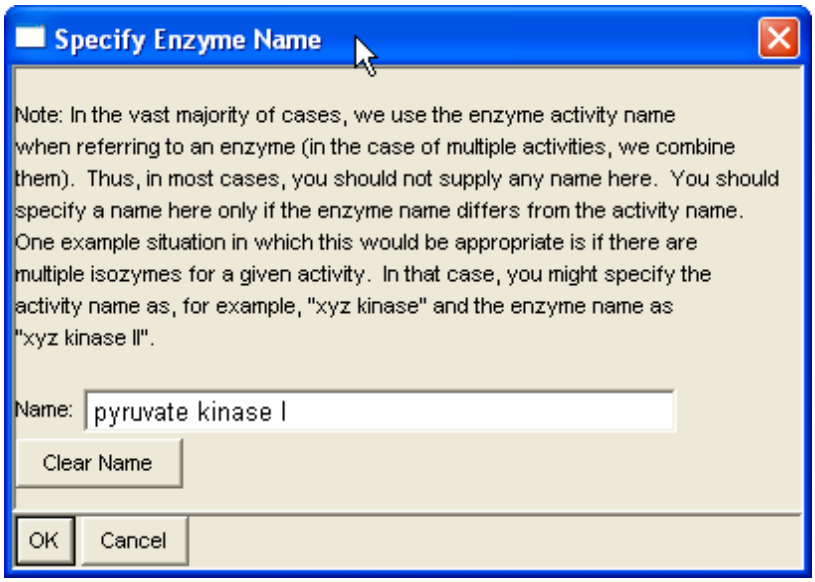


Figure 7.52: Specifying an enzyme name

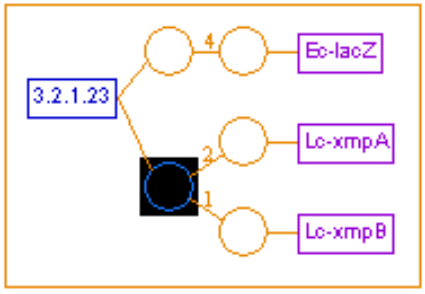


Figure 7.53: The gene-reaction diagram

enzyme (in the Product Types box). However, this dialog box lets you specify other types, such as membrane anchors or regulators.

If known, enter the transcription direction as well. This information is available in Entrez gene entries.

7.6.3.7 Curating Full Information for Proteins

Open the Protein Editor by right-clicking the enzyme name. If the enzyme is a complex, there will be several sections. Scroll down toward the bottom and you will see sections for each of the subunits that make up this enzyme. Each subunit has the name that was specified in the previous step. Next to it is a box called "Coefficient" (see Figure 7.54). Type the number of subunits of this type that are present in the enzyme complex. In our

example, we need to type "2" for the α subunit, and 1 for the β subunit.

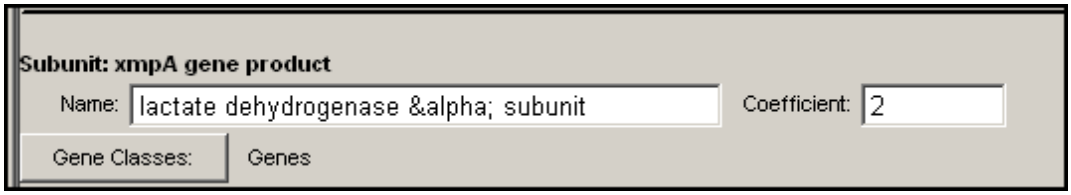


Figure 7.54: The Protein Editor: specifying subunit coefficients

Additional information that should be curated for the subunits includes the UniProt accession number (if available), which you can search at the UniProt Web site. In organism-specific PGDBs, if the gene encoding the subunit is known, the sequence-based protein size is filled automatically. In MetaCyc you would need to fill in this information manually (this information is available at UniProt). In addition, you should fill in the experimental size of the subunit, if this information is available. Include information about the technique used for determining the molecular weight, such as SDS-PAGE or gel-filtration. Be sure to enter a citation of the publication that provides this information. An example for a typical subunit editor section is shown in Figure 7.55.

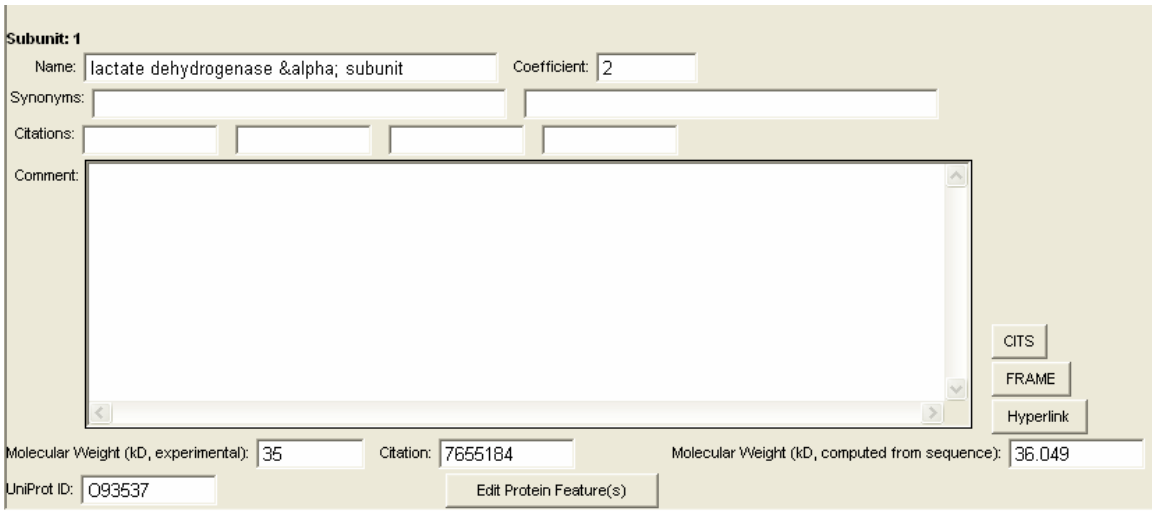


Figure 7.55: The subunit section of the protein editor

As for the enzyme complex itself, there are two applicable sections. The top section in the Protein Editor (Figure 7.56) has fields for some general information, such as the cellular location, experimental size (for the full complex), and the isoelectric point (pI) of the enzyme complex. There are also fields for adding links to additional databases. Any additional data, including useful general information about this enzyme, should be entered in the summary (comment) section.

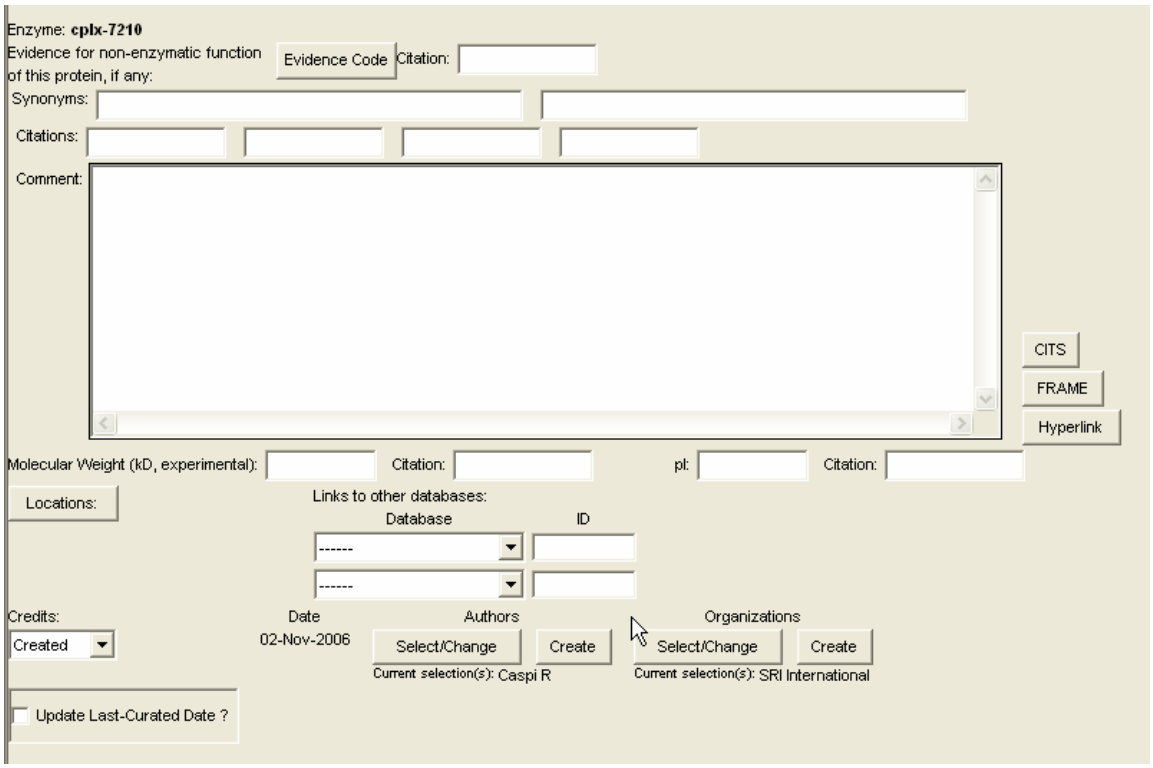


Figure 7.56: The Protein Editor: the top section

The second section is specific to the catalysis of a specific reaction by the enzyme (Figure 7.57). Note that if an enzyme catalyzes multiple reactions, there would be several such sections, one for each reaction. The information curated into this section is actually not stored within the enzyme frame, but in an enzymatic-reaction frame, which describes the relation between an enzyme and a reaction.

In this section you should curate as much information as possible. Available fields include synonyms for the enzyme activity name (enter as many as possible), optimal pH and temperature, reaction reversibility, activators, inhibitors, cofactors, alternative substrates, and Km values for the different substrates. If a molecule that you enter as an enzyme activator or inhibitor is marked as "physiologically relevant" and is present in the pathway, the interaction between this substance and the enzyme would show up in the pathway diagram. Any additional data that is specific for this reaction should be entered in the comment section. Supplement all data with an appropriate citation of the publication that reported it. In addition, every enzyme activity should have an evidence code. For up-to-date definitions of the different evidence codes, go to SRI's evidence code Web page (`http://brg.ai.sri.com/evidence-ontology/downloads/evidence.html`).

Once again, cite an appropriate publication that supports the evidence.

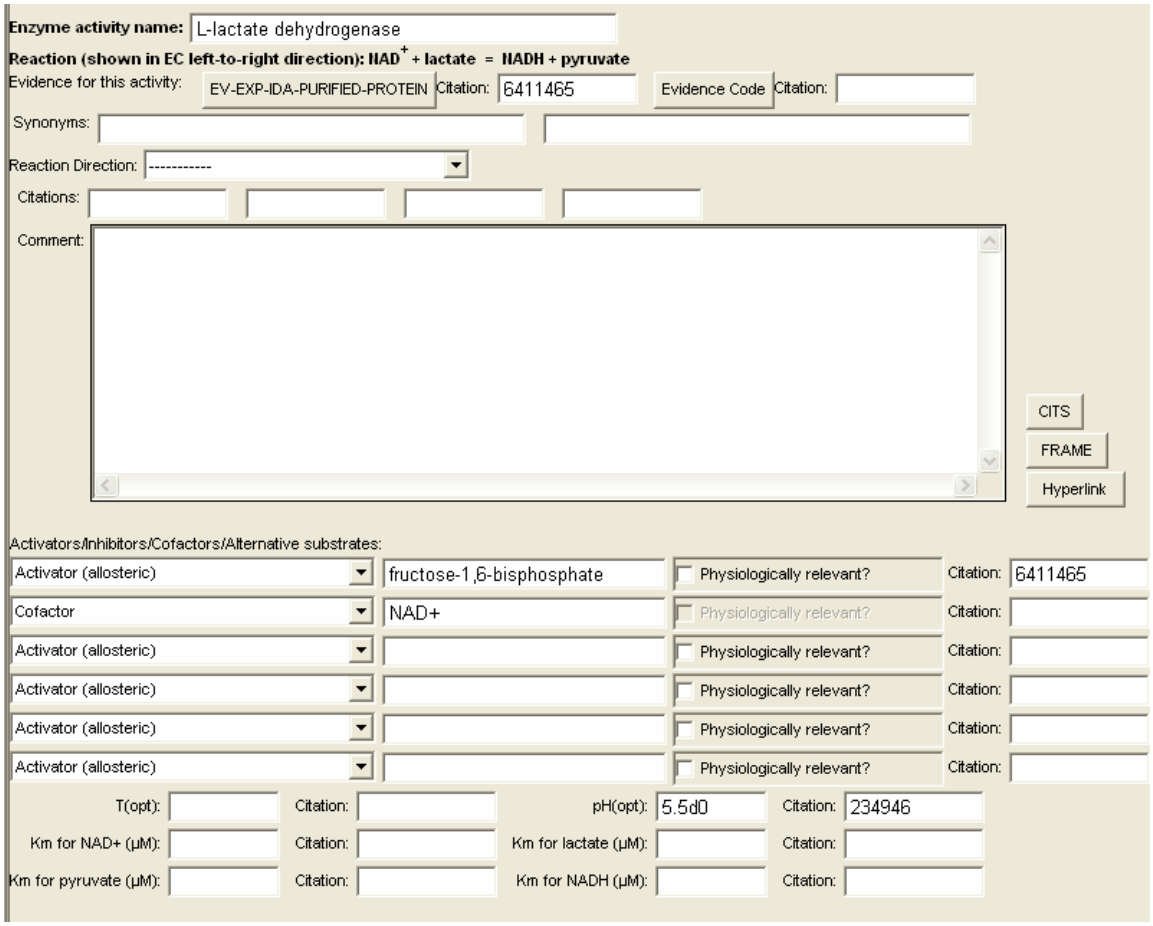


Figure 7.57: The Protein Editor: the enzymatic reaction section

7.6.3.8 Quality Assurance

When you finish curating all the enzymes in the pathway, you are almost done! Before you finish, make sure you have not forgotten anything. Verify that every pathway has an associated class. Make sure all enzymes and pathways have evidence codes. Make sure you have imported all the PubMed references, and look for spelling mistakes.

7.6.4 Exporting a pathway

There are two methods to export a pathway to another database: one where the two Pathway / Genome Databases are loaded in the same Pathway Tools session, and one where the two Pathway / Genome Databases are not necessarily loaded in the same Pathway Tools server.

If both of the Pathway / Genome Databases are on the same server, then you can right-click on the Pathway object handle (as described in Section 7.1.1 and Section 7.5.3), and select **Edit** → **Export Pathway to DB...** . This will pop open a dialog that will prompt you to select a destination Pathway / Genome Database that the Pathway should be copied to.

If you would like to export a Pathway to a Pathway / Genome Database that is not on the same Pathway Tools server, then you can perform the following steps:

1. Right-click on the pathway name, and choose **Edit→Add pathway to File Export List**.
2. Click on the file menu, and choose **Export→Selected Pathways to Lisp-Format File**.

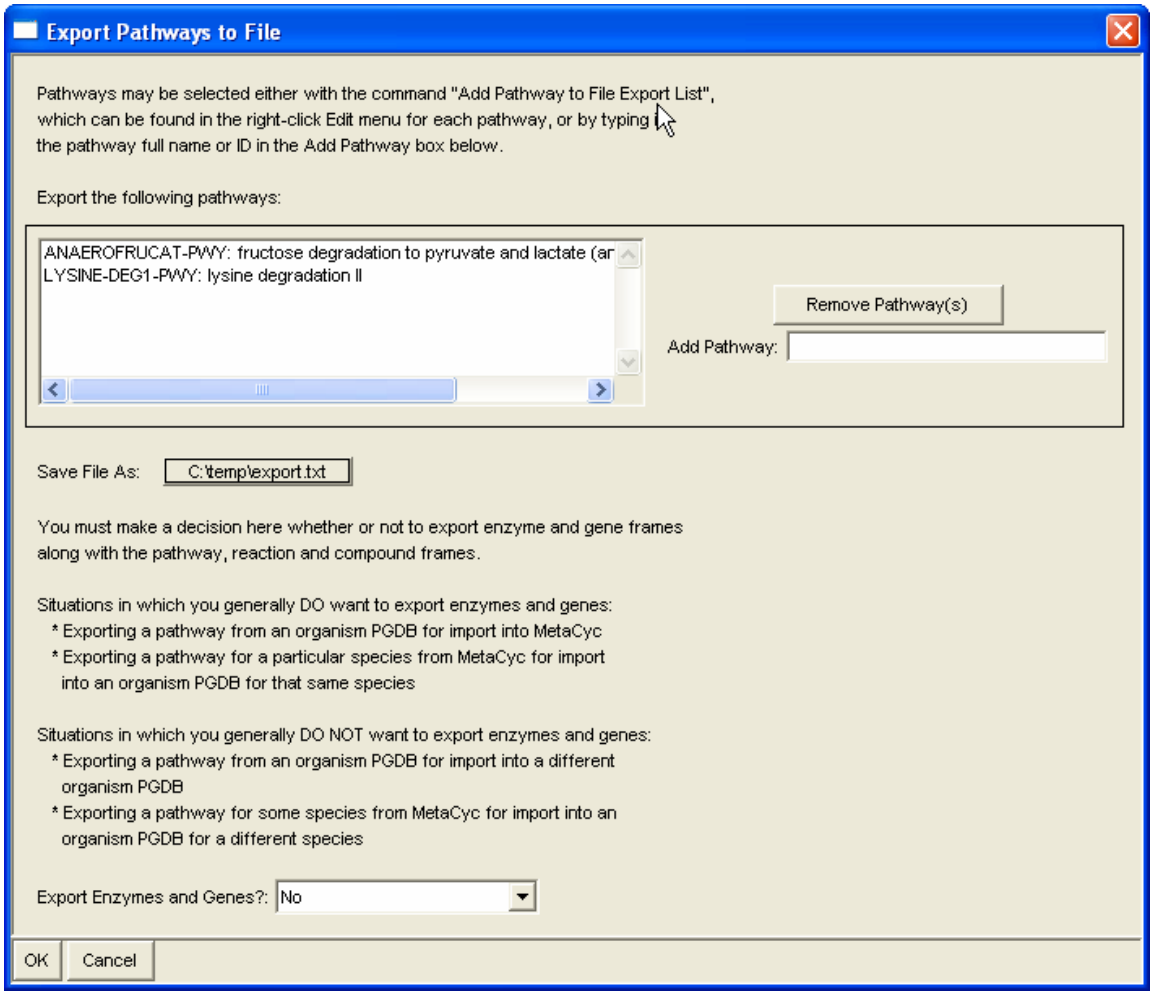


Figure 7.58: Exporting pathways

3. A window will appear, as shown in Figure 7.58. Choose a name and a location for the exported file by clicking on the **Save File As** button.

4. Determine whether you should choose "Yes" or "No" for **Export Enzymes and Genes** as described in the editor, and click **OK** to save the file.

5. In the target database, click on the File menu and choose **Import→Pathways from File**.

6. Choose the exported file and click **OK**.

7. Open the new pathway, and make sure everything looks good.

Chapter 8

Pathway Tools Web Server Operation

There are at least two strong reasons to run Pathway Tools in Web server mode. First to publish on your Web site PGDBs that you have

created using Pathway Tools, for wide access within your organization, or throughout the Web. Second to access software functionality that is not available in desktop mode, such as some comparative tools available within Pathway Tools. For more details on Web mode versus desktop mode, see `http://biocyc.org/desktop-vs-web-mode.shtml` .

The most important step for running the Web server is to correctly configure the relevant parameters in the `ptools-init.dat` file. Please consult Section 2.1.1.

One consideration in operating a Pathway Tools Web server is that the server must be running continually during the time when you expect users to issue Web requests to the server. You may want to define scripts that will automatically start the Web server when its host computer is rebooted. Some hints on how to do this and how to run the Web server as a background process with a display-less X-server are available at `http://bioinformatics.ai.sri.com/ptools/web-logout.html` . If the Web server is heavily queried by users, it is also recommended setting up a script to kill and restart it once daily, as the server performance can get more sluggish over time.

By default, we have defined the Pathway Tools init file such that a Pathway Tools Web server will listen on internet port 1555. This choice will allow Pathway Tools to share a computer with an `httpd` server running on the standard Web port of 80 – two Web server processes cannot listen on the same Internet port on the same computer. Note that an `httpd` server is **not** required when running a Pathway Tools Web server. We do not in fact recommend the use of port 1555 because many firewalls block this port, which will make it difficult for some users to access your server. We recommend using the standard port 80, but in this case be sure that no `httpd` is running on the same computer. On UNIX computers, root access is required to listen on port 80 – the `-user` command line argument can be used to automatically switch to a less privileged account after setting

up on port 80. We have not encountered any security problems with Pathway Tools Web servers.

After installation of Pathway Tools is complete, you can start the Pathway Tools Web server on a UNIX computer by typing the following. This command starts an active Web server running on your computer.

pathway-tools -www *args*

The text *args* consists of zero or more command line arguments, as defined under "Command Line Arguments" in Section 2.3.1. To specify the organism that will be selected by default in the banner in the quick search menu box, use the `-org` command line argument. This would be the default selected organism if the user has no preferred database when logged in or the user is not logged in.

Users can access the Pathway Tools Web server at URL

`http://hostname:1555/index.shtml`

where hostname is the name of the computer at your site on which the Web server is running. The page served by that URL provides a number of different ways of querying the Pathway Tools Web server, and documentation on how to use the Web server. To access this page with a particular organism preselected (i.e., not the default organism specified using the `-org` command line argument), access `http://hostname:1555/<`*ORGID*`>/index.shtml`, substituting the ID for the desired organism for *<ORGID>*.

Note that even when Web server operation is desired, the UNIX environment must still be properly configured for X-Windows operation, as described under "X-Windows Basics" in Section 2.2. The `DISPLAY` environment variable must have an appropriate value, and if necessary, an `xhost` command must have been issued to establish appropriate access permissions. When running in Web server mode, we recommend that the X-Windows compute host and display host are the same computer. Thus, the X-Windows system itself must be actively running on that computer.

When the Pathway Tools Web server executes, it will create a window for the Pathway/Genome Navigator on the display host. That window may be minimized (iconified) during operation of the Web server, but the window should not be destroyed, nor should you invoke the Exit command from the Navigator. In order for the Web server to function, the Navigator window must be kept open the entire time that the Web-server is running.

To stop the Pathway Tools Web server, type the command `(exit)` in the same terminal window in which the original `pathway-tools` command was issued. The program will exit back to the UNIX shell.

If you wish to be able to run a Web server without being logged in, using a display-less X-server, see instructions at `http://bioinformatics.ai.sri.com/ptools/`

`web-logout.html`.

8.1 Operational Procedures

Different groups choose to operate their Web server in different ways. Here are some alternative approaches to consider.

Will the Pathway Tools Web server run on the same computer as the computer on which you edit PGDBs, or a different computer? Either approach can work, but it is safest to insulate the two by placing them on different computers.

If you put them on different computers, you may need to copy PGDBs from the computer used for editing to the Web server. Instructions on moving PGDBs are at `http://brg.ai.sri.com/ptools/faq.html#4`.

Will the Web server make use of the latest development versions of your PGDBs, or frozen stable snapshots of your PGDBs? We recommend the latter approach because the PGDBs can be of higher quality (sometimes bugs creep into PGDBs during editing), and because it is useful to be able to refer to distinct versions of your PGDB. Unfortunately, you cannot use the same Pathway Tools installation to both serve an older version via the Web server and edit a newer version — you will have to make a separate copy of the `ptools-local` directory, each specifying its own default version. At SRI, we release new versions of our PGDBs four times per year. The release process includes the following steps.

- Perform quality assurance on the PGDB by running the Pathway Tools consistency checker (**Tools→Consistency Checker**) on the PGDB to search for and correct errors it may contain.
- For Oracle or MySQL PGDBs, write a backup copy of the PGDB to a disk file by selecting the PGDB as the current DB, and then running the Lisp command `(backup-ekb)`. That command may be run by typing it to the Pathway Tools command prompt.
- Publish the PGDB for downloading by other Pathway Tools users using the Pathway Tools Registry (**Tools→Publish DBs**).
- Create a new version directory for the PGDB (**File→Create New Version For Selected DBs**) so that the file copy of the PGDB will not be overwritten by subsequent save or backup operations. For example, if the new version is version 2.0, and the previous version was version 1.0, version 1.0 is a stable snapshot of the PGDB that can be copied to a separate Web server computer and published on the Web.
- Write release notes for the PGDB that describes major updates we have performed since the last version. See below for serving this as an HTML file.

8.2 Customizing Pathway Tools Web Server Pages

The pages served by your Pathway Tools Web server will generally look similar to those that appear on the BioCyc.org Web site. However, there are a few ways in which you can customize their content and/or appearance. For the purpose of these examples we assume that your httpd root directory is `/home/htdocs/`.

You should keep a copy of any file that you edit or create under the root directory in a different location not under the installation directory of Pathway Tools. This would avoid losing these modified or added files when reinstalling the current or a new version of Pathway Tools. Note that this can be done right before reinstalling Pathway Tools.

You can also specify your own root directory by using the parameter `WWW-Html-Root-Dir` in the `ptools-init.dat` file. You could, for example, put all your static pages in a root directory not under the Pathway Tools installation directory. This may simplify the reinstallation of Pathway Tools without copying your own static web files.

There are two classes of web pages: dynamic and static. For example, the home page `index.shtml` is a static page whereas the output of a quick search is presented in a dynamic page. The static pages are stored under `/home/htdocs/` with either a ".html" or ".shtml" extension. The ".shtml" files can use a virtual inclusion mechanism, as described in Section 8.3, whereas the ".html" files cannot use it. The dynamic pages are generated by the Pathway Tools server and are not residing in the file system of the web server, although generated images are stored in the /tmp directory.

- The home page of the website is the `index.shtml` file in the root directory. You should edit this file to add content to your web site. The Section 8.3 describes the general structure of such a file to get the standard look and feel of the web site.
- To link to a page of release notes for a PGDB, create a directory <***ORGID***>cyc (substituting the ***ORGID*** for your PGDB) in your http root directory, if one does not yet exist. Create a file called `release-notes.shtml` and save it in this directory. For example, if your http root directory is `/home/htdocs/`, and your ***ORGID*** is TEST, you would save release notes to the file `/home/htdocs/testcyc/release-notes.shtml`. Users will see this page when they select the command "History of updates" under Tools$rightarrow$Reports from the top menu bar.
- The list of popular databases that will be shown when using the organism selector page is specified in the file `userWebsiteCustomization.js`. This file is under the directory `htdocs` of your Pathway Tools installation directory. There is a variable named `YAHOO.ptools.popularDatabases` declared as of type array assigned the value "[]", that is an empty array. You can add a list of your own databases identifier as in:

```
YAHOO.ptools.popularDatabases = ["ECOLI", "MTBRV"];
```

This list will be shown in the pane (i.e., the organism selector) which opens when the "change" link is clicked in the standard Pathway Tools banner or one of the menu list commands from the top menu bar.

- To include a small organism graphic to the left of the page header for each data page, save the image file as `/home/htdocs/<`*ORGID*`>cyc/organism-image.<gif/jpg/png>`, substituting the ***ORGID*** of your PGDB, and using the correct file suffix for your image type. Some Web browsers can display text over an image when the user's mouse points to the image; if you want to provide this text for your organism graphic, put the text in file `/home/htdocs/<`*ORGID*`>cyc/organism-image-text.txt`.

- The main CSS style sheet for Pathway Tools Web pages is stored in `/home/htdocs/style.css`. Note that the colors of generated graphics are set to show up best against a white background. For customizing the CSS definitions you should edit the file `userWebsiteCustomization.css` as the `style.css` file may change when a new version of Pathway Tools is installed. For all web pages, the file `userWebsiteCustomization.css` is loaded after the file `style.css` so that you can extend or override any definitions in it. See Section 8.2.2 for more details on customizing the style sheet.

- You may add HTML text to the beginning and/or end of every generated page by creating files `banner.html` and/or `footer.html` in the directory `ptools-local/html/`. The banner though would bypass the look and feel of the standard web site. There is another way to customize the banner by keeping the standard look and feel by using the templates described in Section 8.3.

- The file `PathwayTools.js` is always loaded for all web pages. It contains basic JavaScript code needed to make the web pages functional. It deals with such thing as the login system, the top menu bar, the quick search, the temporary message, etc. It is better to not modify this file. To add new functionality to your website via some JavaScript code, you can modify the `userWebsiteCustomization.js` file. This file is always loaded after `PathwayTools.js` so that you can override any definition in it. You should make a copy of your edited `userWebsiteCustomization.js` file in a secure place before installing a new version of Pathway Tools. See Section 8.2.1 for more information on customizing the JavaScript for your website.

- You may add HTML text in the standard banner by editing the file `temporary-message.shtml` in the directory `/home/htdocs`. You should make a copy of this file in a secure place before installing a new version of Pathway Tools if your http root directory is the default one.

8.2.1 JavaScript Customization

The file `PathwayTools.js` contains the JavaScript code to handle the basic functionality of all web pages. For example, the login mechanism is handled by the code in it. You should not modify this code but add JavaScript code to the file `userWebsiteCustomization.js` if you need to modify or add functionalities to the web pages using JavaScript. The file `userWebsiteCustomization.js` is always loaded after the file `PathwayTools.js` so that you can override any definition in it.

All web pages call the function `initOnWindowLoad` at load time. That is, the tag body parameter `onload` is bound to the expression `initOnWindowLoad()` for all web pages. This function is defined in `PathwayTools.js`. You do not need to modify this function to call your code defined in `userWebsiteCustomization.js`. Two functions are provided for this: `userDefinedBeforePathwayToolsInit` and `userDefinedAfterPathwayToolsInit`. As their name suggest, the former is called before Pathway Tools JavaScript initialization function is called and the latter after it. You define one, or both, as needed, in the file `userWebsiteCustomization.js`. They will replace the current empty function definitions in `PathwayTools.js`. From these two functions, you can call other functions you would need to setup your page. Note that the rendering of the web page is done after all the initialization is completed. This is the usual semantics of the onload parameter of the HTML tag body.

Typical JavaScript code sets parameters to already existing elements of the page or adds content to the page. The parameters can be simple values as well as JavaScript expressions that use functions that you define in the file `userWebsiteCustomization.js`.

8.2.2 CSS Style Sheet Customization

The look of your web site largely depends on the content of the style sheet(s) used. A style sheet is also known as a Cascading Style Sheet (CSS). A style sheet is a file containing definitions to control the appearance of various HTML elements.

Pathway Tools used several style sheets each one affecting different components (e.g., menu bar, login). The main style sheet of Pathway Tools is `style.css`. It is under the `htdocs` directory of your Pathway Tool installation directory. It is not recommended to modify that file. Instead, for CSS customization, only one file should be modified: `userWebsiteCustomization.css`. This file is loaded last for all Pathway Tools web pages. Because the last loaded CSS file can extend or redefine any previous style sheet definition, you can, in particular, redefine or extend all definitions appearing in `style.css`. Naturally, you should consult `style.css` to find the current style sheet definitions used in Pathway Tools.

For example, the temporary message look is controlled by the following definition in

style.css:

```
div.temporaryMsg {
  display: none;
  border: 0px solid black;
  float: left;
  height: 50px;
  /* The width is modified by the JavaScript */
  width: 0px;
  margin-top: 3px;
  margin-left: 20px;
  margin-bottom: 3px;
  overflow: hidden;
}
```

As can be seen, the temporary message has a 0px (i.e., zero pixel) border since the `border` parameter is declared to have a 0px width. That is, no border will appear around the message. You could change this, without affecting all the other parameters, by declaring the following definition in `userWebsiteCustomization.css` file:

```
div.temporaryMsg {
  border:  1px solid black;
  padding: 5px;
}
```

This would create a thin 1px black border around the message. The padding between the message and the border is also given in this definition as 5px. Any other parameters for any other definition can be modified in a similar manner. Parameters can also be added at will according to the CSS syntax.

Similarly, new definitions can be given in the file `userWebsiteCustomization.css` to control the look of HTML elements in your page. No other style sheets in Pathay Tools will override these definitions.

8.2.3 Adding, Removing, Modifying the Top Menu Bar

The standard top menu bar for the web pages of Pathway Tools can be modified to suit your web site. This can be done in different ways depending if the modifications are static or dynamic (e.g., a command is added based on parameters).

The file `template-top-menubar.shtml` contains the standard Pathway Tools top menu bar definition. It is a HTML file with a series of `<li>` tags, one for each top menu command. The style sheets and JavaScript used for this menu bar comes from the Yahoo libraries. The complete definition of how to alter such a menu bar could be found at

their website: `http://developer.yahoo.com/yui/menu/#using`. But it can readily be seen from this HTML code how to add new top menu bar commands and menu commands.

For example, to add on the left of the top menu bar the command "Great News" that would go to a web page named "great-news.shtml", you would add the following HTML code:

```
<li class="yuimenubaritem"><a class="yuimenubaritemlabel"
    href="/great-news.shtml">Great News</a>
</li>
```

right after the `<ul class=toolbar>` tag. Naturally, to add such a command at any other place, you would insert this HTML code before or after the appropriate existing `<li class="yuimenubaritem">` tags.

If you modify the `template-top-menubar.shtml` file, and wants to use it again when reinstalling Pathway Tools, you should keep a copy of this file, with your modifications, in a different directory, not under Pathway Tools installation so that you can replace the standard `template-top-menubar.shtml` file provided in Pathway Tools installation with your own saved one.

You can also dynamically add some commands by using JavaScript code. This code should be added to your `userWebsiteCustomization.js` file in the function `addSiteSpecificMenuItems`.

For example, the following would add a command "Logical Based Search" to the list of commands under the top menu bar command "Search":

```
function addSiteSpecificMenuItems () {
 var searchMenu = YAHOO.widget.MenuManager.getMenu("search");
 searchMenu.addItem ({url:"log-search.shtml",
                      text:"Logical Based Search"});
}
```

Other operations are possible, like removing and deactivating some commands dynamically. The complete list of operations is available on the Yahoo website: `http://developer.yahoo.com/yui/menu/#using`.

8.3 Template Files and HTML Virtual Inclusion

There are predefined template files to create the standard web sites with the top menu bar, the temporary message, the banner with logo, the footer, login, quick search, etc. These template files use a dynamic include mechanism (aka virtual inclusion) implemented in

the web server. This mechanism can be used only in files with the extension `.shtml`. In particular, files ending with `.html` are **not** parsed for virtual inclusion.

For example, the file `template-part2of3.shtml` is similar to the following:

```
</head>
<!--#include virtual="/template-beginning-body.shtml" -->
<!-- Ending of template-part2of3.shtml -->
```

The line with the "virtual=" indicates to include verbatim the content of the file "/template-beginning-body.shtml". The slash indicates that the file should be found in the http root directory. In general, a relative path can be present. It is always relative to the root directory of the web server. If no path is specified, the file is either in the same directory as the parent file (i.e., the file doing the virtual inclusion) or in the root directory, in that order. If the file is not found, no inclusion occurs. This simple mechanism allows you to create "template files".

There are several predefined template files provided in the installation of Pathway Tools to implement the look and feel of the standard web site. These are (in alphabetical order):

- `template-after-beginning-body.shtml` — it is a rather complex file that includes the entire login HTML code, quick search box and button, the database selection mechanism, and the top menu bar.
- `template-before-head.shtml` — a simple file that is used to declare the type of the document in a web page. It currently does not specify any declaration of the type of documents to expect in the web pages, such as character set encoding or language.
- `template-beginning-body-dynamic.shtml` — applies to dynamic pages only. It is similar in fonction to the following static version.
- `template-beginning-body.shtml` — contains the body tag with the parameter onload specified to call the standard JavaScript code to initialize the page. It also includes the standard top banner via the `template-after-beginning-body.shtml` file.
- `template-closing-head.shtml` — the necessary tag to close the head of the page is in this file. It is typically short but could have some more links to include more CSS and/or JavaScript files that are deemed too important to be overridden by the regular web static pages.
- `template-closing-html.shtml` — typically a very short file that simply include the standard `template-footer.shtml` file and the closing tags of the page.

- `template-footer.shtml` — it is included by the `template-part3of3.shtml` file mentioned below. It typically contains the standard website privacy policy and disclaimer of a web site.
- `template-part1of3.shtml` — this file as well as the two following ones are the backbone of the templates used in the standard web site. This one is included in every static web page of the standard web site. After this file you can assume that your own HTML code is the head of the web page. The standard head content has been included once this file is included.
- `template-part2of3.shtml` — this file closes the head and includes the standard beginning of the body of the web page. Therefore, any HTML code following its inclusion must assume to be in the body of the page, and not in the head for example. For

  the standard web site, the banner, and all its content, in particular the top menu bar, is included by it. Typically, the content of the page
- `template-part3of3.shtml` — the third part of a static web page that uses the standard look and feel of the website.
- `template-standard-head-content.shtml` — contains the CSS and JavaScript references needed by the rest of the web page.
- `template-top-menubar.shtml` — contains the HTML code for the top menu bar.

The three template files `template-part*of3.shtml` are high level templates that directly or indirectly include all other template files. Most static pages include these, and only these, to get the standard look and feel of the web site. So a typical static web page would have the form

```
<!--#include virtual="/template-part1of3.shtml" -->

... perhaps some HTML code for the title of the page ...
... perhaps some links to include more CSS and JavaScript ...

<!--#include virtual="/template-part2of3.shtml" -->

... the main part of the file, the content of the web page ..

<!--#include virtual="/template-part3of3.shtml" -->
```

The creation of new static web pages can follow this pattern in almost all cases. Typically, there is not much HTML code between part1 and part2: a simple line that contains

<title> and </title> is used to put a title in the status window of the browser (not the content of the web page). As can be seen, these three high level templates make it easy to write new static web pages without worrying about JavaScript, CSS files, and all the rest of the look and feel of the website.

In some cases that you would like to specify your own body parameters, you could follow the following template:

```
<!--#include virtual="/template-part1of3.shtml" -->

... perhaps some HTML code for the title of the page ...
... perhaps some links to include more CSS and JavaScript ...

<body class="yui-skin-sam Mainpage" onload=initOnWindowLoad()>

<!--#include virtual="/template-after-beginning-body.shtml" -->

... the main part of the file, the content of the web page ..

<!--#include virtual="/template-part3of3.shtml" -->
```

In this case, the class name "Mainpage" was added to the body tag. This is the typical way to CSS control the home page of the web site in a different way from the rest of the web site.

8.4 Setting Up BLAST Access

By default, the command Search→BLAST provides the option of performing a BLAST search for some sequence within the genome of a single organism (see the description of the command line argument `-no-blast` to disable this functionality). The Pathway Tools distribution does not include a copy of the BLAST software. For installation instructions, please consult `http://bioinformatics.ai.sri.com/ptools/installation-guide/released/blast.html`.

Once BLAST has been installed, the BLAST sequence databases for each organism must be created before Web users can access them. Select the menu command **Tools→Prepare Blast Reference Data→Both**.

8.4.1 Web Server Log File

The Pathway Tools Web server logs its requests in the file `aicexport/pathway-tools/ptools/<version>/logfiles/server.log`.

You may wish to copy the logfiles in this directory to a more permanent location when installing a new version of Pathway Tools. A corresponding `error.log` is also produced, which may be useful when debugging problems.

8.5 Creating Links to Pathway Tools Pages

For most generated display pages, the URL remains constant, so you can visit the page that you wish to link to using the normal querying or browsing tools, and copy the URL in order to create your link. However, there are a few links that you may wish to create whose URLs cannot be reached using the normal interface. These are described here.

8.5.1 Omics Viewer

In order to use the Omics Viewer through the Web interface, users must fill out a form and upload a data file. This makes it difficult to share omics displays or for you to offer omics displays of your own internal data files. Thus, in addition to the more usual POST interface, we also offer a way to request an omics display using a GET request, specifying the filename or URL for the data in the request string, along with any other display parameters. The format of the URL is as follows:

```
http://hostname:port/<ORGID>/overview-expression-map?url=<url>&
expressiontype=[relative|absolute]&log=[on|off]&numcolumns=[1|2]&
column1=<column-numbers>&column2=<column-numbers>&
class=[gene|compound|reaction|NIL]&color=[computed|specify|3-color]&
maxcutoff=<number>&threshold=<number>&display=[cellular|genome|table]&
tablethreshold=<number>
```

Any of the parameters that have default values may be omitted from the URL. The meanings of the parameters are as follows:

url – either a URL (beginning with http://) or a filename. If a file is specified, its path will be relative to your HTTP root directory. The URL should be properly escaped (i.e. any internal special characters such as &, = or ? should be replaced by their corresponding hex values).

expressiontype – specifies whether the data values should be considered relative (distributed around some center value of either zero or one) or absolute (all positive, no center). Defaults to absolute.

log – if on, the data is in log format, or centered around 0; if off (the default), the data is in linear format, or centered around 1.

numcolumns – if 1 (the default), the data is taken from a single column; if 2, a ratio of two columns is used.

column1 – the single column when numcolumns=1, or the numerator when numcolumns=2. If a static display is desired, this should be a single number. When generating an animation, multiple columns should be specified, separated by +, e.g. 1+2+3. Note that 1 here refers to the first potential data column, i.e. the second column in the file, since the very first column should contain object names or identifiers.

column2 – this need be supplied only when numcolumns=2, and represents the denominator. Its format is as for column1.

class – specifies the types of identifiers in the first column, whether genes (the default), compounds or reactions. If NIL is specified, then the identifiers could be of any type.

color – if computed (the default), the color scheme will be computed from the data values; if specify, a full color spectrum will be generated with a maximum cutoff indicated by the maxcutoff field; if 3-color, a three-color display will be generated using the threshold argument.

maxcutoff – supply this only if color=specify; it should be a number to be used for the maximum color bin.

threshold – supply this only if color=3-color.

display – this parameter can be supplied multiple times if multiple views are desired. If it is not supplied, it defaults to cellular.

tablethreshold – if a table display is requested, the table will include all pathways with at least one gene (or reaction or compound) whose data value exceeds this threshold.

8.5.2 Pathway Images

The Pathway Tools Web server generates its images as temporary GIF files. You cannot create permanent hyperlinks to these image filenames, as the files are deleted on a regular basis. However, you may embed a customized pathway image in another page using the following URL template:
`http://hostname:port/<ORGID>/diagram-only?type=PATHWAY&object=<PWYID>&detail-level=`

If the detail-level parameter is omitted, then the default level of detail will be used, which depends on the size and complexity of the specific pathway. The meanings of the possible detail-level values are as follows:

0 – Minimal detail – only endpoint and branch-point metabolite names are shown.

1 – All metabolites along the main pathway backbone are shown.

2 – All metabolites, enzyme and gene names are shown

3 – Same as 2, plus structures for metabolites along the main pathway backbone are shown (except for some ubiquitous metabolites)

4 – Same as 3, plus structures for the side metabolites are shown.

You can retrieve an image map for a pathway diagram generated in this manner using the following URL template: `http://hostname:port/<ORGID>/imagemap-only?type=PATHWAY&object=<PWYID>&detail-level=[0|1|2|3|4]` This will enable the internal objects (compounds, reactions, proteins, etc.) in the diagram to be hyperlinked to their corresponding pages. Note that some additional javascripting will be required in order to correctly link up the imagemap to the image.

8.6 Web Accounts

The optional Web Accounts system allows a Pathway Tools Web server site to support tracking and differentiation of users through an internal user account system. Web site users may sign in and obtain accounts in your system, using their email addresses as account names, along with a user-managed password. This allows the site to tell the difference between users. Then each user can define preferences as to how the site is presented to them. Users also register with their physical addresses, contact information, and specialties as they sign up, which allows the people running the site the chance to learn about and contact their users. Finally, the Web Accounts system will be the base for a number of upcoming features.

The Web Accounts system for Pathway Tools Web servers is optional but recommended. It is not necessary for a site to provision and use this system. However, it should be useful to larger installations.

The Web Accounts system is also intentionally designed to be optional for the users. A user will not be blocked out of the system merely because she does not log in. However, it will not be possible for that user to define individual preferences.

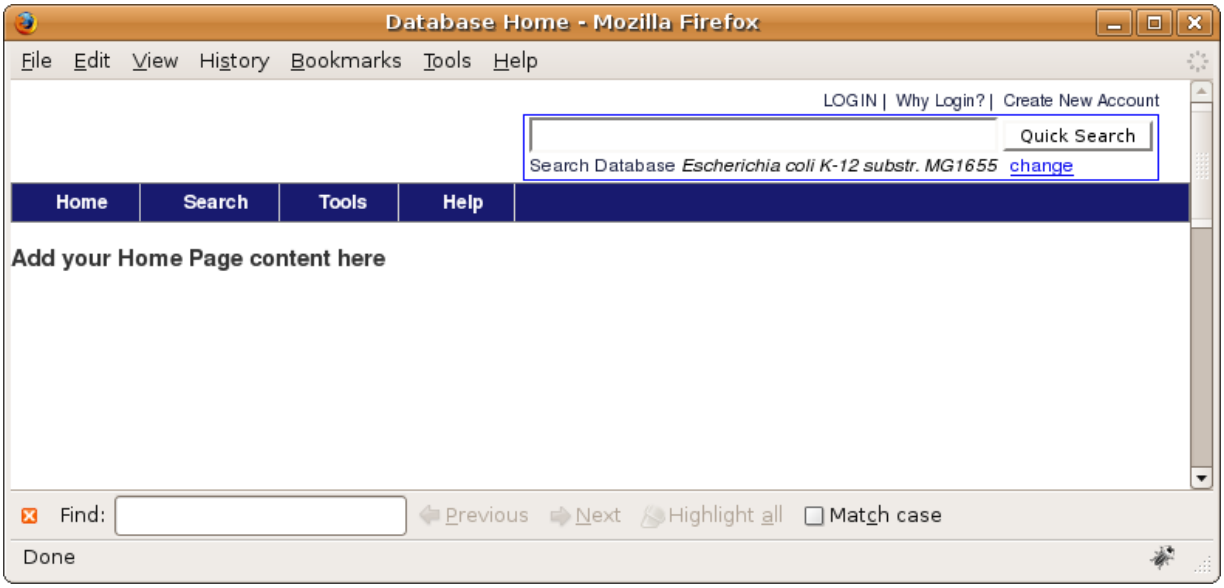


Figure 8.1: Position of the Login Panel at the top of the page

8.6.1 How To Use The System for Everyday Users

8.6.1.1 The Login Panel

The Login Panel is found in the upper right-hand corner of all pages, including the Pathway Tools Home Page `index.shtml`. Its commands support the following operations.

8.6.1.2 Creation of New Accounts

To sign up as a new user, simply click on the Create New Account link at the top right of the main page, then follow the directions.

You'll be asked for your full email address, which will be your account name. Type your password underneath this, and then type the `same` password in the "Repeat password" box so that you can be sure you didn't make a typing mistake. Your first name and family name are required, as is the checkbox answer to the question of whether you're a Principal Investigator (PI) or not. All other questions are optional. Hit the "Create Account" button at the bottom right of the panel after you've finished. The system will not let you submit if the required information has not been filled out, or if the two copies of your password don't match. In this case, the missing item will be outlined in red.

8.6.1.3 Login by Existing Users

The full Login Panel opens up when you click the word "Login" in the upper right-hand corner.

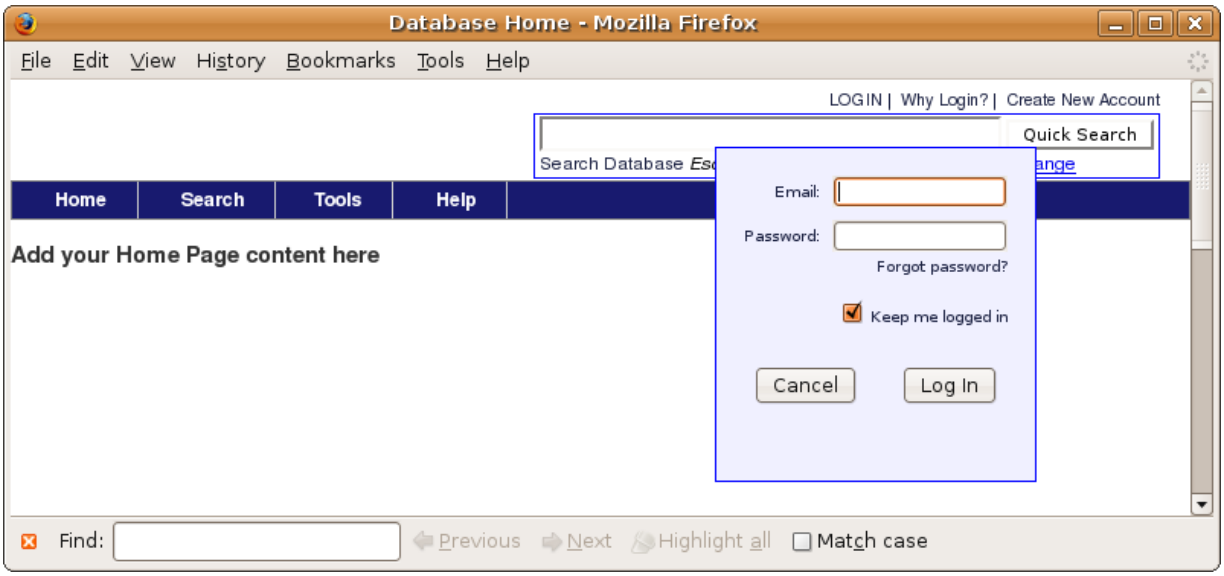


Figure 8.2: The full login panel, opened by mousing over "Login (Optional)"

Type in your full email address, which is your account name in this system, and your previously-registered password. You may use the tab key to jump from field to field if you wish. Then hit the Log In button to log in.

It is required that cookies be enabled in your browser in order to log in. Pathway Tools supports popular browsers such as Firefox and Internet Explorer (IE).

If you have forgotten your password, click the "Forgot password" link, and it will be mailed out to your email address that you gave.

8.6.1.4 Changing User Preferences

Preferences are set on page preferences.html, where you can change settings on how you want the system to show up for you.

Use the tab folders at the top to flip between panels. You can also change your password or your contact information, if you want to, under User Information.

Be sure to hit the "OK" button at the bottom right to lock in your changes. Or hit the "Cancel" button, or simply change away from the page, if you decide not to change your preferences.

8.6.1.5 Logging Out

If you do not check the Remember Me box upon login, the system will log you out when you close your browser application (the Firefox or Internet Explorer program).

If, however, you check the Remember Me box, you stay logged in until you explicitly log out, even if you close your browser application and turn your machine off. This is useful if you are a single user on your computer. Use the "Log out" link at the top right of regular pages in order to log out. Make sure to log out if you are using a public computer or if someone else might be using your computer.

8.7 Installation and Operation of Web Accounts System By System Administrator

8.7.0.6 Web Accounts Database

First, create a MySQL database on a database server computer, using your MySQL administration tool that came with your MySQL installation. The database server computer does not have to be the same computer as your regular web-server computer. Write down the database account and its password used to access this database, as these will be required for the step following this paragraph. Define the database tables by evaluating file `aic-export/mysql/web-accounts.sql`.

8.7.1 Configure the Pathway Tools Initialization File

Settings for the Web Accounts system are found in the Pathway Tools `ptools-init.dat` file as described in Section 2.1, and must be configured for proper operation.

8.7.1.1 Setting up reCaptcha

In order to prevent spammers, hackers, or other malicious users from automatically creating fraudulent Web Accounts, Pathway Tools utilizes the reCaptcha visual-word human authentication tool from Carnegie-Mellon University. A CAPTCHA is an obscured image depicting a series of alphanumeric characters that can be deciphered easily by a human but would be difficult for a computer algorithm.

A picture of two words is shown in an authentication box, found at the bottom of the Web Accounts sign-up page and preferences editing page `/preferences.html`, right

before the Save button. Then the user has to type both words from this picture into the authentication box entry, before submitting the form with the `Save` button, in order to ensure a successful submission. Failure to type both words (separated by a space) will result in the form being rejected. In this case the user should hit the `Back` button on the browser and try again.

In order for your site to configure the **reCaptcha** system, it is necessary for your site to have both a special reCaptcha public key and private key for security. These two keys may be obtained for free by signing up for a **reCaptcha** account with CMU at website `http://www.recaptcha.net`. The procedure is simple and only takes one page. **reCaptcha** will then issue you both your public and private security pass keys for your website server domain. Each key is a string of letters and digits, currently 40 characters long.

The security keys are issued to a particular domain, that is, `anything.anything.yoursite.com`. The **reCaptcha** system locks out any web server that is attempting to display a **reCaptcha** authentication box from a server that is not authorized with a corresponding private key.

Note however that only the last two parts of the host name (i.e., `yoursite.com`), in the URL have to be signed up for authorization; all subdomains are also authorized automatically. So `example1.yoursite.com` and `example2.yoursite.com` should work for you using just one key pair. Also, if you have multiple domains, **reCaptcha** lets you sign up separately for each one, and manage them all together from your **reCaptcha** account.

Your reCaptcha pass keys need to be placed in your `ptools-init.dat` configuration file. The relevant variables are:

- `*User-Account-ReCaptcha-public-key*`
- `*User-Account-ReCaptcha-private-key*`

Each key is a single string of mixed letters and numbers, typically 40 characters long. Double-quotes at the beginning and end of the string are not necessary.

In order to test out the operation, it may be useful for the site system administrator to pull up the test harness page: `http://yoursite.com/test-recaptcha.html`

This test page displays a live **reCaptcha** panel from your website. It is currently 3" x 1" in maroon and gold. If you can see this panel inside the blue box, then you know that at least your private key has been configured properly.

Type the two black words from the image into the gold box in the lower left, and then hit the `Submit` button down at the bottom.

The results of the previous submission are displayed underneath the current **reCaptcha** panel.

If the results of the previous **reCaptcha** are displaying "Authorization: PASS", then you know that your public key has been successfully configured as well.

Using **reCaptcha** requires access from your server out to their secure server for authorization over HTTP. If your firewall prevents access to the outside world, it may be the case that you cannot run **reCaptcha**.

If you ever want to disable the **reCaptcha** system, simply do not configure both the `*User-Account-ReCaptcha-public-key*` and `*User-Account-ReCaptcha-private-key*` variables. Leave these both blank or commented out. This will set these variables to NIL, which will disable the system.

8.7.1.2 Tips

As described in Section 3.10.13, Pathway Tools includes a "tip" feature that displays helpful documentation on various features of the software.

As of version 12.5, the tip system is always enabled in web mode.

If user accounts are enabled, a table in the web accounts database is created which stores the state of the tip system (i.e., which tips a user has viewed, etc.) for each logged-in user. Non-logged-in users are tracked via cookies.

If user accounts are disabled, all users are tracked via cookies.

Currently only logged-in users may unsubscribe from the tip system.

8.7.1.3 Test the System

The system should be tested by using the test harness page `/test-prefs.html`, signing up, and then asking the system to send you your password through the `Forget Password?` Link.

Go to page `/test-prefs.html`. There should be no session variables under the `*user-preferences*` variable, since you're not logged in yet. You should be able to see a blue "plastic bubble" file tab in the center of the screen. All filenames underneath this tab should resolve to pathways in black, and not be listed as "not found" in red.

Sign up for a new account with your email address. The system should send you an email confirming your password, and welcoming you to the system.

Check the `/test-prefs.html` page again. You should see a full list of more than 20 parameters. This indicates the system is tracking signed-in users with session information.

Sign out, then hit the `Forget Password?` link, and give it your email address. The system should send an email with your password out to you.

Check your email, and make sure you received the mails the system sent.

8.7.1.4 Ask Users to Sign Up

Users should use their email address as their account name when signing up. This is required in order to support the automatic emailing of forgotten passwords.

8.7.2 Disabling Web Accounts

To disable the use of Web Accounts by this Pathway Tools Web site, the Server Hostname, found in the Pathway Tools `ptools-init.dat` file, should be cleared to contain no value. This clears global flag `*use-login-preferences-system*`, which then refuses to call `handle-restoring-session-for-this-user`, the entry point for sessions behavior.

8.7.3 Background: Which files are required to be included?

All of the includes are referenced from the `cwest::*html-root-directory*`. If the system does not have its Web site-includes stored in the target directory, various different functionalities of the system will break. When the system is not working as expected, the first thing to do is to check and see whether this html root directory has been reset.

The runtime system pulls out key components of the Web Accounts system as users request html pages, and serves these up as includes in the pages. These include:

tabber.js This is the engine that drives the machinery for dynamically creating and running the tab folders on the page after the page has been loaded. Without this, the tab folders simply don't show up.

looks This directory contains various .CSS files that determine the background color, the foreground color, and the default font text size of the page presentations–not simply the tab folder pages, but all Web pages. Note that, even when these do get included in the Web page, sometimes they may be ineffectual due to other .CSS style sheets being included by different parts of the system. Files are named for their looks, including `black-on-white.css`, `colors-on-black.css`, `normal-text-size.css`, etc.

dropdowntabfiles This directory contains `dropdowntabs.js` and various `.CSS` files that support various looks for the folder tab appearances. The important file here is `slidingdoorsM.css` which implements the Macintosh X grey plastic bubble look.

Other looks are not supported. The `dropdowntabfiles` directory also contains subdirectory `media` which implements the gif image support for the looks for the folder tabs. The important files here are `slide-left.gif`, `slide-leftB.gif`, `slide-right.gif`, and `slide-rightB.gif`, which together support the appearance of the grey Macintosh X plastic bubble tabs that light up blue.

8.7.3.1 Usage Considerations

8.7.3.1.1 System and Browser Caching In some configurations, the Pathway Tools system will create a page and then remember it. Then the next time this page is asked for, the system will return the previous page instead of going to the trouble of creating a new page. This is called "caching". In some configurations, your browser will also cache a page that you visited previously. Then it will serve that page up to you instead of going to the trouble of downloading it all over again.

When you are working with a user accounts system, occasionally this kind of behavior can cause problems. In particular, users can get confused when the browser re-shows a page that shows they were logged in or logged out, even when it is the opposite and they have changed their state. And administrators can occasionally get fooled by a test-prefs.html page that got cached and is re-serving old information. This occurs especially with the Internet Explorer browser.

The conclusion is to hit the browser reload button when you are not sure if the page is accurate and up-to-date or not. This will at least solve the browser cache problem.

8.7.3.1.2 Flushing the User Tracking System The Web Accounts system user tracking component was designed for a system that gets restarted around once a week or more often, with up to a few tens of thousands of users. If however the system runs in an environment where the server is expected to run for, say, a year without being restarted, or there are several million active users, then the system may become slow. Try restarting the Pathway Tools system in this case.

8.7.3.2 Known Failure Modes

8.7.3.3 Can't Log In, Even Though The Login Screen Comes Up

This happens when the html root directory has been mislaid. Or, it's also a symptom of cookies not working properly in the browser. Neither of these should happen under normal operation.

8.7.3.3.1 Login Cookie Inconsistent With Database This condition can occur in at least one of three rare cases: (1) The user *was* logged in, has not explicitly logged out, and then sometime later the user database got wiped and the user is no longer found in the database; (2) The user is logged in, and the database computer crashes after its database link has been opened but before the verification has been pulled; (3) The user is a hacker who has borrowed a cookie from someone else, has maliciously modified it, and is testing the system to try to break into it.

None of these are probable enough that they need to be worried about.

The system now handles case (1), obsolete live logins, relatively gracefully by taking a no-show on the login database query to be the same as not logged in. Significantly, this lets the Preferences page come up with a blank slot in the user email name, even though the cookie is saying that the user has a perfectly good user ID and user email. This allows an obsolete login to look like a new user trying to sign up.

The other two cases are handled in this manner as well.

8.7.3.3.2 No User Profile or No Preferences Profile Even Though User Login ID Exists Both a user demographics profile and a preferences profile are automatically created every time a new user signs up. It is theoretically possible, however, since these are kept in different database tables, for the user login entry to exist in one database, but one of the other database entries to not exist in the other databases.

These cases will in general only happen in the rare case that a system administrator decides to go in and manually delete an entry row from one or two of the databases, but not all four. There is no reason to manually edit the database, so there is no reason for this state to occur. This does not mean it will never happen.

In these cases, the system is supposed to gracefully degrade and keep going. In certain cases a new blank entry may be created; in other cases this may be simply ignored. The system is not supposed to break. This has been implemented but not tested extensively.

8.7.3.4 The Test Harness

/test-prefs.html This page prints out the current value of the system glue variable `*user-preferences*`, which is then used by the rest of the system. As this variable is restored live for each user at each page view by the sessions code, this not only tests out the glue but also tests out that the cookie-based sessions code, as well as the login code, is working properly. **Note that when a user is not logged in, there is no `*user-preferences*` content. This is proper behavior.**

8.7.4 Technical Details

The Web Accounts system implements a tightly-integrated capability for the Web version of Pathway Tools to support allowing users to log in, and allowing users to work with a variety of preference settings. These are supported by a back-end database that keeps track of the users, their personal contact information, and their desired preferences.

Users are not locked out from usage of the system when they're not logged in. However, the system does not track their preferences in this case, and they're forced to use the system defaults. The system knows if a user is logged in or not, and serves each page accordingly.

A logged-in user is represented by a cookie kept on the user's machine. Then that same user is logged in as long as the user desires, until the user explicitly logs out, or until the user's cookies are flushed. If the user checks the Remember Me box, a persistent cookie is used to keep the user logged in even after the browser application (e.g., Firefox) is closed. If the user unchecks the Remember Me box, a session cookie is used to keep track of the user. However, this expires after the user closes the browser application.

When a user logs out, this cookie is set to a user ID of 0. Having no cookie at all also indicates a user is not logged in.

Each time a user requests a page to be served, his credentials are checked in a cache of all logged-in users. Then a session variable list of all of that user's preferences, and their values, is returned in an assoc list in system variable `*user-preferences*`. The rest of the Pathway Tools server system is then free to use these preferences as it desires. Since this session variable list gets reset for each and every Web page request that hits the system, and since Web page request threads are disjoint, the server in fact supports multiple simultaneous users, each with his or her own particular preferences vector.

The results are an optional login system, coupled with a preferences system, that act in predictable, understandable ways.

The Login and Preferences systems have been implemented to a version 1.5 level. They have been tested as working under both Firefox and Internet Explorer (IE) browsers under Windows Vista, and Firefox under Sun Solaris and Ubuntu Linux.

8.7.5 How to Debug an Installation

8.7.6 Default HTML directory not set properly

Symptoms: Pathway Tools Query Page does not come up by itself on /, you get the CYC page, but it doesn't have most of the pictures filled in.

This indicates that the default HTML directory, `cwest::*html-root-dir*`, has not been set properly. You need to set this to a good directory. Development uses `aic/pathway-tools/ptools/beta/install/htdocs` for the value for this global. Check to see what this global is bound to on the Lisp console on the server. If it's unbound, then that's a problem–open nav/www.lisp and make sure that this variable is being set. Often this problem can also be fixed by recompiling the cwest server and the nav system.

8.7.6.1 Cookies Issues

Symptoms: Can't log in (so it sticks), even though you get a login screen ...or, you were already logged in, but you can't log out.

- It is possible that the necessary internal helper `cookies.js` javascript file is not being pulled in from its directory. This happens when the html root directory has not been set properly.
- It is possible that cookies have been turned off on your browser. Please turn them on.

8.7.6.2 No folders shows in preferences.html

You probably are not pulling in include file `tabber.js`, which dynamically generates the folders on the fly after the page has been loaded. This can be caused by the html home directory not resolving correctly. See above. It can also be caused by permission problems on `tabber.js`. The other file to check is `dropdowntabs.js`, inside the `dropdowntabfiles` directory again in the html home directory.

8.7.6.3 Folders shown, but the tabs look strange

The tab appearances are governed by the `dropdowntabs.js` file in directory `dropdowntabfiles` under the html home directory; by css files in this directory, in particular `slidingdoorsM.css`; and by the gif picture media contained in directory `media` under directory `dropdowntabfiles`.

This can also occur if you're using a browser other than Firefox or Internet Explorer.

The tabs have been tightly adjusted so that their width fits on normal browsers. Linewrapping can occur in the following cases:

1. You have added some extra tabs, or changed the tab headings to longer words.

2. You have switched fonts to a wider font. In these cases, the dynamically-generated tabs may not all fit on one line, causing an ugly appearance. The best solution in this case is to widen the display of the parameters panel.

8.7.7 Summary

A Web Accounts system for the Pathway Tools Web server runs under both Firefox and Internet Explorer browsers, under at least PC and Unix platforms. The code allows each recognized logged-in user to have his or her own sessions variables, which are restored in `*user-preferences*` for each separate page view. And it still allows not-logged-in users to get to the pages they need. The system is presently useful for tracking users, and supporting preferences. In the future it will be used by Pathway Tools installation sites for much more.

8.8 Troubleshooting

If the Pathway Tools Web server malfunctions, check the following:

If the Web server is not functioning at all:

- Be sure you have defined the X-windows environment properly.
- Be sure not to exit from the Pathway/Genome Navigator window after Pathway Tools starts up.
- Be sure the `/tmp` directory is accessible on your computer; the Web server writes some temporary files there.
- If some PGDBs are not visible in the "Select a dataset" selector: Review the `-www-publish` argument discussed in the preceding section. It determines which PGDBs are accessible via the Web server.

Chapter 9

Troubleshooting

Pathway Tools detects the occurrence of an internal error. A dialog window pops up that contains an error message and asks you if you want to exit back to UNIX, or to reset the software. You can select either choice with the mouse.

9.1 Frequently Asked Questions

A list of Pathway Tools Frequently Asked Questions is available on the SRI Web site at `http://brg.ai.sri.com/ptools/faq.html`.

9.2 Reporting Problems

If you encounter problems with the software, or if you see errors in the scientific information in a PGDB, or if you have suggestions about the program, contact us by sending electronic mail to **ptools-support@ai.sri.com**.

When the Pathway Tools software detects an internal error, it writes a descriptive file called **error.tmp** in your home directory. That file helps SRI Technical Support track down the problem, so you will need to mail the file to **ptools-support@ai.sri.com**. Mail the file only once for every distinct bug that you encounter; that is, if you repeatedly encounter what appears to be the same bug, tell us about it only once.

Include a description of the problem, and be as thorough as possible. Include the names of relevant database objects, or a description of the operations you had performed using Pathway Tools before an error occurred. Also tell us what version of Pathway Tools you are running. The version number appears in the title bar of the Navigator main window.

More complete version information can be obtained by typing this command at the UNIX command line: **pathway-tools -id**.

Chapter 10

Guide to the Pathway Tools Schema

All Pathway/Genome Databases (PGDBs) used by the Pathway Tools software — including the EcoCyc and MetaCyc PGDBs — must conform to the schema (ontology) described herein. The objects and the relationships between these objects are utilized in this computerized description of metabolic and genomic information. Understanding the schema is essential for both users and developers of Pathway/Genome Databases who are using the Pathway Tools software.

In defining a conceptualization of knowledge for computer use, it is essential to employ precise definitions and distinctions. The fidelity of a computer representation determines the degree to which meaningful computations and analyses can be performed with the information in computer form. Unfortunately, many concepts in biology are not defined with the required precision. For example, a half dozen biologists could easily supply a half dozen conflicting definitions for the terms "gene," or "metabolic pathway." You may discover that our definitions of the class names and attribute names employed herein do not match the definitions that you prefer. We ask you to acknowledge that (a) biology is not yet well enough formalized that every biologist can expect to employ the same definitions, and (b) the definitions used in this document are much more thorough and precise (and therefore useful) than those offered in most biological databases.

Much of the discussion in this document refers to the EcoCyc database (DB), but the same schema is used for all other DBs managed by Pathway Tools. This schema may change in future versions of the software.

PGDBs are stored within a *frame knowledge representation system* (FRS). An FRS is a kind of object-oriented database system. The DB consists of a collection of *frames*, where each frame encodes information about a single object, such as an enzyme, a gene, or a biochemical pathway. For a more precise discussion of FRSs, see [5].

Instance frames describe specific biological objects, such as a specific gene or a specific metabolic pathway. *Class frames* describe general types of biological objects, such as the

class of all genes. Each frame contains one or more *slots*. A slot describes an attribute or a property of the object that the frame represents. Each slot makes sense for (is valid in) a particular set of classes. For example, the slot **EC-Number** makes sense only for frames in the Reactions class, whereas the slot **Synonyms** is valid in all classes.

The current Pathway Tools ontology contains several hundred classes arranged in a taxonomic hierarchy. shows some of the major classes, and their relationships. An arrow that points from class A to class B indicates that B is a child of A, and therefore that A is a more general class that subsumes B. For example, the class **Proteins** can be subdivided into the subclasses **Polypeptides** (monomers) and **Protein-Complexes** (multimers). Subclasses inherit slots from their parents, for example, **Polypeptides** inherits all slots defined in **Proteins**, and some additional slots are also defined in **Polypeptides**. The classes in this figure whose names are shown in bold are described in more detail in the remainder of this appendix.

The top-level classes in Figure A-1 describe physical entities and processes. More specifically, **Chemicals** describes atoms and complete chemical compounds, **Polymer-Segments** describe regions within polymers such as proteins and DNA, and **Organisms** describes the biological organism modeled within a PGDB. The class **Chemicals** is subdivided into small-molecular weight compounds (class **Compounds**) and atoms (not shown), and into macromolecules (**Macromolecules**). **Macromolecules** include subclasses such as DNA and **RNA; DNA** includes subclasses that describe different types of replicons such as chromosomes and plasmids. The different subclasses of **Polymer-Segments** include different types of DNA sites such as transcription start sites and terminators, and longer regions such as genes. On the process side, **Generalized-Reactions** describe both individual biochemical reactions, and biochemical pathways. The class **Enzymatic-Reactions** describes information specific to the pairing of an enzyme with a reaction that the enzyme catalyzes, such as its activators, inhibitors, and cofactors.

The sections that follow describe the major classes and their slots in more detail. For additional discussions regarding the representations employed in the Pathway Tools ontology, see publications listed at `http://BioCyc.org/publications.shtml` in the section entitled "Publications on the Pathway Tools Ontology."

10.1 Slots Valid in Multiple Classes

In this discussion of slots that are used in several different EcoCyc classes, slot names are sometimes capitalized, and sometimes in lowercase. In fact, all slot names are all uppercase in the database itself. However, for the purpose of writing Lisp queries to Pathway/Genome Databases, all slot names can be written as lowercase because the Lisp interpreter translates all symbol names to uppercase (except for symbol names written between vertical bars). Hyphens separate multiple words in a slot name.

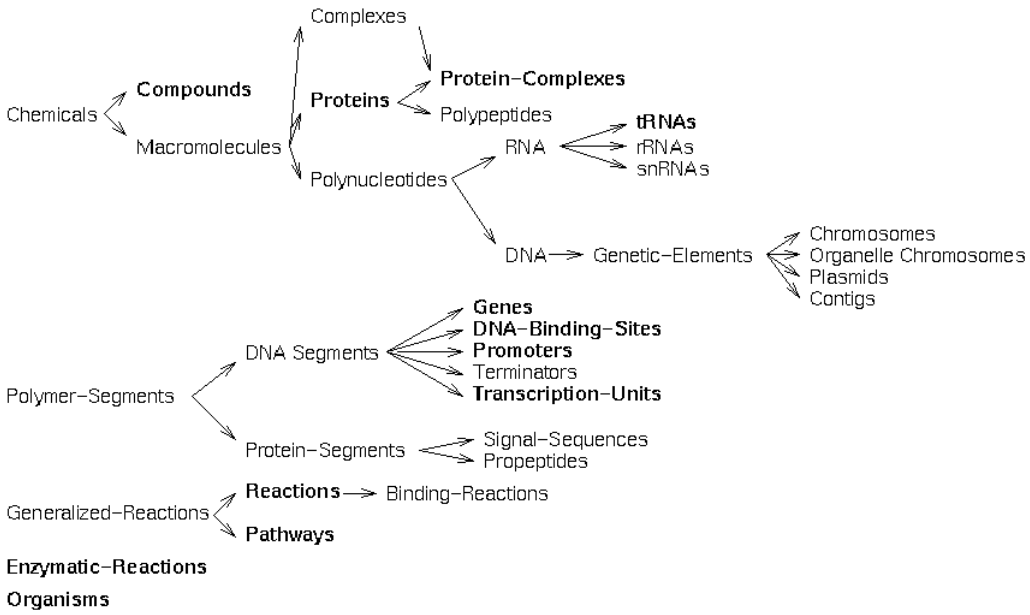


Figure 10.1: Some of the Main Classes Defined in the Schemas of Pathway/Genome Databases. The arrows denote the specialization–generalization relationship; for example, this figure indicates that all polypeptides are proteins, because class Proteins is the parent (superclass) of class Polypeptides.

10.1.1 Common-Name

This slot defines the primary name by which an object is known to scientists — a widely used and familiar name (in some cases arbitrary choices must be made). This field can have only one value; that value must be a string.

10.1.2 Synonyms

This field defines one or more secondary names for an object — names that a scientist might attempt to use to retrieve the object. These names may be out of date or ambiguous, but are used to facilitate retrieval — the Synonyms should include any name that you might use to try to retrieve an object. In a sense, "Synonyms" is misleading because the names listed in this slot may not be exactly synonymous with the preferred name of the object.

10.1.3 Abbrev-Name

This slot stores an abbreviated name for an object. It is used in some displays.

10.1.4 Names

Values of this slot are computed by combining the values of all other name-related slots for this frame: slots Common-Name, Systematic Name, Synonyms, Abbrev-Name, N-Name, N-1-Name, and N+1-Name.

10.1.5 Comment

The Comment slot stores a general comment about the object that contains the slot. The comment should always be enclosed in double quotes.

10.1.6 Citations

This slot lists general citations pertaining to the object containing the slot. Citations may or may not have evidence codes attached to them. Each value of the slot is a string of the form [reference-ID] or [reference-id:evidence-code:timestamp:curator:probability:with], where:

1. reference-ID is a PubMed unique identifier or the identifier of a Publications object (without the leading "PUB-").
2. evidence-code is the object identifier of some class belonging to the Evidence class, e.g. EV-EXP.
3. timestamp is a lisp universal time (not human readable) corresponding to the time the evidence code was assigned.
4. curator is the username of the curator who assigned the evidence code.
5. probability is a number between 0 and 1 describing the probability that the evidence is correct, where available.
6. with is a free text string that modifies the evidence-code when the citation annotates a GO term. This is the "with" field described in GO documentation.

Any of the above components may be omitted, but it is meaningless to supply a timestamp, curator or probability if the evidence-code is omitted. Trailing colons should be omitted, but if a value contains an evidence-code with no accompanying citation, the leading colon must be present. The square brackets are optional.

Examples:

123456 – a PubMed or Medline reference

SMITH95 – a non-PubMed reference

123456:EV-IDA – an evidence code with associated PubMed reference

:EV-HINF – an evidence code with no associated reference

123456:EV-IGI:9876543:paley – a time- and user-stamped evidence code with associated reference

10.2 Class Compounds

The Class Compounds describe small-molecular-weight chemical compounds — typically, compounds that are substrates of metabolic reactions or compounds that activate or inhibit metabolic enzymes.

10.2.1 Appears-In-Left-Side-Of, Appears-In-Right-Side-Of

Lists the one or more reactions in which this compound occurs as a reactant or product, respectively.

10.2.2 Aromatic-Rings

Each value in this slot is a list of atom numbers; that list of atoms constitutes a single aromatic ring. For example, the list might specify that atoms 1, 2, 5, 6, 10, 20 are in one aromatic ring (see slot Structure-Atoms).

10.2.3 Atom-Charges

This slot lists the charges of specific atoms within the compound. Each value of the slot is a list of the form (A C) where A is the index of an atom in slot Structure-Atoms, and C is the charge of that atom.

10.2.4 Charge

Lists the chemical charge for this compound.

10.2.5 Chemical-Formula

Lists the empirical formula for this compound. Each value of this slot is a list of the form (ATOM COUNT) where ATOM is the ID of a frame for the corresponding chemical element, and COUNT is the number of occurrences of that atom in this compound. For example, molecular oxygen, O_2, would be represented as (O 2) with a space between the letter O and the number 2. The value of this slot is computed automatically.

10.2.6 Display-Coords-2D

This slot lists coordinates for the display of the chemical structure of this compound in two dimensions. The values of this slot correspond one-to-one to the values of slot Structure-Atoms. Each value of this slot is a list of the form (X Y) and consists of the X-Y display coordinate of the corresponding atom in Structure-Atoms. The coordinates are real numbers with no specified minimum or maximum values. They are rescaled at display time.

10.2.7 Gibbs-0

Provides the standard Gibbs free energy of formation of the compound. The values are in units of kilocalories/mol, assuming the common state in aqueous solution at pH=7 and T=25C.

10.2.8 Molecular-Weight

Provides the molecular weight of this compound in daltons.

10.2.9 N-Name, N-1-Name, N+1-Name

These slots are used when displaying the names of polymeric compounds in pathways that increase or decrease the lengths of the polymers. The names indicate a polymer of length N, length N-1, and length N+1. As an example, see the compound at **http://biocyc.org/META/NEW-IMAGE?type=COMPOUND&object=|Folatepolyglutamate-n|**.

10.2.10 Regulates

For compounds that have regulatory activity (e.g. as activators or inhibitors of enzymes), this slot points to the Regulation frames that describe the regulation and link to the regulated entity.

10.2.11 Smiles

Provides a representation of the chemical structure of this compound using the SMILES chemical encoding system. Note that the value of this slot is computed using an attached procedure; do not attempt to store a value into this slot.

10.2.12 Structure-Atoms

This slot is one of several that are used to encode the chemical structure of a compound. This slot lists all the distinct atoms in the compound, with multiple entries for atoms of the same element that occur more than once. For example, water could be described as the list (H H O). The atoms are listed in no special order. However, other slots refer to the atoms in the compound according to their position in this list; for example, the first hydrogen is atom 0, and the oxygen is atom 2.

10.2.13 Structure-Bonds

This slot describes the chemical bonds within a compound. Each bond is encoded as a list of the form (A1 A2 B-TYPE) where A1 is the index in slot Structure-Atoms of the first atom in the bond, A1 is the index of the second atom in the bond, and B-TYPE encodes the type of the chemical bond. Valid bond types are the numbers 1, 2, and 3 for single, double, and triple bonds, and the symbol: AROMATIC for aromatic bonds. For example, to specify that a double bond exists between the first and fifth atoms, use the list (0 4 2).

10.3 Class DNA-Binding-Sites

This class describes DNA regions that are binding sites for transcription factors.

10.3.1 Abs-Center-Pos

This slot defines the position on the replicon of the center of this binding site.

10.3.2 Involved-in-Regulation

This slot links the binding site to the Regulation frame describing the regulatory interaction in which this binding site participates.

10.3.3 Site-Length

This slot defines the extent of a binding site in base pairs. If a value for this slot is omitted, the site length will be computed based on the DNA-Footprint-Size of the binding protein. Thus, a value for this slot should only be supplied here if the site length for a particular transcription factor is not consistent across all its sites.

10.4 Class Enzymatic Reactions

Frames in the class Enzymatic-Reactions describe attributes of an enzyme with respect to a particular reaction. For reactions that are catalyzed by more than one enzyme, or for enzymes that catalyze more than one reaction, multiple Enzymatic-Reactions frames are created, one for each enzyme/reaction pair. For example, Enzymatic-Reactions frames can represent the fact that two enzymes that catalyze the same reaction may be controlled by different activators and inhibitors. See [10] for more details.

10.4.1 Enzyme

This slot lists the enzyme whose activity is described in this frame. More specifically, the value of this slot is the key of a frame from the class Protein-Complex or Polypeptide.

10.4.2 Required-Protein-Complex

Some enzymes catalyze only a particular reaction when they are components of a larger protein complex. For such an enzyme, this slot identifies the particular protein complex of which the enzyme must be a component.

10.4.3 Reaction

The value of this slot is the key of a frame from the Reaction class — the second half of the enzyme/reaction pair that the current frame describes. In fact, this slot can have multiple values, which encode the multiple reactions that one catalytic site of an enzyme catalyzes.

10.4.4 Regulated-By

The values of this slot are members of the Regulation class, describing activator or inhibitor compounds for this enzymatic reaction.

10.4.5 Cofactors, Prosthetic-Groups

The literature uses terms such as coenzyme, cofactor, and prosthetic group in an extremely inconsistent fashion. In version 2.8 of EcoCyc (March 1996), we adopted the usage of terms that were developed by Evgeni Selkov (Gene Selkov) for use in the Enzymes and Metabolic Pathways (EMP) database.

"Class Reactions" in Section defines the substrates of a reaction as the union of its reactants and its products. After Selkov, we define a coenzyme to be a specialization of substrates, namely, substrates with a relatively stable, conserved moiety, whose main function is group transfer among different enzymes and pathways. Example: NAD. EcoCyc does not define a special slot for coenzymes.

Also after Selkov, we define cofactors and prosthetic groups to be compounds that are required for an enzyme to catalyze a reaction, but that are unchanged by the reaction. Thus, cofactors and prosthetic groups are (loosely speaking) activators of an enzyme in the sense that the enzyme is not active when these compounds are absent. However, cofactors and prosthetic groups have an infinite "activation degree", thus distinguishing them from those compounds that are activators described by Regulation frames listed in the Regulated-By slot; when those activators are missing, the enzyme still functions, but at a lower rate.

The distinction between cofactors and prosthetic groups is that prosthetic groups are covalently or tightly bound to an enzyme, whereas cofactors are not. The corresponding slot names are **Cofactors** and **Prosthetic-Groups**.

A slot called **Cofactors-Or-Prosthetic-Groups** identifies compounds whose binding affinity to the enzyme is unclear.

10.4.6 Alternative-Substrates, Alternative-Cofactors

These slots record variability in the substrates and cofactors that have been observed for this enzymatic reaction. If, for example, the literature indicates that Mn+2 can substitute for Mg+2 as a cofactor in this reaction, we would list the following as a value for the Alternative-Cofactors slot: (Mg+2 Mn+2).

The Alternative-Substrates slot describes the substrate specificity of an enzymatic reaction. We use the Alternative-Substrates when the complete equation is not known for an

alternative reaction, or when the alternative reaction is not physiologically important, or is not a member of a known pathway. Each value of the **Alternative-Substrates** slot is a list whose first member is a compound that was specified as a substrate; the remaining elements of the list are compounds that can serve as alternative substrates for the first compound.

Each value of the **Alternative-Cofactors** slot is a list whose first member is a compound that was specified as a cofactor or prosthetic group; the remaining elements of the list are compounds that can serve as alternatives for the first compound.

An annotation on a value for either of these slots is assumed to apply to each alternative substrate/cofactor listed in the value. If an annotation is intended to apply to only one such compound (or other subset), two (or more) values should be used instead, where the substrate is repeated as the first element of each value, and the alternative compounds are divided among the values according to the applicability of the annotations.

10.4.7 Reaction-Direction

This slot specifies the directionality of a reaction. The slot is particularly important to fill for reactions that are not part of a pathway, because for such reactions, the direction cannot be determined automatically, whereas for reactions within a pathway, the direction can be inferred from the pathway context. This slot aids the user and software in inferring the direction in which the reaction typically occurs in physiological settings, relative to the direction in which the reaction is stored in the database. Possible values of this slot are

:REVERSIBLE: The reaction occurs in both directions in physiological settings.

:PHYSIOL-LEFT-TO-RIGHT, :PHYSIOL-RIGHT-TO-LEFT: The reaction occurs in the specified direction in physiological settings, because of several possible factors including the energetics of the reaction, local concentrations of reactants and products, and the regulation of the enzyme or its expression.

:IRREVERSIBLE-LEFT-TO-RIGHT, :IRREVERSIBLE-RIGHT-TO-LEFT: For all practical purposes, the reaction occurs only in the specified direction in physiological settings, because of chemical properties of the reaction.

10.5 Class Genes

Each frame in the class **Genes** describes a single gene, meaning a region of DNA that defines a coding region for one or more gene products. Multiple gene products may be produced because of modification of an RNA or protein.

10.5.1 Left-End-Position, Right-End-Position

These slots encode the position of the left and right ends of the gene on the chromosome or plasmid on which the gene resides. "Left" means the end of the gene toward the coordinate-system origin (0). Therefore, the **Left-End-Position** is always less than the **Right-End-Position**.

In the EcoCyc DB, the values of this slot were taken directly from GenBank entry U00096 submitted by the Blattner laboratory.

10.5.2 Centisome-Position

This slot lists the map position of this gene on the chromosome in centisome units (percentage length of the chromosome). The centisome-position values are computed automatically by Pathway Tools from the **Left-End-Position** slot. The value is a number between 0 and 100, inclusive.

10.5.3 Transcription-Direction

This slot specifies the direction along the chromosome in which this gene is transcribed; allowable values are "+" and "-".

10.5.4 Product

This slot holds the ID of a polypeptide or tRNA frame, which is the product of this gene. This slot may contain multiple values for two possible reasons: a given gene might be translated from more than one start codon, giving rise to products of different lengths; the product of the gene may undergo chemical modification. In the latter case, the gene lists all modified forms of the protein in its **Product** slot.

10.5.5 Interrupted?

If True, indicates that the specified gene is interrupted, that is, has a premature stop codon.

10.6 Class Organisms

The Organisms class is used in different ways in organism-specific PGDBs versus in multiorganism PGDBs such as MetaCyc. The next paragraph discusses what is common to both types of PGDBs. Subsequent paragraphs describe the differences.

In all PGDBs, subclasses of Organisms define biological taxa, at all possible taxonomic levels. Class-subclass relationships between subclasses of Organisms describe their taxonomic relationships, since, for example, the class Bacteria includes as a subclass the class Alphaproteobacteria. Generally, most of the taxonomic groups under Organisms correspond to entries from the NCBI Taxonomy database (which is stored in its entirety in a separate Ocelot KB). But in addition, to taxa from NCBI Taxonomy, a PGDB can contain subclasses for additional taxa that are not present in the NCBI Taxonomy.

Organism-specific PGDBs: In an organism-specific PGDB, the only frames that exist as children of Organisms are those frames needed to describe the taxonomic lineage of the organism described by the PGDB. An organism-specific PGDB contains a single instance frame that describes information about the PGDB itself. A parent class P of that instance must exist to describe the lowest taxonomic group defined for the organism. Additional parent classes exist as parents of P and children of Organisms that describe the other known taxonomic parents of P. No other children of Organisms exist in the PGDB.

Multiorganism PGDBs such as MetaCyc: Multiorganism PGDBs contain no instances of class Organisms, but only subclasses of this class. Those subclasses define each of the different organisms for which MetaCyc (for example) defines pathways and enzymes. For economy of storage, only those taxa (and their parent taxa) actually referenced in the PGDB are stored in the PGDB, so that only a subset of the NCBI Taxonomy is replicated in the PGDB. There is only one instance of Multi-Organism-Groupings that describes the properties of the PGDB.

10.6.1 PGDB-Authors

A list of the names of the authors of this DB. The names are displayed on a summary page for this organism. It is appropriate to suffix each name with the author's institution, for example, "John Doe, University of New Jersey". Use one slot value per author.

10.6.2 PGDB-Copyright

The contents of this slot should be a copyright notice for this database, if one is desired. The copyright notice should preferably fit in one line because it will be printed at the bottom of every Web page served for this organism database by the Pathway Tools Web server. Example: "Copyright 1999 University of New Jersey."

10.6.3 PGDB-Footer-Citation

The value of this slot should be a single literature citation, in the form of a string, such as "Bioinformatics 12:155 2002". This citation, if present, is printed at the bottom of each Web page served for this organism, within the following text: "Please cite XYZCyc as **CITATION** in publications resulting from its use."

10.6.4 PGDB-Home-Page

The URL of a Web page describing this PGDB. Authors can use this page to provide more background information about the PGDB.

10.6.5 PGDB-Name

The name of the database for this organism, when the database name is to be printed somewhere by Pathway Tools. Examples: "EcoCyc," "PlasmoCyc." The suffix "Cyc" is not required.

10.6.6 PGDB-Unique-ID

An integer unique ID for this PGDB that differentiates it from other PGDBs. This ID is used to build unique IDs for frames that are newly created in this PGDB so that (a) when frames are copied among PGDBs, we know what PGDB the frame originated in, and (b) we can ensure that two frames in two different PGDBs that have the same ID do in fact refer to the same biological entity. The MetaCyc DB has a PGDB-Unique-ID of NIL; all other DBs should have a non-NIL value for this slot.

10.6.7 Strain-Name

Specifies the strain name for the organism.

10.6.8 Contact-Email

The email address of a person who serves at the primary contact for this PGDB, such as to receive questions or bug reports from users of the PGDB.

10.6.9 Genome

A list of all replicons (chromosomes and plasmids) in the genome of the organism.

10.7 Class Pathways

Frames in class Pathways encode metabolic and signaling pathways.

10.7.1 Pathway-Interactions

This slot holds a comment that describes interactions between this pathway and other biochemical pathways, such as those pathways that supply an important precursor.

10.7.2 Predecessors

This slot describes the linked reactions that compose the current pathway. Since pathways have a variety of topologies — from linear to circular to tree structured — pathways cannot be represented as simple sequences of reactions. A pathway is a list of reaction/predecessor pairs. That is, each value of this slot is of the form (reaction-ID pred-ID*) where reaction-ID is the key of a reaction in the pathway, and each pred-ID is the key of a reaction in the pathway that directly precedes the reaction-ID reaction. For example, to represent the combined pathway for tyrosine and phenylalanine synthesis (see), this predecessor list might be used:

(chorismatemut-rxn)

(prephenatedehydrat-rxn chorismatemut-rxn)

(pheaminotrans-rxn prephenatedehydrat-rxn)

(prephenatedehydrog-rxn chorismatemut-rxn)

(tyraminotrans-rxn prephenatedehydrog-rxn)

The first reaction in the pathway has no predecessor, so there is only one key within the first value. Since prephenate is a branch point in the pathway, two reactions in the pathway list the reaction that synthesizes prephenate as a predecessor.

Alternatively, any value for this slot can be another pathway key, which means that the current pathway inherits all the predecessor values of the indicated pathway. In other words, the current pathway is a superpathway of the indicated pathway. Thus, a more

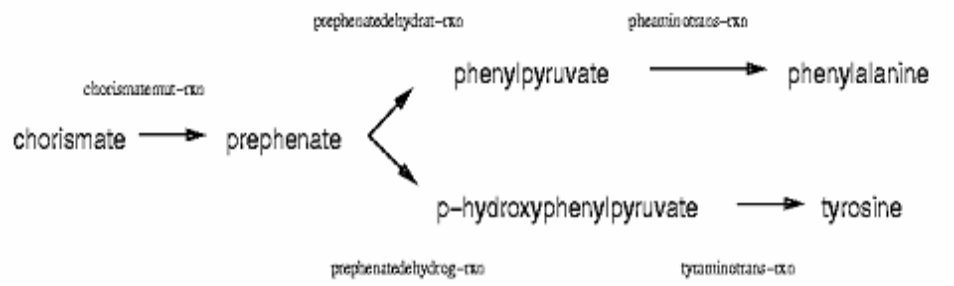


Figure 10.2: Pathway showing the Combined Synthesis of the Amino Acids tyrosine and phenylalanine from chorismate. Each reaction is labeled by its key in the EcoCyc DB. For example, the key for the reaction that converts prephenate to phenylpyruvate is prephenatedehydrog-rxn.

compact way of representing the combined pathway for tyrosine and phenylalanine synthesis would be to use the following predecessor list:

predecessors: tyrsyn, phesyn

In actuality, this latter representation is the preferred one and is required in order for the combined pathway to be determined to be a superpathway of either the tyrosine or phenylalanine (which of course should be the case). The advantage of specifying the predecessor list in this way (aside from being more compact and easy to read) is that if the subpathway is ever modified, the changes will automatically propagate to the superpathway.

10.7.3 Reaction-List

This slot lists all reactions in the current pathway, in no particular order.

10.7.4 Hypothetical-Reactions

A list of reactions in this pathway that are considered hypothetical, probably because presence of the enzyme has not been demonstrated.

10.7.5 Assume-Unique-Enzymes

By default it is assumed that all enzymes that can catalyze a reaction will do so in each pathway in which the reaction occurs. That default assumption is encoded by the default

value of **FALSE** for this slot; when you want to assume that only one enzyme exists in the DB to catalyze every reaction in this pathway, this slot should be given the value **TRUE**.

This slot can be used for consistency-checking purposes, that is, in a pathway for which this slot is **TRUE**, there should not be any reactions that are catalyzed by more than one reaction.

10.7.6 Enzyme-Use

By default it is assumed that all enzymes that can catalyze a reaction will do so in each pathway in which the reaction occurs. This slot is used in the case that this assumption does not hold, that is, if a reaction is catalyzed in a particular pathway by only a subset (or none) of the possible enzymes that are known to catalyze that reaction. Therefore, this slot can be used only when the value of the **assume-unique-enzymes** slot is **FALSE** (because multiple enzymes catalyze some step in the pathway).

The form of a value for the slot is **(reaction-ID enzymatic-reaction-ID-1... enzymatic-reaction-ID-n)**. That is, each value specifies a reaction, and specifies the one or more enzymatic reactions that catalyze that reaction in this pathway. If no enzymatic reactions are specified, then none of the enzymes that are known to catalyze the reaction do so in this pathway.

For example, under aerobic conditions the oxidation of succinate to fumarate is catalyzed by succinate dehydrogenase in the forward direction, and, under anaerobic conditions, by fumarate reductase in the reverse direction. The TCA cycle is active only in aerobic conditions, so only succinate dehydrogenase is used in this pathway. This fact would be recorded as follows:

enzyme-use: (succ-fum-oxred-rxn succinate-oxn-enzrxn)

10.7.7 Enzymes-Not-Used

Proteins or protein-RNA complexes listed in this slot are those which would otherwise have been inferred to take part in the pathway or reaction, but which in reality do not. The protein may catalyze a reaction of the pathway in other circumstances, but not as part of the pathway (e.g. it may be not be in the same cellular compartment as the other components of the pathway, or it may not be expressed in situations when the pathway is active.).

10.7.8 Primaries

When drawing a pathway, the Navigator software usually computes automatically which compounds are primaries (mains) and which compounds are secondaries (sides). Occasionally, the heuristics used are not sufficient to make the correct distinction, in which case you can specify primary compounds explicitly. This slot can contain the list of primary reactants, primary products, or both for a particular reaction in the pathway. Each value for this slot is of the form **(reaction-ID (primary-reactant-ID-1 ... primary-reactant-ID-n) (primary-product-ID-1 ... primary-product-ID-n))**, where an empty list in either the reactant or product position means that that information is not supplied and should be computed.

For example, in the purine synthesis pathway, we want to specify that the primary product for the final reaction in the pathway should be AMP and not fumarate. The primary reactants are still computed. The corresponding slot value would be

primaries: (ampsyn-rxn () (amp))

10.7.9 Species

This slot is used only in pathway frames in the MetaCyc DB, in which case the slot identifies the one or more species in which this pathway is known to occur experimentally.

10.7.10 Disable Display

When the value is true, this slot disables display of the pathway drawing for a pathway.

10.7.11 Super-Pathways

This slot lists direct super-pathways of a pathway.

10.7.12 Sub-Pathways

This slot is the inverse of the Super-Pathways slot. It lists all the direct subpathways of a pathway.

10.7.13 Pathway-Links

This slot indicates linkages among pathways in pathway drawings. Each value of this slot is a list of the form (**cpd other-pwy***). The Navigator draws an arrow from the specified compound pointing to the names of the specified pathways, to note that the compound is also a substrate in those other pathways. If no other pathways are specified, then links are drawn to and from all other pathways that the compound is in (i.e., if the compound is produced by the current pathway, then links are drawn to all other pathways that consume it, and vice versa).

10.7.14 Polymerization-Links

This slot controls drawing of polymerization relationships within a pathway. Each value of this slot is of the form **(cpd-class product-rxn reactant-rxn)**. When both reactions are non-nil, an identity link is created between the polymer compound class cpd-class, a product of product-rxn, and the same compound class as a reactant of reactant-rxn. The **PRODUCT-NAME-SLOT** and **REACTANT-NAME-SLOT** annotations specify which slot should be used to derive the compound label in product-rxn and reactant-rxn above, respectively, if one or both are omitted, **COMMON-NAME** is assumed. Either reaction above may be nil; in this case, no identity link is created. This form is used solely in conjunction with one of the name-slot annotations to specify a name-slot other than **COMMON-NAME** for a polymer compound class in a reaction of the pathway.

10.7.15 Class-Instance-Links

Each value of this slot is a reaction in the pathway. Two annotations (in addition to the usual possibilities) are available on this slot: **REACTANT-INSTANCES** and **PRODUCT-INSTANCES**, whose values are compounds. If one of the reactants of the slot-value reaction is a class C and the **REACTANT-INSTANCES** are instances of C, then the instances are drawn as part of the pathway, with identity links to the class. The **PRODUCT-INSTANCES** are treated similarly.

10.7.16 Layout-Advice

Each value of this slot is a dotted pair of the form (**advice-keyword . advice**, and represents some piece of advice to the automatic pathway layout code. Currently supported advice keywords are

1. **`:CYCLE-TOP-CPD`:** The advice is a compound key. In pathways containing a cycle,

the cycle will be rotated so that the specified compound is positioned at twelve o'clock.

2. **:REVERSIBLE-RXNS:** The advice is a list of reactions that should be drawn as reversible, even when the pathway is being drawn to show pathway flow (rather than true reversibility).

3. **:CASCADE-RXN-ORDERING:** The advice is a list of reactions that form a partial order for reactions in a cascade pathway (i.e., the 2-component signaling pathways).

10.8 Class Polypeptides

Frames of class Polypeptides are monomers consisting of a single polypeptide chain.

10.8.1 Gene

This slot contains a value that identifies the gene that encodes the polypeptide.When a polypeptide exists in two forms, modified and unmodified, both forms contain the same value in their **Gene** slots.

10.8.2 Features

This slot links the polypeptide to any protein features that have been defined for it. When a polypeptide exists in multiple forms, each form will link to the same set of features.

10.8.3 Splice-Form-Introns

This slot lists any introns that were spliced out of the gene in order to generate this polypeptide. Values of this slot are of the form (start-bp end-bp).

10.9 Class Promoters

Frames in this class define transcription start sites.

10.9.1 Absolute-Plus-1-Pos

The absolute base pair position of the transcription start site on the DNA strand.

10.9.2 Binds-Sigma-Factor

This slot links to the one or more sigma factors that can bind to a promoter, thereby initiating transcription.

10.9.3 Component-Of

This slot links to the transcription-unit(s) to which the promoter belongs.

10.9.4 Minus-35-Left, Minus-35-Right, Minus-10-Left, Minus-10-Right

These slots list chromosomal coordinates of the left and right ends of the -35 and -10 boxes associated with the promoter.

10.10 Class Complexes

The class Complexes is subdivided into several subclasses.

Frames of class Protein-Complexes are multimeric proteins composed of multiple subunits. The subunits of a protein complex may themselves be protein complexes, although eventually the subunits must bottom out as polypeptides.

Frames of class Protein-Small-Molecule-Complexes are the result of a protein (either a polypeptide or protein complex) binding with a small molecule ligand.

Frames of class Protein-RNA-Complexes are the result of one or more proteins forming a complex with one or more RNA molecules.

Frames of class Protein-DNA-Complexes are the result of a protein (or a complex that includes a protein, e.g., a protein complex or a complex of a protein and small-molecule ligand) binding with a segment of DNA. Examples of this include the binding of a transcription factor to a DNA binding site, or an RNA polymerase molecule binding to a promoter region of DNA in order to initiate transcription.

10.10.1 Components

This slot lists the subunits of a complex. The nature of the subunits depends on the type of complex. For a protein complex, each subunit is either a polypeptide or a protein complex; therefore, each slot value is the key of a polypeptide frame or a protein-complex frame. For other types of complexes, subunits may also include small-molecules, RNA molecules, or regions of DNA.

The coefficient of each component of the protein complex is listed as an annotation of the component value under the label **Coefficient** [5].

10.11 Class Proteins

The class of all proteins is divided into two subclasses: protein complexes and polypeptides. A polypeptide is a single amino acid chain produced from a single gene. A protein complex is a multimeric aggregation of more than one polypeptide subunit. A protein complex may in some cases have another protein complex as a component. Many of the slots that are applicable to Proteins are also applicable to members of the RNA class.

10.11.1 Component-Of

This slot lists the complex(es) that this protein is a component of, if any, including protein complexes, protein-small-molecule complexes, protein-RNA complexes, and so on.

10.11.2 DNA-Footprint-Size

For proteins that bind to DNA, the number of base pairs on the DNA strand that the binding protein covers.

10.11.3 GO-Terms

Values of this slot are the Gene Ontology terms to which this object is annotated. Each value should be annotated with citations, including evidence codes.

10.11.4 Locations

This slot describes the one or more cellular locations in which this protein is found. It's values are members of the CCO (Cell Component Ontology) class.

10.11.5 Modified-Form

This slot points from the unmodified form of a protein to one or more chemically modified forms of that protein. For example, the slot might point from the unmodified form of a polypeptide (or a protein complex) to a phosphorylated form of that polypeptide (or protein complex).

10.11.6 Molecular-Weight-KD

This computed slot lists the known molecular weight(s) of a macromolecule by taking the union of the slots Molecular-Weight-Seq and Molecular-Weight-Exp. Units: kilodaltons.

10.11.7 Molecular-Weight-Seq

This slot lists the molecular weight of the protein complex or polypeptide, as derived from sequence data. Units: kilodaltons.

10.11.8 Molecular-Weight-Exp

This slot lists the molecular weight of the protein complex or polypeptide, derived experimentally. Multiple values of this slot correspond to multiple experimental observations. Units: kilodaltons.

10.11.9 pI

This slot lists the pI of the polypeptide.

10.11.10 Regulates

For proteins that have regulatory activity (e.g. as transcription factors), this slot points to the Regulation frames that describe the regulation and link to the regulated entity.

10.11.11 Species

This slot is used in proteins only in the MetaCyc DB, in which case it identifies the species in which the current protein is found.

10.11.12 Unmodified-Form

This slot points from a chemically modified form of some protein, to the native unmodified form of that protein (e.g., from a phosphorylated form to the unphosphorylated form).

10.12 Class Protein-Features

This class describes sites of interest (such as binding sites, modification sites, cleavage sites) on a polypeptide. Instances of this class define a region of interest on a polypeptide, plus, optionally, one or more states associated with the region. Different subclasses are used to specify single amino acid sites, linear regions, and regions involving noncontiguous segments of an amino-acid chain. For example, an instance F of this class could define an amino acid residue that can be phosphorylated, plus the fact that this residue can take on two possible states: PHOSPHORYLATED and UNPHOSPHORYLATED.

The feature instance itself does not describe the state of a particular protein. Instead, we would represent the phosphorylated and unphosphorylated forms of a protein by creating two instances of class Polypeptides. Both of those instances would link to the same feature F via the FEATURES slot. However, in the two proteins, F would be annotated differently to indicate the state of that feature. One protein would use an annotation label STATE with the value PHOSPHORYLATED to denote that the residue is phosphorylated, while the other would use the same annotation label STATE with the value UNPHOSPHORYLATED.

10.12.1 Attached-Group

For a binding feature, this slot lists the entity that binds to the protein feature — it can be either an instance of Chemicals or another Protein-Feature (e.g., in the case of crosslinks forming between two sites on the same or different polypeptide).

10.12.2 Feature-Of

This slot points to the polypeptide frames with which this feature is associated (there could be more than one such frame, if all are different forms of the same protein, e.g., a modified and an unmodified form).

10.12.3 Left-End-Position

For a feature that consists of a contiguous linear stretch of amino acids, this slot encodes the residue number of the leftmost amino acid, with number 1 referring to the N-terminal amino acid.

10.12.4 Possible-Feature-States

For a given feature class, this slot describes the possible states available to instances of the class. For example, a feature that represents a binding site can have either a bound or unbound state. The list of possible states is stored at the class level as values for this slot. A particular instance F of the class (a specific feature of a specific protein) can then be labeled with this state information using the STATE annotation when F appears in the FEATURES slot of the protein. For example, two forms of the same protein would link to the same feature F, but one form P1 would have the feature annotated label STATE and value BOUND, whereas the other form P2 would use the label STATE and value UNBOUND.

10.12.5 Residue-Number

For a feature that consists of a single amino acid or some number of noncontiguous amino acids, this slot contains the numeric index or indices of the amino acid residue or residues that make up this site. Number 1 corresponds to the N-terminal amino acid.

10.12.6 Right-End-Position

For a feature that consists of a contiguous linear stretch of amino acids, this slot encodes the residue number of the rightmost amino acid, relative to the start of the protein.

10.13 Class Reactions

Frames within the Reaction class describe properties of a biochemical reaction independent of any enzyme or enzymes that catalyze that reaction. A reaction is a biochemical transformation that interconverts two sets of chemical compounds (which includes small metabolites, proteins, and DNA regions), and may translocate compounds from one cellular compartment to another. Most reactions are written in a conventional direction that has been assigned by the Enzyme Nomenclature Commission, but that direction may or may not be the predominate physiological direction of the reaction. Reaction substrates can include small-molecular-weight compounds (for metabolic reactions), proteins (such as in signaling pathways), and DNA sites (such as for reactions involving binding of transcription factors to DNA).

Reactions are organized into two parallel ontologies. Most reaction frames will have one or more parents in both ontologies. The first classifies reactions by the nature of their substrates, for example, small-molecule reactions are reactions in which all substrates are small molecules, whereas protein reactions are reactions in which at least one substrate is a protein. The second ontology classifies reactions by conversion type. For example, chemical reactions are those in which a chemical transformation takes place, transport reactions are those in which a substrate is transported from one compartment to another (some reactions may be both transport reactions and chemical reactions if the substrate is chemically altered during transport), and binding reactions are those in which substrates weakly bind to each other to form a complex.

Two novel features of our conceptualization with respect to previous metabolic databases are to separate reactions from the enzymes that catalyze them, and to use the EC numbers defined by the International Union of Biochemistry and Molecular Biology (IUBMB) to uniquely identify reactions, not enzymes. (In database terms, the EC number is a key for the Reaction class.) The reason for this separation is that the *catalyzes* relationship between reactions and enzymes is many-to-many: a given enzyme might catalyze more than one reaction, and the same reaction might be catalyzed by more than one enzyme. Frames in the class Enzymatic-Reaction describe the association between an enzyme and a reaction. The entire EC taxonomy can be found under the Chemical-Reactions class.

You should always write transport reactions in the predominate direction in which the reaction occurs. Transport reactions are encoded by labeling substrates with their cellular compartment. For example, if a given substrate is transported from the periplasm to the cytoplasm, it would be labeled with "periplasm" as its compartment as a reactant, and with "cytoplasm" as its compartment as a product. The default compartment is the cytoplasm, so the cytoplasm label may be omitted. These labels are implemented as annotations in Ocelot.

10.13.1 EC-Number

This slot holds the EC (Enzyme Commission) number associated with the current reaction, if such a number has been assigned by the IUBMB. This slot is single valued.

10.13.2 Official-EC?

The value of this slot is NO if the current reaction either was not defined at all by the Enzyme Commission, or if the current equation stored for that reaction is not the equation assigned by the EC (e.g., we have corrected the EC equation). Otherwise, the value is YES, which is the default inherited value.

10.13.3 Left, Right

These slots hold the compounds from the left and right sides, respectively, of the reaction equation. Each value is either the key of a compound frame, or a string that names a compound (when the compound is not yet described within the DB as a frame). The terms *reactant* and *product* are not used because these terms may falsely imply the physiological direction of the reaction.

The coefficient of each substrate, when that coefficient is not equal to 1, is stored as an annotation on the substrate value. The annotation label is **COEFFICIENT**.

The substrates of transport reactions are also described using the **Left** and **Right** slots. However, the values of these slots are annotated to indicate their compartments. For example, a transporter that moves succinate from the periplasm to the cytoplasm, accompanied by hydrolysis of ATP in the cytoplasm, would be described with **succinate** and **ATP** as the values of the **Left** slot, and with **succinate**, **ADP**, and $\mathrm{P}i$ as the values of the **Right** slot. The **succinate** in the **Left** slot would be annotated with **Periplasm** under the label **Compartment**. The other substrates need not be annotated with a compartment because the default compartment is taken to be the cytoplasm.

10.13.4 Substrates

The value of this slot is computed automatically — its values may not be changed by the user. The values of the slot are computed as the union of the values of the **Left** and **Right** slots.

10.13.5 Enzymes-Not-Used

Proteins or protein-RNA complexes listed in this slot are those which would otherwise have been inferred to take part in the pathway or reaction, but which in reality do not. In other words, the protein may catalyze a general reaction with non-specific substrates, but is known not to catalyze this specific form of the reaction.

10.13.6 DeltaG0

This slot contains the change in Gibbs free energy for the reaction in the direction the reaction is written.

10.13.7 Spontaneous?

This slot is true in the case when this reaction occurs spontaneously, that is, it is not catalyzed by any enzyme.

10.13.8 Species

This slot is used to indicate that a reaction is known to occur in an organism in the case where the enzyme that catalyzes the reaction is unknown. In such cases, the value for this slot in a given reaction would be the symbolic identifier of the species for the organism for the current PGDB.

10.13.9 Cannot-Balance?

This slot is used to track which unbalanced reactions have been examined by curators and which have not been examined. It is a boolean 'true' if the curator has marked the reaction as unable to be balanced, and it is a boolean 'false' otherwise.

This slot is true when a curator has checked a reaction for mass balance, but is unable to determine how to balance the reaction, such as if the authors who published the reaction knew it to be incomplete, or if it was apparently not known to be unbalanced but the proper changes to the reaction cannot be determined.

10.14 Class Transcription-Units

Frames in this class encode transcription units, which are defined as a set of genes and associated control regions that produce a single transcript. Thus, there is a one-to-one correspondence between transcription start sites and transcription units. If a set of genes is controlled by multiple transcription start sites, then a PGDB should define multiple transcription-unit frames, one for each transcription start site.

10.14.1 Components

The **Components** slot of a transcription unit lists the DNA segments within the transcription unit, including transcription start sites (Promoters), Terminators, DNA binding sites, and genes.

10.14.2 Extent-Unknown?

The value of this slot should be True when it is not known to how many genes the transcription unit extends; that is, it is not known which is the last gene in the transcription unit.

10.15 Class tRNAs

Frames of this class encode both charged and uncharged tRNAs.

10.15.1 Anticodon

This slot contains a string as a single value, which lists the three letters that make up the anticodon bases on the tRNA. The direction in which the letters are listed is 5′ to 3′ with respect to the tRNA. This is the reverse of, and complementary to, the sequence of the recognized codons.

10.15.2 Codons

This slot contains possibly multiple values as strings, which list the three letters that make up the base triplets recognized by the anticodon on the tRNA. The direction in which the letters are listed is 5′ to 3′ with respect to the coding strand of genes.

10.16 Class Regulation

This class describes most forms of protein, RNA or activity regulation. Regulation can be either by a direct influence on the protein's activity (e.g. allosteric inhibition of an enzyme) or by influencing the quantity of active protein available (e.g. by inducing or blocking its transcription or translation). The one form of regulation that is not covered by this class is when the quantity of a protein is regulated as a result of chemical or binding reactions that either produce or consume the active form of a protein – these are represented as Reactions instead. There can be some ambiguity as to what should be represented as a reaction and what should be represented as a regulation event. In general, an event that can be represented as a reaction should be when a) there is sufficiently detailed information known to model it as a reaction, b) both reactants and products exist as stable, independent entities, and c) our schema supports referring to both reactant and product of the reaction independently and there is some justification for wanting to go down to that level of detail. For example, a transcription factor bound to a small molecule will generally have a different activity than the unbound transcription factor. This could be represented either as the reaction TF + x -¿ TF-x or as a regulation event in which x activates or inhibits the activity of TF. However, because both TF and TF-x are stable molecules which can potentially regulate different transcription units (not all will, but some do), or TF could bind another small molecule y and regulate yet another set of transcription units, we prefer to model this kind of interaction as a reaction when the data is available. On the other hand, an enzyme binding to some inhibitor could also be represented as a reaction, but since there is rarely any reason to refer to the enzyme-inhibitor complex outside of the context of the reaction the enzyme catalyzes, we choose instead to model these events as regulation events in which the inhibitor regulates the activity of the enzyme.

Instances of this class represent a one-to-one mapping between regulator and regulated-entity (i.e. an entity may regulate many processes, or a process may be regulated by many entities, but each one requires its own instance of Regulation to represent it)

Some of the slots listed below are applicable only to certain subclasses of Regulation.

10.16.1 Associated-Binding-Site

This slot is applicable to regulation of transcription or translation in which an entity (protein, small-molecule or RNA) binds to DNA or the mRNA transcript. Its values are instances of either DNA-Binding-Sites or mRNA-Binding-Sites, depending on the type of regulation.

10.16.2 Mechanism

This slot optionally contains a keyword which describes the mechanism of the regulation. Appropriate possible values will vary depending on the particular subclass of regulation. Some subclasses will not use this slot at all.

10.16.3 Mode

This slot specifies whether the regulator activates or inhibits the regulated-entity. Possible values are:

"+" — The regulator activates or increases quantity or activity of the regulated-entity (an exception is transcription attenuation, in which even though the regulated-entity is a terminator object, "+" means activation of transcription of the downstream genes rather than of the terminator).

"-" — The regulator inhibits or decreases quantity or activity of the regulated-entity (with the same caveat about transcription attenuation as above)..

10.16.4 Regulated-Entity

This slot links the regulation frame to the object that is being regulated. In the case of enzyme modulation, this object will be an Enzymatic-Reaction frame. In the case of transcription initiation regulation, it will be a Promoter frame. In the case of transcription attenuation, it will be a Terminator frame. In other cases, it could be a gene or a protein frame. The regulated entity will link back to the regulation frame using the inverse of this slot, Regulated-By

10.16.5 Regulator

This slot links the regulation frame to the object that is doing the regulating, typically a protein, RNA or small molecule. The regulator frame will link back to the regulation frame using the inverse of this slot, Regulates.

10.16.6 Antiterminator-Start-Pos, Antiterminator-End-Pos, Anti-Antiterm-Start-Pos, Anti-Antiterm-End-Pos, Pause-Start-Pos, Pause-End-Pos

These slots provide positional information for the Antiterminator, Anti-Antiterminator, and Ribosome Pause Site for instances of certain classes of Transcriptional Attenuation. All values are relative to the start of the chromosome.

Bibliography

[1] Bairoch. The enzyme data bank in 1995. *Nuc. Acids R*es., 24:221-222, 1996.

[2] F.R. Blattner, G. Plunkett III, C.A. Bloch, N.T. Perna, V. Burland, M. Riley, J. Collado-Vides, J.D. Glasner, C.K. Rode, G.F. Mayhew, J. Gregor, N.W. Davis, H.A. Kirkpatrick, M.A. Goeden, D.J. Rose, B. Mau, and Y. Shao. The complete genome sequence of *Escherichia coli* k-12. *Science*, 277:1453-1462, 1997.

[3] NCBI DDBJ, EMBL. *The DDBJ/EMBL/GenBank Feature Table Definition*, version 2.0 edition, December 1997. `http://www.ncbi.nlm.nih.gov/collab/FT/index.html`.

[4] P. Karp. Database links are a foundation for interoperability. *Trends in Biotechnology*, 14:273-279, August 1996.

[5] P. Karp. The EcoCyc user's guide. Unpublished; see WWW URL `http://ecocyc.PangeaSystems.com/ecocyc/doc/ecocyc-uguide/paper.html`, 1996.

[6] P. Karp and T. Gruber. The generic frame protocol. Available via WWW URL `http://www.ai.sri.com/~gfp/doc/paper.html`, 1995.

[7] P. Karp and M. Mavrovouniotis. Representing, analyzing, and synthesizing biochemical pathways. *IEEE Expert*, 9(2):11-21, 1994.

[8] P. Karp and S. Paley. Representations of metabolic knowledge: Pathways. In R. Altman, D. Brutlag, P. Karp, R. Lathrop, and D. Searls, editors, *Proceedings of the Second International Conference on Intelligent Systems for Molecular Biology*, pages 203-211, Menlo Park, CA, 1994. AAAI Press.

[9] P. Karp and S. Paley. Automated drawing of metabolic pathways. In H. Lim, C. Cantor, and R. Robbins, editors, *Proceedings of the Third International Conference on Bioinformatics and Genome Research*, pages 225-238. World Scientific Publishing Co., 1995. See also WWW URL ftp://ftp.ai.sri.com/pub/papers/karp-bigr94.ps.Z.

[10] P. Karp and S. Paley. Integrated access to metabolic and genomic data. *Journal of Computational Biology*, 3(1):191-212, 1996.

[11] P. Karp and M. Riley. Representations of metabolic knowledge. In L. Hunter, D. Searls, and J. Shavlik, editors, *Proceedings of the First International Conference on Intelligent Systems for Molecular Biology*, pages 207-215, Menlo Park, CA, 1993. AAAI Press.

[12] P. Karp, M. Riley, S. Paley, A. Pellegrini-Toole, and M. Krummenacker. EcoCyc: Electronic encyclopedia of E. coli genes and metabolism. *Nuc. Acids Res.*, 26(1):50-53, 1998.

[13] P.D. Karp. A knowledge base of the chemical compounds of intermediary metabolism. *Computer Applications in the Biosciences*, 8(4):347-357, 1992.

[14] M. Riley. Functions of the gene products of Escherichia coli. *Microbiological Reviews*, 57:862-952, 1993.

[15] P. R. Romero and P. Karp. Nutrient-related Analysis of Pathway/Genome Databases. *Pacific Symp. Biocomputing*, pages 471-482. See also WWW URL http://www.ai.sri.com/pkarp/pubs/01psb.pdf

[16] J.-F. Tomb et al. The complete genome sequence of the gastric pathogen helicobacter pylori. *Nature*, 388:539-547, 1997.

[17] R.A. VanBogelen, P. Sankar, R.L. Clark, J.A. Bogan, and F.C. Neidhardt. The gene-protein database of *Escherichia coli:* Edition 5. *Electrophoresis*, 13:1014-1054, 1992.

[18] Edwin C. Webb. Enzyme Nomenclature, 1992: Recommendations of the nomenclature committee of the International Union of Biochemistry and Molecular Biology on the nomenclature and classification of enzymes. Academic Press, 1992.

[19] D. Weininger. SMILES, a chemical language and information system. 1. Introduction to methodology and encoding rules. *J. Chem. Inf. Comput. Sci.*, 28:31-36, 1988.

Index

-acl-file *filepath*, 12
-allow-webcrawlers, 13
-api, 11
-background-color *color*, 14
-blast, 14
-dbdef, 11
-email *email-address*, 14
-eval *expression*, 11
-gene-link-db *db*, 14
-google-text-search, 14
-id, 11
-linkdef *filepath*, 11
-lisp, 11
-load *filepath*, 12
-no-blast, 14
-no-google-text-search, 14
-org *orgid*, 12
-passwd-file *filepath*, 15
-patch, 12
-patho, 12
-port *NNN*, 15
-proxy-port *NNN*, 15
-rdbmstest, 12
-user *username*, 15
-www, 15
-www-publish, 317
-www-publish *pubspec*, 15
/EC_number (Genbank Qualifier), 125
/alt_name (Genbank Qualifier), 126
/db_xref (Genbank Qualifier), 126
/gene (Genbank Qualifier), 125
/gene_comment (Genbank Qualifier), 125
/go_component (Genbank Qualifier), 126
/go_function (Genbank Qualifier), 126
/go_process (Genbank Qualifier), 126
/locus_tag (Genbank Qualifier), 125
/note (Genbank Qualifier), 126
/product (Genbank Qualifier), 126
/product_comment (Genbank Qualifier), 125
/pseudo (Genbank Qualifier), 126

Abort Changes (Pathway Editor), 219
Access control, 12, 15
Access Control List, 12
ACL, 12
Activate Installed Patches (Software Patch), 75
Activators, 55, 59, 62, 322, 328, 329
Add current history item to Complex (Button), 155
Add Pathway to File Export List (Import/Export), 98
Add Connection (Pathway Editor), 219
Add Link from/to Pathway (Pathway Editor), 221
Add or Replace Sequence File, 68
Add or Replace Sequence File (Chromosome Menu), 145
Add Reaction (Pathway Editor), 219
Add Reaction(s) from History (Pathway Editor), 219
Add Subpathway by Class (Pathway Editor), 221
Add Subpathway by Name (Pathway Editor), 220

Add Subpathway by Substring (Pathway Editor), 220
Add to history (Submenu), 253
Adding an Enzyme to a Reaction (Protein Editor), 214
Additional Pathway Tools Publications and Web Sites, xii
Advanced Editing Topics, 247
Advanced Search (Compound Menu), 65
Alignment (Genome Browser), 72
Ambiguous Matches (PathoLogic Outcome), 139
Amino Acid Sequence (Retrieving), 62
Animation (Overview), 50
Annotation File, 119, 121, 125, 128, 135, 136, 139, 141
Annotations, 97, 205, 330, 338, 345
Answer List, 253
Answer List, 25
Answer List, 25, 26, 74
Answer List (Preferences), 83
Answer List (Tools Menu), 76
Are data values log values, or do they use a zero-centered scale? (SAM Output File), 48
Artemis (Import/Export), 99
Ask user (Import Dialog), 106
Assign an energy coupling to transporter (PathoLogic), 162
Assign colors manually (SAM Output File), 49
Assign Modified Proteins (PathoLogic), 156
Assign Polymerization Name (Pathway Editor), 221
Assign Probable Enzymes (PathoLogic), 140, 148
Assign to Reaction(s) (PathoLogic), 150
Assigning Evidence Scores to Predicted Pathways (PathoLogic), 141, 143
Attempt to Reconnect to Database Server (File Menu), 251
Attribute-value file format, 107
Author Crediting, 266
Automated Build (PathoLogic), 136
Automatic Tasks (Consistency Checker), 78, 79
Auxiliary Nutrients, 53

Back (Button), 73
Back (History), 73, 91
Back (In History), 26, 90
Backup DB to File (PathoLogic), 148
Balance State (Reaction Editor), 229
Batch Mode (PathoLogic), 189
Binding Reactions (Reaction Editor), 225
Binding Sites (In Transcription Units), 65
Biomass Composition, 53
Biomass Composition Editor, 53
BLAST, 14, 303
  Setting up Web Server access, 303
BLAST Reference Data (Tools Menu), 76
Browse Downloadable PGDBs (Tools Menu), 77
Browse Frames (Export Settings), 101
Build Pathway/Genome Database (PathoLogic), 136
Buttons
  Stop, 156
  Add Current History Item to Complex, 155
  Back, 73
  Clone, 74, 75
  Enzyme View, 57
  Fix, 75, 93
  Forward, 73
  History, 26, 74
  Home, 20, 73
  Make Complex(es), 156
  More Detail/Less Detail, 57
  Next Answer, 74
  OK, 156, 229
  Show, 157
  Skip, 154, 156
  Unfix, 75

Catalytic Activity (Enzyme), 59
Cellular Overview, 31
  Commands, 33
Changing a Gene's Functional Annotation, 243
Changing the Annotation for an Enzyme Gene, 244
Checkpoint Current DB Updates to File (File Menu), 250
Checkpoints, 249
Chemical Reactions (Reaction Editor), 225
Choose Main Compounds for Reaction (Pathway Editor), 220
Choose Slots (Export Settings), 101
Choose Species Menu, 90
Choose Species Menu, 91
Chromosome (Genome Browser), 66
Chromosome Menu, 68
  Add or Replace Sequence File, 68, 145
  Print Poster, 68
  Select & Browse Chromosome/Replicon, 68
  Show Sequence of a Segment, 68
Citation Reference Style (Preferences), 81
Citations, 29, 131, 150, 204, 215, 229, 242, 243, 257
  Importing from PubMed, 107
Citations (Reaction Editor), 229
Citations from PubMed (Import/Export), 242
Citations from PubMed (File Menu), 107
Classes, 30
Classification Systems, 30
Clear All (Overview Highlight), 89
Clone (Button), 74, 75
Clone (Overview), 93, 94
Clone a Reaction Frame (Pathway Editor), 219
Cofactors (Enzyme), 59, 62
Color (Preferences), 81
Color Scales (Overview), 46
Column delimited file format, 107
Column Delimiter (Export Settings), 102
Column Delimiter (Import Dialog), 103
Column zero contains (SAM Output File), 48
Command Buttons (Pathways), 57
Command Line Arguments
  Web Mode
    -www-publish, 317
Command Line Arguments, 6, 9, 11
  -acl-file *filepath*, 12
  -allow-webcrawlers, 13
  -api, 11
  -background-color *color*, 14
  -blast, 14
  -dbdef, 11
  -email *email-address*, 14
  -eval *expression*, 11
  -gene-link-db *db*, 14
  -google-text-search, 14
  -id, 11
  -linkdef *filepath*, 11
  -lisp, 11
  -load *filepath*, 12
  -no-blast, 14
  -no-google-text-search, 14
  -org *orgid*, 12
  -passwd-file *filepath*, 15
  -patch, 12
  -port *NNN*, 15
  -proxy-port *NNN*, 15
  -rdbmstest, 12
  -user *username*, 15
  -www, 15
  -www-publish *pubspec*, 15
Command Line Arguments, 294
Comment Slot, 123
Comments, 29
  Bold face, 256
  Centering, 256
  Greek letters, 256
  Italics, 256
  Subscripts, 256

Superscripts, 256
Underlining, 256
Comments (Reaction Editor), 229
Commercial users, 77
Common Name (Gene), 121
Comparative Analyses
Comparisons of specific instances, 86
Shared, 87
Species Comparison Dialog Box, 88
Comparative Analyses
ALL, 88
Any, 88
Current Organism
Dependent Comparisons, 89
Independent Comparisons, 90
Global Comparative Analyses, 86
Global Comparisons, 86
Not-Shared, 87
Parallel Comparison, 91
Pathways and Genomes, xi, 86
Select All, 88
Sequential Comparison, 89
Species Comparisons, 88
Comparative Genome Browser, 72
Comparative Operations, 86
Complex Processes (Reaction Editor), 226
Compound Editor, 201
Compound Duplicate Checker, 238
Delete, 240
Merge, 240
Replace Entered cpd with X, 240
Show, 239
Compound Editor, 232
Compound Menu, 64
Advanced Search, 65
New, 201, 232
Search by Class, 65
Search by Name or Frame ID, 64
Search by SMILES substructure, 64
Search by Substring, 64
Compound Resolution Tool, 224
Limitations, 225
Compound Structure Editor, 232
Compounds, 64
Compounds (Overview Highlight), 34
Compression, 7
Concurrency Control, 247
Conditions for Pathway Exclusion (PathoLogic), 142, 143
Consistency Checker (Tools Menu)
Manual Tasks, 79
Show object in the navigator, 80
Consistency Checker (Tools Menu), 78
Automatic Tasks, 79
Automatic Tasks, 78
Invoke the object editor, 80
Manual Tasks, 78
Revert Database, 80
Save Database, 80
Show History, 80
Construct enzymatic reaction linking reaction with transport protein (PathoLogic), 165
Construct full reaction (PathoLogic), 164
Conversion Type (Reaction Editor), 228
Convert File DB to Oracle or MySQL DB (PathoLogic), 148
Cookie, 72
Copy frame name to Clipboard (Submenu), 252
Create New Curator (Curator Editor), 266
Create (Substrate), 229
Create Binding Interaction, 200, 210
Create frame (Submenu), 252
Create genetic-elements.dat file (PathoLogic), 132
Create New (PathoLogic), 130
Create New Organism (PathoLogic), 130
Create New Organization (Organization Editor), 266
Create New Reaction Frame (Pathway Editor), 219
Create Polymerization Link (Pathway Editor), 221

Create Protein Complexes (PathoLogic), 153
Create/Add Enzyme, 200, 201, 214, 245
Creating a Pathway/Genome Database (PathoLogic), 118, 129
Creating Active/Inactive Forms of an Enzyme (Protein Editor), 215
Creating New Publication Frames (Publication Editor), 242
Creating Protein Features (Protein Editor), 215
Crediting Authors of PGDB Objects, 266
Credits, 267
CSS Style Sheet
  Customization, 298
  userWebsiteCustomization.css, 298
Curator Crediting, 264
Curator Editor, 266
Current Organism, 21, 22
Customizing Pathway Tools
  Desktop Operation, 80
  Init File, 6
  Site-Wide Configuration, 6
  User Preferences, 80
Customizing the Web Server Pages, 296
  CSS Style Sheet, 298
  HTML Virtual Inclusion, 300
  JavaScript, 298
  Top Menu Bar, 299

Database Sharing
  Publishing
    Moving the archive to an FTP server, 111
    Packaging the database, 111
    Registering, 111
Database Generation Perspective (PathoLogic), 118
Database Links, 11, 30, 100, 258, 261, 353
  Creating Links, 258, 260
  Import from File, 261
  Manual Creation, 260
  Relationship Links, 263
  Unification Links, 263
Database Sharing, 109
  Publishing, 110
  Setting Preferences, 112
    Directory to which archive files will be copied, 113
    FTP server, 113
    Rerun initial setup, 113
    Username for storing to the FTP server above, 113
Database Sharing (Preferences), 84
DBLink (PathoLogic File Format), 122
DDBJ/EMBL/GenBank Feature Table Definition, 124
Default color scheme with a specified maximum value bin cutoff (SAM Output File), 49
Delete (Compound Duplicate Checker), 240
Delete a DB (File Menu), 251
Delete frame (Submenu), 252
Delete Polymerization Link (Pathway Editor), 221
Delete Predecessor/Successor Link (Pathway Editor), 219
Delete Reaction from Pathway (Pathway Editor), 220
Delete Subpathway (Pathway Editor), 221
Dependent Comparisons (Comparative Analyses), 89
Diagrams (Pathways), 54
Directory to which archive files will be copied (Database Sharing), 113
Disconnect All Reactions (Pathway Editor), 220
Disconnect Reaction (Pathway Editor), 220
Disconnecting an Enzyme from a Reaction (Protein Editor), 215
Display
  Gene, 62
  Pathways, 55
  Proteins, 59
  Reaction, 58

RNA, 62
Display (Pathways), 57
Display (Pathways), 55, 56
Display (Transcription Unit), 65
Display all connections for compounds in this pathway (Overview Submenu), 37
Display all connections for products of this reaction (Overview Submenu), 37
Display all connections for reactants of this reaction (Overview Submenu), 37
Display all connections for substrates of this reaction (Overview Submenu), 37
Display all connections for this compound, 36
Display compound information in main display (Overview Submenu), 36
Display compound information in pop-up window (Overview Submenu), 36
Display pathway information in main display (Overview Submenu), 37
Display pathway information in pop-up window (Overview Submenu), 37
Display reaction information in main display (Overview Submenu), 36
Display reaction information in pop-up window (Overview Submenu), 36
Display on Cellular Overview, 53
Download and Activate All Patches (Tools Menu), 75
Download speed, 78
Downloading PGDBs from the Registry, 110
Drawings, 338
Pathways, 55, 57

EC (PathoLogic File Format), 122
EC number (Reaction Editor), 228
EC number \t See Enzyme Commission Number, 58
EcoCyc, xi, 138
ecocyc-prefs File (Preferences), 80
Edit (Overview Submenu), 36–38
Edit External Database, 261
Edit Protein Feature(s) (Protein Editor), 215
Edit Reaction Frame (Pathway Editor), 220
Edit Remote Database Info, 261
Edit Submenu (on Right-Button menu), 252
Editing Examples, 243
Editing Existing Publication Frames (Publication Editor), 243
Editing Restrictions, 251
Editors, 197, 198, 202, 205
Chemical Compound Editor, 201
Compound Editor, 232
Compound Structure Editor, 201
Create Binding Interaction, 200
Create/Add Enzyme, 200, 201
Curator Editor, 266
Edit Publication Data, 202
Frame Editor, 252
Gene Editor, 198, 199, 206, 252
Intron Editor, 199, 211
JME Chemical Structure Editor, 201, 236
Marvin Chemical Structure Editor, 232
Marvin Compound Structure Editor, 201
Object-Type Editor, 252
Ontology Editor, 252
Organization Editor, 266
Pathway Editor
Invoking Pathway Editor, 217
Pathway Editor, 216–219
Pathway Info Editor, 201, 216, 217
Protein Editor, 212
Protein Editor, 200, 210, 212–215
Protein Subunit Structure Editor, 212
Publication Editor, 242
Reaction Editor, 201, 215, 225, 229
Balance State, 229
Cancel, 229
Citations, 229
Comment, 229
Conversion Type, 228
EC number, 228
Entering Reaction Equations, 228

Invoking the Reaction Editor, 225
Mass Balance of Equation, 228
Official EC?, 228
OK, 229
Orphaned reaction?, 229
Quick Entry of Equation, 228
Spontaneous?, 229
Regulation Editor, 241
Regulatory Interaction Editor, 200
Relationships Editor, 252
RNA Editor, 200
Segment Editor, 222, 223
Segment Editor, 216, 217, 222, 224
Synonyms Editor, 252
Terminator Editor, 200
Terminator Editor
Invoke Terminator Editor, 209
Transcription Unit Editor, 199, 207, 208
Invoking Transcription Unit Editor, 208
Electron Transport Reactions, 230
Enable PGDB sharing Functionality, 113
Endbase (PathoLogic File Format), 121, 122
Enter a Linear Pathway Segment (Pathway Editor), 220, 246
Entering a New Pathway, 245
Entering Reaction Equations (Reaction Editor), 228
Entire DB to attribute-value and BioPAX files (Import/Export), 100
Entries
SwissProt, 30
Enzymatic Reaction (PathoLogic), 119
Enzyme, 178
Catalytic Acitvity, 59
Substrate Specificity, 59
Enzyme Commission Number, 58, 59, 138, 141
ENZYME database, 138
Enzyme View (Pathways), 57
enzyme-mappings.dat file (PathoLogic), 138
ESTs (Prepare BLAST Reference Data), 76
Evidence, 186
Evidence Codes, 159, 160
Exit
File Menu, 80
Pathway Tools, 80
Transporter Inference Parser, 168
Exit Menu (Pathway Editor), 219
Experiment Title (SAM Output File), 47
Export (File Menu), 251
Export \t See Import/Export, 97
Export compound structure to molfile... (MDL Molfile), 241
Export Object-Type to DB... (Submenu), 252
Export Pathway to DB... (Import/Export), 98
Expression Data, 31, 43
Expression Dataset File Format, 45
External Database Description, 260

Facets, 204
FAQ, 319
FASTA file (PathoLogic Input File Format), 119, 121
Feature Table (PathoLogic), 124
File
Create
External Database Description, 260
File (SAM Output File), 47
File Format
GenBank, 124
Gene Expression Data, 45
genetic-elements.dat, 120
Molfile, 241
Omics Data, 45
PathoLogic, 121
ptools-init.dat, 6
File format
Attribute-value, 107
Column delimited, 107
Import/Export, 106
File Format (Export Settings), 102
File Format (Import Dialog), 103

File Menu, 250
Attempt to Reconnect to Database Server, 251
Checkpoint Current DB Updates to File, 250
Checkpoint Current DB Updates to File, 250
Create
Curator, 266
Organization, 266
Delete a DB, 251
Exit, 10, 80
Export, 251
Import, 251
Citations from PubMed, 242
Citations from PubMed, 107, 202
List Unsaved Changes in Current DB, 250
Print, 74, 251
Refresh All Open DBs, 249, 250
Restore Updates from Checkpoint File, 250
Restore Updates from Checkpoint File, 251
Revert Current DB, 106, 202
Revert Current DB, 249, 250
Save Current DB, 202
Save Current DB, 251
Summarize Current Organism, 250
Summarize Databases, 20, 250
Filter (PathoLogic), 150
Filter candidates (PathoLogic), 161
Find candidate transporter proteins (PathoLogic), 161
Find Curator (Curator Editor), 266
Find Organization (Organization Editor), 266
Firewall, 78
Fix (Button), 75, 93
fna file suffix (PathoLogic Input File Format), 121
Font size for mains (Pathway Display Preferences), 82
Font size for sides and enzymes (Pathway Display Preferences), 82
Forward (Button), 73
Forward (History), 26, 73, 90
Forward Propagation, 51
Frame Data Model, 202
Frame Editor, 252
Frame Export (Import/Export), 101
Frame Import, 103
Frame import/export, 100
Frame Name Conventions, 254
Frame References, 258
Frames, 203
Classes, 142, 225
Frames from File...(Import/Export), 100, 103
Frequently Asked Questions, 319
FTP server
Database Sharing, 113
Full Flat File Dump (Import/Export), 100
Function (PathoLogic File Format), 122
Function-Comment (PathoLogic File Format), 123
Function-Synonym (PathoLogic File Format), 123

gbk file suffix, 121
GenBank File Format, 124
GenBank file format (PathoLogic Input File Format), 121
Genbank format export, 99
Genbank Qualifiers (PathoLogic File Format), 124, 126
Gene
Gene Introns, 63
Gene Display, 62
Nucleotide Sequence, 63
Nucleotide Sequence Neighborhood, 64
Protein Sequence, 64
Operons, 60, 67

Orthologs, 72
Paralogous Groups, 63
Replicons, 72
Stop codon, 63
Terminators, 63
Transcription Unit, 63, 67, 157
Transcription Unit, 60, 65
Binding Sites, 65
Display, 65
Start Site, 65
Transcription Factor, 65
Gene Menu
Search by Substring (Gene), 244
Gene (Overview Highlight), 34
Gene Description Page (Genome Browser), 67
Gene Editor, 199, 206
Gene Expression Data, 31, 43
Gene Frame, 123, 125
Gene Menu, 63
New Gene (Gene), 198
New Operon, 199, 208
Search by Class, 63
Search by Name or Frame ID, 63
Search by Substring, 63
Gene Ontology (PathoLogic), 127
Gene Page
Terminators, 65, 67
Gene Position (Pathways), 56
Gene-Comment (PathoLogic File Format), 123
Gene-Reaction Schematic, 28, 29
General (Preferences), 84
Generate Link Tables... (Import/Export), 100
Genes, 62
Genetic Element (PathoLogic Input File Format), 119
Genetic Element (PathoLogic), 119
genetic-elements.dat file (PathoLogic Input File Format), 120, 132
genetic-elements.dat file (PathoLogic Input), 119
Genome Browser
Alignment, 72
Genome Browser
Alignment, 72
Phantom-genes, 67
Selected Gene, 67
Genome Browser, 66
Chromosome, 66
Select & Browse Chromosome/Replicon, 66
Comparative Genome Browser, 72
Gene Description Page, 67
Intergenic Region, 67
Navigate, 67, 72
Organism Set, 72
Plasmid, 66
Pseudo-genes, 67
Start and end base-pair positions, 67
Tickmarks, 66
Transcription Direction, 67
Zoom, 67
Genome Overview, 42
Commands, 43
Genomic-Map Display \t See Genome Browser, 66
GO (PathoLogic File Format), 122
Greek letters, 256
Group into complexes (PathoLogic), 164
Growth Medium, 53
Growth Medium Editor, 53
Guess Pathway Predecessor List (Pathway Editor), 220
gzip, 7

Help Menu, 80
Highlight (Overview), 33
Highlight all reactions of this compound (Overview Submenu), 36
Highlight Genes and Regulatory Relationships (Regulatory Overview), 41

Highlight reactions involving genes in same operon/regulon (Overview Submenu), 37
Highlight this compound everywhere it appears as a main (Overview Submenu), 36
Highlight this pathway (Overview Submenu), 37
Highlight this reaction everywhere it appears (Overview Submenu), 36
Highlighting Color Scheme (SAM Output File), 49
History, 25, 26, 73
  Backward in History, 26, 90, 91
  Backward in History, 26, 73
  Button, 74
  Forward in History, 26, 73, 90
  History List, 26
  Preferences, 83
  Tools Menu, 75
Hole Filler, 178
Home (Button), 20, 73

ID (PathoLogic File Format), 121
Identify any co-substrate (PathoLogic), 162
Identify substrate(s) (PathoLogic), 162
Identify the compartment of each substrate (PathoLogic), 163
Identify transporters that consist of subunits (PathoLogic), 164
Ignore completely (Import Dialog), 106
Import (File Menu), 251
Import compound structure from molfile... (MDL Molfile), 241
Import/Export, 97, 98
  Add Pathway to File Export List, 98
  Artemis, 99
  Citations from PubMed, 107
  Entire DB to attribute-value and BioPAX files, 100
  Export Settings
    Include file header, 103
  Export compound structure to molfile..., 241
  Export Pathway to DB..., 98
  Export Settings
    Browse Frames, 101
    Choose Slots, 101
    Column Delimiter, 102
    File Format, 102
    Include, 102
    Multiple Slot Value Delimiter, 102
    Output file, 103
    Source of frames to be exported, 101
  Frame Export, 101
  Frame Import, 103
  Frame import/export, 100
  Frames from File..., 100, 103
  Full Flat File Dump, 100
  Genbank format export, 99
  Generate Link Tables..., 100
  Import compound structure from molfile..., 241
  Import Dialog, 105
    About the log file, 106
    Column Delimiter, 103
    File Format, 103
    Input file, 106
    Multiple Slot Value Delimiter, 104
  Linking Table Export, 100
  MDL Molfile, 241
  Pathways from File..., 99
  SBML export, 99
  Selected Frames to File, 101
  Selected Chromosome to Genbank File..., 99
  Selected Frames to File, 100
  Selected Pathways to File..., 98
  Selected Reactions to SBML File..., 99
  Supported file formats, 106
    Attribute-value format, 107
    Column delimited format, 107
Importing Citations From PubMed, 107
Include (Export Settings), 102

Include file header (Export Settings), 103
Independent Comparisons (Comparative Analyses), 90
Individual transporter dialog (Transporter Inference Parser), 166
Inhibitors (Enzyme), 55, 59, 62
Initialization File (Pathway Tools), 6
Input (PathoLogic), 119
Input file (Import Dialog), 106
Input File Format (PathoLogic), 120
Input Project Information Window (PathoLogic), 128
Installation
  JME, 236
  Marvin, 233
Instant Patch, 75
Intergenic Region (Genome Browser), 67
Intron (PathoLogic File Format), 123
Intron Editor, 199, 211
Introns (Gene), 63
Invoke Relationships Editor (Pathway Editor), 220
Invoke the object editor (Consistency Checker), 80
Invoke Reachability Analysis, 53
Invoking the Pathway Editors, 217
Invoking the Reaction Editor, 225

Java applet (JME), 236
JavaCyc, 11
JavaScript
  Customization, 298
  userWebsiteCustomization.js, 298
JME Chemical Structure Editor, 201, 236, 237
  Installation, 236
  Limitations, 238
  Structure Editing, 237

Keep Changes (Pathway Editor), 219
Key (Overview), 50
Knowledge Base, 203, 354

Layout for linear pathways (Pathway Display Preferences), 82
Leave Unconnected (PathoLogic), 157
Legend (Overview), 50
Licensing, 114, 116
Limitations (JME), 238
Limitations (Marvin), 236
Linking Table Export (Import/Export), 100
Links
  Relationship Links, 30
  Unification Links, 30
List Unsaved Changes in Current DB (File Menu), 250
Load from File (Overview Highlight), 35
Load Regulatory Overview from File (Regulatory Overview), 41
Load Regulatory Overview from File (Regulatory Overview), 43
Log and continue (Import Dialog), 106
Log File, 303
Lost DB updates, 249

Macromolecular Metabolism, 117
Main display (Transporter Inference Parser), 165
Make Complex(es) (Button), 156
Manual Tasks (Consistency Checker), 79
Manual Tasks (Consistency Checker), 78
Marvin Chemical Structure Editor, 201, 232
  Installation, 233
  Limitations, 236
  Structure Editing, 233
Mass Balance of Equation (Reaction Editor), 228
Matching of Enzymes to Reactions (PathoLogic), 138
MDL Molfile (Import/Export), 241
Menus
  Choose Species, 91
  Choose Species, 90, 91
  Chromosome Menu, 68
  Color , 81
  Compound Menu, 64

Gene Menu , 63
Help Menu, 80
Pane Layout Menu, 81
Pathway Display, 82
Pathway Menu , 57
Preferences Menu , 80
Protein Menu , 60
Reaction Menu, 58, 62
Tools Menu, 75
Menus and Dialogs \t See Pathway/Genome Navigator-Menus and Dialogs, 18
Merge (Compound Duplicate Checker), 240
Metabolic Pathways, 117
Metabolite Tracing (Overview), 50
Metabolomics Data, 44
MetaCyc, xi, 117, 118, 138, 142
Propagate Data Updates to PGDB, 192
MetaCyc (PathoLogic Input File Format), 119
Microarray Gene Expression Data, 43, 44
Microsoft Windows users, 77
Molfile (Import/Export), 241
Monomers, 153, 156
More Detail/Less Detail (Pathways), 57
Moving the archive to an FTP server (Publishing Database), 111
Multiple Slot Value Delimiter (Export Settings), 102
Multiple Slot Value Delimiter (Import Dialog), 104
MySQL server mode
Parameters
RDBMS-Database-Name, 8
MySQL server mode
Parameters, 8
RDBMS-Password, 8
RDBMS-Server-Hostname, 8
RDBMS-Server-Port, 8
RDBMS-Username, 8

Name (PathoLogic File Format), 121
Name-matching outcome (PathoLogic), 139
Name-matching Tool (PathoLogic), 119
Naming
Gene Frames, 256
Slots, 256
Navigate (Genome Browser), 67, 72
Navigator, 10
New
Compound, 232
Operon, 199, 208
Pathway, 245
Publication, 243
Reaction, 215, 225
New Operon (Gene Menu), 208
New Operon (Gene), 199
New PGDB Version (PathoLogic), 148
Next Answer (Button), 74
No Match (PathoLogic Outcome), 139
Not-Shared (Comparative Analyses), 87
Nucleotide Sequence (Gene), 63
Nucleotide Sequence Neighborhood (Gene), 64

Object Correspondence, 259
Object Displays and Queries, 27
Object Names Visible to PGDB Users, 253
Object-Type Editor, 252
Ocelot, 202, 203, 248, 249
Official EC? (Reaction Editor), 228
OK (Button), 156
OK (Reaction Editor), 229
Omics Dataset File Format, 45
Omics Viewer, 31, 43, 44, 46–48
Metabolomics Data, 44
Microarray Gene Expression Data, 43, 44
Other Experimental Data, 43, 44
Proteomics Data, 44
Reaction Flux Data, 44
Omitted PGDBs, 78
Ongoing PGDB Curation (PathoLogic), 192
Ontology Browser (Tools Menu), 76
Ontology Editor, 252
Ontology Browser, 255

Open Reading Frame, 122
Open Reading Frame, 126
Operons (Gene), 60, 67
Oracle server mode
Parameters
RDBMS-Database-Name, 8
Oracle server mode
Parameters, 8
RDBMS-Password, 8
RDBMS-Server-Hostname, 8
RDBMS-Server-Port, 8
RDBMS-Username, 8
ORF \t See Open Reading Frame, 122
Organism Set (Genome Browser), 72
Organism Summary Display, 19
organism-params.dat file (PathoLogic), 190
Organization Editor, 266
ORGID, 12, 13, 16, 128, 294, 296
Orphaned reaction? (Reaction Editor), 229
Orthologs (Gene), 72
Other Experimental Data, 43, 44
Output (PathoLogic), 120
Output file (Export Settings), 103
Overview, 36, 43, 45
Animation, 50
Cellular Overview
Metabolite Tracing, 50
Cellular Overview
Reachability Analysis, 51
Color Scales, 46
Displaying Reactions Corresponding to a Set of Genes, 38
Highlight, 45
Clear All, 45
Legend, 50
Omics Dataset File Format, 45
Omics Viewer, 47
Overview Key, 50
Regulatory Overview
Load Regulatory Overview from File, 41
Preferences for Regulatory Overview, 41
Redisplay Highlighted Genes Only, 41
Show Complete Regulatory Overview, 40
Regulatory Overview
Multi-regulon, 38
Regulatory Overview
Highlight Genes and Regulatory Relationships, 41
Load Regulatory Overview from File, 43
Preferences for Regulatory Overview, 39
Save Current Regulatory Overview to File, 41
Zoom Regulatory Overview, 41
Show Overview, 94
Clone, 93, 94
Usage, 47
Overview Menu
Show Key, 33
Overview Menu
Highlight, 33
Overview Menu
Show Overview, 33
Overview Menu
Show Overview, 91
Overview Display (Preferences), 81
Overview Menu
Highlight
Clear All, 89
Compound(s), 34
Gene, 34
Load from File, 35
Redo, 34, 89
Save to File, 35
Species Comparison, 33
Undo, 34, 88
Omics Viewer, 31
Overlay Experimental Data from, 47

Overlay Experimental Data from, 31
Print as Poster, 35
Show Key, 88
Show/Hide Transport Links, 33
Submenu (on Right-Button menu)
Reaction Menu, 35
Submenu (on Right-Button menu)
Compound Menu, 35
Pathway Menu, 35
Zoom, 36–38
Zoom Menu, 35
Update, 35
Overviews, 31

Packaging the database (Publishing Database), 111
Pane (Tools Menu), 75
Pane Layout (Preferences), 81, 92
Parallel Comparison (Comparative Analyses), 91
Paralogous Groups (Gene), 63
Password protection, 15
PathoLogic, xi, 117, 118
Assign Evidence Score, 141, 143
Assign Modified Proteins, 156
Assign to Reaction(s), 150
Batch Mode, 12, 189
organism-params.dat file, 190
Build
Automated Build, 136
Pathway/Genome Database, 136
Trial Parse, 134
Conditional Pathway Exclusion, 142, 143
Create genetic-elements.dat File, 132
Create New Organism, 130
Create Protein Complexes, 153
Creating Pathway/Genome Database, 118, 129
Database Generation Perspective, 118
Enzymatic Reactions, 119
enzyme-mappings.dat file, 138
Feature Table, 124
File Format, 121
DBLink, 122
EC, 122
Endbase, 121, 122
Function, 122
Function-Comment, 123
Function-Synonym, 123
Genbank Qualifiers, 124, 126
Gene-Comment, 123
GO, 122
ID, 121
Intron, 123
Name, 121
Product-ID, 123
Product-Type, 122
Startbase, 121
Synonym, 122
Filter, 150
GenBank File Format (PathoLogic), 124
Gene Ontology, 127
Genetic Element, 119
Input, 119
Input File Format, 120
FASTA file, 119, 121
fna file suffix, 121
gbk file suffix, 121
GenBank file format, 121
Genetic Element, 119
genetic-elements.dat, 119, 120, 132
MetaCyc, 119
pf file suffix, 121
Input Project Information Window, 128
Invoke PathoLogic, 129
KB
Revert, 149
Save, 149
Leave Unconnected, 157
local-enzyme-mappings.dat file, 139
Matching of Enzymes to Reactions, 138
Name-matching outcome, 139
Ambiguous Matches, 139
No Match, 139

Probable Metabolic Enzymes, 139
Name-matching Tool, 119
Ongoing PGDB Curation, 192
Operation, 119
Organism
Backup DB to File, 148
Convert File DB to Oracle or MySQL DB, 148
Create New, 130
New Version, 148
Reinitialize DB, 145
Revert KB, 136
Save KB, 136
Select Organism, 145
Specify Reference PGDB(s), 132
Output, 120, 185
Pathway/Genome Database, 185
Summary Pages, 185, 186
Pathway Hole Filler, 173
Algorithm Overview, 173
Candidates to fill pathway hole, 179
Candidates to fill pathway hole, 178, 179
Prediction, 174
Training, 173, 176
PGDB Directory Structure, 128
PGDB Housekeeping Tasks, 145
Protein-complex Building Tool, 153, 154
Re-Run Name Matcher, 152
Refine
Assign Probable Enzymes, 140, 148
Create Protein Complexes, 153
Pathway Hole Filler, 175
Predict Transcription Units, 157, 164
Re-Run Name Matcher, 141
Run Consistency Checker, 185
Transport Inference Parser, 160
Update Overview, 185
Refining the PGDB, 148
Rescore Pathways, 152
Specify Reference PGDB, 132
Split Function Name into Multiple Functions, 150
Suggested PGDB Release Procedures, 194
Transport Inference, 161
Assign an energy coupling to transporter, 162
Construct enzymatic reaction linking reaction with transport protein, 165
Construct full reaction, 164
Filter candidates, 161
Find candidate transporter proteins, 161
Group into complexes, 164
Identify any co-substrate, 162
Identify substrate(s), 162
Identify the compartment of each substrate, 163
Identify transporters that consist of subunits, 164
Transport Inference Parser
Reviewing and Modifying Results, 165
Transporter Inference Parser
Annotations, 169
Individual transporter dialog, 166
Main display, 165
Repeated Invocations, 168
Report, 172
Trial Parse, 134
PathoLogic (Tools Menu), 75
Pathway Editor
Submenu (on Right-Button menu), 221
Pathway Display (Preferences), 82
Pathway Editor, 216, 218, 219
Exit Menu, 219
Limitations, 221
Pathway Menu, 220
Add Subpathway by Class, 221
Add Subpathway by Name, 220
Add Subpathway by Substring, 220
Delete Subpathway, 221

Disconnect All Reactions, 220
Enter a Linear Pathway Segment, 220
Guess Pathway Predecessor List, 220
Invoke Relationships Editor, 220
Reaction Menu, 219
Add Connection, 219
Add Reaction, 219
Add Reaction (s) from History, 219
Choose Main Compounds for Reaction, 220
Clone a Reaction Frame, 219
Create New Reaction Frame, 219
Delete Predecessor/Successor Link, 219
Delete Reaction from Pathway, 220
Disconnect Reaction, 220
Edit Reaction Frame, 220
Submenu (on Right-Button menu)
Add Link from/to Pathway, 221
Assign Polymerization Name, 221
Create Polymerization Link, 221
Delete Polymerization Link, 221
Place this Compound at Cycle Top, 221
Pathway Hole Filler (PathoLogic), 173, 175
Pathway Info Editor, 201, 216, 217
Pathway Menu, 57
Enter a linear pathway segment (Pathway), 246
New, 201, 217, 245
Search by Class, 23, 57
Search by Name or Frame ID, 57, 92
Search by Species, 57
Search by Substrates, 57
Search by Substring, 57, 90
Pathway Menu (Pathway Editor), 220
Pathway Tools, xi, 1, 3
Configurations
BioCyc Configuration, xi
Full Pathway Tools Configuration, xi
Customization, 6
Exit, 10, 80
Init File, 6
Invoking, 5, 9
Pathway/Genome Navigator, 17
ptools-init.dat, 6
Remote Login, 9
Running Pathway Tools from the Command Line, 9
User Preferences, 80
Version, 319
Pathway/Genome Database, xi
Pathway/Genome Editors, xi
Pathway/Genome Editors \t See Editors, 197
Pathway/Genome Navigator, xi, 10, 17
Menus and Dialogs, 18
Aborting out of Menus and Dialogs , 18
Dialogs, 18
Menu Bar, 18
Multiple Choice, 18
Single Choice, 18
Pathways, 54, 203, 353
Command Buttons, 57
Enzyme View, 57
More Detail/Less Detail, 57
Diagrams, 54
Drawings, 55, 57
Pathway Display, 55
Gene Position, 56
Pathway Display, 55–57
Pathway-layout Algorithms, 55
Reaction Arrows, 55
Subpathways, 55
Superpathways, 55
Pathways from File... (Import/Export), 99
Pathway Tools
Exit, 294
PDGB Directory Structure (PathoLogic), 128
PerlCyc, 11
pf file suffix (PathoLogic Input File Format), 121

PGDB Housekeeping Tasks (PathoLogic), 145
Phantom-genes (Genome Browser), 67
Place this Compound at Cycle Top (Pathway Editor), 221
Plasmid (Genome Browser), 66
Predict Transcription Units (PathoLogic), 157, 164
Prediction (Pathway Hole Filler), 174
Preferences, 75, 80, 91, 93
  Answer List, 83
  Citation Reference Style, 29
  Citation Reference Style, 81
  Color, 81
  Database Sharing, 84
  Ecocyc-prefs File, 80
  General, 84
  History, 83
  Init File, 6
  Overview Display, 81
  Pane Layout, 81
  Pathway Display, 82
    Font size for mains, 82
    Font size for sides and enzymes, 82
    Layout for linear pathways, 82
    Reaction Arrow Emphasis, 82
    Show enzyme names, 82
    Show names when structures are shown, 82
    Show pathway graph only, without title or text, 83
    Show side compounds, 82
    Show side structures, 82
    Show structures for main compounds, 82
  Reaction Display, 83
  Reverting and Saving User Preferences, 84
  Site-Wide Configuration, 6
  Text Font Size, 81
Preferences for Regulatory Overview (Regulatory Overview), 41
Preferences (Tools Menu), 75
Print
  File Menu, 74
  Frame (Submenu), 253
  Genome Poster, 68
  Metabolic Map Poster, 35
  Pathway and its enzymes (Submenu), 253
  Pathway or reaction frames (Submenu), 253
Print (File Menu), 251
Print Poster (Chromosome Menu), 68
Probable Metabolic Enzymes (PathoLogic Outcome), 139
Product-ID (PathoLogic File Format), 123
Product-Type (PathoLogic File Format), 122
Propagate MetaCyc Data Updates, 192
Prosthetic Groups (Enzyme), 59
Protein, 59
  Activators, 55, 59, 62
  Amino Acid Sequence, 62
  Cofactors, 59, 62
  Inhibitors, 55, 59, 62
  Modified Proteins, 156, 215, 263
  Protein Sequence, 62
  Protein Display, 59
  Subunits, 59, 60
  Unmodified Proteins, 342, 343
  Unmodified Proteins, 29, 156, 157, 263
Protein Editor, 200, 210, 212, 213, 215
  Edit Protein Feature(s), 215
Protein Menu, 60
  New, 200
  Search by Name or Frame ID, 60
  Search by Pathway, 61
  Search by Substring, 23, 60
  Search by SwissProt ID (Protein), 61
  Search by Weight, pI (Protein), 62
  Search for Enzyme by Modulation (Protein), 62
Protein Sequence (Gene), 64
Protein Sequence (Protein), 62

Protein Subunit Structure Editor, 212
Protein-complex Building Tool (PathoLogic), 153, 154
Proteomics Data, 44
Pseudo-genes (Genome Browser), 67
ptools-init.dat, 6
Publication Editor, 202, 242
  Creating New Publication Frames, 242
  Editing Existing Publication Frames, 243
Publish PGDBs (Tools Menu), 78
Publishing
  Database Sharing, 110

Queries
  Complex Queries, 26
  Direct Queries, 24
  Indirect Queries, 25
  Object Displays and Queries, 27
    Citations and Comments, 29
    Classes, 30
    Database Links, 30
    Gene-Reaction Schematic, 28
    Shared Display Characteristics, 28
  Programmatic Queries, 26
  Queries with Multiple Answers, 25
  Query Facilities, 23, 89
Quick Entry of Equation (Reaction Editor), 228

RDBMS-Database-Name, 8
RDBMS-Password, 8
RDBMS-Server-Hostname, 8
RDBMS-Server-Port, 8
RDBMS-Username, 8
Re-Run Name Matcher (PathoLogic), 141, 152
Reachability Analysis (Overview), 51
Reaction Editor, 201
Reaction Arrows (Pathways), 55
Reaction Menu, 18
Reaction Arrow Emphasis (Pathway Display Preferences), 82
Reaction Display, 58
  Preferences, 83
Reaction Editor, 215, 225
  Binding Reactions, 225
  Chemical Reactions, 225
  Complex Processes, 226
  Transport Reactions, 226
  Unknown Conversions, 227
Reaction Flux Data, 44
Reaction Menu, 58
  New, 215, 225
  New (Reactions), 201
  Search by Class, 58
  Search by EC#, 22, 58
  Search by Name or Frame ID (Reactions), 58
  Search by Pathway, 58
  Search by Substrates, 58
  Search by Substring, 58
Reaction Menu (Pathway Editor), 219
Reaction Mode, 17
Reactions, 57
Recommendations, 264
Redisplay Highlighted Genes Only (Regulatory Overview), 41
Redo (Overview Highlight), 89
Redo (Overview Highlight), 34
Redox Half Reactions, 230
Reference Pathway, 141
Refining the PGDB (PathoLogic), 148
Refresh All Open DBs (File Menu), 250
Refresh object display (Submenu), 253
Registering (Publishing Database), 111
Registering a PGDB, 114
Regulation Editor, 241
Regulatory Interaction Editor, 200, 211
Regulatory Overview, 38
  Commands, 40
Reinitialize DB (PathoLogic), 145
Relationship Links, 263
Relationships Editor, 252
Release Notes, 296

Reload? (SAM Output File), 47
Remove Reaction, 215
Repeated Invocations (Transporter Inference Parser), 168
Replace (Substrate), 229
Replace Entered cpd with X (Compound Duplicate Checker), 240
Replicon column, 21
Replicons (Gene), 72
Reporting Problems, 319
Rerun initial setup (Database Sharing), 113
Rescore Pathways (PathoLogic), 152
Restore Defaults (Preferences), 84
Restore Saved Preferences (Preferences), 84
Restore Updates from Checkpoint File (File Menu), 250, 251
Retrieve Saved Color Scheme Parameters (SAM Output File), 50
Revert (PathoLogic), 149
Revert Current DB (File Menu), 249
Revert Current DB (File Menu), 106, 202, 250
Revert Database (Consistency Checker), 80
Revert KB (PathoLogic), 136
Reverting and Saving User Preferences (Preferences), 84
RNA, 62
  RNA Display, 62
RNA Editor, 200, 212
RNA Menu, 62
  New, 212
  New (RNA), 200
  Search by Class, 62
  Search by Name or Frame ID, 62
  Search by Substring, 62
Run Consistency Checker (PathoLogic), 185

SAM Output File (Omics Viewer), 31, 46, 47
Save
  Transporter Inference Parser, 168
Save (Preferences), 84
Save Color Scheme Parameters (SAM Output File), 50
Save Current DB (File Menu), 202, 251
Save Current Regulatory Overview to File (Regulatory Overview), 41
Save Database (Consistency Checker), 80
Save DB, 248
Save KB (PathoLogic), 136
Save to File (Overview Highlight), 35
Save(PathoLogic), 149
Saving DB Updates (Advanced Editing Topic), 247
SBML export (Import/Export), 99
Search (Substrate), 229
Search by Class
  Compound Menu, 65
  Gene, 63
  Pathway Menu, 57
  Reactions, 58
  RNA, 62
Search by EC#
  Reactions, 58
Search by Name or Frame ID
  Compound Menu, 64
  Gene, 63
  Pathway, 57, 92
  Protein, 60
  Reactions, 58
  RNA, 62
Search by Pathway
  Protein, 61
  Reactions, 58
Search by SMILES substructure (Compound Menu), 64
Search by Species
  Pathway, 57
Search by Substrates
  Pathway, 57
  Reactions, 58
Search by Substring
  Compound Menu), 64
  Gene, 63
  Pathway, 57, 90
  Protein, 60

Reactions, 58
RNA, 62
Search by Substring (Gene), 244
Search by SwissProt ID
Protein, 61
Search by Weight, pI (Protein), 62
Search for Enzyme by Modulation (Protein), 62
Segment Editor, 216, 222
Segment Editor (Pathway), 217
Select & Browse Chromosome/Replicon (Chromosome Menu), 68
Select & Browse Chromosome/Replicon (Genome Browser), 66
Select All (Comparative Analyses), 88
Select Organism (PathoLogic), 145
Selected Chromosome to Genbank File... (Import/Export), 99
Selected Frames to File (Import/Export), 100, 101
Selected Gene (Genome Browser), 67
Selected Pathways to File... (Import/Export), 98
Selected Reactions to SBML File... (Import/Export), 99
Sequence File
Add or Replace, 68
Setting Preferences (Database Sharing), 112
Shared (Comparative Analyses), 87
Shared Display Characteristics, 28
Show (Button), 157
Show (Compound Duplicate Checker), 239
Show (Overview Submenu), 36, 37
Show changes (Submenu), 253
Show Complete Regulatory Overview (Regulatory Overview), 40
Show compound/reaction/pathway in overview (Submenu), 253
Show enzyme names (Pathway Display Preferences), 82
Show enzymes and genes of this reaction in listener window (Overview Submenu), 36
Show frame (Submenu), 253
Show frame in all DBs (Submenu), 253
Show frame in other species (Submenu), 253
Show frame name (Submenu), 253
Show History, 80
Show History (Consistency Checker), 80
Show Key (Overview), 88
Show Key (Overview), 33
Show names when structures are shown (Pathway Display Preferences), 82
Show object in the navigator (Consistency Checker), 80
Show on Console, 26
Show Overview, 33
Show Overview (Overview), 91, 94
Show pathway graph only, without title or text (Pathway Display Preferences), 83
Show Sequence of a Segment of Chromosome (Chromosome Menu), 68
Show side compounds (Pathway Display Preferences), 82
Show side structures (Pathway Display Preferences), 82
Show structures for main compounds (Pathway Display Preferences), 82
Show Submenu (on Right-Button menu), 253
Show/Hide Transport Links (Overview), 33
Significance Analysis of Microarrays \t See SAM Output File (Omics Viewer), 46
Single Organism Display, 20
Skip (Button), 154, 156
Slots, 203
Comment (Slot), 123, 125
sockets, 11
Software Patches
Activate Installed Patches, 75
Software Patches, 75

Source of frames to be exported (Export Settings), 101
Special Formatting of Text, 256
Species Comparison Dialog Box (Comparative Analyses), 88
Species Comparison (Overview), 33
Species comparisons (Comparative Analyses), 88
Specify color value cutoffs, assign colors automatically (SAM Output File), 49
Specify color value cutoffs, assign colors manually (SAM Output File), 49
Specify Reference PGDB (PathoLogic), 132
Split Function Name into Multiple Functions (PathoLogic), 150
Spontaneous? (Reaction Editor), 229
Start and end base-pair positions (Genome Browser), 67
Start Site (Transcription Unit), 65
Startbase (PathoLogic File Format), 121
Stop Codon (Gene), 63
Stop (Button), 156
Structure Editing (JME), 237
Structure Editing (Marvin), 233
Submenu (on Right-Button menu), 252
  Add to History, 253
  Edit
    Add Reaction(s), 214
    Compound Editor, 201, 232
    Copy frame name to Clipboard, 252
    Create Binding Interaction, 200, 210
    Create Frame, 252
    Create/Add Enzyme, 200, 201, 214, 245
    Delete Frame, 252
    Export Object-Type to DB..., 252
    Frame Editor, 252
    Gene Editor, 199
    Intron Editor, 199, 211
    JME Chemical Structure Editor, 237
    JME Compound Structure Editor, 201
    Marvin Chemical Structure Editor, 201
    Object-Type Editor, 252
    Ontology Editor, 252
    Pathway Editor, 217
    Pathway Info Editor, 201, 217
    Protein Editor, 200
    Reaction Editor, 201, 225
    Regulatory Interaction Editor, 200
    Regulatory Interaction Editor, 211
    Relationships Editor, 252
    Remove Reaction, 215
    RNA Editor, 200, 212
    Synonyms Editor, 252
    Terminators, 200
    Transcription Unit Editor, 199
  Editors, 198
  Notes, 253
  Publication Data, 202
  Show, 33, 90
    Print frame, 253
    Print pathway and its enzymes, 253
    Print pathway or reaction frames, 253
    Refresh object display, 253
    Show changes, 253
    Show compound/reaction/pathwayin overview, 253
    Show frame, 253
    Show frame in all DBs, 253
    Show frame in other species, 253
    Show frame name, 253
Subpathways, 337
Subpathways (Pathways), 55
Substrate Specificity (Enzyme), 59
Subunits (Protein), 59
Subunits (Protein), 60
Suggested PGDB Release Procedures (PathoLogic), 194
Summaries
  Bold face, 256
  Centering, 256

Greek letters, 256
Italics, 256
Subscripts, 256
Superscripts, 256
Underlining, 256
Summarize Current Organism (File Menu), 250
Summarize Databases (File Menu), 250
Summarize Pathway Evidence, 21
Summary of Organisms, 186
Summary Pages (PathoLogic), 185, 186
Superimpose on previous display? (SAM Output File), 48
Superpathways, 55
Superpathways (Pathways), 55
Support-Email-Address, 6
Swiss-Prot, 61, 122
Synonym (PathoLogic File Format), 122
Synonym Editor, 252
Synonyms, 122

Taxonomic Hierarchies, 30
Terminator Editor, 209
Invoke Terminator Editor, 209
Terminators, 67, 200, 322
(Gene Page), 63
On Gene Page), 65
Text File (Omics Viewer), 31
Text Font Size (Preferences), 81
The Single Organism Display, 21
Tickmarks (Genome Browser), 66
TIP \t See Transporter Inference Parser, 160
Tools Menu, 75
Answer List, 76
Browse Downloadable PGDBs, 77
Consistency Checker, 78
Automatic Tasks, 78, 79
Invoke the object editor, 80
Manual Tasks, 78, 79
Revert Database (Consistency Checker), 80
Save Database, 80
Show History, 80
Show object in the navigator, 80
Credits, 264
History, 26, 75
Show on Console, 26
Instant Patch, 75
Download and Activate All Patches, 75
Ontology Browser, 76
Pane, 75
PathoLogic, 75, 129
Preferences, 75
Pane Layout, 92
Restore Defaults, 84
Restore Saved Preferences, 84
Save, 84
Text Font Size, 81
Prepare Blast Reference Data, 76
Publish PGDBs, 78
Search
Curators, 266
Organizations, 266
Upgrade Schema of Current DB, 78
Tools Menu
Reachability Analysis, 52
Top Menu Bar
Customization, 299
Training (Pathway Hole Filler), 173, 176
Transcription Units, 348
Transcription Units (Gene), 60, 65
Transcription Direction (Genome Browser), 67
Transcription Factor (Transcription Unit), 65
Transcription Unit Editor, 199, 207, 208
Invoking Transcription Unit Editor, 208
Transcription Units, 65
Modify, 199
Transcription Units (Gene), 63, 65, 67, 157
Transport Inference Parser
Exit, 168
Saving, 168

Transport Inference Parser (PathoLogic), 160, 161
Reviewing and Modifying Results (PathoLogic), 165
Transport Pathways, 117
Transport Reactions (Reaction Editor), 226
Trial Parse (PathoLogic), 134
Troubleshooting, 319
Troubleshooting (Web Server Operation), 317
TUs \t See Transcription Units, 157
Type of data (SAM Output File), 48
Type of display (SAM Output File), 48

Undo (Overview Highlight), 34, 88
Unfix (Button), 75
Unification Links, 30, 263
Unknown Conversions (Reaction Editor), 227
Unmodified Form, 29, 263, 342, 343
Update Overview (PathoLogic), 35, 185
Update slots (Import Dialog), 105
Updating for New Versions of MetaCyc, 192
Upgrade Schema of Current DB (Tools Menu), 78
Usage (Overview), 47
Use data from (SAM Output File), 48
Use expression data from data column (SAM Output File), 49
User access control, 12, 15
User Preferences, 29
Citation Reference Style, 29
User-Account-RDBMS-Database-Name, 8
User-Account-RDBMS-Password, 9
User-Account-RDBMS-Server-Hostname, 8
User-Account-RDBMS-Server-Port, 8
User-Account-RDBMS-Username, 9
Username for storing to the FTP server above (Database Sharing), 113

Version, 128
Pathway Tools, 11
Viewing
Experimental Data, 31

Web Accounts, 306
Web Server Log File, 303
Web Server Mode
Additional Parameters, 6
Support-Email-Address, 6
WWW-Publish, 7
WWW-Server-Proxy-Port, 7
WWW-Server-User, 7, 8
Mandatory Parameters, 6
WWW-Server-Hostname, 6
WWW-Server-Port, 6
User Accounts Parameters
User-Account-RDBMS-Database-Name, 8
User-Account-RDBMS-Password, 9
User-Account-RDBMS-Server-Hostname, 8
User-Account-RDBMS-Server-Port, 8
User-Account-RDBMS-Username, 9
User Accounts RDBMS Parameters, 8
Web Server Operation, 10
Creating URLs to Pathway Tools Pages, 304
Web Server Log File, 303
Web Server Mode
Additional Parameters
WWW-Html-Root-Dir, 7
WWW-Use-Gzip, 7
Web Server Operation, 293, 294
BLAST
Setting up access, 303
Customizing Web Server Pages, 296
Customizing Web Server Pages
Add HTML text, 297
Include a small organism graphic, 297
Link to a page of release notes, 296
Style sheet, 297
Operational Procedures, 295
Troubleshooting, 317
WWW-Html-Root-Dir, 7
WWW-Publish, 7

WWW-Server-Hostname, 6
WWW-Server-Port, 6
WWW-Server-Proxy-Port, 7
WWW-Server-User, 7, 8
WWW-Use-Gzip, 7

X Windows, 5, 9, 10
X Windows, 294

Zoom (Genome Browser), 67
Zoom (Overview Submenu), 36–38
Zoom Regulatory Overview (Regulatory Overview), 41