Pathway Tools Schema and Semantic Inference Layer: Pathways and the Overview

Outline

- **Pathways**
  - Representation of Pathways
  - Querying Pathways Programmatically
  - How Pathway Diagrams are Generated
  - Future Work: Signalling Pathways
- **Cellular Overview Diagram**
  - New Functionality
  - Under the Hood
  - How Overview Diagram is Generated
  - Using Overview Diagram for Global Queries

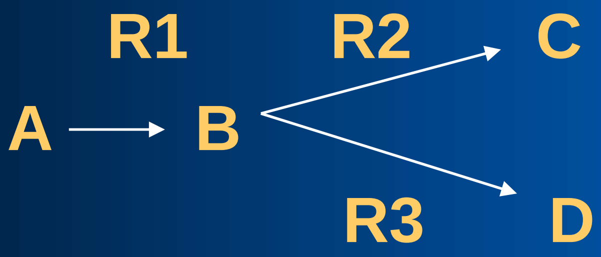
What is a pathway?

- **An ordered set of interconnected, directed biochemical reactions**
- **Reactions form a coherent unit, e.g.**
  - Regulated as a single unit
  - Conserved across organisms as a single unit
  - When combined, perform a single cellular function
  - Historically grouped together as a unit
- **Includes metabolic pathways and signalling pathways**
- **Evidence for all reactions in a single organism**
- **Pathways can be linear, cyclical, branched, or some combination**

Internal Representation of Pathways

- **REACTION-LIST: unordered list of reactions that comprise the pathway**
- **PREDECESSORS: list of pairs that define ordering relationship between pathways.**

E.g.



R1 R2 C

A B

R3 D

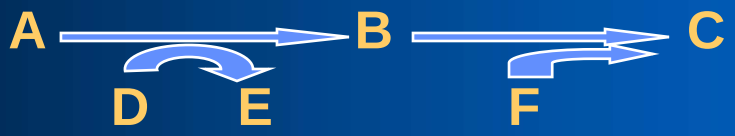
(R2 R1) : Predecessor of R2 is R1

(R3 R1) : Predecessor of R3 is R1

(R1) : R1 has no predecessor (can be omitted)

What is missing from Pathway Representation?

- **Reaction directions**
  - Some reactions are unidirectional, but many are reversible – how do we know in which direction to draw the reaction?
- **Main vs. side substrates**



  - Main compounds form the backbone of the pathway
    - substrates shared between connecting reactions
    - major inputs and outputs.
  - Side compounds omitted from pathway diagrams at low detail levels
  - Individual reactions do not necessarily have main and side compounds – a particular substrate may be either a main or a side depending on the pathway context.

Computing Directionality and Mains/Sides

Our philosophy: Enable curator to specify as little as possible. Compute as much as possible. This reduces redundancy and potential for inconsistencies.

Example:

Reactions R1: A + B ⇔ C + D

R2: B ⇔ E

Predecessors: (R2 R1)

- **Only substrate overlap is B**
- **B must be a main substrate**
- **A must be a side substrate,**
- **R1 must proceed from right to left**
- **R2 must proceed from left to right**

$$\begin{tikzcd}
C + D \arrow[r] & B \arrow[r] & E \\
& A \arrow[ur] &
\end{tikzcd}$$

But…

Unfortunately, mains, sides and reaction directions are sometimes ambiguous:

- **At beginnings and ends of pathways**
  - Use heuristics to determine main/side substrates at beginnings, ends of pathways
  - Not always what the curator wants
- **Substrate overlap with both sides of a reaction,**

  e.g. A + B ⇔ C + D

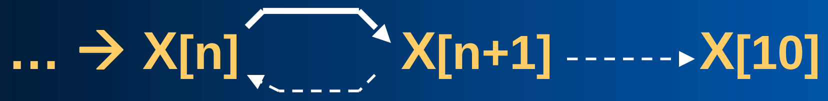
  C + B ⇔ E
- **Solution: Additional slot PRIMARIES, should only be populated when necessary:**

  PRIMARIES: (R (A B) (C)) says that for reaction R, A and B are both main reactants, and C is a main product.

Even More Complications…

- **ENZYME-USE: a reaction may be catalyzed by multiple enzymes, but not all the enzymes may participate in a given pathway**
  - Not present in the same compartment with rest of pathway enzymes
  - Down-regulated or not expressed under conditions in which pathway is active
  - ENZYME-USE slot tells us which enzymes catalyze reaction in pathway, if not all.
- **LAYOUT-ADVICE: helps software draw pathway correctly, e.g. in a cyclical pathway, tells which substrate should be at the top.**
- **HYPOTHETICAL-REACTIONS: list of reactions in the pathway that are considered hypothetical (i.e. no direct experimental evidence)**

Polymerization Pathways



… → X[n] X[n+1] X[10]

- **POLYMERIZATION-LINKS: specifies reactions which should be connected by a polymerization link**

  (X R1 R1) --- REACTANT-NAME-SLOT: N-NAME

  --- PRODUCT-NAME-SLOT: N+1-NAME

- **CLASS-INSTANCE-LINKS: specifies when a link should be drawn between a substrate class and some instance of it (necessary only if instance is not a member of some reaction, so no predecessor relationship can be defined)**

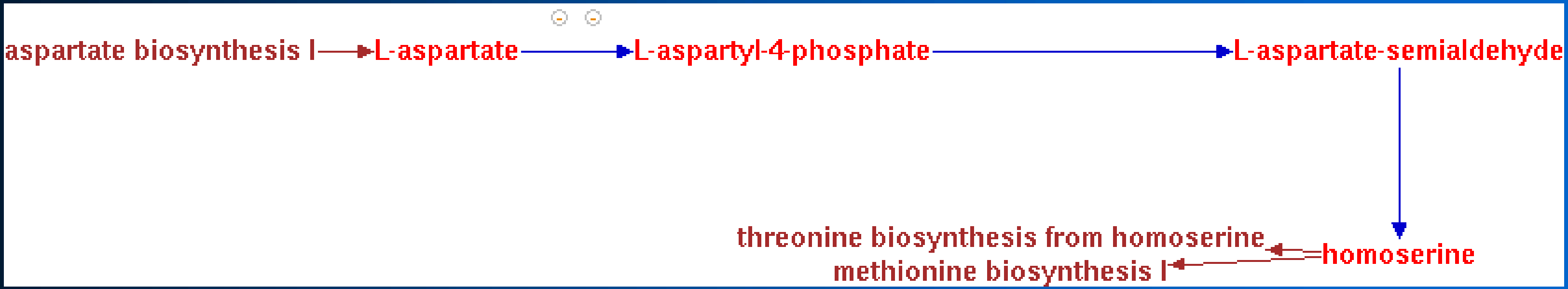
  R1 --- PRODUCT-INSTANCES: X[10]

Super-pathways

- Collection of pathways that connect to each other via common substrates or reactions, or as part of some larger logical unit
- Can contain both sub-pathways and additional connecting reactions
- Can be nested arbitrarily
- REACTION-LIST: a pathway ID instead of a reaction ID in this slot means include all reactions from the specified pathway
- PREDECESSORS: a pathway ID instead of a tuple in this slot means include all predecessor tuples from the specified pathway

Pathway Links

- Can be used as an alternative or in addition to defining super-pathways
- Link must be to or from some main substrate in the pathway
- Other end of link can be a pathway, a reaction, or an arbitrary text string
- Software automatically computes direction of link, but curator can override it



Querying Pathways Programmatically

- **See http://bioinformatics.ai.sri.com/ptools/ptools-resources.html**
- **(all-pathways)**
- **(base-pathways)**
  - Returns list of all pathways that are not super-pathways
- **(genes-of-pathway pwy)**
- **(unique-genes-of-pathway pwy)**
  - Returns list of all genes of a pathway that are not also part of other pathways
- **(enzymes-of-pathway pwy)**
- **(compounds-of-pathway pwy)**
- **(variants-of-pathway pwy)**
  - Returns all pathways in the same variant class as a pathway
- **(get-predecessors rxn pwy), (get-successors rxn pwy)**
- **(get-rxn-direction-in-pathway pwy rxn)**
- **(pathway-inputs pwy), (pathway-outputs pwy)**
  - Returns all compounds consumed (produced) but not produced (consumed) by pathway (ignores stoichiometry)

Example Queries

- **Find all genes involved in metabolic pathways:**

```
(remove-duplicates
   (loop for p in (all-pathways)
             append (genes-of-pathway p)))
```

- **Find all compounds that are unique to a single pathway:**

```
(loop for p in (base-pathways)
    append
       (loop for c in (compounds-of-pathway p)
                when (null (remove p (pathways-of-compound c)))
                collect (list c p)))
```

Why Automated Pathway Layout?

- **Pros:**
  - Less effort for curators to generate/edit pathways
  - No need to store coordinates or other graphical information in database
  - When data changes (i.e. new enzyme added, reaction substrates changed slightly, substrate or enzyme name changed), diagram updates automatically
  - Can show at arbitrary and different levels of detail and/or magnification without having to regenerate diagram
- **Cons:**
  - Curators have less control over how pathway looks – can be very hard or impossible to fix a pathway when the software displays it incorrectly
  - Pathways can be made much more compact when laid out manually

Grasper-CL

- Graph program developed at SRI in 80’s-90’s
- A single graph, called a space, contains nodes, edges
- Nodes: can have icon, label
- Edges: can have label, arrowhead, knot points
- Appearance of both nodes and edges is fully customizable – font, line style, color, shape, size, label placement, etc., either individually or using defined styles
- Arbitrary data values can be attached to both nodes and edges, as well as to space as a whole
- Extensible: can write programs to define new customizations, e.g. new icon shape for chemical structure.
- Includes toolbox of layout algorithms, e.g. tree, circle, array
- Spaces can be defined hierarchically, i.e. a group of nodes in one space can be grouped into a single supernode in another, and manipulated as a group

Why are Biochemical Pathways Hard to Lay Out Automatically?

- **Biologists have definite expectations about how they want things to look**
- **Side substrates have to be positioned specially**
- **Reactions (edges) have auxiliary information that must be placed next to the edge, but is not connected to any other node**
- **Node names (substrates, enzymes) are often very long**
- **Arbitrary topology**
- **No existing general graph layout algorithm handles all these complexities and produces graphs that would be pleasing to biologists, who are accustomed to textbook diagrams**

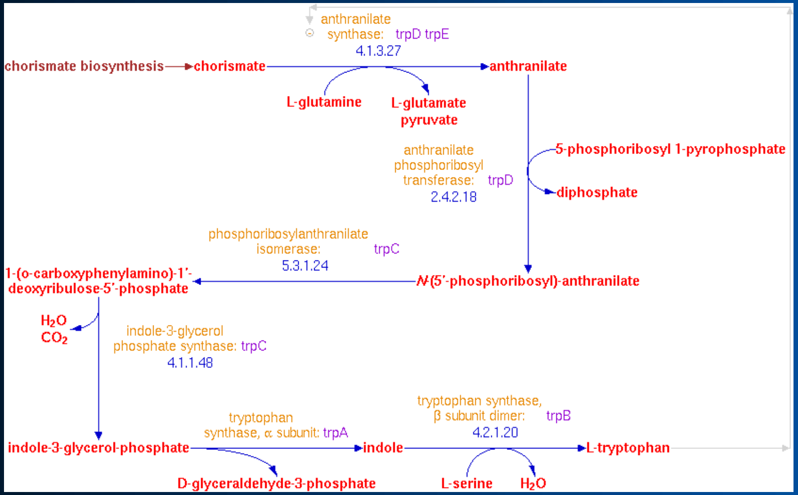
Our Pathway Layout Algorithm

- Create nodes for every main substrate
- Create edges between main nodes for every reaction
- Create nodes for side substrates, enzymes, etc. – associate these with a reaction edge, but do not create any edges connecting them
- Compute topology of main nodes and edges
- Compute extra space required for side nodes
- Apply a standard graph layout algorithm to main nodes, leaving space for sides/enzymes
- Position side/enzyme nodes (and curved arrows) after the fact, add any necessary knot points to reaction edges

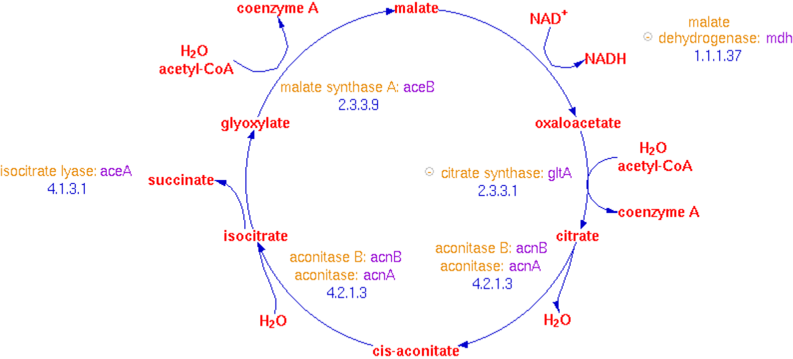
Standard Graph Layout Algorithms

- **Linear pathways: use horizontal, vertical, or “snake” layout algorithm**
- **Branched pathways: use tree layout algorithm**
- **Cycles: use circular layout algorithm**
- **Combination pathways: use a hierarchical layout algorithm that combines above algorithms:**
  - Find largest cycle in graph
  - Determine and lay out nodes (if any) that should be drawn inside circle
  - Use circular algorithm to lay out cycle around inside nodes
  - Divide outside nodes into connected components, and lay out each according to its topology
  - Position outside components relative to connecting nodes on the circle

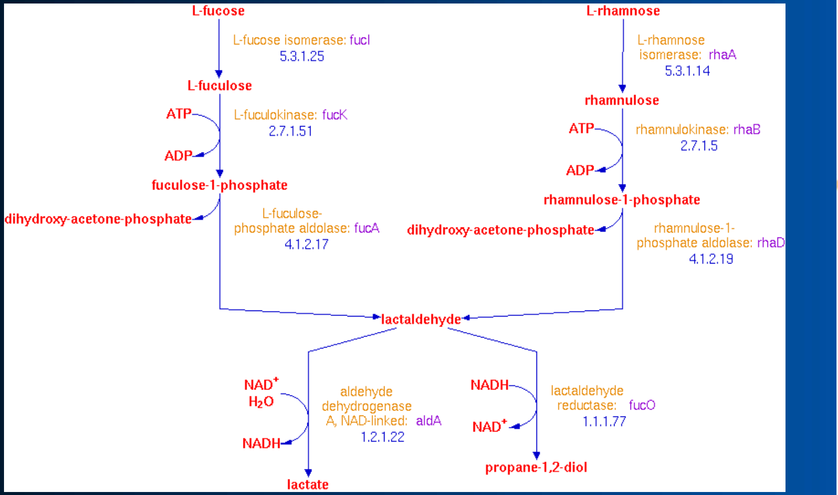
Examples



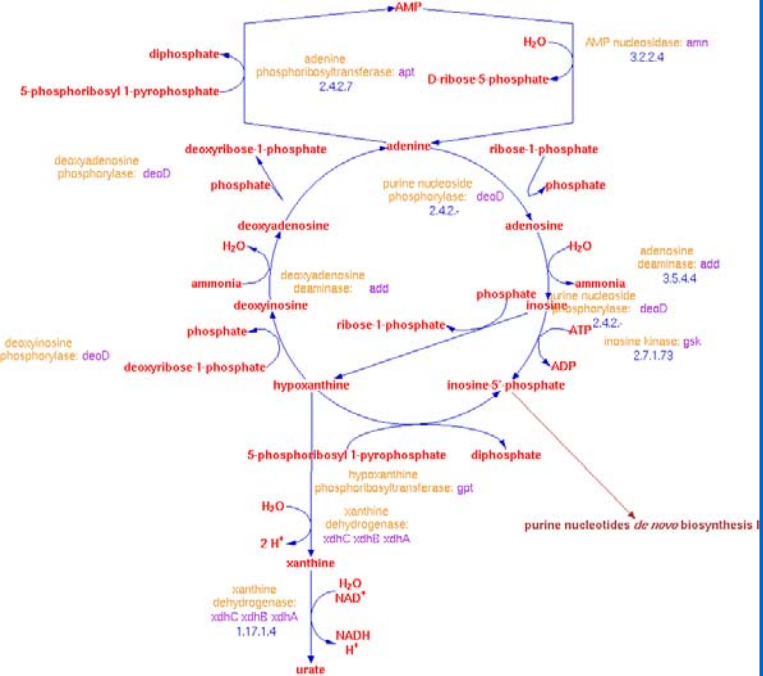
anthranilate synthase: trpD trpE
4.1.3.27
chorismate biosynthesis
chorismate
anthranilate
L-glutamine
L-glutamate
pyruvate
anthranilate phosphoribosyl transferase: trpD
2.4.2.18
5-phosphoribosyl 1-pyrophosphate
diphosphate
phosphoribosylanthranilate isomerase: trpC
5.3.1.24
1-(o-carboxyphenylamino)-1'-deoxyribulose-5'-phosphate
N-(5'-phosphoribosyl)-anthranilate
H_2O
CO_2
indole-3-glycerol phosphate synthase: trpC
4.1.1.48
tryptophan synthase, α subunit: trpA
tryptophan synthase, β subunit dimer: trpB
4.2.1.20
indole-3-glycerol-phosphate
indole
L-tryptophan
D-glyceraldehyde-3-phosphate
L-serine
H_2O



coenzyme A
malate
NAD^+
NADH
malate dehydrogenase: mdh
1.1.1.37
H_2O
acetyl-CoA
malate synthase A: aceB
2.3.3.9
glyoxylate
oxaloacetate
H_2O
acetyl-CoA
isocitrate lyase: aceA
4.1.3.1
succinate
citrate synthase: gltA
2.3.3.1
coenzyme A
isocitrate
citrate
aconitase B: acnB
aconitase: acnA
4.2.1.3
aconitase B: acnB
aconitase: acnA
4.2.1.3
H_2O
H_2O
cis-aconitate



L-fucose
L-fucose isomerase: fucI
5.3.1.25
L-fuculose
ATP
ADP
L-fuculokinase: fucK
2.7.1.51
fuculose-1-phosphate
dihydroxy-acetone-phosphate
L-fuculose-phosphate aldolase: fucA
4.1.2.17
L-rhamnose
L-rhamnose isomerase: rhaA
5.3.1.14
rhamnulose
ATP
ADP
rhamnulokinase: rhaB
2.7.1.5
rhamnulose-1-phosphate
dihydroxy-acetone-phosphate
rhamnulose-1-phosphate aldolase: rhaD
4.1.2.19
lactaldehyde
NAD^+
H_2O
NADH
aldehyde dehydrogenase A, NAD-linked: aldA
1.2.1.22
lactate
NADH
NAD^+
lactaldehyde reductase: fucO
1.1.1.77
propane-1,2-diol



AMP
diphosphate
5-phosphoribosyl 1-pyrophosphate
adenine phosphoribosyltransferase: apt
2.4.2.7
H_2O
D-ribose-5-phosphate
AMP nucleosidase: amn
3.2.2.4
adenine
deoxyribose-1-phosphate
ribose-1-phosphate
deoxyadenosine phosphorylase: deoD
phosphate
purine nucleoside phosphorylase: deoD
2.4.2.-
phosphate
deoxyadenosine
adenosine
H_2O
ammonia
deoxyadenosine deaminase: add
H_2O
ammonia
adenosine deaminase: add
3.5.4.4
deoxyinosine
phosphate
inosine
purine nucleoside phosphorylase: deoD
2.4.2.-
ribose-1-phosphate
phosphate
ATP
ADP
inosine kinase: gsk
2.7.1.73
deoxyinosine phosphorylase: deoD
deoxyribose-1-phosphate
hypoxanthine
inosine-5'-phosphate
5-phosphoribosyl 1-pyrophosphate
diphosphate
hypoxanthine phosphoribosyltransferase: gpt
H_2O
2 H^+
xanthine dehydrogenase: xdhC xdhB xdhA
purine nucleotides de novo biosynthesis
xanthine
xanthine dehydrogenase: xdhC xdhB xdhA
1.17.1.4
H_2O
NAD^+
NADH
H^+
urate

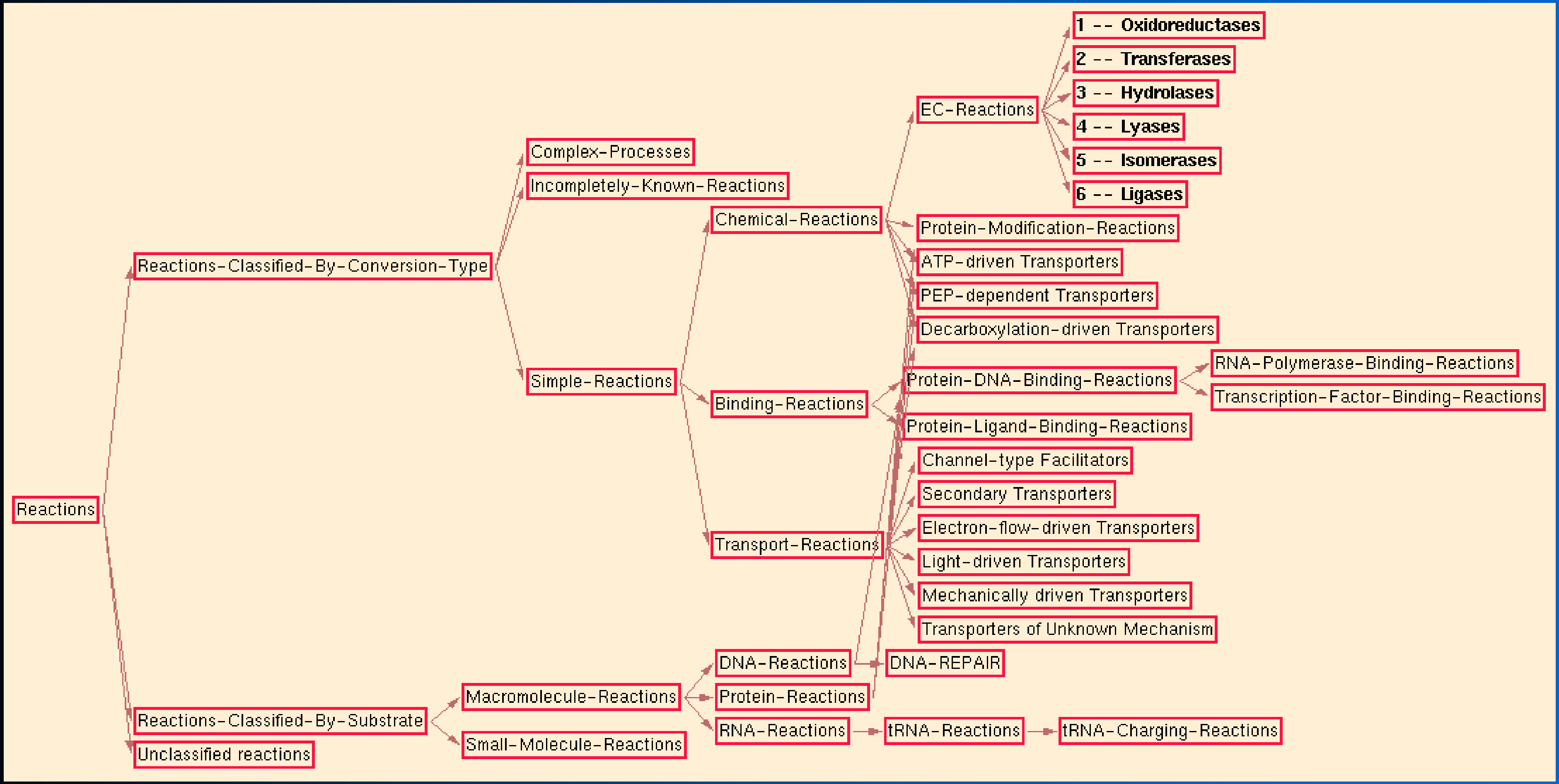
Problems with Pathway Layouts

- **Complicated pathways, particularly those that use the tree layout algorithm but have several off-tree edges, or highly interconnected pathways, give us trouble:**
  - Edge crossings
  - Sides/Enzymes can overlap with other nodes
  - Pathway can “blow up” and become very spread out
- **Can’t have connections to side substrates**
- **Limited toolbox of pathway algorithms**

Signalling Pathways

- Need to extend our representation to handle complexities of signalling pathways
- Pathways will need to include traditional enzyme-catalyzed reactions, transport, protein binding and modification reactions, and possibly larger processes, e.g. transcription, protein degradation
- Automated layout beyond the scope of our current algorithms

First Step: Reorganizing Reaction Ontology



1 -- Oxidoreductases

2 -- Transferases

3 -- Hydrolases

EC-Reactions

4 -- Lyases

5 -- Isomerases

Complex-Processes

6 -- Ligases

Incompletely-Known-Reactions

Chemical-Reactions

Protein-Modification-Reactions

Reactions-Classified-By-Conversion-Type

ATP-driven Transporters

PEP-dependent Transporters

Decarboxylation-driven Transporters

RNA-Polymerase-Binding-Reactions

Simple-Reactions

Protein-DNA-Binding-Reactions

Transcription-Factor-Binding-Reactions

Binding-Reactions

Protein-Ligand-Binding-Reactions

Channel-type Facilitators

Reactions

Secondary Transporters

Electron-flow-driven Transporters

Transport-Reactions

Light-driven Transporters

Mechanically driven Transporters

Transporters of Unknown Mechanism

DNA-Reactions

DNA-REPAIR

Macromolecule-Reactions

Protein-Reactions

Reactions-Classified-By-Substrate

RNA-Reactions

tRNA-Reactions

tRNA-Charging-Reactions

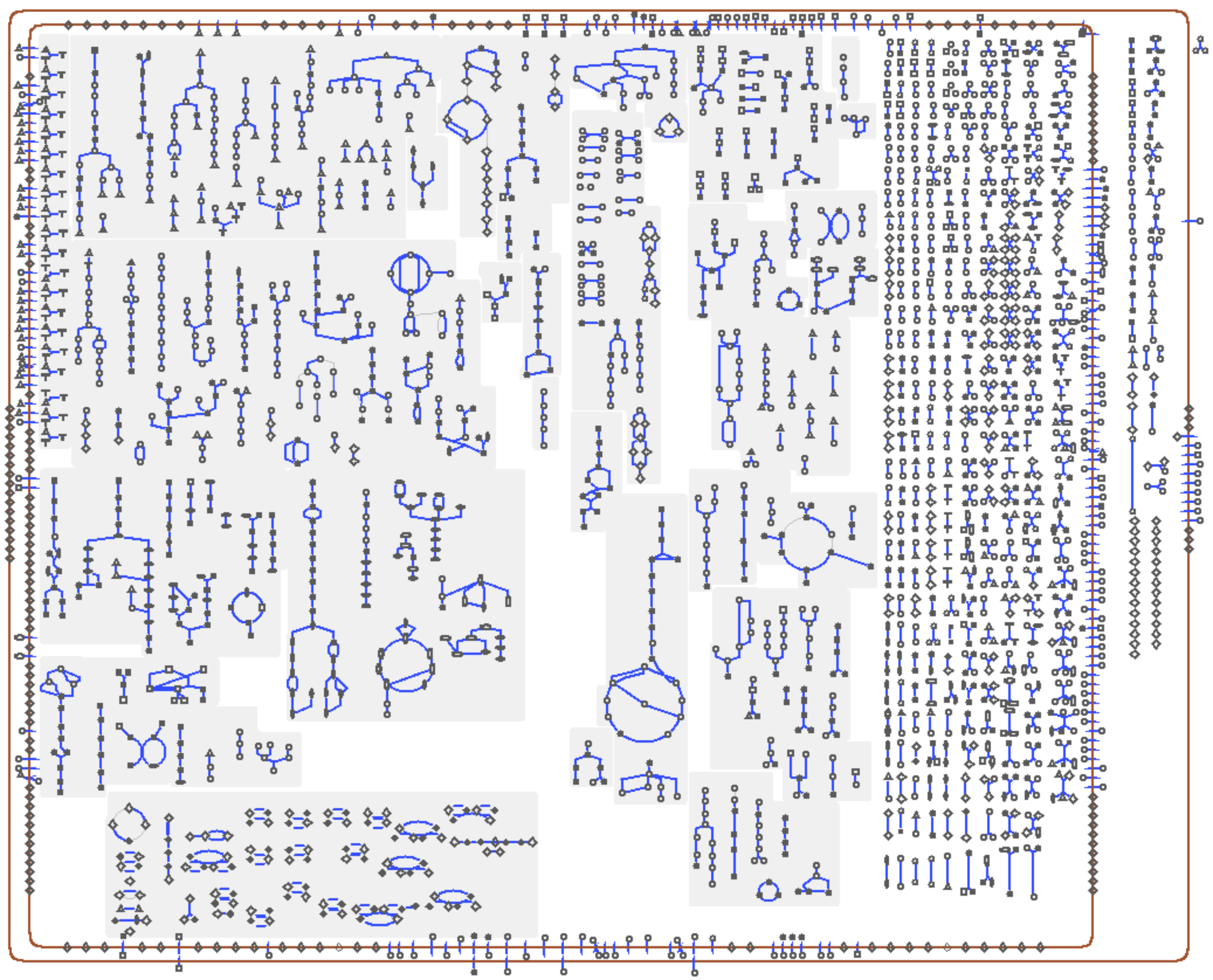
Small-Molecule-Reactions

Unclassified reactions

Next Steps

- Upgrade tool to convert current data to new ontology
- Automatic classifier to place reactions in proper class in new ontology

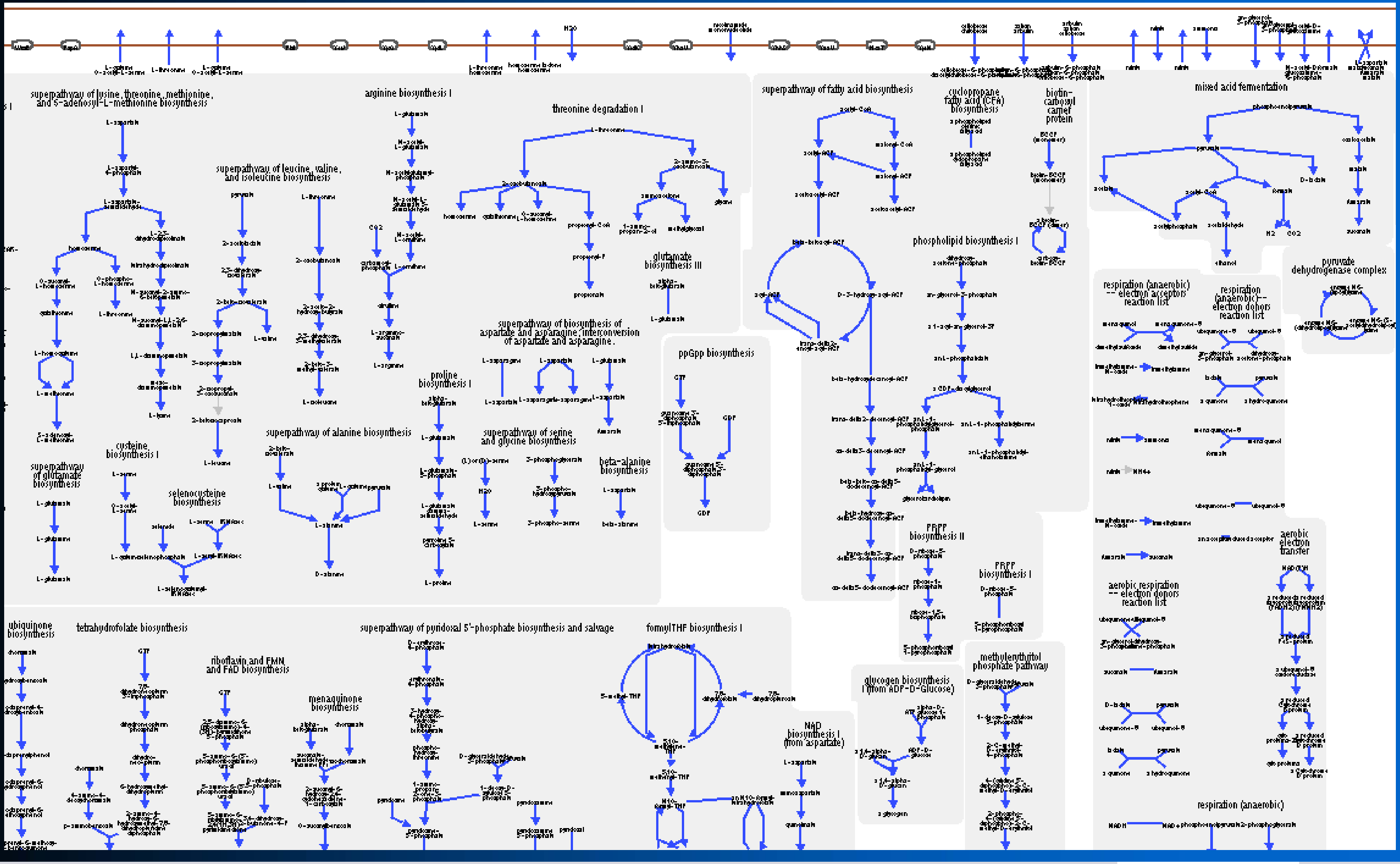
Cellular Overview Diagram



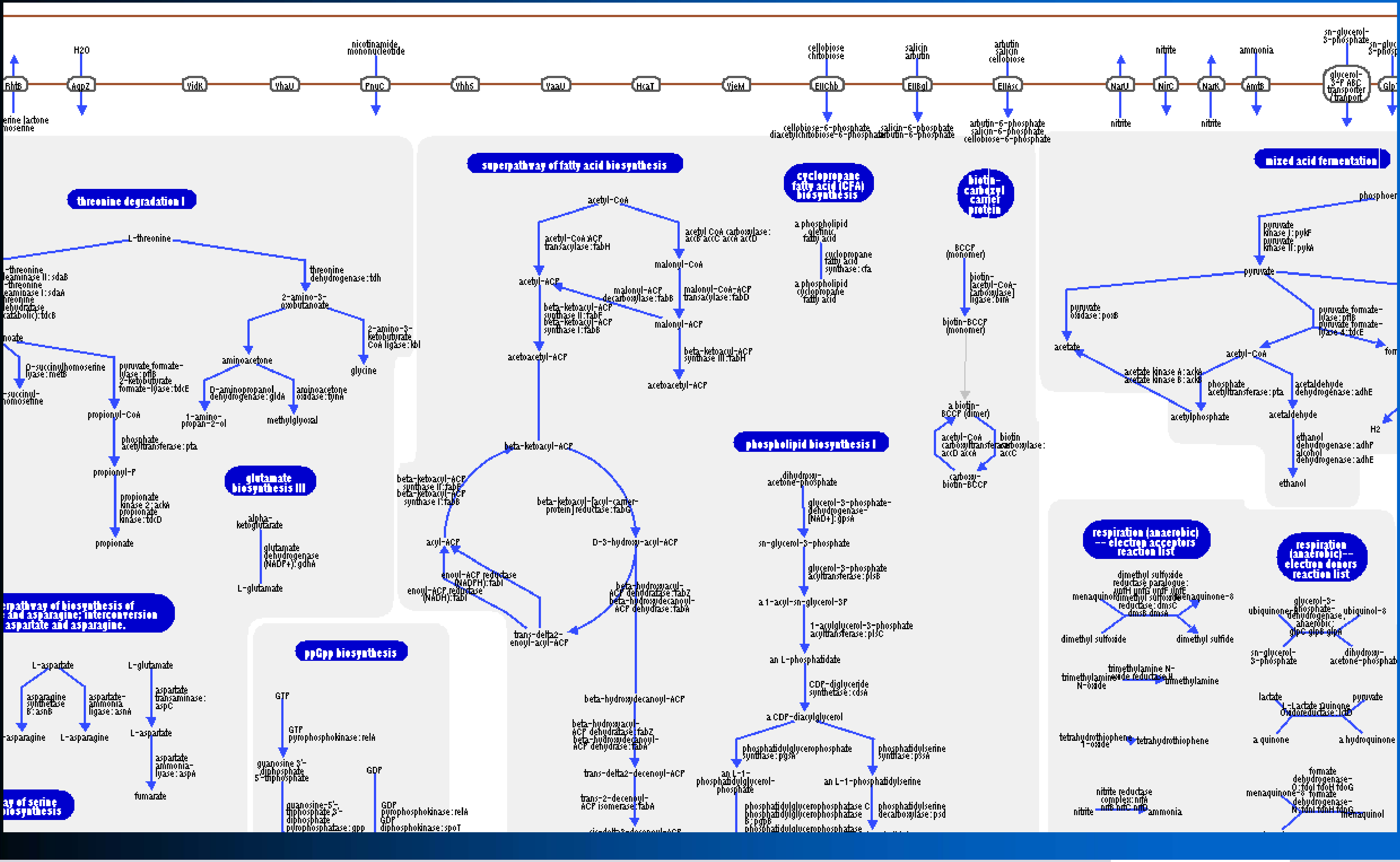
New Semantic Zooming Capabilities

- **Can enlarge overview diagram to show**
  - Arrowheads on reaction arrows (120%)
  - Substrate names and pathway labels (200%)
  - Enzyme, gene names (300%, but more readable at 400%)
  - At 400%, you have a diagram suitable for poster printing
- **Automatic poster printing facility**
  - Can customize title, text, highlighting, etc.
  - Can custom build overview specifically for poster
    - Include/exclude enzyme names, gene names, EC numbers
    - Change font sizes
    - Alter aspect ratio
- **Unfortunately, overview diagram now takes longer to generate (approx 1 hour vs. several minutes)**

Fragment of Overview at 200% Zoom


superpathway of lysine, threonine, methionine, and S-adenosyl-L-methionine biosynthesis
arginine biosynthesis I
threonine degradation I
superpathway of leucine, valine, and isoleucine biosynthesis
superpathway of fatty acid biosynthesis
cyclopropane fatty acid (CFA) biosynthesis
biotin-carboxyl carrier protein
mixed acid fermentation
phospholipid biosynthesis I
glutamate biosynthesis III
pyruvate dehydrogenase complex
respiration (anaerobic)-- electron acceptors reaction list
respiration (anaerobic)-- electron donors reaction list
superpathway of biosynthesis of aspartate and asparagine; interconversion of aspartate and asparagine.
ppGpp biosynthesis
proline biosynthesis I
superpathway of alanine biosynthesis
superpathway of serine and glycine biosynthesis
beta-alanine biosynthesis
cysteine biosynthesis I
superpathway of glutamate biosynthesis
selenocysteine biosynthesis
PRPP biosynthesis II
PRPP biosynthesis I
aerobic electron transfer
aerobic respiration -- electron donors reaction list
ubiquinone biosynthesis
tetrahydrofolate biosynthesis
superpathway of pyridoxal 5'-phosphate biosynthesis and salvage
formylTHF biosynthesis I
riboflavin and FMN and FAD biosynthesis
menaquinone biosynthesis
glucogen biosynthesis I (from ADP-D-Glucose)
methylerythritol phosphate pathway
NAD biosynthesis I (from aspartate)
respiration (anaerobic)

Fragment of Overview at 400% Zoom



Under the Hood of the Overview Diagram

- **Overview is a Grasper graph**
  - Substrates, proteins, pathway class boxes, and membranes are all nodes
  - Reactions are edges
  - Nodes and edges use defined sets of shape parameters, which can be changed when zoom level changes
- **Not generated dynamically, so does not update automatically when data changes. Use Overview →Update command to rebuild**
- **Diagram is not saved as part of PGDB, but in a separate file: xyzcyc/version/data/overview.graph**

How Overview Diagram is Generated

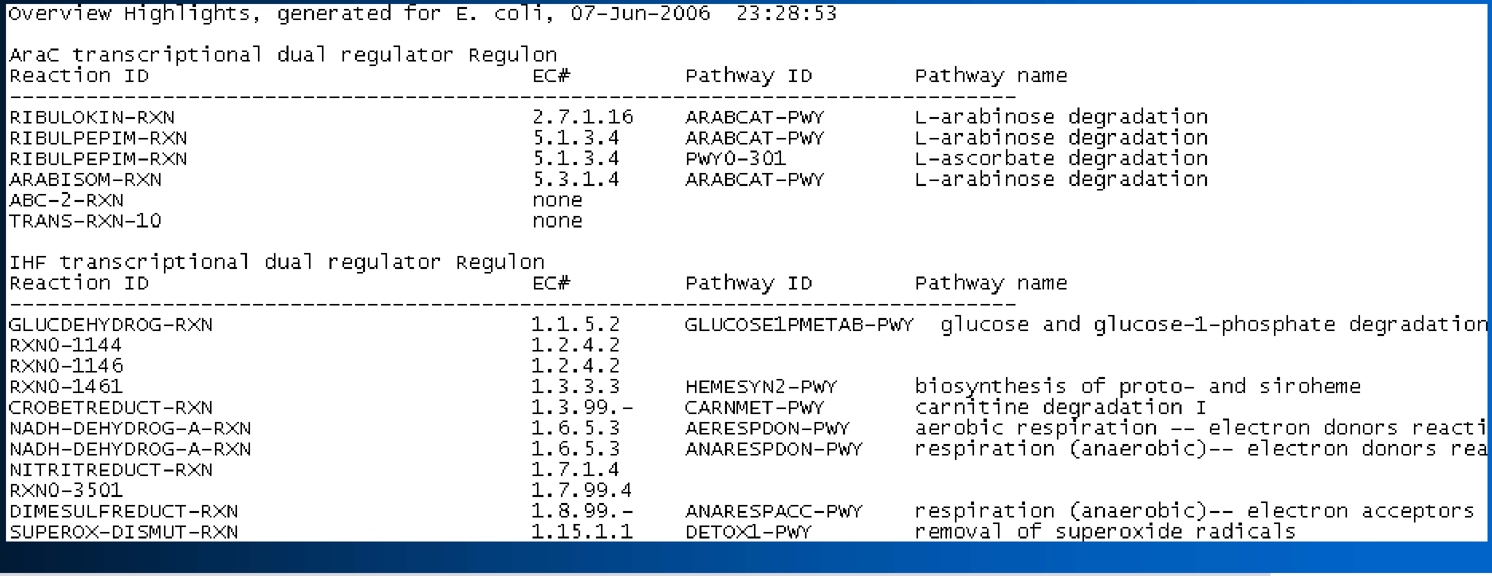
- **Hierarchical algorithm**
- **Space is apportioned into regions for biosynthetic, degradative, and energy metabolism pathways**
- **Each pathway is laid out using regular pathway layout algorithm**
- **All pathways in a single class (e.g. amino acid biosynthesis) are packed together as compactly as possible using simple greedy algorithm**
- **All classes in a top-level class (e.g. biosynthesis) are packed together using greedy algorithm**
- **Three top-level classes are positioned side by side**
- **Reaction “maze” is added to the right, signal transduction pathways at the bottom**
- **Membranes, transport reactions, membrane proteins, periplasmic and extracellular reactions are added around the outside**

Implications

- Overview is built from scratch each time
- Positions of pathways can change greatly from run to run or from organism to organism
- Can’t predict final dimensions of overview diagram until it is built

Using Overview Diagram for Global Queries

- Species Comparison
- Highlight list of genes or reactions from file
- Variety of “canned” queries
- See all connections from one or more selected metabolites
- API to highlight based on user computations
- Can save highlights to (& reload from) a human-readable file



```
Overview Highlights, generated for E. coli, 07-Jun-2006  23:28:53

AraC transcriptional dual regulator Regulon
Reaction ID                              EC#         Pathway ID        Pathway name
-------------------------------------------------------------------------------
RIBULOKIN-RXN                            2.7.1.16    ARABCAT-PWY       L-arabinose degradation
RIBULPEPIM-RXN                           5.1.3.4     ARABCAT-PWY       L-arabinose degradation
RIBULPEPIM-RXN                           5.1.3.4     PWY0-301          L-ascorbate degradation
ARABISOM-RXN                             5.3.1.4     ARABCAT-PWY       L-arabinose degradation
ABC-2-RXN                                none
TRANS-RXN-10                             none

IHF transcriptional dual regulator Regulon
Reaction ID                              EC#         Pathway ID        Pathway name
-------------------------------------------------------------------------------
GLUCDEHYDROG-RXN                         1.1.5.2     GLUCOSE1PMETAB-PWY  glucose and glucose-1-phosphate degradation
RXN0-1144                                1.2.4.2
RXN0-1146                                1.2.4.2
RXN0-1461                                1.3.3.3     HEMESYN2-PWY      biosynthesis of proto- and siroheme
CROBETREDUCT-RXN                         1.3.99.-    CARNMET-PWY       carnitine degradation I
NADH-DEHYDROG-A-RXN                      1.6.5.3     AERESPDON-PWY     aerobic respiration -- electron donors reacti
NADH-DEHYDROG-A-RXN                      1.6.5.3     ANARESPDON-PWY    respiration (anaerobic)-- electron donors rea
NITRITREDUCT-RXN                         1.7.1.4
RXN0-3501                                1.7.99.4
DIMESULFREDUCT-RXN                       1.8.99.-    ANARESPACC-PWY    respiration (anaerobic)-- electron acceptors
SUPEROX-DISMUT-RXN                       1.15.1.1    DETOX1-PWY        removal of superoxide radicals
```

Overview API

- (highlight-compounds ‘(cpd1 … cpdN) [:color color])
- (highlight-reactions ‘(rxn1 … rxnN) [:color color])
- (highlight-pathways ‘(pwy1 … pwyN) [:color color])
- (unhighlight-ov-all)

Examples

- **Highlight all amino acids (color chosen automatically by software)**

  ```
  (highlight-compounds (get-class-all-instances ‘|Amino-Acids|))
  ```

- **Highlight all reactions that appear in only one pathway in red**

  ```
  (highlight-reactions (loop for r in (all-rxns)
                             when (= (length (get-slot-values r ‘in-pathway)) 1)
                             collect r)
                       clim:+red+)
  ```

- **Highlight all pathways that produce a compound that is not involved in any other pathway. Define a color using rgb values.**

  ```
  (highlight-pathways
     (loop for p in (base-pathways)
           when (loop for c in (pathway-outputs p)
                      thereis (null (remove p (pathways-of-compound c))))
           collect p)
     (clim:make-rgb-color 0.2 0.7 0.8))
  ```

Using Omics Viewer for Global Analyses

- **Show gene expression, proteomics, metabolomics data**
- **Customizable color schemes**
- **Can superimpose results of multiple datasets on single display, or show as animation**
- **Can also be used to show results of global computational analyses – anything that assigns a number to a gene, protein, reaction or substrate, or subdivides them into groups**
- **Navigate from Omics Viewer to pathway displays to see omics data on a single pathway**