

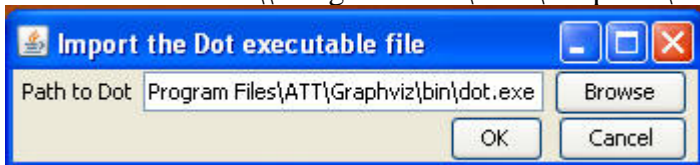
# Dot Layout

A graph layout algorithm must compute positions of the node and edge figures. Many graph layout algorithms are already implemented in Cytoscape. The Dot Layout plugin adds the dot layout from the Graphviz package developed at AT&T Research ([www.graphviz.org](http://www.graphviz.org)) to Cytoscape by encapsulating the dot code with a java handler. A brief description of the dot layout from the Graphviz website: “dot - makes hierarchical or layered drawings of directed graphs. The layout algorithm aims edges in the same direction (top to bottom, or left to right) and then attempts to avoid edge crossings and reduce edge length.” There are two types of layout that can be computed by Dot. One is Simple Dot Layout, and the other is Rank-Cluster Dot Layout. Download the sample data archive, *Dot-sample-data.zip* and unzip for all examples in this document.

## 1 Installing the Dot Plugin

- Download the dot.jar file and put it in the Cytoscape Plugin directory.
- Download and install the appropriate Graphviz package for your operating system from [www.graphviz.org](http://www.graphviz.org).
- Open Cytoscape. Before using the dot plugin for the first time, go to the menu item Layout and select the “Configure dot layout” option. This will set the path to the dot.exe application which should be in the following locations:

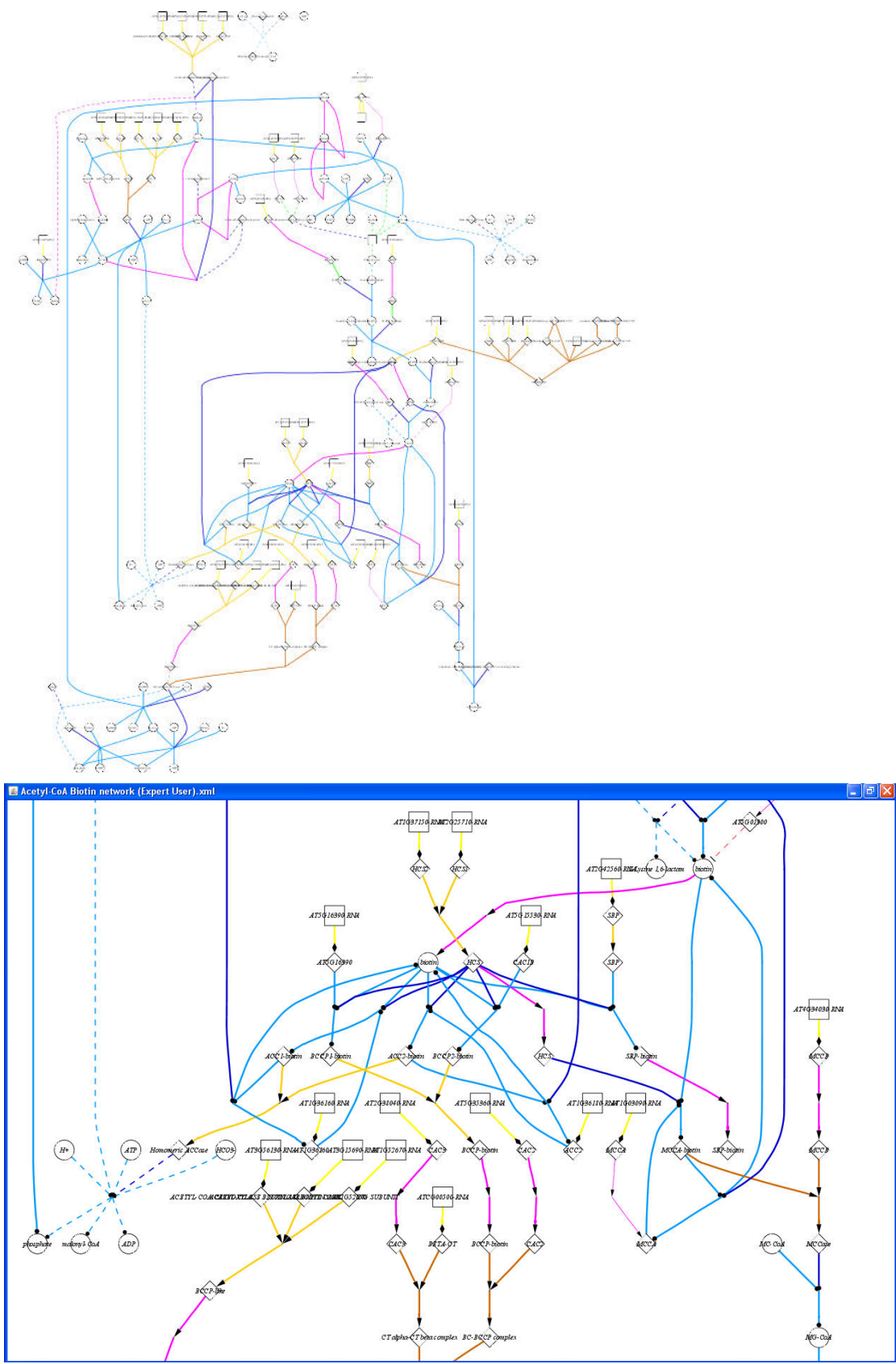
- Windows: c:\\Program Files\\ATT\\Graphviz\\bin\\dot.exe



- Mac - /opt/graphviz/bin/dot
- Linux – either /usr/bin/dot or /usr/local/bin/dot

## 2 Simple Dot Layout

The generic dot layout is called Simple Dot Layout. It can be selected from Layout Menu. Figure 1 shows a graph layout done using the simple dot layout. As previously quoted from the Graphviz website: “dot - makes hierarchical or layered drawings of directed graphs. The layout algorithm aims edges in the same direction (top to bottom, or left to right) and then attempts to avoid edge crossings and reduce edge length.” Note how the graph is laid out on distinct levels with few edge crossings. This layout is especially good for trees and pathways that are fairly linear. It also scales up well to large graphs that follow the small world edge distribution.



**Figure 1.** Example of the simple dot layout with an overview and the detail using the network file: Acetyl CoA Biotin network (Expert User).xml. This file is available in the MetNet sample data folder.

### 3 Rank-Cluster Dot Layout

It can be also used to compute a more customized layout. Based on their values of certain node attributes, the node figures can be placed on horizontal ranks or into clusters. To try this, load the Cytoscape session file, *Acetyl-CoAbiotin.cys* using the *File > Open* command. From the Layout Menu, select *Dot Rank-Cluster Layout*, then an open file dialog pops up to show in which a layout XML file must be selected (see Figure 2 below). Figure 3 below shows the same graph as in Figure 1 using the rank-cluster layout.

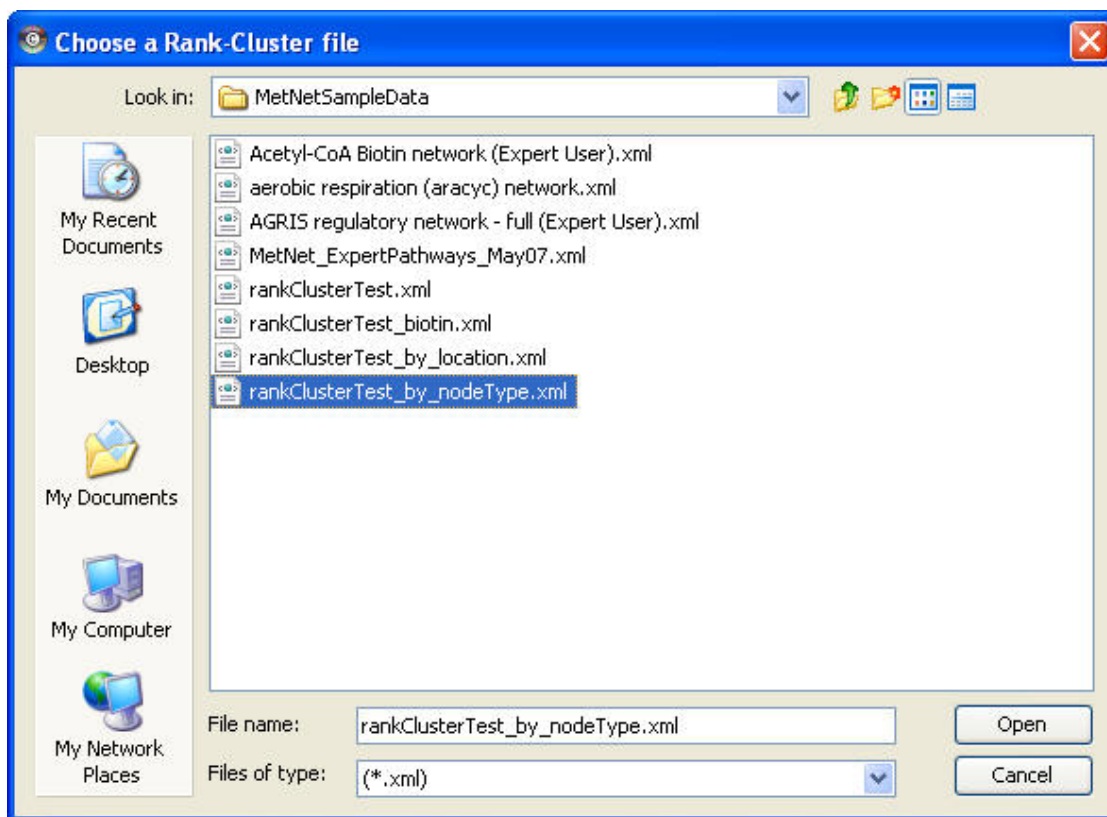


Figure 2 Dialog Box to choose a layout XML file

#### 3.1 Layout XML File and Node Attribute file

The Dot Rank-Cluster layout is based on node properties. Any categorical node property in the network can be selected. The ranks and clusters of dot layout are specified in a layout XML file based on node properties. A rank is a row in the graph window. The file structure consists of a file header that gives information on the Dot XML format and rank-cluster information for the specified graph.

##### (A) File header

```
<?xml version="1.0" standalone="yes"?>
<!DOCTYPE layout [
<!ELEMENT layout (rank*, cluster*)>
<!ELEMENT rank (atom*, composite*)>
<!ATTLIST rank
```

```

label CDATA #REQUIRED>
<!ELEMENT cluster (atom*, composite*, cluster*)>
<!ATTLIST cluster
label CDATA #REQUIRED>
<!ELEMENT atom EMPTY>
<!ATTLIST atom
property CDATA #REQUIRED
value CDATA #REQUIRED>
<!ELEMENT composite ((atom | composite), connective, (atom | composite))>
<!ELEMENT connective EMPTY>
<!ATTLIST connective
type (and | or) #REQUIRED>
]>

```

### **(B) Layout XML file for specifying ranks by Node Type**

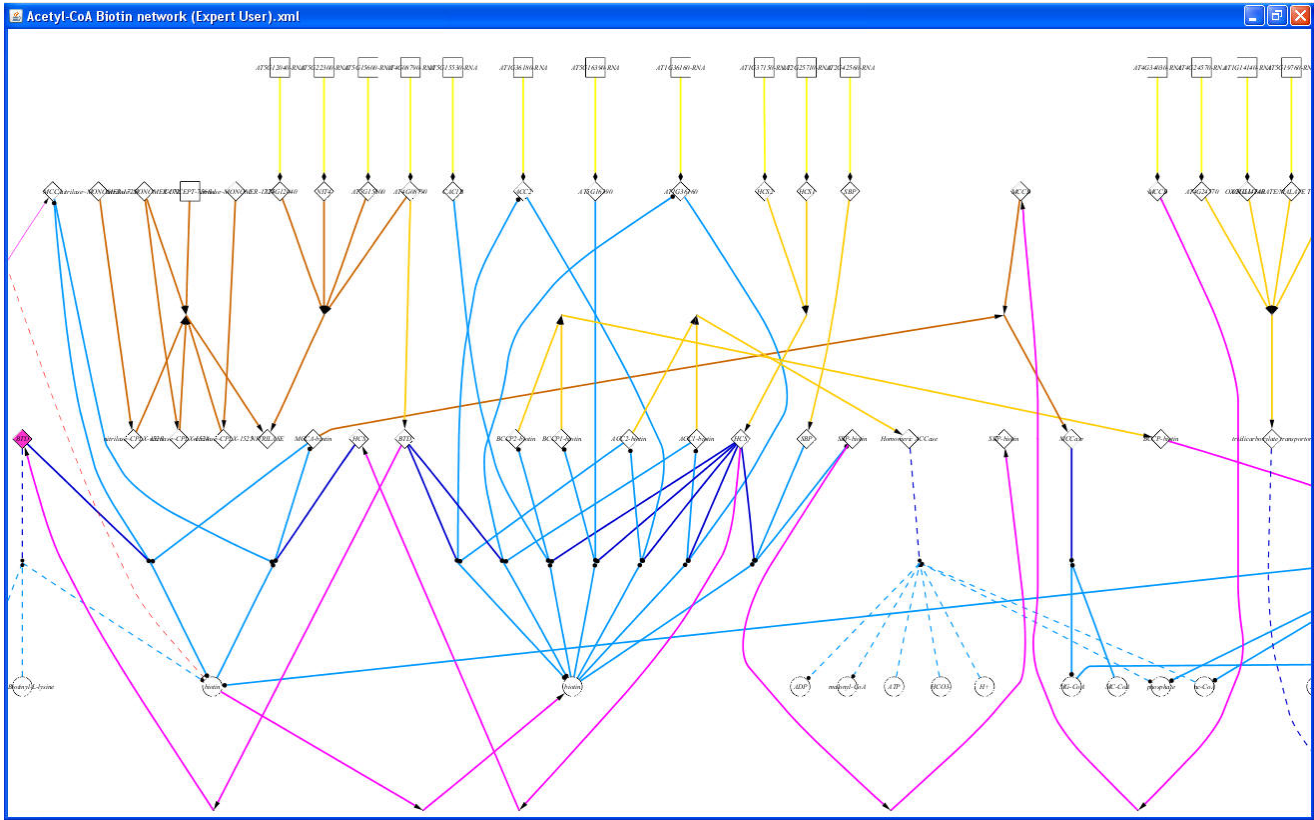
The <rank> tag is used to specify the nodes that should be placed in a certain horizontal rank. Each rank has a label, specified by the label attribute, which is shown on the left-hand side of the graph view. Nodes are selected using the same type of XML as in the property-to-visual-attribute mappings. Different node types can be combined in a single rank using the *composite* tags and *AND* and *OR* connectives. Figures 3 and 4 below show the aerobic pathway network divided into rank by node type and by location.

***Header information goes first.***

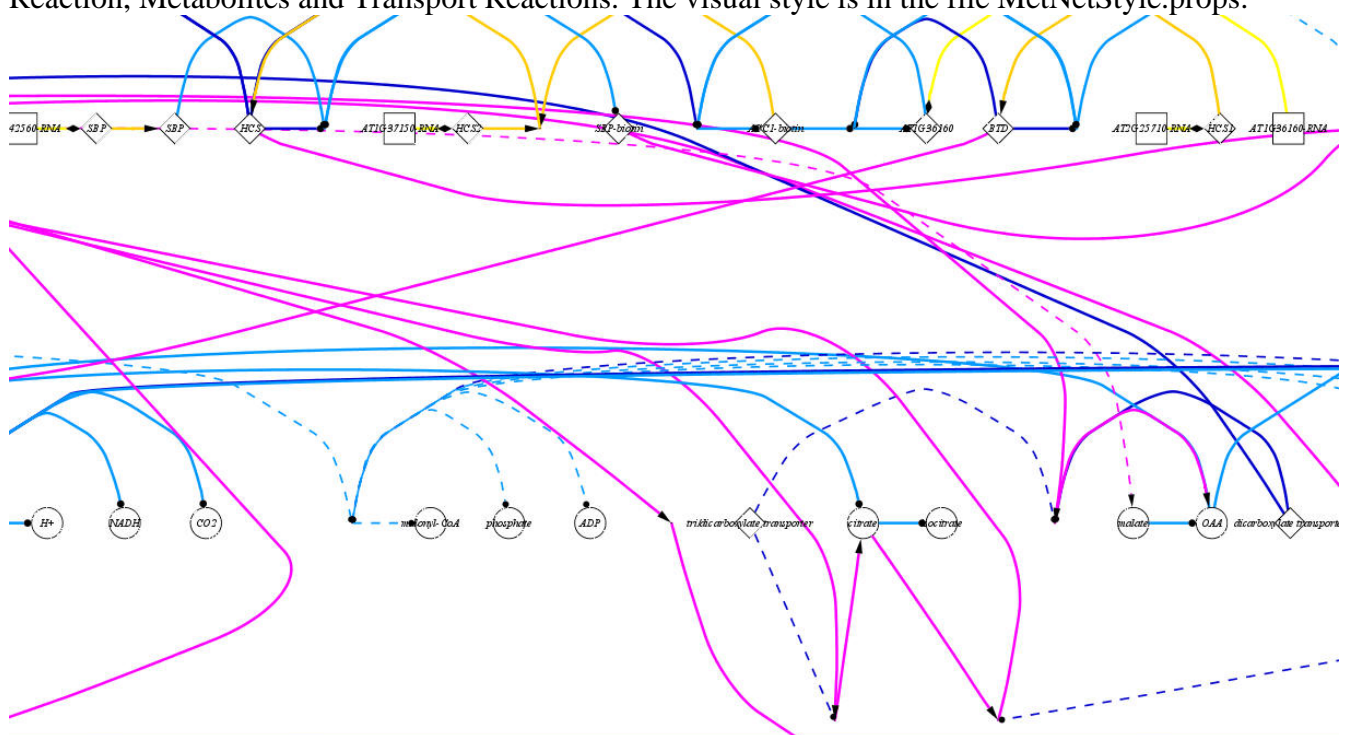
```

<layout>
<rank label="rank1">
  <atom property="nodeType" value="RNA"/>
</rank>
<rank label="rank2">
  <atom property="nodeType" value="polypeptide"/>
</rank>
<rank label="rank3">
  <composite>
    <atom property="nodeType" value="Composition-AND"/>
    <connective type="or"/>
    <atom property="nodeType" value="Composition-OR"/>
  </composite>
</rank>
<rank label="rank4">
  <atom property="nodeType" value="protein complex"/>
</rank>
<rank label="rank5">
  <atom property="nodeType" value="Enzymatic reaction"/>
</rank>
<rank label="rank6">
  <atom property="nodeType" value="metabolite"/>
</rank>
<rank label="rank7">
  <atom property="nodeType" value="Transport"/>
</rank>
</layout>

```



**Figure 3** Rank-cluster dot layout of the Acetyl CoA Biotin pathway based on node type using the layout file. The order of the levels is: RNA, Polypeptide, Reaction, Protein complex, Enzymatic Reaction, Metabolites and Transport Reactions. The visual style is in the file MetNetStyle.props.



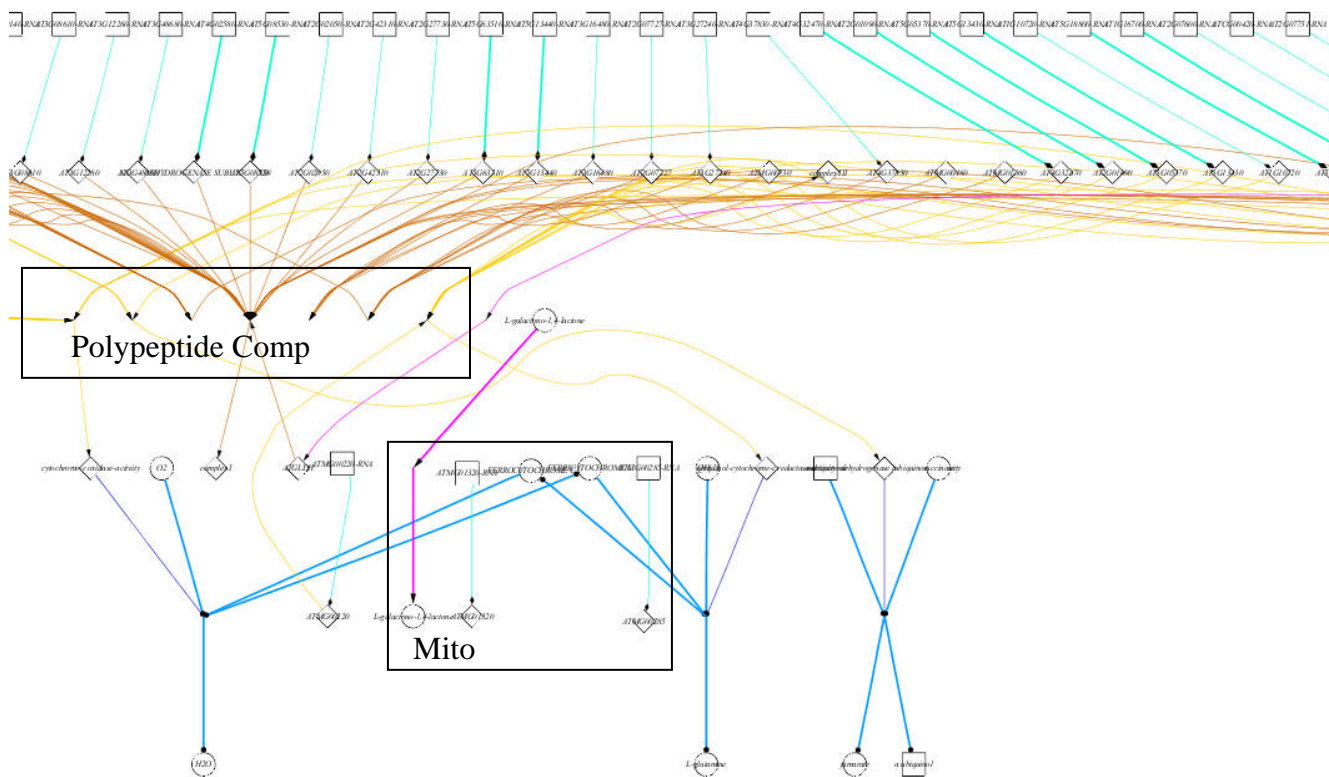
**Figure 4** Rank-cluster dot layout of the Acetyl CoA Biotin pathway based on cellular location using the layout file, rankClusterTest\_bylocation.xml. The order of the levels is: Cytosol, Mitochondrial, unknown, and plastid stroma. The visual style is in the file MetNetStyle.props.

### (C) Layout XML file for specifying clusters by Node Type

The <cluster> tag is used to specify nodes to place together in a cluster. The <cluster> tag uses the same node selection mechanism as the <rank> tag. <cluster> tags can be nested inside each other, creating nested clusters. Figure 5 below shows an example using clusters.

**Header information goes first.**

```
<layout>
<cluster label="Polypeptide Comb">
  <composite>
    <atom property="nodeType" value="Composition-AND" />
    <connective type="or" />
    <atom property="nodeType" value="Composition-OR" />
  </composite>
</cluster>
<cluster label="Mito">
  <atom property="location" value="mitochondrion" />
</cluster>
<cluster label="RNA">
  <atom property="nodeType" value="RNA" />
</cluster>
</layout>
```



**Figure 5.** Aerobic respiration pathway with the clustering layout defined above applied. The polypeptide composition reactions, RNA nodes, and the nodes that lie in the mitochondrion are clustered. The rest of the graph is laid out around these structures.