

version

2.1

BILKENT UNIVERSITY Information Visualization Research Group

MSKCC Computational Biology Center

Chisio BioPAX Editor

User's Guide

i-Vis@Bilkent & cBio@MSKCC

Chisio BioPAX Editor User's Guide



© i-Vis@Bilkent and cBio@MSKCC 2009
Bilkent University
Ankara 06800, TURKEY
Phone +90 312.290.1401 • Fax +90 312.266.4047

Table of Contents

Getting Started	3
Uses.....	3
Architecture.....	3
How to Download and Start Chisio BioPAX Editor	4
Basics	5
Pathway Model	5
Pathway View	5
Graphical User Interface and Notation.....	7
BioPAX Pathway Model and File Operations.....	10
Loading a Model	10
Merging Another Model.....	11
Saving Current Model	11
Viewing and Editing Pathways	12
Select and Zoom Tools	12
Changing Topology	14
Hiding Selected.....	14
Hiding Compartments	15
Reconnecting an Interaction	15
Changing Geometry	15
Layout	17
Highlighting	20
Pathway Operations	22
Opening and Closing Pathways	22
Renaming Pathways	22
Creating and Deleting Pathways	22
Duplicate a Pathway	22
Create a Pathway with Neighborhood of a Subpathway	22
Cropping to a Subpathway	23
Deleting Pathways	23
Updating All Open Pathways	23
Saving as Images and Printing	23

Querying Operations	24
Local Querying	24
Neighborhood Query	25
Paths Between Query	27
Paths From To Query	30
Common Stream Query	33
Compartment Query	36
Path Iteration Query	39
Querying Pathway Commons Database	42
Pathway Commons Queries (Level 2)	42
Pathway Commons Queries (Level 3)	42
SIF Operations	47
Simple Interaction Format	47
SIF Views in ChiBE	47
Creating SIF Views	47
Persisting SIF Views	50
Working with Large SIF Files	50
Visualizing Experiment Data	51
Loading Experiment Data	51
Reference Matching	52
Multiple experiments	53
Using Conversion Wizard	53
Fetching From GEO	59
Loading ChiBE Formatted Data	59
Fetching From cBio Portal	59
Visualizing Microarray Data	62
Causative Paths Based on the Loaded Experiment Data	64
Highlight With Data Values	64
Connect To DAVID	65
Third-Party Software License Agreements	68
References	69

Getting Started

In this chapter, we introduce the tool Chisio BioPAX Editor, and explain how to start the tool.

Chisio BioPAX Editor, or simply **ChiBE**, is an editing and visualization tool for pathway models represented by the BioPAX format [1]. The tool features user-friendly display, viewing, and editing of pathway models. Pathway views are rendered in a feature-rich format mostly compliant with SBGN process description notation [15], and may be laid out and edited with state-of-the-art layout and visualization technologies. In addition, facilities for querying a set of integrated pathways in Pathway Commons [2] are provided. Furthermore, visualization of experimental data overlaid on pathway views, including cancer genomics data from cBio Cancer Genomics Portal [14] is supported in ChiBE.

Uses

ChiBE can be used in a number of different ways, including:

- Automatically laying out and visualizing pathway models represented with BioPAX Level 2 or Level 3 format;
- Producing SBGN-compliant views of BioPAX Level 2 or Level 3 models;
- Querying Pathway Commons Database to construct pathways of user's interest and presenting to the user visually;
- Overlaying experimental data on pathway views;
- Querying cBio Cancer Genomics Portal for cancer genomics data of current associated with currently loaded pathway model;
- Creating simple binary views;
- Highlighting paths between molecules or sub-network of interest;
- Searching causative paths on loaded experiment data.

Architecture

Figure 1 shows the architecture of ChiBE at a high-level. Paxtools [3] library is used to parse the provided model (a BioPAX file) and create Java objects in memory for each BioPAX object in the model. A view is then created in a separate drawing canvas, for each pathway (notice that, with BioPAX Level 2 and 3 pathways, a pathway model is defined recursively, potentially made up of other pathways) that the user is interested in.

In case, the user executes a query, it is passed to the Pathway Commons Database using the http protocol through Paxtools. The results presented to ChiBE as Java objects in memory are converted to a ChiBE view and displayed in a separate canvas as a new pathway after an automatic layout is performed.

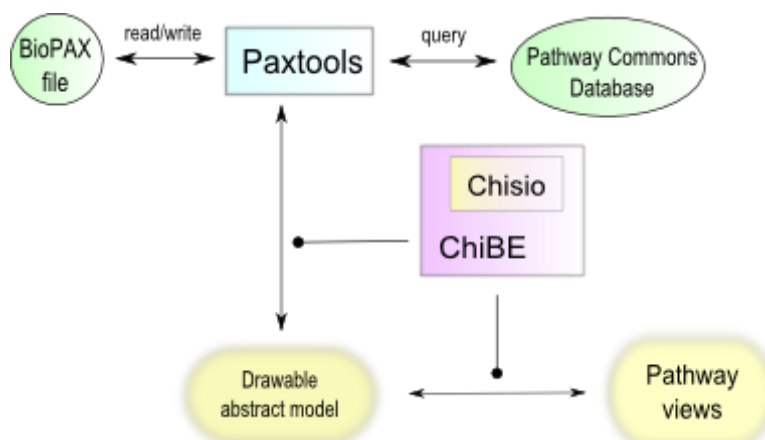


Figure 1 High-level architecture of ChiBE

How to Download and Start Chisio BioPAX Editor

ChiBE 2.1 is built on top of Chisio 2.0 [4] using Eclipse GEF (Graph Editing Framework) version 3.1 [5]. ChiBE runs on Windows, Mac OS X, and Linux operating systems. To download and set up ChiBE 2.1, simply go to the tool Web site and follow the instructions provided:

<http://www.bilkent.edu.tr/~bcbi/ChiBE.html>

Basics

In this chapter, we define the basic concepts and notation used in ChiBE.

ChiBE is a tool to load biological pathway models into memory and create views of this model for effective analysis. Each pathway model represented in BioPAX Level 2 and Level 3 format is potentially composed of other pathways, defined recursively. ChiBE's notation is based on the notation used in PATIKA tools [9][10] and is mostly compliant with SBGN process description notation [15].

Pathway Model

We call the entire set of biological information loaded from a BioPAX file a *pathway model*, or simply a *model*. Note, however, that a pathway model may be expanded by merging with another BioPAX file (see section on Merging Another Model on page 11). Each BioPAX file or each pathway model is potentially composed of one or more pathways, which may be defined in a hierarchical manner. To be exact, we use the notion of a pathway as defined by BioPAX as follows:

A pathway is a set or series of interactions, often forming a network, which biologists have found useful to group together for organizational, historic, biophysical or other reasons.

Although ChiBE supports both Level 2 and Level 3 models, it is important to note that they cannot be viewed simultaneously. You can only work on either a Level 2 or a Level 3 model at a certain time.

Pathway View

ChiBE allows users to choose the exact set of pathways to be visualized upon load. A separate drawing canvas is used for each such pathway. Each such pathway drawing is called a *pathway view*, or simply a *view* (Figure 2). The initial set of pathways from which a view is created may be changed later on, by closing currently viewed ones or by opening currently un-opened ones (see Chapter titled Pathway Operations on page 22 for more details).

Name and level of the BioPAX file containing the current pathway model

Object inspector

ID	http://www.reactome.org/biopax/4
Location	nucleoplasm
Data Source	Reactome;
Data Source	Reactome;reactome;
Reference	Reactome Database ID:1031708
Reference	Reactome:REACT_26763.1

The cBioPortal gene alteration inspector

Mutation freq	0.5%
- Unknown change	1
Expression change freq	18.9%
- Activating	36
- Inhibiting	3

Highlighted path resulting from a query

Nodes color coded with profiling data

Compound nodes for cellular locations and complexes

Name of currently visible pathway view

Pathway view in associated canvas in SBGN PD notation

Figure 2 A sample screenshot from ChiBE explaining some basic concepts

Graphical User Interface and Notation

The graphical user interface of ChiBE is composed of a menubar, a toolbar, and a number of drawing canvases (one per pathway view), organized with tabs (Figure 2).

A pathway view is composed of pathway objects (nodes and compound nodes) and their interactions. Compound nodes are exclusively used to represent *molecular complexes* and *cellular compartments* (Figure 3).

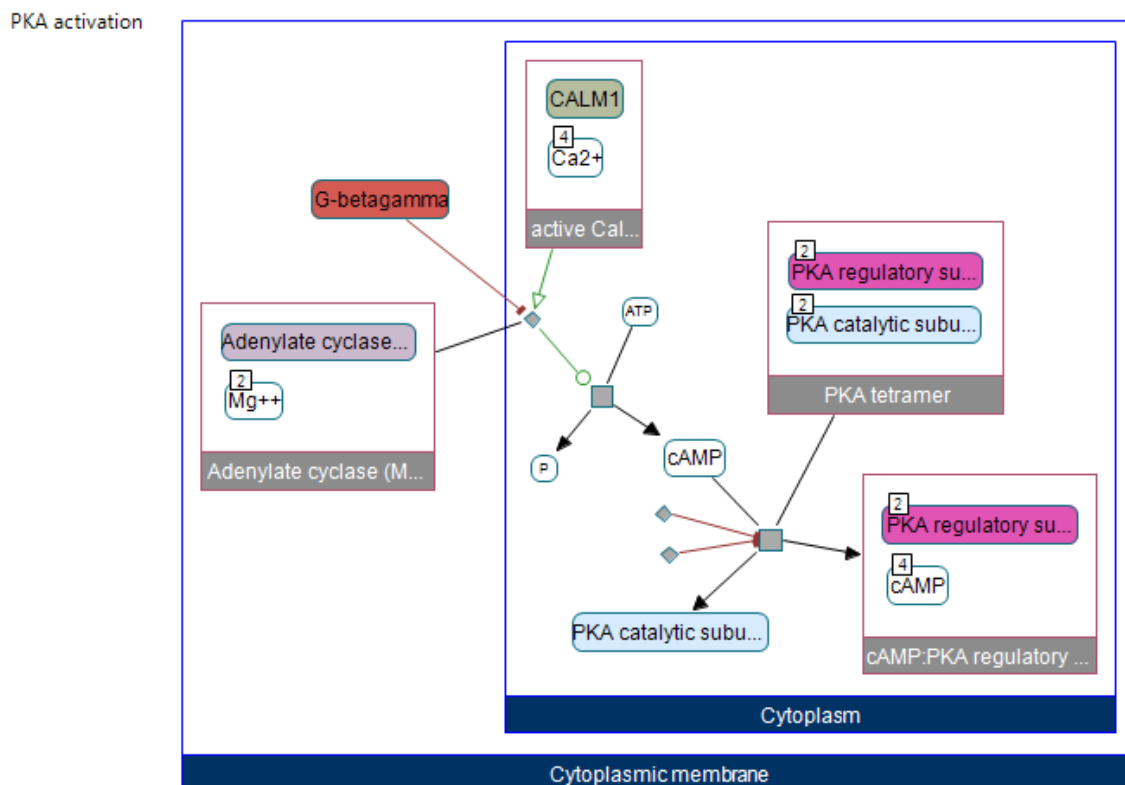
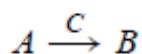


Figure 3 Sample notation for pathway views; grey colored compound nodes in this view represent complexes and blue colored ones are cellular locations (Cytoplasm and Cytoplasmic Membrane).

Non-compound (or simple) nodes are of two types: small gray boxes represent *transitions*, and each rounded rectangle represents a distinct *state* of a biological entity.

A transition represents group additions or removals, complex formations and disassociations as well as transportations and other cellular events. This model is very similar to the chemical equations of the form



where A is a substrate, B is a product and C is an effector. The states associated with a transition are connected to the corresponding transition box through *substrate*, *product* and *effector edges*, respectively. Substrate and product edges of a transition are drawn in black, whereas effectors are drawn in green and red for *activation* and *inhibition*, respectively. When an effector edge is directed towards another effector edge, or when there are more than one

controller molecules, small diamond nodes are used for showing this control. You can refer to Figure 4 to see all node and edge types in ChiBE and different glyphs used to represent them.

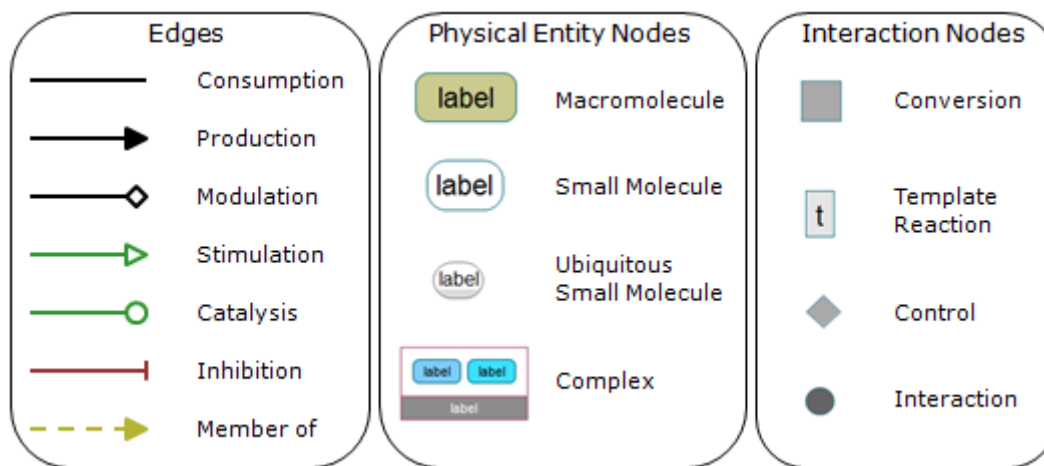


Figure 4 Different glyphs found in ChiBE along with their meaning

When the BioPAX pathway contains interaction data without details of the process, these interactions are shown with undirected edges between molecules when the interaction involves two participants (Figure 5). In case more than two participants are involved, the interaction is represented with a small circle that is connected to the participants. You may refer to [9] and [15] for more details of the ontology used for pathway views.

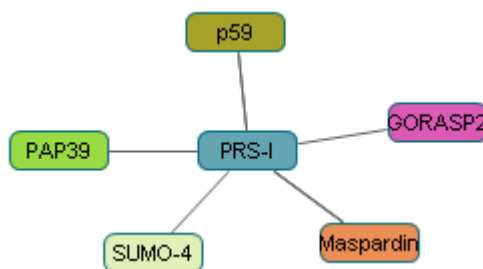




Figure 5 Neighborhood of PRS-1. Interactions have no process information; thus represented with simple edges between associated molecules.

ChiBE has context-sensitive pop-up menus associated with nodes, edges, and the entire pathway. These menus provide fast access to popular operations for the associated pathway object or pathway itself.

All nodes, edges, and compound nodes of a pathway view in ChiBE have distinct properties and UIs. These properties can be changed by using properties windows or inspectors for each graph object. The inspector of a pathway object is accessible through its pop-up menu (right-click while on the object) or by double-clicking on the object. The associated inspector can be opened up by “Edit |  Object Properties ...” menu item while the object to be inspected is the only current selection. The inspector for the pathway itself can be invoked by right-clicking on the drawing canvas when no nodes or edges are at the mouse location and choosing “ ...”

Properties ...” menu item. When you change your object’s properties, these changes are transferred to the model only after you press the “OK” button.

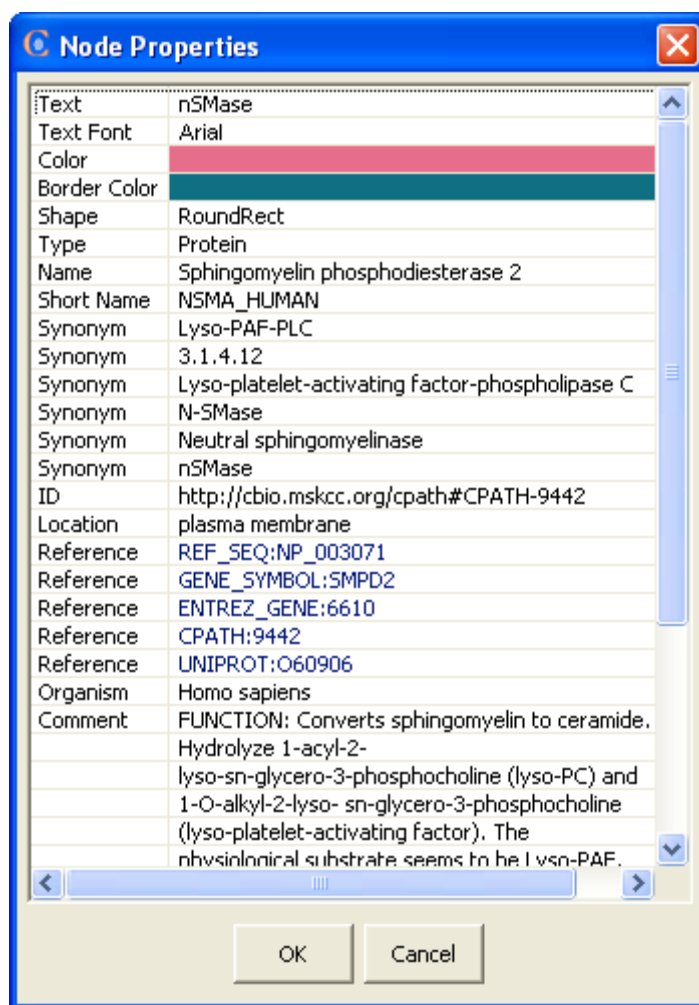


Figure 6 A node and its properties window or inspector


Notice that, sometimes, value of a certain property is actually an external reference (a hyperlink to a web page; say, associated page for a protein with a certain UniProt ID). Such references are displayed in blue and the related web page can be opened up by simply left-clicking on the reference.

BioPAX Pathway Model and File Operations

In this chapter, we describe how you can load, save, and merge pathway models in BioPAX formatted files.

Pathway models stored in BioPAX formatted files may be loaded into ChiBE as the current pathway model for analysis. ChiBE allows only one pathway model to be analyzed at a time. However, you may merge other pathway models into the currently loaded one to form a new, extended pathway model, assuming both models are of the same BioPAX level.

Loading a Model

Pathway models stored in BioPAX formatted files may be loaded into ChiBE by using “Model |  Load ...” menu item. This opens up the “Pathway Selection Dialog” (Figure 7), listing all pathways in the associated model. In case, no pathway is defined in the loaded file, one, whose name is the same as the file name, is created automatically and listed.

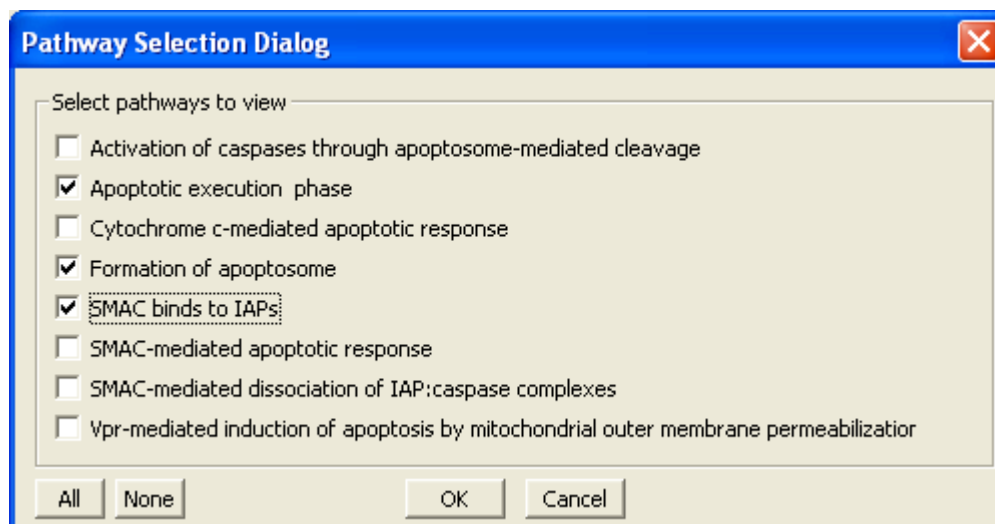



Figure 7 “Pathway Selection Dialog” is used to indicate the initial set of pathways (in this case three out of the eight available), from which views are to be constructed



Upon clicking the “OK” button, a separate drawing for each chosen pathway will be created. Each such pathway is automatically laid out using the algorithm by Dogrusoz et al. [7][8] before it is presented to the user.

Merging Another Model

If you would like to expand your pathway model with a different one stored in another BioPAX file, choose “Model |  Merge With ...” menu item. Upon selection of the BioPAX file, the “Pathway Selection Dialog” will open up, listing the pathways from the newly loaded file, as well as the ones that are part of the previously loaded model. The user may then change the list of pathway views to visualize (e.g. add other pathways from newly loaded model).

While performing this operation, user must bear in mind that Level 2 and Level 3 models are not compatible. Therefore, Level 2 and Level 3 models cannot be merged.

Saving Current Model


You may save your current model with the same file name or under a different name using “Model |  Save” or “Model |  Save as”, respectively. These operations are used to persist geometry information of pathway view objects as well; thus the layout of each pathway view is preserved across save/load operations. In addition, any newly created pathways (see Chapter titled Pathway Operations on page 22).

Viewing and Editing Pathways

In this chapter, we describe how you can navigate over and edit pathway views as desired.

As described earlier, each pathway model consists of one or more pathways. Each such pathway is presented to the user as a separate view in an associated drawing canvas. Various mechanisms to navigate over and edit the topology as well as the geometry of such pathway views are facilitated to the user.

Select and Zoom Tools

The “Select Tool” can be chosen from the menubar using “View |  Select Tool” or from the toolbar by clicking on the associated icon. This tool is used to select nodes, edges and compound nodes. Multiple selections are supported by holding the Shift or the Control key while left-clicking. When you select an object, handle points are drawn around the object.

Handle points for nodes are filled. This means that, node can be resized. Handle points for compound nodes are empty as they cannot be resized (Figure 8).

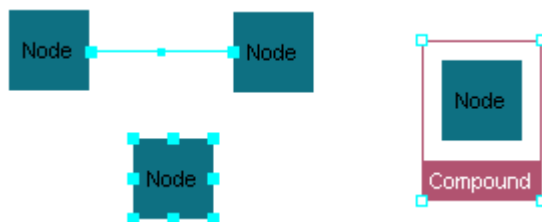


Figure 8 Example selection handles for all types of graph objects

When you press the left mouse button on the drawing canvas outside the boundaries of any graph objects, and drag your mouse, the select tool is automatically switched to the “Marquee Selection Tool”. When you release the mouse button, all simple and compound nodes and edges that are *completely* included in the marquee selection area is selected (Figure 9).

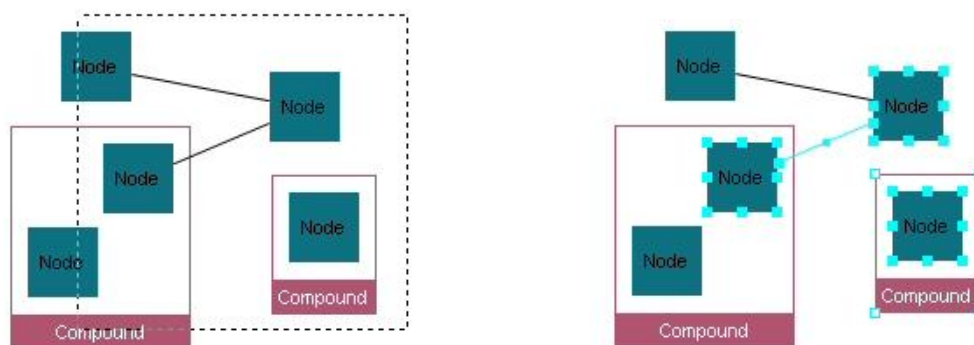



Figure 9 Example use of the Marquee Selection Tool

ChiBE has several zooming facilities: marquee zoom, zoom-in, zoom-out, zoom to specified level, and fit-in-window.

The “Marquee Zoom Tool” can be chosen from the menubar using “View |  Marquee Zoom Tool” or directly from the toolbar by clicking on the associated icon. This tool is used to zoom into a specified rectangular area of the pathway view (Figure 10 and Figure 11).

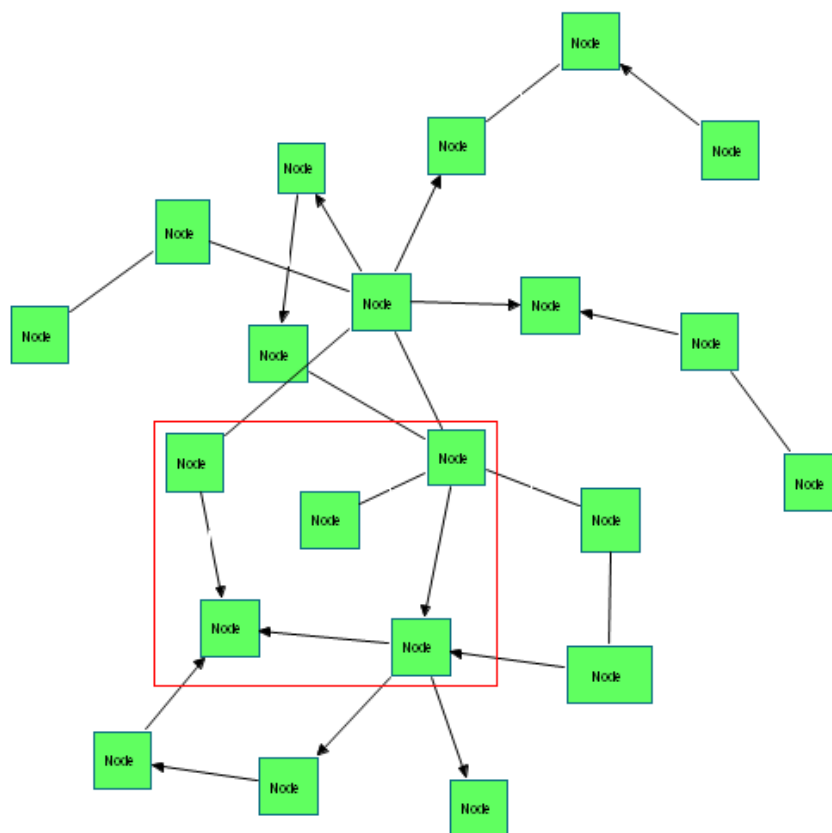


Figure 10 Before marquee zoom to the area specified by the red rectangle

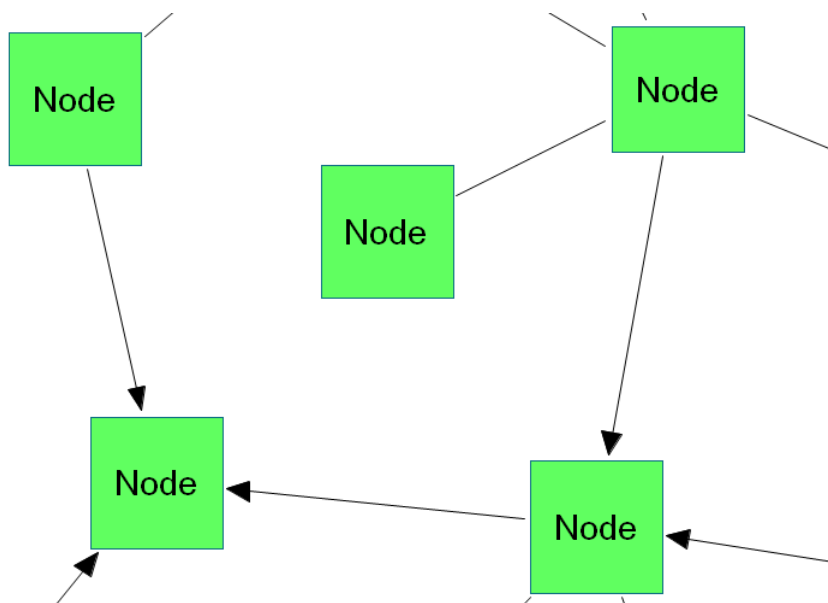


Figure 11 After marquee zooming into the area as described by Figure 10

Other zoom operations like zoom-in (🔍), zoom-out (🔍) and zoom to specified level are also supported. These operations can be found in the menubar under “View | Zoom” and in the toolbar. In addition, pathway popup menu provides zoom-in and zoom-out capabilities.




Fit-in-window operation is another useful zoom operation, which shows the whole pathway view in the window by properly scaling it. You can fit your pathway in window from the menubar using “View | 🔍 Fit In Window” or directly from the toolbar by clicking on the associated icon.

Changing Topology


Ability to interactively change the topological information of your pathways is rather limited with this version of ChiBE. You may, however, hide pathway nodes or edges from your pathway views or simply reconnect interactions in your pathway views to change the source and/or target pathway nodes.

Hiding Selected


To hide pathway nodes or interactions, you must select them first. Then you can hide them in one of the following ways:

- Pressing “DEL” button on your keyboard;
- Using the toolbar item ;
- Using “Edit |  Hide Selected” in the menubar;
- Using “ Hide Selected” item under the node popup menu.

Multiple graph objects may be hidden at once by using selection of these objects together. Remember that hidden pathway objects may be unhidden by using the update operations described on page 23.

One may also remove existing compound nodes while keeping their contents (child nodes and edges) at their current locations. This can be done from the menubar using “Edit |  Hide Compound” or by directly using the associated icon in the toolbar after selecting the compound pathway node(s) to be hidden.

Hiding Compartments

ChiBE uses compound nodes to divide pathway view into compartments based on cellular locations of physical entities. However, there may be cases where user is not interested in the cellular organization of the pathway which will make the presence of these compartment nodes unnecessary. Therefore, ChiBE offers a way of temporarily removing compartments from the view. This feature can be reached through the menubar using “Edit |  Hide Compartments”. Upon selection, all compound nodes representing compartments are removed from the view. Users can restore these compartment nodes by using “Edit | Show Compartments” through the menubar.

The presence of compartment nodes can also increase the complexity of resulting pathway view. Bearing this in mind, upon initial loading of a model, ChiBE hides all the compartments by default if the number of edges in the view exceeds 20. If interested, users can easily retrieve these hidden compartments through “Edit | Show Compartments” menu item.

Reconnecting an Interaction

You can change the source or target of an interaction. When an interaction is selected, its handle points become visible. When you move your mouse to the handle point associated with the target or source node, the cursor is changed into a plus icon. You can click, drag and drop this point onto its new target or source. While dragging, the cursor indicates potential new target or source for this edge (Figure 12).



Figure 12 Edge reconnection; after selection of the edge (left), during dragging (middle) and after reconnection (right).

Changing Geometry

You can change the geometry of nodes by moving or resizing them. The geometry of the edges, on the other hand, can be changed by re-routing them by creating new bend points, by moving or deleting existing ones. The layout operation also changes the geometry of graph objects as will be explained later on.

To move a node, first you must select it. Then click anywhere on the node and start dragging; you will see a “ghost shape” of your node as you drag it. You can drop it to the location you like. Upon release of the mouse the operation is completed (Figure 13).



Figure 13 Move a node by selecting and dragging it

To resize a node, first you must select it upon which its handle points will be visible. By using these points, you can resize the node. Just select a handle point and move it; a ghost shape will appear indicating the new size of the node upon release of the mouse (Figure 14).

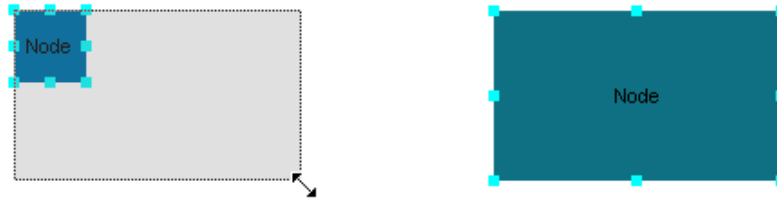


Figure 14 Resize a node; before (left) and after (right)

You cannot resize compound nodes as their geometry is auto-calculated by the geometry of its contents as explained earlier. This is indicated by empty handles upon selection of compound nodes (Figure 15).

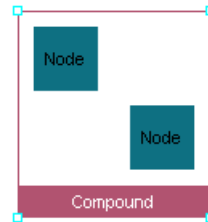


Figure 15 Compound nodes are automatically resized.

Both move and resize operations may be applied to multiple objects simultaneously for convenience.

ChiBE supports bend points to freely route edges around other graph objects. You can create, delete and relocate bend-points. When an edge is selected two handles of two different sizes becomes visible. The large ones (except for the end points of the edge used for reconnecting edges) correspond to the bends of the edge. The smaller ones, on the other hand, can be clicked on and dragged to create new bend points (Figure 16). Initial position of the newly created bend point will be right in between the two nearest existing bends on the edge but it will move with the mouse.

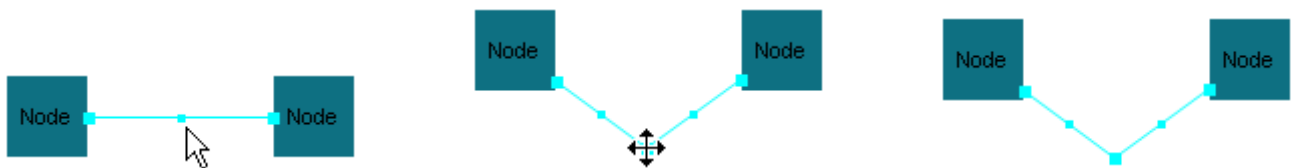






Figure 16 Bend point creation; after selection (left), during drag of the small handle (middle), and upon completion with the release of the mouse (right)



You can relocate existing bend points by dragging them to new locations as you like. If the final location of a bend point upon release of the mouse is to be aligned with its current neighbors of the bend point to make a straight line (or very near such a position), then the bend point is automatically deleted (Figure 17).



Figure 17 Bend point deletion; initially the edge has two bends (left); upon drag and release of the left bend, it is deleted to leave the edge with only one bend (right).

Layout

You may arrange the pathway view automatically by calling “Layout |  CoSE Layout” or “Layout |  Spring Layout” or by clicking on  or , respectively on the toolbar. The former operation is based on the algorithm by Dogrusoz et al [7][8], whereas the latter calls a regular spring-embedder implementation, which does not support compound nodes.

Both algorithms can be configured through a properties dialog using “Layout |  Layout Properties” in the menu or by clicking  on the toolbar.

Options common to all layout styles are gathered on a separate tab in Layout Properties dialog (Figure 18).

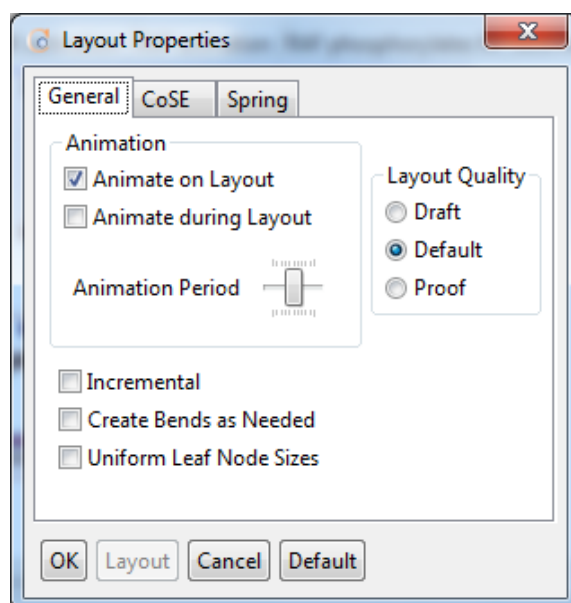


Figure 18 Common layout options

Animate on Layout: Whether the results of layout should be displayed in an animated fashion or not.

Animate during Layout: Whether the results of layout should be displayed as the operation is progressing or not. This is especially useful for debugging purposes for a layout developer.

Animation Period: How often the canvas should be refreshed for animation during layout.

Incremental: Should layout start from current positions (respecting current layout) or not (usually starts from random positioning).

Create Bends as Needed: Should the algorithm create bend points for edges as needed (e.g. for multi-edges).

Uniform Leaf Node Sizes: Whether nodes (except for any compound nodes) are of uniform dimensions or not. If nodes are of uniform dimensions, setting this option might speed up the layout process since, then, force calculations will be based on node centers as opposed to clipping points.

Layout Quality: Quality of layout; usually the higher the quality, the slower the layout will be.

CoSE layout options are shown in Figure 19.

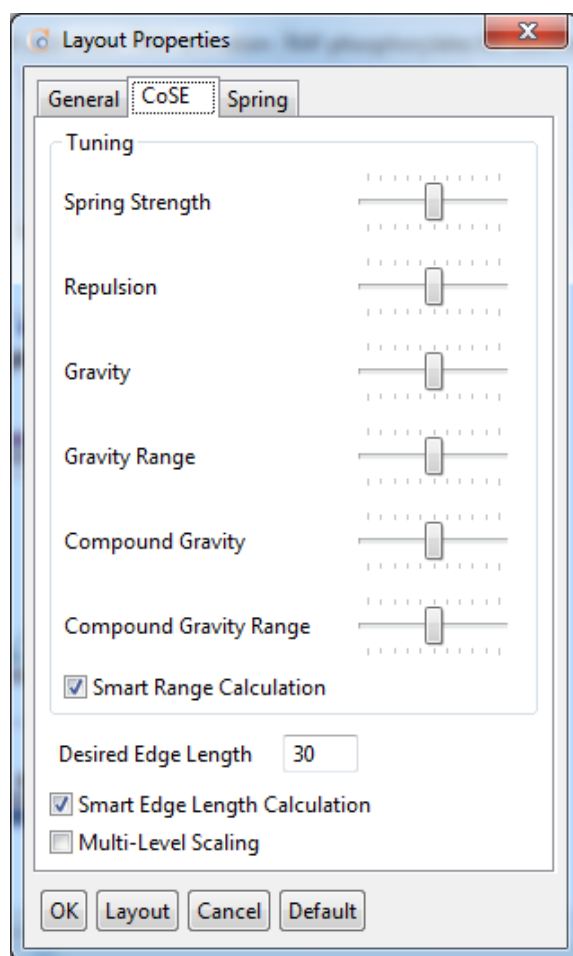


Figure 19 CoSE layout options

Spring Strength: Constant for spring forces corresponding to edges; the higher this value is the stronger spring forces will be.

Repulsion Strength: Constant for repulsion forces that are applied to node pairs; the lower this value is, the closer the nodes will be.

Gravity Strength: Factor for the gravity of each graph; the higher this value is, the closer disconnected parts of the graph will be.

Gravity Range: This defines the range, outside which gravity will be effective. The default range is calculated using the size of the graph, with the hope that within the range no gravity is needed. The larger the percentage of this range is the larger the area in which gravity will not be applied.

Compound Gravity Strength: The factor for the gravity inside compound nodes; the higher this value is, the closer disconnected parts of the graph inside compound nodes will be.

Compound Gravity Range: This is the compound version of Gravity Range defined above.

Desired Edge Length: Desired length of an intra-graph edge; inter-graph edges are allowed to be longer as they need to span their own compound bounds.

Smart Range Calculation: Whether or not FR-Grid Variant [16] method is used during repulsion force calculation. This calculation eliminates unnecessary repulsion calculations and checks each node's repulsion with its immediate nearby neighbors using a grid.

Smart Edge Length Calculation: Whether or not smart calculation of ideal edge length should be performed for inter-graph edges. When this is enabled, we calculate the number of nesting levels each edge spans and calculate the desired edge length accordingly.

Multi-Level Scaling: Whether or not multi-level scaling [17] should be used to speed up the layout process. Use of multi-level scaling captures the structure of the graph better, often yielding better layouts.

Spring layout options are shown in Figure 20.

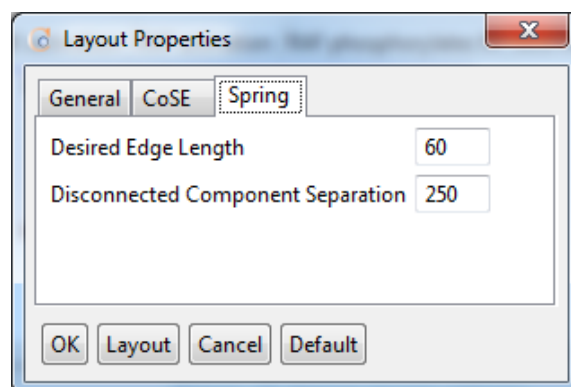



Figure 20 Spring layout options

Desired Edge Length: Desired length of an edge

Disconnected Component Separation: Adjusts the separation of disconnected parts of the graph

Highlighting

Pathway objects may be highlighted using the highlight facility to differentiate them from others (e.g. a path, a cycle or a particular subgraph of interest). Selected objects may be highlighted through the “Highlight Selected” item under the node pop-up menu or using “Edit |  Highlight Selected” in the menubar. Multiple objects may be highlighted together (Figure 21).

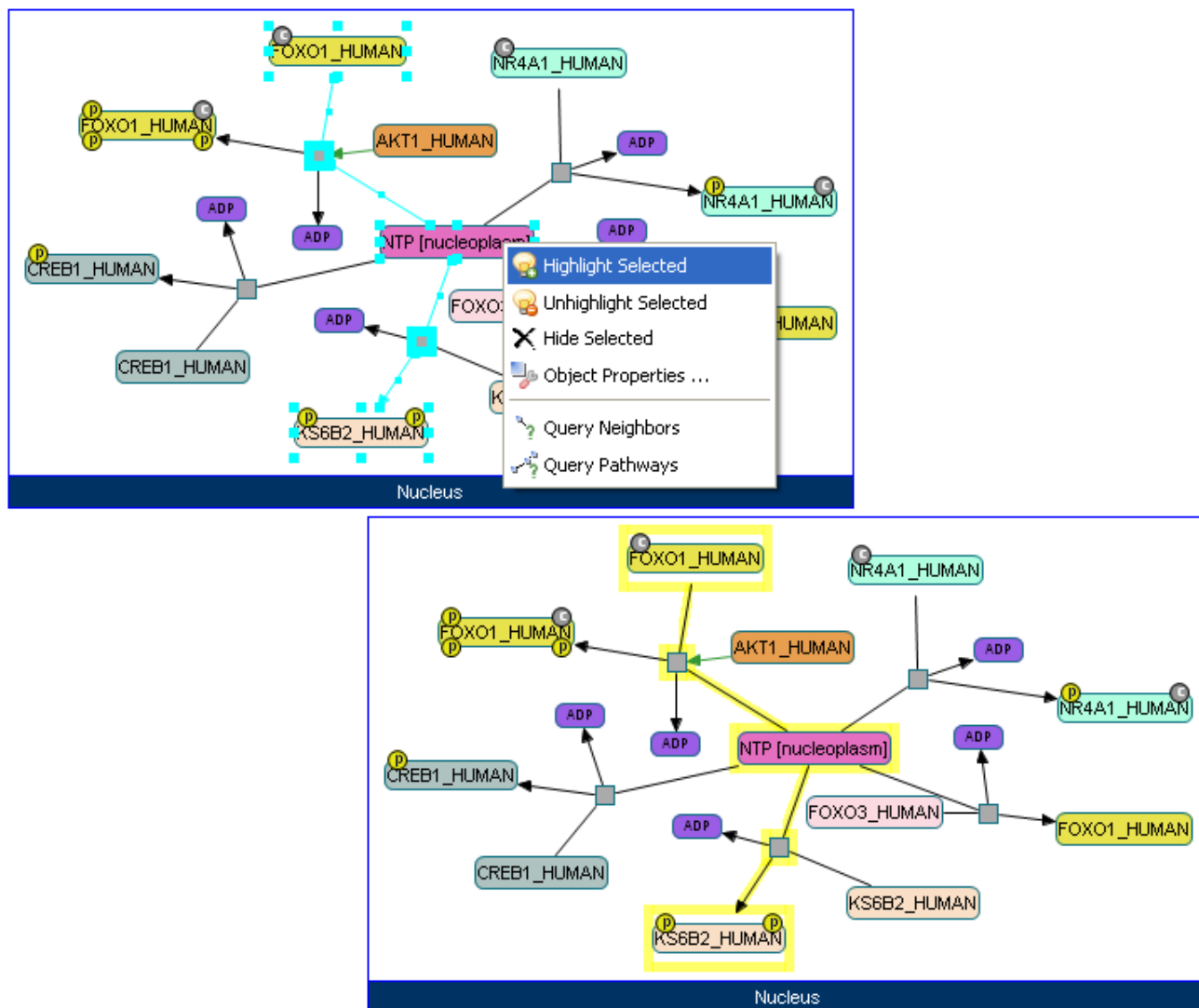




Figure 21 Highlighting objects

You can change the highlight color from the pathway inspector in the pathway pop-up menu. Thus different objects may be highlighted with different colors if desired.

You can remove highlights of all objects by selecting “Edit |  Unhighlight All” item in the menubar or from the graph pop-up menu.


Alternatively, a subset of currently highlighted objects may be unhighlighted by first selecting them and then choosing “Edit |  Unhighlight Selected” in the menubar or “ Unhighlight Selected” in the node pop-up window.



Pathway Operations

In this chapter, we list the operations to manage pathway views.


A subset of the list of pathways associated with the current pathway model are presented to the user in separate drawing canvases. This subset is initially determined by the user and may be modified later through opening and closing pathways and certain other operations that create pathway views according to the user's interest.

Opening and Closing Pathways

You may change the currently viewable set of pathways that belong to the current pathway model using “Pathway |  Open Pathway ...” menu item. This opens up the “Pathway Selection Dialog” mentioned earlier (Figure 7).

The currently visible pathway may also be closed using “Pathway | Close |  Close Pathway” menu item. You may also close all open pathways using “Pathway | Close |  Close All Open Pathways” menu item.


Renaming Pathways

The name of the currently visible pathway might be changed using the “Pathway |  Rename Pathway ...” menu item.

Creating and Deleting Pathways

New pathways and their views may be created using the pathway objects in the currently loaded model in a number of ways as follows.



Duplicate a Pathway

By using “Pathway | Create |  Duplicate Pathway” menu item, a duplicate of the currently visible pathway may be created. This creates a replica of the currently visible pathway except the name of the pathway is appended “ (2)” (or “ (3)”, etc. for subsequent duplicates) to uniquely name it. This might be especially useful if you would like to modify a particular pathway but would like to refer to a view of the “original” one as needed.



Create a Pathway with Neighborhood of a Subpathway

If you are interested in creating a pathway that contains a subset of the current pathway plus a certain neighborhood of these pathway objects in the current model, then you should first select the set of pathway objects of interest, and click “Pathway | Create | Create Pathway With Neighbors Of Selected”.

Cropping to a Subpathway

When you are interested in only a certain subpathway of the currently visible pathway, use “Pathway | Create |  Crop To Highlighted” menu item. Alternatively, you can use “Pathway | Create |  Crop To Selected” menu item for the same purpose. Upon selecting states on the view, clicking on this menu item will display the cropped pathway in a new view. The name of the newly created pathway contains the name of the original pathway with “ cropped” appended to it.

Deleting Pathways

The currently visible pathway or all open pathways could be deleted from the current pathway model by issuing “Pathway | Delete |  Delete Pathway” or “Pathway | Delete |  Delete All Open Pathway”, respectively. Notice that this operation does not delete the contents of the associated pathway from the model but simply the abstraction itself.

Updating All Open Pathways

This operation ensures that currently open pathways are up-to-date with respect to topology. In other words, any hidden parts of a pathway (see Chapter titled Viewing and Editing Pathways on page 12) are added back to these pathway views. Other changes on the topology and geometry of the pathway are also changed back to what they are in the model. For instance, any interactions whose source and/or target have been changed are reverted back to their original source and target.

Saving as Images and Printing

You may create static images of the currently visible pathway views, in BMP (.bmp), JPEG (.jpg), or PNG (.png) file formats. An image of the entire pathway view may be created by using “Pathway | Save Pathway As Image ...”. If you would like to create an image of what is currently visible in the drawing canvas, you should use “Pathway | Save View As Image ...” menu item.

The currently visible pathway view may also be printed by using “Pathway |  Print Pathway ...” menu item.

Querying Operations

In this chapter, we discuss how you can generate novel pathways by performing local queries or Pathway Commons database queries.

Biological pathways are usually too complex just to layout and spot what you might be needing. You may have a specific molecule at hand to search for; but due to this complexity, you may fail to actually find it. To solve this problem, ChiBE offers a query framework, with which you can analyze a specific set of molecules more deeply without getting lost in huge network views.

Querying operations can be divided into two groups: local queries and Pathway Commons [2] database queries. Local queries search for relationships within the loaded pathway, whereas Pathway Commons queries are performed remotely within an online database containing many different pathways.

Local Querying

Local querying support in ChiBE offers the user to analyze a specific set of molecules or relationships in the currently loaded model. Upon selection of some parameters, the program generates a result which is a sub-pathway of the current model. We currently offer the following local queries: *Neighborhood*, *Paths Between*, *Paths From To*, *Common Stream*, *Compartment*, and *Path Iteration*.

There are two different ways of performing local queries; either through menubar or through node pop-up menu. You can use “Query | Local” item from menubar to choose any of the local queries. In addition, for convenience, node pop-up menu offers a subset of these queries, where the query works over specific (selected) states of an entity: *Neighborhood*, *Paths Between*, and *Common Stream*. For instance, if you select the phosphorylated state of BRCA1 in a pathway view and perform 1-neighborhood from the context menu of the associated node, you will get 1-neighborhood of this specific state *only* in the resulting pathway. However, if you perform the neighborhood query from the menubar and select (add) the BRCA1 as an entity, 1-neighborhood of *all* states of BRCA1 (including its phosphorylated and native states) will be computed and returned as the result.

There are two options for visualization of the query results: they can be merged into the existing view and the part of the merged view relevant to the query will be highlighted; or result can be displayed in a new view.

Neighborhood Query

Neighborhood query is useful when you try to understand what is affecting and/or affected by a specific molecule. You can find neighbors of the selected molecule in different directions within a specified length limit by the neighborhood query.

To perform this query through menubar, you can use “Query | Local | Neighborhood ...” item, which will lead you to a window to determine parameters of query (Figure 22).

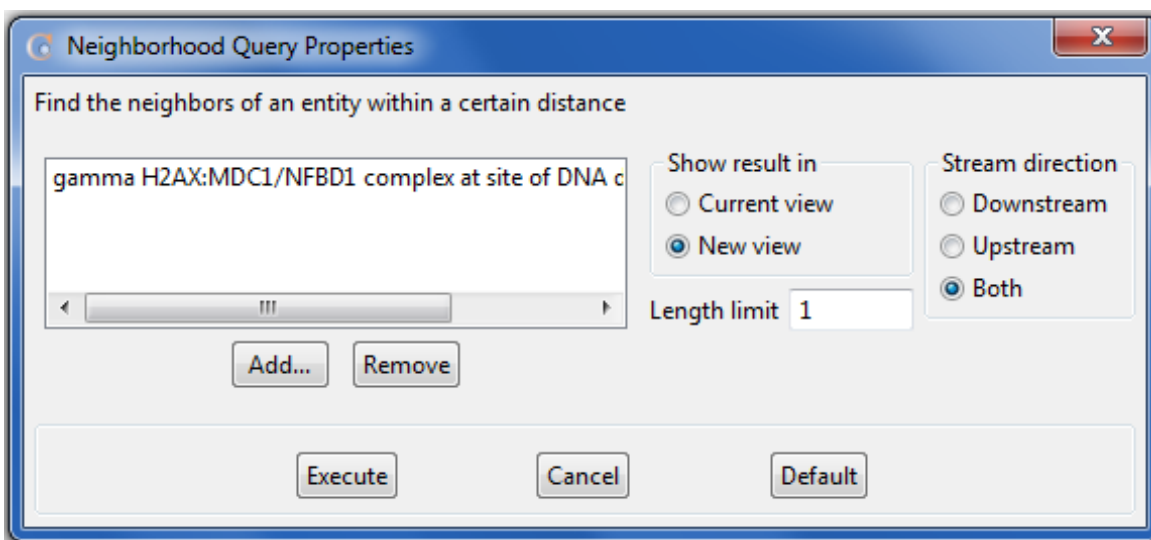


Figure 22 Sample Neighborhood Query Properties dialog (entity based)

Following parameters should be specified:

- **Show result in:** This selection between current view and new view determines how to display query result; merging into existing view or showing in a new canvas, respectively.
- **Stream direction:** This parameter will determine the direction of neighborhood query. Selecting *downstream* will result in looking for molecules only in downstream of the chosen entities. Selecting *upstream* will similarly result in looking for molecules only in upstream of the chosen entities. If you select *both*, then neighborhoods in both directions will be searched.
- **Length limit:** This parameter is a nonnegative integer that limits distance of a molecule from selected entities in order to be included in the query result.

Text field in left half of the dialog holds selected entities that will be source entities of neighborhood query (Figure 22). In order to add an entity to that field, the user should click the “Add...” button, which will open a new window that displays all entities found in that particular model (Figure 23). You can choose any entity from this dialog and upon clicking “Add” button, selected entity will be added to the text field described earlier. Multiple selections are possible. You can type a text in text field at the top of the dialog and then click “Filter” button in order to visualize only those entities that contain the specified text in their name and synonyms. Clicking “Show All” button will reset and result in display of all entities in current model. After adding entities to the text field, you can use “Remove” button to remove selected entities from text field.

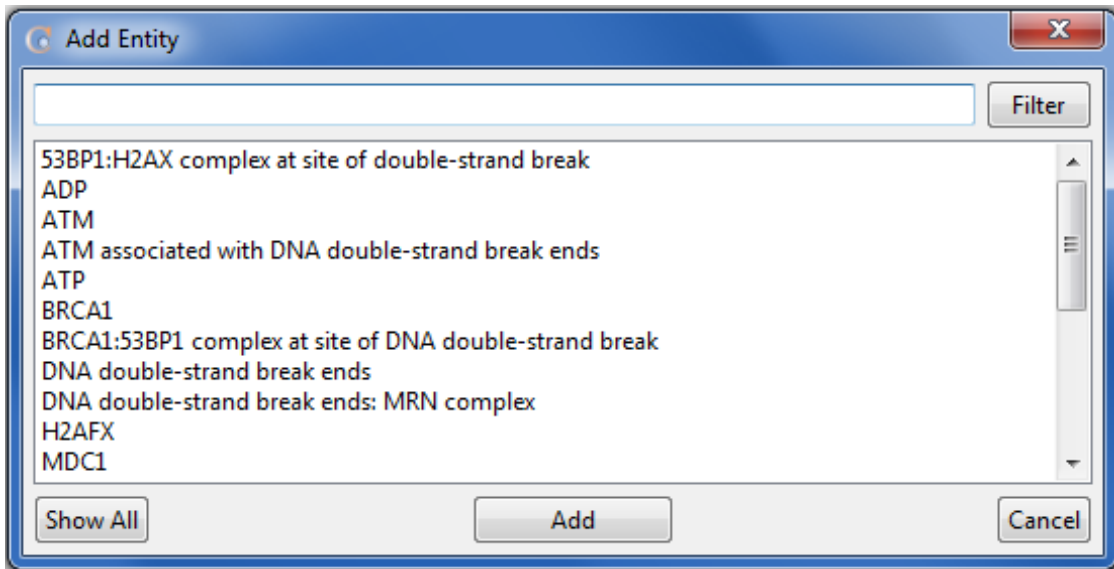


Figure 23 Add Entity dialog that is used to add entities into text field in the query properties dialogs

After determining all parameters of the query, you can start query by clicking “Execute” button (Figure 24). You can also return to default values of parameters by clicking “Default” button, which are “current view”, “both”, and 1, respectively.

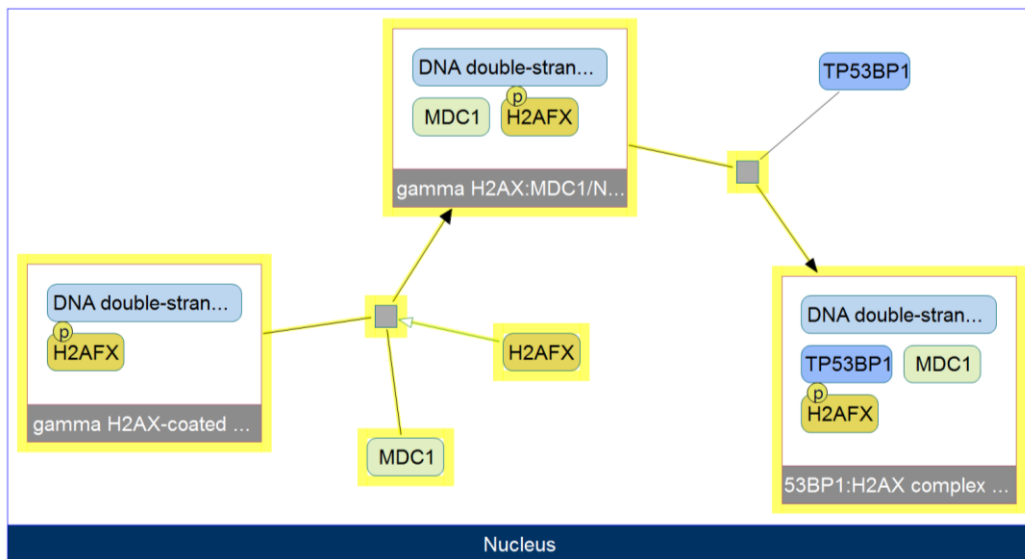


Figure 24 Result of the query in Figure 22. 1-level neighborhood of gamma H2AX:MDC1/NFBD1 complex at site of DNA double-strand break in both directions is shown (yellow)

As mentioned earlier, you can also perform a neighborhood query through the node pop-up menu. To do this, possible multiple source state(s) should be chosen from the graph. Choosing “Local Query | Neighborhood ...” in pop-up menu will open a dialog to obtain remaining parameters of the query (Figure 25). These are the same parameters as in Figure 22.

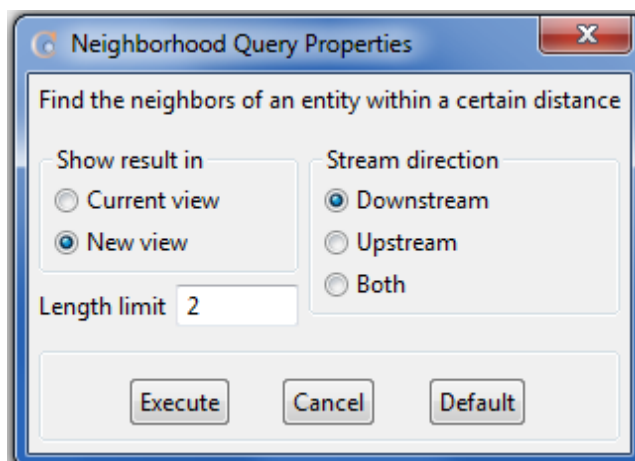


Figure 25 Sample Neighborhood Query Properties dialog (state based)

After determining all parameters of a query, you can start query by clicking “Execute” button (Figure 26). You can also return to default values of parameters by clicking “Default” button which are “current view”, “both”, and 1.

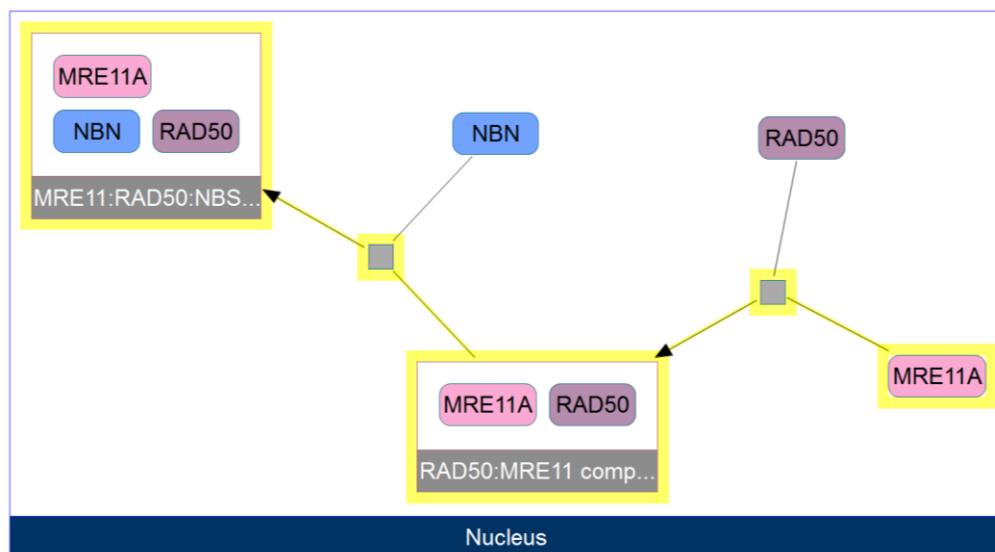


Figure 26 Result of the query in Figure 25; 2-level neighborhood of MRE11A in downstream direction is shown (yellow)

Paths Between Query

Paths Between query can be used when you want to find out about every possible, possibly indirect, interaction among a set of molecules within a specified length limit.

To perform this type of query through menubar, you can use “Query | Local | Paths Between ...” item, which will lead you to a window to determine parameters of the query (Figure 27).

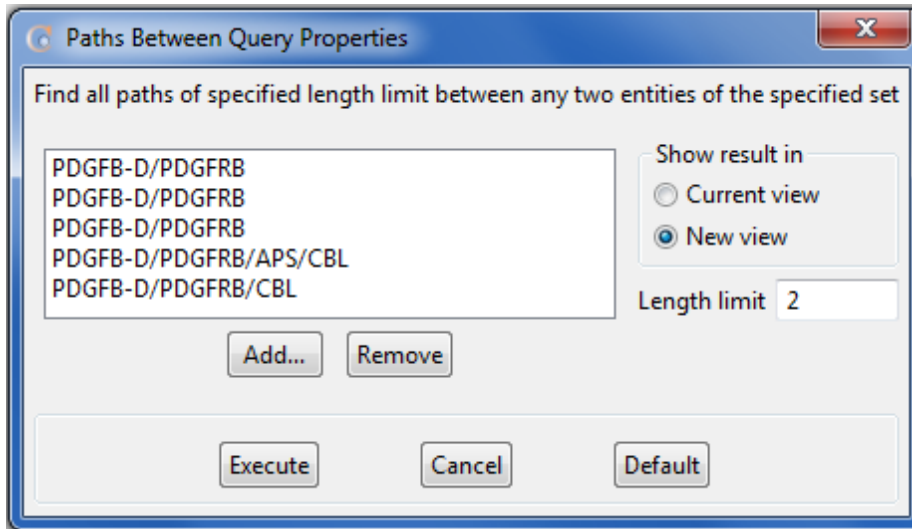


Figure 27 Sample Paths Between Query Properties dialog (entity based)

Following parameters should be specified:

- **Show result in:** This selection between current view and new view determines how to display query result; merging into existing view or showing in a new canvas, respectively.
- **Length limit:** This parameter is a nonnegative integer that determines length of the longest directed path found between entities of interest.

Text field in the left half of the dialog holds selected entities that will be source entities of the query. In order to add an entity to that field, user should click “Add...” button which will open a new window that displays all entities found in that particular graph (Figure 23). You can choose any entity from this dialog and upon clicking “Add” button; selected entity will be added to the list in Figure 27. Multiple selection is possible. You can type a text in text field at the uppermost of dialog and then click “Filter” button in order to visualize entities only which contain that text in their name and synonyms. Clicking “Show All” button will return list of entities into its initial state. You can use “Remove” button to remove selected entities from the list.

After setting all parameters of the query, you can start the query by clicking “Execute” button (Figure 28). You can also return to default values of parameters by clicking “Default” button, which are “current view”, and 1, respectively.

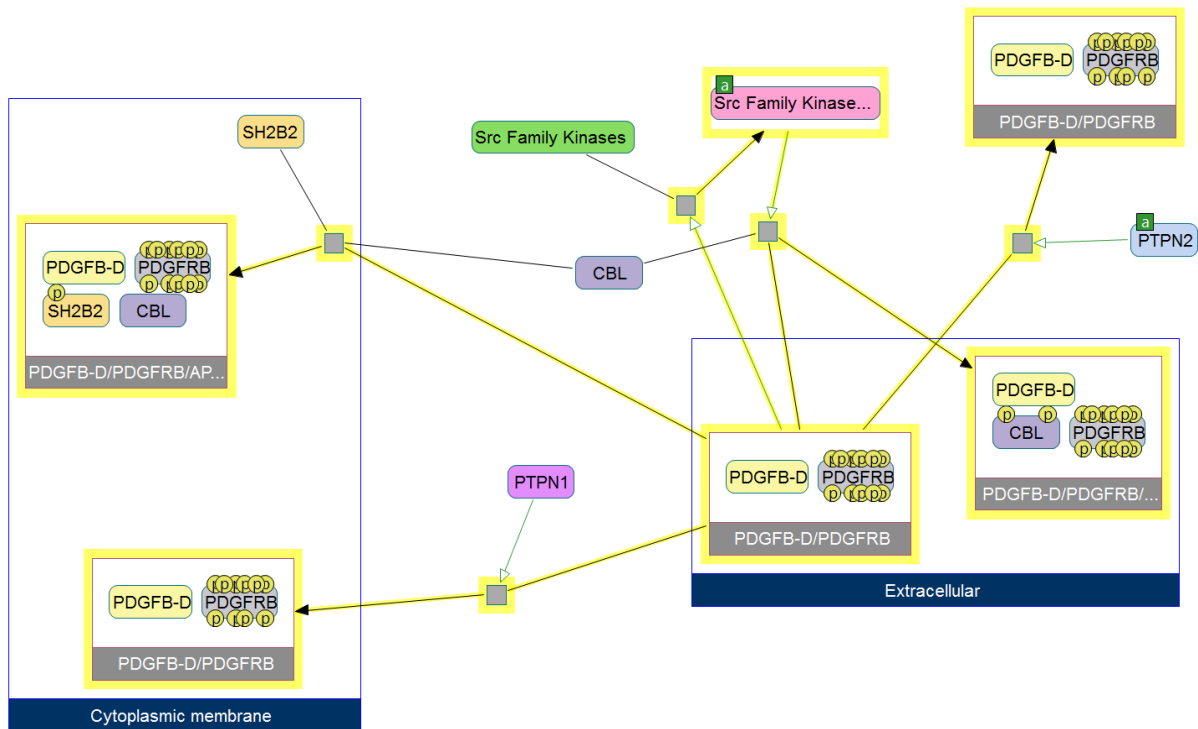


Figure 28 Result of the query in Figure 27; sub-network composed of paths between selected entities is shown (yellow)

As mentioned earlier, you can also perform this query through node pop-up menu. Select the source states on the view and right click on one of the selected nodes. Choosing “Local Query | Paths Between ...” in pop-up menu will open a dialog to obtain remaining parameters of the query (Figure 29 and Figure 30). These are the same parameters as in Figure 27.

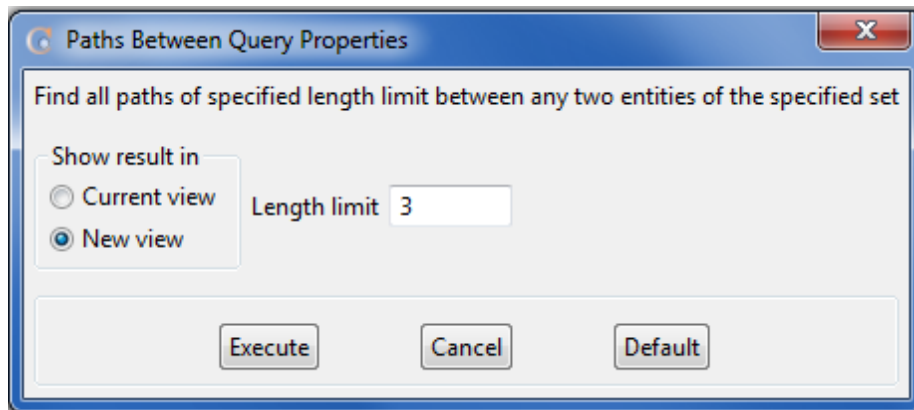


Figure 29 Sample Paths Between Query Properties dialog (state based)

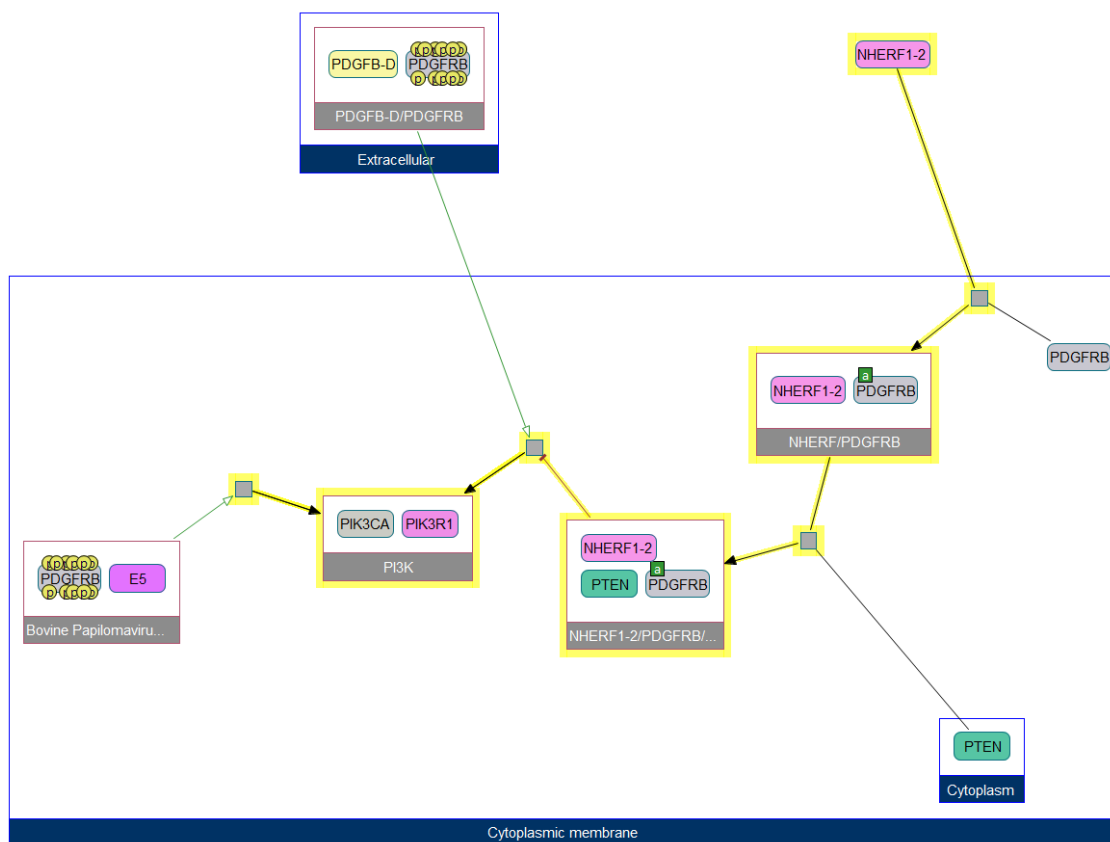


Figure 30 Result of the query in Figure 29; sub-network composed of paths between selected entities is shown (yellow)

Paths From To Query

Paths From To query is similar to Paths Between Query, but finds paths from a specified source set of molecules to a specified target set.

To perform this query through menubar, you can use “Query | Local | Paths From To ...” item, which will lead you to a window to set parameters of the query (Figure 31).

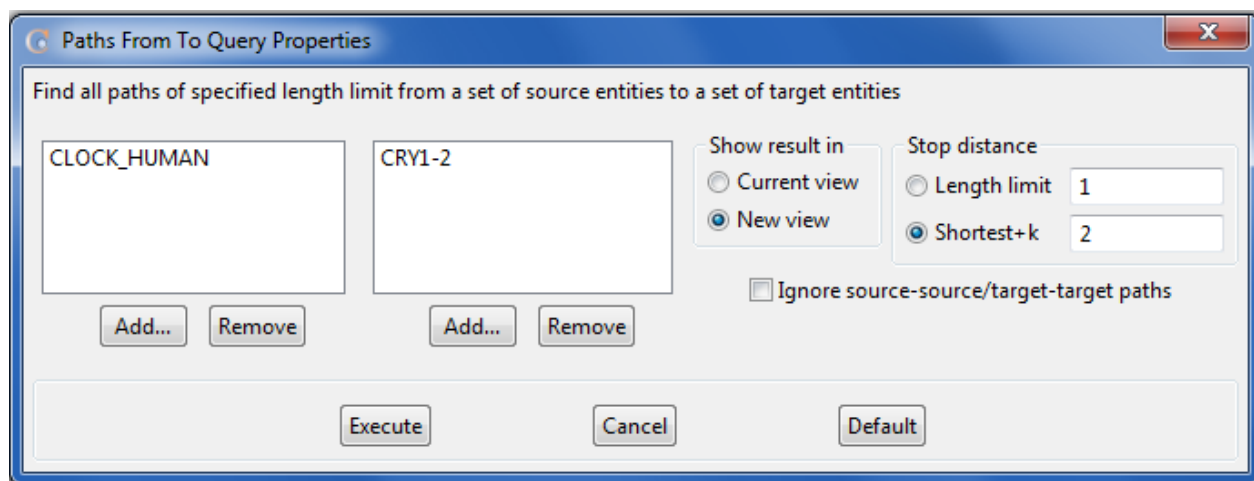


Figure 31 Sample Paths From To Query Properties dialog (entity based)

Following parameters should be specified:

- **Show result in:** This selection between current view and new view determines how to display query result; merging into existing view or showing in a new canvas, respectively.
- **Stop distance:** *Length limit* should be a nonnegative integer that determines length of the longest path found between entities of interest. For *shortest+k*, the user should specify a nonnegative integer that will be added up to the length of a shortest path to determine length of the longest path found between entities of interest.
- **Ignore source-source/target-target paths:** When this parameter is checked, paths between two source molecules or paths between two target molecules will not be included in the query result; you will only see paths from a source to a target molecule.

First text field in the left half of dialog shows the list of selected entities that will be source entities of the query. Second text field shows the list of selected entities that will be targets. In order to add an entity to those fields, user should click “Add...” button which will open a new window that displays all entities found in that particular graph (Figure 23). You

can choose any entity from this dialog and upon clicking “Add” button; selected entity will be added to text field in

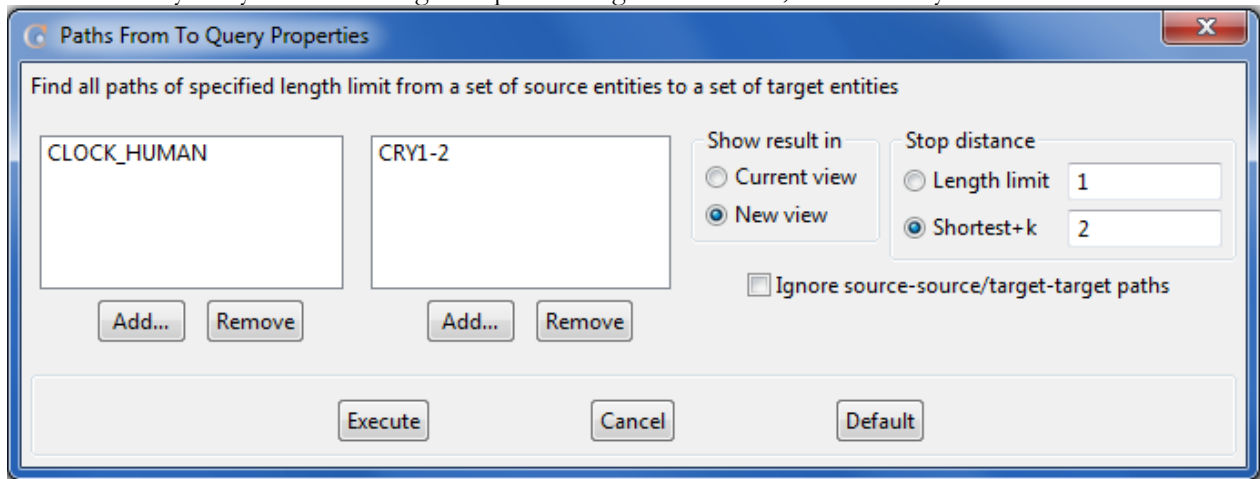


Figure 31. Multiple selections are possible. You can type a text in text field at the uppermost of dialog and then click “Filter” button in order to visualize entities only which contain that text in their name and synonyms. Clicking “Show All” button will return list of entities into its initial state. After adding entities to the text field, you can use “Remove” button to remove selected entities from text field.

You can run the query by clicking “Execute” button (Figure 32). You can also return to default values of parameters by clicking “Default” button, which are “current view”, 1 for length limit (selected), 0 for shortest+k, and unchecked for ignore source-source/target-target paths.

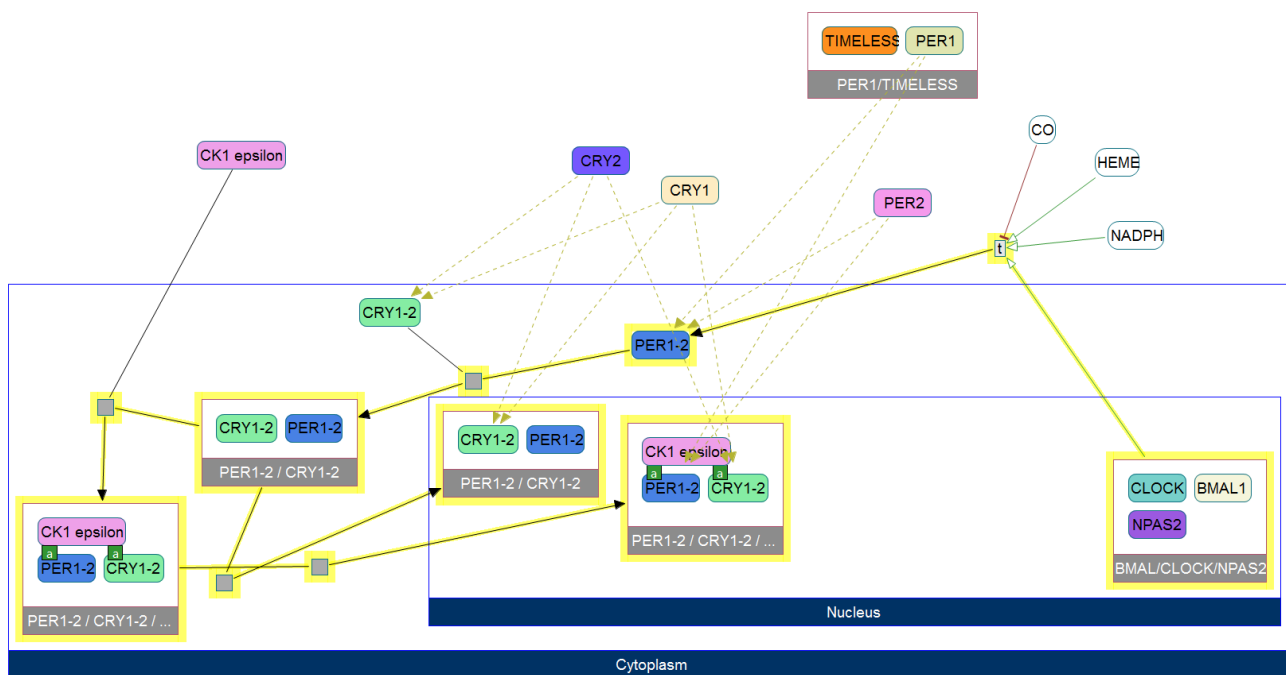
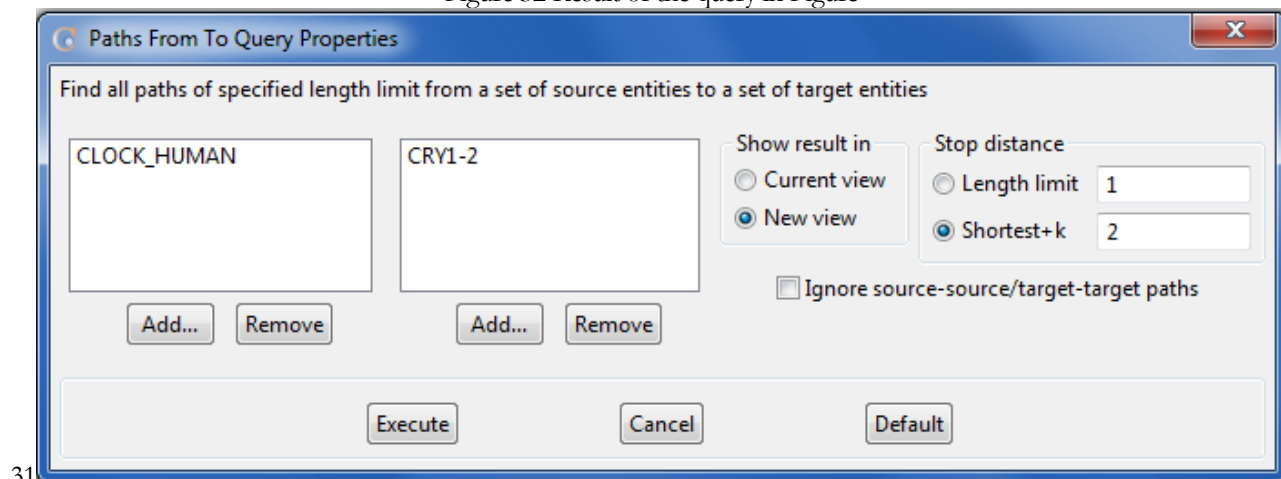


Figure 32 Result of the query in Figure



31

Figure 31. Paths from CLOCK_HUMAN to CRY1-2 within specified length limit is shown (yellow)

Common Stream Query

Common stream query enables one to examine molecules that are affecting and/or affected by a specific set of molecules in common; you will be able to see paths that are diverging from a molecule or converging into a molecule. These paths are useful when you are looking for intersection points of different molecules.

To perform this query through menubar, you can use “Query | Local | Common Stream ...” item, which will lead you to a window to set the parameters of the query (Figure 33).

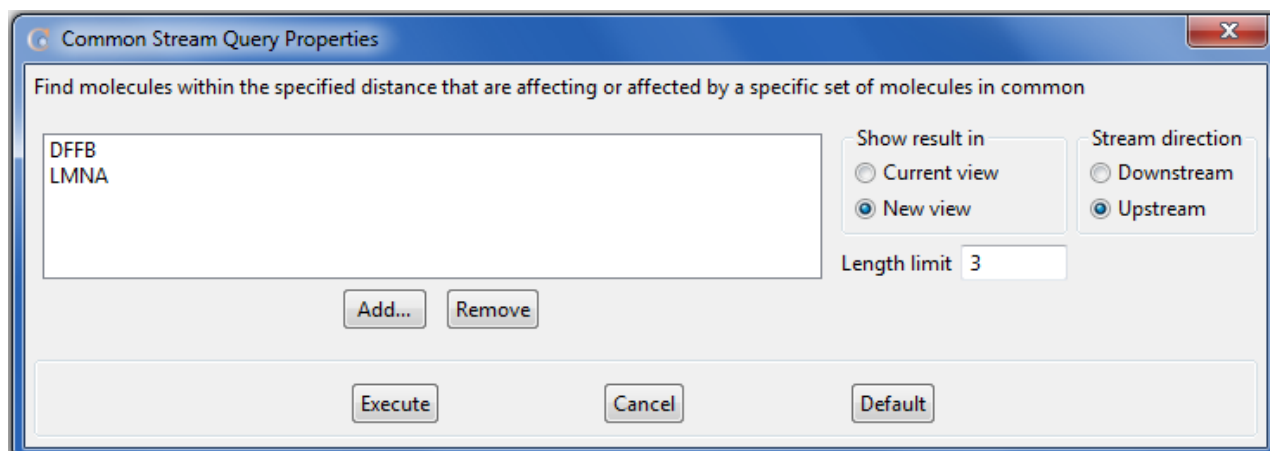


Figure 33 Sample Common Stream Query Properties dialog (entity based)

Following parameters should be specified:

- **Show result in:** This selection between current view and new view determines how to display query result; merging into existing view or showing in a new canvas, respectively.
- **Stream direction:** This parameter will determine the direction of common stream query. Selecting *downstream* will result in looking for common molecules only in downstream of the chosen entities. Selecting *upstream* will result in looking for common molecules only in upstream of the chosen entities.
- **Length limit:** This parameter is a non-negative integer that limits maximum distance of a molecule from selected entities in order to be included in result.

Text field in the left half of dialog is the list of selected entities that will be source entities of common stream query. In order to add an entity to that field, user should click “Add...” button which will open a new window that displays all entities found in that particular graph (Figure 23). You can choose any entity from this dialog and upon clicking “Add” button; selected entity will be added to text field in Figure 33. Multiple selections are possible. You can type a text in text field at the uppermost of dialog and then click “Filter” button in order to visualize entities only which contain that text in their name and synonyms. Clicking “Show All” button will return list of entities into its initial state. After adding entities to the text field, you can use “Remove” button to remove selected entities from text field.

You can run the query by clicking “Execute” button (Figure 34). You can also return to default values of parameters by clicking “Default” button which are “current view”, “downstream”, and 1, respectively.

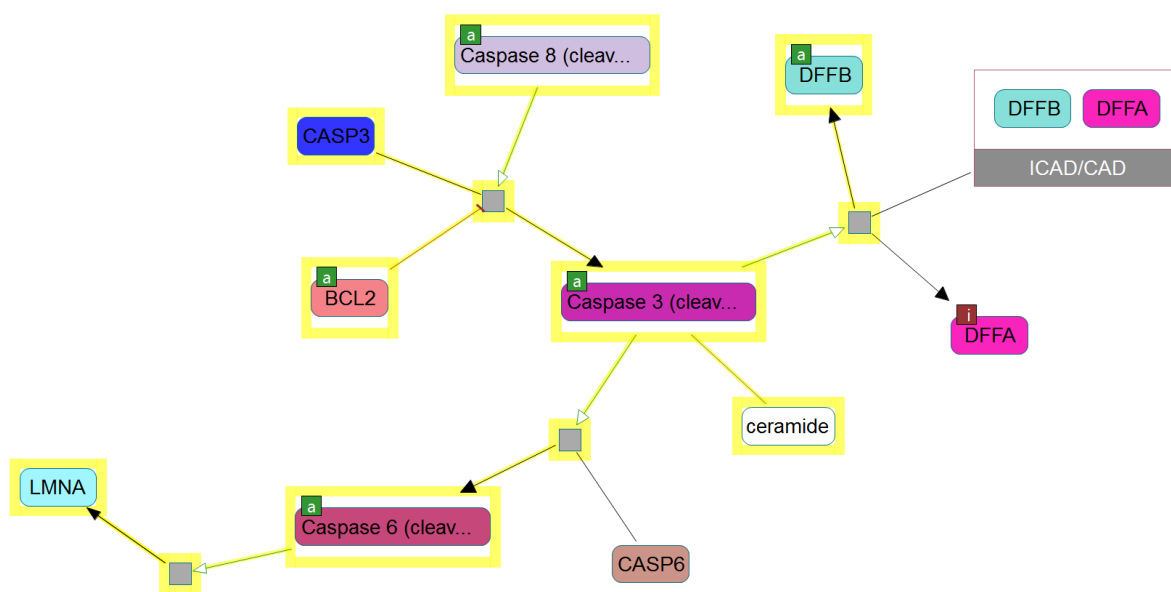


Figure 34 Result of the query in Figure 33; molecules in common upstream of LMNA and DFFB are shown (yellow)

As explained above, you can also perform common stream query through node pop-up menu. Select the source states on the view and right click on a selected state. Choosing “Local Query | Common Stream ...” in pop-up menu will open a dialog to obtain remaining parameters of the query (Figure 35). These are the same parameters as in Figure 33.

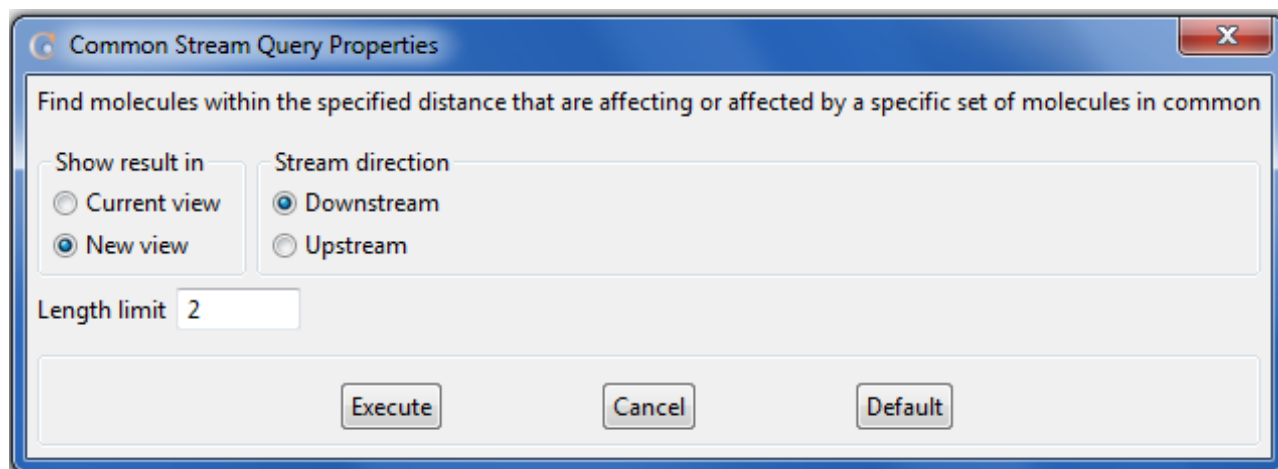


Figure 35 Sample Common Stream Query Properties dialog (state based)

You can run the query by clicking “Execute” button (Figure 36).

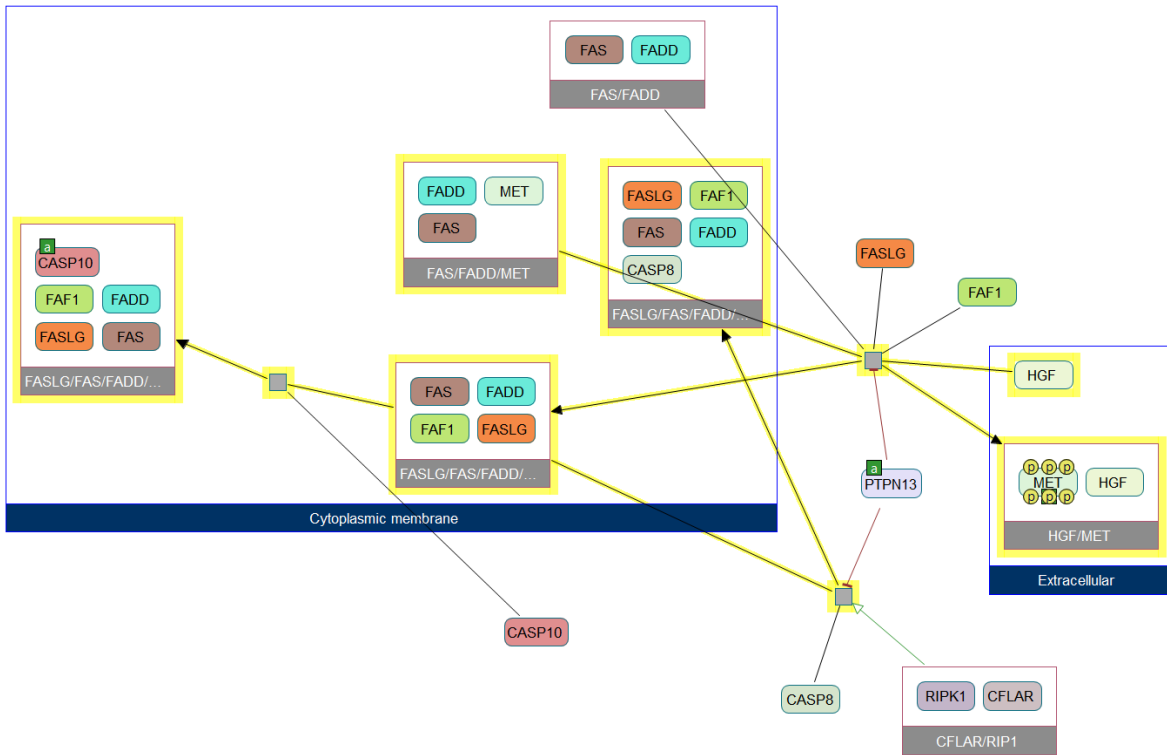


Figure 36 Result of the query in Figure 35.; molecules in common downstream of HGF and FAS/FADD/MET are shown (yellow)

Compartment Query

Compartment query finds paths that originate in a specified compartment and ends in another specified compartment.

To perform this query through menubar, you can use “Query | Local | Compartment ...” item which will lead you to a window to set the parameters of the query (Figure 37).

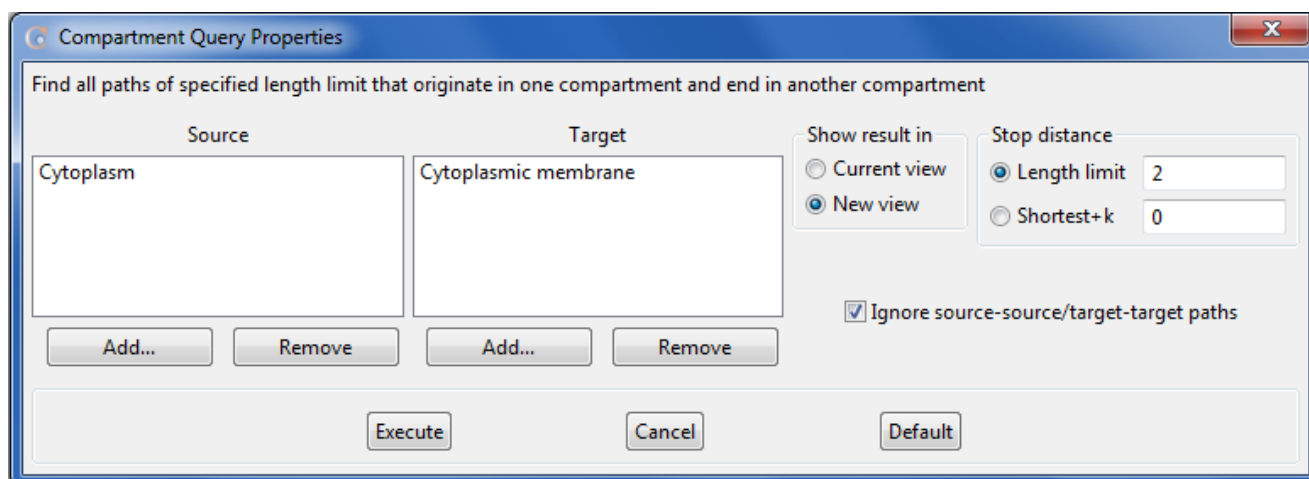


Figure 37 Sample Compartment Query Properties dialog

Following parameters should be specified:

- **Show result in:** This selection between current view and new view determines how to display query result; merging into existing view or showing in a new canvas, respectively.
- **Stop distance:** *Length limit* should be a nonnegative integer that determines length of the longest path found between compartments of interest. For *shortest+k*, the user should specify a nonnegative integer that will be added up with the length of shortest path to determine length of the longest path found between compartments of interest.
- **Ignore source-source/target-target paths:** When this parameter is checked, paths between source molecules and between target molecules are not included into query result; you will only see paths from source to target compartments.

First text field in the left half of the dialog is the list of the selected compartments that will generate source entities of compartment query. Second text field is the list of the selected compartments that will generate targets of compartment query. In order to add a compartment to those fields, user should click “Add...” button which will open a new window that displays all compartments found in that particular graph (Figure 38). Upon choosing a compartment from this dialog and clicking “Add” button, selected compartment will be added to text field in Figure 37. Multiple selections are possible. You can use “Remove” button to remove selected compartment from text field.

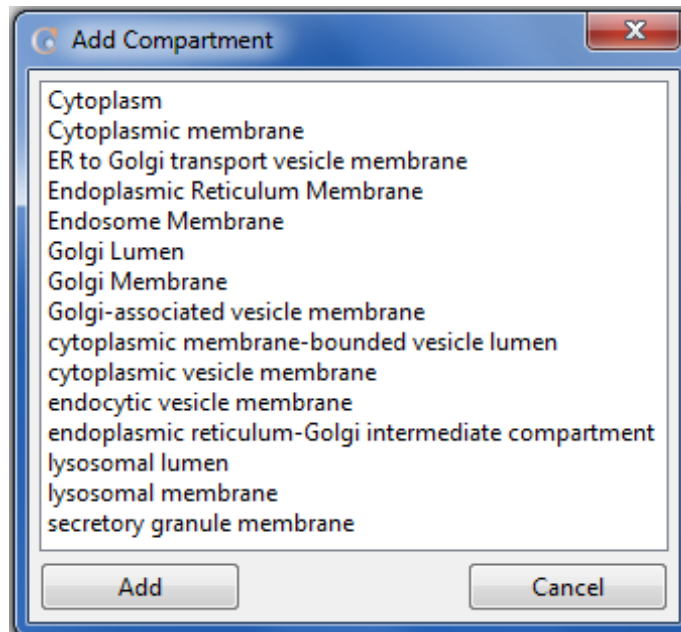


Figure 38 Add Compartment dialog that is used to add compartments into the list found in query properties dialogs

You can run the query by clicking “Execute” button (Figure 39). You can also return to default values of parameters by clicking “Default” button which are “current view”, 1 for length limit (selected), 0 for shortest+k, and unchecked for ignore source-source/target-target paths.

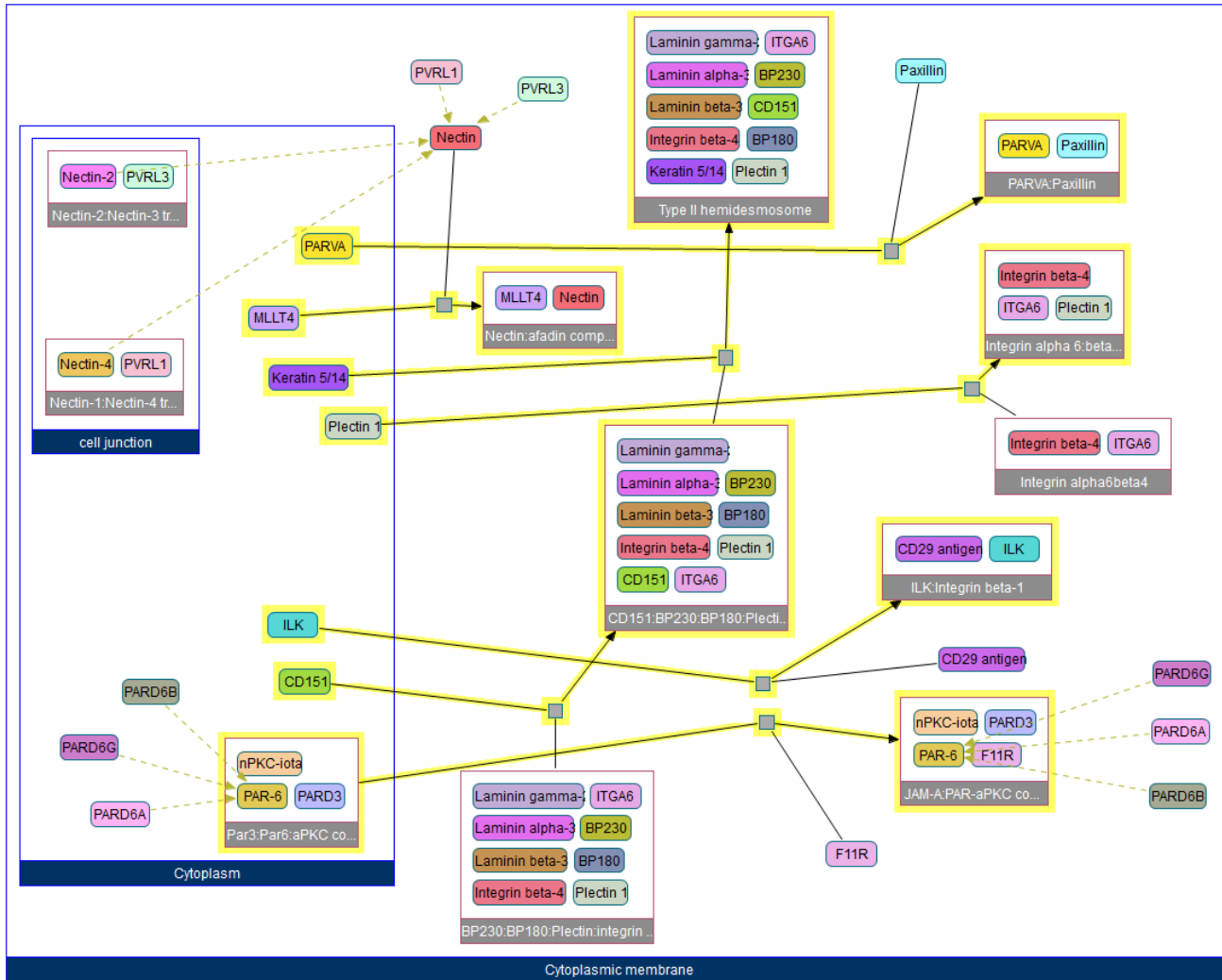


Figure 39 Result of the query in Figure 37; paths from cytoplasm to cytoplasmic membrane with maximum length of 2 are shown (yellow)

Path Iteration Query

Path iteration query is useful when you have little idea about where you want to get started and do not know which specific query will help you. In this query, we offer the user some of the long paths in the model, from which user can select the one that will be visualized. To be specific, we want these paths to be short enough not to confuse the user but long enough to be informative. Thus, this query first computes all pairs shortest paths and lists some of (a fixed number of) the longest of such paths.

To perform this query through the menubar, you can use “Query | Local | Path Iteration ...” item, which will lead you to a window that displays pairs of entities with length of distance between them (Figure 40).

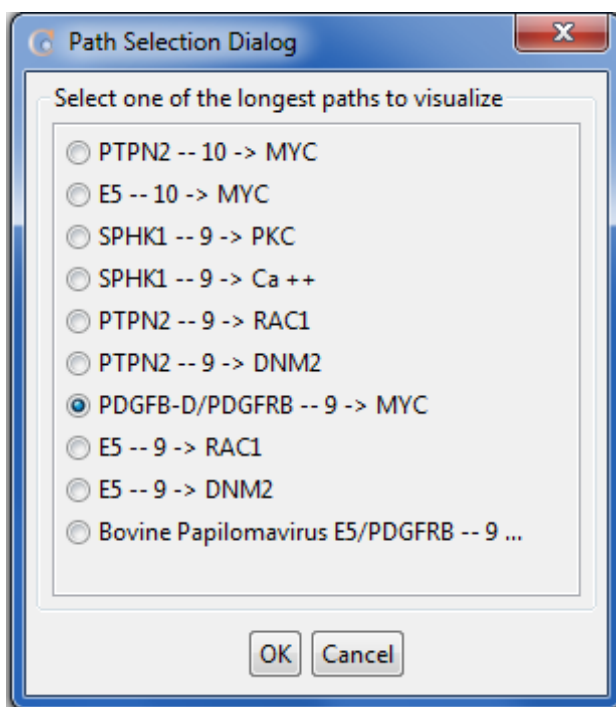


Figure 40 Sample Path Selection Dialog

The user can choose one of these listed paths and upon clicking “OK” button, the resulting shortest path will be visualized in a new view (Figure 41).

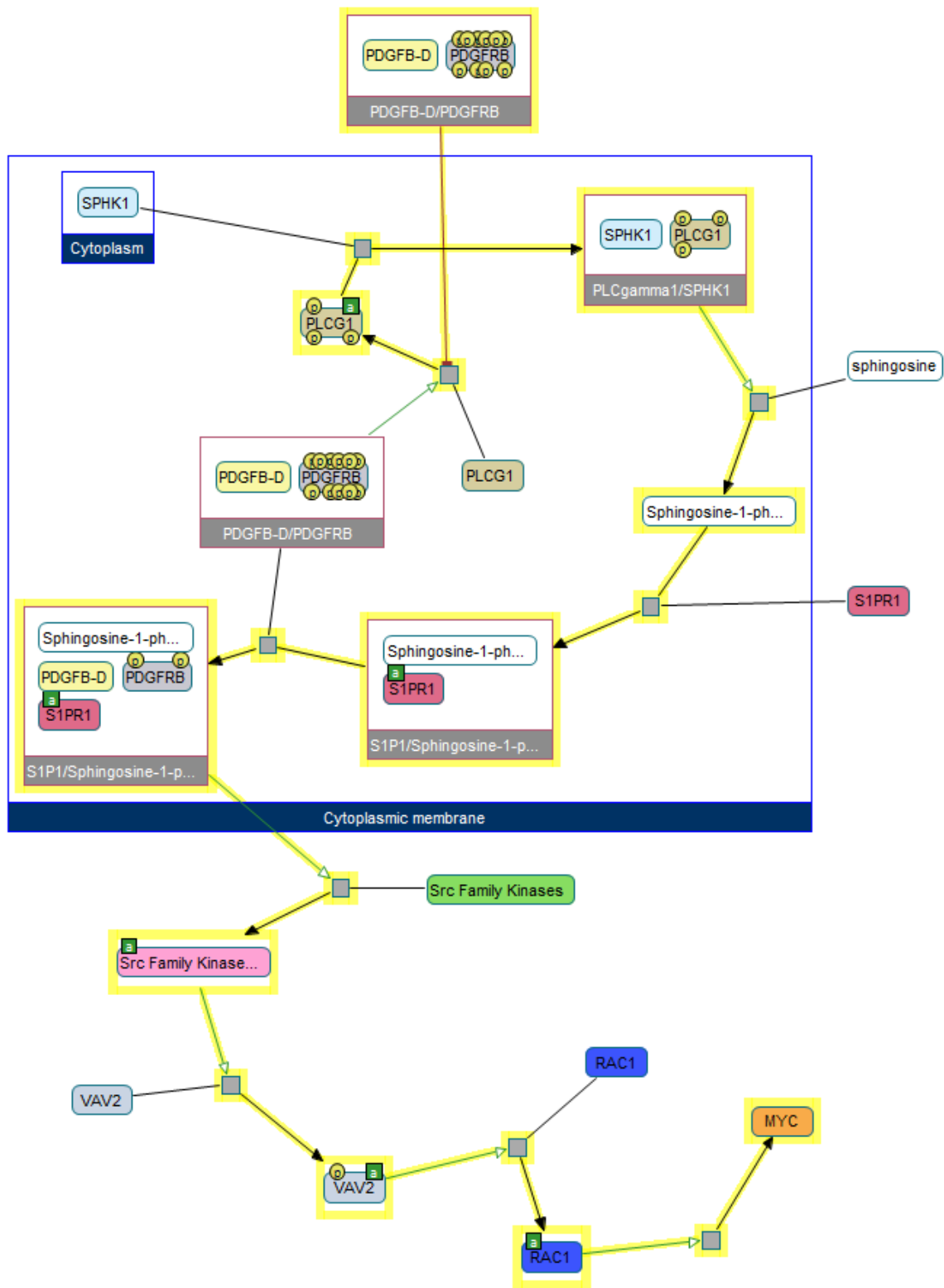


Figure 41 Result of the query in Figure 40; path starting from PDGFB-D/PDGFRB and ending at MYC with length of 9 is shown (yellow)


Querying Pathway Commons Database

Pathway Commons [2] is a convenient point of access to biological pathway information collected from public pathway databases, which you can browse or search. ChiBE provides a graphical user interface to search this knowledgebase and present the results in a visual form. As opposed to local queries, which help the user identify significant parts of the current model, these queries are aimed at forming a new pathway model or merging query results with the current model.


Pathway Commons Queries (Level 2)

Queries under this menu item return results in BioPAX Level 2 format and therefore, they cannot be used when there is a previously loaded Level 3 model and their results cannot be merged with Level 3 models.

QUERYING NEIGHBORHOOD

ChiBE provides a mechanism to search all pathways in Pathway Commons database for finding neighbors (i.e. directly related molecules) of a specified molecule. To perform neighborhood query, use “Query | Pathway Commons (Level 2) |  Neighborhood ...”, and when prompted, enter the UniProt ID or Entrez Gene ID of the protein or the gene, whose neighborhood is to be searched for. The result is presented to the user in a new pathway view with name “Neighborhood of < ID>”.

QUERYING PATHWAYS

One can also search for all pathways containing information about a specific molecule by using “Query | Pathway Commons (Level 2) |  Pathways ...” menu item. The molecule may be specified using either its UniProt or Entrez Gene ID. All pathways containing the specified molecule are listed in a result dialog and the user is prompted to choose the ones for which views are to be created.


Pathway Commons Queries (Level 3)

Queries under this menu item return results in BioPAX Level 3 format and therefore, they cannot be used when there is a previously loaded Level 2 model and their results cannot be merged with Level 2 models.

Four of the six queries offered for Level 3 are analogous to their counterpart local queries: *Neighborhood*, *Paths Between*, *Paths From To*, and *Common Stream* queries. In contrast to their counterpart local queries, however, these queries require the user to supply HGNC symbols [18] for genes of their interest.

NEIGHBORHOOD QUERY

With this query, the user can search Pathway Commons Database to obtain neighbors of a specific set of genes within a certain distance limit and in a certain direction.

To perform this query, you can use “Query | Pathway Commons (Level 3) |  Neighborhood ...” menu item. The dialog seen in Figure 42 will be displayed for you to specify the parameters of the query.

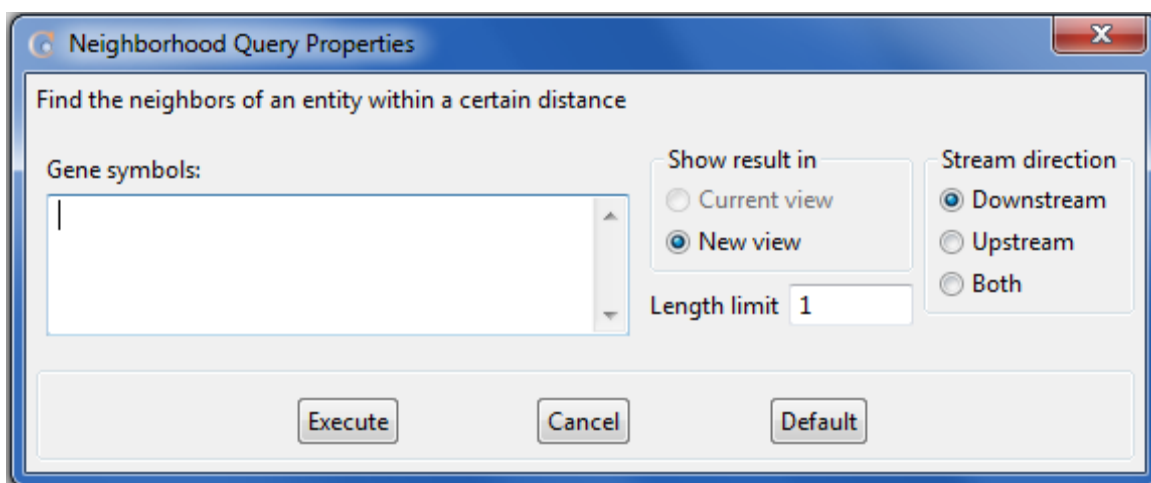


Figure 42 Sample Neighborhood Query Properties dialog

HGNC symbols of gene(s) that you are interested in should be specified in “Gene Symbols” text field. If the user is interested in neighbors that are found in the downstream of the specified gene(s), then “Downstream” option from “Stream Direction” group must be selected. Similarly, “Upstream” option must be selected to obtain neighbors in the upstream. “Both” option can be selected if you are interested in both directions. For length limit, user needs to specify a nonnegative integer that will determine the upper limit of distance from the specified set of genes. After specifying all required parameters, user can run the query by clicking on “Execute” button. The resulting view will be displayed in a new view by default.

Alternatively, you can perform this query through the node pop-up menu by selecting “Pathway Commons Query (Level 3) | Neighborhood”. Three different direction options are offered under this menu. After selecting gene(s) of interest on the view, you should choose one of these three directions possibilities: Downstream, Upstream or Both Streams. Then a neighborhood query with predefined length limit of 1 will be run in the specified direction. The result will be displayed in a new view.

PATHS BETWEEN QUERY

Paths Between query can be used to obtain the network composed of directed paths between a set of source entities from Pathway Commons Database.

To perform this query, “Query | Pathway Commons (Level 3) | ? Paths Between ...” will be used. The dialog shown in Figure 43 will be displayed upon clicking this menu item.

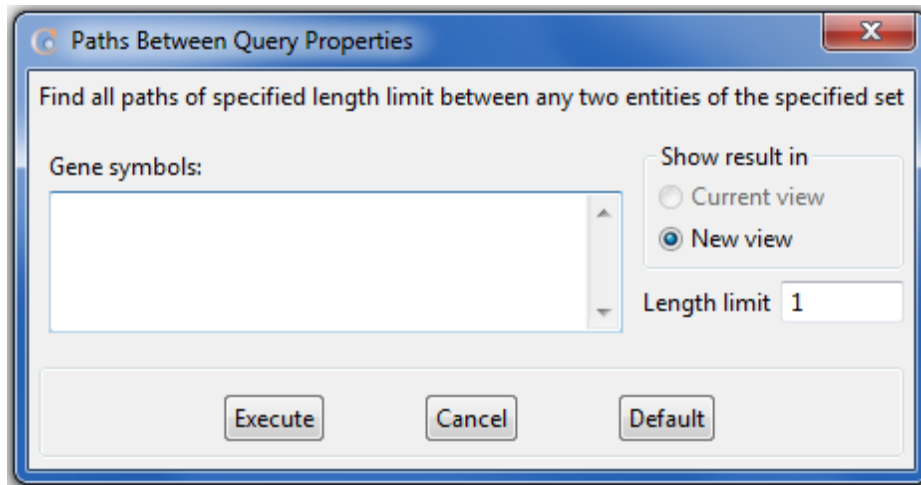




Figure 43 Sample Paths Between Query Properties dialog

HGNC symbols of genes of interests will be written to “Gene symbols” text field. You need to specify at least two genes. The only other required parameter is length limit. This must be a nonnegative integer that will determine the length of longest path found between specified set of genes. After specifying all required parameters, user can run the query by clicking on “Execute” button. The result will be displayed in a new view by default.

This query can also be performed by using “Pathway Commons Query (Level 3) |  Paths Between ...” menu item found in pop-up menu. User first needs to select at least two genes on the view. Then clicking on this menu item will run the default query on selected genes with length limit specified as 1. The result will be displayed in a new view.

PATHS FROM TO QUERY

Paths From To query is similar to Paths Between query. In this case, there are two distinct sets specifically defining source and target genes.

To perform this query, “Query | Pathway Commons (Level 3) |  Paths From To ...” will be used. The dialog shown in Figure 44 will be displayed to obtain query parameters.

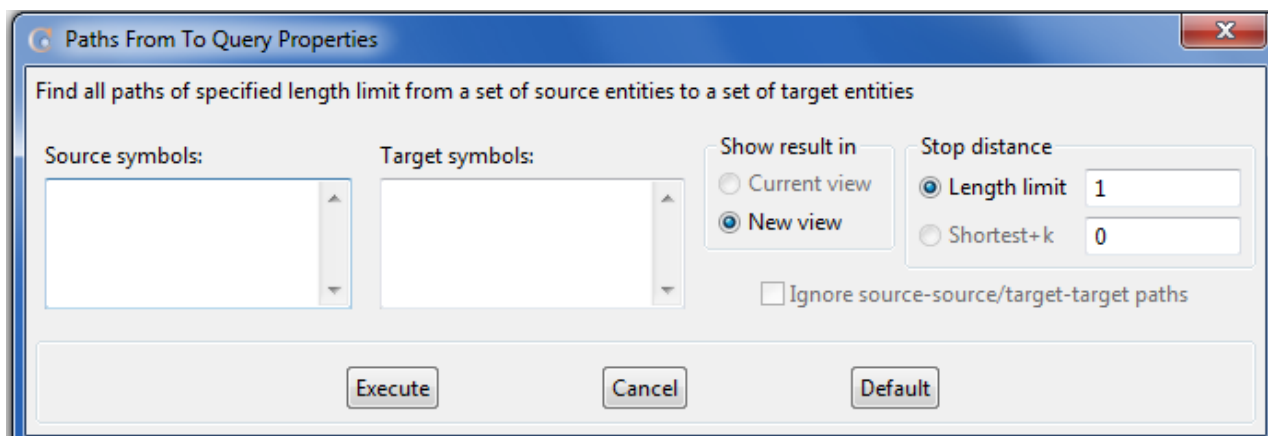



Figure 44 Sample Paths From To Query Properties dialog

HGNC symbols of source and target sets will be specified in respective text fields. At least one gene is required for both source and target symbols. A nonnegative integer is required as length limit to determine the length of longest path found from source set to target set. After required parameters are specified, user can run the query by clicking on “Execute” button. The result will be displayed in a new view.

COMMON STREAM QUERY

This query is useful to obtain genes that are found in the downstream or upstream of a specified set of genes. The resulting genes are in up/downstream of all entities in the initial set; partially common genes are not identified as common stream.

To perform Common Stream query, “Query | Pathway Commons (Level 3) |  Common Stream ...” will be used. The dialog shown in Figure 45 will be displayed upon clicking this menu item.

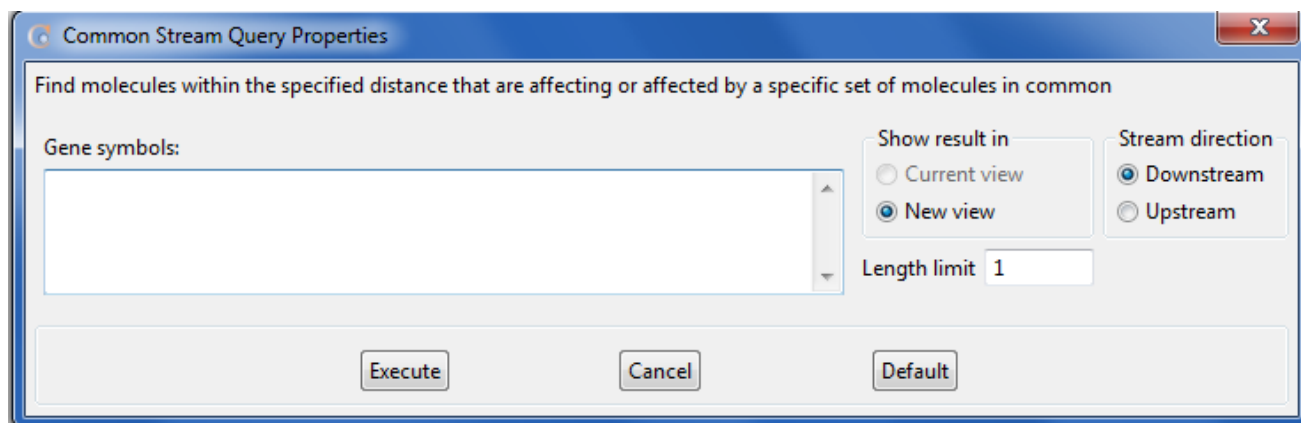





Figure 45 Sample Common Stream Query Properties dialog

HGNC symbols of genes of interest should be specified in “Gene symbols” text field; at least two entities are required. Stream direction needs to be specified as either “Downstream” or “Upstream”. Depending on this choice, the direction of query will be determined. A nonnegative integer is required as length limit in order to


determine the upper bound of distance from set of source entities. After required parameters are specified, user can run the query by clicking on “Execute” button. The result will be displayed in a new view by default.

Alternatively, pop-up menu can be used to perform common stream queries. There are two menu items for this query: “Pathway Commons Query (Level 3) | Common Stream |  Downstream” and “Pathway Commons Query (Level 3) | Common Stream |  Upstream”. After selecting at least two states on the view, “Downstream” can be used to search for common downstream molecules with length limit specified as two. “Upstream” alternative offers the same query but in upstream direction. The result will be displayed in a new view.

PATHWAYS WITH KEYWORD

All previously described queries return sub-pathways based on a set of genes but users can also perform queries to obtain whole pathways. For this purpose, “Query | Pathway Commons (Level 3) |  Pathways With Keyword ...” menu item can be used. Upon clicking this item, user will be asked a keyword that will be used for search. Pathways containing this keyword in their names will be listed in a result dialog and then user can select one of these pathways for which a view will be created.

OBJECT WITH DATABASE ID

ChiBE also enables users to obtain specific objects from Pathway Commons Database. “Query | Pathway Commons (Level 3) |  Object With Database ID ...” menu item can be used for this purpose. Type Pathway Commons Database ID of the object when prompted and the object will be displayed by ChiBE. However, there is no guarantee that it will be viewable in ChiBE.

SIF Operations

In this chapter, we introduce the SIF views and operations on them.

SIF (simple interaction format) is a simple format introduced by Cytoscape [11] for describing interactions in a biological network. ChiBE supports a second type of view for displaying interactions in SIF as well as mechanisms for persisting models in this format.

Simple Interaction Format

SIF is a format for expressing interactions in a biological network in a simple way. Each line of a SIF file is typically of the following format:


```
nodeA <relationship type> nodeB
```

where a relationship or interaction of the specified type exists between two biological entities represented with nodes A and B.

SIF Views in ChiBE

ChiBE supports SIF views as simplified views of pathway models.

Creating SIF Views

You may create a SIF view of the current pathway model by choosing “SIF |  Create SIF View ...” from the menubar. This prompts you to select one or more rules (Figure 46) to determine the type of interactions to include in the newly created view.

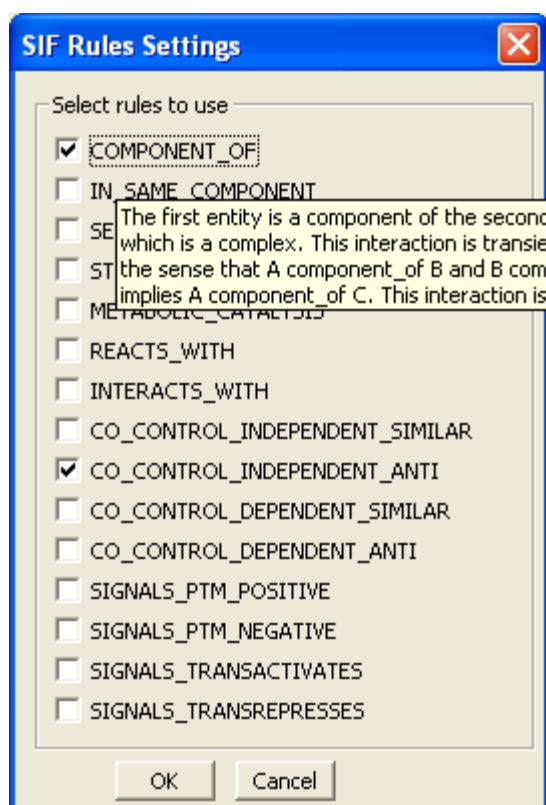


Figure 46 SIF Rules Settings Dialog. Any number of rules may be included to construct a SIF view. Tooltips provide a more complete description of the rules.

Following are possible choices for rules:

- **COMPONENT_OF**: The first entity is a component of the second entity, which is a complex. This interaction is transient in the sense that A component_of B and B component_of C implies A component_of C. This interaction is directed.
- **IN_SAME_COMPONENT**: Two entities belong to the same molecular complex. This does not necessarily mean they interact directly. In a complex with n molecules, this rule will create a clique composed of $n(n-1)/2$ interactions. This interaction is directed.
- **CO_CONTROL_DEPENDENT_SIMILAR**: This rule infers an interaction if the first and second entities have control over the same process, their control is dependent, i.e. one of them have effect over control of the other one, and their effect is in the same direction (both activates or both inhibits). This interaction is undirected.
- **CO_CONTROL_DEPENDENT_ANTI**: This rule infers an interaction if the first and second entities have control over the same process, their control is dependent, i.e. one of them have effect over control of the other one, and their effect is in different directions (one of them activates, the other inhibits). This interaction is undirected.
- **CO_CONTROL_INDEPENDENT_SIMILAR**: This rule infers an interaction if the first and second entities have control over the same process, their control is independent, i.e. they act without affecting

each other's activity, and their effect is in the same direction (both activates or both inhibits). This interaction is undirected.

- CO_CONTROL_INDEPENDENT_ANTI: This rule infers an interaction if the first and second entities have control over the same process, their control is independent, i.e. they act without affecting each other's activity, and their effect is in different directions (one of them activates, the other inhibits).
- SEQUENTIAL_CATALYSIS: The entities catalyze two conversions that are connected via a common molecule, e.g. the first entity produces a substrate that is consumed by the second entity. This interaction is directed.
- METABOLIC_CATALYSIS: The first entity catalyzes a reaction that either consumes or produces the second entity. This interaction is directed.
- STATE_CHANGE: The first entity controls a reaction that changes the state of the second entity, e.g. by phosphorylation or other posttranslational modification, or by a change in subcellular location. This interaction is directed.
- REACTS_WITH: The entities participate in a conversion as substrates or products. Controllers are not included. This interaction is undirected.
- INTERACTS_WITH: The entities participate in an interaction. Controllers are not included. This interaction is undirected.

Figure 47 illustrates SIF views with an example.

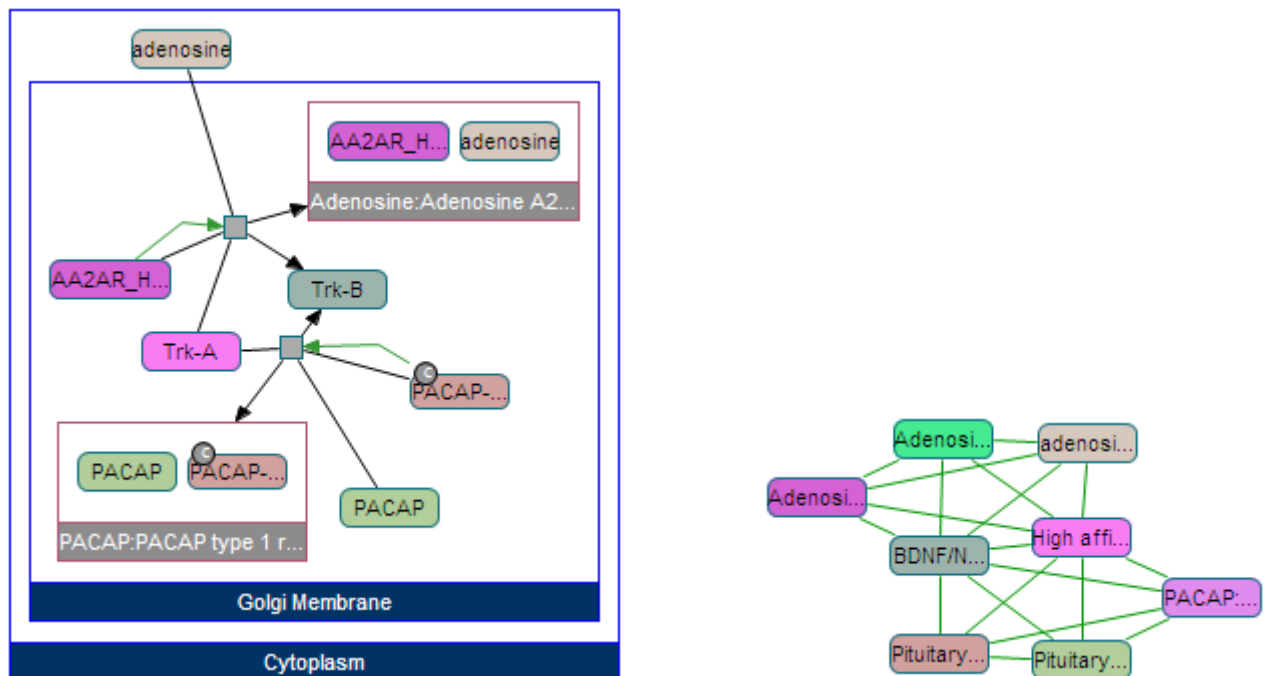




Figure 47 A process diagram view (left) and its SIF view (right) created with “REACTS_WITH” rule.

Persisting SIF Views

SIF views may be saved on disk and loaded back later as desired using “SIF |  Save As SIF File ...” and “SIF |  Load SIF File ...” menu items, respectively.

Notice that SIF views lose their association with current pathway model once they are persisted on disk. Thus, any experiment data loaded into ChiBE (see Chapter titled Visualizing Experiment Data on page 51 for details) will not be overlaid on SIF views unless the SIF view is directly created from the current pathway model (as opposed to being loaded from disk).

Working with Large SIF Files

Loading an entire SIF file may lead to very long layout execution time and the resulting view could be cluttered. ChiBE lets users specify a set of genes of interest, and shows the subgraph that is created with the paths between these genes of interest.

Subgraph of interest may be created using “SIF | Paths Between Query on SIF File ...” menu item. This opens a dialog that asks for the SIF file name, a genes-of-interest file, and several options for the operation. Genes-of-interest file must be a text file that contains the names of the genes, one per line, and names that would match with the ones used in the SIF file. Options include a *search distance* (i.e. the limit on the size of the path between two genes of interest), and *types of SIF edges* to be retrieved. The “*directed*” checkbox is not editable by the user, and is checked whenever the selected SIF edge types consist of only directed edge types indicating that the path search will be executed in a directed manner.

Visualizing Experiment Data

In this chapter, we describe how experiment data can be visualized within ChiBE.

Experiment data such as gene expression profiles can be loaded into ChiBE and overlaid onto views of the currently loaded pathway model.

ChiBE has a native expression data file format (.ced) and supports conversion of tab-delimited data files into this format. This is an XML-based format containing information about the experiments (description and values to be analyzed). Previously created “.ced” files can be directly loaded into ChiBE. Upon load, all visualized objects are associated with the first experiment in the loaded data file. The type of experiment data that could be visualized with ChiBE includes

- Expression data
- Mass spectrometry data
- Copy number variation data
- Mutation data
- Alteration data

As the pathway models change, on new queries for instance, the loaded data is associated to any newly introduced objects whenever possible. Notice here that the reference mapping is performed based on the pathway model at the time of the load operation. Thus, any new pathway objects that do not use these references will not have any associated experiment data.

Loading Experiment Data

ChiBE supports loading tab-delimited experiment data files using “Data | Load Raw Data ...” menu item. Examples of such files include NCBI GEO [12] files and Stanford Microarray Database [13] files. During this load operation, the user is asked to specify the columns of references and experiments in the file(s) by the “Experiment Data Conversion Wizard” (Figure 48), and then the data in tab-delimited file(s) is extracted and converted into ChiBE experiment data format (.ced) for later convenient use.

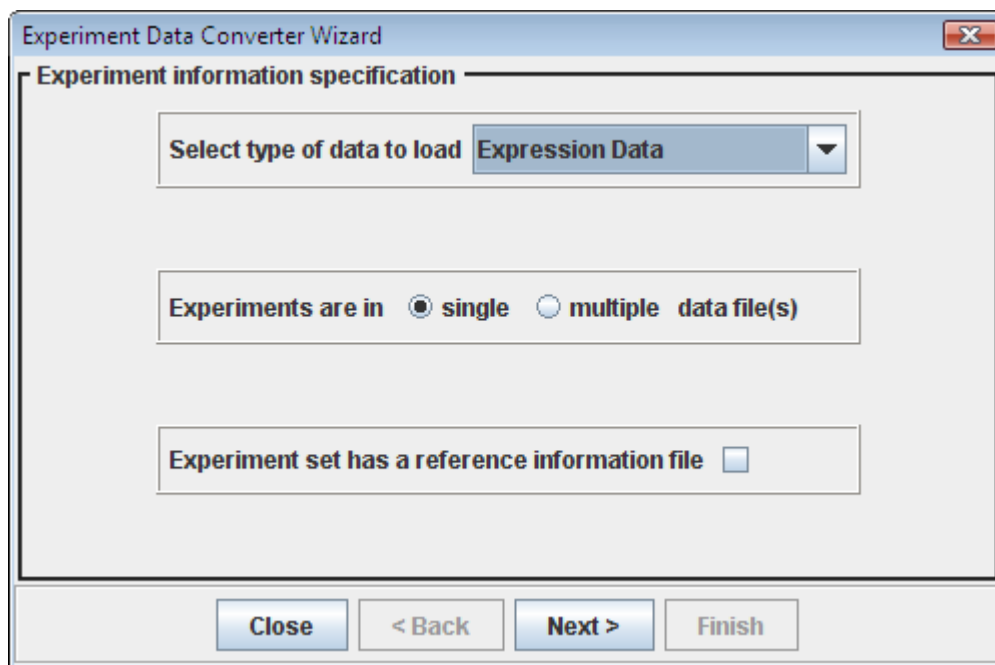


Figure 48 ChiBE Experiment Data Conversion Wizard

The tab-delimited data file should contain some external reference IDs that will be matched with corresponding external reference synonyms. These predefined external references are stored in a text file along with their widely used synonyms encountered in different experiments.

After the conversion is performed, a “.ced” file is prepared for local storage.

A tab-delimited file consists of rows and columns. Typically, each row represents a gene and the columns along a row represent the values measured. These values might be normalized amounts measured in different experiments or different types of measurements extracted from a single experiment.

Reference Matching

Sometimes it is reasonable to specify references of rows in a file, independent from specific experiments. Reference files, for instance, may contain many external references that correspond to some spots in the microarray chip, which are specified by an internal ID. Reference files help reduce size of data files by not repeating external references. Platform file in GEO database is an example of reference file.

Reference file:

Internal ID	GenBank Accession	Gene Symbol	Unigene
473847	AB000114	OMD	Hs.94070
473625	AB000115	IFI44L	Hs.389724

Data files:

Internal ID	Value
473847	100.4
473625	152.7

Internal ID	Value
473847	20.3
473625	263.1

When there is no reference file, these references must be written into data files.

Data files:

Internal ID	GenBank Accession	Gene Symbol	Unigene	Value
473847	AB000114	OMD	Hs.94070	100.4
473625	AB000115	IFI44L	Hs.389724	152.7

Internal ID	GenBank Accession	Gene Symbol	Unigene	Value
473847	AB000114	OMD	Hs.94070	20.3
473625	AB000115	IFI44L	Hs.389724	263.1

Converter needs external references to map rows with pathway objects, and supports both kinds of formats.

Multiple experiments

Several experiment results may be presented in a single data file or each experiment may be given in a separate data file (as given above). When given in a single file, values of each experiment are presented in a separate column.

Two experiments in a single file (assuming there is a reference file not given here):

Internal ID	Experiment 1	Experiment 2
473847	100.4	20.3
473625	152.7	263.1

Using Conversion Wizard

In the first page, the user picks the type of data, specifies if experiments are given in a single or multiple data files, as well as whether or not there is a reference file (Figure 48). The following wizard pages appear accordingly as described below.

SINGLE DATA FILE WITH A REFERENCE FILE

When there is a reference file and data is in a single file, then the wizard prompts for the reference file, parses it and maps its columns to some known references (

Figure 49). Some column names may be automatically identified, whereas others should be mapped by the user. The “Key to Data File(s)” item should be mapped to the internal ID column, which data files use.

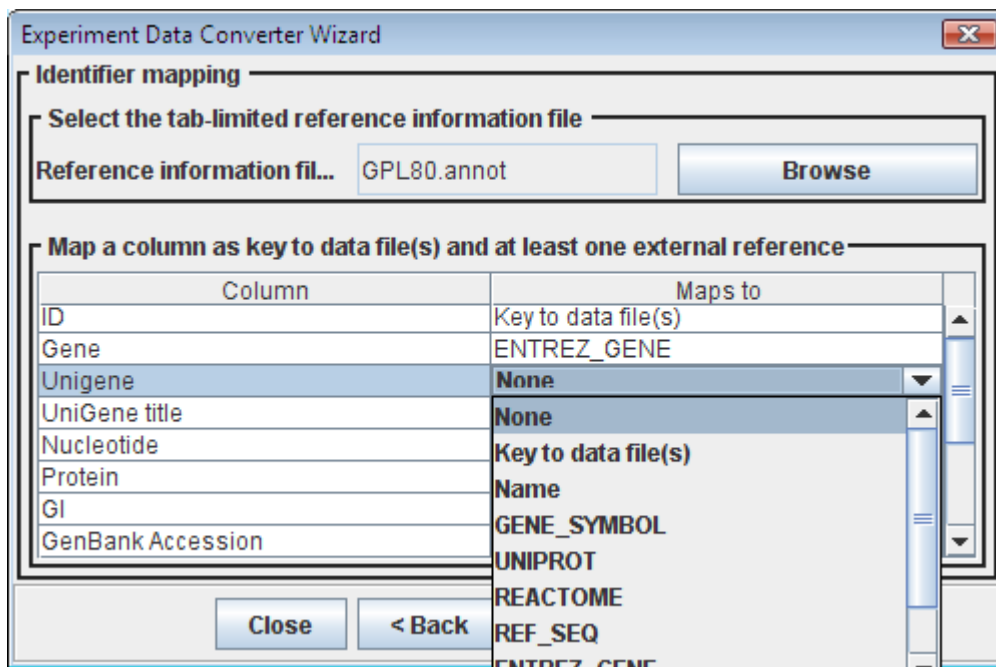


Figure 49 Platform file is parsed and some reference columns are mapped. The “Key to Data File(s)” item should be mapped to the internal ID column, which data files use.

The next page is for specifying the data file to be loaded. After that, internal IDs and experiment value columns are asked for (Figure 50). All experiment values must be in adjacent columns.

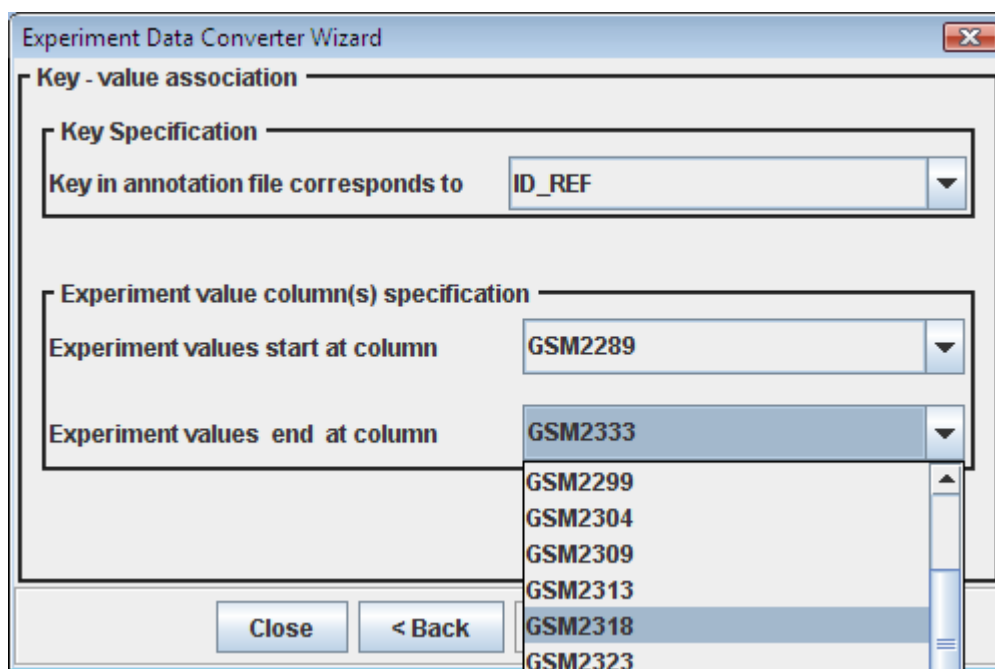


Figure 50 Key – value association with single data file containing multiple experiments. Here “ID_REF” column is said to be the internal ID that related the data file to the platform file.

MULTIPLE DATA FILES WITH REFERENCE FILE

When there is a reference file and each of multiple experiments is given in a separate file, the user is first asked to locate these data files (Figure 51).

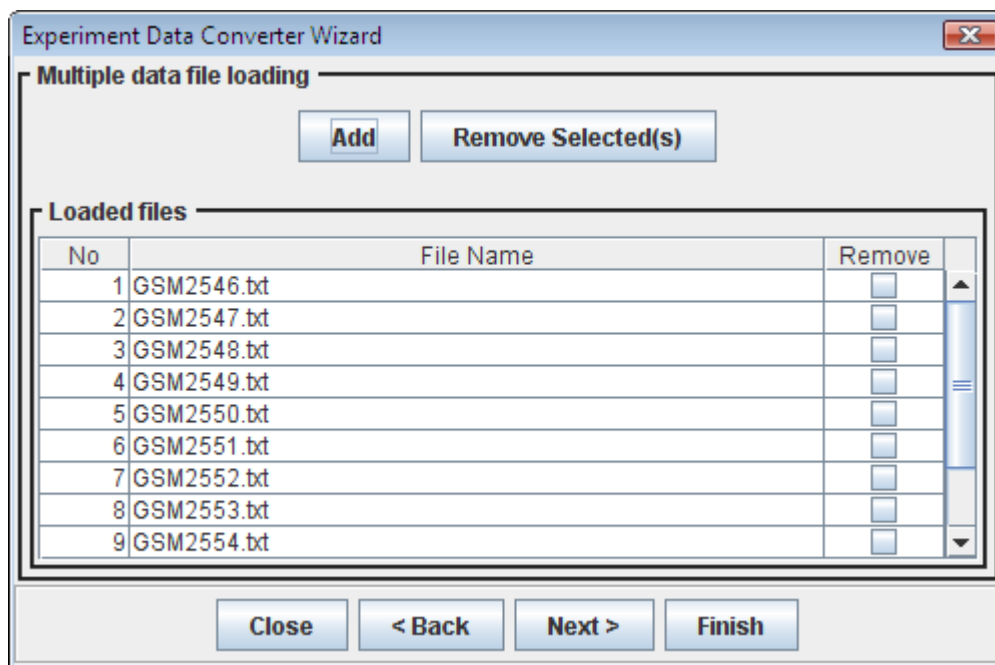


Figure 51 Multiple Data File Loading

Then the internal ID column (key to annotation file) and the value column are set by the user (Figure 52). Here, the format of the data files is assumed to be the same; thus these settings should apply to each loaded data file, otherwise mismatches may occur.

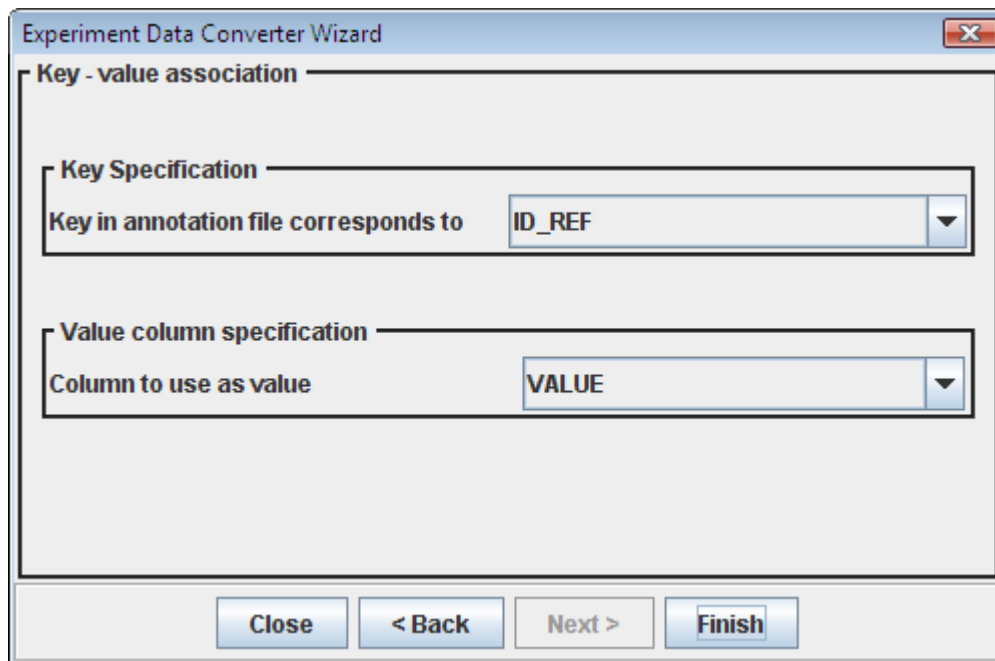


Figure 52 Key – value association with multiple data files

SINGLE DATA FILE WITH NO REFERENCE FILE

When there is no reference file and all data is presented in a single data file, the user should select the data file (Figure 53) and specify the data columns that contain experiment values (Figure 54). Columns containing the experiment values are assumed to be adjacent in the data file. Since there is no reference file, the user should also specify external reference columns.

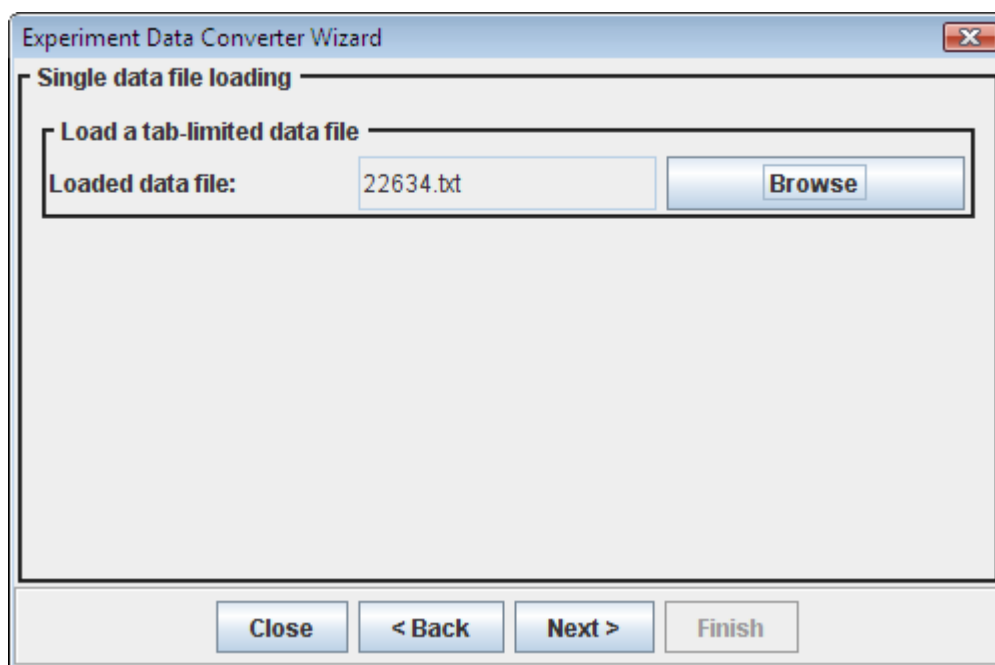


Figure 53 Loading single data file that contains multiple experiments which has no reference file

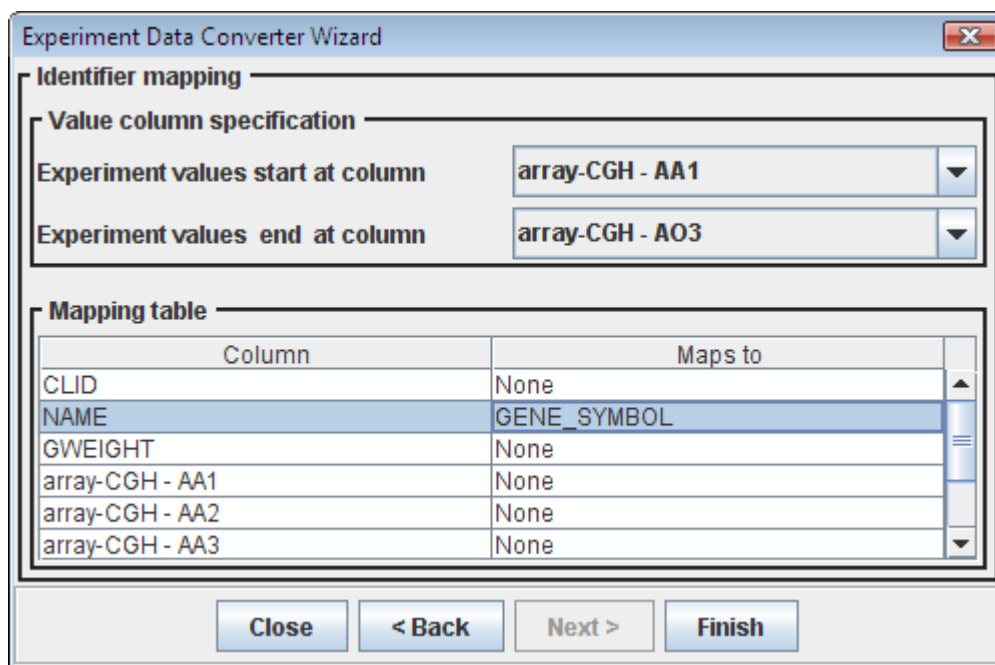


Figure 54 ID mapping when there is a single data file with no reference file

MULTIPLE DATA FILES WITH NO REFERENCE FILE

When there is no platform file and values of each experiment are provided in a separate data file, the user first specifies the data files (Figure 55). Then he selects the value column and external reference columns in data files (Figure 56). The format of all data files should be the same; otherwise files will not load successfully.

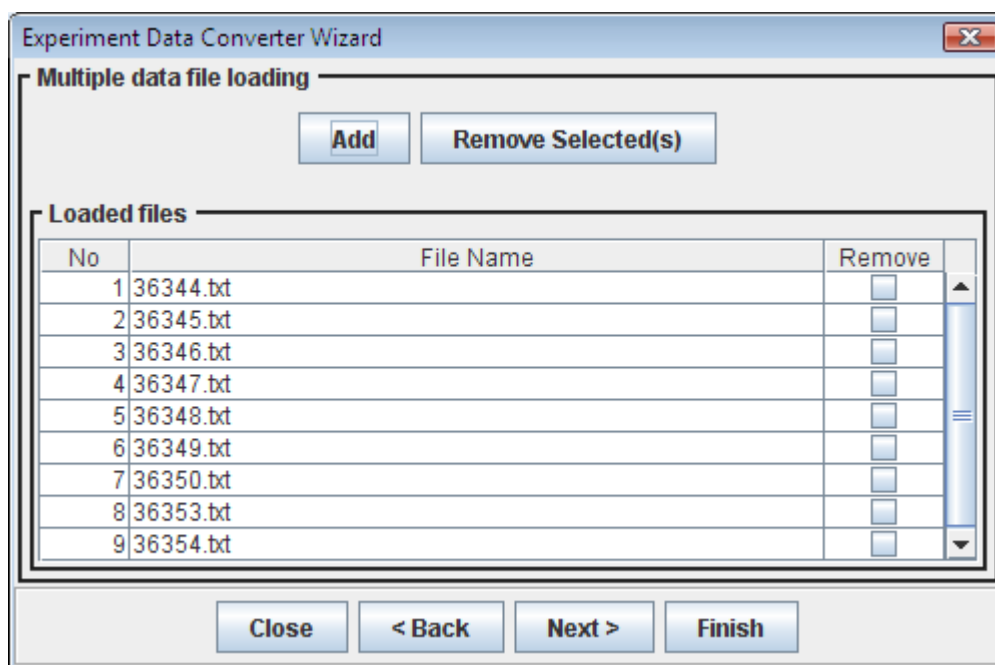


Figure 55 Loading multiple data files with no reference file

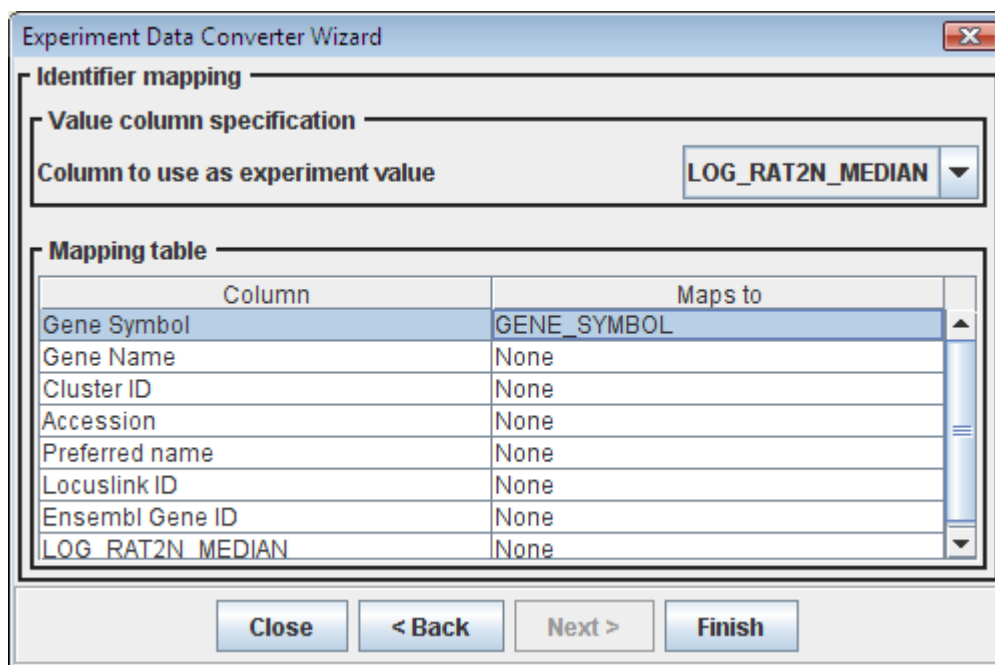


Figure 56 ID mapping when there are multiple files with no reference file

When the raw data file is converted to ChiBE format, the user is asked to save the file with “.ced” extension, which is automatically loaded into ChiBE.

Fetching From GEO

ChiBE offers an alternative way for visualizing expression data on pathways, if an expression data stored in NCBI GEO [12] is to be visualized on a pathway from Pathway Commons [2]. With providing the GEO accession number of series file only, user will be able to visualize the particular expression set in ChiBE.

“Data | Fetch from GEO ...” will open a dialog (Figure 57) which asks the series accession number. You can either type the number in the specified area or choose one from the list below.

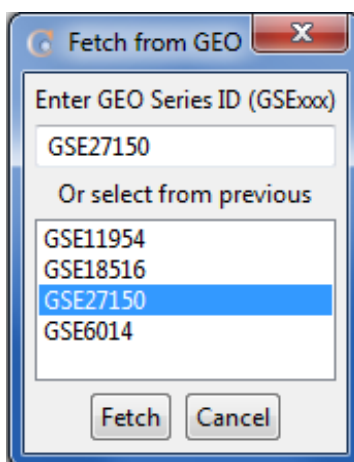



Figure 57 Sample Fetch from GEO dialog


Upon clicking “Fetch” button, ChiBE will perform the required actions based on the files present in the directory. If “.ced” file for that particular dataset is present, then ChiBE will load the specific file. If not, then series matrix file and platform file of the dataset – both downloaded from NCBI GEO [12] – will be used to create a “.ced” file automatically, which is then loaded into ChiBE. As a result, expression values found in the specified microarray dataset will be visualized on the pathway graph as seen in Figure 61.

Loading ChiBE Formatted Data

As desired, “.ced” files can be directly loaded into ChiBE using “Data |  Load ChiBE Formatted Data ...”. Upon load, all visualized objects of the current pathway model are associated with the first experiment in the loaded data file. As the pathway models change, on new queries for instance, the loaded data is associated with any newly introduced objects whenever possible.

Fetching From cBio Portal

cBio Cancer Genomics Portal [14] is a cancer genomics analysis portal, which offers the chance to explore cancer genomes with respect to mutations, copy number alterations, mRNA expression changes, DNA methylation values, and protein levels. ChiBE enables users analysis of such data at pathway level. It collects data from cBio Portal and integrates with the model to display the alterations associated with each entity.

Clicking on “Data |  Fetch from cBio Portal ...” menu item will lead the user to the dialog seen in Figure 58. The user is required to specify the following parameters: *a cancer study*, *a case list*, and *genomic profile(s)*. As seen in Figure 58, you first need to select a cancer study. Currently, there are twenty different cancer data sets. Based on this selection, case list set will be updated and you are expected to choose one of these. This will determine

which subset of the available samples will be used. For instance, if you choose “All Complete Tumors”, only the samples that have mRNA, CNA and sequencing data will be taken into consideration. As the last step, you are required to choose genomic profile. There are numerous alternatives based on the cancer type; some of which are putative copy-number alterations, mRNA expression and mutations. For this option, multiple selections are possible. After specifying these three parameters and clicking on “Load Data” button, each node will be colored according to the alteration frequency which is calculated based on the data obtained from cBio Portal.

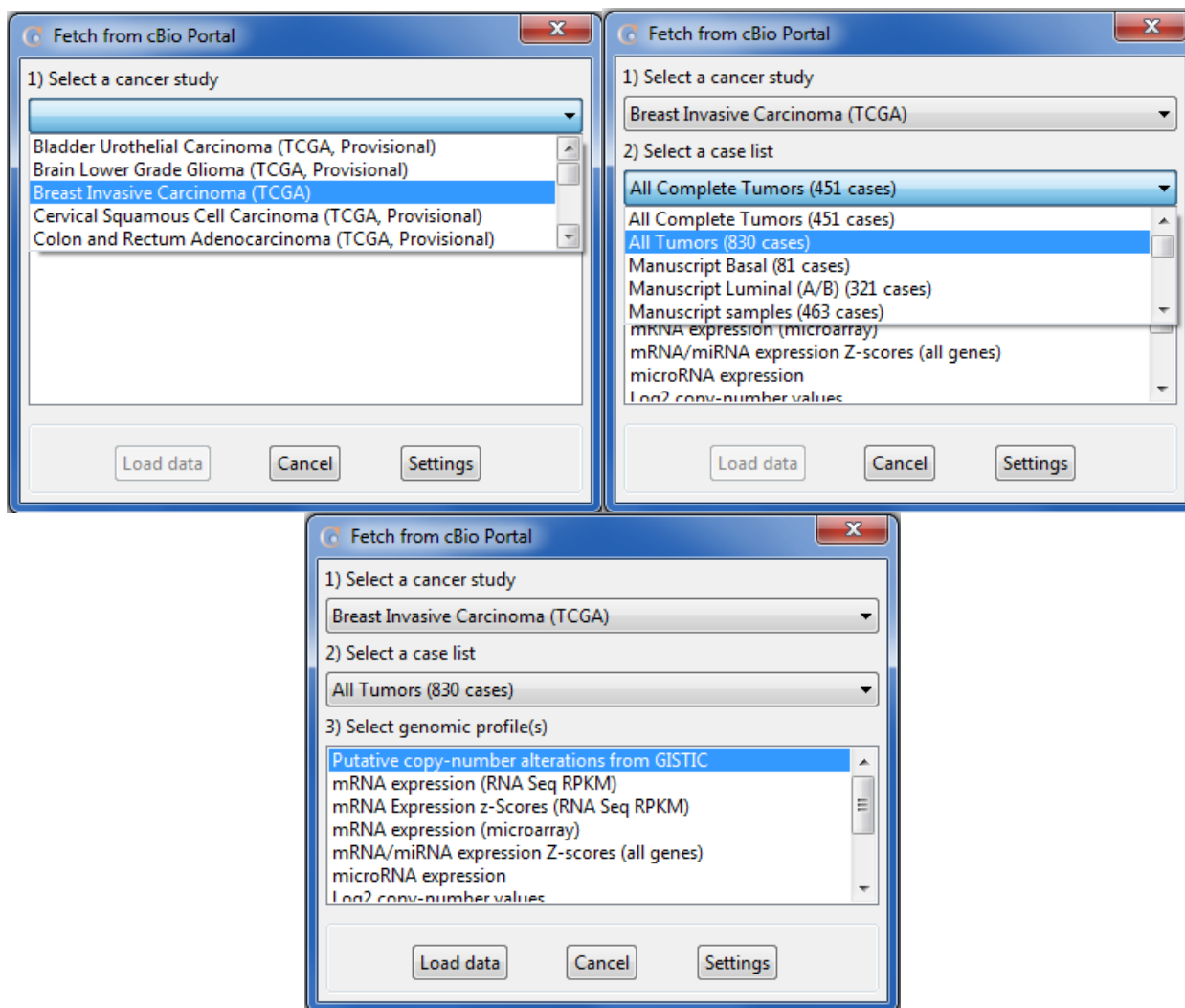


Figure 58 Sample Fetch from cBio Portal dialog

ChiBE uses some predefined thresholds to decide whether a gene is altered in a particular sample. For instance for expression profile, values greater than 2.0 or smaller than -2.0 are considered as altered. Users have the chance to change these threshold values to meet their specific needs. As seen in Figure 58, there is a “Settings” button in Fetch from cBio Portal dialog. Clicking this button opens the dialog seen in Figure 59. Four different categories are bound by thresholds, which are expression, copy number, protein and methylation. You can change these thresholds as you wish as long as the lower thresholds are smaller than upper thresholds. If you click “Save” button, these new values will become threshold values for the following operations.

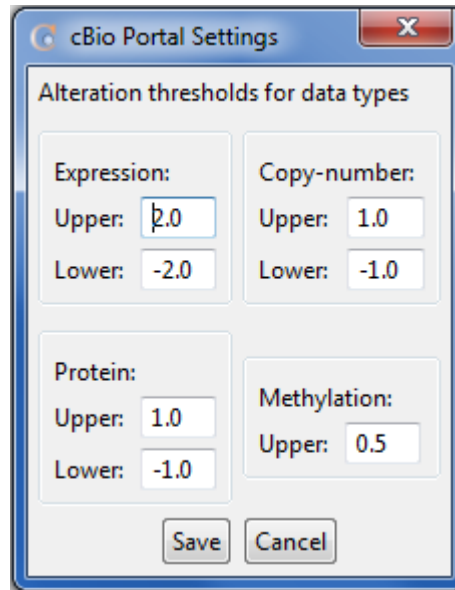




Figure 59 cBio Portal Settings dialog where you can adjust thresholds for different data types

You can follow two different ways to get a closer look into this genomic data. If you open the node pop-up menu, you will see “ Show cBio Portal Data Details” menu item. Clicking on this item will lead you to the dialog seen in Figure 60. The first three rows show cancer study, data profiles and case set which you have selected before loading the data. Total alteration frequency along with activating and inhibiting alteration frequencies are available. The frequency of cases where there are no changes or no data available are also shown in this dialog. The other way of using “Data |  Data Selection ...” menu item offers a more detailed analysis of the loaded data. You can use this menu item to find the specific samples in which a particular entity is altered. Details of using this menu are provided in the next section.

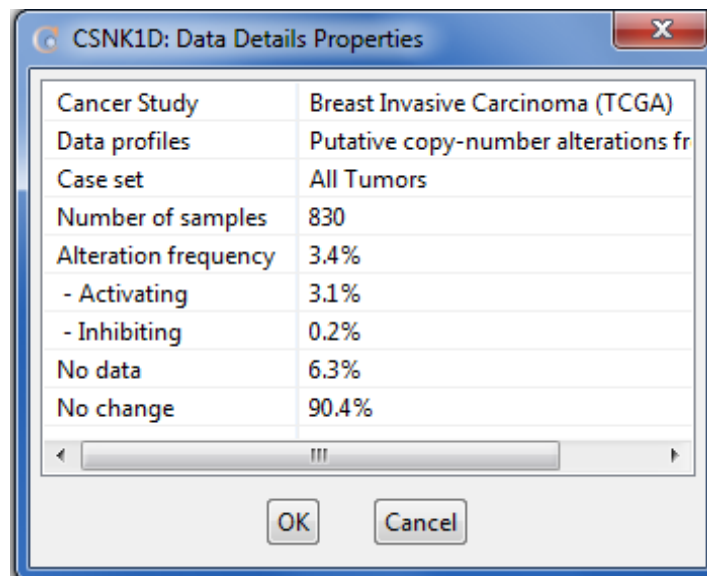


Figure 60 cBio Portal Data Details dialog

Visualizing Microarray Data

Figure 61 shows an example pathway view with microarray data mapped onto the pathway objects. The data is shown by color-coding. Up-regulation is represented with red; whereas, down-regulation is represented with blue. In-between values are displayed with in-between colors computed accordingly.

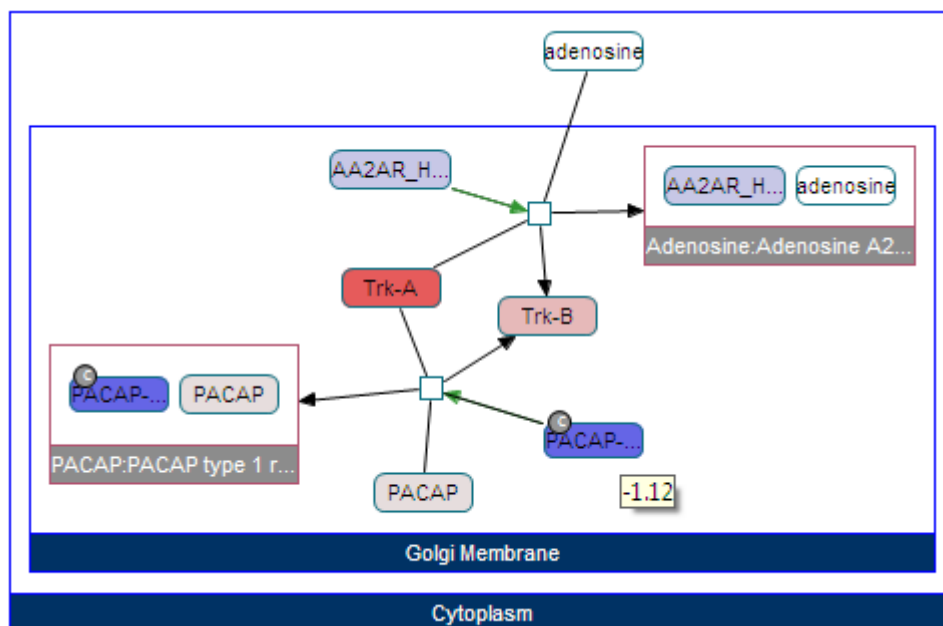



Figure 61 Example pathway view with microarray data mapped onto the pathway objects

Experiment data being currently visualized can be managed through the Experiment Data Management Dialog, available in the menu through “Data |  Data Selection ...” (Figure 62). In the top area, general information about the loaded experiments is displayed. The right panel shows information on the selected experiment, which are listed in the left panel. The user may choose the option to visualize average of a group of experiments, or alternatively choose to compare two groups of experiments. Upon clicking “Update”, the colors and/or values of pathway objects are updated.

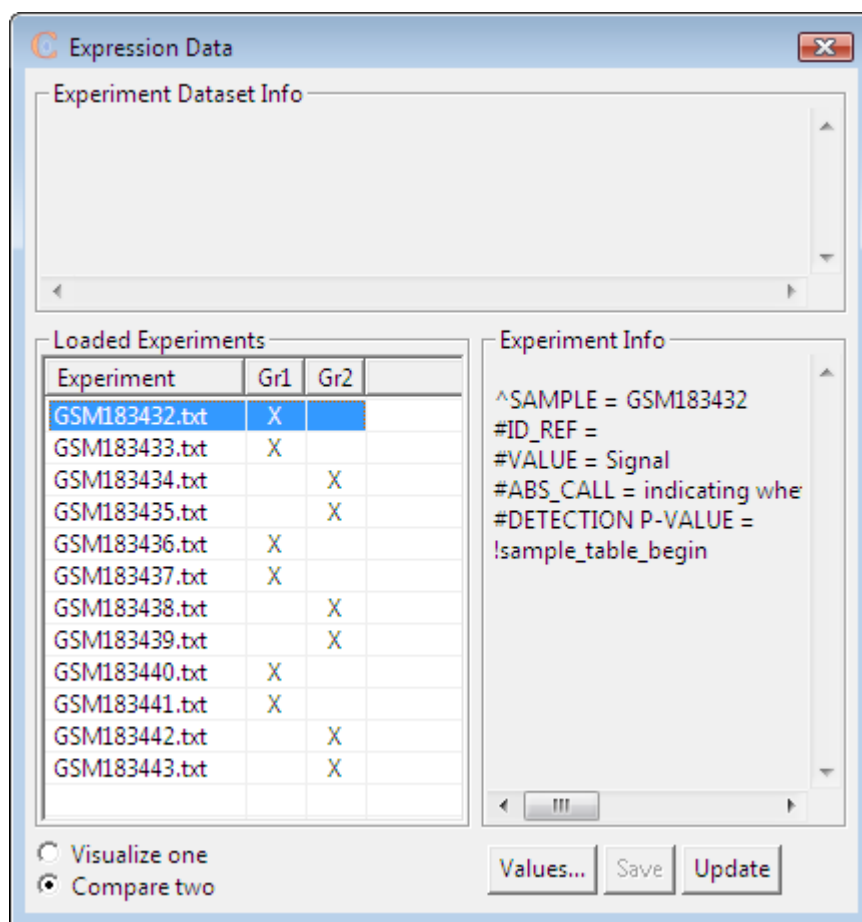


Figure 62 Experiment Data Management Dialog

Values in the loaded microarray data can also be viewed by clicking the “Values...” button in the management dialog. The “Values Dialog” opens and the values are listed in a table in pages. Page size is adjustable and viewing all rows in one page is also an option. First column contains the references of the rows. Values displayed in the table are colored according to the current grouping. If only one experiment is in use, then it is colored yellow. When average of one group is visualized, then the experiments in the group are colored yellow, and the average is shown in red, in the second column. When there are two groups compared, the second group is colored blue and the log-2 ratio of two averages is shown in the second column, in red color, as in Figure 63.

The screenshot shows a dialog box titled "Values" with a table of data. At the top, there are controls for "Filter rows" (with a "Filter" button), "Page size" (set to 20), and "Sort" (with a checkbox for "Show values sorted in page"). Below these is a "Values" section with navigation buttons ("prev", "next"), a "Retrieve" button, and a dropdown menu set to "Neighbors in Database". The table below has 15 columns: "References" and 14 "GSM..." columns. The row for "GENE_SYMBOL:PAX8 REF..." is highlighted in blue.

References	Log ...	GSM...	GSM...	GSM...	GSM...	GSM...	GSM...	GSM...	GSM...	GSM...	GSM...	GSM...	GSM...	GSM...
GENE_SYMBOL:DDR1 REF...	0.051...	1859...	1894...	2063...	2164...	1075...	1165...	1016...	1056...	2334...	2384...	2206...	2596...	
GENE_SYMBOL:RFC2 REF...	0.119...	9626...	8904.5	1149...	1037...	9502...	9820...	1026...	1104...	1427...	1259...	1303...	1410...	
GENE_SYMBOL:HSPA6 RE...	-0.23...	291.1...	219.8...	325.5...	283.2...	334.1...	620.4...	408.9...	367.1...	179.2...	249.2...	67.68...	157.3...	
GENE_SYMBOL:PAX8 REF...	0.066...	1367...	1529...	1615.2	1551.5	3998...	4331.0	3953...	3870...	1155...	1297...	1511...	1824...	
GENE_SYMBOL:GUCA1A ...	0.309...	72.73...	28.45...	34.28...	122.83	53.43...	162.8...	14.96...	148.3...	21.33...	52.75...	31.72...	133.2...	
GENE_SYMBOL:UBE1L RE...	0.245...	1259...	1423...	1732...	1566...	684.5...	689.3...	505.7...	974.7	448.4...	451.1...	455.7...	638.8...	
GENE_SYMBOL:THRA RE...	-0.04...	317.4...	320.6...	236.9...	178.4...	114.9...	351.8...	388.6...	301.6...	704.5...	825.2...	702.2...	754.4...	
GENE_SYMBOL:PTPN21 R...	-0.12...	293.8...	267.2...	174.5...	52.98...	47.18...	79.98...	100.5...	151.7...	994.7...	879.5...	943.1...	921.4...	
GENE_SYMBOL:CCL5 REF...	0.657...	244.0...	52.59...	605.7...	505.2...	96.55	41.77...	16.29...	19.84...	77.61...	273.3...	24.86...	67.72...	
GENE_SYMBOL:CYD2E1 R...	0.405...	264.0...	46.60...	234.1	130.0	307.3	312.5	434.2	305.1	126.5	186.5	131.1	314.5	

Figure 63 Experiment Values Table

Searching among the rows is facilitated by the filtering panel at the top of the dialog. When a filter word is entered and the “Filter” button is pressed, the rows which contain this word in their references are displayed.

The page can also be viewed sorted (in ascending order) according to the value column (log ratio or average) when “Show values sorted in page” option is selected. Note that this sorting is only within the current page and does not change its content. If the user wants to see all rows sorted, then they should select “All” as the page size, then sort again.

Rows of this table can be selected for retrieving 1-level (immediate) neighborhood of the graph objects associated with these molecules in the loaded pathway model or in the Pathway Commons database.

Causative Paths Based on the Loaded Experiment Data

When the loaded experiment data can be interpreted as an indicator for the activity of the states of the genes and proteins in the loaded model (e.g. expression data), one may ask to find the links between those activities. A positive path between two up-regulations or two down-regulations can be a hypothesis that explains the cause of the target regulation. Similarly a negative path between an up-regulation and a down-regulation is a possible cause for the target regulation. We call such paths *causative paths*.

Causative paths can be searched on the current pathway view by using “Data | Search Causative Paths” menu item. If any causative path is found, a list of the result paths is displayed and selected path is highlighted on the current view. “Data | Search Causative Paths On Entire Model” menu item can be used to search causative paths in the loaded model. In this case, a new view that is union of resulting causative paths is created together with the list of the paths and the selected path is highlighted on the view.

Highlight With Data Values

ChiBE offers users a way to analyze the graph based on loaded experiment data. After loading an experiment data, clicking “Data | Highlight With Data Values ...” from the menubar leads you to a dialog as seen in Figure 64. There are two fields in this dialog labeled as “Min” and “Max”. These will be used to specify the lower bound and upper bound of the range of interest. The initial value in “Min” field shows the smallest data

value found in the graph. Similarly, “Max” field initially shows the highest data value. User can use the sliders to achieve the desired values for these two parameters. Alternatively, writing the desired value into the text fields and then pressing “Enter” will update the sliders based on the given values. You can choose to specify only one boundary by selecting associated checkboxes and if you choose to determine both, minimum value must always be smaller than the maximum value.

User is also asked to determine whether nodes whose expression values fall within the specified range or outside the specified range will be highlighted. If “Within specified range” option is selected, then upon clicking “OK”, nodes whose data value greater than the lower bound and smaller than the upper bound will be highlighted. However, if “Outside specified range” option is selected, then values smaller than the lower bound or greater than the upper bound will qualify a node for highlighting. If only one of the lower or upper boundaries is specified, then it will always act as if “Within specified range” option is selected.

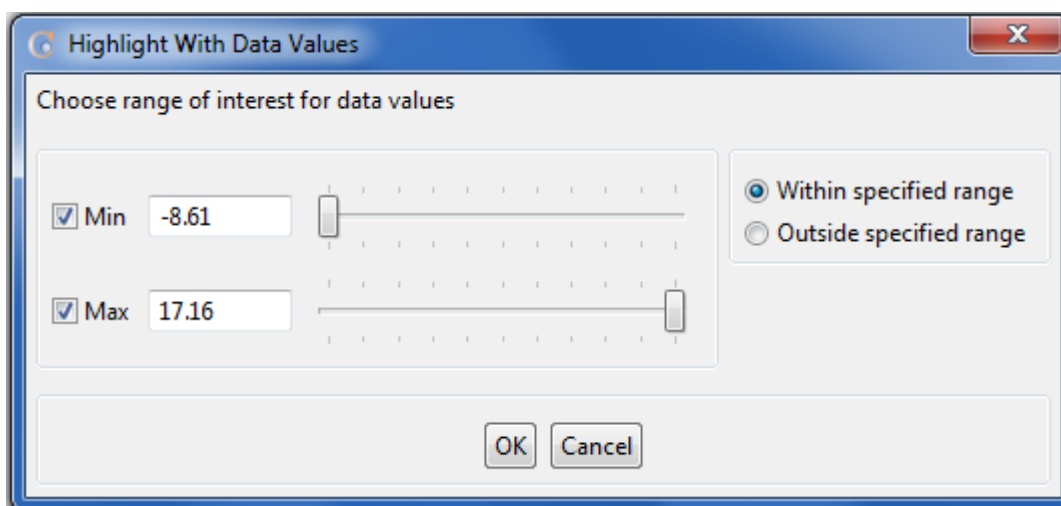


Figure 64 Sample Highlight With Data Values Dialog

Connect To DAVID

DAVID (The Database for Annotation, Visualization and Integrated Discovery) [19] is a powerful tool that can help the scientific community to analyze and annotate large biological datasets through use of known relationships, such as Gene Ontology terms. ChiBE offers an easy-to-use connection to this portal to ease annotation and understanding of gene or protein subsets present within pathway views.

To start an analysis in DAVID, you need to supply an ID list for genes or proteins of interest, in a format supported by DAVID. Extracting this information manually from the pathway data would be an inefficient way of moving from ChiBE to DAVID. Therefore, ChiBE automatically performs this step for you. The user is only required to select the nodes of interest on the pathway view (Figure 65). Then, the database IDs associated with these nodes will be extracted and a list will be generated that is compatible with DAVID requirements. This ID selection will be based on the most common ID type found within the selected set.

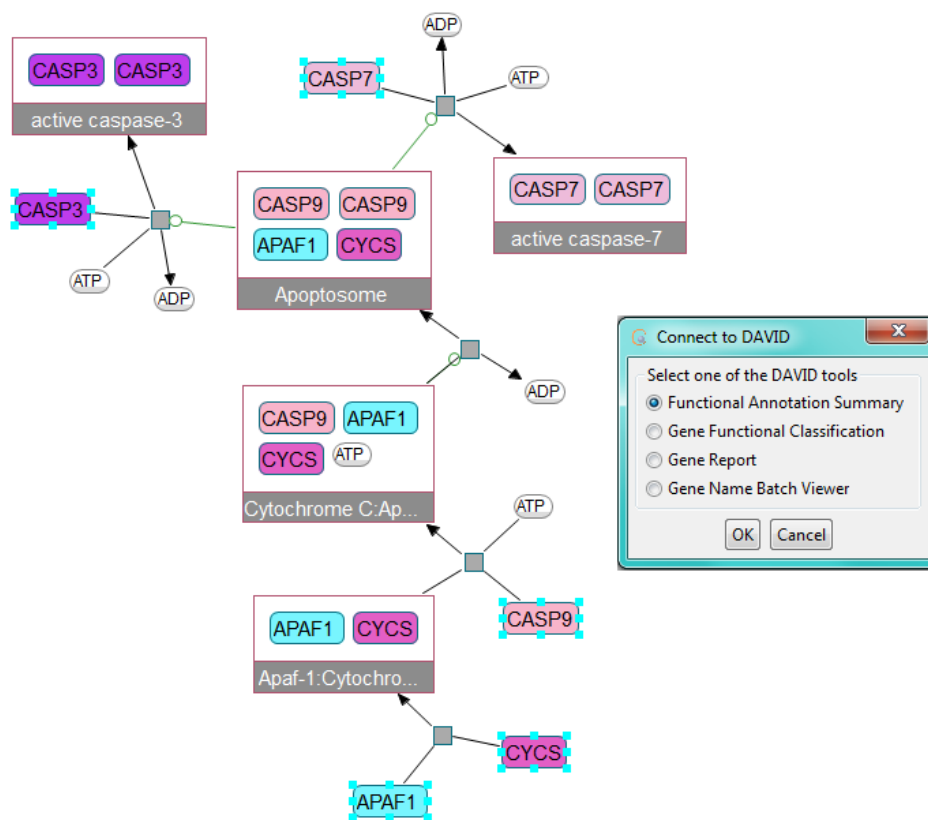


Figure 65 A sample pathway view with nodes selected for analysis in DAVID and associated dialog

The only other required user input is the type of analysis that will be performed. DAVID offers different analysis approaches and ChiBE gives users the freedom of selecting from these various options. Upon clicking “Data | Connect To DAVID ...”, the user will be directed to a dialog with four different options, as seen in Figure 65. “Functional Annotation Summary” will lead the user to “Functional Annotation” tool of DAVID, which performs annotation enrichment analysis. “Gene Functional Classification” will send the list to “Gene Functional Classification” option for clustering the list based on functional relationships and annotation terms. “Gene Report” can be used to obtain detailed information about each entry found in the set. Finally, “Gene Name Batch Viewer” will return the list of selected genes along with their IDs and genes related to them.

Figure 66 displays a sample result achieved through “Connect To DAVID”. In this example, “Gene Name Batch Viewer” analysis is performed with the nodes selected in Figure 65.


Gene List Report

[Help and Manual](#)

Current Gene List: List_1

Current Background: Homo sapiens

5 DAVID IDs

 [Download File](#)

UNIPROT_ACCESSION	Gene Name	Related Genes	Species
B2R4I1	cytochrome c, somatic	RG	Homo sapiens
B5BU45	caspase 7, apoptosis-related cysteine peptidase	RG	Homo sapiens
O14727	apoptotic peptidase activating factor 1	RG	Homo sapiens
Q96KP2	caspase 3, apoptosis-related cysteine peptidase	RG	Homo sapiens
Q9UIJ8	caspase 9, apoptosis-related cysteine peptidase	RG	Homo sapiens

Figure 66 Result of “Gene Name Batch Viewer” analysis with 4 nodes selected in the pathway view of Figure 65

Third-Party Software License Agreements

In this chapter, we list any third-party software license agreements.

ChiBE is distributed under Eclipse Public License, version 1.0 (<http://www.eclipse.org/org/documents/epl-v10.php>). Notwithstanding the terms and conditions of this license and any agreement you have with Information Visualization Research Group, the third-party software code, products, or files identified below are "Excluded Components" and are subject to the following terms and conditions.

Component	Version	By	License
Paxtools	1.0	MSKCC, INOH	LGPL
Java WSDP	1.5	Sun Microsystems, Inc.	Sun Microsystems Inc. Software License Agreement
GraphML	-	GraphML Community	Creative Commons Attribution License, v 3.0
Silk Icons	2.5	Mark James	Creative Commons Attribution License, v 3.0
Apache Jakarta Project	-	Apache Software Foundation	Apache License
JTA	1.0.1	Sun Microsystems, Inc.	License Agreement for Java(TM) Transaction API (JTA) Specification

References

- [1] <http://www.biopax.org>
- [2] Pathway Commons, <http://www.pathwaycommons.org>
- [3] Paxtools, <http://www.biopax.org/paxtools>
- [4] Chisio, Compound or Hierarchical Graph Visualization Tool, <http://www.cs.bilkent.edu.tr/~ivis/chisio.html>
- [5] Eclipse GEF (Graph Editing Framework) version 3.1, <http://www.eclipse.org/gef>
- [6] Silk icons version 2.5, <http://www.famfamfam.com/lab/icons/silk>
- [7] U. Dogrusoz, E. Giral, A. Cetintas, A. Civril, and E. Demir, "A Compound Graph Layout Algorithm for Biological Pathways", In Graph Drawing (Proc. GD '04), LNCS, vol. 3383, pp. 442-447, 2005.
- [8] U. Dogrusoz, E. Giral, A. Cetintas, A. Civril, and E. Demir, "A Layout Algorithm For Undirected Compound Graphs", Information Sciences, 179, pp. 980-994, 2009.
- [9] E. Demir, O. Babur, U. Dogrusoz, A. Gursoy, A. Ayaz, G. Gulesir, G. Nisanci, and R. Cetin-Atalay, "An Ontology for Collaborative Construction and Analysis of Cellular Pathways", Bioinformatics, 20(3), pp. 349-356, 2004.
- [10] PATIKA, <http://www.patika.org>
- [11] Cytoscape, <http://www.cytoscape.org>
- [12] NCBI GEO, <http://www.ncbi.nlm.nih.gov/geo>
- [13] Stanford MicroArray Database, <http://genome-www5.stanford.edu>
- [14] cBio Cancer Genomics Portal, <http://cbioportal.org>
- [15] SBGN (Systems Biology Graphical Notation), <http://www.sbgm.org>
- [16] T.M.J. Fruchterman, E.M. Reingold, "Graph drawing by force-directed placement", Software Practice and Experience 21(11), pp. 1129-1164, 1991.
- [17] C. Walshaw, "A Multilevel Algorithm for Force-directed Graph Drawing", (Proc. Graph Drawing) LNCS, 1984, pp. 171-182, 2001.
- [18] The HUGO Gene Nomenclature Committee (HGNC), <http://www.genenames.org/>

- [19] D. W. Huang, B. T. Sherman, and R. A. Lempicki, “Systematic and integrative analysis of large gene lists using DAVID bioinformatics resources”, *Nature Protocols*, vol. 4, pp. 44-57, Jan. 2009.