



BILKENT UNIVERSITY

i-Vis Information Visualization Research Group

Chisio User's Guide

BILKENT I-VIS RESEARCH GROUP

Chisio User's Guide



© i-Vis Research Group
Computer Engineering Department • Bilkent University
Ankara 06800, TURKEY
Phone +90 312.290.1401 • Fax +90 312.266.4047

Table of Contents

Overview	1
Use Cases	1
How to Download and Start Chisio	1
Graph and Drawing Model	2
Graph Model	2
Drawings	4
Nodes	4
Compound Nodes	5
Edges	5
Graphs	6
Editing in Chisio	10
Chisio Tools	10
Select Tools	10
Zoom Tools	11
Changing Topology	12
Creating Graph Objects	12
Deleting Graph Objects	13
Transferring Graph Objects	14
Reconnecting an Edge	12
Changing Geometry	13
Highlighting	14
Cluster IDs	15
Persistent Storage	17
Static Images and Printing	18
Layout	19
CoSE Layout	20
CoSE Layout Steps	21
Layout Options	21
Cluster Layout	22
Cluster Layout Steps	23
Layout Options	23

CiSE Layout	24
CiSE Layout Steps	25
Layout Options	25
Circular Layout	26
Circle Layout Steps	27
Layout Options	27
Spring Layout	27
Understanding the Spring Layout Algorithm	28
Layout Options	28
Hierarchical Layout	29
Hierarchical Layout Steps	29
Layout Options	30
References	39

Overview

In this chapter, we describe the tool Chisio, give its use cases and explain how to start the tool.

Chisio is a graph visualization tool for creating, editing and layout of *compound* or *hierarchically structured* graphs. The tool features user-friendly, interactive creation and manipulation of compound graphs. In addition, a number of popular graph layout algorithms, including ones designed by our group, have been implemented.

Use Cases

Chisio can be used for different purposes. If you would like to simply use Chisio as a generic graph editor, please continue reading this manual. However, if your goal is one of the following, then please refer to the Chisio Programmer's Guide instead:

- Customize the graph editor for a specific application (e.g. one that is used to draw UML class diagrams or a tool for visualization of social networks);
- Customize the tool for implementing a new layout algorithm (e.g. an algorithm that you developed and would like to test in an interactive tool).

How to Download and Start Chisio

Chisio 1.1 is supported on both 32-bit Windows and Linux operation systems. To download and set up Chisio 1.1, simply go to the tool Web site and follow the instructions provided:

<http://www.cs.bilkent.edu.tr/~ivis/chisio.html>

Graph and Drawing Model

In this chapter, we present the graph and drawing model used in Chisio.

The graph model used in Chisio is managed by a compound graph. A compound graph manages a list of nodes (possibly compound) and edges. Compound nodes, in turn, manages a list of nodes and edges, potentially creating an arbitrary level of nesting hierarchies.

Graph Model

A *Chisio graph*, or simply a *graph*, is composed of a *root graph*, which contains a set of *nodes* and *edges*, which are called *graph objects*. Nodes are joined to or connected to each other by edges. For example, Figure 1 shows a Chisio graph with three nodes (A, B, and C) and three edges (A to B, A to C, and B to C).

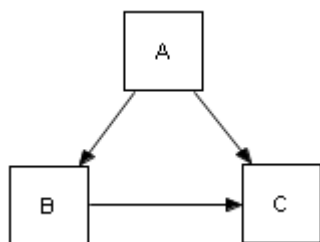


Figure 1 A Chisio graph with a root graph of 3 simple nodes (A, B, and C) and 3 edges (A-B, A-C, and B-C)

An edge has a *source* node and *target* node (Figure 2). If there are multiple edges between the same source and target nodes, then these edges are called *multi-edges* (Figure 3). If source and target nodes are the same then this edge is called a *reflexive* edge, simply a *loop* (Figure 4).

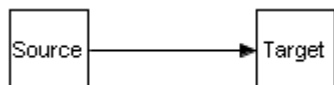


Figure 2 An edge with its source and target

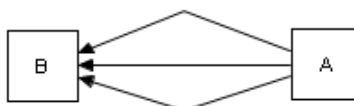


Figure 3 Example multi-edges

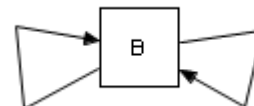


Figure 4 Example reflexive edges (loops)

A node maintains lists of its *incoming* and *outgoing* edges. Incoming edges are directed toward the node and outgoing edges are directed away from the node. In other words, node is the target node for its incoming edges and it is the source node for its outgoing edges (Figure 5).

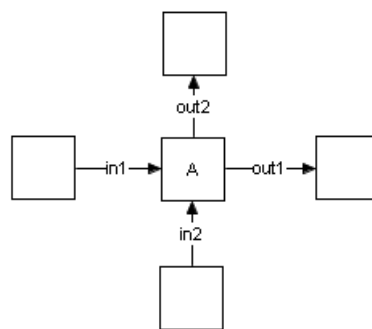


Figure 5 A node with incoming and outgoing edges

In a *directed graph*, each edge represents a one-way relationship from its source node to its target node. Directed edges use an arrow to indicate the direction of each edge. In an undirected graph, the directions of edges are ignored. Undirected edges are usually represented by lines without arrowheads (Figure 6).

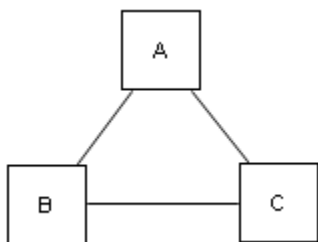


Figure 6 An undirected graph

Compound nodes are those that recursively contain a set of nodes and edges, composing its *child graph*, that are hierarchically below the compound node in the nesting tree. Edges can be categorized into two, depending on whether or not their end-nodes are in the same graph. If both source and target nodes of an edge are in the same graph, then the edge is said to be an *intra-graph edge*; it's called an *inter-graph edge*, otherwise.

For example, the graph in Figure 7 contains two compound nodes (G, H) and one simple node (J) in its root graph. Compound node G, in turn, has one compound (E) and two simple nodes (F, D) in its child graph. Finally, compound node E has one simple node (I). Compound node H has three nodes (A, B, C). Also, there are several inter- and intra-graph edges between these compounds and nodes as shown on the figure.

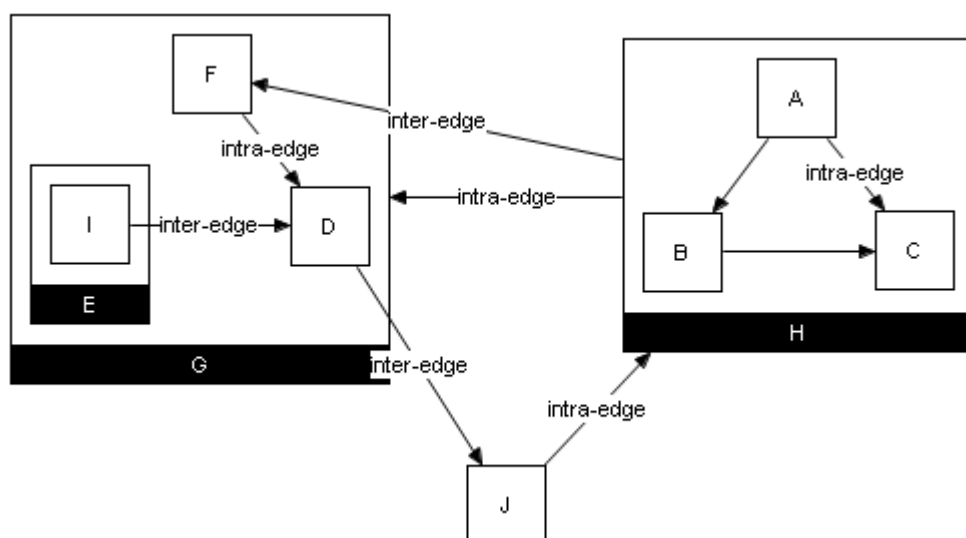


Figure 7 A compound graph with nesting of two levels, and intra and inter-graph edges

Drawings

Nodes, edges and compound nodes in Chisio have distinct properties and UIs. These properties can be changed by using properties windows or inspectors for each graph objects. When you change your object's properties, these changes are transferred to the model only after you press the "OK" button.

Nodes

Nodes are represented by a *rectangle* shape as default. But there are other pre-determined shapes that can be added to the graph by user interaction. These are *ellipse* and *triangle*. If you would like to add more types of shapes, you can easily add them by following the instructions in the Chisio Programmer's Guide.

Figure 8 shows a sample "Node Properties" window. By using this window, you can change your node's appearance. This window can be opened up by either double clicking the node or by using the node's popup menu.

Node Properties	
Name	Node
Text Font	Arial
Color	
Border Color	
Shape	Rectangle
Cluster ID	0
<div> Set As Default OK Cancel </div>	



Figure 8 A node and its properties window or inspector

You can also change the label (font type, size, and color), color, border color, shape and cluster ID of a node by using its "Properties" window.

"Set As Default" button is used for setting this node's properties as the default for the nodes that are to be created later on.

Compound Nodes

Similar to a simple node, a compound node's properties may be changed via the “Compound Properties” window (Figure 9). This window may also be opened up similar to the Node Properties window by double-clicking on it or through its pop-up menu.

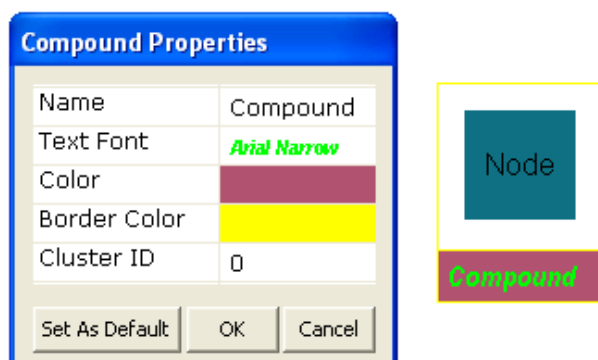


Figure 9 A compound node (right) and its properties window

You can change the label, color and border color of a compound node through this window. The size of a compound node is auto-calculated by the geometry of its contents, and the compound node is always just large enough to tightly bound its contents plus some user-defined margins.

“Set As Default” button is used for setting this compound’s properties as the default for the compounds that are to be created later on.

Edges

Both directed and undirected graphs can be visualized in Chisio. Edges are assumed to be connected to the center of their source and target nodes. An edge is drawn clipped, according to the specific shape and position of its source and target nodes (Figure 10).

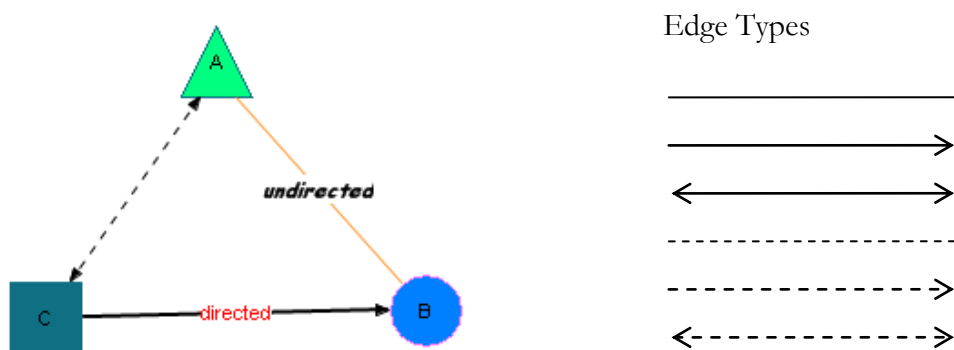


Figure 10 Examples of edges with different styles (left) and available edge types (right)

The appearance of edges can be changed by using the “Edge Properties” window. This window can be opened up by double clicking the edge or by the pop-up menu (Figure 11).

You can change the label, color, style (Solid, Dashed), arrow type (None, Source, Target, Both) and width (thickness) through this window.

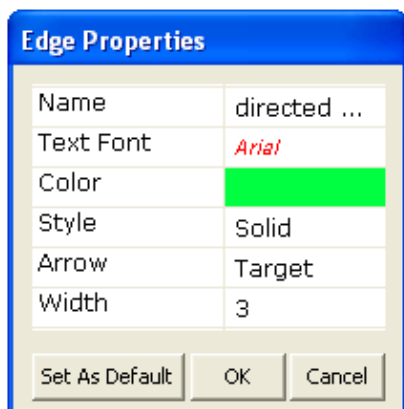


Figure 11 Edge properties window or edge inspector

“Set As Default” button is used for setting this edge’s properties as default for the edges that are to be created later on.

Graphs

There are certain general properties independent of specific graph objects. These can be adjusted through the “Graph Properties” window (Figure 12).

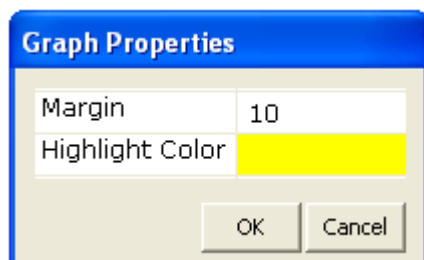


Figure 12 Graph properties window

Specifically, margins and highlight color (page 14) can be customized through this window. Graph margins are used to separate graph objects from their bounding boxes (such as nodes in a compound graph). Change of this value affects the margins around Chisio drawings when you fit them into the window (page 11).

Random graph creation

Chisio has a facility for randomly generating graphs of various classes using “File | Create Random Graph”. This pops up a dialog for configuring the random graph to be created. General parameters (number of nodes and edges, minimum and maximum node dimensions, whether disconnected nodes should be removed after creation) and type specific parameters allow the user to configure the random graph to be created.

Three different types of graphs can be created through this facility:

Flat graphs

Random flat (non-clustered, non-compound) graphs can be generated with Chisio (Figure 13). The user has the option to create *mesh-like graphs* (i.e. a graph that is a random subgraph of a mesh).

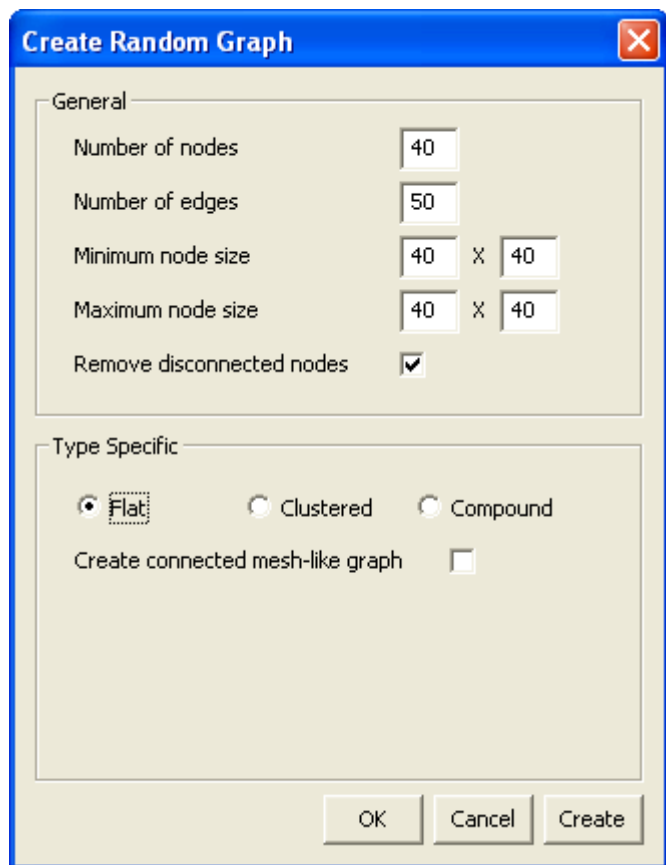


Figure 13 Options for a random flat graph

Clustered graphs

Random clustered graphs can be generated with Chisio (Figure 14). Through the options dialog, inter-cluster *edge ratio* (the ratio of number of inter-cluster edges to number of all edges) and *maximum cluster size* (maximum number of nodes in a cluster) may be set.

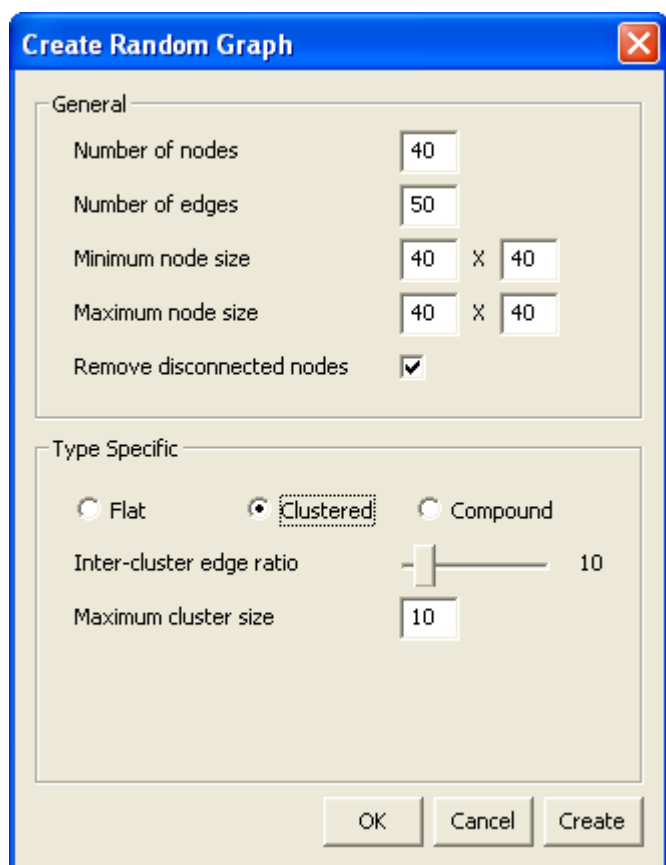
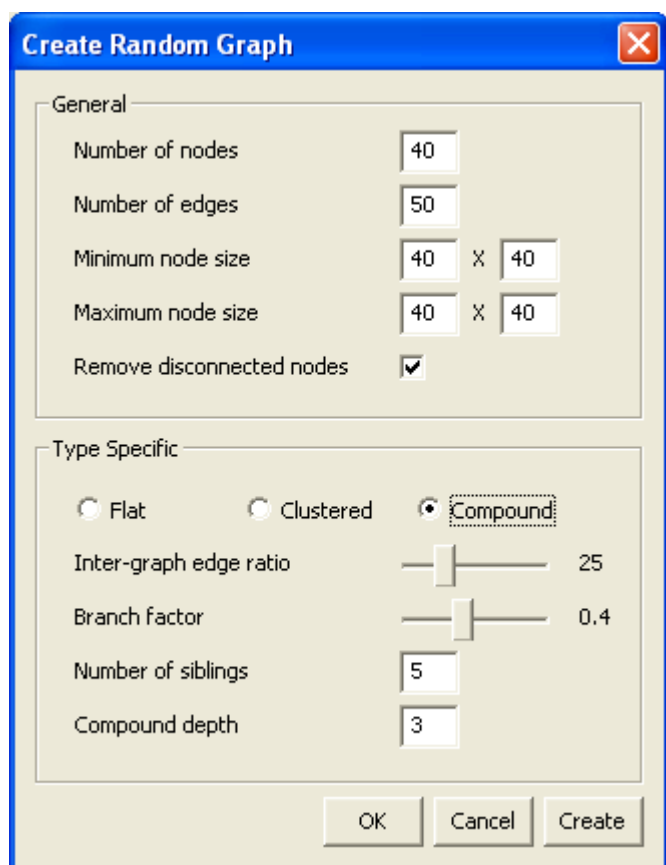


Figure 14 Options for a random clustered graph

Compound graphs

Random compound graphs can be generated with Chisio (Figure 15). Through the options dialog, *inter-graph edge ratio* (ratio of number of inter-graph edges to number of all edges), *branch factor* (probability of pruning a child in the nesting tree to avoid nesting trees that are too uniform in structure), *number of siblings* (maximum branching or number of children of a node in the nesting tree), and *compound depth* (maximum depth of a node or depth of the nesting tree) may be set.



The image shows a 'Create Random Graph' dialog box with a blue title bar and a close button. It contains two sections: 'General' and 'Type Specific'. The 'General' section has five controls: 'Number of nodes' (40), 'Number of edges' (50), 'Minimum node size' (40 x 40), 'Maximum node size' (40 x 40), and 'Remove disconnected nodes' (checked). The 'Type Specific' section has three radio buttons ('Flat', 'Clustered', 'Compound'), with 'Compound' selected. It also has two sliders ('Inter-graph edge ratio' at 25, 'Branch factor' at 0.4) and two text boxes ('Number of siblings' at 5, 'Compound depth' at 3). At the bottom are 'OK', 'Cancel', and 'Create' buttons.

General	
Number of nodes	40
Number of edges	50
Minimum node size	40 x 40
Maximum node size	40 x 40
Remove disconnected nodes	<input checked="" type="checkbox"/>

Type Specific	
<input type="radio"/> Flat <input type="radio"/> Clustered <input checked="" type="radio"/> Compound	
Inter-graph edge ratio	25
Branch factor	0.4
Number of siblings	5
Compound depth	3

OK Cancel Create

Figure 15 Options for a random compound graph

Editing in Chisio

In this chapter, we describe how graphs can be edited interactively in Chisio.

Graphs created with other tools and loaded into Chisio or graphs created in Chisio can be edited interactively as described below.

Chisio Tools

There are several tools to interact with the graphs; namely: *Select Tools*, *Zoom Tools* and *Create Tools*.

Select Tools

The “Select Tool” can be chosen from the top menubar using “Edit | Select Tool” or from the toolbar menu (🖱️). 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 can not be resized (Figure 16).

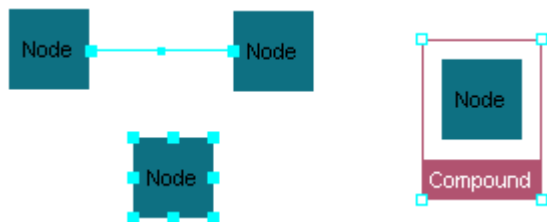


Figure 16 Example selection handles for all types of graph objects

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 17).

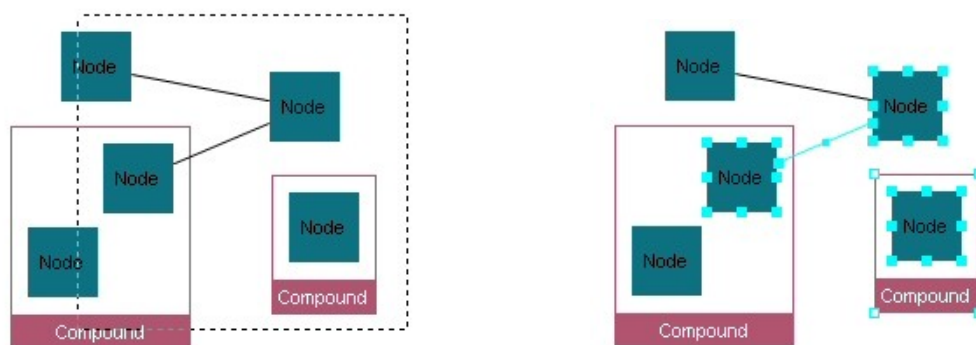



Figure 17 Example use of the Marquee Selection Tool

Zoom Tools

Chisio 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 top menubar using “Edit | Marquee Zoom Tool” or from the toolbar menu (). This tool is used to zoom into a specified rectangular area of the graph (Figure 18 and Figure 19).

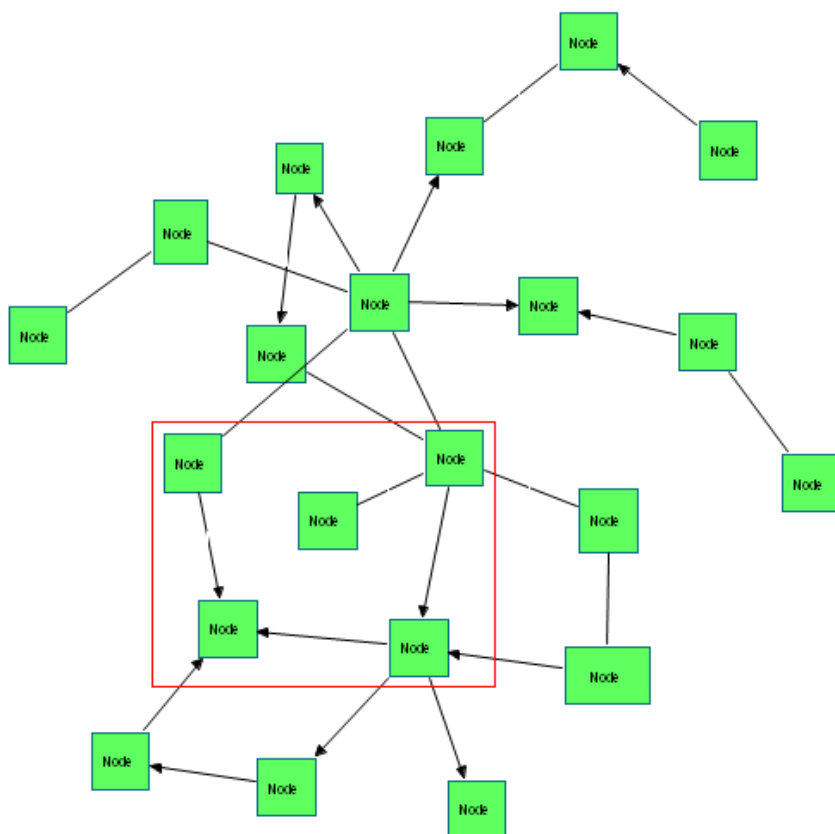


Figure 18 Before marquee zoom to the area specified by the red rectangle

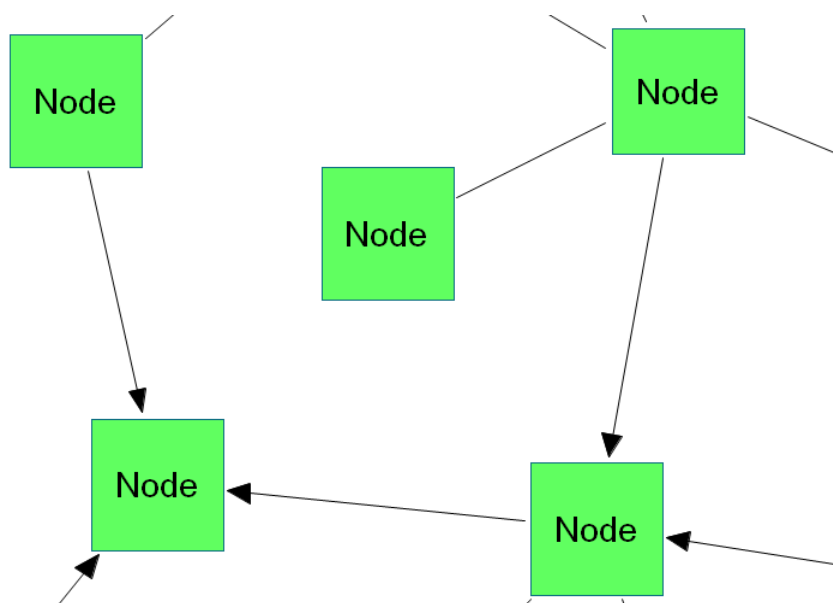


Figure 19 After marquee zooming into the area as described by Figure 18

Other zoom operations like zoom-in (🔍), zoom-out (🔍) and zoom to specified level are also supported. These operations can be found in the top menubar under “View | Zoom” and in the toolbar menu. In addition, graph popup menu provides zoom-in and zoom-out capabilities (Figure 20).

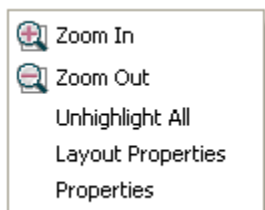


Figure 20 Graph pop-up menu includes zoom-in and zoom-out operations

Fit-in-window operation is another useful zoom operation, which shows the whole graph in the window by properly scaling it. You can fit your graph in window from the top menu bar using “View | Fit in window” or from the toolbar menu (🔍).

Changing Topology

You can interactively change the topological information of your graphs. You can create and delete graph objects easily. In addition, nodes and compounds can be transferred from one graph (root or child) to another (root or child). Finally edges can be reconnected by changing their source and/or target nodes.

Creating Graph Objects

Creating a new node or a new compound can be done from the top menubar under the “Edit” menu. Also using the toolbar menu, you can choose 🏠 (create a node) or 🏠 (create a compound node) items. When you





select creation tools, the cursor is changed to  (create a node) or  (create a compound node), respectively. When you click any place on the drawing canvas, the creation is performed (Figure 21).



Figure 21 Create a node by simply clicking on the drawing canvas, where you would like your new node to be placed; before the creation (left) and after the creation (right).

Creating a new edge can be done from the top menubar under the Edit menu. Also using the toolbar menu, you can choose  item. When you select the edge creation tool, the cursor is changed to . You must click first on the source node for this new edge. Your second click must be on to the target node for this edge to complete the creation of the new edge (Figure 22).

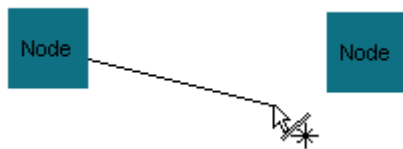




Figure 22 Create an edge by first clicking on the source node and then on the target node

Deleting Graph Objects

To delete nodes, compound nodes or edges, you must select them first. Then you can delete them in one of the following ways:

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

Multiple graph objects may be deleted at once using selection of these objects together.

One may also remove existing compound nodes while keeping their child nodes and edges at their current locations. This can be done from the top menubar using “Edit | Remove Compound” or by using the toolbar menu  after selecting the compound node(s) to be removed (Figure 23).

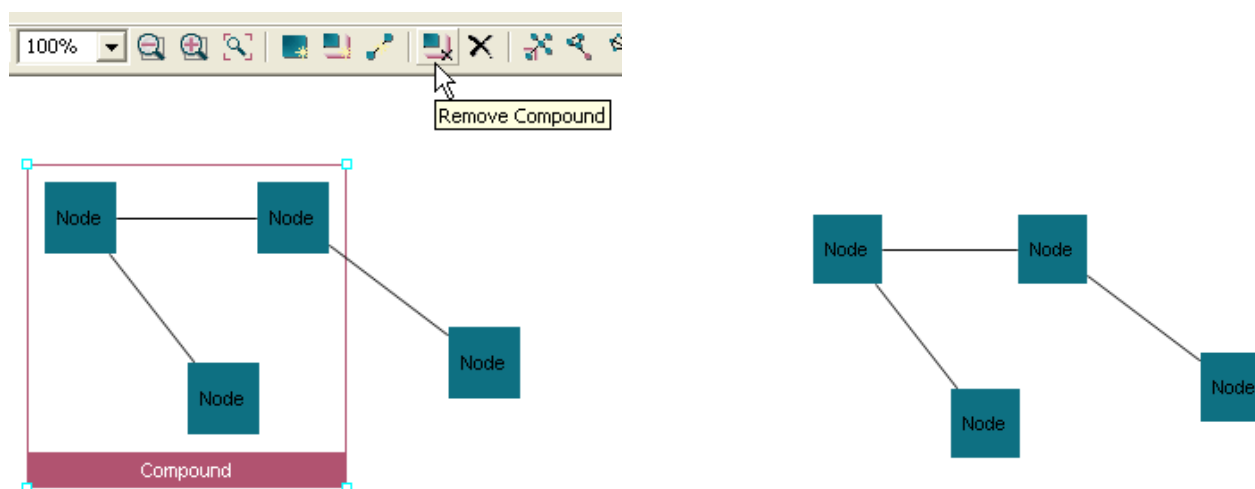


Figure 23 Remove compound; before and after

Transferring Graph Objects

Another way to change the topology is to move a node from one compound node (or root graph) to another compound node (or root graph). In other words, you can change the parent of a node.

This can only be done when the *mode* of the selection tool is “Transfer Mode”. You can change the mode to “Transfer Mode” from the toolbar menu (Figure 24) or from the top menubar using “Edit | Transfer Mode”.

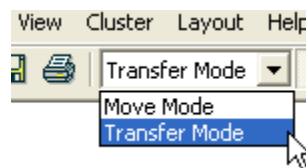


Figure 24 Move/Transfer Mode combo

When in the transfer mode, drag operations might mean change of ownership or transfer. So you can transfer a node/compound node A into another compound node B by dragging and dropping A onto (within the bounds of) the compound node B. When you select a node and move it on to a compound node, compound nodes change background color to cyan to indicate that you will be transferring the selected node into the highlighted compound node should you release it at this location (Figure 25).



Figure 25 When transferring a node into a compound node, the compound node's background color changes to indicate the transfer; initially (left), when the transfer of the selected node is about to be performed.

Also when you are in the transfer mode, you can clone the selected graph objects by pressing the CTRL key while dragging them. When you press the CTRL key, the cursor changes to indicate the cloning operation (Figure 26). In cloning is not possible in the “Move Mode”.

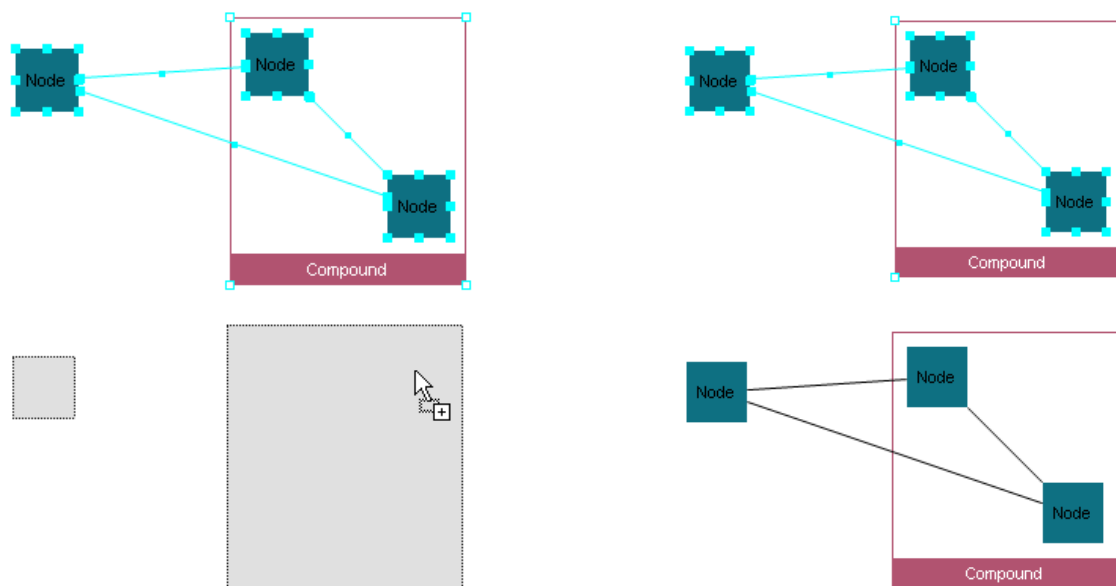


Figure 26 Clone operation; drag selected nodes pressing the CTRL key in the transfer mode and click to clone.

Another transfer operation is via the “Create Compound from Selected” operation in the node pop-up menu. You can create a compound node from a set of selected nodes and compounds easily. This operation is also available through the top menubar using “Edit | Create Compound from Selected” (Figure 27).

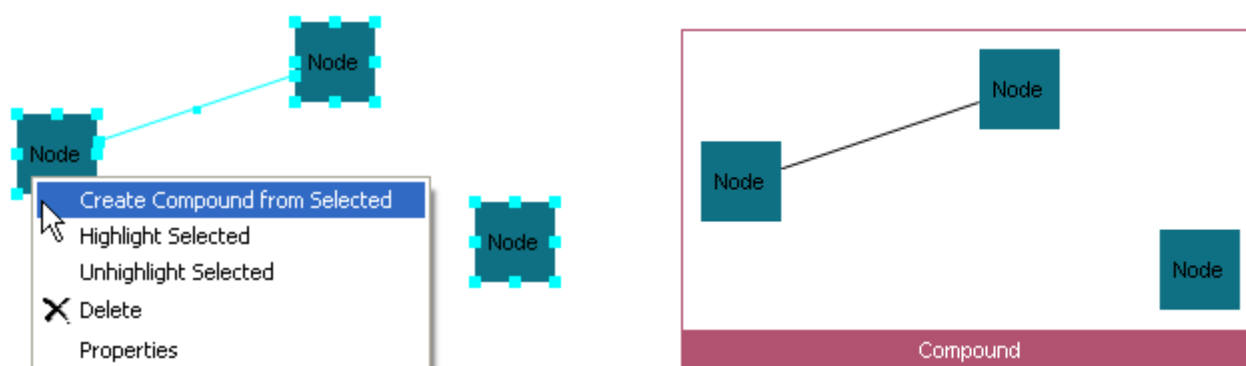


Figure 27 Create compound (right) from selected nodes (left)

Reconnecting an Edge

You can change the source or target of an edge after creation of it. When an edge 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 28).

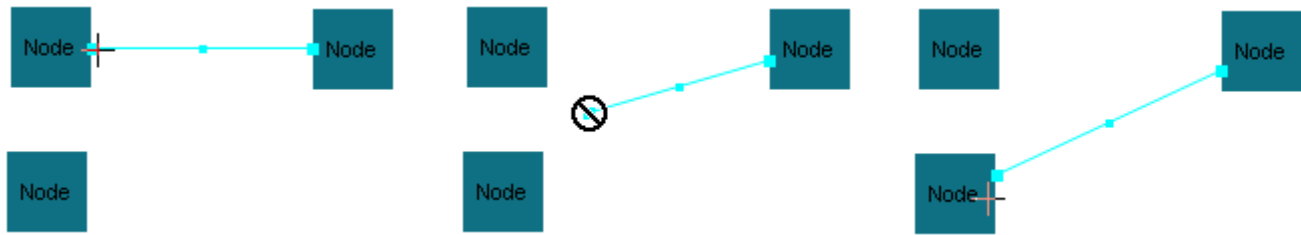


Figure 28 Edge reconnection; after selection of the edge (left), during dragging (middle) and after reconnection (right).

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 29). We assume the “Move Mode” here; during “Transfer Mode” this might mean a transfer operation as explained earlier in Section on Transferring Graph Objects on page 14.



Figure 29 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 30).

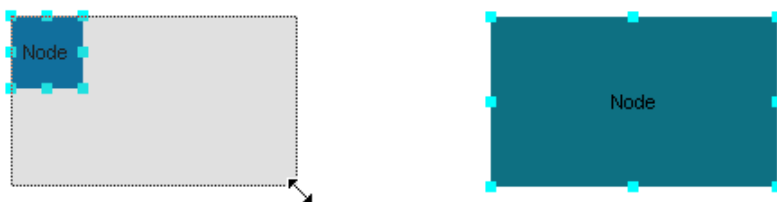


Figure 30 Resize a node; before (left) and after (right)

You can not 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 31).

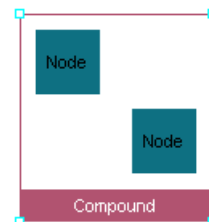


Figure 31 Compound nodes are automatically resized.

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

Chisio 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 32). 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 32 Bend point creation; after selection (left), during drag of the small handle (middle), and upon completion with the release of the mouse (right)

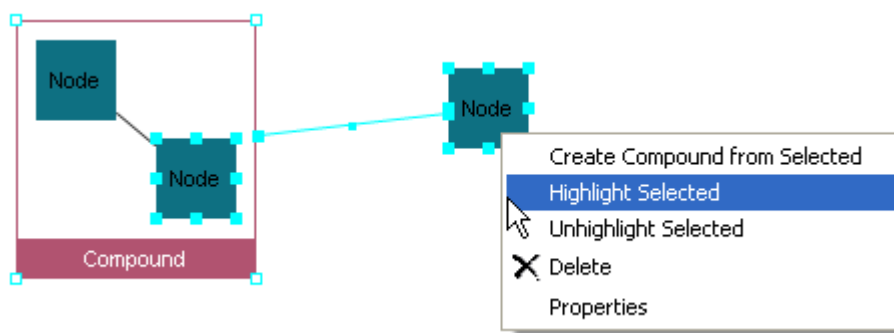
You can relocate existing bend points by dragging them to new locations as you like. If the final location of a bend point upon release of the mouse is to be aligned with its current neighbors of the bend point to make a straight line (or very near such a position), then the bend point is automatically deleted (Figure 33).



Figure 33 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).

Highlighting

Graph 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 “View | Highlight Selected” in the top menubar. Multiple objects may be highlighted together (Figure 34).



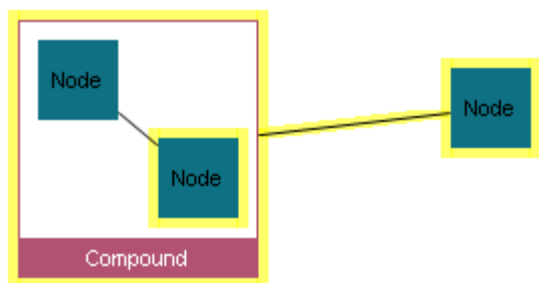


Figure 34 Highlighting objects

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

You can remove highlights of all objects by selecting “Unhighlight All” item in the top 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 “View | Unhighlight Selected” in the top menubar or “Unhighlight Selected” in the node pop-up window.

Cluster IDs

You can group or cluster a set of nodes in a Chisio graph. To do that, simply select the nodes you would like to assign a new cluster to, and then select “Cluster | Assign Selected to New Cluster” from the top menubar (Figure 35). This operation gives a new, unused cluster ID to selected nodes.

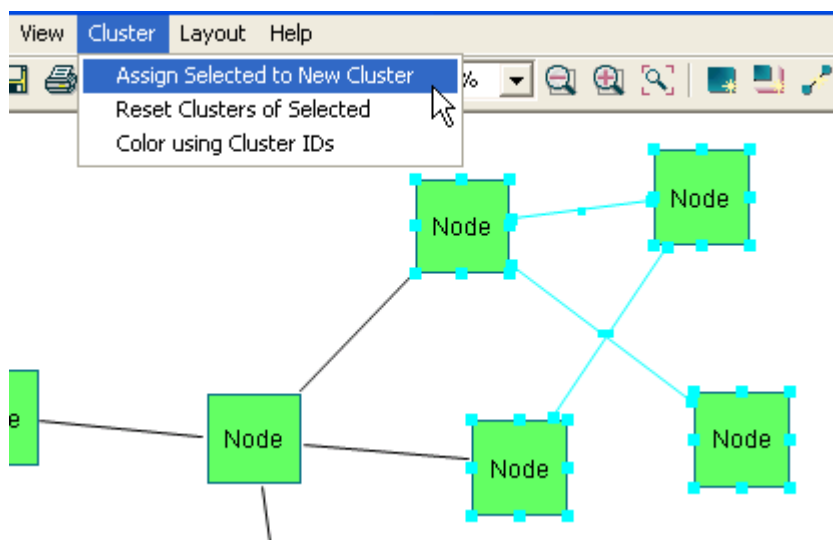


Figure 35 Creating a new cluster

You can find out about the new cluster IDs of these nodes through the node inspector (Figure 36). You may change the cluster ID of a node as needed from this window as well.

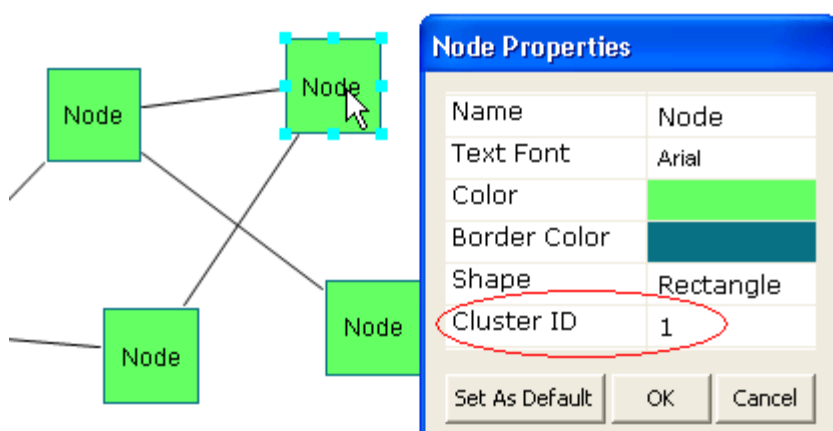


Figure 36 New cluster ID of the node as confirmed by the node properties window

You can also color-code the nodes in your Chisio graph according to their cluster IDs. Nodes in the same cluster are colored with the same, unique color. In addition, intra-cluster edges are colored to be gray (a little lighter than the default edge color, black, used for inter-cluster edges), so they can be better differentiated from inter-cluster edges. Simply select “Cluster | Color using ClusterIDs” from the top menubar (Figure 37).

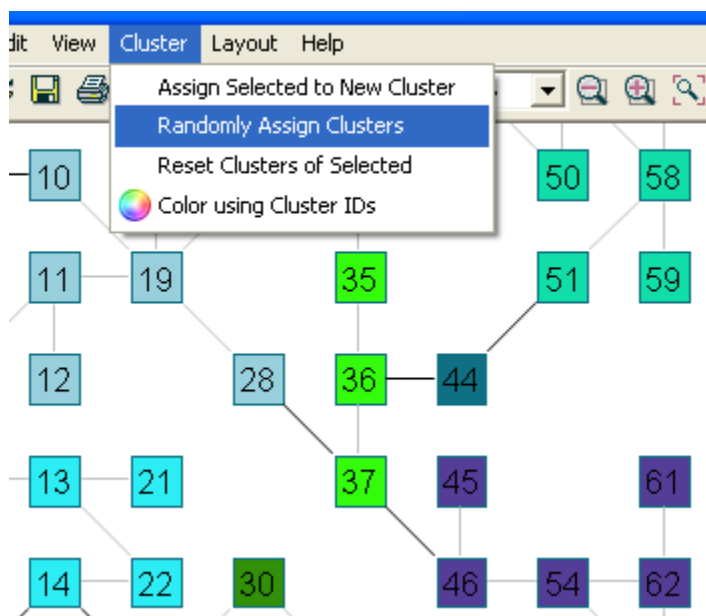


Figure 37 Coloring with cluster IDs results in three clusters, each with a unique color

You can reset the cluster information of selected nodes. If ClusterID of a node is “0”, this means that node does not belong to any cluster. Select the nodes you want to reset their cluster information, and then select “Cluster | Reset Clusters of Selected”.

Random cluster assignment

You may also assign random clusters to the current graph using “Cluster | Randomly Assign Clusters”. This operation re-colors the graph using cluster IDs that are randomly assigned.

Persistent Storage

You can save your graphs into a file for later use. Graphs are saved in GraphML file format [1], which is an XML-based graph format. In addition, your own files in GraphML format, possibly created by other programs may be loaded up into Chisio. Drag and drop is supported for convenient file loading as well.

Every property of a graph object is written into the file in a regular format. Below you can find parts of a GraphML file for specific examples.

Example 1: A highlighted node (Figure 38)

```
...
<node id="n0">
  <data key="x">108</data>
  <data key="y">90</data>
  <data key="height">40</data>
  <data key="width">40</data>
  <data key="color">14 112 130</data>
  <data key="borderColor">14 112 130</data>
  <data key="text">Node</data>
  <data key="textFont">1|Arial|8|0|WINDOWS|1|-11|0|0|0|0|0|0|0|1|0|0|0|0|Arial</data>
  <data key="textColor">0 0 0</data>
  <data key="clusterID">0</data>
  <data key="highlightColor">255 255 0</data>
  <data key="shape">Rectangle</data>
</node>
...
```

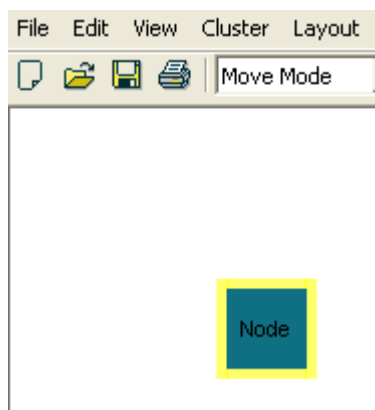


Figure 38 A Highlighted node for example GraphML format

Example 2: A dashed edge (Figure 39)

```
...
<edge id="e0" source="n0" target="n1">
  <data key="color">255 128 192</data>
  <data key="text">edge</data>
  <data key="textFont">1|Comic Sans MS|12|2|WINDOWS|1|-16|0|0|0|400|1|0|0|0|3|2|1|66|Comic
  Sans MS</data>
</edge>
...
```



```

<data key="textColor">0 0 255</data>
<data key="style">Dashed</data>
<data key="arrow">Target</data>
<data key="width">2</data>
</edge>
...

```



Figure 39 An edge for example GraphML format

Static Images and Printing

You may save the current drawings of your graphs as static images as well. Supported image formats are BMP and JPEG. You have the option of recoding the entire drawing or only the currently viewable part of the drawing (Figure 40).

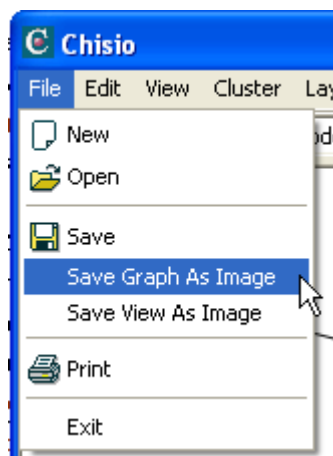


Figure 40 Save as image menu item

In addition, you can print your graphs from the top menubar “File | Print” or the toolbar menu (🖨️). Print operation prints the whole graph in regardless of the currently viewable part of the drawing.

Layout

In this chapter, we present automatic layout operations available in Chisio.

Automatic layout of graphs is extremely important with most graphs as it becomes a cumbersome operation, if not impossible, to manually layout the nodes of a graph and route the edges to produce aesthetically pleasing drawings. Chisio provides a number of popular and useful layout styles as described here.

Each layout style can be customized through the “Layout Properties” window, accessible through “Layout | Properties” in the topmenu bar or under the graph pop-up menu. Each style gets a tab of its own for its available options.

Layout operation is run on a separate thread in Chisio. This facilitates a smooth animation as well as allowing the layout operation to be cancelled using “Layout | Stop Layout” menu item or from the toolbar menu (🛑).

General Layout Options

Options common to all layout styles are gathered on a separate tab in Layout Properties dialog (Figure 41).

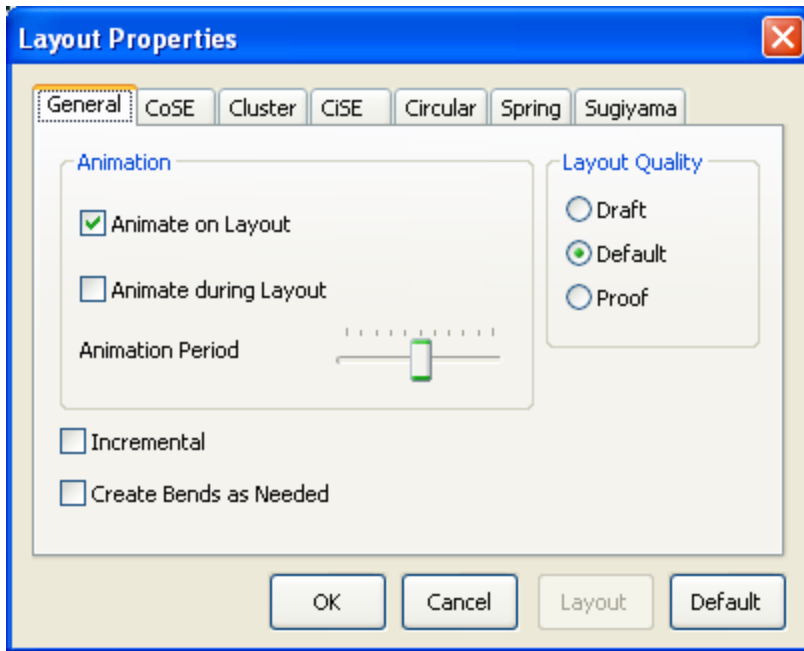


Figure 41 Common layout options

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).

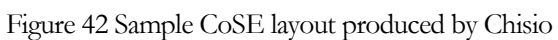
Create Bends as Needed: Should the algorithm create bend points for edges as needed (e.g. for multi-edges).

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

CoSE Layout

CoSE (**C**ompound graph **S**pring **E**mitter) layout is an algorithm specifically designed for compound graphs [2]. It has been designed by our group, based on the traditional force-directed layout scheme with extensions to handle multi-level nesting, varying node sizes, and possibly other application-specific constraints.

An expanded node and its associated nested graph are represented as a single entity, similar to a “cart”, which can move freely in orthogonal directions (no rotations allowed). Multiple levels of nesting is modeled with smaller carts on top of larger ones.



- Initialize data structures, perform random positioning for static layout and reduce trees for efficiency;
- Lay out the remaining, “skeleton graph” using a modified spring embedder as described earlier;
- Grow trees and continue iterating over the spring embedder model;
- Polish layout during this stabilization phase using a cooling schema for convergence.

CoSE layout options are shown in Figure 43.

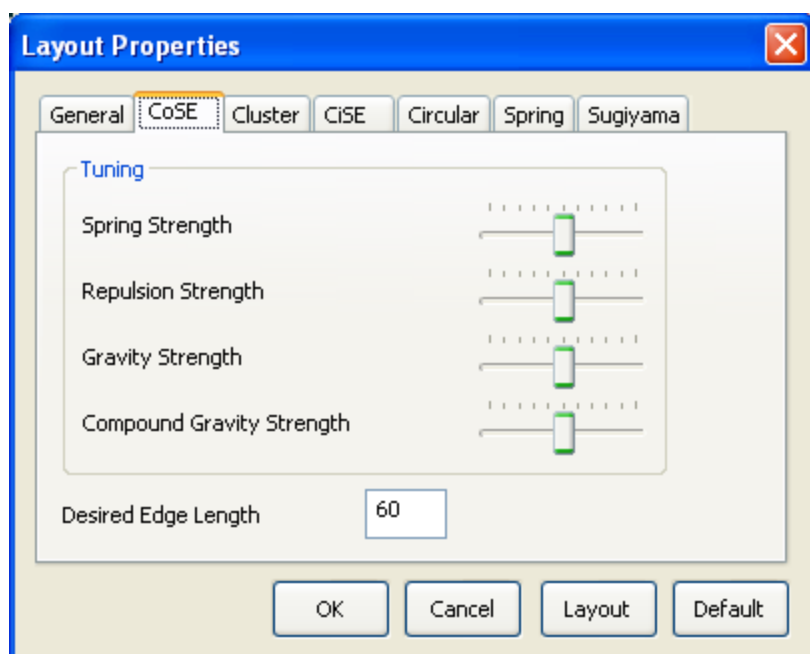


Figure 43 CoSE layout options

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.

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.

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.

Cluster Layout

Many applications group or cluster the nodes in graphs to refer to say molecules with similar functionality or network devices in a LAN. Naturally users of such graphs would like the nodes in the same cluster to be placed near each other.

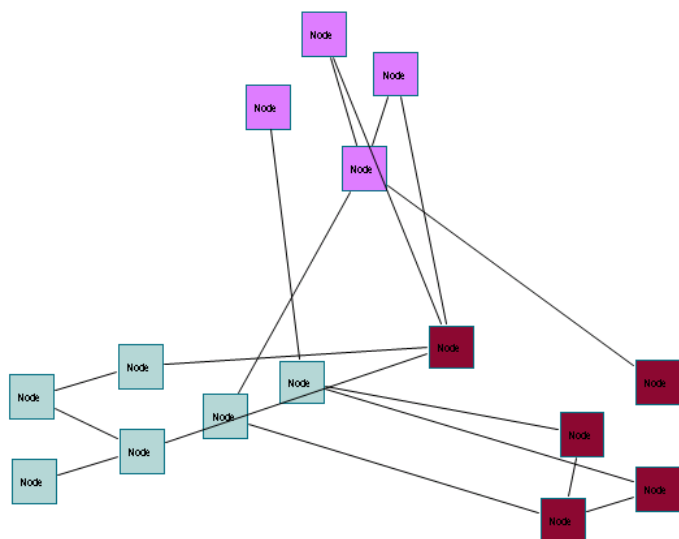


Figure 44 Sample Cluster layout produced by Chisio, where nodes are color-coded by their clusters

Cluster layout uses the CoSE layout algorithm to establish this as follows:

Cluster Layout Steps

- Create a dummy compound node for each cluster;
- Move clustered nodes into these dummy compounds according to their cluster information (cluster IDs);
- Run CoSE layout for this graph;
- Remove dummy compounds and leave nodes in these compounds in their current absolute locations.

Layout Options

Cluster layout options are shown in Figure 45.

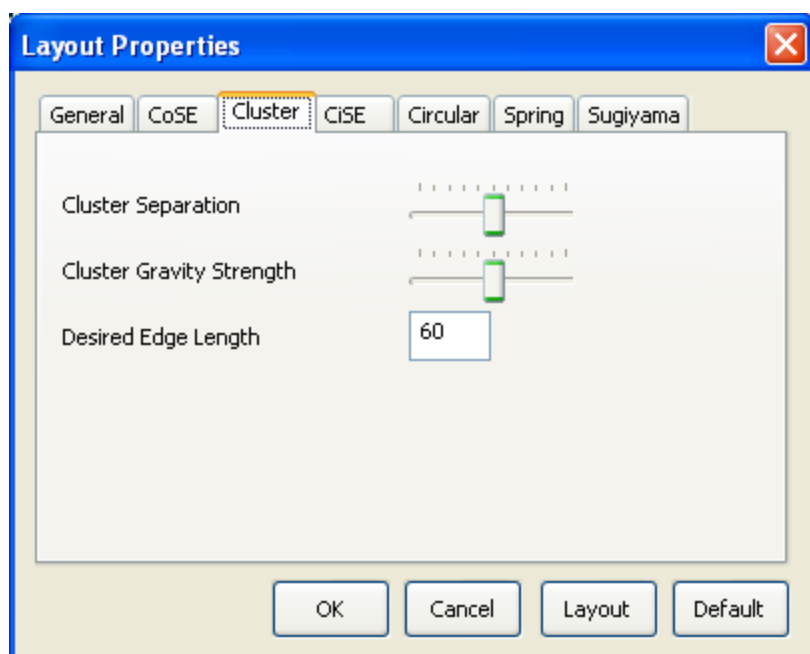


Figure 45 Cluster layout options

Cluster Separation: Distance between neighboring clusters.

Cluster Gravity Strength: The gravity inside clusters; the higher this value is, the closer the nodes in each cluster will be.

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/cluster bounds.

CiSE Layout

A popular way to draw clustered graphs is in a circular fashion. In other words, a circle of appropriate size is created for each cluster and the nodes in that cluster are placed around this circle trying to minimize edge crossings. Circular layout algorithms address the issue of placing nodes of a cluster nicely around a circle, trying to minimize the number of crossing and edge lengths; however most of these algorithms do not address the issue of how the cluster graph should be laid out (i.e. how the individual circles should be placed with respect to each other to minimize crossing and edge lengths of inter-cluster edges. CiSE (**C**ircular **S**pring **E**MBEDDER) layout algorithm has been developed by our group and uses a spring embedder based approach to do this.

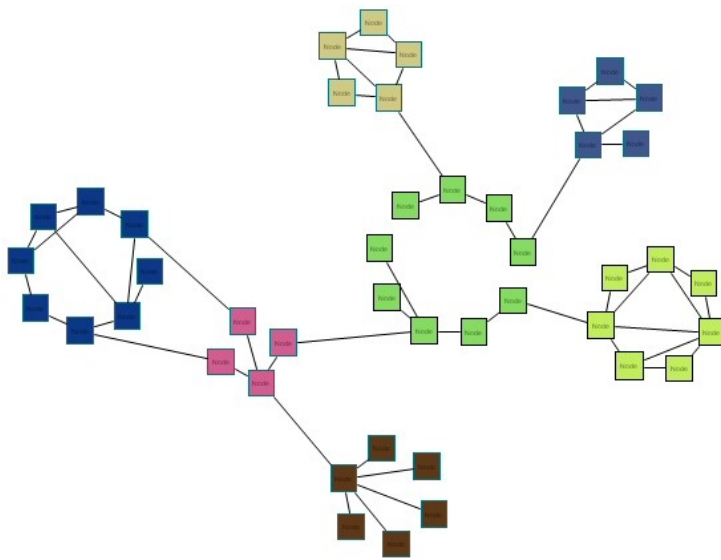


Figure 46 Sample CiSE layout produced by Chisio, where nodes are color-coded by their clusters

CiSE Layout Steps

- Lay out individual clusters using Circular (AVSDF) layout as described on page 26;
- Do an initial placement of the cluster graph using a basic spring-embedder;
- Perform a modified spring embedder, where clusters/circles are allowed to move and rotate;
- Perform a modified spring embedder, where nodes in each cluster are also allowed to move (swap).

Layout Options

CiSE layout options are shown in Figure 47.

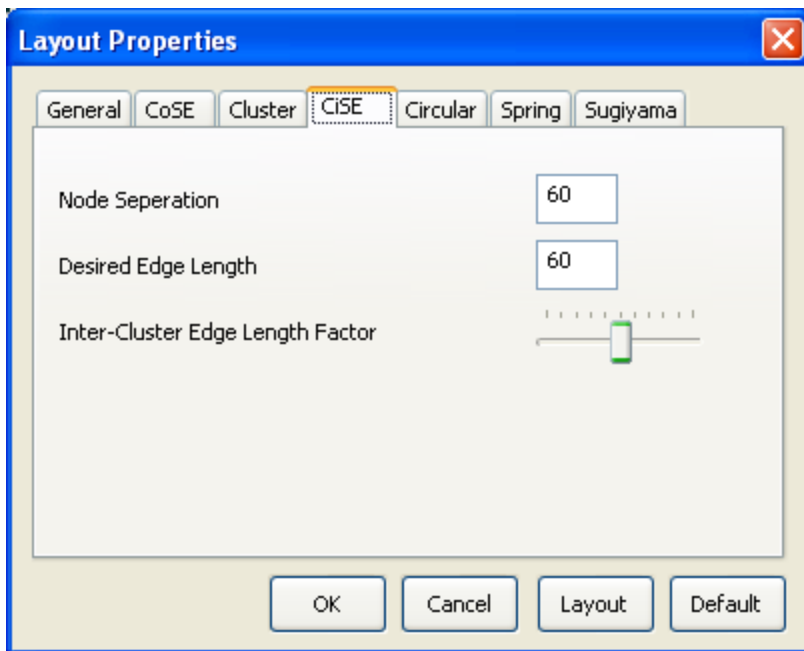


Figure 47 CiSE layout options

Node Separation: Distance between neighboring nodes in a cluster.

Desired Edge Length: Desired length of an intra-graph edge.

Inter-Cluster Edge Length Factor: The proportion of inter-cluster edge length to regular edges. The higher this value is over 1.0, the longer inter-cluster edges will be with respect to intra-cluster edges.

Circular Layout

This layout algorithm can be used as the base of such a clustered graph layout algorithm. It is based on that of He and Sykora [3] and places all nodes of the current graph around a single circle, taking into account non-uniform node dimensions and desired node separation parameter.

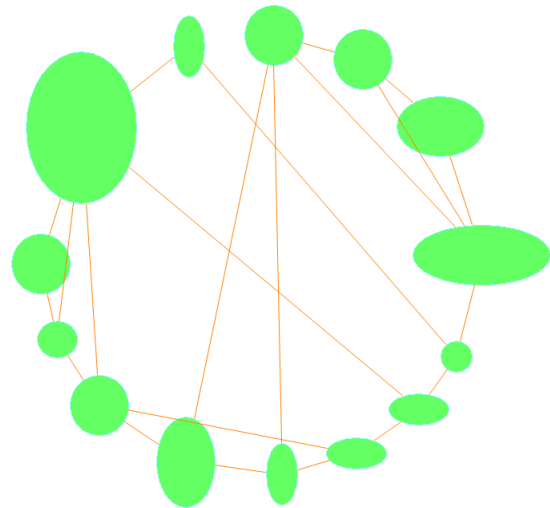


Figure 48 Sample Circular layout

Circle Layout Steps

- Radius of the circle is calculated by using the total number of nodes, their dimensions and desired spacing between neighboring nodes;
- Each node is located onto the circle by AVSDF (adjacent vertex with smallest degree first) approach;
- Numbers of edge crossings are reduced with a post processing step.

Layout Options

Circular layout options are shown in Figure 49.

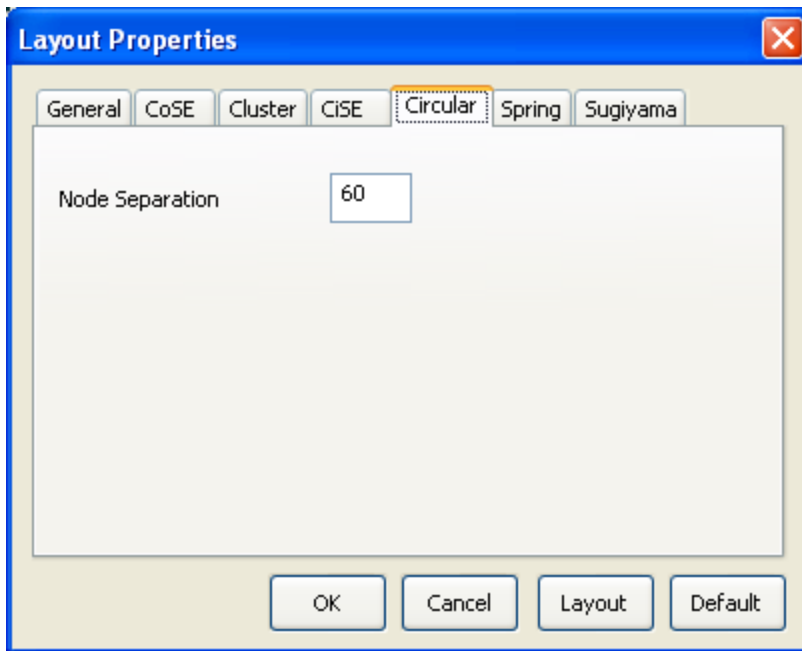


Figure 49 Circular layout options

Node Separation: Desired distance between two neighboring nodes on the circle

Spring Layout

The spring layout is perhaps the simplest force-directed layout. Here the edges of the network are represented as springs. All the springs have a natural length, measured in the same units as the screen co-ordination system, which they attempt to achieve constantly. If the spring is shorter than its natural length it extends, pushing the nodes at either end of the edge apart. If the spring is longer than its natural length it contracts, pulling the nodes at either end together. The force exerted by the spring is proportional to the difference between its current length and its natural length. Nodes linked together tend to form cluster so a repulsive force is also added. The lengths are changed iteratively to obtain a well spaced out layout by minimizing the total energy.

We use an implementation by the GINY graph library [4], based on the Kamada-Kawai Spring Layout Algorithm [5] for this style.

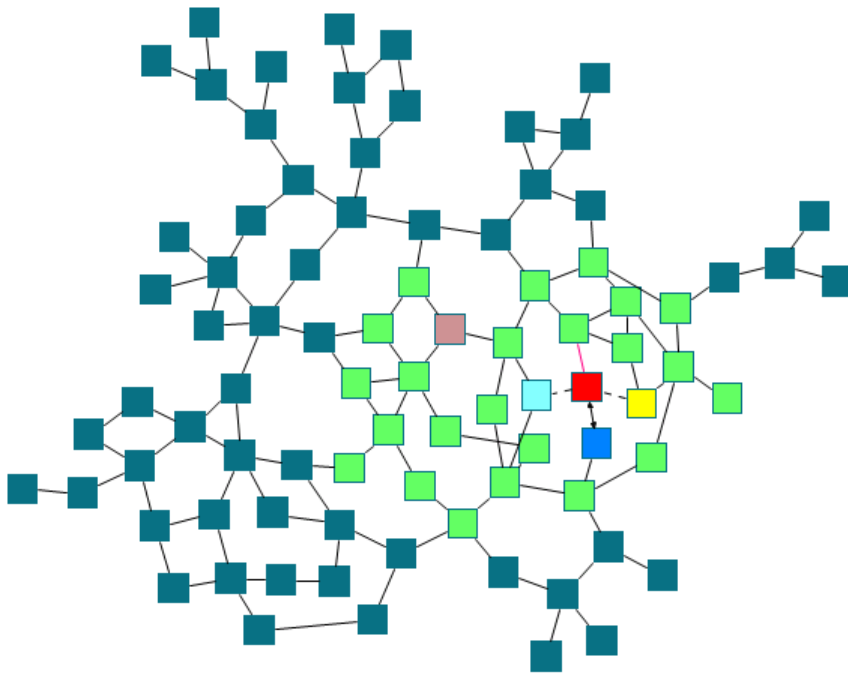


Figure 50 Sample Spring layout produced by Chisio

Understanding the Spring Layout Algorithm

- Compute distances between nodes by using shortest paths algorithm;
- At each iteration, move a node in the direction, which decreases the total energy most.

Layout Options

Spring layout options are shown in Figure 51.

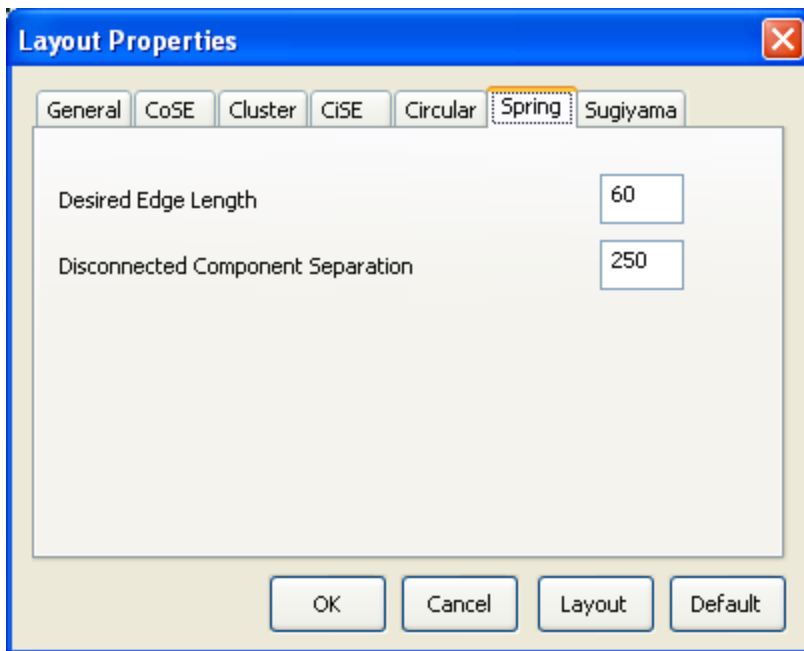


Figure 51 Spring layout options

Desired Edge Length: Desired length of an edge

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

Hierarchical Layout

The hierarchical layout is by far the most popular layout style for directed graphs, emphasizing the precedence relationships among graph objects. This layout can represent organizational and information management system dependencies, as well as process models, software call graphs, and workflows.

Our hierarchical layout is an implementation of the Sugiyama algorithm [7], taken from JGraph [5]. We have extended the implementation to support compound graphs and variable node dimensions.

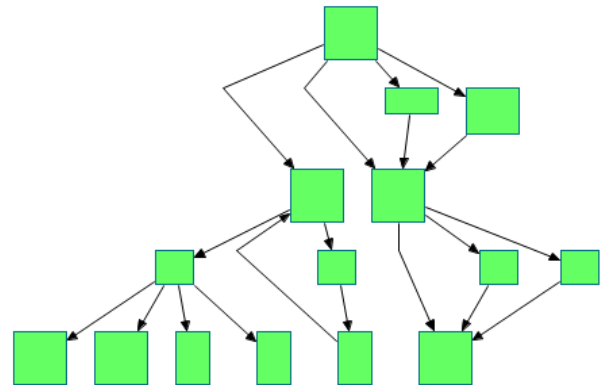


Figure 52 Sample Hierarchical layout

Hierarchical Layout Steps

The hierarchical layout algorithm divides the nodes of a graph into levels and directs edges from top to bottom in a drawing oriented top-to-bottom as follows:

- For compound node support, inter-graph edges are converted into intra-edges;
- All roots in the graph are identified;

- By using these roots, nodes are topologically sorted with a DFS. If the graph contains cycles, the edges causing the cycles are reversed;
- Calculated topological information is used to create levels;
- If bend points are allowed to be created, bend points are created for multi level edges;
- Number of edge crossings between neighboring levels are reduced via the barycenter method;
- Levels of nodes are adjusted to minimize edge lengths;
- Spacing between levels and nodes are calculated;
- These values are converted into specific positions.

Layout Options

Hierarchical layout options are shown in Figure 53.

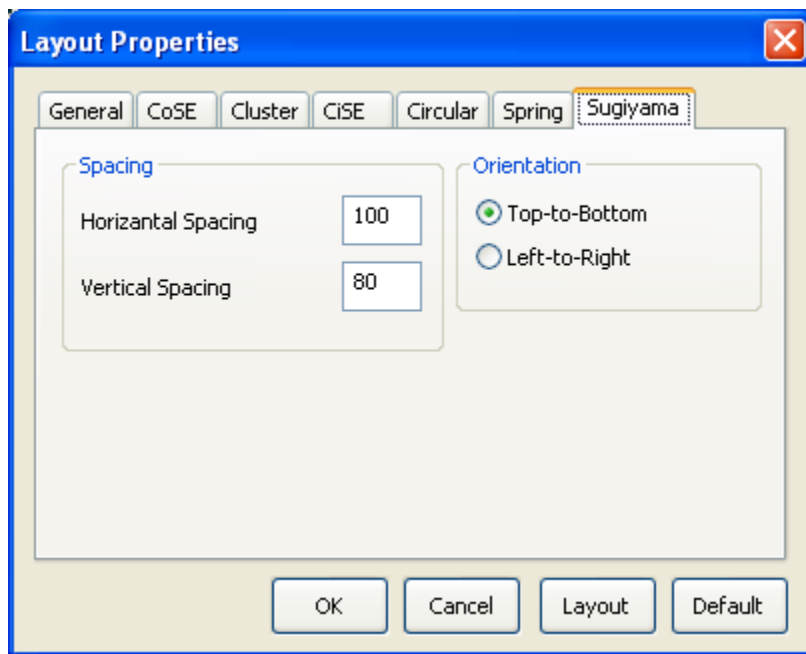


Figure 53 Hierarchical layout options

Horizontal Spacing: The space between two nodes in the same level

Vertical Spacing: The space between two subsequent levels

Orientation: Whether the drawing should from left-to-right (roots at the left, leaves at the right of the drawing) or top-to-bottom.

Third-Party Software License Agreements

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

Chisio makes use of Eclipse Graph Editing Framework (GEF) version 3.1 distributed under license Eclipse Public License – v 1.0, for drawing and editing pathways. Silk icons, which are distributed under Creative Commons Attribution 2.5 License, have been used to design icons for the toolbar.

Eclipse Public License – v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

- a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and
- b) in the case of each subsequent Contributor:

- i) changes to the Program, and

- ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents " mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

- a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.
- b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.
- c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.
- d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

- a) it complies with the terms and conditions of this Agreement; and
- b) its license agreement:
 - i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
 - ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;
 - iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and
 - iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

- a) it must be made available under this Agreement; and

b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

Silk Icons

Creative Commons Attribution 2.5 License

THE WORK (AS DEFINED BELOW) IS PROVIDED UNDER THE TERMS OF THIS CREATIVE COMMONS PUBLIC LICENSE ("CCPL" OR "LICENSE"). THE WORK IS PROTECTED BY COPYRIGHT AND/OR OTHER APPLICABLE LAW. ANY USE OF THE WORK OTHER THAN AS AUTHORIZED UNDER THIS LICENSE OR COPYRIGHT LAW IS PROHIBITED.

BY EXERCISING ANY RIGHTS TO THE WORK PROVIDED HERE, YOU ACCEPT AND AGREE TO BE BOUND BY THE TERMS OF THIS LICENSE. THE LICENSOR GRANTS YOU THE RIGHTS CONTAINED HERE IN CONSIDERATION OF YOUR ACCEPTANCE OF SUCH TERMS AND CONDITIONS.

1. Definitions

1. "Collective Work" means a work, such as a periodical issue, anthology or encyclopedia, in which the Work in its entirety in unmodified form, along with a number of other contributions, constituting separate and independent works in themselves, are assembled into a collective whole. A work that constitutes a Collective Work will not be considered a Derivative Work (as

defined below) for the purposes of this License.

2. "Derivative Work" means a work based upon the Work or upon the Work and other pre-existing works, such as a translation, musical arrangement, dramatization, fictionalization, motion picture version, sound recording, art reproduction, abridgment, condensation, or any other form in which the Work may be recast, transformed, or adapted, except that a work that constitutes a Collective Work will not be considered a Derivative Work for the purpose of this License. For the avoidance of doubt, where the Work is a musical composition or sound recording, the synchronization of the Work in timed-relation with a moving image ("synching") will be considered a Derivative Work for the purpose of this License.

3. "Licensor" means the individual or entity that offers the Work under the terms of this License.

4. "Original Author" means the individual or entity who created the Work.

5. "Work" means the copyrightable work of authorship offered under the terms of this License.

6. "You" means an individual or entity exercising rights under this License who has not previously violated the terms of this License with respect to the Work, or who has received express permission from the Licensor to exercise rights under this License despite a previous violation.

2. Fair Use Rights. Nothing in this license is intended to reduce, limit, or restrict any rights arising from fair use, first sale or other limitations on the exclusive rights of the copyright owner under copyright law or other applicable laws.
3. License Grant. Subject to the terms and conditions of this License, Licensor hereby grants You a worldwide, royalty-free, non-exclusive, perpetual (for the duration of the applicable copyright) license to exercise the rights in the Work as stated below:
 1. to reproduce the Work, to incorporate the Work into one or more Collective Works, and to reproduce the Work as incorporated in the Collective Works;
 2. to create and reproduce Derivative Works;
 3. to distribute copies or phonorecords of, display publicly, perform publicly, and perform publicly by means of a digital audio transmission the Work including as incorporated in Collective Works;
 4. to distribute copies or phonorecords of, display publicly, perform publicly, and perform publicly by means of a digital audio transmission Derivative Works.

5. For the avoidance of doubt, where the work is a musical composition:

i. Performance Royalties Under Blanket Licenses. Licensor waives the exclusive right to collect, whether individually or via a performance rights society (e.g. ASCAP, BMI, SESAC), royalties for the public performance or public digital performance (e.g. webcast) of the Work.

ii. Mechanical Rights and Statutory Royalties. Licensor waives the exclusive right to collect, whether individually or via a music rights agency or designated agent (e.g. Harry Fox Agency), royalties for any phonorecord You create from the Work ("cover version") and distribute, subject to the compulsory license created by 17 USC Section 115 of the US Copyright Act (or the equivalent in other jurisdictions).

6. Webcasting Rights and Statutory Royalties. For the avoidance of doubt, where the Work is a sound recording, Licensor waives the exclusive right to collect, whether individually or via a performance-rights society (e.g. SoundExchange), royalties for the public digital performance (e.g. webcast) of the Work, subject to the compulsory license created by 17 USC Section 114 of the US Copyright Act (or the equivalent in other jurisdictions).

The above rights may be exercised in all media and formats whether now known or hereafter devised. The above rights include the right to make such modifications as are technically necessary to exercise the rights in other media and formats. All rights not expressly granted by Licensor are hereby reserved.

4. Restrictions. The license granted in Section 3 above is expressly made subject to and limited by the following restrictions:

1. You may distribute, publicly display, publicly perform, or publicly digitally perform the Work only under the terms of this License, and You must include a copy of, or the Uniform Resource Identifier for, this License with every copy or phonorecord of the Work You distribute, publicly display, publicly perform, or publicly digitally perform. You may not offer or impose any terms on the Work that alter or restrict the terms of this License or the recipients' exercise of the rights granted hereunder. You may not sublicense the Work. You must keep intact all notices that refer to this License and to the disclaimer of warranties. You may not distribute, publicly display, publicly perform, or publicly digitally perform the Work with any technological measures that control access or use of the Work in a manner inconsistent with the terms of this License Agreement. The above applies to the Work as incorporated in a Collective Work, but this does not require the Collective Work apart from the Work itself to be made subject to the terms of this License. If You create a Collective Work, upon notice from any Licensor You must, to the extent practicable, remove from the Collective Work any credit as required by clause 4(b), as requested. If You create a Derivative Work, upon notice from any Licensor You must, to the extent practicable, remove from

the Derivative Work any credit as required by clause 4(b), as requested.

2. If you distribute, publicly display, publicly perform, or publicly digitally perform the Work or any Derivative Works or Collective Works, You must keep intact all copyright notices for the Work and provide, reasonable to the medium or means You are utilizing: (i) the name of the Original Author (or pseudonym, if applicable) if supplied, and/or (ii) if the Original Author and/or Licensor designate another party or parties (e.g. a sponsor institute, publishing entity, journal) for attribution in Licensor's copyright notice, terms of service or by other reasonable means, the name of such party or parties; the title of the Work if supplied; to the extent reasonably practicable, the Uniform Resource Identifier, if any, that Licensor specifies to be associated with the Work, unless such URI does not refer to the copyright notice or licensing information for the Work; and in the case of a Derivative Work, a credit identifying the use of the Work in the Derivative Work (e.g., "French translation of the Work by Original Author," or "Screenplay based on original Work by Original Author"). Such credit may be implemented in any reasonable manner; provided, however, that in the case of a Derivative Work or Collective Work, at a minimum such credit will appear where any other comparable authorship credit appears and in a manner at least as prominent as such other comparable authorship credit.

5. Representations, Warranties and Disclaimer

UNLESS OTHERWISE MUTUALLY AGREED TO BY THE PARTIES IN WRITING, LICENSOR OFFERS THE WORK AS-IS AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND CONCERNING THE WORK, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, INCLUDING, WITHOUT LIMITATION, WARRANTIES OF TITLE, MERCHANTIBILITY, FITNESS FOR A PARTICULAR PURPOSE, NON-INFRINGEMENT, OR THE ABSENCE OF LATENT OR OTHER DEFECTS, ACCURACY, OR THE PRESENCE OF ABSENCE OF ERRORS, WHETHER OR NOT DISCOVERABLE. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO SUCH EXCLUSION MAY NOT APPLY TO YOU.

6. Limitation on Liability. EXCEPT TO THE EXTENT REQUIRED BY APPLICABLE LAW, IN NO EVENT WILL LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY FOR ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES ARISING OUT OF THIS LICENSE OR THE USE OF THE WORK, EVEN IF LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. Termination

1. This License and the rights granted hereunder will terminate automatically upon any breach by You of the terms of this License. Individuals or entities who have received Derivative Works or Collective Works from You under this License, however, will not have their licenses terminated provided such individuals or entities remain in full compliance with those licenses. Sections 1, 2, 5, 6, 7, and 8 will survive any termination of this License.

2. Subject to the above terms and conditions, the license granted here is perpetual (for the duration of the applicable copyright in the Work). Notwithstanding the above, Licensor reserves the right to release the Work under different license terms or to stop distributing the Work at any time; provided, however that any such election will not serve to withdraw this License (or any other license that has been, or is required to be, granted under the terms of this License), and this License will continue in full force and effect unless terminated as stated above.

8. Miscellaneous

1. Each time You distribute or publicly digitally perform the Work or a Collective Work, the Licensor offers to the recipient a license to the Work on the same terms and conditions as the license granted to You under this License.
2. Each time You distribute or publicly digitally perform a Derivative Work, Licensor offers to the recipient a license to the original Work on the same terms and conditions as the license granted to You under this License.
3. If any provision of this License is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this License, and without further action by the parties to this agreement, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.
4. No term or provision of this License shall be deemed waived and no breach consented to unless such waiver or consent shall be in writing and signed by the party to be charged with such waiver or consent.
5. This License constitutes the entire agreement between the parties with respect to the Work licensed here. There are no understandings, agreements or representations with respect to the Work not specified here. Licensor shall not be bound by any additional provisions that may appear in any communication from You. This License may not be modified without the mutual written agreement of the Licensor and You.

References

- [1] <http://graphml.graphdrawing.org>
- [2] 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.
- [3] H. He and O. Sýkora, “New circular drawing algorithms”, In Proc. Workshop on Information Technologies - Applications and Theory (ITAT'04), 2004.
- [4] <http://csbi.sourceforge.net>
- [5] T. Kamada and S. Kawai, “An algorithm for drawing general undirected graphs”, Information Processing Letters, 31, pp. 7-15, 1989.
- [6] <http://www.jgraph.com>
- [7] K. Sugiyama, S. Tagawa, and M. Toda, “Methods for Visual Understandings of Hierarchical System Structures”, IEEE Transactions in Systems, Man, and Cybernetics, vol. smc-11, no.2, pp. 109-125, February 1981.