Cytoscape 2.0 Manual

1. Introduction

Cytoscape is an open-community software project for integrating biomolecular interaction networks with high-throughput expression data and other molecular states into a unified conceptual framework. Although applicable to any system of molecular components and interactions, Cytoscape is most powerful when used in conjunction with large databases of protein-protein, protein-DNA, and genetic interactions that are increasingly available for humans and model organisms. A software "Core" provides basic functionality to layout and query the network; to visually integrate the network with expression profiles, phenotypes, and other molecular states; and to link the network to databases of functional annotations. The Core is extensible through a straightforward PlugIn architecture, allowing rapid development of additional computational analyses and features. **The central organizing metaphor of Cytoscape is a network graph, with genes, proteins, and molecules represented as nodes and interactions represented as links, i.e. edges, between nodes.**

Development

Cytoscape is a collaborative project between the Institute for Systems Biology (Dr. Benno Schwikowski), the University of California San Diego (Dr. Trey Ideker), and Memorial Sloan-Kettering Cancer Center (Dr. Chris Sander).

Schwikowski Lab: http://www.systemsbiology.org/personal/benno
Ideker Lab: http://www-bioeng.ucsd.edu/faculty/area/ideker_lab/
Sander Lab: http://www.cbio.mskcc.org/

License

Cytoscape is protected under the GNU LGPL (Lesser General Public License). The License is included as an appendix to this manual, but can also be found online: http://www.gnu.org/copyleft/lesser.txt Cytoscape also includes a number of other open source libraries, which are detailed in Section 10, Acknowledgements below.

2. Launching Cytoscape

Currently, Cytoscape runs under Java on Linux, Windows, and Mac OS X. Although Cytoscape handles arbitrary types and sizes of interaction network, it is most powerful when used in conjunction with large interaction data sets such as are currently available for species such as *Saccharomyces cerevisiae* (budding yeast).

(1) Download and unpack the distribution. Cytoscape is distributed as a compressed archive (tar.gz or zip) containing the following files and directories:

`cytoscape.jar`	Main Cytoscape application (Java archive)
`cytoscape.props`	User-configurable properties and preferences
`vizmap.props`	User-configurable visual mapping preferences
`cytoscape.sh`	Shell script used to run Cytoscape (Linux, Mac OS X)
`cytoscape.bat`	Shell script used to run Cytoscape (Windows)
`LICENSE.txt`	Cytoscape GNU License
`Cytoscape2_0Manual.pdf`	Cytoscape 2.0 Manual (the document you are reading now)
`sampleData/`	
`galFiltered.gml`	Sample molecular interaction network file *
`galFiltered.sif`	Identical network in Simple Interaction Format *
`galExpData.pvals`	Sample gene expression matrix file *
`BINDyeast.sif`	Network of all yeast protein-protein interactions in the BIND database as of March, 2004 **
`GO/`	Directory containing Gene Ontology database entries (currently for yeast only). Info in this directory is used to associate gene names with synonyms as well as process, function, and cellular location data.

`plugins/` Directory containing cytoscape PlugIns, in .jar format.

* Sample data sets taken from Ideker et al, Science 292:929 (2001)
** Obtained from data hosted at http://www.blueprint.org/bind/bind_downloads.html.

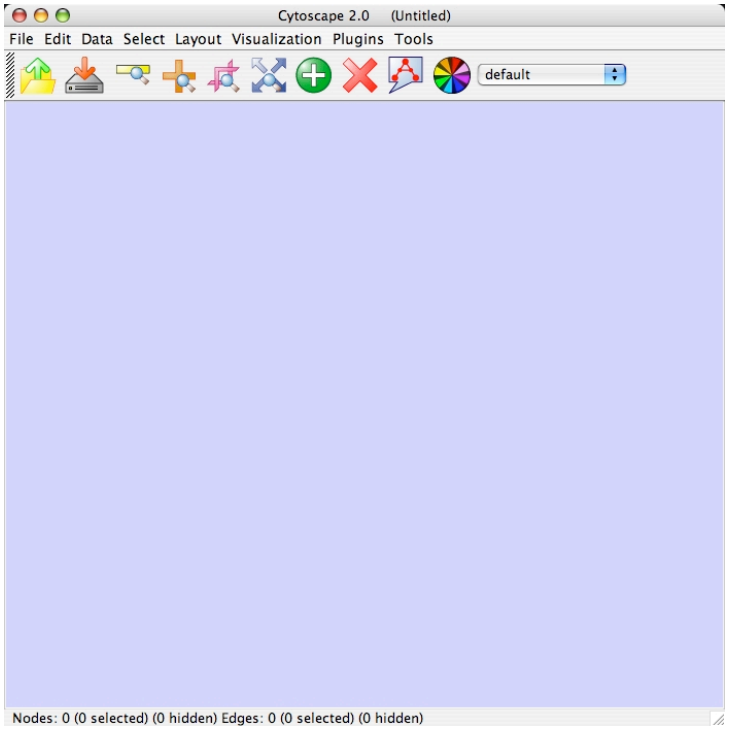
(2) If necessary, install Java. If not already installed on your computer, download and install the Java 2 Runtime Environment, version 1.4.1 or higher. It can be found at:

http://java.sun.com/j2se/index.jsp

(3) Launch the application by running "cytoscape.sh" from the command line (Linux or Mac OS X) or double-clicking "cytoscape.bat" (Windows). Alternatively, you can pass the .jar file to Java directly using the command "java -jar cytoscape.jar". In Windows, it is also possible to directly double-click the .jar file to launch it. However, this does not allow specification of command-line arguments (such as the location of the GO data directory, see the *Command Line Arguments* section for details).

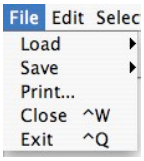
! Important Note:
For the application to work properly, ALL FILES MUST BE LEFT IN THE DIRECTORY IN WHICH THEY ARE UNPACKED. The core Cytoscape application assumes this directory structure when looking for certain files, such as cytoscape.props, vizmap.props, and the GO/ database.

Cytoscape Window
When you succeed in launching Cytoscape, a window will appear that looks like this:



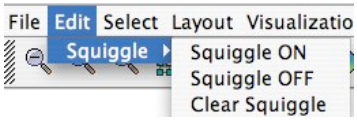
3. Quick Tour of The Menus

File



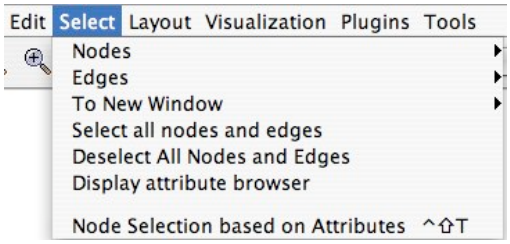
The File menu contains most basic file functionality: File / Load for loading files; File / Save for saving; File / Print for printing to either a printer or a PostScript file. File / Close closes only this window of Cytoscape, leaving other Cytoscape windows open. If this is this is the last open Cytoscape window, File / Close also exits Cytoscape. File / Exit closes all windows of Cytoscape and exits the program. Details of the Load and Save sub-menus can be found in the *Building and Storing Networks* section.

Edit



The Edit menu contains a Squiggle feature that enables you to mark up your network. This can be particularly useful during live presentations.

Select

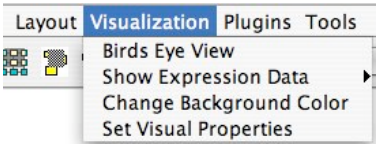


The Select menu contains methods and operations for selecting nodes and edges, operating on existing selections, and displaying the attribute browser. More details about the Nodes, Edges, and To New Window sub-menus can be found in the *Selection and Filtering* section.

Layout

The Layout menu has an array of features for organizing the graph visually; these features are explored in-depth in the *Visualization* section. The main features include arranging the entire graph according to one of several algorithms; aligning and rotating groups of nodes; and adjusting the size of the graph.

Visualization



The Visualization menu provides options for changing the mapping from biological data to a visual representation: colors of nodes, thickness of edges, etc. These features are explored in-depth in the *Visual Styles* section. This menu also provides a Bird's Eye view of your entire graph, and multiple options for viewing expression data.

PlugIns

The PlugIns menu will contain all PlugIns that you have chosen to load in your cytoscape.props file.

Note: A complete list of Cytoscape PlugIns is available online at: http://cytoscape.org/plugins.html

4. Command Line Arguments

Cytoscape recognizes a number of optional command line arguments, including run-time specification of network files and expression data:

-g <GML network filename>	(xxx.gml) Loads a network file in GML format (see *5. Building and Storing Interaction Networks*)
-i <SIF interactions filename>	(yyy.sif) Loads a network file in SIF format (see *5. Building and Storing Interaction Networks*)
-b <bioData directory>	(e.g. GO/annotationsAndSynonyms) Specifies which directory to use for the BioDataServer annotations.
-e <expression filename>	(zzz.pvals) Loads an expression data file (see 6. Loading Gene Expression Data)
-x	Prevents expression file from also loading into Cytoscape graph attributes. (See *7. Node and Edge Attributes*)
-n <nodeAttributes filename>	(one or more) Loads node attributes files (see *7. Node and Edge* Attributes)
-j <edgeAttributes filename>	(one or more) Loads edge attributes files (see *7. Node and Edge* Attributes)
-h	help: display these command line arguments
-v	display version
--JLD	specifies a directory in which PlugIn .jar's reside.

Most data sets may also be loaded after Cytoscape is running. See the sections on *6. Loading Gene Expression* Data and *7. Node and Edge* Attributes for details.

Additional command line arguments that are not recognized by the Cytoscape core are passed to the PlugIn modules. Please refer to the documentation for each specific PlugIn for more details.

5. Building and Storing Interaction Networks

Cytoscape reads an interaction network in two ways: from a simple interaction file (SIF or .sif format) or from a universal format known as Graph Markup Language (GML or .gml format). SIF specifies nodes and interactions only, while GML stores additional information about network layout and allows network data exchange with a variety of other network display programs. Typically, SIF is used to import interactions when building a network for the first time. Once the interactions have been loaded and layout has been performed, the network may be saved to and subsequently reloaded from GML format in future Cytoscape sessions. Both SIF and GML are ASCII text files, and you can edit and view them in a regular text editor.

SIF FORMAT:
Lines in the SIF file specify a source node, an interaction type, and one or more target nodes:

```
geneA <interaction type> geneB
geneC <interaction type> geneA
geneD <interaction type> geneE geneF geneB
geneG
...
geneY <interaction type> geneZ
```

In the network specified by this file, genes are represented by nodes, and interactions are represented by edges between nodes. For compactness, a gene also represents its corresponding protein. Nodes may also be used to represent compounds and reactions (or anything else) instead of genes, but this is non-standard, as yet. Note that it is possible to specify an isolated node with no interactions, as in the line "geneG" above.

Gene names must be unique. If the network is to be integrated with Gene Ontology (GO) or gene expression data, the gene names must exactly match the systematic ORF names specified in the other data files. We strongly encourage naming genes and proteins by their systematic ORF name; common names may be displayed on the screen for ease of interpretation, so long as these are available to the program in the bioData directory (Cytoscape ships with all yeast ORF-to-common name mappings in a synonym table within the GO/ directory).

The tag `<interaction type>` should be one of:
- `pp` ..... protein – protein interaction
- `pd` ..... protein -> DNA
  (e.g. transcription factor binding upstream of a regulating gene.)

Additional interaction types are also possible, but as yet, nonstandard, e.g.:
- `pr` ..... protein -> reaction
- `rc` ..... reaction -> compound

`cr` ..... compound -> reaction
`gl` ..... genetic lethal relationship

Any text string will work, but these are the conventions that have been followed thus far.

GML FORMAT:

In contrast to SIF, GML is a rich graph format language supported by many other graph visualization packages. Its file format specification is available at:

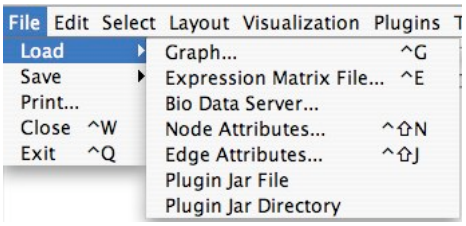
http://www.infosun.fmi.uni-passau.de/Graphlet/GML/

It is generally not necessary to modify the content of a GML file directly. Once a network is built in SIF format and then laid out, the layout is preserved by saving to and loading from GML.

COMMANDS:

Load and save network files using the File menu of Cytoscape. Network files may also be loaded directly from the command line using the –i (SIF format) or -g (GML format) options.

FOR EXAMPLE:



To load a sample molecular interaction network in SIF format, use the menu File / Load / Graph. In the resulting file dialog box, select the file "sampleData/galFiltered.sif". After a few seconds, a small network of 329 nodes should appear in the main window. To load the same interaction network as a GML, use the menu: File / Load / Graph again. In the resulting file dialog box, select the file "sampleData/galFiltered.gml". As of Cytoscape version 1.1, PlugIns can also be loaded from the File / Load menu, as can node and edge attribute files.

6. Loading Gene Expression Data

Interaction networks are certainly useful as stand-alone models. However, they are most powerful when integrated with information about the biological associated with the network, such as gene or protein expression levels. Once loaded, expression ratios/levels may be visually superimposed on the network, used in a filter to select a subset of nodes, or used to identify active modules and subsystems (see sections below). Expression data are only relevant once a network has been loaded.

FORMAT:

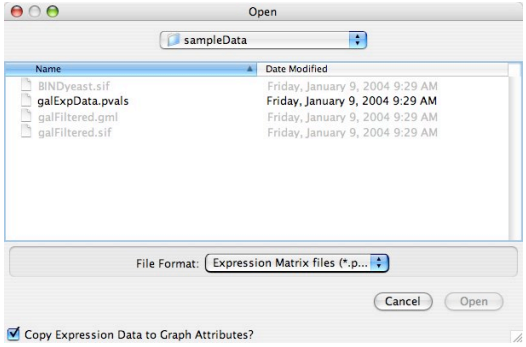
Gene expression ratios are specified over one or more experiments using an ASCII text file. The file consists of a number of space- or tab-delimited fields, one line per gene, with the following format:

```
GeneName [CommonName] ratio1 ratio2 ... ratioN [pval1 pval2 ... pvalN]
```

Brackets [] indicate fields that are optional. The first two fields are the systematic gene name followed by an optional common name. Expression ratios are provided for each experiment, optionally followed by a p-value per experiment or other measure of the significance of each ratio, i.e. whether the ratio represents a true change in expression (according to some statistical model.) Significance values are generated by a variety of software packages for analyzing expression data generated by DNA microarrays, for instance our program VERA (http://www.systemsbiology.org/VERAandSAM). A list of other microarray analysis packages is available at: http://linkage.rockefeller.edu/wli/microarray/soft.html

COMMANDS:
Load an expression data file using the File menu of Cytoscape, or by specifying the filename using the -e option at the command line. The –x command line option indicates that the expression data should not be loaded into node attributes. This is an advanced option, and is typically only used when the number of expression conditions is sufficiently large that it becomes unwieldy in the normal user interface.

FOR EXAMPLE:
Load a sample gene expression data set using the menu: File / Load / Expression Matrix File. In the resulting file dialog box (shown at right), select the file "sampleData/galExpData.pvals". As described in the following sections, Cytoscape is now ready to integrate these data with the underlying molecular interaction network. **Note:** the checkbox in the lower left corner of the file dialog asks whether to "Copy Expression Data to Graph Attributes" – un-checking this box has the same effect is as the command line option -x, and it is left checked by default.



7. Node and Edge Attributes

Node and edge attribute files are very simply formatted: A node attribute file begins with the name of the attribute on the first line, and on each following line, has the name of the node, followed by an equals sign, followed by the value of that attribute. For example:

```
FunctionalCategory
YAL001C = metabolism
YAR002W = apoptosis
YBL007C = ribosome
```

An edge attribute file has much the same structure, except that the name of the edge is the source node name, followed by the interaction type in parentheses, followed by the target

node name. Directionality counts, so switching the source and target will refer to a different (or perhaps non-existent) edge. Following is an example edge attributes file:

```
InteractionStrength
YAL001C (pp) YBR043W = 0.82
YMR022W (pd) YDL112C = 0.441
YDL112C (pd) YMR022W = 0.9013
```

Note that the second and third edge attribute values refer to two different edges (source and target are reversed, though the nodes involved are the same).

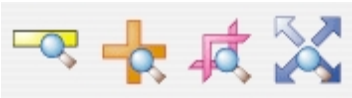
Node and edge attributes may be loaded at the command line using the –n and –j options, via the File / Load menu, or using Ctrl-Shift-N and Ctrl-Shift-J.

When expression data is loaded using an expression matrix file (See *6. Loading Gene Expression Data*), it is automatically copied into the Node Attributes data structure unless explicitly specified not to.

Edge and Node attributes can be mapped to visual properties (colors, shapes, etc.) using Visual Styles (See *9. Visual Styles*).

8. Navigation and Layout

BASIC FEATURES:
Use the zooming buttons located on the toolbar to zoom in / out of the interaction network shown in the current network display. Zoom icons are detailed below:



From Left to Right:

- Zoom In
- Zoom Out
- Zoom Selected Region
- Zoom out to Display all of Current Graph

You can also zoom in/out by right clicking and moving the mouse to the right (zoom in) or left (zoom out).

Use the left mouse button to select a node (hold down the Shift key to select more than one node). Use the right mouse button to launch a context sensitive menu with additional information about the node.

NETWORK LAYOUT:

To lay out your network using a Spring Embedded Layout, select Layout → Apply Spring Embedded Layout from the main menu. Sample screenshot is provided below:

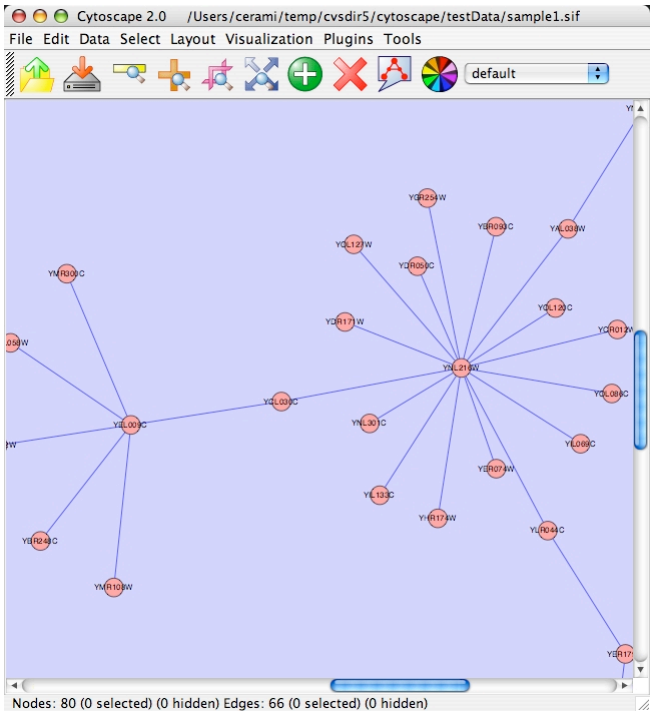


Figure: Applying the Spring Embedded Layout to a sample graph.

9. Visual Styles

With the Cytoscape Visual Style feature, you can easily customize the visual appearance of your graph. For example, you can specify a default color and shape for all nodes, use specific line types to indicate different types of interactions, or visualize gene expression data using a color gradient. All these features are available by selecting Visualization → Set Visual Properties from the main menu or clicking on the color wheel in the main button bar menu.

9.1 Introduction to Visual Styles

The Cytoscape distribution you have downloaded includes three predefined visual styles to get you started. To demonstrate these styles, try out the following example:

- Load a sample graph: From the main menu, select File → Load → Graph, and select sampleData/galFiltered.sif.

- Load a sample set of expression data: From the main menu, select File → Load → Expression Matrix File, and select sampleData/galExpData.pvals.

By default, the Visual Style labeled "default" will be automatically applied to your graph. This default style has a blue background, circular pink nodes, and blue edges (see sample screenshot below).

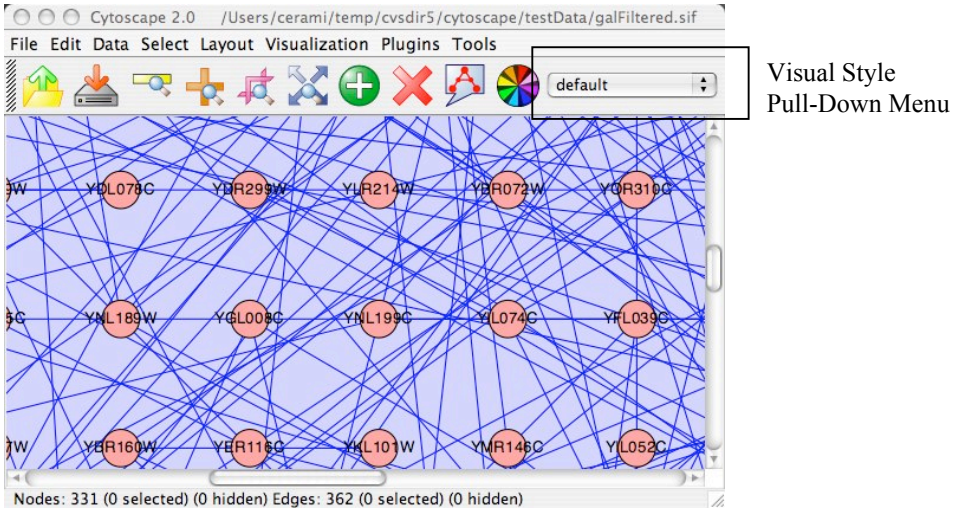


Figure: Using the default Visual Style.

The VizMap.props File: All Cytoscape Visual Style settings are automatically stored in a file called vizmap.props. Upon startup, Cytoscape will first try to locate the vizmap.props file in the "user home" directory. For example, on Windows XP, this corresponds to the user "Documents and Settings" directory, e.g. c:\Documents and Settings\cerami. On Linux or Mac OS X, this corresponds to the user home directory, e.g. /Users/cerami or ~. If no vizmap.props file is found in the user's home directory, Cytoscape will next search the current local directory.

[!] If you are upgrading from Cytoscape 1.1: If you are upgrading from Cytoscape 1.1, you may have an existing vizmap.props file in your home directory. If this is the case, you will not have the sample1 and sample2 visual styles described below. To get around this issue, backup your current vizmap.props file to safe place, and copy the new Cytoscape 2.0 vizmap.props file to your home directory.

You can flip through different visual styles by making a selection from the Visual Style pull down menu. For example, if you select "Sample1", a new visual style will be applied to your graph, and you will see a green background and round blue nodes. Additionally, protein-DNA interactions (specified with the label: pd) are drawn with dashed edges, whereas protein-protein interactions (specified with the label: pp) are drawn with solid edges (see sample screenshot below).

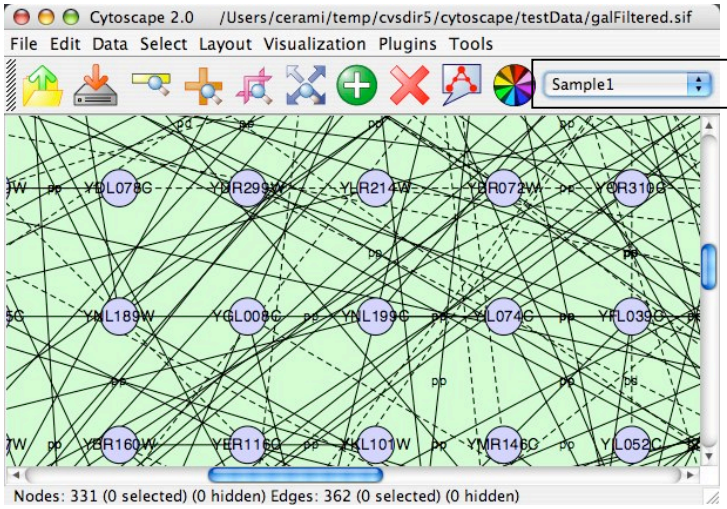


Figure: Using the Sample1 Visual Style. Protein-Protein interactions (solid lines) are now distinguishable from Protein-DNA interactions (dashed lines).

Finally, if you select "Sample2", gene expression values for each node will be colored along a color gradient between red and green (where red represents a high expression ratio, and green represents a low expression ratio.) See sample screenshot below:

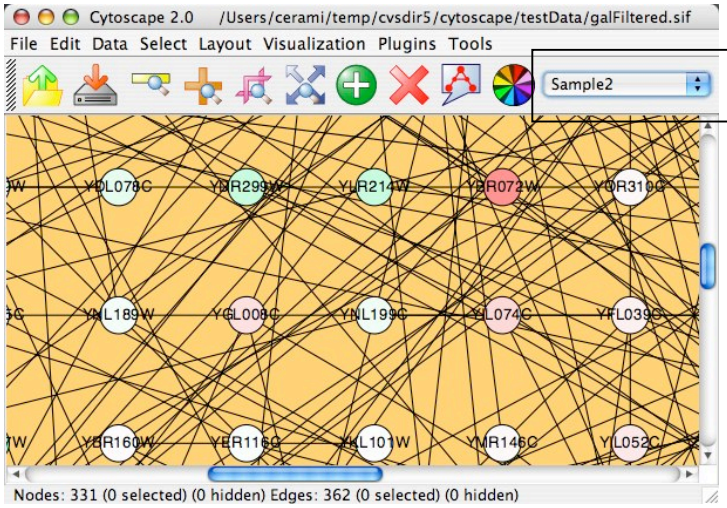


Figure: Using the Sample2 Visual Style. Gene expression values are now displayed along a red/green color gradient.

9.2 Visual Attributes, Graph Attributes and Visual Mappers

The Cytoscape Visual Mapper has three core components: *visual attributes, graph attributes* and *visual mappers*:

- A *visual attribute* is any visual setting that can be applied to your graph. For example, you can change all nodes to squares by setting the node shape visual property.

- A *graph attribute* is any attribute associated with a node or an edge. For example, each edge in a graph may be associated with a label, such as "pd" (protein-DNA interactions), or "pp" (protein-protein interactions).

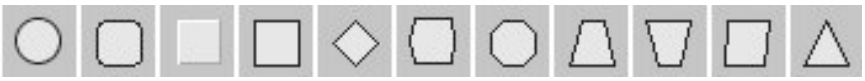
- A *visual mapper* maps graph attributes to visual attributes. For example, a visual mapper can map all protein-DNA interactions to the color blue, and all protein-protein interactions to the color red.

Cytoscape includes a large number of visual attributes. These are summarized in the tables below.

Visual Attributes Associated with Nodes:

- Node Color
- Node Border Color
- Node Border Type. The following options are available:

- Node Shape. The following options are available:



- Node Size: width and height of each node.
- Node Label: the text label for each node.
- Node Font: node font and size.

Visual Attributes Associated with Edges:

- Edge Color
- Edge Line Type. The following options are available:

- Edge Source Arrow. The following options are available:

- Edge Target Arrow. The following options are available:

- Edge Label: the text label for each edge.
- Edge Font: edge font and size.

Global Visual Properties:

- Background Color

For each visual attribute, you can specify a default value or define a visual mapping. Cytoscape currently supports three different types of visual mappers:

- **Passthrough Mapper:** graph attributes are passed directly through to visual attributes. A passthrough mapper only works for node / edge labels. For example, a passthrough mapper can draw the gene name on all nodes.
- **Discrete Mapper:** discrete graph attributes are mapped to discrete visual attributes. For example, a discrete mapper can map all protein-protein interactions to the color blue.
- **Continuous Mapper**: continuous graph attributes are mapped to visual attributes. Depending on the visual attribute, there are two types of continuous mappers:
    - **continuous to continuous mapper**: for example, you can map a continuous value (0..1) to a color gradient (red..green) or node/font size (10..100).
    - **continuous to discrete mapper**: for example, all values below 0 are mapped to square nodes, and all values above 0 are mapped to circular nodes. However, there is no way to smoothly morph between circular nodes and square nodes.

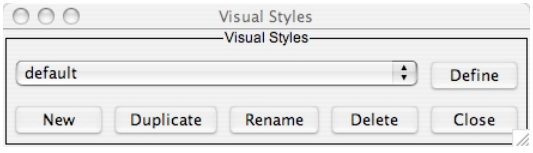
The matrix below shows visual mapper support for each visual property.

	Passthrough Mapper	Discrete Mapper	Continuous Mapper
Node Properties			
Node Color	⊘	●	●
Node Border Color	⊘	●	●
Node Border Type	⊘	●	◗
Node Shape	⊘	●	◗
Node Size	⊘	●	●
Node Label	●	●	◗
Node Font Family	⊘	●	◗
Node Font Size	⊘	●	●
Edge Properties			
Edge Color	⊘	●	●
Edge Line Type	⊘	●	◗
Edge Source Arrow	⊘	●	◗
Edge Target Arrow	⊘	●	◗
Edge Label	●	●	◗
Edge Font Family	⊘	●	◗
Edge Font Size	⊘	●	●

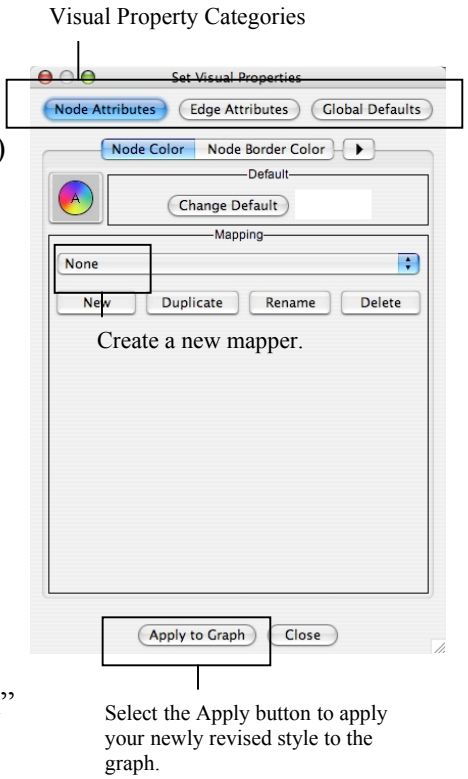
Legend	
⊘	Mapping is not supported for specified visual property.
●	Mapping is fully supported for specified visual property.
◗	Mapping is partially supported for specified visual property. Support for "continuous to continuous" mapping is not supported.

9.3 Tutorial: Creating a New Visual Style

To create a new visual style, select Visualization → Set Visual Properties from the main menu, or select the color wheel icon in the main button bar. You will now see a new Visual Styles dialog box (shown at right.)



Click the New button, and enter a name for your new visual style when prompted. Then click the Define button. You will now see the main Visual Styles Properties dialog box (shown at right.)



From this dialog box, you can flip between Node Attributes, Edge Attributes, and Global Defaults. You can also specify default values for any visual property, or define a new custom mapping.

For example, to set the default node shape to triangles, select Node Attributes → Node Shape. Then, click the "Change Default" button, and select the Triangle icon from the selection list.

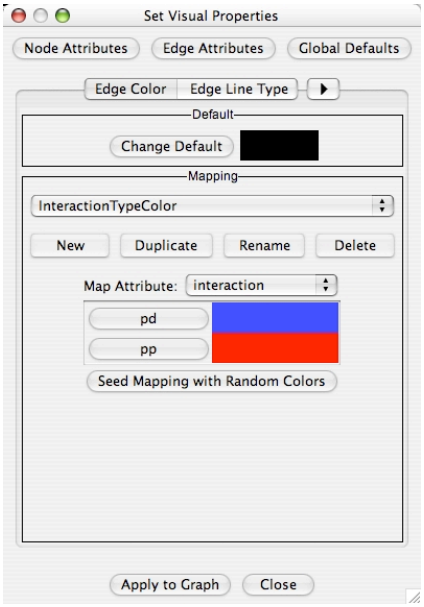
Applying Changes to the Graph

To apply your visual style to your graph, hit the "Apply to Graph" button, available in the bottom of the dialog panel.

Saving a Visual Style

When you exit Cytoscape, new visual styles or newly modified visual styles will automatically be saved in the vizmap.props file. You can therefore create a new visual style and apply it to all future graphs.

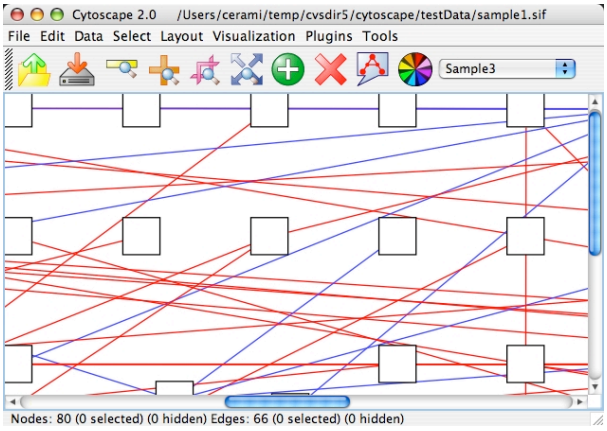
9.4 Tutorial: Creating a New Visual Style with a Discrete Mapper



The following tutorial demonstrates how to create a new visual stype with a discrete mapper. The goal is to draw Protein-DNA interactions with blue edges, and Protein-Protein interactions with red edges.

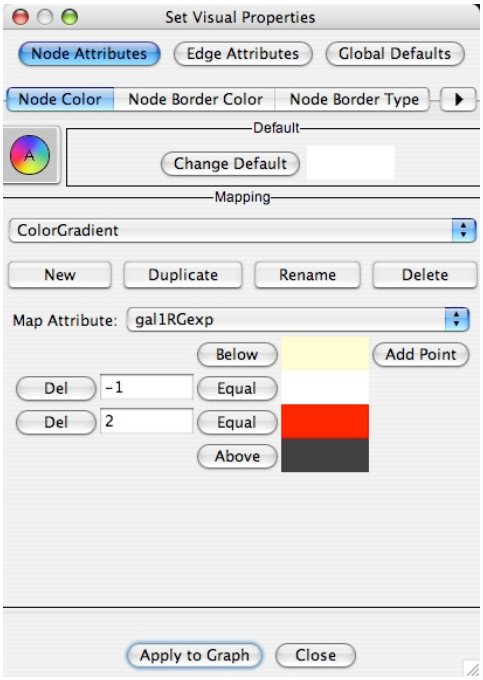
- Load a sample graph: From the main menu, select File → Load → Graph, and select sampleData/galFiltered.sif.
- Select Visualization → Set Visual Properties.
- Select "New" to create a new Visual Style. Name your new style: "Sample3".
- In the "Set Visual Properties" Dialog box, select Edge Attributes → Edge Color.
- Click the New button in the mapping panel.
- You will be prompted to select a mapping type: passthrough mapper, discrete mapper or continuous mapper (for an overview of the differences between these mappers, please refer back to section 8.2.) Select "discrete mapper", and enter a descriptive name. For example, enter: "InteractionTypeColor".
- From the "Map Attribute" pull-down menu, select "interaction." You should now see two buttons, one for pd (Protein-DNA interactions), and one for pp (Protein-Protein interactions).
- Click the "pd" button and select a blue color.
- Click the "pp" button and select a red color.
- Click the "Apply to Graph" button.

You graph should now show "pd" interactions in blue, and "pp" interactions in red. Sample screenshot is provided at right



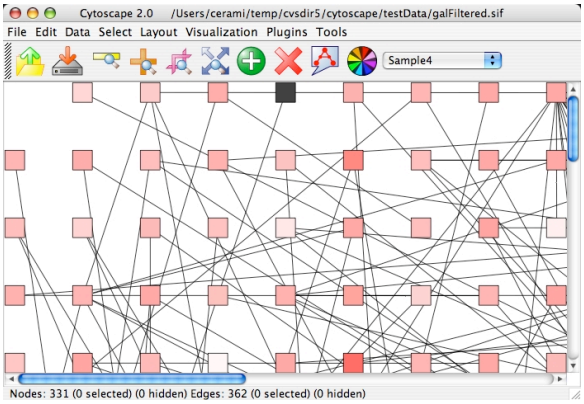
9.5 Tutorial: Visualizing Expression Data on a Network

The following tutorial demonstrates how to create a new continuous mapper. The goal is to superpose gene expression data onto a network, and to display gene expression values along a color gradient.



- Load a sample graph: From the main menu, select File → Load → Graph, and select sampleData/galFiltered.sif.
- Load a sample set of expression data: From the main menu, select File → Load → Expression Matrix File, and select sampleData/galExpData.pvals.
- Select Visualization → Set Visual Properties.
- Select "New" to create a new Visual Style. Name your new style: "Sample4".
- In the "Set Visual Properties" Dialog box, select Node Attributes → Node Color.
- Click the New button in the mapping panel.
- You will be prompted to select a mapping type: passthrough mapper, discrete mapper or continuous mapper (for an overview of the differences between these mappers, please refer back to section 8.2.) Select "continuous mapper", and enter a descriptive name. For example, enter: "ColorGradient".
- From the "Map Attribute" pull-down menu, select "gal1RGexp."
- Click the "Add Point" button twice to add two data points.
- Set the first point to "-1", Below = Yellow, Equal = White.
- Set the second point to "2", Equal = Red, Above = Black.
- Click the "Apply to Graph" button.

This visual mapper will set all nodes with a gal1RGexp value less than –1 to Yellow, and all nodes with a gal1RGExp value greater than 2 to Black. Additionally, all values between –1 and 2 will be painted with a white/red color gradient. Sample screenshot is shown at right.



10. Acknowledgements

Cytoscape is built with a number of open source 3rd party Java libraries. The Cytoscape team gratefully acknowledges the following libraries:

- The Colt Distribution: Open Source Libraries for High Performance Scientific and Technical Computing in Java. Information is available at: http://hoschek.home.cern.ch/hoschek/colt/.

- GNU Getopt in Java. Information is available at: http://www.urbanophile.com/arenn/hacking/download.html.

- Graph INterface librarY a.k.a. GINY. Information is available at: http://csbi.sourceforge.net/.

- JDOM. Information is available at: http://jdom.org.

- JUnit. Information is available at: http://junit.org.

- JGoodies Looks. Information is available at: http://www.jgoodies.com/freeware/looks/index.html.

- Piccolo. Information is available at: http://www.cs.umd.edu/hcil/jazz/.

- Type-Specific Collections Library, from Sosnoski Software Solutions, Inc. Information is available at: http://www.sosnoski.com/opensrc/tclib/.

- Xerces Java XML parser. Information is available at: http://xml.apache.org/xerces-j/.

This product includes software developed by the Apache Software Foundation (http://www.apache.org/).

This product includes software developed by the JDOM Project (http://www.jdom.org/).

Appendix: GNU Lesser General Public License

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

[This is the first released version of the Lesser GPL. It also counts as the successor of the GNU Library Public License, version 2, hence the version number 2.1.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know

that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of

its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and

therefore falls outside the scope of this License.

  However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

  When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

  If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

  Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

  6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

  You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

    a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

    b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

    c) Accompany the work with a written offer, valid for at

least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further

restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing

and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS