



# The gViz

(graph visualization tool)

## Manual

*gViz Version 1.0*

### Contents

How to cite gViz.....	1
Introduction .....	2
Java prerequisites.....	2
Connection to the gViz database .....	2
Memory settings .....	2
Troubleshooting .....	2
Main window and main toolbar.....	3
Left toolbar.....	5
Right toolbar .....	7
Search window .....	8
Statistics .....	9
Case study .....	10

### How to cite gViz

“gViz: a novel co-expression networks visualization tool”

*Raphaël Helaers<sup>#</sup>, Éric Bareke<sup>#</sup>, Bertrand de Meulder, Michael Pierre, Sopia Depiereux, Naji Habra & Éric Depiereux*

**Journal ...**

<sup>#</sup> These authors contributed equally

## Introduction

gViz allows the display and manipulation of interactions networks among a selection of probe sets, based on a given dataset and its adjacency matrix. The adjacency matrix should be obtained by software that can infer networks from data, like MINET (<http://minet.meyerp.com>) that uses mutual information to achieve it. The input matrix must be based on probe set id's, but gViz can convert them and display other associated identifiers: genes (Gene symbol, Entrez, Ensembl, Unigene), proteins (SwissProt) or diseases (OMIM). gViz can display the whole network, or sub-networks by simply selecting the probe sets to use, and it can also filter the displayed edges by their weight. Many tools are available to manipulate and traverse the network, generate statistics or linking external references (like Pubmed, Gene Ontology or related KEGG networks).

gViz is written in Java and uses JUNG (Java Universal Network/Graph Framework), a Java library that provides a common and extendible language for the modeling, analysis, and visualization of data that can be represented as a graph or a network (<http://jung.sourceforge.net/>).

## Java prerequisites

The Java 1.6 Virtual Machine (VM) must be installed on your computer for running gViz. If you only have earlier Java version(s) installed, gViz won't start (you'll get an error saying the JVM cannot be launched). The Java 1.6 VM can be installed for Windows and Linux at <http://java.com>. For Mac OS X, simply run the 'Software Update' feature available on the 'Apple menu'. To check, on your Mac, if Java 6 is installed and active, simply launch the 'Java Preferences.app' available in the "Utilities" sub-folder of the "Applications" folder. Make sure that Java 6 (or later) is in the list AND active (i.e., marked). You DON'T need to remove earlier Java versions (that might be required for older softwares). You can also check the version of Java in a terminal, using the command "java -version".

## Connection to the gViz database

The gViz interface (*available for Windows, Mac OSX, and Linux at <http://urbm-cluster.urbm.fundp.ac.be/qviz>*) connects to a MySQL database running on a server at the Facultés Universitaires Notre-Dame de la Paix Namur (Belgium). It requires a connection through the port 3306. This port might be blocked by a firewall, so ask your network administrator to unblock it for your machine, if necessary.

## Memory settings

The user may define the maximum amount of memory used by gViz in the file 'gViz.vmoptions' in the gViz directory. The file contains two parameters: 'Xms' and 'Xmx' defining respectively the minimum and the maximum amount of memory used by gViz. For example '-Xmx1024m' set the maximum memory to 1024 MegaBytes. The maximum amount is defined depends on the computer capacities and the minimum is 256 Mb. If you run gViz on a 32-bit OS, you are limited to 2GB. If you run gViz on a 64-bit OS, you are not limited in the amount of allowable memory.

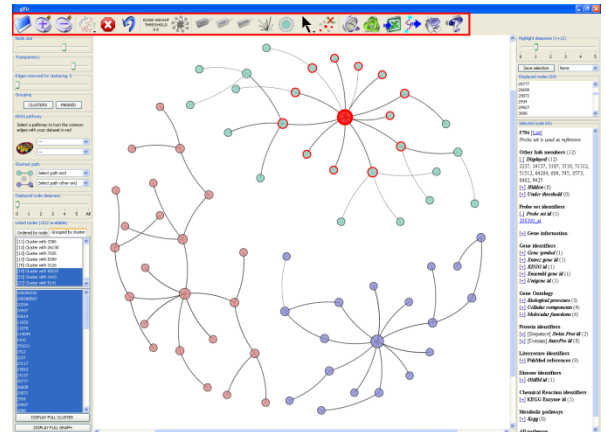
## Troubleshooting

If gViz crashes at launch, check that Java 6 or later is installed and active (see Java prerequisites) and available system memory is lower than the Xmx value in gViz.vmoptions (see memory settings).

## Main window and main toolbar



Open a file containing probe sets in an adjacency matrix. gViz will ask for converting probe sets id's in their associated identifiers (Gene Symbol, Ensembl, Entrez, Unigene, SwissProt or OMIM). Note that if more than one gene is associated with a probe set id, each gene will inherit the whole relationship of the probe set.



Tools for **zooming in/out** the networks.

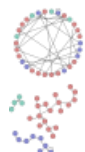


You can also use the mouse wheel (up= *zoom out*; down= *zoom in*).

**The network layout.** Drawing a network from the list of pairwise interactions is a difficult task. The network drawing algorithms (from the Java Universal Network/Graph Framework, JUNG) that are implemented within gViz are listed below. Each algorithm is run for 50 iterations to maximize quality and minimize computing time (*i.e.*, the quality of the display is usually not significantly improved after 50 iterations).



Kamada-Kawai algorithm  
Fruchterman-Reingold algorithm (default)



Circle algorithm (does not require iterations)  
Force-directed algorithm



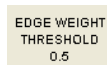
Meyer's "Self-Organizing Map" algorithm



Stop the network layout algorithm. Use if the nodes continuously move on the screen.



Restart the network layout algorithm.



Filter the considered/displayed edges on their weight. For example setting the threshold to 0.5 will not display edges having a smaller weight, or consider them the statistics or when clustering.



Filter the list of identifiers according to the number of hub members. For example, one can restrict the list of identifiers to those exhibiting >2 hub members.



Show/hide the weight of each edge.



Show/hide the name of each node.



Show/hide the name of each selected node.



Make / do not make the thickness of edges relative to their weight. Edges with higher weights are black and thick, whereas edges with lower weights are grey and thin.



Make / do not make the size of nodes relative to the number of hub members. *i.e.* nodes with  $n$  interactions can be displayed with a size  $n$  times larger than nodes with 1 interaction.

## The mouse (i.e., cursor) mode.



**Transforming mode.** Allows moving/transforming the whole network:

- Click and drag to pan (move) the whole displayed network,
- Shift-click to rotate the whole displayed network
- Ctrl-click ("apple"-click on a Macintosh) to stretch (deform) the whole network.

**Picking mode.**



- Click to select a specific node,
- 'Shift-Click' to successively select (or deselect) several nodes,
- Click outside a node and 'drag' to select all nodes included in the drawn rectangular window (or to add them to the current selection with shift-clickout-drag),
- Ctrl-click a node to select it and to centre the view on it,
- Click ON a selection and move it by dragging.



**Shortest path mode.** Using 'Shift-Click', the user selects two nodes for which the shortest path will be drawn (the selected nodes become star-shaped and the path is highlighted in cyan).  
-All picking mode options are available in shortest-path mode as well.



Remove non-highlighted elements from the displayed graph. This cannot be undone.



Open the search window.



Clicking on this icon prompts the display of an interactive window with network statistics computed on the basis of the identifiers (nodes) in the displayed network.



Export the current displayed graph as different text format : TXT (adjacency matrix), XML (GraphML), GML (Graph Modeling Language), LGL (Large Graph Layout) or NCOL (vertices couples).



Export information on the nodes participating in the shortest path between two selected nodes. Note that the order of the nodes in the exported table does not follow their order in the path.

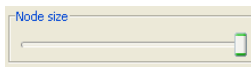


Export the displayed graph as a PNG file.



Shows the about dialog and the gViz version number.

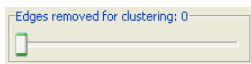
## Left toolbar



Allow adjusting the size of nodes.

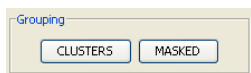


Allow adjusting the transparency of nodes.

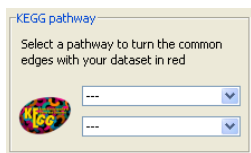


Allow generating clusters. The user chooses the number of edges to remove, and Girvan

& Newman's (2002) algorithm is used to identify which edges will be removed to generate highly connected clusters. Nodes belonging to different clusters are coloured differently and the (removed) edges are coloured in grey.



Allow grouping displayed nodes either by clusters and/or separated into "masked" and "unmasked" nodes. This tool is useful for easily select (with the cursor) e.g., clusters because selection is maintained when nodes are ungrouped.




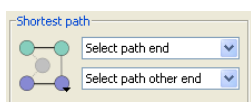
Allow selection of a KEGG pathway, to turn in red the displayed edges of your dataset that are also present in the selected pathway. You can select a pathway using its KEGG id (upper scrolling box) or its name (lower scrolling box).

When a pathway is selected you can display in KEGG (using your web browser) by clicking on the KEGG icon.

Note that you **MUST** open your dataset with ENTREZ identifiers conversion to enable the KEGG pathway functionality.

Allow:

- Selecting (using scrolling boxes) the pair of genes between which the shortest path is calculated (*i.e.*, an alternative to the use of the  cursor). The shortest path is displayed in cyan.



- Constraining the shortest path computation as follows:



Consider the 'Complete network'



Consider 'cluster limits', meaning that edges separating clusters cannot be in the shortest path.



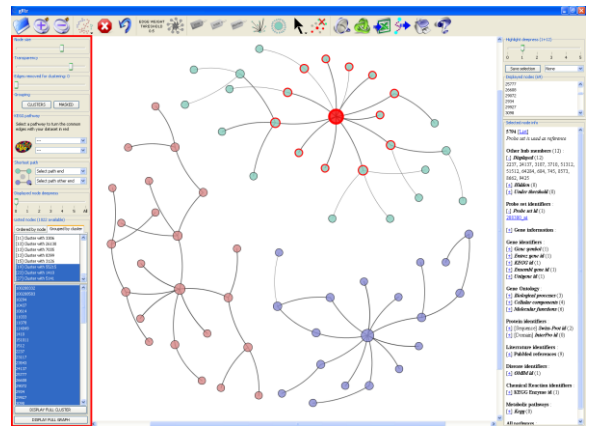
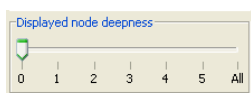
'Ignore masked nodes and edges', meaning that masked elements cannot be included in the shortest path.

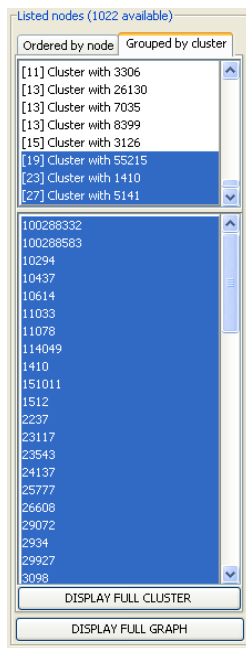



Both of the above constraints are taken into account.

Allow adjusting deepness. If  $n$  identifiers are selected and a deepness of  $d$  is chosen, the displayed graph will include the  $n$  identifiers plus all their neighbours at a distance of maximum  $d$  edges. For example, deepness of 1 means that the selected identifiers and only their direct neighbours are displayed.

The 'All' parameter allow to include all the attainable neighbours of the selected identifiers in the graph, effectively displaying the entire cluster(s) containing this (those) identifiers.



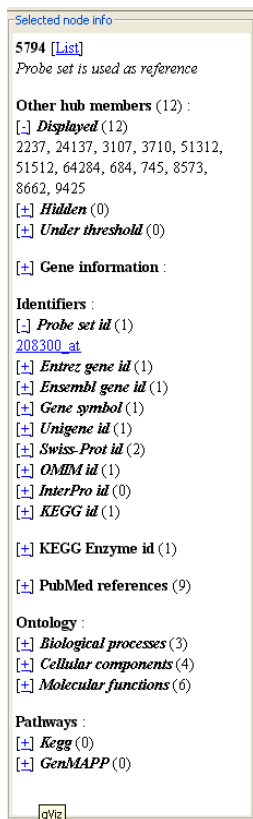
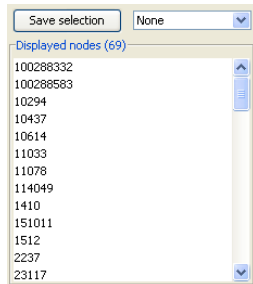
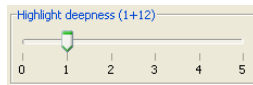


The identifiers listed in this window initially constrain all probe sets (or the corresponding genes if you use conversion) in your dataset. The list can be filtered according to hub size with the  button.

The first tab gives you the list of all identifiers sorted alphabetically. The “DISPLAY FULL GRAPH” button selects all identifiers of the list to build the corresponding graph and set the deepness slider to 0 (to minimize graph computation).

The second tab gives the list of all clusters existing in your dataset (using the available edges accordingly to the current weight threshold). There are sorted by cluster size (given between squared brackets [ ]), and clicking on one cluster gives you the list of identifiers composing it. You can select more than one cluster to display all identifiers composing them. The “DISPLAY FULL CLUSTER” button selects all identifiers in the selected clusters to build the corresponding graph and set the deepness slider to 0 (to minimize graph computation).

## Right toolbar



The “highlight deepness” may be adjusted. The number of selected and highlighted nodes is indicated.

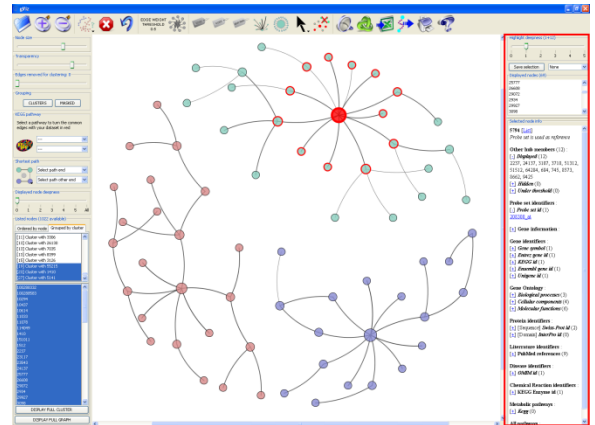
The list of displayed nodes is provided. Selecting a node in the list will also select it in the network, and center the view on it.

You can also save your current selection of nodes (with the “Save selection” button), and retrieve it later with the scrolling box.

Selecting a node on the graph or in the list of displayed nodes provides detailed information on the corresponding probe set. If more than one node is selected, clicking on [\[List\]](#) allows you to go back and forth to the list of selected nodes and each one of them. Each information category its total number of elements given between parenthesis and you can expand it by clicking the [\[+\]](#). When information is a [hyperlink \(underlined blue\)](#), you can click it to have its corresponding information in its source site (e.g. NCBI site entry if you click the Entrez gene id).

- Other hub members: all other nodes connected to the selected one. *Hidden* are the connections existing in the full graph, but absent in the current sub-graph. *Under threshold* are the connections that exist in your dataset but not considered due the current edge weight threshold.
- Probe set identifiers
- Gene information: gene description and chromosome location.
- Gene identifiers (Gene symbol, Entrez, Kegg, Ensembl, Unigene ; see important note below).
- Gene Ontology: biological processes, cellular components and molecular functions from Gene Ontology (The Gene Ontology Consortium, 2000) in which this gene is involved.
- Protein identifiers (sequences from SwissProt and domains from Interpro)
- Literature identifiers (PubMed references)
- Disease identifiers (OMIM)
- Chemical reaction identifiers (KEGG Enzyme id)
- Pathways: KEGG and GenMAPP pathways in which this gene is involved.

**IMPORTANT** : the gViz database always uses probe set id as primary reference. So even if you use the conversion of your dataset in Entrez gene id’s for example, this information panel will show the Ensemble id’s (for example) of the corresponding probe set, not only the Ensembl id of the selected Entrez gene.

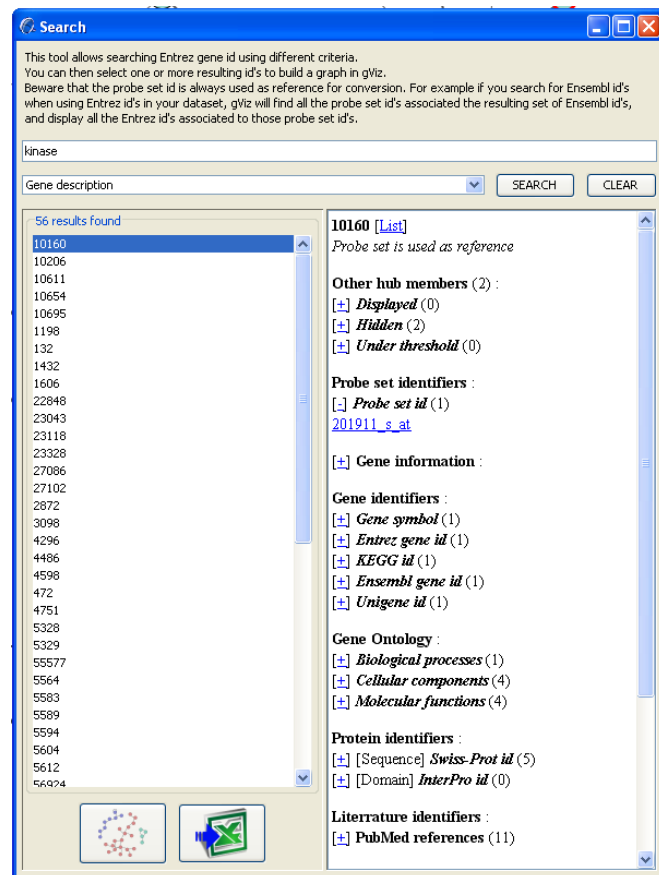




## Search window

This tool allows searching identifiers using different criteria. The resulting id's will be of the same source than the opened dataset (e.g. Entrez id's if you have converted it to Entrez). The criteria available are:

- Gene description.
- Identifiers (probe set, Entrez, Ensembl, gene symbol, Unigene, SwissProt, OMIM, Interpro, Kegg).
- Kegg Enzyme id.
- Gene Ontology term (biological process, cellular component or molecular function).

You can give uncomplete criteria, for example searching the gene description “kinase” will find “adenosine **kinase**” and “hexo**kinase** 1” ; and searching the Ensembl id “565” will find “ENSG00000122**565**” and “ENSG000001**565**15”.



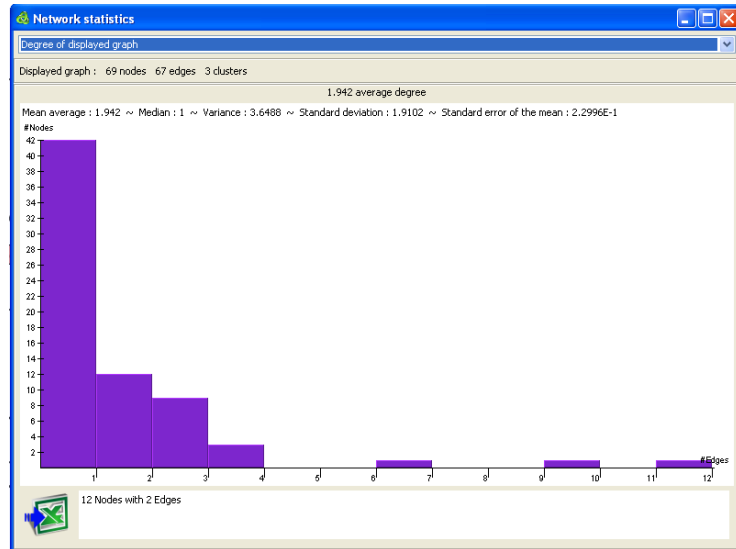
You can select one or more resulting identifiers to build a graph in gViz with the  button, or export the list with the  button.


Beware that, as for the information panel above, the probe set id is always used as reference for conversion. For example if you search for Ensembl id's when using Entrez id's in your dataset, gViz will find all the probe set id's associated the resulting set of Ensembl id's, and display all the Entrez id's associated to those probe set id's.



## Statistics

This tool gives you histograms relative to the **currently displayed** graph. The first line gives you the number of nodes, edges and clusters composing the displayed graph. Next line gives you statistical estimators for the selected data: mean average, median, variance, standard deviation and standard error of the mean. You can select any column of the histogram to have more precise information on it (on the lower white box).



The  button allows you to export the displayed histogram to a PNG image file and an Excel file with all the data (and statistical estimators).

The available histograms are:

- Degree of displayed graph: gives the distribution of the nodes degree (*i.e.*, the number of edges leaving a node).
- Diameter of displayed graph: gives the distribution of all the paths length (across all possible pairs of nodes).
- Edge weights distribution (the weights are multiplied by 100).
- Clustering coefficient of displayed graph. Gives the distribution of the clustering coefficient of each node. The clustering coefficient ( $CC\%$ ) of a node  $N$  is computed like this :
  - $K_v$  is the number of neighbours of  $N$ .
  - $N_v$  is the number of interactions between the neighbours of  $N$ , without interactions with  $N$  itself.
  - $CC\% = \frac{2 N_v}{K_v (K_v - 1)}$
- Cluster size in displayed graph: gives the distribution of the sizes of the clusters present in the displayed graph.

## Case study

In this section, we will give an example of how to collect data, generate a GraphML using Minet and R and explore the resulting graph in gViz.

**Data collection:** There are several web repositories for microarray data (GEO, Array Express). We recommend the use of our database PathEx, for easier and faster data collection.

### Network Computation:

We use the following packages to compute our networks: GCRMA, Minet and Infotheo (all of which freely available for download in Bioconductor).

Here follows the R code to compute the network from microarray *.cel* files (for R>= R2.10)

```
library(gcrma)

library(minet)

cel<-list.celfiles()

a<-ReadAffy(filename=cel)

b<-gcrma(a)

c<-exprs(b)

d<-t(c)

disc<-discretize(d, disc= "equalfreq ", nbins=sqrt(nrow(d)))

mim<-mutinformation(disc)

net<-mrnet(mim)

write.table(net, file= "net.txt ", sep= " \t")
```

Once these steps are done (which can be long, depending on the computer's speed and dataset size), one can import the network computed into gViz.

### gViz exploration

Once the network is loaded in gViz (using the 'open' button {1}, see figure 4 and the gViz manual), the list of the available nodes (genes) is shown in the lower left panel {2}. To display the entire graph, first select the 'circle' layout {3} then click on 'display full graph' {4}. The circle layout is recommended for its low computational needs; rendering a very large network can be extremely resource consuming. However, this step allows for a first overview of the network. At this step, one might want to filter his graph, using the filter on the Minet score {5}, or the filter on the number of neighbors {6}. One can also use the clustering option {7} to group similar nodes, by removing progressively the weakest edges in the graph (controlled by the 'edge removed for clustering' slider {8}). Then, using the selection tool {9}, one highlight the nodes of interest, then hit the 'remove non-

selected nodes' button {10}. The resulting sub-graph can then be shown using another, more explicit, layout (for example, 'force-directed').

The different layouts available in gViz have different uses: the 'circle' layout suits best the very big graphs, as it requires less computational power to be displayed. When working on a mid-sized sub-graph (less than 1000 nodes), the 'Kamada-Kawai' or 'Fruchterman Reingold' layouts are suggested. 'Force directed' or 'Meyer's self organizing' layouts are suited best for small (less than 100 nodes) networks, as they need more computational power to work but better discriminate the edges.

To compute the shortest path between two nodes, first select the 'shortest path' tool {11}, click on the first node of interest then maintain the SHIFT key and click on the second node. gViz automatically computes the shortest path between those two nodes, with respect to the filters possibly applied. This feature can also be controlled via the left panel {11'}. The shortest path can be exported using the 'Export shortest path' button {12}.

It is possible to export the current graph in image (click on 'export network (png)' button {12}) or in various text formats (using the 'export network (text)' button {13}).

To obtain information on a certain node, simply click on its name in the lower right panel {14}, displaying all the information contained in the PathEx database for this particular gene (for more information about PathEx, see <http://urbm-cluster.urbm.fundp.ac.be/webapps/pathex/> - Bareke E, Pierre M, Gaigneaux A, De Meulder B, Depiereux S, Berger F, Habra N, Depiereux E: *PathEx: a novel multi factors based datasets selector web tool.* BMC Bioinformatics 2010, 11:528.).

