# Package 'pathRender'

November 12, 2026

**Version** 1.80.0

**Title** Render molecular pathways

**Author** Li Long <lilong@isb-sib.ch>

**Maintainer** Vince Carey <stvjc@channing.harvard.edu>

**Depends** graph, Rgraphviz, RColorBrewer, cMAP, AnnotationDbi, methods, stats4

**Suggests** ALL, hgu95av2.db

**Description** build graphs from pathway databases, render them by Rgraphviz.

**License** LGPL

**URL** http://www.bioconductor.org

**LazyLoad** yes

**biocViews** GraphAndNetwork, Pathways, Visualization

**git_url** https://git.bioconductor.org/packages/pathRender

**git_branch** RELEASE_3_23

**git_last_commit** 5229ebd

**git_last_commit_date** 2026-04-28

**Repository** Bioconductor 3.23

**Date/Publication** 2026-11-11

## Contents

coloredGraph-class . . . . . 2
colorNodes . . . . . 3
graphcMAP . . . . . 4
plotExGraph . . . . . 4
pwayGraph-class . . . . . 5
reduceES . . . . . 6
rendercMAPPathway . . . . . 7

**Index** **9**

coloredGraph-class *Class "coloredGraph"*

## Description

a graph to which color attributes have been attached

## Objects from the Class

Objects can be created by calls of the form `new("coloredGraph", nodes, edgeL, edgemode)`. these are graphNEL instances with some additional graphData

## Slots

**nodes:** Object of class `"vector"` ~~

**edgeL:** Object of class `"list"` ~~

**edgeData:** Object of class `"attrData"` ~~

**nodeData:** Object of class `"attrData"` ~~

**renderInfo:** Object of class `"renderInfo"` ~~

**graphData:** Object of class `"list"` ~~

## Extends

Class `"graphNEL-class"`, directly. Class `"graph-class"`, by class "graphNEL", distance 2.

## Methods

**plot** `signature(x = "coloredGraph")`: ...

## Examples

```
showClass("coloredGraph")
example(randomGraph)
nn = nodes(g1)
x = runif(length(nn))
names(x) = nn
h1 = colorNodes(g1, x, colorRampPalette(brewer.pal(9, "Blues"))(length(nn)),
 pwayRendAttrs)
h1
plot(h1)
```

colorNodes *attach node coloring information to a graphNEL instance*

## Description

attach node coloring information to a graphNEL instance

## Usage

```
colorNodes(g, nodeAss, pal, attgen)
```

## Arguments

| | |
|---|---|
| g | graphNEL instance |
| nodeAss | color map for nodes: vector with elements evaluating to colors and nodes as element names |
| pal | a palette (use colorRampPalette for color interpolation) |
| attgen | attribute generating function – pwayRendAttrs is prototype |

## Value

a graphNEL instance with additional rendering data

## Author(s)

Vince Carey <stvjc@channing.harvard.edu>

## Examples

```
example(randomGraph)
nn = nodes(g1)
x = runif(length(nn))
names(x) = nn
h1 = colorNodes(g1, x, colorRampPalette(brewer.pal(9, "Blues"))(length(nn)),
 pwayRendAttrs)
h1
```

graphcMAP *obtain a graph object corresponding to a cMAP pathway*

## Description

obtain a graph object corresponding to a cMAP pathway

## Usage

```
graphcMAP(pname)
```

## Arguments

| | |
|---|---|
| pname | character token identifying a KEGG or cMAP pathway |

## Details

reuses code from pathRender but emits a graphNEL-class instance with some additional information for rendering

## Value

an instance of pwayGraph, which extends graphNEL

## Author(s)

Vince Carey <stvjc@channing.harvard.edu>

## Examples

```
G1 = graphcMAP("p53pathway")
G1
nodes(G1)
if (require(Rgraphviz)) plot(G1)
```

plotExGraph *plot a gene network, coloring nodes according to relative expression values*

## Description

plot a gene network, coloring nodes according to relative expression values

## Usage

```
plotExGraph(g, es, sampind=1, pal=colorRampPalette(brewer.pal(9, "Blues"))(length(nodes(g))), attgen=
```

### Arguments

| | |
|---|---|
| g | graph representing a gene network |
| es | an ExpressionSet instance |
| sampind | sample to be used to obtain relative expression values |
| pal | palette for coloring the nodes |
| attgen | attribute generating function |

### Details

plots a colored network on the current graphics display

### Value

as returned by Rgraphviz plot method for graphNEL instances

### Author(s)

Vince Carey <stvjc@channing.harvard.edu>

### Examples

```
library(graph)
data(pancrCaIni)
library(ALL)
data(ALL)
library(hgu95av2.db)
collap1 = reduceES( ALL, nodes(pancrCaIni), revmap(hgu95av2SYMBOL), "symbol", mean )
library(RColorBrewer)
plotExGraph( pancrCaIni, collap1, 1 )
```

---

## pwayGraph-class *Class "pwayGraph" – extension to graphNEL for pathway rendering*

---

### Description

extension to graphNEL for pathway rendering

### Objects from the Class

Objects can be created by calls of the form `new("pwayGraph", nodes, edgeL, edgemode)`.

There is a plot method that will work reasonably well if the plotting surface is big enough.

## Slots

`pwaySource`: Object of class `"character"` KEGG or BIOCARTA

`nodes`: Object of class `"vector"` pathway constituents in the native vocabulary

`edgeL`: Object of class `"list"` constituent relations in the native vocabulary

`edgeData`: Object of class `"attrData"` relationship attributes

`nodeData`: Object of class `"attrData"` node attributes

`renderInfo`: Object of class `"renderInfo"` render info

`graphData`: Object of class `"list"` this holds the special rendering attributes for edges and nodes, for nodes it seems particularly important to have fixedsize = FALSE

## Extends

Class `graphNEL-class`, directly. Class `graph-class`, by class "graphNEL", distance 2.

## Methods

**plot** `signature(x = "pwayGraph")`: renders the pathway

## Author(s)

Vince Carey <stvjc@channing.harvard.edu>

## Examples

```
showClass("pwayGraph")
G1 = graphcMAP("stresspathway")
G1@graphData$nAttrs$labels[1:10]
```

---

`reduceES` *collapse the assay values in an ExpressionSet to a set of specified genes, using a statistic when multiple probes map to a given gene*

---

## Description

collapse the assay values in an ExpressionSet to a set of specified genes, using a statistic when multiple probes map to a given gene

## Usage

```
reduceES(es, annovec, ann2featMap, pdvname="symbol", collapseFun=NULL)
```

### Arguments

| | |
|---|---|
| es | ExpressionSet instance |
| annovec | genes to retain |
| ann2featMap | either an AnnDbBimap from AnnotationDbi (typically constructed with revmap(), or a named vector mapping from symbols to probe set IDs |
| pdvname | featureData variable name to be used to hold the annotations of variables kept |
| collapseFun | statistical function for collapsing data across probes mapping to the same gene |

### Value

An ExpressionSet instance limited to genes in annovec, condensed if necessary using collapseFun to get one number per gene from multiple probes

### Author(s)

Vince Carey <stvjc@channing.harvard.edu>

### Examples

```
library(ALL)
data(ALL)
library(hgu95av2.db)
rr = revmap(hgu95av2SYMBOL)
exprs(reduceES(ALL[,1:3], c("BCL2", "CPNE1"), rr, "sym", mean))
```

---

## rendercMAPPathway *Render pathways from cMAP*

---

### Description

Build graphs based on pathway or interaction data from cMAP database, render them using Rgraphviz.

### Usage

```
rendercMAPPathway(pname, ino=0)
```

### Arguments

| | |
|---|---|
| pname | name of the pathway to render |
| ino | index of the interaction in the given pathway to render |

### Details

For a given pathway in cMAP database, we build a subgraph for each interaction in the pathway, join them together to form the graph for the complete pathway. The subgraphs for interactions and the graph for the pathway include info for rendering, such as labels/shapes/fillcolors for nodes, colors/styles/weights for edges. If user specifies an index of interaction, only the interaction is rendered. Otherwise, the complete pathway is rendered.

## Value

None. A graphical output is presented.

## Author(s)

Li Long <li.long@isb-sib.ch>

## Examples

```
rendercMAPPathway("plateletapppathway")
rendercMAPPathway("plateletapppathway", 5)
rendercMAPPathway("hsa00601")
rendercMAPPathway("hsa00601", 10)
```

# Index

∗ **classes**
coloredGraph-class, 2
pwayGraph-class, 5
∗ **graphs**
rendercMAPPathway, 7
∗ **models**
colorNodes, 3
graphcMAP, 4
plotExGraph, 4
reduceES, 6

coloredGraph-class, 2
colorNodes, 3
colorNodes,graphNEL,numeric,character,function-method
*(*colorNodes*)*, 3

graphcMAP, 4

plot,coloredGraph,ANY-method
*(*coloredGraph-class*)*, 2
plot,pwayGraph,ANY-method
*(*pwayGraph-class*)*, 5
plotExGraph, 4
pwayGraph-class, 5
pwayRendAttrs *(*coloredGraph-class*)*, 2

reduceES, 6
rendercMAPPathway, 7