Package ‘gridGraphviz’

October 13, 2022

Version 0.3-1
Depends R (>= 2.15.0), grid, graph, Rgraphviz
Imports methods
Suggests gridSVG
SystemRequirements graphviz
Title Drawing Graphs with 'grid'
Description Functions for drawing node-and-edge graphs that have been laid out by 'graphviz'. This provides an alternative rendering to that provided by the 'Rgraphviz' package, with two main advantages: the rendering provided by 'gridGraphviz' should be more similar to what 'graphviz' itself would draw; and rendering with 'grid' allows for post-hoc customisations using the named viewports and grobs that 'gridGraphviz' produces.
License GPL (>= 2)
URL https://r-forge.r-project.org/projects/gridgraph/
NeedsCompilation no
Author Paul Murrell [cre, aut], Ashley Noel Hinton [aut]
Maintainer Paul Murrell <p.murrell@auckland.ac.nz>
Repository CRAN
Date/Publication 2022-04-11 00:02:29 UTC

R topics documented:

agopenTrue ................................................................. 2
graphWidth ................................................................. 3
grid.graph ................................................................. 4

Index 6
agopenTrue

---

A function to obtain a laidout Ragraph object

Description

This function is a wrapper for the `Rgraphviz` function `agopen` and will produce a laidout Ragraph object with layout more true to the result produced by graphviz.

Usage

```r
agopenTrue(graph, name, nodes, edges, kind = NULL, layoutType = "dot",
          attrs = list(), nodeAttrs = list(), edgeAttrs = list(),
          subGList = list(), edgeMode = edgemode(graph),
          recipEdges = c("combined", "distinct"))
```

Arguments

- **graph**: An object of class `graphNEL`
- **nodes**: A list of `pNode` objects
- **edges**: A list of `pEdge` objects
- **name**: The name of the graph
- **kind**: The type of graph
- **layoutType**: Defines the layout engine. Defaults to dot, and see `graphvizCapabilities()`$layoutTypes for possible values.
- **attrs**: A list of graphviz attributes
- **nodeAttrs**: A list of specific node attributes
- **edgeAttrs**: A list of specific edge attributes
- **subGList**: A list describing subgraphs for the graph parameter
- **edgeMode**: Whether the graph is directed or undirected
- **recipEdges**: How to handle reciprocated edges, defaults to combined

Details

As of `Rgraphviz` version 2.2.1 (2013-01-31) `agopen`:

- Produces graphs of the same size as the current device, or at a default size of 7x7 inches.
- Forces nodes to fixed default height and width.
- Does not pass through edge weight information.

This function returns an Ragraph object with graph and node sizes set by graphviz or by the user. It also ensures edge weight information is passed through.

Value

An object of class `Ragraph`
graphWidth

Author(s)
Ashley Noel Hinton

References
graphviz

See Also
agopen

Examples

gnel <- new("graphNEL",
  nodes=letters[1:3],
  edgeL=list(a=list(edges=c("b", "c")),
    b=list(),
    c=list(),
    edgemode="directed")
rag <- agopenTrue(gnel, "")
grid.graph(rag)

graphWidth

Get width/height of an Ragraph object

Description
These functions will return the width or height of an Rgraphviz Ragraph object in inches.

Usage

graphWidth(graph)
graphHeight(graph)

Arguments

graph a laidout Ragraph object

Value
A numeric vector of length 1.

Author(s)
Ashley Noel Hinton

References
graphviz
See Also

Ragraph

Examples

```r
gnel <- new("graphNEL",
    nodes=letters[1:3],
    edgeL=list(a=list(edges=c("b", "c")),
               b=list(),
               c=list()),
    edgemode="directed")
rag <- agopenTrue(gnel, "")
graphWidth(rag)
graphHeight(rag)
```

---

**grid.graph**

*Draw a Node-and-Edge Graph*

**Description**

Take a description of the layout of a graph, as produced by the `Rgraphviz` package, and draw it.

**Usage**

```r
grid.graph(rag, newpage = FALSE, nodesOnTop = TRUE)
```

**Arguments**

- `rag` - An Ragraph object (must be laid out).
- `newpage` - Logical value indicating whether to draw the graph on a new page.
- `nodesOnTop` - Logical value indicating whether nodes should be drawn after edges (or vice versa).

**Author(s)**

Paul Murrell

**References**

`graphviz`
Examples

gnel <- new("graphNEL",
    nodes=letters[1:3],
    edgeL=list(a=list(edges=c("b", "c")),
                b=list(),
                c=list())),
    edgemode="directed")
rag <- agopen(gnel, "")
grid.graph(rag)
Index

* dplot
  agopenTrue, 2
  graphWidth, 3
  grid.graph, 4

agopen, 3
agopenTrue, 2

graphHeight (graphWidth), 3
graphWidth, 3
grid.graph, 4

Ragraph, 4