

Package ‘svglite’

August 29, 2016

Version 1.1.0

Title An 'SVG' Graphics Device

Description A graphics device for R that produces 'Scalable Vector Graphics'.
'svglite' is a fork of the older 'RSvgDevice' package.

Encoding UTF-8

Depends R (>= 3.0.0)

Imports Rcpp, gdtools

LinkingTo Rcpp, gdtools, BH

Suggests htmltools, testthat, xml2, covr

License GPL (>= 2)

URL <https://github.com/hadley/svglite>

BugReports <https://github.com/hadley/svglite/issues>

RoxygenNote 5.0.1

NeedsCompilation yes

Author Hadley Wickham [aut, cre],
T Jake Luciani [aut],
Matthieu Decorde [aut],
Vaudor Lise [aut],
Tony Plate [ctb] (Early line dashing code),
David Gohel [ctb] (Line dashing code and raster code),
Yixuan Qiu [ctb] (Improved styles; polypath implementation),
Håkon Malmedal [ctb] (Opacity code),
RStudio [cph]

Maintainer Hadley Wickham <hadley@rstudio.com>

Repository CRAN

Date/Publication 2016-02-09 09:47:13

R topics documented:

editSVG	2
htmlSVG	2
svglite	3
svgstring	4
xmlSVG	5

Index	6
--------------	----------

editSVG	<i>Run plotting code and open svg in OS/system default svg viewer or editor.</i>
---------	--

Description

This is useful primarily for testing or post-processing the SVG.

Usage

```
editSVG(code, ...)
```

Arguments

code	Plotting code to execute.
...	Other arguments passed on to svglite .

Examples

```
if (interactive()) {
  editSVG(plot(1:10))
  editSVG(contour(volcano))
}
```

htmlSVG	<i>Run plotting code and view svg in RStudio Viewer or web browser.</i>
---------	---

Description

This is useful primarily for testing. Requires the `htmltools` package.

Usage

```
htmlSVG(code, ...)
```

Arguments

code Plotting code to execute.
... Other arguments passed on to [svglite](#).

Examples

```
if (require("htmltools")) {  
  htmlSVG(plot(1:10))  
  htmlSVG(hist(rnorm(100)))  
}
```

svglite

An SVG Graphics Driver

Description

This function produces graphics compliant to the current w3 svg XML standard. The driver output is currently NOT specifying a DOCTYPE DTD.

Usage

```
svglite(file = "Rplots.svg", width = 10, height = 8, bg = "white",  
        pointsize = 12, standalone = TRUE)
```

Arguments

file The file where output will appear.
height, width Height and width in inches.
bg Default background color for the plot (defaults to "white").
pointsize Default point size.
standalone Produce a standalone svg file? If FALSE, omits xml header and default namespace.

Author(s)

This driver was written by T Jake Luciani <jakeluciani@yahoo.com> 2012; updated by Matthieu Decorde <matthieu.decorde@ens-lyon.fr>

References

W3C Scalable Vector Graphics (SVG): <http://www.w3.org/Graphics/SVG/Overview.htm8>

See Also

[pictex](#), [postscript](#), [Devices](#)

Examples

```
# Save to file
svglite("Rplots.svg")
plot(1:11, (-5:5)^2, type = 'b', main = "Simple Example")
dev.off()
```

svgstring

Access current SVG as a string.

Description

This is a variation on [svglite](#) that makes it easy to access the current value as a string.

Usage

```
svgstring(width = 10, height = 8, bg = "white", pointsize = 12,
standalone = TRUE)
```

Arguments

width	Height and width in inches.
height	Height and width in inches.
bg	Default background color for the plot (defaults to "white").
pointsize	Default point size.
standalone	Produce a standalone svg file? If FALSE, omits xml header and default namespace.
...	Arguments passed on to svglite .

Value

A function with no arguments: call the function to get the current value of the string.

Examples

```
s <- svgstring(); s()

plot.new(); s();
text(0.5, 0.5, "Hi!"); s()
dev.off()

s <- svgstring()
plot(rnorm(5), rnorm(5))
s()
dev.off()
```

xmlSVG	<i>Run plotting code and return svg</i>
--------	---

Description

This is useful primarily for testing. Requires the xml2 package.

Usage

```
xmlSVG(code, ..., standalone = FALSE)
```

Arguments

code	Plotting code to execute.
...	Other arguments passed on to svglite .
standalone	Produce a standalone svg file? If FALSE, omits xml header and default namespace.

Value

A `xml2::xml_document` object.

Examples

```
if (require("xml2")) {  
  x <- xmlSVG(plot(1, axes = FALSE))  
  x  
  xml_find_all(x, ".//text")  
}
```

Index

*Topic **device**
svglite, 3

Devices, 3

editSVG, 2

htmlSVG, 2

pictex, 3

postscript, 3

svglite, 2, 3, 3, 4, 5

svgstring, 4

xmlSVG, 5