

Package: rsvg (via r-universe)

June 28, 2024

Type Package

Title Render SVG Images into PDF, PNG, (Encapsulated) PostScript, or
Bitmap Arrays

Version 2.6.0

Description Renders vector-based svg images into high-quality
custom-size bitmap arrays using 'librsvg2'. The resulting
bitmap can be written to e.g. png, jpeg or webp format. In
addition, the package can convert images directly to various
formats such as pdf or postscript.

License MIT + file LICENSE

URL <https://docs.ropensci.org/rsvg/>, <https://github.com/ropensci/rsvg>

BugReports <https://github.com/ropensci/rsvg/issues>

Suggests ggplot2, knitr, magick, rmarkdown, spelling, svglite,
testthat (>= 3.0.0), webp, png

VignetteBuilder knitr

Encoding UTF-8

Language en-US

RoxygenNote 7.1.2

SystemRequirements librsvg2

Config/testthat/edition 3

Repository <https://ropensci.r-universe.dev>

RemoteUrl <https://github.com/ropensci/rsvg>

RemoteRef master

RemoteSha 15edd7af1bb5492771f265453470a3f94dd3caec

Contents

| | |
|---------------------------|---|
| librsvg_version | 2 |
| rsvg | 2 |

| | |
|--------------|----------|
| Index | 4 |
|--------------|----------|

| | |
|-----------------|------------------------|
| librsvg_version | <i>librsvg version</i> |
|-----------------|------------------------|

Description

Print the version of the librsvg library

Usage

```
librsvg_version()
```

| | |
|------|-------------------------------|
| rsvg | <i>Render SVG into Bitmap</i> |
|------|-------------------------------|

Description

Render svg image into a high quality bitmap. When both width and height are NULL, the output resolution matches that of the input. When either width or height is specified, the image is scaled proportionally. When both width and height are specified, the image is stretched into the requested size.

Usage

```
rsvg(svg, width = NULL, height = NULL, css = NULL)
rsvg_raw(svg, width = NULL, height = NULL, css = NULL)
rsvg_nativeraster(svg, width = NULL, height = NULL, css = NULL)
rsvg_webp(svg, file = NULL, width = NULL, height = NULL, css = NULL)
rsvg_png(svg, file = NULL, width = NULL, height = NULL, css = NULL)
rsvg_pdf(svg, file = NULL, width = NULL, height = NULL, css = NULL)
rsvg_svg(svg, file = NULL, width = NULL, height = NULL, css = NULL)
rsvg_ps(svg, file = NULL, width = NULL, height = NULL, css = NULL)
rsvg_eps(svg, file = NULL, width = NULL, height = NULL, css = NULL)
```

Arguments

| | |
|--------|---|
| svg | path/url to svg file or raw vector with svg data. Use charToRaw to convert an SVG string into raw data. |
| width | output width in pixels or NULL for default. |
| height | output height in pixels or NULL for default |
| css | path/url to external css file or raw vector with css data. This requires your system has a recent version of librsvg. |
| file | path to output file or NULL to return content as raw vector |

Examples

```
# create some svg
options(example.ask=FALSE)
tmp <- tempfile()
svglite::svglite(tmp, width = 10, height = 7)
ggplot2::qplot(mpg, wt, data = mtcars, colour = factor(cyl))
dev.off()

# convert directly into a vector or bitmap graphics format
rsvg_pdf(tmp, "out.pdf")
rsvg_png(tmp, "out.png")
rsvg_svg(tmp, "out.svg")
rsvg_ps(tmp, "out.ps")
rsvg_eps(tmp, "out.eps")

# render into raw bitmap array
bitmap <- rsvg(tmp, height = 1440)
dim(bitmap) # h*w*c

# render to native raster object
nr <- rsvg_nativeraster(tmp)
# grid::grid.raster(nr)

# read in your package of choice
magick::image_read(bitmap)
webp::write_webp(bitmap, "bitmap.webp", quality = 100)

# cleanup
unlink(c("out.*", "bitmap.webp"))
```

Index

`charToRaw`, [3](#)

`librsvg_version`, [2](#)

`rsvg`, [2](#)

`rsvg_eps` (`rsvg`), [2](#)

`rsvg_nativeraster` (`rsvg`), [2](#)

`rsvg_pdf` (`rsvg`), [2](#)

`rsvg_png` (`rsvg`), [2](#)

`rsvg_ps` (`rsvg`), [2](#)

`rsvg_raw` (`rsvg`), [2](#)

`rsvg_svg` (`rsvg`), [2](#)

`rsvg_webp` (`rsvg`), [2](#)