Package 'rvg'

June 11, 2018

Type Package

Title R Graphics Devices for Vector Graphics Output

Version 0.1.9

Description Vector Graphics devices for 'SVG', 'DrawingML' for Microsoft PowerPoint and Excel. Functions extending package 'officer' are provided to embed 'DrawingML' graphics into 'Microsoft PowerPoint' presentations and 'Microsoft Excel' workbooks.

License GPL-3

Depends R (>= 3.0)

Imports grDevices, Rcpp (>= 0.12.12), officer (>= 0.2.0), gdtools (>= 0.1.6), xml2 (>= 1.0.0)

LinkingTo Rcpp, gdtools

Suggests htmltools, testthat, covr, grid

URL https://github.com/davidgohel/rvg

BugReports https://github.com/davidgohel/rvg/issues

RoxygenNote 6.0.1.9000

NeedsCompilation yes

Author David Gohel [aut, cre],
Bob Rudis [ctb] (the javascript code used by function set_attr),
Francois Brunetti [ctb] (clipping algorithms)

Maintainer David Gohel <david.gohel@ardata.fr>

Repository CRAN

Date/Publication 2018-06-10 22:10:00 UTC

R topics documented:

`body_add_vg` *add a plot output as vector graphics into a Word object*

Description

produces a vector graphics output from R plot instructions and add the result in an Word document object produced by `read_docx`.

Usage

```
body_add_vg(x, code, pos = "after", ...)
```

Arguments

`x`	an `rdocx` object produced by `officer::read_docx`
`code`	plot instructions
`pos`	where to add the new element relative to the cursor, one of "after", "before", "on".
`...`	arguments passed on to `dml_docx`.

Note

The function is maintained but using it should be avoided: Word text boxes, the elements used to put text in a graphic, are adding extra space on top and bottom of the shape. As there is no clear rule available to handle that, it makes impossible to compute what should be the exact position of a text. This can affect the whole rendering of the graphic.

The function should then be considered as a failed experience. An alternative is to use EMF format, this will not allow editing the graphic but the display is made as vector graphic.

Examples

```
library(officer)
x <- read_docx()
x <- body_add_vg(x, code = barplot(1:5, col = 2:6) )
print(x, target = "vg.docx")
```

dml_docx *DrawingML graphic device for Microsoft Word*

Description

Graphics devices for Microsoft Word DrawingML format.

Usage

```
dml_docx(file = "Rplots.dml", width = 6, height = 6, bg = "white",
  fonts = list(), pointsize = 12, editable = TRUE, id = 1L,
  last_rel_id = 1L, raster_prefix = "raster_", standalone = TRUE)
```

Arguments

file	DrawingML file.
height, width	Height and width in inches.
bg	Default background color for the plot (defaults to "white").
fonts	Named list of font names to be aliased with fonts installed on your system. If unspecified, the R default families sans, serif, mono and symbol are aliased to the family returned by match_family().
pointsize	default point size.
editable	should vector graphics elements (points, text, etc.) be editable.
id	specifies a unique identifier (integer) within the document that will contain the DrawingML instructions.
last_rel_id	specifies the last unique identifier (integer) within relationship file that will be used to reference embedded raster images if any.
raster_prefix	string value used as prefix for png files produced when raster objects are printed on the graphical device.
standalone	produce a standalone drawingml file? If FALSE, omits xml header and namespaces.

Note

Text rendering is not optimal, this device should not be considered as a valid R graphical device.

The DrawingML implementation for 'Microsoft Word' is different from standard DrawingML particulary with text boxes. The major point is that the exact size and position of text boxes cannot be exactly defined regarding to text widths and heights.

Autofit option has been set as a workaround, this moves text slightly on the produced graphic when edited in 'Microsoft Word' but this makes sure the text can be read.

This function is deprecated and should not be used as it will be removed in version > 0.1.9.

See Also

Devices

Examples

```
dml_docx( file = tempfile() )
plot(1:11,(-5:5)^2, type='b', main="Simple Example")
dev.off()
```

---

dml_pptx *DrawingML graphic device for Microsoft PowerPoint*

Description

Graphics devices for Microsoft PowerPoint DrawingML format.

Usage

```
dml_pptx(file = "Rplots.dml", width = 6, height = 6, offx = 1,
  offy = 1, bg = "white", fonts = list(), pointsize = 12,
  editable = TRUE, id = 1L, last_rel_id = 1L, raster_prefix = "raster_",
  standalone = TRUE)
```

Arguments

file	the file where output will appear.
height, width	Height and width in inches.
offx, offy	top and left origin of the plot
bg	Default background color for the plot (defaults to "white").
fonts	Named list of font names to be aliased with fonts installed on your system. If unspecified, the R default families sans, serif, mono and symbol are aliased to the family returned by match_family().
pointsize	default point size.
editable	should vector graphics elements (points, text, etc.) be editable.
id	specifies a unique identifier (integer) within the slide that will contain the DrawingML instructions.
last_rel_id	specifies the last unique identifier (integer) within relationship file that will be used to reference embedded raster images if any.
raster_prefix	string value used as prefix for png files produced when raster objects are printed on the graphical device.
standalone	produce a standalone drawingml file? If FALSE, omits xml header and namespaces.

See Also

Devices

Examples

```
dml_pptx( file = tempfile() )
plot(1:11,(-5:5)^2, type='b', main="Simple Example")
dev.off()
```

---

dml_xlsx *DrawingML graphic device for Microsoft Excel*

Description

Graphics devices for Microsoft Excel DrawingML format.

Usage

```
dml_xlsx(file = "Rplots.dml", width = 6, height = 6, offx = 1,
  offy = 1, bg = "white", fonts = list(), pointsize = 12,
  editable = TRUE, id = 1L, last_rel_id = 1L, raster_prefix = "raster_",
  standalone = TRUE)
```

Arguments

file	the file where output will appear.
height, width	Height and width in inches.
offx, offy	top and left origin of the plot
bg	Default background color for the plot (defaults to "white").
fonts	Named list of font names to be aliased with fonts installed on your system. If unspecified, the R default families sans, serif, mono and symbol are aliased to the family returned by match_family().
pointsize	default point size.
editable	should vector graphics elements (points, text, etc.) be editable.
id	specifies a unique identifier (integer) within the slide that will contain the DrawingML instructions.
last_rel_id	specifies the last unique identifier (integer) within relationship file that will be used to reference embedded raster images if any.
raster_prefix	string value used as prefix for png files produced when raster objects are printed on the graphical device.
standalone	produce a standalone drawingml file? If FALSE, omits xml header and namespaces.

See Also

Devices

Examples

```
dml_xlsx( file = tempfile() )
plot(1:11,(-5:5)^2, type='b', main="Simple Example")
dev.off()
```

---

dsvg *SVG Graphics Driver*

Description

This function produces SVG files (compliant to the current w3 svg XML standard) where elements can be made interactive.

Usage

```
dsvg(file = "Rplots.svg", width = 6, height = 6, bg = "white",
  pointsize = 12, standalone = TRUE, canvas_id = 1, fonts = list(),
  fontname_serif = NULL, fontname_sans = NULL, fontname_mono = NULL,
  fontname_symbol = NULL)
```

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 stand alone svg file? If FALSE, omits xml header and default namespace.
canvas_id	svg id within HTML page.
fonts	Named list of font names to be aliased with fonts installed on your system. If unspecified, the R default families sans, serif, mono and symbol are aliased to the family returned by match_family().
fontname_serif, fontname_sans, fontname_mono, fontname_symbol	font names for font faces. Used fonts should be available in the operating system. These arguments are deprecated in favor of the fonts argument.

See Also

Devices

Examples

```
dsvg()
plot(rnorm(10), main="Simple Example", xlab = "", ylab = "")
dev.off()
```

`dsvg_view` *Run plotting code and view svg in RStudio Viewer or web broswer.*

Description

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

Usage

```
dsvg_view(code, ...)
```

Arguments

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

Examples

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

`ph_with_vg` *add a plot output as vector graphics into a PowerPoint object*

Description

produces a vector graphics output from R plot instructions and add the result in a PowerPoint document object produced by read_pptx.

Usage

```
ph_with_vg(x, code, ggobj = NULL, type = "body", index = 1, ...)

ph_with_vg_at(x, code, ggobj = NULL, left, top, width, height, ...)
```

Arguments

x	an rpptx object produced by officer::read_pptx
code	plot instructions
ggobj	ggplot objet to print. argument code will be ignored if this argument is supplied.
type	placeholder type
index	placeholder index (integer). This is to be used when a placeholder type is not unique in the current slide, e.g. two placeholders with type 'body'.
...	arguments passed on to dml_pptx.
left, top	left and top origin of the plot on the slide in inches.
height, width	Height and width in inches.

Examples

```
library(officer)
doc <- read_pptx()
doc <- add_slide(doc, "Title and Content", "Office Theme")
doc <- ph_with_vg(doc, code = barplot(1:5, col = 2:6), type = "body")
doc <- add_slide(doc, "Title and Content", "Office Theme")
doc <- ph_with_vg_at(doc, code = barplot(1:5, col = 2:6),
  left = 1, top = 2, width = 6, height = 4)
print(doc, target = "vg.pptx")
```

---

rvg_tracer_off *trace off id colection*

Description

get collected id of an rvg device and stop collecting.

Usage

```
rvg_tracer_off()
```

Value

graphical elements id as integer values.

rvg_tracer_on *trace on id colection*

Description

Start collecting id of an rvg device.

Usage

```
rvg_tracer_on()
```

set_attr *set attributes to graphical elements*

Description

set attributes with javascript instructions to graphical elements.

Usage

```
set_attr(ids, attribute, str)
```

Arguments

ids	integer vector of graphical elements identifiers (returned by rvg_tracer_off).
attribute	name of the attribute to set.
str	values to set for the attribute.

xl_add_vg *add a plot output as vector graphics into an Excel object*

Description

produces a vector graphics output from R plot instructions and add the result in an Excel sheet. by read_xlsx.

Usage

```
xl_add_vg(x, sheet, code, left, top, width, height, ...)
```

Arguments

x	an rxlsx object produced by officer::read_xlsx
sheet	sheet label/name
code	plot instructions
left, top	left and top origin of the plot on the slide in inches.
height, width	Height and width in inches.
...	arguments passed on to dml_xlsx.

Examples

```
library(officer)
my_ws <- read_xlsx()
my_ws <- xl_add_vg(my_ws, sheet = "Feuil1",
  code = barplot(1:5, col = 2:6), width = 6, height = 6, left = 1, top = 2 )
print(my_ws, target = "vg.xlsx")
```

Index