Package: rvg (via r-universe)

October 8, 2026

Type Package

Title R Graphics Devices for 'Office' Vector Graphics Output

Version 0.4.2

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

License GPL-3

URL https://ardata-fr.github.io/officeverse/, https://davidgohel.github.io/rvg/

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

Depends R (>= 3.0)

Imports gdtools (>= 0.5.0), grDevices, officer (>= 0.7.4), Rcpp (>= 0.12.12), rlang, systemfonts, xml2 (>= 1.0.0)

Suggests grid, testthat

LinkingTo Rcpp, systemfonts

Encoding UTF-8

RoxygenNote 7.3.3

SystemRequirements libpng

NeedsCompilation yes

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

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

Config/pak/sysreqs libpng-dev

Repository https://cran.r-universe.dev

Date/Publication 2026-04-24 14:11:44 UTC

RemoteUrl https://github.com/cran/rvg

RemoteRef HEAD

RemoteSha e989722cf1f39f94e10c1c901fb9e1fcbc01d2f7

Contents

dml 2
dml_pptx 4
dml_xlsx 5
ph_with.dml 6
sheet_add_drawing.dml 7
xl_add_vg 8

Index **10**

`dml` *Wrap plot instructions for DrawingML plotting in Powerpoint*

Description

A simple wrapper to mark the plot instructions as Vector Graphics instructions. It produces an object of class 'dml' with a corresponding method `ph_with`.

The function enables usage of any R plot with argument `code` and with ggplot objects with argument `ggobj`.

Note: the output is a vector graphic, not a native Microsoft Office chart. The underlying data is not embedded in the document.

Usage

```
dml(
  code,
  ggobj = NULL,
  bg = "white",
  fonts = list(),
  pointsize = 12,
  editable = TRUE,
  ...
)
```

Arguments

`code`	plotting instructions
`ggobj`	ggplot object to print. argument code will be ignored if this argument is supplied.
`bg`, `fonts`, `pointsize`, `editable`	Parameters stored and later passed to `dml_pptx` by `ph_with.dml`.
`...`	unused arguments

background color

When dealing with a ggplot object, the bg parameter sets the device background but the ggplot theme's plot.background will typically draw over it. To control the background appearance, set it in the theme instead:

```
theme(
  panel.background = element_rect(fill = "#EFEFEF"),
  plot.background = element_rect(fill = "wheat"))
```

Limitations

The DrawingML format does not support the following R graphics features:

- **Patterns and gradients**: grid patterns (GridPattern), linear and radial gradients used as fill are not translated into DrawingML. ggplot2 will emit a warning when these are used.
- **Rich text (ggtext/gridtext)**: packages such as ggtext render HTML/Markdown into composite grid grobs. The rvg device can only capture low-level primitives (text strings, lines, polygons), so the rich formatting (inline images, coloured spans, HTML markup) is lost. Use standard ggplot2 text functions or apply formatting after export in PowerPoint.
- **Plotmath expressions**: R's graphics engine decomposes expression() into individual text primitives before the device sees them. Compound expressions (e.g. expression("a" * "b"), superscripts, subscripts) are rendered as separate text boxes, causing vertical misalignment.
- **Clipping paths, masks, compositing and transformations**: these R >= 4.1 features have no DrawingML equivalent and are silently ignored.

See Also

ph_with.dml

Examples

```
anyplot <- dml(code = barplot(1:5, col = 2:6), bg = "wheat")

library(officer)
doc <- read_pptx()
doc <- add_slide(doc, "Title and Content", "Office Theme")
doc <- ph_with(doc, anyplot, location = ph_location_fullsize())
fileout <- tempfile(fileext = ".pptx")
# fileout <- "vg.pptx"
print(doc, target = fileout)
```

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 families detected by gdtools::font_set_auto(). **Important:** fonts are referenced by name in the DrawingML output and are **not embedded**. The reader of the PowerPoint file must also have the same fonts installed, otherwise Office will substitute them. For maximum portability, prefer fonts bundled with Microsoft Office such as "Arial", "Times New Roman" and "Courier New". Fonts must be installed on your system; use gdtools::font_family_exists() to verify availability. Example: list(sans = "Arial", serif = "Times New Roman", mono = "Courier New").
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,
  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 families detected by gdtools::font_set_auto(). **Important:** fonts are referenced by name in the DrawingML output and are **not embedded**. The reader of the Excel file must also have the same fonts installed, otherwise Office will substitute them. For maximum portability, prefer fonts bundled with Microsoft Office such as "Arial", "Times New Roman" and "Courier New". Fonts must be installed on your system; use gdtools::font_family_exists() to verify availability. Example: list(sans = "Arial", serif = "Times New Roman", mono = "Courier New").
pointsize	default point size.
editable	should vector graphics elements (points, text, etc.) be editable.
id	specifies a unique identifier (integer) within the sheet 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.
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()
```

---

ph_with.dml *add a plot output as vector graphics into a PowerPoint object*

Description

Produces a vector graphics output from R plot instructions stored in a dml object and adds the result in an rpptx object produced by read_pptx.

Usage

```
## S3 method for class 'dml'
ph_with(x, value, location, ...)
```

Arguments

x	a pptx device
value	dml object
location	a location for a placeholder.
...	Arguments to be passed to methods

Examples

```
anyplot <- dml(code = barplot(1:5, col = 2:6), bg = "wheat")

library(officer)
doc <- read_pptx()
doc <- add_slide(doc, "Title and Content", "Office Theme")
doc <- ph_with(doc, anyplot, location = ph_location_fullsize())

fileout <- tempfile(fileext = ".pptx")
print(doc, target = fileout)
```

---

sheet_add_drawing.dml *Add a dml drawing to an Excel sheet*

Description

Add a dml() object to a sheet in an xlsx workbook created with officer::read_xlsx(). The plot is rendered as editable vector graphics (DrawingML).

Usage

```
## S3 method for class 'dml'
sheet_add_drawing(
  x,
  value,
  sheet,
  left = 1,
  top = 1,
  width = 6,
  height = 6,
  ...
)
```

Arguments

x	an rxlsx object (created by officer::read_xlsx())
value	a dml() object
sheet	sheet name (must already exist)
left, top	left and top origin of the plot on the sheet in inches
width, height	width and height in inches
...	additional arguments passed to dml_xlsx()

Value

the rxlsx object (invisibly)

Examples

```
library(officer)
library(rvg)

x <- read_xlsx()
x <- sheet_add_drawing(x,
  value = dml(code = barplot(1:5, col = 2:6)),
  sheet = "Feuil1",
  left = 1, top = 2, width = 6, height = 6
)
print(x, target = tempfile(fileext = ".xlsx"))
```

---

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

Description

Deprecated, use officer::sheet_add_drawing() with a dml() object instead.

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 sheet in inches.
height, width	Height and width in inches.
...	arguments passed on to dml_xlsx.

Examples

```
library(officer)
my_ws <- read_xlsx()
my_ws <- sheet_add_drawing(my_ws,
  value = dml(code = barplot(1:5, col = 2:6)),
  sheet = "Feuil1",
  width = 6, height = 6, left = 1, top = 2
)
fileout <- tempfile(fileext = ".xlsx")
print(my_ws, target = fileout)
```

Index

∗ **device**
dml_pptx, 4
dml_xlsx, 5

Devices, *5, 6*
dml, 2, *6, 7*
dml(), *7, 8*
dml_pptx, *2*, 4
dml_xlsx, 5, *8*
dml_xlsx(), *8*

gdtools::font_set_auto(), *4*, *6*

officer::read_xlsx(), *7, 8*
officer::sheet_add_drawing(), *8*

ph_with, *2*
ph_with.dml, *2, 3*, 6

read_pptx, *6*

sheet_add_drawing.dml, 7

xl_add_vg, 8