Package: ggbond (via r-universe)

June 25, 2026

Title Interactive Layout Editor for 'R' Plots

Version 0.0.5

Description Provides a 'shiny'-based layout editor for arranging 'R' plot objects on a fixed-size canvas. It supports 'ggplot2' plots, 'graphics' functions and recorded plots, 'pheatmap' objects, 'ComplexHeatmap' objects, 'grid' grobs, 'gtable' objects, and local raster images, with live preview and PDF or PNG export.

License GPL (>= 3)

Encoding UTF-8

RoxygenNote 7.3.3

Imports ggplot2, graphics, grDevices, grid, gridGraphics, jpeg, jsonlite, png, shiny, stats, tiff, tools, utils

Suggests ComplexHeatmap, pheatmap

NeedsCompilation no

Author BeiHao Li [aut, cre, cph]

Maintainer BeiHao Li <szright2000@gmail.com>

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

Date/Publication 2026-06-25 11:50:07 UTC

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

RemoteRef HEAD

RemoteSha 5e7f7ae6f2deab1908790770632603db09bd7a15

Contents

`ggbond_demo_plots` *Create demo plots for ggbond*

Description

This function creates a small set of example plot objects used by the ggbond demo Shiny app, including ggplot2 and optional non-ggplot examples.

Usage

```
ggbond_demo_plots()
```

Value

A named list of plot objects.

Examples

```
plots <- ggbond_demo_plots()
names(plots)
length(plots)
```

`print.ggbond` *Print a ggbond layout object*

Description

Print a ggbond layout object

Usage

```
## S3 method for class 'ggbond'
print(x, ...)
```

Arguments

`x`	A ggbond object.
`...`	Unused.

Value

Invisibly returns `x`.

Examples

```
layout <- structure(
  list(
    layout = data.frame(id = "panel_1", label = "A"),
    canvas = list(width_px = 700, height_px = 500),
    device = list(width_in = 7, height_in = 5),
    image_assets = data.frame(),
    exit_reason = "example",
    created_at = Sys.time()
  ),
  class = "ggbond"
)
print(layout)
```

---

read_ggbond_json *Read a ggbond layout object from JSON*

Description

Read a ggbond layout object from JSON

Usage

```
read_ggbond_json(file)
```

Arguments

file	Input JSON file path produced by save_ggbond_json().

Value

A ggbond object.

Examples

```
layout <- structure(
  list(
    layout = data.frame(
      id = "panel_1",
      label = "A",
      x = 0,
      y = 0,
      width = 700,
      height = 500,
      plot = "scatter",
      source = "plot:scatter",
      lock_aspect = FALSE,
      show_border = FALSE,
```

```
      z = 1
    ),
    canvas = list(width_px = 700, height_px = 500),
    device = list(width_in = 7, height_in = 5),
    image_assets = data.frame(),
    exit_reason = "example",
    created_at = Sys.time()
  ),
  class = "ggbond"
)
file <- tempfile(fileext = ".json")
save_ggbond_json(layout, file)
read_ggbond_json(file)
```

`render_ggbond` *Render a ggbond layout object*

Description

Re-renders a layout returned by `run_ggbond()` using a supplied plot list.

Usage

```
render_ggbond(
  x,
  plot_list,
  file = NULL,
  device = NULL,
  width = x$device$width_in,
  height = x$device$height_in,
  res = 300
)
```

Arguments

`x`	A ggbond object returned by `run_ggbond()`.
`plot_list`	A named list of plot objects matching the plot sources used in the layout.
`file`	Optional output path. When `NULL`, rendering is drawn to the current graphics device.
`device`	Output device when `file` is supplied. Supported values are `"pdf"` and `"png"`. When `NULL`, the device is inferred from `file`.
`width`	Device width in inches. Defaults to the width stored in `x`.
`height`	Device height in inches. Defaults to the height stored in `x`.
`res`	PNG resolution in dots per inch.

Value

Invisibly returns `x`.

Examples

```
plots <- list(
  scatter = ggplot2::ggplot(mtcars, ggplot2::aes(wt, mpg)) +
    ggplot2::geom_point()
)
layout <- structure(
  list(
    layout = data.frame(
      id = "panel_1",
      label = "A",
      x = 0,
      y = 0,
      width = 700,
      height = 500,
      plot = "scatter",
      source = "plot:scatter",
      lock_aspect = FALSE,
      show_border = FALSE,
      z = 1
    ),
    canvas = list(width_px = 700, height_px = 500),
    device = list(width_in = 7, height_in = 5),
    image_assets = data.frame(),
    exit_reason = "example",
    created_at = Sys.time()
  ),
  class = "ggbond"
)
file <- tempfile(fileext = ".png")
render_ggbond(layout, plots, file = file, res = 72)
file.exists(file)
```

---

run_ggbond *Run the ggbond Shiny app*

Description

Run the ggbond Shiny app

Usage

```
run_ggbond(
  plot_list = NULL,
  canvas_width_px = 700,
```

```
  canvas_height_px = 500,
  device_width_in = 7,
  device_height_in = 5,
  launch.browser = TRUE
)
```

Arguments

plot_list	A named list of plot objects. If NULL, demo plots are used. Supported objects include ggplot2 plots, base graphics functions or recorded plots, pheatmap objects, ComplexHeatmap objects, grobs, gtables, and local image panels uploaded in the app.
canvas_width_px	Canvas width in pixels.
canvas_height_px	Canvas height in pixels.
device_width_in	Graphics device width in inches.
device_height_in	Graphics device height in inches.
launch.browser	Passed to shiny::runApp().

Details

Canvas and device sizes are linked at 100 pixels per inch. If only one size pair is supplied, the other pair is derived automatically.

Value

A ggbond layout object containing panel positions, canvas metadata, graphics device metadata, uploaded image metadata, and the app exit reason.

Examples

```
if (interactive()) {
  plots <- list(
    scatter = ggplot2::ggplot(mtcars, ggplot2::aes(wt, mpg)) +
      ggplot2::geom_point()
  )
  layout <- run_ggbond(plots)
  layout
}
```

save_ggbond_json *Save a ggbond layout object to JSON*

Description

Save a ggbond layout object to JSON

Usage

```
save_ggbond_json(x, file, pretty = TRUE)
```

Arguments

x	A ggbond object returned by run_ggbond().
file	Output JSON file path.
pretty	Whether to write pretty-formatted JSON.

Value

Invisibly returns file.

Examples

```
layout <- structure(
  list(
    layout = data.frame(
      id = "panel_1",
      label = "A",
      x = 0,
      y = 0,
      width = 700,
      height = 500,
      plot = "scatter",
      source = "plot:scatter",
      lock_aspect = FALSE,
      show_border = FALSE,
      z = 1
    ),
    canvas = list(width_px = 700, height_px = 500),
    device = list(width_in = 7, height_in = 5),
    image_assets = data.frame(),
    exit_reason = "example",
    created_at = Sys.time()
  ),
  class = "ggbond"
)
file <- tempfile(fileext = ".json")
save_ggbond_json(layout, file)
restored <- read_ggbond_json(file)
```

```
restored
```

Index