

Package: rustgd (via r-universe)

June 13, 2026

Title Rust-Backed Graphics Device and Viewer Windows for Terminal R Sessions

Version 0.0.0.9000

Description Provides a Rust-backed graphics device and companion viewer windows for plots, HTML widgets and live URLs, and data frames, so that terminal and editor R sessions without an IDE viewer pane (such as radian, Zed, or Neovim) can display graphics, htmlwidgets, Shiny apps, and data frames in native windows.

License MIT + file LICENSE

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.3

Config/rextendr/version 0.5.0

SystemRequirements Cargo (Rust's package manager), rustc

Depends R (>= 4.2)

Imports arrow, grDevices, later, utils

Suggests shiny

Config/pak/sysreqs cmake libssl-dev libclang-dev

Repository <https://alcocer-jj.r-universe.dev>

Date/Publication 2026-06-13 09:39:15 UTC

RemoteUrl <https://github.com/alcocer-jj/rustgd>

RemoteRef HEAD

RemoteSha a72bded6724ed58b041e18e77a4f84337b6c4283

Contents

<code>.rustgd_frames_bin</code>	2
<code>.rustgd_frames_dir</code>	2
<code>.rustgd_frames_register_cleanup</code>	3

<code>.rustgd_record_current</code>	3
<code>.rustgd_register_cleanup</code>	3
<code>.rustgd_widgets_dir</code>	4
<code>rustgd</code>	4
<code>rustgd_browse</code>	5
<code>rustgd_disable</code>	5
<code>rustgd_enable</code>	6
<code>rustgd_handle_clear_all</code>	6
<code>rustgd_handle_clear_plot</code>	7
<code>rustgd_input_handler</code>	7
<code>rustgd_poll_resize</code>	8
<code>rustgd_webview_bin</code>	8
<code>unuse_rustgd</code>	8
<code>unuse_rustgd_webview</code>	9
<code>use_rustgd</code>	9
<code>use_rustgd_webview</code>	10
<code>view</code>	11

Index**12**

<code>.rustgd_frames_bin</code>	<i>Internal: absolute path to the staged rustgd-frames binary.</i>
---------------------------------	--

Description

Internal: absolute path to the staged rustgd-frames binary.

Usage

```
.rustgd_frames_bin()
```

<code>.rustgd_frames_dir</code>	<i>Internal: per-process frames directory, a sibling of the plots and widgets directories. Uses the session temp directory so it works on every platform (Windows has no /tmp); the binary receives this path as an argument.</i>
---------------------------------	---

Description

Internal: per-process frames directory, a sibling of the plots and widgets directories. Uses the session temp directory so it works on every platform (Windows has no /tmp); the binary receives this path as an argument.

Usage

```
.rustgd_frames_dir()
```

.rustgd_frames_register_cleanup

Internal: remove the frames directory when R exits.

Description

Internal: remove the frames directory when R exits.

Usage

.rustgd_frames_register_cleanup()

.rustgd_record_current

Internal: snapshot the current device's graphics display list, provided rustgd is the active device. Returns NULL silently in any situation where recordPlot is not safe or not meaningful (e.g. the user switched devices mid-expression, or there's nothing to record). Recording is cheap compared to the actual drawing that just happened, so calling it on every drew/new_page flag fire (not just new_page) is fine.

Description

Internal: snapshot the current device's graphics display list, provided rustgd is the active device. Returns NULL silently in any situation where recordPlot is not safe or not meaningful (e.g. the user switched devices mid-expression, or there's nothing to record). Recording is cheap compared to the actual drawing that just happened, so calling it on every drew/new_page flag fire (not just new_page) is fine.

Usage

.rustgd_record_current()

.rustgd_register_cleanup

Internal: register a one-time finalizer that removes the widget directory when R exits, so nothing is left in the temp folder. The webview binary, if still open, sees its directory vanish on its next poll and exits.

Description

Internal: register a one-time finalizer that removes the widget directory when R exits, so nothing is left in the temp folder. The webview binary, if still open, sees its directory vanish on its next poll and exits.

Usage

```
.rustgd_register_cleanup()
```

```
.rustgd_widgets_dir
```

Internal: per-process widget directory. Separate from the device's rustgd- directory so the two lifecycles never interfere. Uses the session temp directory so it works on every platform; the binary receives this path as an argument.

Description

Internal: per-process widget directory. Separate from the device's rustgd- directory so the two lifecycles never interfere. Uses the session temp directory so it works on every platform; the binary receives this path as an argument.

Usage

```
.rustgd_widgets_dir()
```

```
rustgd
```

Start the rustgd graphics device.

Description

Launches a native viewer window. When the viewer is resized, all plots in the session history reflow automatically during R's idle time, with no need to press Enter or run any command. Each plot's recorded graphics display list (captured via `grDevices::recordPlot` immediately after the plot was drawn) is replayed onto the resized device via `grDevices::replayPlot`, which lets the underlying plotting package (`ggplot2`, `tmap`, base R, `lattice`, etc.) lay itself out fresh at the new dimensions without re-executing any of the user's R code. If recording or replay fails for any reason, the captured top-level expression is re-evaluated as a fallback.

Usage

```
rustgd()
```

Details

Closing the viewer window propagates back to R: the next idle-time poll detects the close, calls `dev.off()` on the rustgd device, and the Rust-side `close()` callback removes the session directory. After closing, calling `rustgd()` again opens a fresh window with no stray state.

rustgd_browse	<i>Show a live URL in the rustgd web window.</i>
---------------	--

Description

Points the rustgd webview window at a running address: a Shiny app, a local dashboard, a development server, or any reachable http(s) page. Unlike an `htmlwidget`, nothing is copied; the window loads the URL directly, so the entry is only meaningful while that server is up.

Usage

```
rustgd_browse(url, title = NULL)
```

Arguments

<code>url</code>	A single <code>http://</code> or <code>https://</code> address.
<code>title</code>	Optional label stored with the entry.

Details

Because a stopped server leaves a dead address behind, pushing any URL first clears any previous URL entry from the window, so there is at most one live URL at a time. Your `htmlwidget` entries are left untouched. The last URL stays until you close it with the window's "Clear widget" button.

Value

The URL, invisibly.

rustgd_disable	<i>Deactivate the rustgd suite in the current session only.</i>
----------------	---

Description

Inverse of `rustgd_enable()`. Restores the previous graphics device, web viewer, and Shiny launcher, and removes the `View()` route, without touching `.Rprofile`. Open rustgd windows are left alone.

Usage

```
rustgd_disable()
```

rustgd_enable	<i>Activate the rustgd suite in the current session only.</i>
---------------	---

Description

Routes plots to the rustgd graphics device, HTML widgets and Shiny apps to the rustgd web viewer, and View() to the rustgd data frame window, for this session only. Does not modify .Rprofile. Used internally by [use_rustgd\(\)](#) and by the .Rprofile snippet it writes; also usable directly for a one-off session.

Usage

```
rustgd_enable(mode = c("lazy", "eager"))
```

Arguments

mode	Either "lazy" or "eager"; affects the plot device only. "eager" opens a plot window immediately.
------	--

See Also

[rustgd_disable\(\)](#), [use_rustgd\(\)](#).

rustgd_handle_clear_all	
-------------------------	--

Internal: process a clear-all signal from the viewer. Empties the plot history, deletes every plot-NNNN.svg in the session dir, and resets the page counter so the next user-issued plot() starts fresh at plot-0001.svg.

Description

Internal: process a clear-all signal from the viewer. Empties the plot history, deletes every plot-NNNN.svg in the session dir, and resets the page counter so the next user-issued plot() starts fresh at plot-0001.svg.

Usage

```
rustgd_handle_clear_all()
```

`rustgd_handle_clear_plot`

Internal: process a clear-active-plot signal from the viewer. The marker file contains the 1-based index of the plot to remove. The entry is dropped from plot_history, the corresponding SVG file is deleted, and any higher-numbered files are renamed down by one so the on-disk numbering stays contiguous and matches the new history indices.

Description

Internal: process a clear-active-plot signal from the viewer. The marker file contains the 1-based index of the plot to remove. The entry is dropped from plot_history, the corresponding SVG file is deleted, and any higher-numbered files are renamed down by one so the on-disk numbering stays contiguous and matches the new history indices.

Usage

```
rustgd_handle_clear_plot()
```

`rustgd_input_handler` *Internal: process a pending resize if one exists. Reads resize.txt, updates the device dimensions, and re-evaluates every plot in the session history at the new size. Each plot is directed to its own plot-NNNN.svg file via rustgd_set_current_page().*

Description

Internal: process a pending resize if one exists. Reads resize.txt, updates the device dimensions, and re-evaluates every plot in the session history at the new size. Each plot is directed to its own plot-NNNN.svg file via rustgd_set_current_page().

Usage

```
rustgd_input_handler()
```

`rustgd_poll_resize` *Internal: scheduled poll for resize signals and viewer-close events. Reschedules itself while the device is active.*

Description

Internal: scheduled poll for resize signals and viewer-close events. Reschedules itself while the device is active.

Usage

```
rustgd_poll_resize()
```

`rustgd_webview_bin` *Internal: absolute path to the staged rustgd-webview binary.*

Description

Internal: absolute path to the staged rustgd-webview binary.

Usage

```
rustgd_webview_bin()
```

`unuse_rustgd` *Stop using rustgd for plots, web content, and data frames.*

Description

Inverse of `use_rustgd()`. Restores the previous graphics device, web viewer, and Shiny launcher in the current session, removes the rustgd `View()` route, and strips the auto-activation snippet from `.Rprofile`. Any rustgd windows already open are left alone. Safe to call when nothing is active.

Usage

```
unuse_rustgd(rprofile_path = path.expand("~/Rprofile"))
```

Arguments

`rprofile_path` Path to the `.Rprofile` to modify. Defaults to `~/Rprofile`.

unuse_rustgd_webview *Stop using the rustgd web viewer.*

Description

Inverse of `use_rustgd_webview()`. Restores whatever options("viewer") and options("shiny.launch.browser") were set before, so an IDE viewer pane (RStudio, Positron) and the default Shiny launcher take over again. Any open rustgd webview window is left alone.

Usage

```
unuse_rustgd_webview()
```

use_rustgd *Use rustgd for plots, web content, and data frames.*

Description

Activates the full rustgd suite in one call: the graphics device for plots, the web viewer for HTML widgets and Shiny apps (and explicit URLs via `rustgd_browse()`), and the data frame viewer behind `View()`. The change takes effect immediately in the current session, and a small snippet is also written to your user-level `.Rprofile` so the same setup is restored automatically in future sessions.

Usage

```
use_rustgd(
  mode = c("lazy", "eager"),
  rprofile_path = path.expand("~/Rprofile")
)
```

Arguments

mode	Either "lazy" or "eager"; see description. Affects the plot device only.
rprofile_path	Path to the <code>.Rprofile</code> to modify. Defaults to <code>~/Rprofile</code> . Pass an explicit path to target a project-local profile instead.

Details

Two startup modes control only the plot device:

"lazy" (the default) registers rustgd as the default device via `options(device = ...)`. The viewer window opens when you draw your first plot and not before, matching base R's `quartz` and `X11`.

"eager" additionally opens a plot window straight away, so it is ready before you plot anything.

The web viewer and the `View()` route are turned on immediately in both modes. The `.Rprofile` snippet is guarded by `interactive()` so non-interactive contexts (Rscript, R CMD BATCH, knitr, testthat, package checks) are unaffected, and by `requireNamespace()` so uninstalling `rustgd` will not break R startup.

Safe to call repeatedly. Re-running with a different mode replaces the existing snippet in place.

See Also

[unuse_rustgd\(\)](#) to undo this, and [rustgd_enable\(\)](#) for a one-session activation that does not touch `.Rprofile`.

use_rustgd_webview *Use the rustgd web viewer for HTML content.*

Description

Registers the `rustgd` webview window as R's HTML viewer by setting `options(viewer = ...)`, and also routes Shiny apps to the same window by setting `options(shiny.launch.browser = ...)`. After this, printing an `htmlwidget` (plotly, leaflet, a `tmap` in view mode, and so on) at the console, or running a Shiny app with `shiny::runApp()`, opens in a `rustgd` webview window instead of a browser tab. This is meant for terminal and editor R sessions (radian, Zed, Neovim) that otherwise have no viewer pane.

Usage

```
use_rustgd_webview()
```

Details

The window launches lazily: it opens the first time you display something, not when this function is called. Later content reuses the same window. Closing the window is fine; the next item reopens one.

Widgets are copied into a per-session directory under the system temp folder, so what the window shows does not depend on R's own temp files. Shiny apps and other live URLs are shown in place by pointing the window at the running address; see [rustgd_browse\(\)](#). For a Shiny app, stopping it (Ctrl+C, which is also what returns your terminal) removes its entry from the window automatically, so no separate clear step is needed. Closing the window deletes the copies, and quitting R removes the whole directory, so nothing is left behind.

The previous values of `options("viewer")` and `options("shiny.launch.browser")` are remembered and restored by [unuse_rustgd_webview\(\)](#).

view	<i>View a data frame in a rustgd frames window.</i>
------	---

Description

Writes the data frame to the frames channel as Arrow IPC and opens, or reuses, the rustgd frames window to display it. Non-blocking: it returns at once and the window appears on its own, the same way the plot and web viewers behave. Each call adds a frame; later stages page through them as a gallery.

Usage

```
view(df, title = NULL)
```

Arguments

df	A data frame, or an object coercible to one.
title	Optional label shown in the window; defaults to the expression passed in, for example <code>view(mtcars)</code> is labeled <code>mtcars</code> .

Details

The frame is written uncompressed so the viewer needs no compression codec. Requires the `arrow` package for `write_feather()`.

Value

The path to the written Arrow file, invisibly.

Index

`.rustgd_frames_bin`, 2
`.rustgd_frames_dir`, 2
`.rustgd_frames_register_cleanup`, 3
`.rustgd_record_current`, 3
`.rustgd_register_cleanup`, 3
`.rustgd_widgets_dir`, 4

`rustgd`, 4
`rustgd_browse`, 5
`rustgd_browse()`, 9, 10
`rustgd_disable`, 5
`rustgd_disable()`, 6
`rustgd_enable`, 6
`rustgd_enable()`, 5, 10
`rustgd_handle_clear_all`, 6
`rustgd_handle_clear_plot`, 7
`rustgd_input_handler`, 7
`rustgd_poll_resize`, 8
`rustgd_webview_bin`, 8

`shiny::runApp()`, 10

`unuse_rustgd`, 8
`unuse_rustgd()`, 10
`unuse_rustgd_webview`, 9
`unuse_rustgd_webview()`, 10
`use_rustgd`, 9
`use_rustgd()`, 6, 8
`use_rustgd_webview`, 10
`use_rustgd_webview()`, 9

`view`, 11