Package: ggCheysson (via r-universe)

May 14, 2026

Title Graphic Styles of Emile Cheysson for 'ggplot2'

Version 1.0.0

Date 2026-01-08

Description Implements for 'ggplot2' the stylistic elements (fonts, hatched patterns, color palettes) used by 'Emile Cheysson' in the 'Albums de Statistique Graphique', sometimes called the pinnacle of the Golden Age of Statistical Graphics.

Imports ggplot2

Suggests ggpattern, ggthemes, gridpattern, systemfonts, showtext, sysfonts, ragg, knitr, rmarkdown, Guerry, sf, tidyr, dplyr

License GPL (>= 3)

Language en-US

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.3

URL https://github.com/friendly/ggCheysson, https://friendly.github.io/ggCheysson/

BugReports https://github.com/friendly/ggCheysson/issues

Depends R (>= 3.5)

LazyData true

VignetteBuilder knitr

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

Date/Publication 2026-01-14 03:13:00 UTC

RemoteUrl https://github.com/friendly/ggCheysson

RemoteRef HEAD

RemoteSha 124f5b6922dc1b3ec1bf30c92f56a339808a0b87

Contents

`albumImages` *Album Images Metadata*

Description

Metadata linking the 25 color palettes from the Albums de Statistique Graphique to their corresponding album years, plate numbers, and David Rumsey collection reference numbers.

Usage

```
albumImages
```

Format

A data frame with 25 rows and 6 variables:

adventDay Advent calendar day number (1-25) from original digitization

RumseyListNo David Rumsey Map Collection list number

Album Year of the album (1880-1906)

Qty Plate number within the album

Type Palette type: "Sequential", "Diverging", "Grouped", or "Category"

link URL to the digitized image in the David Rumsey Map Collection

Details

This dataset provides the mapping between the original SVG pattern files (named by Advent calendar days by RJ Andrews) and the actual album years and plate numbers from the *Albums de Statistique Graphique*.

The naming convention "adventDay" comes from RJ Andrews' original digitization project where he released one palette per day during December as an Advent calendar. The package uses the Album year and plate number for more intuitive palette naming (e.g., "1880_07" instead of "dec06").

Source

- David Rumsey Map Collection: https://www.davidrumsey.com/
- RJ Andrews Album Colors: https://github.com/infowetrust/albumcolors
- Tom Shanley Observable: https://observablehq.com/@tomshanley/cheysson-color-palettes

See Also

cheysson_palettes, cheysson_patterns

Examples

```
# View the dataset
head(albumImages)

# Find information about a specific album
subset(albumImages, Album == 1880)

# Count palettes by type
table(albumImages$Type)

# Get the Rumsey link for a specific palette
albumImages[albumImages$Album == 1881 & albumImages$Qty == 3, "link"]
```

---

cheysson_font *Get Cheysson font family name*

Description

Returns the appropriate Cheysson font family name for a given purpose.

Usage

```
cheysson_font(type = c("regular", "italic", "sans", "outline", "title"))
```

Arguments

type	Font type: "regular", "italic", "sans", "outline", or "title"

Value

Character string with font family name

Examples

```
cheysson_font("title")  # Returns "CheyssonTitle"
cheysson_font("regular")  # Returns "Cheysson"
```

`cheysson_fonts` *Cheysson Font Families*

Description

Metadata about the five Cheysson font families included in the package, describing their characteristics and recommended uses.

Usage

```
cheysson_fonts
```

Format

A data frame with 5 rows and 4 variables:

family Font family name (e.g., "Cheysson", "CheyssonTitle")

description Brief description of the font style

use Recommended use cases for the font

file TrueType font filename in the inst/fonts/ directory

Details

The package includes five hand-drawn font families created by Kenneth Fields (ESRI) to match the lettering style of the original Albums de Statistique Graphique:

- **Cheysson**: Regular serif font suitable for body text, axis labels, and legends
- **CheyssonItalic**: Italic variant for emphasis and annotations
- **CheyssonSansCaps**: Sans-serif capitals for axis labels and category names
- **CheyssonOutlineCaps**: Outlined capitals for decorative plot titles and headings
- **CheyssonTitle**: Decorative font for main plot titles

These fonts must be loaded before use with `load_cheysson_fonts`. The Cheysson themes (`theme_cheysson`, etc.) automatically select appropriate fonts for different plot elements.

Source

Font families created by Kenneth Fields (ESRI) based on the lettering style of Émile Cheysson's Albums de Statistique Graphique.

See Also

load_cheysson_fonts, cheysson_fonts_available, theme_cheysson

Examples

```
# View font metadata
cheysson_fonts

# Get recommended uses
cheysson_fonts[, c("family", "use")]

# Find the title font
subset(cheysson_fonts, grepl("title", use, ignore.case = TRUE))
```

---

cheysson_fonts_available

Check if Cheysson fonts are loaded

Description

Checks whether Cheysson fonts have been registered and are available for use.

Usage

```
cheysson_fonts_available(method = NULL)
```

Arguments

method	Check for "systemfonts" or "showtext". If NULL (default), checks both methods.

Value

Logical indicating whether fonts are available

Examples

```
if (cheysson_fonts_available()) {
  message("Cheysson fonts are ready to use!")
}
```

`cheysson_pal` *Get a Cheysson color palette*

Description

Returns colors from a specified Cheysson palette. Palettes can be referenced by name (e.g., "1880_07") or by selecting a palette of a particular type.

Usage

```
cheysson_pal(palette = "1880_07", n = NULL, type = 1)
```

Arguments

`palette`	Name of palette (e.g., "1880_07"), or palette type ("sequential", "diverging", "grouped", "category"). If a type is specified, the first palette of that type is returned.
`n`	Number of colors to return. If NULL, returns all colors in the palette. If n is greater than the number of colors in the palette, colors will be interpolated.
`type`	If palette is a type name, optionally specify which palette of that type to use (default is 1 for the first).

Value

A character vector of hex color codes

Examples

```
# Get all colors from a specific palette
cheysson_pal("1880_07")

# Get 5 colors from a palette
cheysson_pal("1880_07", n = 5)

# Get colors from first sequential palette
cheysson_pal("sequential")

# Get colors from second category palette
cheysson_pal("category", type = 2)
```

`cheysson_palettes` *Cheysson Color Palettes*

Description

Color palettes extracted from the Albums de Statistique Graphique produced under the direction of Émile Cheysson. These palettes are organized by album year and plate number.

Usage

```
cheysson_palettes
```

Format

A list of 20 color palettes, each containing:

colors Character vector of hex color codes

type Palette type: "sequential", "diverging", "grouped", or "category"

album Year of the album

plate Plate number within the album

rumsey_no David Rumsey collection reference number

dec_day Advent calendar day from original source

Details

The palettes are named using the convention `YYYY_PP` where YYYY is the album year and PP is the zero-padded plate number. For example, "1880_07" refers to plate 7 from the 1880 album.

Palette types:

- **Sequential** (7 palettes): Ordered colors for quantitative data
- **Diverging** (2 palettes): Two contrasting colors with neutral midpoint
- **Grouped** (5 palettes): Related colors for comparing groups
- **Category** (6 palettes): Distinct colors for categorical data

Source

Color patterns digitized by RJ Andrews from the David Rumsey Map Collection https://github.com/infowetrust/albumcolors

Observable implementation by Tom Shanley https://observablehq.com/@tomshanley/cheysson-color-palettes

See Also

`cheysson_pal`, `scale_color_cheysson`

Examples

```
# List available palettes
names(cheysson_palettes)

# Get colors from a specific palette
cheysson_palettes$`1880_07`$colors

# Find palettes by type
sequential_pals <- Filter(function(x) x$type == "sequential", cheysson_palettes)
names(sequential_pals)
```

`cheysson_pattern` *Get Cheysson pattern specifications*

Description

Returns pattern specifications from a Cheysson palette for use with ggpattern.

Usage

```
cheysson_pattern(palette = "1881_03", n = NULL, type = 1)
```

Arguments

`palette`	Name of palette (e.g., "1881_03") or palette type ("sequential", "diverging", "grouped", "category").
`n`	Number of patterns to return. If NULL, returns all patterns.
`type`	If palette is a type name, which palette of that type to use (default 1).

Value

A list of pattern specifications suitable for ggpattern

Examples

```
# Get all patterns from a palette
cheysson_pattern("1881_03")

# Get first 3 patterns
cheysson_pattern("1881_03", n = 3)

# Get patterns from a sequential palette
cheysson_pattern("sequential")
```

cheysson_pattern_params *Create ggpattern-compatible pattern parameters*

Description

Converts Cheysson pattern specifications to parameters suitable for ggpattern geoms.

Usage

```
cheysson_pattern_params(patterns, param = "fill")
```

Arguments

patterns	List of pattern specifications from cheysson_pattern()
param	Which parameter to extract: "type", "fill", "pattern_fill", "pattern_angle", "pattern_density", "pattern_spacing", or "pattern_type"

Value

Vector of parameter values

Examples

```
patterns <- cheysson_pattern("1881_03")
cheysson_pattern_params(patterns, "fill")
cheysson_pattern_params(patterns, "pattern_angle")
```

cheysson_patterns *Cheysson Pattern Data*

Description

Pattern specifications (fills and hatching) extracted from the Albums de Statistique Graphique. These patterns combine solid colors with line hatching (stripes and crosshatching) as used in Cheysson's maps.

Usage

```
cheysson_patterns
```

Format

A list of 20 pattern palettes, each containing:

patterns List of pattern specifications with fill colors and hatching parameters

type Palette type: "sequential", "diverging", "grouped", or "category"

album Year of the album

plate Plate number within the album

rumsey_no David Rumsey collection reference number

dec_day Advent calendar day from original source

n_patterns Number of patterns in the palette

Details

Each pattern specification includes:

- **type**: "solid", "stripe", or "crosshatch"
- **fill**: Base fill color
- **pattern_fill**: Color for pattern lines
- **pattern_angle**: Angle of stripes (in degrees)
- **pattern_density**: Density of pattern lines (0-1)
- **pattern_spacing**: Spacing between pattern lines
- **pattern_linewidth**: Width of pattern lines

Source

Pattern specifications digitized from the David Rumsey Map Collection

See Also

`cheysson_pattern`, `scale_pattern_fill_cheysson`

Examples

```
# List available pattern palettes
names(cheysson_patterns)

# Get patterns from a specific palette
cheysson_patterns$`1881_03`
```

list_cheysson_pals *List available Cheysson palettes*

Description

Returns information about available Cheysson color palettes, optionally filtered by type.

Usage

```
list_cheysson_pals(type = NULL)
```

Arguments

type	Optional palette type to filter by: "sequential", "diverging", "grouped", or "category". If NULL (default), returns all palettes.

Value

A data frame with columns: name, type, album, plate, n_colors

Examples

```
# List all palettes
list_cheysson_pals()

# List only sequential palettes
list_cheysson_pals("sequential")

# List only category palettes
list_cheysson_pals("category")
```

list_cheysson_patterns *List available Cheysson pattern palettes*

Description

Returns information about available Cheysson pattern palettes.

Usage

```
list_cheysson_patterns(type = NULL)
```

Arguments

type	Optional palette type to filter by: "sequential", "diverging", "grouped", or "category". If NULL (default), returns all palettes.

Value

A data frame with columns: name, type, album, plate, n_patterns

Examples

```
# List all pattern palettes
list_cheysson_patterns()

# List only sequential palettes
list_cheysson_patterns("sequential")
```

---

load_cheysson_fonts *Load Cheysson fonts*

Description

Registers the Cheysson font families for use in plots. This function should be called before using Cheysson fonts in ggplot2.

Usage

```
load_cheysson_fonts(method = c("systemfonts", "showtext"))
```

Arguments

method	Font loading method: "systemfonts" (default) or "showtext". systemfonts is recommended for most uses. showtext is useful for saving plots to files.

Details

The package includes five Cheysson font families:

- **Cheysson**: Regular serif font for body text
- **CheyssonItalic**: Italic variant
- **CheyssonSansCaps**: Sans-serif capitals
- **CheyssonOutlineCaps**: Outlined capitals for titles
- **CheyssonTitle**: Decorative font for titles

When using showtext, you must call `showtext::showtext_auto()` before creating plots, and `showtext::showtext_auto(FALSE)` when done.

Windows users: The systemfonts method works for saved plots (with ragg) but custom fonts won't appear in the on-screen plot window. For on-screen preview with fonts:

- Use `method = "showtext"` instead, or
- In RStudio: Tools > Global Options > General > Graphics > Backend: "AGG"

For saving plots with systemfonts, use `ggsave(..., device = ragg::agg_png)`.

Value

Invisibly returns a character vector of loaded font family names

Examples

```
# Load fonts (default method)
load_cheysson_fonts()

# Use in a plot
library(ggplot2)
p <- ggplot(mtcars, aes(wt, mpg)) +
  geom_point() +
  labs(title = "Using Cheysson Fonts") +
  theme(
    text = element_text(family = "Cheysson"),
    plot.title = element_text(family = "CheyssonTitle")
  )

# Save to temporary file
tmp <- tempfile(fileext = ".png")
if (requireNamespace("ragg", quietly = TRUE)) {
  ggsave(tmp, p, device = ragg::agg_png)
} else {
  ggsave(tmp, p)
}
unlink(tmp)
```

---

scale_cheysson *Cheysson color scales for ggplot2*

Description

Color and fill scales using Cheysson palettes from the Albums de Statistique Graphique.

Usage

```
scale_color_cheysson(
  palette = "1880_07",
  discrete = TRUE,
  reverse = FALSE,
  ...
)

scale_colour_cheysson(
  palette = "1880_07",
  discrete = TRUE,
  reverse = FALSE,
```

```
  ...
)

scale_fill_cheysson(palette = "1880_07", discrete = TRUE, reverse = FALSE, ...)
```

Arguments

palette	Name of palette (e.g., "1880_07") or palette type ("sequential", "diverging", "grouped", "category"). Default is "1880_07".
discrete	Whether to use a discrete (TRUE) or continuous (FALSE) scale. Default is TRUE.
reverse	Whether to reverse the palette colors. Default is FALSE.
...	Additional arguments passed to ggplot2 scale functions

Value

A ggplot2 scale object that can be added to a plot. For discrete scales, returns a discrete_scale object. For continuous scales, returns a continuous scale object (gradient).

Examples

```
library(ggplot2)

# Discrete color scale
ggplot(iris, aes(Sepal.Length, Sepal.Width, color = Species)) +
  geom_point() +
  scale_color_cheysson()

# Use a specific palette
ggplot(iris, aes(Sepal.Length, Sepal.Width, color = Species)) +
  geom_point() +
  scale_color_cheysson(palette = "1881_04")

# Use a sequential palette for continuous data
ggplot(iris, aes(Sepal.Length, Sepal.Width, color = Petal.Length)) +
  geom_point() +
  scale_color_cheysson(palette = "sequential", discrete = FALSE)

# Fill scale with category colors
ggplot(iris, aes(Species, Sepal.Width, fill = Species)) +
  geom_boxplot() +
  scale_fill_cheysson(palette = "category")
```

scale_fill_cheysson_pattern *Apply Cheysson patterns to fill aesthetic*

Description

Convenience function that applies the base fill color from Cheysson patterns. Use in combination with pattern_* scales for full pattern effect.

Usage

```
scale_fill_cheysson_pattern(palette = "1881_03", reverse = FALSE, ...)
```

Arguments

palette	Name of palette (e.g., "1881_03") or palette type ("sequential", "diverging", "grouped", "category"). Default is "1881_03".
reverse	Whether to reverse the pattern order. Default is FALSE.
...	Additional arguments passed to ggplot2 scale functions

Value

A ggplot2 discrete scale object for the fill aesthetic. Applies the base fill colors from Cheysson patterns.

Examples

```
# Requires ggpattern package
if (requireNamespace("ggpattern", quietly = TRUE)) {
  library(ggplot2)
  library(ggpattern)

  data <- data.frame(
    category = LETTERS[1:4],
    value = c(15, 23, 18, 20)
  )

  ggplot(data, aes(category, value, fill = category)) +
    geom_col_pattern(aes(pattern_type = category), pattern = "stripe") +
    scale_fill_cheysson_pattern("category") +
    scale_pattern_type_cheysson("category") +
    theme_minimal()
}
```

scale_pattern_cheysson *Cheysson pattern scales for ggpattern*

Description

Pattern fill scales using Cheysson patterns from the Albums de Statistique Graphique. These scales work with ggpattern geoms to apply both colors and hatching patterns.

Usage

```
scale_pattern_fill_cheysson(palette = "1881_03", reverse = FALSE, ...)

scale_pattern_type_cheysson(palette = "1881_03", reverse = FALSE, ...)

scale_pattern_angle_cheysson(palette = "1881_03", reverse = FALSE, ...)

scale_pattern_density_cheysson(palette = "1881_03", reverse = FALSE, ...)
```

Arguments

palette	Name of palette (e.g., "1881_03") or palette type ("sequential", "diverging", "grouped", "category"). Default is "1881_03".
reverse	Whether to reverse the pattern order. Default is FALSE.
...	Additional arguments passed to ggplot2 scale functions

Details

These scales require the ggpattern package. Use with ggpattern geoms like `geom_col_pattern()`, `geom_bar_pattern()`, etc.

The scales apply multiple pattern aesthetics simultaneously:

- `fill`: Base fill color
- `pattern_type`: Type of pattern (none, stripe, crosshatch)
- `pattern_fill`: Color of pattern lines
- `pattern_angle`: Angle of stripes
- `pattern_density`: Density of pattern lines

Value

A ggplot2 discrete scale object for the specified pattern aesthetic (pattern_fill, pattern_type, pattern_angle, or pattern_density). These scales apply the historically accurate Cheysson patterns to ggpattern geoms.

Examples

```
# Requires ggpattern package
if (requireNamespace("ggpattern", quietly = TRUE)) {
  library(ggplot2)
  library(ggpattern)

  # Basic bar chart with patterns
  data <- data.frame(
    category = LETTERS[1:4],
    value = c(15, 23, 18, 20)
  )

  ggplot(data, aes(category, value, fill = category)) +
    geom_col_pattern(
      aes(
        pattern_type = category,
        pattern_fill = category,
        pattern_angle = category
      ),
      pattern = "stripe",
      pattern_density = 0.3,
      color = "black"
    ) +
    scale_pattern_fill_cheysson("category") +
    scale_pattern_type_cheysson("category") +
    scale_pattern_angle_cheysson("category") +
    theme_minimal()
}
```

`show_palette` *Display a Cheysson palette with color swatches and hex codes*

Description

Creates a visual display of a color palette showing color swatches along with their hex codes. This is useful for documentation, presentations, and exploring the available palettes.

Usage

```
show_palette(palette = "1880_07", n = NULL, show_info = TRUE, cex = 1)
```

Arguments

`palette`	Name of palette (e.g., "1880_07"), or palette type ("sequential", "diverging", "grouped", "category").
`n`	Number of colors to display. If NULL (default), shows all colors in the palette. If specified, will interpolate if n > palette size.

show_info	Logical; if TRUE (default), displays palette metadata (type, album, plate) above the swatches.
cex	Text size multiplier for hex codes (default 1).

Value

Invisibly returns a character vector of the displayed hex codes. The function is called primarily for its side effect of creating a plot.

Examples

```
## Not run:
# Display a specific palette
show_palette("1880_07")

# Display palette without metadata
show_palette("1881_03", show_info = FALSE)

# Display 10 interpolated colors
show_palette("1895_04", n = 10)

# Display first sequential palette
show_palette("sequential")

## End(Not run)
```

---

show_palettes *Display multiple Cheysson palettes*

Description

Creates a visual display of multiple color palettes, useful for comparing palettes or showing all palettes of a certain type.

Usage

```
show_palettes(palettes = NULL, ncol = 1, cex = 0.8)
```

Arguments

palettes	Character vector of palette names. If NULL (default), shows all palettes. Can also be a palette type ("sequential", "diverging", "grouped", "category") to show all palettes of that type.
ncol	Number of columns for layout (default 1).
cex	Text size multiplier (default 0.8).

Value

Invisibly returns NULL. The function is called for its side effect of creating a plot.

Examples

```
## Not run:
# Show all sequential palettes
show_palettes("sequential")

# Show specific palettes
show_palettes(c("1880_07", "1881_03", "1895_04"))

## End(Not run)
```

---

theme_cheysson *Cheysson theme for ggplot2*

Description

A ggplot2 theme inspired by the visual style of the Albums de Statistique Graphique, using Cheysson fonts and appropriate styling.

Usage

```
theme_cheysson(
  base_size = 11,
  base_family = "auto",
  title_family = "auto",
  axis_title_family = "auto",
  load_fonts = TRUE
)
```

Arguments

base_size	Base font size (default: 11)
base_family	Base font family. If "auto" (default), uses Cheysson if available, otherwise falls back to sans-serif
title_family	Font family for titles (default: "auto")
axis_title_family	Font family for axis titles (default: "auto")
load_fonts	Automatically load Cheysson fonts if not already loaded (default: TRUE)

Details

This theme applies the following styling:

- Cheysson fonts for all text elements
- Minimal grid lines
- Classic axis styling
- Subtle colors matching historical aesthetics

Font selection:

- Plot title: CheyssonTitle (decorative)
- Axis titles: CheyssonSansCaps (capitals)
- Body text: Cheysson (regular)

Value

A ggplot2 theme object

Examples

```
## Not run:
library(ggplot2)

# Load fonts first (required for proper rendering)
load_cheysson_fonts()

# Basic usage
ggplot(mtcars, aes(wt, mpg)) +
  geom_point() +
  labs(title = "Automobile Statistics") +
  theme_cheysson()

# With Cheysson color palette
ggplot(iris, aes(Sepal.Length, Sepal.Width, color = Species)) +
  geom_point(size = 3) +
  scale_color_cheysson("1881_04") +
  theme_cheysson()

## End(Not run)
```

---

theme_cheysson_map *Cheysson map theme*

Description

A theme specifically designed for maps, with no grid and minimal elements, in the style of Cheysson's cartographic works.

Usage

```
theme_cheysson_map(
  base_size = 11,
  base_family = "auto",
  title_family = "auto",
  load_fonts = TRUE
)
```

Arguments

base_size	Base font size (default: 11)
base_family	Base font family. If "auto" (default), uses Cheysson if available, otherwise falls back to sans-serif
title_family	Font family for titles (default: "auto")
load_fonts	Automatically load Cheysson fonts if not already loaded (default: TRUE)

Value

A ggplot2 theme object that can be added to a plot with +.

Examples

```
## Not run:
# For use with spatial data/maps
library(ggplot2)

# Load fonts first
load_cheysson_fonts()

# Example with spatial data (requires sf package)
if (requireNamespace("sf", quietly = TRUE)) {
  # ggplot(map_data) + geom_sf() + theme_cheysson_map()
}

## End(Not run)
```

---

theme_cheysson_minimal

Minimal Cheysson theme

Description

A more minimal version of theme_cheysson with fewer grid lines, suitable for maps and diagrams.

Usage

```
theme_cheysson_minimal(
  base_size = 11,
  base_family = "auto",
  title_family = "auto",
  axis_title_family = "auto",
  load_fonts = TRUE
)
```

Arguments

base_size	Base font size (default: 11)
base_family	Base font family. If "auto" (default), uses Cheysson if available, otherwise falls back to sans-serif
title_family	Font family for titles (default: "auto")
axis_title_family	Font family for axis titles (default: "auto")
load_fonts	Automatically load Cheysson fonts if not already loaded (default: TRUE)

Value

A ggplot2 theme object that can be added to a plot with +.

Examples

```
## Not run:
library(ggplot2)

# Load fonts first
load_cheysson_fonts()

ggplot(mtcars, aes(wt, mpg)) +
  geom_point() +
  theme_cheysson_minimal()

## End(Not run)
```

Index