

Package: mwmap (via r-universe)

May 12, 2026

Title Create Maps of Malawi Administrative Boundaries

Version 1.0.0

Description Provides a tidy, high-level interface for creating polished maps of Malawi at country, region, district, and Traditional Authority level. Functions handle spatial data retrieval, administrative-name matching, joins from ordinary data frames, numeric and categorical choropleths, labels, highlights, and professional 'ggplot2' styling. Spatial boundary data are provided by the companion package 'mwmapdata'.

License MIT + file LICENSE

Encoding UTF-8

RoxygenNote 7.3.3

Depends R (>= 3.5)

Imports dplyr, ggplot2, grid, mwmapdata (>= 1.0.0), grDevices, rlang, sf, tools, utils

Suggests ggrepel, ggspatial, plotly, knitr, rmarkdown, testthat (>= 3.0.0)

URL <https://github.com/bitacanalytics/mwmap>

BugReports <https://github.com/bitacanalytics/mwmap/issues>

Config/testthat/edition 3

NeedsCompilation no

Author Chifundo Bita [aut, cre]

Maintainer Chifundo Bita <bitac.analytics@gmail.com>

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

Date/Publication 2026-05-12 21:45:26 UTC

RemoteUrl <https://github.com/cran/mwmap>

RemoteRef HEAD

RemoteSha df9f08405338a39ece8733f95c210dc9b055cc0b

Contents

get_mw_palette	2
get_mw_palettes	3
mw_choropleth	4
mw_clean_names	5
mw_districts	5
mw_districts_simple	6
mw_get_map	7
mw_highlight	7
mw_highlight_multi	9
mw_join	10
mw_label_repel	11
mw_labels	12
mw_layout	13
mw_map	16
mw_points	19
mw_regions	21
mw_suggest_matches	22
mw_ta_map	23
mw_tas	24
scale_fill_mw	25
view_mw_palettes	26
Index	27

get_mw_palette	<i>Get Malawi Palette Colors</i>
----------------	----------------------------------

Description

Extract specific colors from Malawi palettes for custom use.

Usage

```
get_mw_palette(name, n = NULL, reverse = FALSE, interpolate = TRUE)
```

Arguments

name	Palette name.
n	Number of colors to return. If NULL, returns all colors in palette. Default: NULL.
reverse	Logical. Reverse palette order. Default: FALSE.
interpolate	Logical. Interpolate to get exactly n colors. If FALSE, returns first n colors. Default: TRUE.

Value

A character vector of hex color codes.

Examples

```
# Get all health palette colors
get_mw_palette("health")

# Get 3 colors from malaria palette
get_mw_palette("malaria", n = 3)

# Get reversed regions palette
get_mw_palette("regions", reverse = TRUE)
```

get_mw_palettes *List Available Malawi Palettes*

Description

Returns a data frame of all available palettes with descriptions.

Usage

```
get_mw_palettes(category = NULL)
```

Arguments

category Optional category filter: "health", "agriculture", "socioeconomic", "sequential", "diverging", "qualitative".

Value

A data frame with palette names and descriptions.

Examples

```
# List all palettes
get_mw_palettes()

# List only health palettes
get_mw_palettes("health")
```

mw_choropleth

Create a Malawi Choropleth Map

Description

A convenient wrapper around `mw_map()` for mapping numeric or categorical values attached to Malawi administrative units.

Usage

```
mw_choropleth(
  data,
  value,
  unit_col,
  level = 2,
  palette = NULL,
  title = NULL,
  subtitle = NULL,
  caption = NULL,
  legend_title = NULL,
  ...
)
```

Arguments

<code>data</code>	Data frame containing values to map.
<code>value</code>	Column to map to fill colour. May be quoted or unquoted.
<code>unit_col</code>	Column containing administrative unit names. May be quoted or unquoted. Defaults by level.
<code>level</code>	Administrative level. Use "ta" for Traditional Authorities.
<code>palette</code>	Malawi palette name, colour vector, or palette function.
<code>title, subtitle, caption</code>	Plot labels.
<code>legend_title</code>	Legend title.
<code>...</code>	Passed to <code>mw_map()</code> .

Value

A ggplot2 object.

Examples

```
district_data <- data.frame(
  district = c("Lilongwe", "Blantyre", "Mzuzu"),
  cases = c(120, 80, 35)
)
```

```
mw_choropleth(district_data, cases)

ta_data <- data.frame(
  ta = c("Mabuka", "Mwaulambia"),
  status = c("On track", "Needs support")
)
mw_choropleth(ta_data, status, level = "ta")
```

mw_clean_names	<i>Clean Malawi Administrative Names</i>
----------------	--

Description

Standardise names for display and backwards-compatible workflows. For joins, `mw_join()` uses a stricter internal key that is robust to punctuation, case, and common suffixes such as "District" and "TA".

Usage

```
mw_clean_names(x)
```

Arguments

`x` Character vector of names.

Value

A character vector.

Examples

```
mw_clean_names(c("lilongwe district", "Nkhata Bay", "T/A Mabuka"))
```

mw_districts	<i>Get Malawi District Names</i>
--------------	----------------------------------

Description

Return district names from the current `mwmappedata` boundary data.

Usage

```
mw_districts(
  region = NULL,
  type = c("standard", "admin", "short", "all"),
  sorted = TRUE,
  include_ta = FALSE,
  quiet = FALSE
)
```

Arguments

region	Optional region filter. Accepts "Northern", "Central", "Southern" and short-cuts such as "n", "c", and "s".
type	"standard", "admin", "short", or "all".
sorted	Sort alphabetically.
include_ta	Deprecated. Use <code>mw_tas()</code> for Traditional Authorities.
quiet	Suppress messages.

Value

A character vector, or a data frame when `type = "all"`.

Examples

```
mw_districts()  
mw_districts("Southern")  
mw_districts(type = "all")
```

`mw_districts_simple` *Simple District Name Helper*

Description

Backwards-compatible alias for `mw_districts()`.

Usage

```
mw_districts_simple(region = NULL)
```

Arguments

region	Optional region filter.
--------	-------------------------

Value

Character vector of district names.

`mw_get_map`*Get Malawi Boundary Data*

Description

Return Malawi administrative boundaries from **mwmappedata**, with optional region, district, TA, and projection filters.

Usage

```
mw_get_map(  
  level = 2,  
  region = NULL,  
  districts = NULL,  
  tas = NULL,  
  projection = "EPSG:4326"  
)
```

Arguments

<code>level</code>	Administrative level: 0/"country", 1/"region", 2/"district", or 3/"ta".
<code>region</code>	Optional region filter.
<code>districts</code>	Optional district filter.
<code>tas</code>	Optional Traditional Authority filter.
<code>projection</code>	Coordinate reference system. Defaults to "EPSG:4326".

Value

An sf object.

Examples

```
mw_get_map("district")  
mw_get_map("ta", districts = "Lilongwe")
```

`mw_highlight`*Highlight Selected Districts on Malawi Map*

Description

Adds emphasis to specific districts by overlaying them with custom fill, border, or label styles. Useful for drawing attention to areas of interest.

Usage

```
mw_highlight(
  districts,
  fill = "red",
  color = "black",
  alpha = 0.8,
  size = 1,
  linetype = "solid",
  label = FALSE,
  label_size = 4,
  label_color = "black",
  label_fontface = "bold",
  data = NULL,
  district_col = "ADM2_EN",
  outline_only = FALSE,
  ...
)
```

Arguments

<code>districts</code>	Character vector of district names to highlight. Case insensitive matching is applied.
<code>fill</code>	Fill colour for highlighted districts. Default: "red". Use NA for transparent fill.
<code>color</code>	Border colour for highlighted districts. Default: "black".
<code>alpha</code>	Transparency level (0-1). Default: 0.8.
<code>size</code>	Border size for highlighted districts. Default: 1.
<code>linetype</code>	Line type for borders. Default: "solid".
<code>label</code>	Logical. Add district labels to highlighted areas? Default: FALSE.
<code>label_size</code>	Numeric. Label text size. Default: 4.
<code>label_color</code>	Character. Label text colour. Default: "black".
<code>label_fontface</code>	Label font style. Default: "bold".
<code>data</code>	Optional sf object. If NULL, uses <code>mwmapdata::mw_level_2</code> .
<code>district_col</code>	Column containing district names. Default: "ADM2_EN".
<code>outline_only</code>	Logical. Show only outline without fill? Default: FALSE.
<code>...</code>	Additional arguments passed to <code>ggplot2::geom_sf()</code> .

Value

A ggplot2 layer object that can be added to a map.

Examples

```
library(ggplot2)

# Basic highlighting
```

```
mw_map() +
  mw_highlight("Lilongwe")

# Multiple districts with custom styling
mw_map() +
  mw_highlight(c("Lilongwe", "Blantyre", "Mzuzu City"),
              fill = "gold", color = "darkred", alpha = 0.5)

# Outline only (for emphasis without obscuring)
mw_map(fill_color = "lightgrey") +
  mw_highlight("Mzimba", outline_only = TRUE, size = 1.2)

# With labels
mw_map() +
  mw_highlight("Zomba", label = TRUE, label_size = 5)
```

`mw_highlight_multi` *Highlight Multiple Districts with Different Colors*

Description

Highlights different districts with potentially different colors.

Usage

```
mw_highlight_multi(district_list, ...)
```

Arguments

`district_list` Named list or vector. Names are district names, values are fill colors. Example:
 `c("Lilongwe" = "red", "Blantyre" = "blue")`

`...` Additional arguments passed to `mw_highlight()` for all layers.

Value

List of ggplot2 layers.

Examples

```
mw_map() +
  mw_highlight_multi(c("Lilongwe" = "gold",
                     "Blantyre" = "steelblue",
                     "Mzimba" = "forestgreen"))
```

mw_join

*Join Data to Malawi Boundaries***Description**

Join a regular data frame to Malawi administrative boundary geometries. The function is level-aware, so the default join key changes automatically for country, region, district, and Traditional Authority maps.

Usage

```
mw_join(
  data,
  unit_col,
  level = 2,
  map = NULL,
  map_col = NULL,
  keep_all = TRUE,
  unmatched = c("message", "warning", "error", "ignore"),
  quiet = FALSE,
  district_col = NULL,
  name_clean = NULL,
  by = NULL,
  verbose = NULL,
  ...
)
```

Arguments

data	A data frame containing values to map.
unit_col	Column in data containing the administrative unit names. May be quoted or unquoted. Defaults to country, region, district, or ta depending on level.
level	Administrative level: 0/"country", 1/"region", 2/"district", or 3/"ta".
map	Optional sf object to join to. Defaults to the corresponding object from mwmappedata .
map_col	Column in map containing administrative unit names. Defaults to the correct ADM column for level.
keep_all	If TRUE, keep all map features and attach matching values. If FALSE, keep only matched features.
unmatched	One of "message", "warning", "error", or "ignore". Controls how unmatched input names are reported.
quiet	Suppress matching messages.
district_col	Deprecated alias for unit_col.
name_clean	Deprecated. Name matching now uses mwmap's internal normalisation.

by	Optional explicit join specification passed to dplyr joins.
verbose	Deprecated alias for !quiet.
...	Passed to dplyr joins.

Value

An sf object with user columns joined to Malawi geometries.

Examples

```
district_data <- data.frame(
  district = c("Lilongwe", "Blantyre", "Mzuzu"),
  cases = c(120, 80, 35)
)
mw_join(district_data)

ta_data <- data.frame(
  ta = c("Mabuka", "Mwaulambia"),
  coverage = c(72, 64)
)
mw_join(ta_data, level = "ta")
```

mw_label_repel

Add Label Repel for Better Placement

Description

Alternative to mw_labels that uses ggrepel to prevent overlapping labels. Requires the ggrepel package.

Usage

```
mw_label_repel(..., force = 1, max.overlaps = 10)
```

Arguments

...	Arguments passed to mw_labels()
force	Numeric. Repulsion force. Default: 1.
max.overlaps	Numeric. Maximum allowed overlaps. Default: 10.

Value

A ggrepel layer object.

Examples

```
if (requireNamespace("ggrepel", quietly = TRUE)) {
  mw_map() + mw_label_repel()
}
```

mw_labels

Add District Labels to Malawi Map

Description

Adds text labels for districts to a Malawi map. Labels are positioned at district centroids with options for customization.

Usage

```
mw_labels(
  districts = NULL,
  size = 3,
  color = "black",
  fontface = "bold",
  family = "",
  alpha = 1,
  angle = 0,
  hjust = 0.5,
  vjust = 0.5,
  check_overlap = TRUE,
  show.legend = FALSE,
  data = NULL,
  label_column = "ADM2_EN",
  ...
)
```

Arguments

<code>districts</code>	Character vector of district names to label. If NULL, labels all districts. Default: NULL.
<code>size</code>	Numeric. Text size in points. Default: 3.
<code>color</code>	Character. Text colour. Default: "black".
<code>fontface</code>	Character or numeric. Font style: "plain", "bold", "italic", "bold.italic". Default: "bold".
<code>family</code>	Character. Font family. Default: "" (system default).
<code>alpha</code>	Numeric. Text transparency (0-1). Default: 1.
<code>angle</code>	Numeric. Text rotation angle in degrees. Default: 0.

hjust	Numeric. Horizontal justification (0-1). Default: 0.5.
vjust	Numeric. Vertical justification (0-1). Default: 0.5.
check_overlap	Logical. If TRUE, prevents overlapping labels. Default: TRUE.
show.legend	Logical. Include in legend? Default: FALSE.
data	Optional sf object. If NULL, uses mwmapdata::mw_level_2.
label_column	Character. Column name containing labels. Default: "ADM2_EN".
...	Additional arguments passed to <code>ggplot2::geom_sf_text()</code> .

Value

A ggplot2 layer object that can be added to a map.

Examples

```
library(ggplot2)

# Basic usage
mw_map() + mw_labels()

# Customized labels
mw_map() +
  mw_labels(size = 4, color = "darkblue", fontface = "italic")

# Labels for specific districts only
library(dplyr)
selected_districts <- mwmapdata::mw_level_2 %>%
  filter(ADM2_EN %in% c("Lilongwe", "Blantyre", "Mzuzu City"))

mw_map() +
  mw_labels(data = selected_districts, color = "red", size = 5)
```

mw_layout

Apply Consistent Layout Styling to Malawi Maps

Description

Adds professionally formatted titles, theme elements, and layout options to Malawi maps. Provides a consistent look and feel across visualizations.

Usage

```
mw_layout(
  title = NULL,
  subtitle = NULL,
  caption = NULL,
```

```

legend_position = "right",
legend_title = NULL,
legend_direction = "vertical",
theme = c("void", "minimal", "classic", "custom"),
font_family = "",
title_size = 16,
title_face = "bold",
subtitle_size = 12,
subtitle_face = "plain",
caption_size = 9,
caption_face = "italic",
legend_text_size = 10,
legend_title_size = 11,
legend_title_face = "bold",
legend_key_size = 1,
legend_spacing = 0.5,
margin = grid::unit(c(0.2, 0.2, 0.2, 0.2), "cm"),
panel_border = FALSE,
panel_border_color = "grey50",
background_color = "white",
grid_color = "grey90",
grid_major = FALSE,
grid_minor = FALSE,
axis_text = FALSE,
axis_ticks = FALSE,
...
)

```

Arguments

<code>title</code>	Character. Main map title. Default: NULL.
<code>subtitle</code>	Character. Map subtitle. Default: NULL.
<code>caption</code>	Character. Map caption (usually data source). Default: NULL.
<code>legend_position</code>	Character or numeric. Legend position: "right", "left", "top", "bottom", "none", or coordinates <code>c(x, y)</code> . Default: "right".
<code>legend_title</code>	Character. Legend title. If NULL, uses existing legend title. Default: NULL.
<code>legend_direction</code>	Character. Legend layout: "vertical" or "horizontal". Default: "vertical".
<code>theme</code>	Character. Base theme: "void", "minimal", "classic", or "custom". Default: "void".
<code>font_family</code>	Character. Base font family. Default: "" (system default).
<code>title_size</code>	Numeric. Title font size. Default: 16.
<code>title_face</code>	Character. Title font face. Default: "bold".
<code>subtitle_size</code>	Numeric. Subtitle font size. Default: 12.
<code>subtitle_face</code>	Character. Subtitle font face. Default: "plain".

caption_size Numeric. Caption font size. Default: 9.

caption_face Character. Caption font face. Default: "italic".

legend_text_size
 Numeric. Legend text size. Default: 10.

legend_title_size
 Numeric. Legend title size. Default: 11.

legend_title_face
 Character. Legend title font face. Default: "bold".

legend_key_size
 Numeric. Legend key size (in cm). Default: 1.

legend_spacing Numeric. Legend spacing (in cm). Default: 0.5.

margin Numeric vector or unit. Plot margins. Default: unit(c(0.2, 0.2, 0.2, 0.2), "cm").

panel_border Logical. Add panel border? Default: FALSE.

panel_border_color
 Character. Panel border color. Default: "grey50".

background_color
 Character. Plot background color. Default: "white".

grid_color Character. Grid line color. Default: "grey90".

grid_major Logical. Show major grid lines? Default: FALSE.

grid_minor Logical. Show minor grid lines? Default: FALSE.

axis_text Logical. Show axis text? Default: FALSE.

axis_ticks Logical. Show axis ticks? Default: FALSE.

... Additional arguments passed to `ggplot2::theme()`.

Value

A list of ggplot2 theme modifications and labs.

Examples

```
library(ggplot2)

# Basic layout
mw_map() +
  mw_layout("Malawi Health Districts")

# Full layout with all elements
mw_map(fill = "population") +
  mw_layout(
    title = "Population Distribution in Malawi",
    subtitle = "Data from 2023 Census",
    caption = "Source: National Statistical Office",
    legend_position = "bottom",
    legend_title = "Population",
    panel_border = TRUE,
    grid_major = TRUE
```

```
)  
  
# Minimal layout  
mw_map() +  
  mw_layout(  
    theme = "minimal",  
    legend_position = "none"  
  )  
  
# Custom colors  
mw_map() +  
  mw_layout(  
    title = "Malawi Map",  
    background_color = "#f5f5f5",  
    panel_border = TRUE,  
    panel_border_color = "darkblue"  
  )
```

mw_map

Create a Professional Malawi Map

Description

`mw_map()` is the main high-level plotting function in `mwmap`. It can draw Malawi boundaries at country, region, district, or Traditional Authority level, join your data by name, and choose an appropriate colour scale for numeric or categorical values.

Usage

```
mw_map(  
  data = NULL,  
  fill,  
  unit_col,  
  level = 2,  
  region = NULL,  
  districts = NULL,  
  tas = NULL,  
  palette = NULL,  
  scale_type = c("auto", "continuous", "discrete"),  
  reverse = FALSE,  
  na_color = "#D7DCE2",  
  fill_color = "#F2F4F3",  
  border_color = "#252222",  
  border_size = 0.25,  
  alpha = 1,  
  lakes = FALSE,
```

```

lake_color = "#A7D8F0",
lake_border_color = "#5D9BC2",
district_borders = NULL,
district_border_color = "#2F3437",
district_border_size = 0.45,
highlight_districts = NULL,
highlight_tas = NULL,
highlight_color = "#D7263D",
highlight_size = 1,
labels = FALSE,
label_column = NULL,
label_size = NULL,
label_color = "#222222",
label_repel = FALSE,
title = NULL,
subtitle = NULL,
caption = NULL,
legend_title = NULL,
legend_position = "right",
projection = "EPSG:4326",
scale_bar = FALSE,
north_arrow = FALSE,
interactive = FALSE,
quiet = FALSE,
...
)

```

Arguments

data	Optional data frame or sf object. If a data frame is supplied it is joined to the selected Malawi boundaries.
fill	Optional column to map to fill colour. May be quoted or unquoted.
unit_col	Column in data containing names to join by. May be quoted or unquoted. Defaults to country, region, district, or ta, depending on level.
level	Administrative level: 0/"country", 1/"region", 2/"district", or 3/"ta".
region	Optional region filter.
districts	Optional district filter. For level = "ta", this maps TAs only inside the selected districts.
tas	Optional Traditional Authority filter.
palette	Name of a Malawi palette, a vector of colours, or a palette function. Defaults to "health" for numeric data and "qualitative_2" for categorical data.
scale_type	"auto", "continuous", or "discrete".
reverse	Reverse the fill palette.
na_color	Fill colour for missing values.
fill_color	Fill colour used when fill is not supplied.

<code>border_color</code>	Boundary colour.
<code>border_size</code>	Boundary line width.
<code>alpha</code>	Fill opacity.
<code>lakes</code>	Add Lake Malawi.
<code>lake_color</code>	Lake fill colour.
<code>lake_border_color</code>	Lake border colour.
<code>district_borders</code>	Add district outlines on TA maps.
<code>district_border_color</code>	District outline colour on TA maps.
<code>district_border_size</code>	District outline width on TA maps.
<code>highlight_districts</code>	Districts to outline.
<code>highlight_tas</code>	Traditional Authorities to outline.
<code>highlight_color</code>	Highlight outline colour.
<code>highlight_size</code>	Highlight outline width.
<code>labels</code>	Add labels for mapped features.
<code>label_column</code>	Optional label column. Defaults to the level name column.
<code>label_size</code>	Label size.
<code>label_color</code>	Label colour.
<code>label_repel</code>	Use ggrepel for label placement if installed.
<code>title, subtitle, caption</code>	Plot labels.
<code>legend_title</code>	Legend title. Defaults to the fill column.
<code>legend_position</code>	Legend position.
<code>projection</code>	Coordinate reference system.
<code>scale_bar</code>	Add a scale bar if ggspatial is installed.
<code>north_arrow</code>	Add a north arrow if ggspatial is installed.
<code>interactive</code>	Return a plotly object if plotly is installed.
<code>quiet</code>	Suppress join messages.
<code>...</code>	Additional arguments passed to <code>ggplot2::geom_sf()</code> .

Value

A `ggplot2` object, or a plotly object when `interactive = TRUE`.

Examples

```
mw_map()

df <- data.frame(
  district = c("Lilongwe", "Blantyre", "Mzuzu"),
  cases = c(120, 80, 35)
)
mw_map(df, fill = cases)

ta_df <- data.frame(
  ta = c("Mabuka", "Mwaulambia"),
  coverage = c(72, 64)
)
mw_map(ta_df, fill = coverage, level = "ta", districts = "Mulanje")
```

mw_points

Add Point Locations to Malawi Map

Description

Plot geographic points (e.g., survey clusters, health facilities, schools) on a Malawi map with extensive customization options.

Usage

```
mw_points(
  data,
  lon,
  lat,
  color = "red",
  size = 2,
  shape = 19,
  alpha = 0.8,
  stroke = 0.5,
  mapping = NULL,
  show.legend = TRUE,
  jitter = FALSE,
  jitter_width = 0.1,
  jitter_height = 0.1,
  label = FALSE,
  label_column = NULL,
  label_size = 3,
  label_color = "black",
  repel_labels = FALSE,
  ...
)
```

Arguments

<code>data</code>	Data frame containing coordinates and optional attributes.
<code>lon</code>	Longitude column (unquoted).
<code>lat</code>	Latitude column (unquoted).
<code>color</code>	Point color. Can be a single color or a column name for color mapping. Default: "red".
<code>size</code>	Point size. Can be a single value or a column name for size mapping. Default: 2.
<code>shape</code>	Point shape. Default: 19 (filled circle).
<code>alpha</code>	Point transparency (0-1). Default: 0.8.
<code>stroke</code>	Border thickness for points. Default: 0.5.
<code>mapping</code>	Optional aesthetic mapping created with <code>ggplot2::aes()</code> .
<code>show.legend</code>	Logical. Show legend? Default: TRUE.
<code>jitter</code>	Logical. Add small random noise to points to reduce overplotting. Default: FALSE.
<code>jitter_width</code>	Numeric. Width of jitter. Default: 0.1.
<code>jitter_height</code>	Numeric. Height of jitter. Default: 0.1.
<code>label</code>	Logical. Add labels to points. Default: FALSE.
<code>label_column</code>	Column for point labels (if <code>label = TRUE</code>).
<code>label_size</code>	Size of point labels. Default: 3.
<code>label_color</code>	Color of point labels. Default: "black".
<code>repel_labels</code>	Use <code>ggrepel</code> to avoid overlapping labels. Default: FALSE.
<code>...</code>	Additional arguments passed to <code>ggplot2::geom_point()</code> or <code>ggrepel::geom_text_repel()</code> .

Value

A `ggplot2` layer object that can be added to a map.

Examples

```
library(ggplot2)

# Sample health facilities
facilities <- data.frame(
  name = c("Lilongwe Central Hospital", "Queen Elizabeth Central Hospital",
           "Mzuzu Central Hospital", "Zomba Central Hospital"),
  lon = c(33.78, 35.00, 34.02, 35.32),
  lat = c(-13.98, -15.78, -11.46, -15.38),
  type = c("Central", "Central", "Central", "Central"),
  beds = c(1200, 1350, 600, 450)
)

# Basic points
mw_map() +
```

```
mw_points(facilities, lon, lat)

# Colored by type (pass column name as a string)
mw_map() +
  mw_points(facilities, lon, lat, color = "type")

# Sized by beds (pass column name as a string)
mw_map() +
  mw_points(facilities, lon, lat, size = "beds", alpha = 0.7)

# With labels
mw_map() +
  mw_points(facilities, lon, lat,
            label = TRUE, label_column = "name")

# Custom styling
mw_map() +
  mw_points(facilities, lon, lat,
            color = "darkblue", shape = 16, size = 4, alpha = 0.6)
```

mw_regions

Create a Malawi Regions Map

Description

Draw Malawi's three administrative regions with the same polished defaults used by `mw_map()`.

Usage

```
mw_regions(
  data = NULL,
  fill,
  region_col,
  palette = "regions",
  labels = FALSE,
  title = NULL,
  subtitle = NULL,
  caption = NULL,
  ...
)
```

Arguments

<code>data</code>	Optional region-level data.
<code>fill</code>	Optional fill column. May be quoted or unquoted.
<code>region_col</code>	Region-name column in data. Defaults to region.

palette	Malawi palette name, colour vector, or palette function.
labels	Add region labels.
title, subtitle, caption	Plot labels.
...	Passed to <code>mw_map()</code> .

Value

A `ggplot2` object.

Examples

```
mw_regions(labels = TRUE)

df <- data.frame(region = c("Northern", "Central", "Southern"),
                 value = c(1, 2, 3))
mw_regions(df, fill = value)
```

`mw_suggest_matches` *Suggest Close Malawi Name Matches*

Description

Suggest Close Malawi Name Matches

Usage

```
mw_suggest_matches(x, candidates, n = 3)
```

Arguments

x	Character. Name to match.
candidates	Character vector of valid names.
n	Number of suggestions.

Value

Character vector of suggested names.

Examples

```
mw_suggest_matches("Lilongwe", mw_districts())
```

mw_ta_map

Create a Traditional Authority Map

Description

Map Malawi Traditional Authorities. This wrapper is optimised for TA-level work and supports numeric or categorical fills by TA name.

Usage

```
mw_ta_map(
  data = NULL,
  fill,
  ta_col,
  districts = NULL,
  region = NULL,
  tas = NULL,
  palette = NULL,
  labels = FALSE,
  title = NULL,
  subtitle = NULL,
  caption = NULL,
  district_borders = TRUE,
  ...
)
```

Arguments

data	Optional data frame with TA-level values.
fill	Optional column to map to fill colour. May be quoted or unquoted.
ta_col	Column in data containing TA names. Defaults to ta.
districts	Optional district filter.
region	Optional region filter.
tas	Optional TA filter.
palette	Malawi palette name, colour vector, or palette function.
labels	Add TA labels.
title, subtitle, caption	Plot labels.
district_borders	Add district outlines.
...	Passed to <code>mw_map()</code> .

Value

A ggplot2 object.

Examples

```
mw_ta_map(districts = "Lilongwe")

ta_data <- data.frame(
  ta = c("Mabuka", "Mwaulambia"),
  coverage = c(72, 64)
)
mw_ta_map(ta_data, fill = coverage, districts = "Mulanje")

ta_status <- data.frame(
  ta = c("Mabuka", "Mwaulambia"),
  status = c("On track", "Needs support")
)
mw_ta_map(ta_status, fill = status)
```

mw_tas

Get Malawi Traditional Authority Names

Description

Return Traditional Authority names, with optional region or district filters.

Usage

```
mw_tas(
  region = NULL,
  districts = NULL,
  type = c("standard", "admin", "short", "all"),
  sorted = TRUE
)
```

Arguments

region	Optional region filter.
districts	Optional district filter.
type	"standard", "admin", "short", or "all".
sorted	Sort alphabetically.

Value

A character vector, or a data frame when type = "all".

Examples

```
mw_tas(districts = "Lilongwe")
mw_tas(region = "Southern", type = "all")
```

scale_fill_mw	<i>Malawi Fill Scale</i>
---------------	--------------------------

Description

Apply Malawi-themed color palettes to ggplot2 maps. Supports both continuous and discrete scales with options for colorblind-friendly palettes.

Usage

```
scale_fill_mw(  
  palette = "health",  
  reverse = FALSE,  
  discrete = FALSE,  
  colorblind_friendly = FALSE,  
  direction = 1,  
  ...  
)
```

Arguments

palette	Name of palette. See get_mw_palettes() for available options. Default: "health".
reverse	Logical. Reverse palette order. Default: FALSE.
discrete	Logical. Use discrete scale. Default: FALSE (continuous).
colorblind_friendly	Logical. Use only colorblind-friendly palettes. Default: FALSE.
direction	Numeric. Direction of palette: 1 = normal, -1 = reversed. Default: 1.
...	Additional arguments passed to ggplot2::scale_fill_gradientn() or ggplot2::discrete_scale()

Value

A ggplot2 scale object.

Examples

```
library(ggplot2)  
  
df <- data.frame(  
  district = c("Lilongwe", "Blantyre", "Mzimba"),  
  value = c(10, 20, 30),  
  category = c("A", "B", "A")  
)  
# Continuous scale  
mw_map(data = df, fill = "value") +  
  scale_fill_mw("population")  
  
# Discrete scale
```

```
mw_map(data = df, fill = "category") +  
  scale_fill_mw("qualitative_1", discrete = TRUE)  
  
# Reversed  
mw_map(data = df, fill = "value") +  
  scale_fill_mw("malaria", reverse = TRUE)
```

view_mw_palettes

Visualize Malawi Palettes

Description

Display all available palettes for visual inspection.

Usage

```
view_mw_palettes(n = NULL, ncol = 3)
```

Arguments

n	Number of colors to show for each palette. Default: NULL (all colors).
ncol	Number of columns in plot. Default: 3.

Value

A ggplot2 object showing palette swatches.

Examples

```
# Show all palettes  
view_mw_palettes()  
  
# Show 5 colors from each palette  
view_mw_palettes(n = 5)
```

Index

`get_mw_palette`, 2
`get_mw_palettes`, 3
`get_mw_palettes()`, 25
`ggplot2::aes()`, 20
`ggplot2::discrete_scale()`, 25
`ggplot2::geom_point()`, 20
`ggplot2::geom_sf()`, 8, 18
`ggplot2::geom_sf_text()`, 13
`ggplot2::scale_fill_gradientn()`, 25
`ggplot2::theme()`, 15
`ggrepel::geom_text_repel()`, 20

`mw_choropleth`, 4
`mw_clean_names`, 5
`mw_districts`, 5
`mw_districts()`, 6
`mw_districts_simple`, 6
`mw_get_map`, 7
`mw_highlight`, 7
`mw_highlight()`, 9
`mw_highlight_multi`, 9
`mw_join`, 10
`mw_label_repel`, 11
`mw_labels`, 12
`mw_layout`, 13
`mw_map`, 16
`mw_map()`, 4, 21–23
`mw_points`, 19
`mw_regions`, 21
`mw_suggest_matches`, 22
`mw_ta_map`, 23
`mw_tas`, 24
`mw_tas()`, 6

`scale_fill_mw`, 25

`view_mw_palettes`, 26