

Package ‘maptiles’

September 24, 2021

Title Download and Display Map Tiles

Version 0.3.0

Description To create maps from tiles, 'maptiles' downloads, composes and displays tiles from a large number of providers (e.g. 'OpenStreetMap', 'Stamen', 'Esri', 'CARTO', or 'Thunderforest').

URL <https://github.com/riatelab/maptiles/>

BugReports <https://github.com/riatelab/maptiles/issues/>

License GPL-3

Depends R (>= 3.5.0)

Imports sf (>= 0.9-5), curl, graphics, grDevices, png, terra, tools, slippymath

Suggests covr, tinytest

Encoding UTF-8

RoxygenNote 7.1.2

NeedsCompilation no

Author Timothée Giraud [cre, aut] (<<https://orcid.org/0000-0002-1932-3323>>),
Diego Hernangómez [ctb] (<<https://orcid.org/0000-0001-8457-4658>>),
Robert J. Hijmans [ctb] (<<https://orcid.org/0000-0001-5872-2872>>),
Hugh A. Graham [ctb]

Maintainer Timothée Giraud <timothee.giraud@cnrs.fr>

Repository CRAN

Date/Publication 2021-09-24 14:50:02 UTC

R topics documented:

get_credit	2
get_tiles	2
maptiles	4
plot_tiles	4

Index	6
--------------	----------

get_credit	<i>Get basemap tiles attribution</i>
------------	--------------------------------------

Description

Get the attribution of map tiles.

Usage

```
get_credit(provider)
```

Arguments

provider	provider name
----------	---------------

Examples

```
get_credit("OpenStreetMap")
```

get_tiles	<i>Get basemap tiles from map servers</i>
-----------	---

Description

Get map tiles based on a spatial object extent. Maps can be fetched from various map servers.

Usage

```
get_tiles(  
  x,  
  provider = "OpenStreetMap",  
  zoom,  
  crop = FALSE,  
  verbose = FALSE,  
  apikey,  
  cachedir,  
  forceDownload = FALSE  
)
```

Arguments

x	an sf, sfc, bbox, SpatRaster, SpatVector or SpatExtent object. If x is a SpatExtent it must express coordinates in lon/lat WGS84 (epsg:4326).
provider	the tile server from which to get the map. It can be a name (see Details for providers) or a named list like this one: <code>provider = list(src = "name of the source", q = "server address", sub = "subdomains", cit = "how to cite the tiles")</code> (see Examples).
zoom	the zoom level (see Details).
crop	TRUE if results should be cropped to the specified x extent, FALSE otherwise. If x is an sf object with one POINT, crop is set to FALSE.
verbose	if TRUE, tiles filepaths, zoom level and citation are displayed.
apikey	API key, needed for Thunderforest servers
cachedir	name of a directory used to cache tiles. If not set, tiles are cached in a tempdir folder.
forceDownload	if TRUE, existing cached tiles may be overwritten

Details

Zoom levels are described on the OpenStreetMap wiki: https://wiki.openstreetmap.org/wiki/Zoom_levels.

Providers:

"OpenStreetMap.MapnikBW", "OpenStreetMap", "OpenStreetMap.DE", "OpenStreetMap.France", "OpenStreetMap.HOT",
 "Stamen.Toner", "Stamen.TonerBackground", "Stamen.TonerHybrid", "Stamen.TonerLines", "Stamen.TonerLabels", "Stamen.TonerLite", "Stamen.Watercolor", "Stamen.Terrain", "Stamen.TerrainBackground", "Stamen.TerrainLabels",
 "Esri.WorldStreetMap", "Esri.DeLorme", "Esri.WorldTopoMap", "Esri.WorldImagery", "Esri.WorldTerrain", "Esri.WorldShadedRelief", "Esri.OceanBasemap", "Esri.NatGeoWorldMap", "Esri.WorldGrayCanvas",
 "CartoDB.Positron", "CartoDB.PositronNoLabels", "CartoDB.PositronOnlyLabels", "CartoDB.DarkMatter", "CartoDB.DarkMatterNoLabels", "CartoDB.DarkMatterOnlyLabels", "CartoDB.Voyager", "CartoDB.VoyagerNoLabels", "CartoDB.VoyagerOnlyLabels",
 "Thunderforest.OpenCycleMap", "Thunderforest.Transport", "Thunderforest.TransportDark", "Thunderforest.SpinalMap", "Thunderforest.Landscape", "Thunderforest.Outdoors", "Thunderforest.Pioneer", "Thunderforest.MobileAtlas", "Thunderforest.Neighbourhood",
 "OpenTopoMap",
 "HikeBike",
 "Wikimedia",

Value

A SpatRaster is returned.

Examples

```

library(sf)
library(maptiles)
nc <- st_read(system.file("shape/nc.shp", package="sf"), quiet = TRUE)
nc_osm <- get_tiles(nc, crop = TRUE, zoom = 6)
plot_tiles(nc_osm)

# Download tiles from OSM, no labels
osmnolbl <- list(
  src = 'osmnolabel',
  q = 'https://{s}.tiles.wmflabs.org/osm-no-labels/{z}/{x}/{y}.png',
  sub = c('a', 'b', 'c'),
  cit = '© OpenStreetMap contributors.'
)
# download tiles and compose raster (SpatRaster)
nc_osmnolbl <- get_tiles(x = nc, provider = osmnolbl, crop = TRUE,
  zoom = 6, verbose = TRUE)

# Plot the tiles
plot_tiles(nc_osmnolbl)

```

maptiles*Download and Display Map Tiles*

Description

To create maps from tiles, `maptiles` downloads, composes and displays tiles from a large number of providers (e.g. OpenStreetMap, Stamen, Esri, CARTO, or Thunderforest).

plot_tiles*Plot map tiles*

Description

Plot a `SpatRaster` object over a map. It can be used to plot tiles.

Usage

```
plot_tiles(x, adjust = FALSE, add = FALSE, ...)
```

Arguments

<code>x</code>	a <code>SpatRaster</code> object.
<code>adjust</code>	if <code>TRUE</code> , plot the raster without zoom-in or zoom-out in the graphic device: add margins if the raster is smaller than the graphic device, zoom-in if the raster is larger than the graphic device.
<code>add</code>	whether to add the layer to an existing plot (<code>TRUE</code>) or not (<code>FALSE</code>).
<code>...</code>	<code>bgalpha</code> , <code>interpolate</code> , or other arguments passed to be passed to <code>plotRGB</code>

Note

This function is a wrapper for [plotRGB](#) from the terra package.

Examples

```
library(sf)
library(maptiles)
nc <- st_read(system.file("shape/nc.shp", package="sf"), quiet = TRUE)
nc_osm <- get_tiles(nc, crop = TRUE)
plot_tiles(nc_osm)
```

Index

`get_credit`, 2
`get_tiles`, 2

`maptiles`, 4

`plot_tiles`, 4
`plotRGB`, 4, 5

`tempdir`, 3