Package 'hwsdr'

May 5, 2025

Title Interface to the 'HWSD' Web Services

Version 1.2

Description Programmatic interface to the Harmonized World Soil Database 'HWSD' web services (<https://daac.ornl.gov/cgi-bin/dsviewer.pl?ds_id=1247>). Allows for easy downloads of 'HWSD' soil data directly to your R workspace or your computer. Routines for both single pixel data downloads and gridded data are provided.

Depends R (>= 4.2)

Imports sf, terra, httr, dplyr, utils

License AGPL-3

LazyData true

ByteCompile true

RoxygenNote 7.3.2

Encoding UTF-8

Suggests ncdf4, knitr, markdown, covr, testthat

VignetteBuilder knitr

URL https://github.com/bluegreen-labs/hwsdr, https://bluegreen-labs.github.io/hwsdr/

BugReports https://github.com/bluegreen-labs/hwsdr/issues

NeedsCompilation no

Author Koen Hufkens [aut, cre] (ORCID: <https://orcid.org/0000-0002-5070-8109>), BlueGreen Labs [cph, fnd]

Maintainer Koen Hufkens <koen.hufkens@gmail.com>

Repository CRAN

Date/Publication 2025-05-05 09:30:02 UTC

Contents

`hwsd2` *HWSD v2.0 database*

Description

Database holding the full HWSD v2.0 database layer information for the main soil type specified. For the fields included (i.e. the column names I refer to the FAO documentation).

Usage

```
hwsd2
```

Format

data.frame

`hwsd_meta_data` *HWSD v1.2 (ORNL DAAC) meta-data*

Description

Data frame with meta-data on the ORNL DAAC parameters one can query using the THREDDS server. In addition a brief description of the various data products and their units is provided.

Usage

```
hwsd_meta_data
```

Format

data.frame

parameter parameter names used in THREDDS server call

subset bands within a data product (only for CLM data)

description general description of the variable

units units of the variable

ws_download *Download HWSD v2.0 data*

Description

Downloads both the database and gridded HWSD v2.0 data products to a desired output path for subsetting.

Usage

```
ws_download(ws_path = file.path(tempdir(), "ws_db"), verbose = FALSE)
```

Arguments

ws_path	the path / directory where to store the HWSD v2.0 database
verbose	verbose messaging of downloading and managing the gridded data file

Details

When an existing path is used which is not the temporary directory an environmental variable WS_PATH can be set by creating an ~/.Renviron file using usethis::edit_r_environ() and entering the path as:

WS_PATH = "/your/full/path"

This variable will override the default temporary directory if it exists. This allows the gridded data to be stored elsewhere and be forgotten (while using the 'hwsdr' package for HWSD v2.0).

Should you delete the gridded file, the environmental variable should be altered and set again by editting the ~/.Renviron file to a new location.

Value

current data path

Examples

```
## Not run:

 # Download the gridded soil map of
 # HWSD v2.0 to the temporary directory
 ws_download()

 # download the same data to a specific
 # directory (which should exist)
 ws_download(
  ws_path = "~/my_path"
 )

 # download the same data to a specific
```

```
 # directory (which should exist) and
 # update the environmental variable
 ws_download(
 ws_path = "~/my_path",
 verbose = TRUE
 )

## End(Not run)
```

`ws_get` *Basic HWSD download function*

Description

Downloads HWSD data, wrapped by `ws_subset()` for convenient use. This is a function mainly for internal use but exposed so people can benefit from it in other (more flexible) setups if so desired.

Usage

```
ws_get(location, param, path, internal = TRUE)
```

Arguments

`location`	file with several site locations and coordinates in a comma delimited format: site, latitude, longitude
`param`	which soil parameter to use
`path`	default is tempdir()
`internal`	return an internal raster or just retain values in the path

Value

HWSD data as a raster file

`ws_subset` *Subset ORNL DAAC HWSD data*

Description

Subset function to query pixel or spatial data from the ORNL DAAC HWSD THREDDS server. Returns a tidy data frame for point locations or raster data to the workspace or disk.

Usage

```
ws_subset(
  location = c(32, -81, 34, -80),
  site = "HWSD",
  param = "ALL",
  layer = "D1",
  path = tempdir(),
  ws_path = file.path(tempdir(), "ws_db"),
  internal = TRUE,
  rate = 0.1,
  version = 1.2,
  verbose = FALSE
)
```

Arguments

location	location of a bounding box c(lon, lat, lon, lat) defined by a bottom-left and top-right coordinates, a single location (lon, lat)
site	sitename for the extracted location
param	soil parameters to provide, the default setting is ALL, this will download all available soil parameters.Check https://daac.ornl.gov/SOILS/guides/HWSD.html for parameter descriptions.
layer	which soil depth layer of HWSD v2.0 to consider, layers are named D1 to D7 from top to bottom
path	path where to download the data to (only applicable to spatial data)
ws_path	path to the gridded HWSD v2.0 data, only required/used if querying v2.0 data
internal	do not store the data on disk
rate	request rate in seconds, determines how long to wait between queries to avoid bouncing because of rate limitations
version	version of HWSD to query (numeric value). By default the package will query the ORNL DAAC v1.2 via their API. If specifying the later version (2.0) it will download or require the gridded spatial data in addition to the included HWSD v2.0 database with soil parameters.
verbose	verbose output during processing, only covers the internal use of the ws_download() function for HWSD v2.0 data

Value

Local geotiff data, or a data frame with HWSD soil information

Examples

```
## Not run:
 # extract sand fraction values
 # for a point location
 values <- ws_subset(
```

```
   site = "HWSD",
   location = c(34, -81),
   param = "T_SAND"
  )

print(values)

# Download a soil fraction map
# of sand for a given bounding box
t_sand <- ws_subset(
   site = "HWSD",
   location = c(32, -81, 34, -80),
   param = "T_SAND",
   path = tempdir(),
   internal = TRUE
  )

terra::plot(t_sand)

## End(Not run)
```

Index