Package: taruwind (via r-universe)

October 22, 2026

Type Package

Title Reduced-Mode Terrain and Surface Wind Downscaling

Version 0.1.1

Description Constructs reusable local wind responses using a neutral terrain-following reduced-vertical-mode adjustment and exact raster-ray integration of finite-fetch surface memory. Explicit metric-grid inputs separate terrain, target aerodynamic roughness and the effective source profile. Prepared responses can be applied to wind-component time series without target-site wind calibration. This diagnostic approximation does not resolve turbulence or guarantee conservation of the final field.

License MIT + file LICENSE

Copyright See inst/COPYRIGHTS for bundled third-party components.

Encoding UTF-8

Depends R (>= 4.2.0)

Imports Rcpp, tibble, curl, jsonlite, parallel

LinkingTo Rcpp

Suggests testthat (>= 3.0.0), terra, Rarr, lutz, ncdf4

Config/testthat/edition 3

SystemRequirements C++17

NeedsCompilation yes

Config/roxygen2/version 8.1.0

Config/pak/sysreqs libssl-dev

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

Date/Publication 2026-10-11 07:55:59 UTC

RemoteUrl https://github.com/cemoze/taruwind

RemoteRef HEAD

RemoteSha 57e52af2c2495bc6b7e1f774d6433acd2cba3cd9

Contents

taruwind_build . . . 2
taruwind_build_many . . . 3
taruwind_cache_info . . . 3
taruwind_download_copernicus_dem . . . 4
taruwind_download_era5 . . . 5
taruwind_download_worldcover . . . 6
taruwind_downscale . . . 7
taruwind_downscale_area . . . 9
taruwind_limits . . . 11
taruwind_modal_predict . . . 11
taruwind_modal_prepare . . . 12
taruwind_neutral_depth . . . 13
taruwind_plot_map . . . 14
taruwind_predict . . . 14
taruwind_prepare_site . . . 15
taruwind_site_layers . . . 16
taruwind_source_profile . . . 17
taruwind_workers . . . 18

Index **19**

`taruwind_build` *Build a current TaruWind response*

Description

This is the current six-mode solver, not the retired spectral/shallow-layer interface. Supply the explicit metric-grid arguments documented below.

Usage

```
taruwind_build(...)
```

Arguments

`...`	Arguments passed to [taruwind_modal_prepare()].

Value

A prepared TaruWind response.

Examples

```
h = matrix(0, 5, 5)
op = taruwind_build(h, h + .03, 90, 0, 0, 180, 180, 900, .03)
taruwind_predict(op, 3, 4)
```

taruwind_build_many *Prepare independent metric-grid sites with portable workers*

Description

Prepare independent metric-grid sites with portable workers

Usage

```
taruwind_build_many(requests, workers = "auto", max_total_cells = 4e+06)
```

Arguments

`requests`	Non-empty list of named argument lists for [taruwind_build()].
`workers`	"auto" by default; see [taruwind_workers()].
`max_total_cells`	Limit on padded cells concurrently in flight. This is an allocation guard, not a measurement of available RAM.

Value

A list of prepared responses in input order.

taruwind_cache_info *Inspect downloaded cache files without changing them*

Description

Reports files and their total size. Files are not removed, opened as data, or uploaded. A caller-selected persistent cache survives R restarts; the default cache is inside the current session's temporary directory.

Usage

```
taruwind_cache_info(cache_dir = file.path(tempdir(), "taruwind-cache"))
```

Arguments

`cache_dir`	Cache root passed to [taruwind_downscale()].

Value

A tibble with relative path, bytes and modification time. Attributes ‘cache_dir‘ and ‘total_bytes‘ identify the root and total file size.

Examples

```
info = taruwind_cache_info()
attr(info, "total_bytes")
```

`taruwind_download_copernicus_dem`

Download Copernicus GLO-30/GLO-90 DEM tiles from the public AWS registry

---

Description

GLO-90 is the default. If explicitly requested, GLO-30 is attempted first and missing tiles fall back to GLO-90. The product is a digital surface model and its licence/attribution remain the user's responsibility. See <https://registry.opendata.aws/copernicus-dem/>. Requests are limited to 16 source tiles; this helper downloads whole DEM tiles and reuses them from cache. It does not silently substitute ocean zero elevations when a source tile is unavailable.

Usage

```
taruwind_download_copernicus_dem(
  bbox,
  target_crs,
  resolution = 90,
  cache_dir = file.path(tempdir(), "taruwind-dem"),
  prefer_glo30 = FALSE,
  output = NULL
)
```

Arguments

`bbox`	Named or unnamed vector ‘c(xmin, ymin, xmax, ymax)‘ in degrees.
`target_crs`	Projected target CRS accepted by ‘terra‘.
`resolution`	Target resolution in metres.
`cache_dir`	Existing or creatable directory for source COG tiles.
`prefer_glo30`	Whether to try public GLO-30 before universal GLO-90.
`output`	Optional projected GeoTIFF path; existing files are not replaced.

Value

A projected ‘terra::SpatRaster‘ with source/citation attributes.

taruwind_download_era5

Download four-node ERA5 wind support and interpolate it to target sites

Description

‘taruwind_download_era5()‘ reads native hourly 10 m or 100 m wind components from the official geo-chunked CDS ARCO store. For every target it retrieves the four surrounding 0.25-degree nodes, performs bilinear interpolation on the selected ‘u‘ and ‘v‘ components separately, and returns a tidy hourly table. Downloaded Zarr metadata and chunks are cached and reused.

Usage

```
taruwind_download_era5(
  sites = NULL,
  lon = NULL,
  lat = NULL,
  start = NULL,
  end = NULL,
  period = NULL,
  api_key = NULL,
  cache_dir = file.path(file.path(tempdir(), "taruwind-cache"), "era5"),
  local_tz = FALSE,
  include_nodes = FALSE,
  refresh = c("auto", "always", "never"),
  quiet = FALSE,
  allow_large = FALSE,
  wind_height = 10L
)
```

Arguments

`sites`	Optional data frame or matrix containing ‘lon‘ and ‘lat‘ columns. ‘longitude‘ and ‘latitude‘ are accepted aliases. An optional ‘site_id‘ column is retained.
`lon, lat`	Numeric longitude and latitude vectors. Use these instead of ‘sites‘; scalar values are allowed.
`start, end`	Start and end of the requested period. Date-only values include the complete UTC day; date-times select exact inclusive hours. Requests before 1940 or after the current time fail immediately. Available recent hours are checked against the live CDS ARCO time axis; no fixed five- or seven-day delay is assumed.
`period`	Optional two-element start/end vector. Supply this instead of separate ‘start‘ and ‘end‘ values.
`api_key`	Optional CDS API key. Prefer ‘CDSAPIKEY‘ or ‘~/.cdsapirc‘ for interactive use so the secret does not appear in scripts.
`cache_dir`	Directory used for the sparse local Zarr cache.

`local_tz`	If ‘FALSE‘ (default), timestamps remain in UTC. If ‘TRUE‘, the time zone is inferred independently for each coordinate with ‘lutz‘ and local ISO timestamps are added. A single Olson/IANA time-zone string can also be supplied explicitly.
`include_nodes`	Include the four raw ERA5 node series in a ‘nodes‘ attribute on the returned tibble.
`refresh`	Refresh mutable ARCO metadata and recent ERA5T chunks automatically once per day (‘"auto"‘), on every call (‘"always"‘), or never (‘"never"‘). Historical cached chunks are immutable and reused.
`quiet`	Suppress download progress messages.
`allow_large`	Bypass the safeguards reported by [taruwind_limits()]. Use this deliberately and split large jobs into recoverable batches.
`wind_height`	ERA5 diagnostic wind height in metres. Supported values are ‘10‘ (default) and ‘100‘. The function retrieves the native ERA5 ‘u10‘/‘v10‘ or ‘u100‘/‘v100‘ fields; it never vertically extrapolates one height from the other.

Details

The Climate Data Store is free to use, but its ARCO endpoint requires a CDS API key. The function checks ‘api_key‘, then ‘CDSAPIKEY‘ and ‘CDS_API_KEY‘, and finally the ‘key:‘ entry in ‘~/.cdsapirc‘. The key is sent only in the HTTPS authorization header and is never written to the cache or result.

Value

A ‘taruwind_era5‘ tibble with hourly vector-bilinear wind at every site. Provenance, interpolation nodes and cache files are stored as attributes.

`taruwind_download_worldcover`
Download WorldCover roughness on a target raster grid

Description

Reads windows from public ESA WorldCover 2021 v200 COGs, samples categorical land cover by nearest neighbour, and applies the WindKit 2.1.0 default WorldCover lookup. This is an empirical land-cover conversion, not a direct roughness measurement. It is not interchangeable with the manuscript’s CORINE-based inputs. Water is floored at 0.0002 m for logarithmic profiles.

Usage

```
taruwind_download_worldcover(
  template,
  cache_dir = file.path(file.path(tempdir(), "taruwind-cache"), "worldcover")
)
```

Arguments

`template`	Single-layer projected SpatRaster defining the target grid.
`cache_dir`	Directory for cached categorical windows.

Value

Roughness SpatRaster in metres, with conversion provenance.

References

ESA WorldCover: <https://esa-worldcover.org/en/data-access>. WindKit roughness tables: <https://docs.wasp.dk/windkit/la

---

`taruwind_downscale` *Download ERA5 and downscale geographic target time series*

Description

Downloads ERA5 and (unless supplied) GLO-90. Roughness and source-profile parameters remain explicit: this function does not equate source roughness with target roughness or assume that ERA5 has one global roughness value. Downloads and raster preparation are serial to avoid cache races. Independent numeric response construction uses [taruwind_build_many()] with its memory guard and portable workers. Prepared responses can then be reused. This is a point workflow, not a national atlas tool.

Usage

```
taruwind_downscale(
  sites = NULL,
  lon = NULL,
  lat = NULL,
  start = NULL,
  end = NULL,
  period = NULL,
  roughness = NULL,
  depth = NULL,
  source_z0 = NULL,
  dem = NULL,
  resolution = 90,
  domain_m = 30000,
  wind_height = 10,
  local_tz = FALSE,
  cache_dir = file.path(tempdir(), "taruwind-cache"),
  api_key = NULL,
  responses = NULL,
  workers = "auto",
  profile_year = 2021,
```

```
  profile_wait_seconds = 60,
  refresh = c("auto", "always", "never"),
  quiet = FALSE
)
```

Arguments

`sites` Optional data frame or matrix containing ‘lon‘ and ‘lat‘ columns. ‘longitude‘ and ‘latitude‘ are accepted aliases. An optional ‘site_id‘ column is retained.

`lon, lat` Numeric longitude and latitude vectors. Use these instead of ‘sites‘; scalar values are allowed.

`start, end` Start and end of the requested period. Date-only values include the complete UTC day; date-times select exact inclusive hours. Requests before 1940 or after the current time fail immediately. Available recent hours are checked against the live CDS ARCO time axis; no fixed five- or seven-day delay is assumed.

`period` Optional two-element start/end vector. Supply this instead of separate ‘start‘ and ‘end‘ values.

`roughness` Raster or filename shared by all sites, or a list of one raster/filename per site, or "worldcover" for automatic acquisition. Raster values must be aerodynamic roughness in metres.

`depth, source_z0` Scalar or one value per site, in metres. If source_z0 is NULL, [taruwind_source_profile()] retrieves the 2021 reference FSR and applies the documented diagnostic convention. If depth is NULL it is computed by [taruwind_neutral_depth()]. Explicit inputs override automation.

`dem` NULL for automatic GLO-90, one raster/filename, or a per-site list.

`resolution, domain_m` Grid spacing and local domain width in metres.

`wind_height` ERA5 diagnostic wind height in metres. Supported values are ‘10‘ (default) and ‘100‘. The function retrieves the native ERA5 ‘u10‘/‘v10‘ or ‘u100‘/‘v100‘ fields; it never vertically extrapolates one height from the other.

`local_tz` If ‘FALSE‘ (default), timestamps remain in UTC. If ‘TRUE‘, the time zone is inferred independently for each coordinate with ‘lutz‘ and local ISO timestamps are added. A single Olson/IANA time-zone string can also be supplied explicitly.

`cache_dir` Directory used for the sparse local Zarr cache.

`api_key` Optional CDS API key. Prefer ‘CDSAPIKEY‘ or ‘~/.cdsapirc‘ for interactive use so the secret does not appear in scripts.

`responses` Optional named list of prepared geographic responses from a previous call, indexed by site_id. Coordinates and heights are checked.

`workers` "auto" (half the detected cores, capped at 16 and site count), or an integer from 1 to 16; the memory guard may reduce concurrency.

`profile_year` Reference FSR climatology year; default 2021, not inferred from the wind period. See [taruwind_source_profile()].

`profile_wait_seconds`
: CDS queue polling limit per site/call.

`refresh`
: Refresh mutable ARCO metadata and recent ERA5T chunks automatically once per day (‘"auto"‘), on every call (‘"always"‘), or never (‘"never"‘). Historical cached chunks are immutable and reused.

`quiet`
: Suppress download progress messages.

Value

A tibble (also a data frame) with one row per site-hour. ‘wind_speed‘ is vector-bilinear ERA5 speed; ‘taruwind_speed‘ is the downscaled speed. ‘attr(result, "responses")‘ is a named list of prepared responses, one per site. ‘attr(result, "forcing_provenance")‘ records ERA5 source and interpolation metadata. These two attributes are separate from the printed columns. The current ERA5 download metadata is also forwarded as attributes, including ‘sites‘, ‘source‘, ‘license‘, ‘citation‘, ‘retrieved_utc‘, ‘wind_height_m‘, ‘component_variables‘, ‘interpolation‘ and ‘cache_dir‘. Inspect ‘names(attributes(result))‘ for the complete set. ‘saveRDS()‘ retains attributes; CSV does not.

`taruwind_downscale_area`

Downscale a small local grid of target points

Description

Each output cell centre receives its own complete TaruWind response. This produces a map of point estimates on a regular metric grid; it is not a single shared-domain flow solve. Work therefore grows with cell count. The square output grid is centred on ‘lon, lat‘ and rounded to an odd number of cells so that one cell represents the centre exactly. Its spacing (‘map_resolution‘) is independent of the terrain grid spacing (‘resolution‘). A call may cover at most 168 hourly timestamps (seven complete UTC days). For longer periods, make consecutive calls and reuse the prepared responses from ‘attr(previous$wind, "responses")‘.

Usage

```
taruwind_downscale_area(
  lon,
  lat,
  start = NULL,
  end = NULL,
  period = NULL,
  area_m = 990,
  map_resolution = 500,
  max_cells = 25,
  roughness = "worldcover",
  depth = NULL,
  source_z0 = NULL,
  dem = NULL,
  resolution = 90,
```

```
  domain_m = 30000,
  wind_height = 10,
  cache_dir = file.path(tempdir(), "taruwind-cache"),
  api_key = NULL,
  workers = "auto",
  responses = NULL,
  profile_year = 2021,
  profile_wait_seconds = 60,
  refresh = c("auto", "always", "never"),
  quiet = FALSE
)
```

Arguments

`lon`, `lat` Centre coordinates in decimal degrees.

`start`, `end`, `period` UTC period, as in [taruwind_downscale()].

`area_m` Requested output-map width in metres.

`map_resolution` Output-cell spacing in metres; does not change the terrain solver's grid resolution.

`max_cells` Maximum output cells, 1 to 100; defaults to 25. Raise this only when the cost of one response per cell is acceptable.

`roughness` Raster or filename shared by all sites, or a list of one raster/filename per site, or "worldcover" for automatic acquisition. Raster values must be aerodynamic roughness in metres.

`depth`, `source_z0` Scalar or one value per site, in metres. If source_z0 is NULL, [taruwind_source_profile()] retrieves the 2021 reference FSR and applies the documented diagnostic convention. If depth is NULL it is computed by [taruwind_neutral_depth()]. Explicit inputs override automation.

`dem` NULL for automatic GLO-90, one raster/filename, or a per-site list.

`resolution`, `domain_m` Grid spacing and local domain width in metres.

`wind_height` ERA5 diagnostic wind height in metres. Supported values are '10' (default) and '100'. The function retrieves the native ERA5 'u10'/'v10' or 'u100'/'v100' fields; it never vertically extrapolates one height from the other.

`cache_dir` Directory used for the sparse local Zarr cache.

`api_key` Optional CDS API key. Prefer 'CDSAPIKEY' or '~/.cdsapirc' for interactive use so the secret does not appear in scripts.

`workers` "auto" (half the detected cores, capped at 16 and site count), or an integer from 1 to 16; the memory guard may reduce concurrency.

`responses` Optional named list of prepared geographic responses from a previous call, indexed by site_id. Coordinates and heights are checked.

`profile_year` Reference FSR climatology year; default 2021, not inferred from the wind period. See [taruwind_source_profile()].

`profile_wait_seconds`
: CDS queue polling limit per site/call.

`refresh`
: Refresh mutable ARCO metadata and recent ERA5T chunks automatically once per day (‘"auto"‘), on every call (‘"always"‘), or never (‘"never"‘). Historical cached chunks are immutable and reused.

`quiet`
: Suppress download progress messages.

Value

A list with ‘wind‘ (site-hour tibble), ‘maps‘ (time-labelled ‘terra::SpatRaster‘ layers for TaruWind and vector-bilinear ERA5), and ‘grid‘ (output grid and coordinate provenance). Use [taruwind_plot_map()] to display an individual map layer.

`taruwind_limits` *Automatic-workflow safety limits*

Description

TaruWind's automatic download workflow is intended for point networks and local terrain domains, not accidental continental raster production. These conservative limits are checked before network or DEM work starts. Advanced users of [taruwind_download_era5()] can explicitly opt out with ‘allow_large = TRUE‘. The main [taruwind_downscale()] workflow does not expose this override; split larger requests into recoverable batches. A point or raw ERA5 request is limited to ten calendar years per call. Local area maps are limited to 168 hourly timestamps (seven days). Longer series can be built in batches with reusable prepared responses. Terrain-cell limits count the reflected computational grid, not just the output square. They do not estimate network bytes or guarantee that a large request will fit in memory.

Usage

```
taruwind_limits()
```

Value

A named list containing the default limits.

`taruwind_modal_predict` *Apply a current reduced-mode response to target-interpolated wind*

Description

Apply a current reduced-mode response to target-interpolated wind

Usage

```
taruwind_modal_predict(object, u, v)
```

Arguments

`object`	Response from [taruwind_modal_prepare()].
`u, v`	Matching numeric component vectors in the grid coordinate frame, already vector-interpolated to the target at the prepared height.

Value

A tibble of wind speeds in metres per second.

---

taruwind_modal_prepare

Prepare the current reduced-mode terrain and surface-memory response

---

Description

Low-level metric-grid interface to the manuscript formulation. Grid rows run south to north. Source roughness is an explicit effective input, not inferred from target land cover. This interface does not download or reproject data. Heights other than 10 m are supported numerically, not validated universally.

Usage

```
taruwind_modal_prepare(
  dem,
  roughness,
  dx,
  x0,
  y0,
  x,
  y,
  depth,
  source_z0,
  height = 10,
  profile_z0 = 0.03,
  min_scale = 10,
  reflection = TRUE,
  max_cells = 2e+06
)
```

Arguments

`dem`, `roughness`	Finite numeric matrices on one square-cell metric grid. Roughness is aerodynamic length in metres, not land-cover class codes.
`dx`	Cell spacing in metres.
`x0`, `y0`	Coordinates of the southwest cell centre.
`x`, `y`	Target coordinates, in the same metric coordinate system.
`depth`	Positive neutral reference depth, in metres; supplied explicitly.
`source_z0`	Effective source-profile roughness, in metres.
`height`	Evaluation height above the local surface, in metres.
`profile_z0`	Terrain-profile reference roughness, in metres.
`min_scale`	Smallest vertical mode scale, in metres.
`reflection`	Reflect terrain by one domain on each side.
`max_cells`	Maximum cells after reflection, checked before allocation.

Value

A prepared response with a 2-by-2 terrain matrix, 72 directional surface factors and solver diagnostics. Construction is single-threaded.

`taruwind_neutral_depth` *Compute the model's neutral modal depth from latitude*

Description

Compute the model's neutral modal depth from latitude

Usage

```
taruwind_neutral_depth(lat)
```

Arguments

`lat`	Numeric latitude(s) in degrees. Automatic use is restricted to absolute latitudes 5 through 85 degrees; provide explicit depth elsewhere.

Value

Neutral depth in metres, using the study convention '0.2 * (0.4 * 10 / log(10 / 0.01)) / abs(1.4544e-4 * sin(latitude))'. This is a fixed numerical scale, not measured or hourly ERA5 BLH.

taruwind_plot_map *Plot aligned terrain or an hourly wind map*

Description

Uses base ‘terra‘ graphics. Pass the two-layer output of [taruwind_site_layers()] or the ‘maps‘ component of [taruwind_downscale_area()]. Layer names are available with ‘names(x)‘. When ‘layer‘ is omitted, all supplied layers are drawn.

Usage

```
taruwind_plot_map(x, layer = NULL, file = NULL, width = 1200, height = 700)
```

Arguments

`x`	A ‘terra::SpatRaster‘ of aligned site layers or wind maps.
`layer`	One or more layer names or indices; all by default.
`file`	Optional PNG filename. An existing file is not overwritten.
`width, height`	PNG dimensions in pixels when ‘file‘ is supplied.

Value

The selected ‘terra::SpatRaster‘, invisibly; draws the plot.

taruwind_predict *Predict wind speed from a prepared response*

Description

Predict wind speed from a prepared response

Usage

```
taruwind_predict(object, u, v, weights = NULL)
```

Arguments

`object`	A response from [taruwind_build()].
`u, v`	Target-interpolated component vectors, or matching matrices with four columns for the four source nodes. Components must use the grid frame.
`weights`	Four non-negative interpolation weights summing to one, required for matrix input. The same weights apply to every row.

Value

A tibble with a speed column in metres per second.

taruwind_prepare_site *Prepare a geographic target using raster terrain and roughness*

Description

A local azimuthal-equidistant grid is centred on the target, so its axes at the target coincide with east/north wind components. Raster rows are reversed explicitly for the native solver. No missing cells are filled implicitly.

Usage

```
taruwind_prepare_site(
  lon,
  lat,
  roughness,
  depth,
  source_z0,
  dem = NULL,
  resolution = 90,
  domain_m = 30000,
  height = 10,
  cache_dir = file.path(file.path(tempdir(), "taruwind-cache"), "dem"),
  quiet = TRUE
)
```

Arguments

`lon`, `lat`	One target longitude and latitude in degrees.
`roughness`	Aerodynamic roughness SpatRaster or raster filename, in metres, or the explicit string "worldcover" for automatic acquisition. Land-cover codes must otherwise be converted before calling this function.
`depth`, `source_z0`	Explicit neutral reference depth and effective source roughness in metres; these are not inferred from target land cover.
`dem`	Optional elevation SpatRaster or filename. If NULL, GLO-90 is downloaded with [taruwind_download_copernicus_dem()].
`resolution`	Metric calculation spacing, default 90 metres.
`domain_m`	Requested square domain width in metres, rounded upwards to an odd number of cells with the target at the central cell centre.
`height`	Evaluation height, 10 or 100 metres.
`cache_dir`	Directory for downloaded DEM tiles.
`quiet`	Suppress preparation messages.

Value

A prepared response usable with [taruwind_predict()], with geographic input provenance. Caller-provided rasters must have a valid CRS.

---

taruwind_site_layers *Retrieve the terrain layers used for one geographic site*

Description

Returns the GLO-90 elevation and aerodynamic roughness after alignment to the same target-centred metric grid used by [taruwind_prepare_site()]. The layers can be plotted with [taruwind_plot_map()]. No wind is downloaded or response constructed. WorldCover roughness is a land-cover lookup, not a measured roughness map.

Usage

```
taruwind_site_layers(
  lon,
  lat,
  roughness = "worldcover",
  dem = NULL,
  resolution = 90,
  domain_m = 30000,
  cache_dir = file.path(file.path(tempdir(), "taruwind-cache"), "dem"),
  quiet = TRUE
)
```

Arguments

`lon, lat`	One target longitude and latitude in degrees.
`roughness`	Aerodynamic roughness SpatRaster or raster filename, in metres, or the explicit string "worldcover" for automatic acquisition. Land-cover codes must otherwise be converted before calling this function.
`dem`	Optional elevation SpatRaster or filename. If NULL, GLO-90 is downloaded with [taruwind_download_copernicus_dem()].
`resolution`	Metric calculation spacing, default 90 metres.
`domain_m`	Requested square domain width in metres, rounded upwards to an odd number of cells with the target at the central cell centre.
`cache_dir`	Directory for downloaded DEM tiles.
`quiet`	Suppress preparation messages.

Value

A two-layer ‘terra::SpatRaster‘ named ‘elevation_m‘ and ‘roughness_m‘.

taruwind_source_profile

Obtain the explicit ERA5 source-profile convention

Description

Retrieves the four-node ERA5 forecast surface roughness (FSR) climatology from CDS monthly means, averages the 12 months and 24 hours equally and bilinearly interpolates the result. The default reference year is 2021, matching the study input convention, not necessarily the requested wind year. At 10 m the effective diagnostic reference is bounded to [0.0002, 0.03] m. This cap is a source-profile convention, not a claim that ERA5's actual roughness is globally 0.03 m. At 100 m the raw FSR is retained with a 0.0002 m floor. The surface multiplier is unity at the 100 m matching reference.

Usage

```
taruwind_source_profile(
  lon,
  lat,
  wind_height = 10,
  reference_year = 2021,
  fsr_file = NULL,
  api_key = NULL,
 cache_dir = file.path(file.path(tempdir(), "taruwind-cache"), "source_profile"),
  wait_seconds = 60,
  quiet = TRUE
)
```

Arguments

`lon`, `lat`	One target coordinate in degrees.
`wind_height`	10 or 100 metres.
`reference_year`	FSR climatology year, default 2021.
`fsr_file`	Optional existing NetCDF containing ‘fsr‘, longitude, latitude and optional time dimensions. Its actual averaging period is the caller's responsibility and is not inferred from ‘reference_year‘.
`api_key`	CDS key; otherwise standard environment/config discovery.
`cache_dir`	Local request, job and NetCDF cache directory.
`wait_seconds`	Maximum queue polling time per call, default 60 seconds.
`quiet`	Suppress cache and CDS queue messages.

Details

CDS credentials and dataset terms acceptance are required. A queued job is cached and resumed on the next call; timeout does not submit duplicate jobs. No Python, keyring or observations are needed. Existing FSR NetCDF files may be supplied for offline use. REST access is subject to CDS service changes.

Value

A list containing raw/effective roughness, neutral depth and provenance.

References

<https://cds.climate.copernicus.eu/how-to-api>

---

taruwind_workers *Resolve portable batch concurrency*

Description

Resolve portable batch concurrency

Usage

```
taruwind_workers(
  workers = "auto",
  n_sites = 1L,
  cores = parallel::detectCores()
)
```

Arguments

workers	"auto", or an integer from 1 to 16.
n_sites	Number of independent requests.
cores	Detected logical cores; injectable for reproducible policy tests.

Value

An integer worker count. Auto uses half the cores, rounded down, bounded below by one and above by 16 and the request count.

Index

taruwind_build, 2
taruwind_build_many, 3
taruwind_cache_info, 3
taruwind_download_copernicus_dem, 4
taruwind_download_era5, 5
taruwind_download_worldcover, 6
taruwind_downscale, 7
taruwind_downscale_area, 9
taruwind_limits, 11
taruwind_modal_predict, 11
taruwind_modal_prepare, 12
taruwind_neutral_depth, 13
taruwind_plot_map, 14
taruwind_predict, 14
taruwind_prepare_site, 15
taruwind_site_layers, 16
taruwind_source_profile, 17
taruwind_workers, 18