Package: hydrodownloadR (via r-universe)

June 5, 2026

Title Hydrologic Station Catalogs and Time Series from Public APIs

Version 0.1.3

Description Provides a unified, extensible interface to discover hydrologic stations and download daily time series (e.g., water discharge, water level, water temperature, and several other water quality parameter) from national and regional public APIs. Includes a provider registry, S3 generics 'stations' and 'timeseries', licensing metadata, date-range and 'complete history' modes, rate limiting and retries, optional authentication via environment variables, tidy outputs, UTF-8 to ASCII transliteration, and WGS84 coordinates. Designed for reproducible workflows and straightforward addition of new providers. Background and use cases are described in Farber et al. (2025) <doi:10.5194/essd-17-4613-2025> and Farber et al. (2023) <doi:10.57757/IUGG23-2838>.

License MIT + file LICENSE

URL https://bafg-bund.github.io/hydrodownloadR/, https://github.com/bafg-bund/hydrodownloadR

BugReports https://github.com/bafg-bund/hydrodownloadR/issues

Depends R (>= 4.1.0)

Imports cli, dataRetrieval, DBI, dplyr, httr, httr2, jsonlite, lubridate, pdftools, progress, rappdirs, ratelimitr, rlang, RSQLite, sf, magrittr, tibble, cellranger, stringi, stringr

Suggests odbc, purrr, readr, readxl, rvest, tidyr, tidyselect, xml2, cachem, curl, memoise, testthat (>= 3.0.0), writexl

Config/testthat/edition 3

Encoding UTF-8

Language en-US

LazyData true

LazyDataCompression xz

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.1

Config/pak/sysreqs
libabsl-dev cmake libgdal-dev gdal-bin libgeos-dev libicu-dev libjpeg-dev libxml2-dev libssl-dev libpoppler-cpp-dev poppler-data libproj-dev libsqlite3-dev libudunits2-dev libx11-dev

Repository https://bafg-bund.r-universe.dev

Date/Publication 2026-05-06 11:01:41 UTC

RemoteUrl https://github.com/bafg-bund/hydrodownloadr

RemoteRef HEAD

RemoteSha c8b2b12d680cb944f365040ffc003ec64831fb09

Contents

`fi_syke_runoff_meta` *SYKE runoff station metadata (area & altitude)*

Description

Catchment area and altitude for Finnish SYKE runoff stations. Area may be `NA` for a few stations; altitude may still be present. Used to compute discharge from runoff time series: `discharge_m3s = (value_lps_per_km2 * area_km2) / 1000`.

Usage

```
data(fi_syke_runoff_meta)
```

Format

A tibble with:

place_id Character. SYKE Paikka_Id.

area Numeric (km^2). May be NA.

altitude Numeric (m). May be NA.

Source

Finnish Environment Institute (SYKE).

`fr_hubeau_meta` *FR_HUBEAU precomputed station metadata*

Description

Preloaded metadata for Hub'Eau stations, built offline from the Hub'Eau referentials plus scraped site/station fiches (area, site altitude, gauge-zero altitude, vertical datum at site). Used to speed up `stations()` for the FR_HUBEAU provider.

Format

A data frame/tibble with columns:

code_site Character. Hub'Eau site code.

station_id Character. Hub'Eau station code.

area Numeric (km^2). Catchment area from site fiche; may be `NA`.

altitude_api Numeric (m). API referential altitude (hydrometry in mm to m; temperature in m).

altitude_site Numeric (m). Site altitude parsed from the site fiche; may be `NA`.

altitude_station Numeric (m). "Cote du zero d'echelle" from station fiche; may be `NA`.

vertical_datum_site Character. Site-level vertical datum label; may be `NA`.

retrieved_at POSIXct (UTC). Timestamp when the row was scraped.

Details

Built by `data-raw/fr_hubeau_meta_build.R`. The file `data/fr_hubeau_meta.rda` is shipped with the package and may be refreshed out-of-band. A build-date string is also stored in the object attribute `metadata_date`.

Source

Hub'Eau APIs and https://www.hydro.eaufrance.fr/ site/station fiches.

hydro_service *Create a hydro service object*

Description

Create a hydro service object

Usage

```
hydro_service(provider_id, ...)
```

Arguments

provider_id	ID as listed by hydro_services()
...	Reserved for future use.

Value

An object of class "hydro_service" (a list) containing the provider configuration used by stations() and timeseries() (e.g. provider_id, provider_name, country, base_url, and other adapter-specific settings).

hydro_services *List available providers*

Description

List available providers

Usage

```
hydro_services()
```

Value

A tibble with columns: provider_id, provider_name, country, base_url, license, license_link, access_class, reuse_class, is_open_data

jp_mlit_meta *Japan MLIT stations metadata snapshot*

Description

A tibble used by the JP MLIT adapter to speed up station discovery.

Usage

```
data(jp_mlit_meta)
```

Format

A tibble/data.frame with one row per station and typical columns:

station_id MLIT station identifier (character)

station_name Station name (character)

river River name, if available (character)

lat Latitude in WGS84 (double)

lon Longitude in WGS84 (double)

area_km2 Drainage area in km^2, if available (double)

altitude_m Altitude in meters, if available (double)

country ISO country code (character)

provider_id Adapter provider id, e.g. `"JP_MLIT"` (character)

provider_name Provider name (character)

Source

MLIT; see package README for licensing.

list_countries *List available countries*

Description

List available countries

Usage

```
list_countries()
```

Value

A character vector of country codes (e.g. ISO 3166-1 alpha-2) for which at least one provider is available.

stations *List stations for a provider*

Description

List stations for a provider

Usage

```
stations(x, ...)
```

Arguments

x	A hydro_service object created by hydro_service().
...	Passed to provider-specific methods.

Value

A tibble with station metadata.

Examples

```
# Offline: enumerate providers (no network)
s <- hydro_services()
head(names(s))


# Online (opt-in): fetch stations
x <- hydro_service("SE_SMHI")
st <- stations(x)
head(st)
```

stations.hydro_service_BR_ANA *ANA stations (Brazil) - cache-first with optional update from inventory*

Description

Loads the cached ANA station catalogue (if present) or rebuilds it from a locally downloaded SNIRH inventory (InventarioDD_MM_YYYY.zip / .mdb) when update = TRUE.

Usage

```
## S3 method for class 'hydro_service_BR_ANA'
stations(x, ...)
```

Arguments

x	A hydro_service created with hydro_service("BR_ANA").
...	Named arguments: • zip_or_mdb: path to InventarioDD_MM_YYYY.zip or .mdb • dest_dir: unzip destination (default: "data-raw/BR_ANA") • cache_dir: cache dir for RDS (default: user cache) • update: TRUE to rebuild from provided inventory

Value

A tibble with ANA station metadata.

---

timeseries *Retrieve time series for a provider*

Description

Retrieve time series for a provider

Usage

```
timeseries(
  x,
  parameter,
  stations = NULL,
  start_date = NULL,
  end_date = NULL,
  mode = c("range", "complete"),
  ...
)
```

Arguments

x	A hydro_service object created by hydro_service().
parameter	One of "water_discharge","water_level","water_temperature","water_velocity".
stations	Optional character vector of station IDs.
start_date, end_date	YYYY-MM-DD strings for mode = "range".
mode	Either "range" or "complete" (1900-01-01 to today).
...	Passed to provider-specific methods.

Value

A tibble with columns: country, provider_id, provider_name, station_id, parameter, timestamp, value, unit, quality_code, source_url.

Examples

```
# Offline: construct a service object (no network)
x <- hydro_service("SE_SMHI")


# Online (opt-in): one station for a short range
st <- head(stations(x)$station_id, 1)
ts <- timeseries(x, parameter = "water_discharge",
                 stations = st,
                 start_date = "2020-01-01", end_date = "2020-01-10")
head(ts)
```

`timeseries_parameters` *List supported parameters/units for a provider*

Description

List supported parameters/units for a provider

Usage

```
timeseries_parameters(x, ...)
```

Arguments

`x`	A `hydro_service` object.
`...`	Reserved for future use.

Value

A tibble with columns: parameter, code, unit.

Index