Package: rCoros (via r-universe)

June 25, 2026

Title Access COROS Training Hub Fitness Data

Version 0.1.0

Date 2026-06-07

Description Provides a tidy interface to the 'COROS' Training Hub API (<https://coros.com/traininghub>), the web platform that accompanies 'COROS' GPS sports watches. Retrieves activities, daily wellness metrics (heart rate variability, resting heart rate, VO2 max and training load), workout programmes and training calendars. All results are returned as tibbles, ready for analysis with 'dplyr' and 'ggplot2'. Both the US and EU regional endpoints are supported.

License MIT + file LICENSE

Encoding UTF-8

Roxygen list(markdown = TRUE)

Depends R (>= 4.1.0)

Imports digest (>= 0.6.30), dplyr (>= 1.1.0), httr2 (>= 1.0.0), purrr (>= 1.0.0), tibble (>= 3.2.0)

Suggests bslib (>= 0.6.0), DT (>= 0.28), ggplot2 (>= 3.4.0), knitr (>= 1.40), plotly (>= 4.10.0), rmarkdown (>= 2.20), shiny (>= 1.8.0), testthat (>= 3.0.0), withr (>= 2.5.0)

VignetteBuilder knitr

URL https://github.com/mattyoreilly/rCoros, https://mattyoreilly.github.io/rCoros/

BugReports https://github.com/mattyoreilly/rCoros/issues

Config/roxygen2/version 8.0.0

Config/testthat/edition 3

Config/pak/sysreqs libssl-dev

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

Date/Publication 2026-06-25 12:15:09 UTC

RemoteUrl https://github.com/mattyoreilly/rcoros

RemoteRef HEAD

RemoteSha bdbff0a430b4eca19565baada2dee4829c3e8aa0

Contents

`coros_activities` *List activities*

Description

Returns a tidy tibble of activities recorded within a date range, one row per activity.

Usage

```
coros_activities(
  auth,
  start_day = format(Sys.Date() - 30, "%Y%m%d"),
  end_day = format(Sys.Date(), "%Y%m%d"),
  page = 1L,
  size = 30L,
  n_max = Inf
)
```

Arguments

`auth`	A `coros_auth` object from `coros_login()`.
`start_day`	Start of date range in `"YYYYMMDD"` format. Defaults to 30 days ago.
`end_day`	End of date range in `"YYYYMMDD"` format. Defaults to today.
`page`	Page number for paginated results (default `1L`).
`size`	Number of results per page (default `30L`).
`n_max`	Maximum total activities to return. Set to `Inf` to fetch all pages automatically (default `Inf`).

Value

A `tibble::tibble()` with columns:

activity_id Unique activity identifier (character).

name Activity name or remark.

sport_type Numeric sport type code.

sport_name Human-readable sport name.

date Date of activity (`Date`).

start_time Start timestamp (`POSIXct`, UTC).

duration_s Duration in seconds.

duration_min Duration in minutes.

distance_m Distance in metres.

distance_km Distance in kilometres.

elevation_gain Elevation gain in metres.

avg_hr Average heart rate (bpm).

calories Calories (kcal).

training_load Training load score.

avg_power Average power (watts).

device Device name.

Examples

```
auth <- coros_login()

# All activities in the last 30 days
acts <- coros_activities(auth)

# Running and trail-running only
library(dplyr)
runs <- coros_activities(auth) |>
  filter(sport_type %in% c(100L, 102L))
```

---

coros_activity_detail *Fetch detailed metrics for a single activity*

Description

Returns a list of three tibbles — a one-row summary, per-lap splits, and time-in-zone heart rate data — for the given activity.

Usage

```
coros_activity_detail(auth, activity_id, sport_type)
```

Arguments

auth	A coros_auth object from coros_login().
activity_id	Activity identifier (from coros_activities() activity_id column).
sport_type	Numeric sport type code (from coros_activities() sport_type column).

Value

A named list with three tibbles:

summary One-row tibble with overall activity metrics.

laps One row per lap with splits.

hr_zones Heart-rate zone breakdown (seconds and percent).

Examples

```
auth <- coros_login()
acts <- coros_activities(auth)

# Detail for the most recent activity
detail <- coros_activity_detail(
  auth,
  activity_id = acts$activity_id[[1]],
  sport_type  = acts$sport_type[[1]]
)
detail$summary
detail$laps
detail$hr_zones
```

---

coros_daily_metrics *Fetch daily health and training metrics*

Description

Returns a tidy tibble of per-day wellness metrics from the COROS /analyse/dayDetail/query endpoint, including HRV, resting heart rate, training load, VO2max, and stamina.

Usage

```
coros_daily_metrics(
  auth,
  start_day = format(Sys.Date() - 28, "%Y%m%d"),
  end_day = format(Sys.Date(), "%Y%m%d")
)
```

Arguments

auth	A coros_auth object from coros_login().
start_day	Start of date range in "YYYYMMDD" format. Defaults to 28 days ago.
end_day	End of date range in "YYYYMMDD" format. Defaults to today.

Value

A tibble::tibble() sorted by date with columns:

date Calendar date (Date).

hrv Average overnight HRV (ms).

hrv_baseline Personal HRV baseline (ms).

rhr Resting heart rate (bpm).

training_load Daily training load.

load_ratio Training load ratio (acute:chronic).

tired_rate Fatigue rate.

ati Acute training impulse.

cti Chronic training impulse.

t7d 7-day training load.

t28d 28-day training load.

vo2max Estimated VO2max (mL/kg/min).

lthr Lactate threshold heart rate (bpm).

ltsp Lactate threshold speed.

stamina Current stamina level.

stamina_7d 7-day stamina level.

performance Performance score.

tib Time in bed (minutes).

Examples

```
auth <- coros_login()

# Last 28 days (default)
metrics <- coros_daily_metrics(auth)

# Custom range
metrics <- coros_daily_metrics(auth, start_day = "20240101", end_day = "20240131")
```

coros_hrv *Fetch recent HRV readings*

Description

Retrieves the last ~7 days of overnight HRV data from the COROS dashboard endpoint.

Usage

```
coros_hrv(auth)
```

Arguments

auth	A coros_auth object from coros_login().

Value

A tibble::tibble() sorted by date with columns:

date Calendar date (Date).

hrv Average overnight HRV (ms).

baseline Personal HRV baseline (ms).

hrv_sd Standard deviation of overnight HRV (ms).

See Also

coros_daily_metrics() for a longer historical HRV series.

Examples

```
auth <- coros_login()
coros_hrv(auth)
```

coros_login *Authenticate with the COROS Training Hub API*

Description

Logs in with an email/password pair and returns an auth object that must be passed to every other coros_* function. Credentials are read from environment variables by default so they are never hard-coded in scripts.

Usage

```
coros_login(
  email = Sys.getenv("COROS_EMAIL"),
  password = Sys.getenv("COROS_PASSWORD"),
  region = c("us", "eu")
)
```

Arguments

email	COROS account e-mail. Defaults to the COROS_EMAIL environment variable.
password	COROS account password. Defaults to COROS_PASSWORD.
region	API region: "us" (default) or "eu".

Details

Set credentials once per session with:

```
Sys.setenv(COROS_EMAIL = "you@example.com", COROS_PASSWORD = "secret")
```

or add them to your ~/.Renviron file for persistence.

Value

A named list with fields access_token, user_id, base_url, region, and timestamp. Treat this object as opaque and pass it directly to other coros_* functions.

Examples

```
auth <- coros_login()  # reads COROS_EMAIL / COROS_PASSWORD from env

# EU region
auth_eu <- coros_login(region = "eu")
```

---

coros_schedule *Fetch the training calendar*

Description

Returns a tibble of planned activities from the COROS training schedule within the given date window.

Usage

```
coros_schedule(
  auth,
  start_day = format(Sys.Date(), "%Y%m%d"),
  end_day = format(Sys.Date() + 14, "%Y%m%d")
)
```

Arguments

auth	A coros_auth object from coros_login().
start_day	Start of the window in "YYYYMMDD" format. Defaults to today.
end_day	End of the window in "YYYYMMDD" format. Defaults to 14 days from today.

Value

A tibble::tibble() with one row per scheduled item and columns:

plan_id Training plan identifier.

id_in_plan Item position within the plan.

plan_program_id Associated workout program identifier.

happen_day Scheduled date (Date).

name Workout name.

sport_type Numeric sport type code.

sport_name Human-readable sport name.

estimated_min Estimated duration in minutes.

completed Logical; TRUE if the workout has been completed.

Examples

```
auth <- coros_login()

# Upcoming two weeks
schedule <- coros_schedule(auth)

# Narrower window
schedule <- coros_schedule(
  auth,
  start_day = format(Sys.Date(), "%Y%m%d"),
  end_day   = format(Sys.Date() + 7, "%Y%m%d")
)
```

---

coros_workouts *List structured workout programs*

Description

Retrieves all workout programs stored in the COROS Training Hub, returning a list of two tidy tibbles: a summary of each workout and its constituent steps.

Usage

```
coros_workouts(auth)
```

Arguments

auth	A coros_auth object from coros_login().

Value

A named list with two tibbles:

workouts One row per workout with columns id, name, sport_type, sport_name, duration_min, and n_steps.

steps One row per step, linked to workouts via workout_id, with columns step_name, duration_s, duration_min, power_low_w, power_high_w, and sets.

Examples

```
auth <- coros_login()
result <- coros_workouts(auth)
result$workouts
result$steps
```

Index