

Package: unirateapi (via r-universe)

October 4, 2026

Title Client for the 'UniRate' Currency Exchange Rate API

Version 0.1.0

Description Access free real-time and historical currency exchange rates, currency conversion, and value-added tax (VAT) rates from the 'UniRate' API <<https://unirateapi.com>>. Supports current rates between 170+ fiat and crypto currencies, historical rates and time series, currency conversion, and per-country VAT rates. Requires a free API key.

License MIT + file LICENSE

Encoding UTF-8

URL <https://unirateapi.com>,
<https://github.com/UniRate-API/unirate-api-r>

BugReports <https://github.com/UniRate-API/unirate-api-r/issues>

Imports curl, jsonlite, utils

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

RoxygenNote 7.3.2

Config/pak/sysreqs libssl-dev

Repository <https://unirate-api.r-universe.dev>

Date/Publication 2026-10-04 05:23:50 UTC

RemoteUrl <https://github.com/UniRate-API/unirate-api-r>

RemoteRef HEAD

RemoteSha 63f146c355fd48bd00b7e065b40000e62c8f8dc3

Contents

unirate_client	2
Index	4

unirate_client

*Create a UniRate API client***Description**

Constructs a client for the UniRate currency exchange rate API. The returned object is a list of functions (methods); call them with \$, e.g. `client$get_rate("USD", "EUR")`.

Usage

```
unirate_client(
  api_key,
  base_url = "https://api.unirateapi.com",
  timeout = 30,
  http_fetch = NULL
)
```

Arguments

<code>api_key</code>	Your UniRate API key. Get a free key at https://unirateapi.com .
<code>base_url</code>	API base URL. Defaults to "https://api.unirateapi.com".
<code>timeout</code>	Request timeout in seconds. Defaults to 30.
<code>http_fetch</code>	Optional fetch function for dependency injection (mainly for testing). It receives a single URL string and must return <code>list(status = <integer>, body = <character>)</code> . Defaults to a curl-backed fetcher.

Value

A `unirate_client` object exposing the following methods:

`get_rate(from = "USD", to = NULL, format = "json", callback = NULL)` Current exchange rate. Returns a single numeric when `to` is set, otherwise a named numeric vector of all rates for `from`.

`convert(to, amount = 1, from = "USD", format = "json", callback = NULL)` Convert amount from one currency to another. Returns a numeric.

`get_supported_currencies(format = "json", callback = NULL)` Character vector of supported currency codes.

`get_historical_rate(date, amount = 1, from = "USD", to = NULL, format = "json", callback = NULL)` Historical rate/conversion on a date (Pro tier).

`get_historical_rates(date, amount = 1, base = "USD", format = "json", callback = NULL)` All historical rates for a base on a date (Pro tier).

`convert_historical(amount, from, to, date, format = "json", callback = NULL)` Convert using a historical rate (Pro tier).

`get_time_series(start_date, end_date, amount = 1, base = "USD", currencies = NULL, format = "json", callback = NULL)` Time series of rates over a date range (Pro tier).

`get_historical_limits(format = "json", callback = NULL)` Historical coverage per currency (Pro tier).

`get_vat_rates(country = NULL, format = "json", callback = NULL)` VAT rates for all countries, or a single country when country is set.

`get_vat_rate(country, format = "json", callback = NULL)` VAT data for a single country.

When format is not "json", methods return the raw response body as a character string.

Examples

```
## Not run:
client <- unirate_client(Sys.getenv("UNIRATE_API_KEY"))
client$get_rate("USD", "EUR")
client$convert(to = "EUR", amount = 100, from = "USD")
client$get_supported_currencies()

## End(Not run)
```

Index

unirate_client, [2](#)