Package: stacr (via r-universe)

May 12, 2026

Title Tidy 'STAC' Workflows for R

Version 0.1.0

Description Wraps the 'rstac' package with a pipe-friendly, tidy API.
All results return 'tibbles' instead of nested lists. Ships
with a catalog registry of known 'STAC' endpoints including
Planetary Computer, Earth Search, and 'USGS', while supporting
any 'STAC' API URL.

License MIT + file LICENSE

URL https://github.com/null-island-labs/stacr

BugReports https://github.com/null-island-labs/stacr/issues

Depends R (>= 4.1.0)

Imports rstac, tibble, cli

Suggests gdalcubes, jsonlite, leaflet, sf, testthat (>= 3.0.0), withr,
knitr, rmarkdown

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.3

Config/testthat/edition 3

VignetteBuilder knitr

Config/pak/sysreqs libabsl-dev cmake libgdal-dev gdal-bin libgeos-dev libjpeg-dev libpng-dev libssl-dev libproj-dev libsqlite3-dev libudunits2-dev

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

Date/Publication 2026-03-10 12:42:09 UTC

RemoteUrl https://github.com/chrislyonsky/stacr

RemoteRef HEAD

RemoteSha 4d785df88ccba807ecc6b1c531f0eca861a040c6

Contents

`has_internet` *Check internet connectivity*

Description

Tests whether a network connection to a STAC endpoint is available. Used in `@examplesIf` guards for network-dependent examples.

Usage

```
has_internet()
```

Value

`TRUE` if an internet connection is available, `FALSE` otherwise.

Examples

```
has_internet()
```

`stac_catalogs` *List Known STAC Catalogs*

Description

Returns a tibble of known STAC API endpoints bundled with the package. Includes Planetary Computer, Earth Search, and USGS catalogs.

Usage

```
stac_catalogs()
```

Value

A tibble::tibble with columns:

name Human-readable catalog name.

url Root URL of the STAC API endpoint.

provider Organization providing the catalog.

Examples

```
stac_catalogs()
```

`stac_collections` *List STAC Collections*

Description

Queries a STAC API endpoint and returns available collections as a tidy tibble. Wraps `rstac::collections()` with tidy output.

Usage

```
stac_collections(url)
```

Arguments

`url`	Character. Root URL of a STAC API endpoint (e.g., `"https://earth-search.aws.element84.com/v1`

Value

A tibble::tibble with one row per collection and columns:

id Collection identifier.

title Human-readable title.

description Collection description.

Examples

```
stac_collections("https://earth-search.aws.element84.com/v1")
```

stac_download *Download STAC Assets*

Description

Downloads assets from STAC items returned by `stac_search()` or `stac_items()`. Wraps `rstac::assets_download()` with tidy output.

Usage

```
stac_download(items, assets = NULL, output_dir = tempdir(), overwrite = FALSE)
```

Arguments

`items`	An rstac `doc_items` object as returned by `stac_search()` or `stac_items()` with the `raw` attribute, or the raw rstac result directly.
`assets`	Character vector. Asset names to download (e.g., `c("red", "green", "blue")`). If `NULL`, downloads all assets.
`output_dir`	Character. Directory where files are saved. Defaults to `tempdir()`.
`overwrite`	Logical. Overwrite existing files? Defaults to `FALSE`.

Value

The rstac items object (invisibly) with assets downloaded to `output_dir`. Downloaded file paths can be retrieved with `rstac::assets_url()`.

Examples

```
# Search for items first
items <- stac_search(
  url = "https://earth-search.aws.element84.com/v1",
  collections = "sentinel-2-l2a",
  bbox = c(-84.5, 38.0, -84.3, 38.2),
  limit = 1
)
```

`stac_items` *List Items in a STAC Collection*

Description

Retrieves items from a specific collection in a STAC API endpoint and returns them as a tidy tibble. Wraps `rstac::items()` with tidy output.

Usage

```
stac_items(url, collection, limit = 100L)
```

Arguments

`url`	Character. Root URL of a STAC API endpoint (e.g., `"https://earth-search.aws.element84.com/v1``
`collection`	Character. The collection ID to list items from.
`limit`	Integer. Maximum number of items to return. Defaults to 100.

Value

A tibble::tibble with one row per item and columns:

id Item identifier.

collection Collection the item belongs to.

datetime Acquisition datetime as a character string.

bbox Bounding box as a numeric list column.

geometry GeoJSON geometry as a list column.

assets Character vector of available asset names.

Examples

```
stac_items(
  url = "https://earth-search.aws.element84.com/v1",
  collection = "sentinel-2-l2a",
  limit = 5
)
```

stac_map *Map STAC Item Footprints*

Description

Creates an interactive 'leaflet' map showing the spatial footprints of STAC items returned by `stac_search()` or `stac_items()`. Requires the 'leaflet' and 'sf' packages to be installed.

Usage

```
stac_map(items)
```

Arguments

`items`	A tibble::tibble of STAC items as returned by `stac_search()` or `stac_items()`, with a `geometry` list column.

Value

A `leaflet` htmlwidget object.

Examples

```
items <- stac_search(
  url = "https://earth-search.aws.element84.com/v1",
  collections = "sentinel-2-l2a",
  bbox = c(-84.5, 38.0, -84.3, 38.2),
  limit = 5
)
stac_map(items)
```

stac_search *Search a STAC API*

Description

Searches a STAC API endpoint for items matching the given filters and returns results as a tidy tibble. Wraps `rstac::stac_search()` with tidy output.

Usage

```
stac_search(
  url,
  collections = NULL,
  bbox = NULL,
  datetime = NULL,
  limit = 100L
)
```

Arguments

`url`	Character. Root URL of a STAC API endpoint (e.g., `"https://earth-search.aws.element84.com/v1`)
`collections`	Character vector. Collection IDs to search within.
`bbox`	Numeric vector of length 4: `c(xmin, ymin, xmax, ymax)`. Only items intersecting this bounding box are returned.
`datetime`	Character. Date/time filter as a single datetime or range (e.g., `"2024-01-01/2024-12-31"`). Follows RFC 3339.
`limit`	Integer. Maximum number of items to return. Defaults to 100.

Value

A tibble::tibble with one row per item and columns:

id Item identifier.

collection Collection the item belongs to.

datetime Acquisition datetime as a character string.

bbox Bounding box as a numeric list column.

geometry GeoJSON geometry as a list column.

assets Character vector of available asset names.

Examples

```
stac_search(
  url = "https://earth-search.aws.element84.com/v1",
  collections = "sentinel-2-l2a",
  bbox = c(-84.5, 38.0, -84.3, 38.2),
  limit = 5
)
```

stac_search_raw *Search STAC and Return Raw rstac Result*

Description

Like `stac_search()` but returns the raw rstac `doc_items` object instead of a tibble. Useful as input for `stac_download()` and `stac_to_cube()`.

Usage

```
stac_search_raw(
  url,
  collections = NULL,
  bbox = NULL,
  datetime = NULL,
  limit = 100L
)
```

Arguments

`url`	Character. Root URL of a STAC API endpoint (e.g., `"https://earth-search.aws.element84.com/v1``
`collections`	Character vector. Collection IDs to search within.
`bbox`	Numeric vector of length 4: `c(xmin, ymin, xmax, ymax)`. Only items intersecting this bounding box are returned.
`datetime`	Character. Date/time filter as a single datetime or range (e.g., `"2024-01-01/2024-12-31"`). Follows RFC 3339.
`limit`	Integer. Maximum number of items to return. Defaults to 100.

Value

An rstac `doc_items` object.

Examples

```
raw <- stac_search_raw(
  url = "https://earth-search.aws.element84.com/v1",
  collections = "sentinel-2-l2a",
  bbox = c(-84.5, 38.0, -84.3, 38.2),
  limit = 1
)
```

stac_to_cube *Convert STAC Items to a gdalcubes Image Collection*

Description

Bridges STAC search results to 'gdalcubes' for raster data cube analysis. Requires the 'gdalcubes' package to be installed.

Usage

```
stac_to_cube(items, asset_names = NULL, ...)
```

Arguments

items	An rstac doc_items object as returned by stac_search_raw() or rstac::post_request().
asset_names	Character vector. Asset names to include in the image collection (e.g., c("red", "green", "blue")).
...	Additional arguments passed to gdalcubes::stac_image_collection().

Value

A gdalcubes image collection object.

Examples

```
raw <- stac_search_raw(
  url = "https://earth-search.aws.element84.com/v1",
  collections = "sentinel-2-l2a",
  bbox = c(-84.5, 38.0, -84.3, 38.2),
  limit = 5
)
cube <- stac_to_cube(raw, asset_names = c("red", "green", "blue"))
```

Index