# Package: rmacrostrat (via r-universe)

May 19, 2026

**Title** Fetch Geologic Data from the 'Macrostrat' Platform

**Description** Work with the 'Macrostrat' (<https://macrostrat.org/>) Web Service (v.2, <https://macrostrat.org/api/v2>) to fetch geological data relevant to the spatial and temporal distribution of sedimentary, igneous, and metamorphic rocks as well as data extracted from them.

**Version** 1.0.0.9000

**License** GPL (>= 3)

**Language** en-US

**Encoding** UTF-8

**LazyData** true

**Roxygen** list(markdown = TRUE)

**RoxygenNote** 7.3.2

**Depends** R (>= 4.0)

**Imports** curl, data.table, geojsonsf, httr, jsonlite, protolite, sf

**Suggests** covr, cowplot, deeptime, devtools, ggplot2, ggrepel, ggspatial, knitr, rmarkdown, rnaturalearth, rnaturalearthhires, rphylopic, palaeoverse, testthat (>= 3.0.0), units

**Additional_repositories** https://ropensci.r-universe.dev

**VignetteBuilder** knitr

**Config/testthat/edition** 3

**URL** https://rmacrostrat.palaeoverse.org, https://palaeoverse.org, https://macrostrat.org, https://github.com/UW-Macrostrat/macrostrat/issues

**BugReports** https://github.com/palaeoverse/rmacrostrat/issues

**Config/pak/sysreqs** libabsl-dev cmake libgdal-dev gdal-bin libgeos-dev libprotobuf-dev libssl-dev libproj-dev protobuf-compiler libprotoc-dev libsqlite3-dev libudunits2-dev

**Repository** https://willgearty.r-universe.dev

**Date/Publication** 2025-08-21 15:45:39 UTC

**RemoteUrl** https://github.com/palaeoverse/rmacrostrat

**RemoteRef** HEAD

**RemoteSha** 9ff452f3c0190bf3685942f31fa0364467dc7257

# Contents

`catalog` *Retrieve a complete set of definitions*

## Description

A function to retrieve all definitions and/or metadata for a given definition set. Caution: some datasets are large and may take time to retrieve.

## Usage

```
catalog(type = NULL)
```

## Arguments

| | |
|---|---|
| `type` | `character`. The name of a definition set to retrieve. See Definitions section below for options. |

## Value

A `data.frame` containing an exhaustive list of the definitions and/or metadata retrievable for the given definition set. The variables in the `data.frame` will match those described in the documentation for that definition set.

## Definitions

- columns: Definitions of Macrostrat columns. See also `def_columns()`.
- drilling_sites: Definitions of eODP drilling cores. See also `def_drilling_sites()`.
- econs: Definitions of economic resources. See also `def_econs()`.
- environments: Definitions of environments. See also `def_environments()`.
- grain_sizes: Definitions of grain sizes. See also `def_grain_sizes()`.
- intervals: Definitions of temporal intervals. See also `def_intervals()`.
- lithologies: Definitions of lithologies. See also `def_lithologies()`.
- lithology_att: Definitions of lithology attributes. See also `def_lithology_att()`.
- measurements: Definitions of measurements. See also `def_measurements()`.
- minerals: Defintions of minerals. See also `def_minerals()`.
- plates: Definitions of tectonic plates. See also `def_plates()`.
- projects: Definitions of Macrostrat projects. See also `def_projects()`.
- references: Definitions of references. See also `def_references()`.
- sources: Definitions of geological maps. See also `def_sources()`.
- strat_name_concepts: Definitions of stratigraphic name concepts. See also `def_strat_name_concepts()`.
- strat_names: Definitions of stratigraphic names. See also `def_strat_names()`.
- structures: Definitions of geological structures. See also `def_structures()`.
- timescales: Definitions of timescales. See also `def_timescales()`.

### Developer(s)

Bethany Allen & Lewis A. Jones

### Reviewer(s)

Lewis A. Jones

### Examples

```
# Get mineral definitions
ex1 <- catalog(type = "minerals")
# Get grain size definitions
ex2 <- catalog(type = "grain_sizes")
```

---

## def_columns *Define Macrostrat columns*

---

### Description

A function to retrieve the definitions for Macrostrat columns. By default, all definitions are returned.

### Usage

```
def_columns(
  column_id = NULL,
  column_group_id = NULL,
  column_name = NULL,
  project_id = NULL,
  status = NULL
)
```

### Arguments

| | |
|---|---|
| column_id | integer. The unique identification number(s) of the desired column(s) to return a definition for. |
| column_group_id | integer. The unique identification number(s) of the desired column group(s) to return a definition for. |
| column_name | character. The name of the desired column to return a definition for. |
| project_id | integer. The unique identification number(s) of the desired Macrostrat project(s) to return a definition for. |
| status | character. The status of the column to return a definition for. Either "active", "in process", or "obsolete". |

## Value

A `data.frame` containing the following columns:

- `col_id`: The unique identification number for the column.
- `col_group_id`: The unique identification number of the group to which the column belongs.
- `col_name`: The name of the column.
- `lat`: Latitude of the centroid of the column.
- `lng`: Longitude of the centroid of the column.
- `col_area`: Area of the Macrostrat column, in $km^2$.
- `max_thick`: Maximum unit thickness in meters.
- `ref_id`: The unique identification number for the reference associated with the column.
- `status`: Indicates the current status of the column (values are 'active', 'in process', and 'obsolete').
- `t_units`: Total number of Macrostrat units contained within the column.
- `project_id`: The unique identification number for the column's project. Corresponds to general geographic region.
- `notes`: Column specific notes.

## Developer(s)

William Gearty

## Reviewer(s)

Christopher D. Dean

## See Also

Macrostrat data entities: `get_age_model()`, `get_columns()`, `get_sections()`, `get_units()`

## Examples

```
# Return all column definitions
ex1 <- def_columns()
# Return subsets of column definitions
ex2 <- def_columns(column_group_id = 17)
ex3 <- def_columns(column_id = c(22,24))
ex4 <- def_columns(column_name = "Eastern Kentucky")
```

`def_drilling_sites` *Define objects associated with eODP*

## Description

Obtain metadata for variables associated with the Extending Ocean Drilling Pursuits (eODP) project. By default, data for all drilling sites are returned.

## Usage

```
def_drilling_sites(program = NULL, exp = NULL, site = NULL, sf = FALSE)
```

## Arguments

| | |
|---|---|
| `program` | `character`. The name of a drilling program (i.e., "DSDP", "ODP", or "IODP") to return a definition for. |
| `exp` | `character`. The unique identification number(s) of drilling expedition(s) to return a definition for (formerly known as 'leg(s)'). |
| `site` | `character`. The unique identification number(s) of drilling site(s) to return a definition for. |
| `sf` | `logical`. Should the results be returned as an `sf` object? Defaults to `FALSE`. |

## Value

A `data.frame` object containing, for each retrieved core:

- `epoch`: The name of the drilling program.
- `leg`: The name of the expedition (formerly known as a 'leg').
- `site`: The name of the drilling site.
- `hole`: The name of the drilling hole.
- `lat`: The decimal degree latitude of the drilling hole.
- `lng`: The decimal degree longitude of the drilling hole.
- `col_id`: The unique identification number of the eODP column.
- `col_group_id`: The unique identification number of the group to which the eODP column belongs.
- `penetration`: The depth of the hole drilled, in meters.
- `cored`: The amount of rock cored from the drill hole, in meters.
- `recovered`: The amount of rock recovered from the core, in meters.
- `recovery`: The proportion of rock recovered from the core.
- `drilled_interval`: The interval drilled.
- `drilled_intervals`: The number of drilled intervals.
- `cores`: The total number of cores drilled at the hole.

- `date_started`: The date on which drilling commenced.
- `date_finished`: The date on which drilling concluded.
- `comments`: Written notes assigned to the core.
- `ref_id`: The unique identification number of the reference.

If `sf` is `TRUE`, an `sf` object is returned instead, with a "geometry" column that contains the spatial data instead of the `lat`/`lng` columns.

### Developer(s)

Bethany Allen

### Reviewer(s)

Christopher D. Dean

### References

Sessa JA, Fraass AJ, LeVay LJ, Jamson KM, and Peters SE. (2023). The Extending Ocean Drilling Pursuits (eODP) Project: Synthesizing Scientific Ocean Drilling Data. Geochemistry, Geophysics, Geosystems, 24 (3) e2022GC010655. doi:10.1029/2022GC010655.

### See Also

External data: `def_measurements()`, `get_eodp()`, `get_fossils()`, `get_measurements()`

### Examples

```
# Return all core information for a specific site
core_info <- def_drilling_sites(site = "U1547")
```

---

## `def_econs` *Define economic resources*

---

### Description

A function to retrieve the definitions for one or more economic resources in the Macrostrat database. By default, all economic definitions are returned.

### Usage

```
def_econs(econ = NULL, econ_type = NULL, econ_class = NULL, econ_id = NULL)
```

## Arguments

| | |
|---|---|
| `econ` | `character`. The name of the economic resource (e.g., "gas reservoir") to return a definition for. |
| `econ_type` | `character`. The type of economic resource (e.g., "hydrocarbon") to return a definition for. |
| `econ_class` | `character`. The class of economic resource (e.g., "energy") to return a definition for. |
| `econ_id` | `integer`. The unique identification number(s) of the economic resource(s) to return a definition for. |

## Value

A `data.frame` containing the following columns:

- `econ_id`: The unique identifier of the economic resource.
- `name`: The name of the economic resource.
- `type`: The type of economic resource.
- `class`: The class of economic resource.
- `color`: The recommended coloring for units based on the dominant lithology.
- `t_units`: The total number of Macrostrat units containing the economic resource.

## Developer(s)

Bethany Allen

## Reviewer(s)

William Gearty

## See Also

Geologic features and attributes: `def_environments()`, `def_grain_sizes()`, `def_lithologies()`, `def_lithology_att()`, `def_measurements()`, `def_minerals()`, `def_structures()`

## Examples

```
# Return all economic resource definitions
econ_inf <- def_econs()
# Return only definitions for hydrocarbon resources
econ_inf <- def_econs(econ_type = "hydrocarbon")
```

def_environments *Define environments*

## Description

A function to retrieve the definitions for one or more environments in the Macrostrat database. By default, all environment definitions are returned.

## Usage

```
def_environments(
  environ = NULL,
  environ_type = NULL,
  environ_class = NULL,
  environ_id = NULL
)
```

## Arguments

| | |
|---|---|
| environ | character. The name of a specific environment (e.g., "reef") to return a definition for. |
| environ_type | character. The name of a type of environment (e.g., "carbonate") to return a definition for. |
| environ_class | character. The name of a class of environment (e.g., "marine") to return a definition for. |
| environ_id | integer. The unique identification number of the desired environment to return a definition for. |

## Value

A data.frame containing the following columns:

- environ_id: The unique identification number of the environment.
- name: The name of the environment.
- type: Environment type, less inclusive than class.
- class: Environment class, more inclusive than type.
- color: Recommended coloring for environment, based on the dominant lithology.
- t_units: The total number of Macrostrat units that are partially, or entirely, composed of the environment.

## Developer(s)

Christopher D. Dean

### Reviewer(s)

Lewis A. Jones

### See Also

Geologic features and attributes: def_econs(), def_grain_sizes(), def_lithologies(), def_lithology_att(), def_measurements(), def_minerals(), def_structures()

### Examples

```
# Return all environment definitions
ex1 <- def_environments()
# Return subsets of environment definitions
ex2 <- def_environments(environ = "reef")
ex3 <- def_environments(environ_type = "carbonate")
ex4 <- def_environments(environ_class = "marine")
```

---

## def_grain_sizes *Define grain sizes*

---

### Description

A function to return grain size definitions from the Macrostrat Database based on user-specified arguments. If no arguments are specified, all definitions are returned.

### Usage

```
def_grain_sizes(
  grain_name = NULL,
  grain_group = NULL,
  soil_group = NULL,
  grain_id = NULL,
  grain_symbol = NULL,
  grain_classification = NULL
)
```

### Arguments

| | |
|---|---|
| grain_name | character. The name of the specific grain (e.g., "Coarse Gravel") to return a definition for. |
| grain_group | character. The name of the group the grain belongs to (e.g., "Gravel") to return a definition for. |
| soil_group | character. The name of the family of soils the grain belongs to (e.g., "Coarse Soil") to return a definition for. |
| grain_id | integer. The unique identification number(s) of the desired grain definitions(s) to return a definition for. |

`grain_symbol` `character`. The abbreviation of the desired grain definition(s) (e.g., "CGr") to return a definition for.

`grain_classification` `character`. The classification scheme the given grain belongs to (e.g., "Wentworth") to return a definition for.

## Value

A `data.frame` containing the following columns:

- `grain_id`: The unique identification number of the grain.
- `grain_symbol`: The grain symbol (or abbreviation) to use for display purposes.
- `grain_name`: The name of the grain.
- `grain_group`: The name of the group the grain belongs to.
- `soil_group`: The name of the soil family the grain belongs to.
- `min_size`: The minimum size of the size in millimeters.
- `max_size`: The maximum size of the grain in millimeters.
- `classification`: The classification scheme the given grain belongs to.

## Developer(s)

Lewis A. Jones

## Reviewer(s)

William Gearty

## See Also

Geologic features and attributes: `def_econs()`, `def_environments()`, `def_lithologies()`, `def_lithology_att()`, `def_measurements()`, `def_minerals()`, `def_structures()`

## Examples

```
# Return all definitions
ex1 <- def_grain_sizes()
# Return grain sizes by group
ex2 <- def_grain_sizes(grain_group = "Gravel")
```

`def_intervals` *Define Macrostrat time intervals*

## Description

A function to retrieve Macrostrat time interval definitions matching a user-specific search criteria. If no arguments are specified (the default), all time interval definitions are returned.

## Usage

```
def_intervals(
  timescale = NULL,
  interval_name = NULL,
  interval_id = NULL,
  age = NULL,
  age_top = NULL,
  age_bottom = NULL,
  rule = NULL,
  true_colors = NULL
)
```

## Arguments

| | |
|---|---|
| `timescale` | `character`. The name of the desired timescale to return a definition for (e.g., "international epochs"). |
| `interval_name` | `character`. The name of the desired interval to return a definition for. |
| `interval_id` | `integer`. The identification number(s) of the desired time interval to return a definition for. |
| `age` | `numeric`. Age in millions of years before present to return an interval definition for. All intervals overlapping with this age will be returned. |
| `age_top` | `numeric`. The minimum age for which interval definitions should be returned, in millions of years before present. If specified, `age_bottom` must also be specified, and this must be younger than `age_bottom`. |
| `age_bottom` | `numeric`. The maximum age for which interval definitions should be returned, in millions of years before present. If specified, `age_top` must also be specified, and this must be older than `age_top`. |
| `rule` | `character`. How should interval definitions be returned for the given `age_top` and `age_bottom`? Use "contains" to return all intervals that fall entirely within `age_top` and `age_bottom`. Use "exact" to return any intervals with both boundaries equal to `age_top` and `age_bottom`. Use "loose" (the default) to return all intervals touching the range of `age_top` and `age_bottom`. If `age_top` and `age_bottom` are not provided, this argument is ignored. |
| `true_colors` | `logical`. Should the original international time scale colors be returned? Defaults to `TRUE`. |

## Value

A `data.frame` containing the following columns:

- `int_id`: The unique identification number of the time interval.
- `name`: The name of the time interval.
- `abbrev`: The standard abbreviation for the time interval name, if one exists.
- `t_age`: The top age (minimum age) in millions of years before present.
- `b_age`: The bottom age (maximum age) in millions of years before present.
- `int_type`: The temporal rank of the interval (e.g., "age", "epoch", "chron").
- `timescales`: A `data.frame` containing the timescale(s) that the interval is included in (see `def_timescales()` for more details), with the following columns:
  - `timescale_id`: The unique identification number of the timescale.
  - `name`: The name of the timescale.
- `color`: The recommended coloring for units based on dominant lithology.

## Developer(s)

Lewis A. Jones

## Reviewer(s)

William Gearty

## See Also

Timescales and time intervals: `def_timescales()`

## Examples

```
# Return all intervals
ex1 <- def_intervals()
# Return all specific timescale intervals
ex2 <- def_intervals(timescale = "international ages")
# Return for specific age
ex3 <- def_intervals(timescale = "international ages", age = 70)
```

`def_lithologies` *Define lithologies*

## Description

A function to retrieve the definitions of one or more lithologies in the Macrostrat database. If no arguments are specified (the default), all lithology definitions are returned.

## Usage

```
def_lithologies(
  lithology = NULL,
  lithology_group = NULL,
  lithology_class = NULL,
  lithology_type = NULL,
  lithology_id = NULL
)
```

## Arguments

| | |
|---|---|
| `lithology` | `character`. The name of the desired lithology (e.g., "limestone") to return a definition for. |
| `lithology_group` | `character`. The name of the desired lithological group (e.g., "unconsolidated") to return a definition for. |
| `lithology_class` | `character`. The name of the desired lithological class (e.g., "sedimentary") to return a definition for. |
| `lithology_type` | `character`. The name of the desired lithological type (e.g., "siliciclastic") to return a definition for. |
| `lithology_id` | `integer`. The unique identification number(s) of one or more lithologies to return a definition for. |

## Value

A `data.frame` containing the following columns:

- `lith_id`: The unique identification number of the lithology.
- `name`: The name of the lithology.
- `type`: The lithology type, less inclusive than class.
- `group`: The lithology group, less inclusive than type.
- `class`: The lithology class, more inclusive than type.
- `color`: The recommended coloring for the lithology.
- `fill`: The code for the fill pattern as established in the Federal Geographic Data Committee's Digital Cartographic Standard for Geologic Map Symbolization.
- `t_units`: The total number of Macrostrat units that are partially or entirely composed of the lithology.

### Developer(s)

William Gearty

### Reviewer(s)

Bethany Allen

### See Also

Geologic features and attributes: def_econs(), def_environments(), def_grain_sizes(), def_lithology_att(), def_measurements(), def_minerals(), def_structures()

### Examples

```
# return all lithology definitions
ex1 <- def_lithologies()
# return a definition for sandstone
ex2 <- def_lithologies(lithology = "sandstone")
# return definitions for lithologies of a specific type
ex3 <- def_lithologies(lithology_type = "sedimentary")
# return definitions for lithologies using their unique ID numbers
ex4 <- def_lithologies(lithology_id = c(1,5))
```

---

## def_lithology_att *Define lithology attributes*

---

### Description

A function to retrieve the definitions for one or more lithology attributes in the Macrostrat database. By default, all lithology attribute definitions are returned.

### Usage

```
def_lithology_att(
  lithology_att = NULL,
  att_type = NULL,
  lithology_att_id = NULL
)
```

### Arguments

| | |
|---|---|
| lithology_att | character. The name of the lithology attribute (e.g., "cross-bedded") to return a definition for. |
| att_type | character. The type of lithology attribute (e.g., "sed structure") to return a definition for. |
| lithology_att_id | integer. The unique identification number(s) of lithology attribute(s) to return a definition for. |

## Value

A `data.frame` containing the following columns:

- `lith_att_id`: The unique identifier of the lithology attribute.
- `name`: The name of the lithology attribute.
- `type`: The lithology attribute type.
- `t_units`: The total number of Macrostrat units to which the lithology attribute is attached.

## Developer(s)

Bethany Allen

## Reviewer(s)

Christopher D. Dean

## See Also

Geologic features and attributes: `def_econs()`, `def_environments()`, `def_grain_sizes()`, `def_lithologies()`, `def_measurements()`, `def_minerals()`, `def_structures()`

## Examples

```
# Return all lithology attribute definitions
ex1 <- def_lithology_att()
# Return subsets of lithology attribute definitions
ex2 <- def_lithology_att(att_type = "sed structure")
```

---

# `def_measurements` *Define measurements*

---

## Description

A function to retrieve the definitions of different measurements that are included in the Macrostrat database. By default, all definitions are returned.

## Usage

```
def_measurements(
  measure_id = NULL,
  measurement_type = NULL,
  measurement_class = NULL
)
```

## Arguments

| | |
|---|---|
| measure_id | integer. The unique identification number(s) of the desired measurement(s) to return a definition for. |
| measurement_type | character. Filter the returned definitions to those of one or more named type(s) of measurement (e.g., "geochronological"). This is a less inclusive grouping than measurement_class. |
| measurement_class | character. Filter the returned definitions to those of one or more named class(es) of measurement (e.g., "geochemical"). This is a more inclusive grouping than measurement_type. |

## Value

A data.frame containing the following columns:

- measure_id: The unique identification number of the measurement.
- name: The name of the measurement.
- type: Measurement type, less inclusive than class
- class: Measurement class, more inclusive than type.
- t_units: The total number of Macrostrat units containing this measurement.

## Developer(s)

William Gearty

## Reviewer(s)

Christopher D. Dean

## See Also

External data: def_drilling_sites(), get_eodp(), get_fossils(), get_measurements()

Geologic features and attributes: def_econs(), def_environments(), def_grain_sizes(), def_lithologies(), def_lithology_att(), def_minerals(), def_structures()

## Examples

```
# Return all definitions
ex1 <- def_measurements()
# Return subsets of definitions
ex2 <- def_measurements(measure_id = c(1, 2, 4))
ex3 <- def_measurements(measurement_class = "geochemical")
```

`def_minerals` *Define minerals*

## Description

A function to retrieve mineral definitions from the Macrostrat database based on user-specified arguments. If no arguments are provided, all definitions are returned.

## Usage

```
def_minerals(mineral = NULL, mineral_type = NULL, element = NULL)
```

## Arguments

| | |
|---|---|
| `mineral` | `character`. The name of a mineral (e.g., "calcite") to return a definition for. |
| `mineral_type` | `character`. The type of mineral (e.g., "feldspar") to return a definition for. |
| `element` | `character`. The chemical element abbreviation of a mineral (e.g., "CaCO3") to return a definition for. Note: abbreviations are case sensitive. |

## Value

A `data.frame` containing the following columns:

- `mineral_id`: The unique identification number of the mineral.
- `mineral`: The name of the mineral.
- `mineral_type`: The name of the type of mineral.
- `formula`: The chemical formula of the mineral.
- `formula_tags`: The chemical formula of the mineral with appropriate sub/superscript html tags.
- `url`: A URL for the mineral's entry in Mindat.
- `hardness_min`: The minimum hardness value (Mohs Hardness Scale) of the mineral.
- `hardness_max`: The maximum hardness value (Mohs Hardness Scale) of the mineral.
- `crystal_form`: The crystal form of the mineral.
- `mineral_color`: A description of the color of the mineral.
- `lustre`: A description of the lustre of the mineral.

## Developer(s)

Lewis A. Jones

## Reviewer(s)

Bethany Allen

## See Also

Geologic features and attributes: def_econs(), def_environments(), def_grain_sizes(), def_lithologies(), def_lithology_att(), def_measurements(), def_structures()

## Examples

```
# Get all mineral definitions
ex1 <- def_minerals()
# Get mineral definitions by type
ex2 <- def_minerals(mineral_type = "element")
```

---

# def_plates *Define tectonic plates*

---

## Description

A function to retrieve tectonic plate definitions for a user-specified plate identification number, as defined by the Wright et al. (2013) Global Plate Model. If no arguments are supplied, all plates are returned.

## Usage

```
def_plates(plate_id = NULL)
```

## Arguments

| | |
|---|---|
| plate_id | integer. The unique identification number(s) of the desired plate(s) to return a definition for. If NULL (default), all plates are returned. |

## Value

A data.frame containing the plate_id of the tectonic plate and the name of the tectonic plate as defined by Wright et al. (2013).

## Developer(s)

Lewis A. Jones

## Reviewer(s)

Christopher D. Dean

## References

Wright, N., Zahirovic, S., Müller, R.D., Seton, M. (2013). Towards community-driven paleogeographic reconstructions: integrating open-access paleogeographic and paleobiology data with plate tectonics. Biogeosciences, 10, 1529–1541. doi:10.5194/bg1015292013.

## See Also

Paleogeography: `get_paleogeography()`

## Examples

```
# Get all tectonic plates
ex1 <- def_plates()
# Get name of tectonic plate by ID
ex2 <- def_plates(plate_id = 604)
```

---

`def_projects` *Retrieve metadata for Macrostrat projects*

---

## Description

A function to obtain information concerning the different Macrostrat projects. If no arguments are specified (default), data for all projects are returned.

## Usage

```
def_projects(project_id = NULL)
```

## Arguments

| | |
|---|---|
| `project_id` | `integer`. The unique identification number(s) of the desired project(s) to return a definition for. |

## Value

A `data.frame` containing:

- `project_id`: The unique identification number of the Macrostrat project.
- `project`: The name of the Macrostrat project.
- `descrip`: Description of the Macrostrat project.
- `timescale_id`: The unique identification number of the timescale used in the Macrostrat project.
- `t_cols`: The total number of Macrostrat columns associated with the project.
- `in_process_cols`: The total number of in-progress Macrostrat columns associated with the project.
- `obsolete_cols`: The total number of obsolete Macrostrat columns associated with the project.
- `t_units`: The total number of Macrostrat units associated with the project.
- `area`: The area covered by the Macrostrat project in $km^2$.

### Developer(s)

Bethany Allen

### Reviewer(s)

Lewis A. Jones

### See Also

Macrostrat database metadata: `def_references()`, `def_sources()`, `get_stats()`

### Examples

```
ex1 <- def_projects()
ex2 <- def_projects(project_id = 4)
```

---

## `def_references` *Define references*

---

### Description

A function to retrieve the definitions for one or more published scientific references in the Macrostrat database. By default, all reference definitions are returned.

### Usage

```
def_references(ref_id = NULL)
```

### Arguments

| | |
|---|---|
| `ref_id` | `integer`. The unique identification number(s) of the desired reference(s) to return a definition for. If `NULL` (default), all references are returned. |

### Value

A `data.frame` containing the following columns:

- `ref_id`: The unique identification number(s) for the reference(s).
- `pub_year`: The year of publication of the reference.
- `author`: The author(s) of the reference.
- `ref`: The name of the reference.
- `doi`: The digital object identifier of the reference.
- `url`: A URL where additional information, the source, or contributing publication can be found.
- `t_units`: The total number of Macrostrat units associated with the reference.

### Developer(s)

William Gearty

### Reviewer(s)

Bethany Allen

### See Also

Macrostrat database metadata: def_projects(), def_sources(), get_stats()

### Examples

```
# Return all references
ex1 <- def_references()
# Return a subset of references
ex2 <- def_references(ref_id = c(1, 2, 3))
```

---

## def_sources *Retrieve geologic map source definitions*

---

### Description

A function to retrieve the definitions for one or more geologic map sources in the Macrostrat database. By default, all source definitions are returned.

### Usage

```
def_sources(
  source_id = NULL,
  lat = NULL,
  lng = NULL,
  shape = NULL,
  buffer = NULL,
  scale = NULL,
  sf = TRUE
)
```

### Arguments

| | |
|---|---|
| source_id | integer. The unique identification number(s) of the desired source(s) to return a definition for. |
| lat | numeric. A valid latitude in decimal degrees to return a source definition for. Must also supply lng. |
| lng | numeric. A valid longitude in decimal degrees to return a source definition for. Must also supply lat. |

| | |
|---|---|
| shape | character. A valid well-known text (WKT) representation of geometry, such as "POINT(30 10)" or "POLYGON((30 10, 40 40, 20 40, 10 20))", to return a source definition for. |
| buffer | integer. The geographic buffer (in meters) that should be applied to the specified shape. |
| scale | character. The desired Burwell scale, either: "tiny" (global), "small" (continental), "medium" (regional), or "large" (local). |
| sf | logical. Should the results be returned as an sf object? Defaults to TRUE. If FALSE, a data.frame is returned. |

## Value

A data.frame containing the following columns:

- source_id: Identification number of the geologic map source.
- name: Name of the geologic map source.
- url: URL where additional information, the source, or contributing publication can be found.
- ref_title: Title of reference for geologic map source.
- authors: Authors of geologic map source.
- ref_year: Year of reference publication.
- ref_source: Original publication source of the reference.
- isbn_doi: The ISBN or DOI for the reference.
- scale: The Macrostrat scale the geologic map source belongs to.
- features: The total number of features (i.e., outcrop shape and point elements) associated with the geologic map source.
- area: The total geographic area of the geologic map source in $km^2$.

If sf is TRUE (the default), an sf object is returned instead, with the same columns plus a "geometry" column that contains the spatial data.

## Developer(s)

William Gearty

## Reviewer(s)

Lewis A. Jones

## See Also

Geologic maps: get_map_legends(), get_map_outcrop(), get_map_points(), get_tiles()

Macrostrat database metadata: def_projects(), def_references(), get_stats()

## Examples

```
# Get all sources
ex1 <- def_sources()
# Get subset of sources
ex2 <- def_sources(source_id = c(1,2,4))
ex3 <- def_sources(lat = 43.03, lng = -89.4, scale = "large")
# Use WKT representation
library(sf)
line <- st_linestring(x = matrix(c(-122.3438, 37,-89.3527, 43.0582),
                                 byrow = TRUE, ncol = 2))
ex4 <- def_sources(shape = st_as_text(line), buffer = 100)
```

---

`def_strat_name_concepts`

*Define stratigraphic name concepts*

---

## Description

A function to retrieve the definitions for stratigraphic name concepts. By default, all stratigraphic name concept definitions are returned.

## Usage

```
def_strat_name_concepts(strat_concept_name = NULL, strat_concept_id = NULL)
```

## Arguments

`strat_concept_name`
: `character`. The name(s) of the desired stratigraphic name concept(s) to return a definition for (e.g., "Hell Creek").

`strat_concept_id`
: `integer`. The unique identification number(s) of the desired stratigraphic name concept(s) to return a definition for.

## Details

Stratigraphic name concepts consist of grouped `strat_names` which represent the same geologic entity. This may be due to stratigraphic names appearing in multiple reference maps or at multiple lithostratigraphic ranks.

## Value

A `data.frame` containing the following columns:

- `concept_id`: The unique identification number for the stratigraphic name concept, which groups variant `strat_names` for same geologic entity.
- `name`: The name of the stratigraphic name concept.

- `geologic_age`: The geologic age information assigned to the stratigraphic name concept, as defined in the original reference in plain text.
- `int_id`: The unique identification number of the chronostratigraphic interval associated with the stratigraphic name concept (see `def_intervals()`).
- `b_int_id`: The unique identification number of the bottom (oldest) chronostratigraphic interval associated with the stratigraphic name concept (see `def_intervals()`).
- `t_int_id`: The unique identification number of the top (youngest) chronostratigraphic interval associated with the stratigraphic name concept (see `def_intervals()`).
- `usage_notes`: Notes on the current validity of the stratigraphic name concept, in plain text.
- `other`: Other associated notes on the stratigraphic name concept, in plain text.
- `province`: The provinces where the stratigraphic name concept appears, in plain text.
- `refs`: The unique identification number of the reference from which the stratigraphic name concept was sourced (see `def_references()`).
- `url`: The URL associated with the stratigraphic name concept, where additional information, the source or contributing publication can be found.
- `author`: The authors of the source reference.

### Developer(s)

Christopher D. Dean

### Reviewer(s)

William Gearty

### See Also

Stratigraphic names: `def_strat_names()`

### Examples

```
# Return a stratigraphic name concept based on a string
ex1 <- def_strat_name_concepts(strat_concept_name = "Dakota")
# Return a stratigraphic name concept based on an ID
ex1 <- def_strat_name_concepts(strat_concept_id = 8212)
```

---

## `def_strat_names` *Define stratigraphic names*

---

### Description

A function to retrieve the definitions for one or more stratigraphic names in the Macrostrat database. By default, all stratigraphic name definitions are returned.

## Usage

```
def_strat_names(
  strat_name = NULL,
  interval_name = NULL,
  rank = NULL,
  concept_id = NULL,
  ref_id = NULL,
  strat_name_id = NULL,
  rule = NULL
)
```

## Arguments

| | |
|---|---|
| strat_name | character. The name(s) of the desired stratigraphic unit(s) to return a definition for. |
| interval_name | character. The chronostratigraphic interval name (e.g., "Campanian") for which to return stratigraphic name definitions. This returns strat_names with an age declared in Macrostrat, whose age has any temporal overlap with the desired interval. |
| rank | character. The formal stratigraphic rank abbreviation of the desired name(s) (i.e., "SGp", "Gp", "SubGp", "Fm", "Mbr", or "Bed") to return a definition for. |
| concept_id | integer. The unique identification number(s) of the stratigraphic concept(s) (grouped strat_names which represent the same entity) to return a definition for. |
| ref_id | character. The unique identification number(s) of the desired reference(s). Only stratigraphic names linked to the specified ref_id will be returned. |
| strat_name_id | integer. The unique identification number(s) of the desired stratigraphic name to return a definition for. |
| rule | character. The hierarchical rule for returning stratigraphic name definitions (either "all" or "down"). The "all" option will return the entire stratigraphic name hierarchy (parents and children of all levels) that the requested name(s) belongs to. The "down" option will return any direct children of the requested stratigraphic name(s). Can only be used with strat_name_id. |

## Value

A data.frame containing the following columns:

- strat_name: The short-form stratigraphic name (e.g. "Hell Creek").
- strat_name_long: The long-form stratigraphic name, including the lithostratigraphic rank of the stratigraphic name (e.g., "Hell Creek Formation").
- rank: The lithostratigraphic rank of the stratigraphic name (e.g., "Formation"), in short form (e.g., "Fm").
- strat_name_id: The unique identification number of the stratigraphic name.
- concept_id: The unique identification number of the associated stratigraphic name concept, which groups variant strat_names for the same geologic entity (see def_strat_name_concepts()).

- bed: The stratigraphic name of the bed associated with the specified stratigraphic name.
- bed_id: The unique identification number (strat_name_id) of the bed associated with the specified stratigraphic name associated with the specified stratigraphic name.
- mbr: The stratigraphic name of the member associated with the specified stratigraphic name.
- mbr_id: The unique identification number (strat_name_id) of the member associated with the specified stratigraphic name.
- fm: The stratigraphic name of the formation associated with the specified stratigraphic name.
- fm_id: The unique identification number (strat_name_id) of the formation associated with the specified stratigraphic name.
- subgp: The stratigraphic name of the subgroup associated with the specified stratigraphic name.
- subgp_id: The unique identification number (strat_name_id) of the subgroup associated with the specified stratigraphic name.
- gp: The stratigraphic name of the group associated with the specified stratigraphic name.
- gp_id: The unique identification number (strat_name_id) of the group associated with the specified stratigraphic name.
- sgp: The stratigraphic name of the supergroup associated with the specified stratigraphic name.
- sgp_id: The unique identification number (strat_name_id) of the supergroup associated with the specified stratigraphic name.
- b_age: The bottom age (maximum age) associated with the stratigraphic name, in millions of years before present.
- t_age: The top age (minimum age) associated with the stratigraphic name, in millions of years before present.
- b_period: The name of the oldest geologic time period associated with the stratigraphic name (see def_intervals()).
- t_period:The name of the youngest geologic time period associated with the stratigraphic name (see def_intervals()).
- c_interval: The name of the central interval associated with the stratigraphic name (see def_intervals()).
- t_units: The total number of Macrostrat units that are associated with the stratigraphic name.
- ref_id: The unique identification number of the reference from which the stratigraphic name was sourced (see def_references()).

## Developer(s)

Christopher D. Dean

## Reviewer(s)

Bethany Allen

## See Also

Stratigraphic names: def_strat_name_concepts()

## Examples

```
# Return a sample of stratigraphic names
ex1 <- def_strat_names()
# Return all stratigraphic names associated with a particular stratigraphic
# name
ex2 <- def_strat_names(strat_name = "Hell Creek")
# Return all stratigraphic names that intersect with a chosen time interval
ex3 <- def_strat_names(interval_name = "Campanian")
```

---

# def_structures *Define geological structure*

---

## Description

A function to retrieve definitions for various geological structures within the Macrostrat database.

## Usage

```
def_structures(
  structure = NULL,
  structure_class = NULL,
  structure_type = NULL,
  structure_id = NULL
)
```

## Arguments

| | |
|---|---|
| structure | character. The name of a geological structure (e.g., "syncline") to return a definition for. |
| structure_class | character. The name of the geological structure class (e.g., "sedimentology") to return a definition for. |
| structure_type | character. The name of the geological structure type (e.g., "fault") to return a definition for. |
| structure_id | integer. The unique idenfication number(s) of geological structures(s) to return a definition for. |

## Value

A data.frame containing the following columns:

- structure_id: The unique identification number of the structure.
- name: The name of the structure.
- structure_type: Structure type, less inclusive than class.
- group: Structure group, less inclusive than type.
- class: Structure class, more inclusive than type.

### Developer(s)

William Gearty

### Reviewer(s)

Lewis A. Jones

### See Also

Geologic features and attributes: def_econs(), def_environments(), def_grain_sizes(), def_lithologies(), def_lithology_att(), def_measurements(), def_minerals()

### Examples

```
# Get all structure definitions
ex1 <- def_structures()
# Get subset of structure definitions
ex2 <- def_structures(structure_id = c(1, 3))
ex3 <- def_structures(structure_class = "igneous")
```

---

## def_timescales *Define timescales*

---

### Description

A function to retrieve all timescale definitions from the Macrostrat database. Function is called without user-specified arguments.

### Usage

```
def_timescales()
```

### Value

A data.frame containing the following columns:

- timescale_id: The unique identification number of the timescale.
- timescale: The name of the timescale.
- n_intervals: The number of intervals within the timescale.
- max_age: The maximum age coverage of the timescale in millions of years before present.
- min_age: The minimum age coverage of the timescale in millions of years before present.
- ref_id: The unique identification number of the associated reference.

### Developer(s)

Lewis A. Jones

### Reviewer(s)

Christopher D. Dean

### See Also

Timescales and time intervals: `def_intervals()`

### Examples

```
# Retrieve all timescale definitions
ex1 <- def_timescales()
```

---

## get_age_model *Retrieve Macrostrat column age models*

---

### Description

A function to obtain information about the age models used for Macrostrat columns or sections. This consists of the temporal boundaries applied to the column's constituent units.

### Usage

```
get_age_model(column_id = NULL, section_id = NULL)
```

### Arguments

| | |
|---|---|
| `column_id` | `integer`. Filter age models to those associated with column(s) as specified by their unique identification number(s). |
| `section_id` | `integer`. Filter age models to those associated with section(s) as specified by their unique identification number(s). |

### Details

The Macrostrat age models are used to determine the ages of the boundaries between units in an internally consistent manner, allowing each boundary to be dated with respect to all age information available for the column.

### Value

A `data.frame` containing, for each boundary between units for each retrieved age model:

- boundary_id: The unique identification number of the boundary between two units.
- col_id: The unique identification number of the Macrostrat column.
- section_id: The unique identification number of the Macrostrat section.
- interval_id: The unique identification number of the time interval which the boundary lies within.

- interval_name: The name of the time interval which the boundary lies within.
- age_bottom: The numerical lower limit on the age of the boundary (maximum age), in millions of years before present.
- age_top: The numerical upper limit on the age of the boundary (minimum age), in millions of years before present.
- rel_position: The relative position of the boundary.
- model_age: The age of the boundary, as correlated with other ages in the column, in millions of years before present.
- boundary_status: The method used to determine the age of the boundary.
- boundary_type: The type of boundary.
- boundary_position: The position of the boundary.
- unit_below: The Macrostrat unit beneath the given boundary.
- unit_above: The Macrostrat unit above the given boundary.
- ref_id: The unique identification number of the reference.

### Developer(s)

Bethany Allen

### Reviewer(s)

Lewis A. Jones

### See Also

Macrostrat data entities: def_columns(), get_columns(), get_sections(), get_units()

### Examples

```
# Get age model for specific columns by ID
ex1 <- get_age_model(column_id = c(503, 504))
# Get age model for specific sections by ID
ex2 <- get_age_model(section_id = c(65, 22))
```

---

## get_columns *Retrieve Macrostrat column data*

---

### Description

A function to retrieve independent, hexagonal, vertical columns which contain chronostratigraphically organised Macrostrat units.

## Usage

```
get_columns(
  column_id = NULL,
  section_id = NULL,
  unit_id = NULL,
  strat_name = NULL,
  strat_name_id = NULL,
  interval_name = NULL,
  interval_id = NULL,
  age = NULL,
  age_top = NULL,
  age_bottom = NULL,
  lat = NULL,
  lng = NULL,
  lithology = NULL,
  lithology_id = NULL,
  lithology_type = NULL,
  lithology_class = NULL,
  lithology_att = NULL,
  lithology_att_id = NULL,
  lithology_att_type = NULL,
  environ = NULL,
  environ_id = NULL,
  environ_type = NULL,
  environ_class = NULL,
  pbdb_collection_no = NULL,
  econ = NULL,
  econ_id = NULL,
  econ_type = NULL,
  econ_class = NULL,
  project_id = NULL,
  adjacents = FALSE,
  sf = FALSE
)
```

## Arguments

| | |
|---|---|
| column_id | integer. Filter columns by their unique identification number(s). |
| section_id | integer. Filter columns to those containing section(s) as specified by their unique identification number(s). |
| unit_id | integer. Filter columns to those containing unit(s) as specified by their unique identification number(s). |
| strat_name | character. Filter columns to those containing a unit that fuzzy matches a stratigraphic name (e.g., "Hell Creek"). |
| strat_name_id | integer. Filter columns to those containing a unit that matches one or more stratigraphic name(s) as specified by their unique identification number(s). |

`interval_name` `character`. Filter columns to those that overlap with a named chronostratigraphic time interval (e.g., "Permian").

`interval_id` `integer`. Filter columns to those that overlap with a chronostratigraphic time interval as specified by their unique identification number(s).

`age` `numeric`. Filter columns to those that overlap with the specified numerical age, in millions of years before present.

`age_top` `numeric`. Filter columns to those that overlap with the age range between the specified numerical age and `age_bottom`. Should be in millions of years before present. `age_bottom` must also be specified, and this must be younger than `age_bottom`.

`age_bottom` `numeric`. Filter columns to those that overlap with the age range between the specified numerical age and `age_top`. Should be in millions of years before present. `age_top` must also be specified, and this must be older than `age_top`.

`lat` `numeric`. Return the column at the specified decimal degree latitude. Must also specify `lng`.

`lng` `numeric`. Return the column at the specified decimal degree longitude. Must also specify `lat`.

`lithology` `character`. Filter columns to those containing a named lithology (e.g., "shale", "sandstone").

`lithology_id` `integer`. Filter columns to those containing one or more lithology(ies) identified by their unique identification number(s).

`lithology_type` `character`. Filter columns to those containing a named lithology type (e.g., "carbonate", "siliciclastic").

`lithology_class` `character`. Filter columns to those containing a named lithology class (e.g., "sedimentary", "igneous", "metamorphic").

`lithology_att` `character`. Filter columns to those containing a named lithology attribute (e.g., "fine", "olivine", "poorly washed").

`lithology_att_id` `integer`. Filter columns to those containing one or more lithology attribute(s) as specified by their unique identification number(s).

`lithology_att_type` `character`. Filter columns to those containing a named category of lithology attribute (e.g., "grains", "lithology", "bedform").

`environ` `character`. Filter columns to those containing a named environment (e.g., "delta plain", "pond").

`environ_id` `integer`. Filter columns to those containing one or more environment(s) as specified by their unique identification number(s).

`environ_type` `character`. Filter columns to those containing a named environment type (e.g., "fluvial", "eolian", "glacial").

`environ_class` `character`. Filter columns to those containing a named environment class (e.g., "marine", "non-marine").

pbdb_collection_no
: integer. Filter columns to those containing one or more Paleobiology Database collection(s) as specified by their unique identification number(s).

econ
: character. Filter columns to those containing a named economic attribute (e.g., "brick", "ground water", "gold").

econ_id
: integer. Filter columns to those containing one or more economic attribute(s) as specified by their unique identification number(s).

econ_type
: character. Filter columns to those containing a named economic attribute type (e.g., "construction", "aquifer", "mineral").

econ_class
: character. Filter columns to those containing a named economic attribute class (e.g., "material", "water", "precious commodity").

project_id
: integer. Filter columns to those contained within a Macrostrat project as specified by its unique identification number.

adjacents
: logical. If column_id or lat/lng is specified, should all columns that touch the specified column be returned? Defaults to FALSE.

sf
: logical. Should the results be returned as an sf object? Defaults to FALSE.

## Details

More information can be found for the inputs for this function using the definition functions (beginning with def_).

## Value

A data.frame containing the following columns:

- col_id: The unique identification number of the Macrostrat column.
- col_name: The name of the Macrostrat column.
- col_group: Name of the group the column belongs to. Generally corresponds to geologic provinces.
- col_group_id: The unique identification number of the group to which the column belongs.
- group_col_id: The original column ID assigned to the column (used in the original source).
- lat: Decimal degree latitude of the column centroid.
- lng: Decimal degree longitude of the column centroid.
- col_area: The area of the Macrostrat column in $km^2$.
- project_id: The unique identification number for project. Corresponds to general geographic region.
- col_type: The type of column.
- refs: The unique identification number(s) for the reference(s) associated with the column.
- max_thick: The maximum unit thickness in meters.
- max_min_thick: The maximum possible minimum thickness in meters.
- min_min_thick: The minimum possible minimum thickness in meters.

- b_age: The age of the bottom of the column, estimated using the continuous time age model, in millions of years before present.
- t_age: The age of the top of the column, estimated using the continuous time age model, in millions of years before present.
- b_int_name: The name of the time interval represented at the bottom of the column.
- t_int_name: The name of the time interval represented at the top of the column.
- pbdb_collections: The number of PBDB collections contained within the column.
- lith: a data.frame containing the lithologies present within the column, with the following columns:
  - name: The named lithology (e.g., "sandstone").
  - type: The named lithology type (e.g., "siliciclastic").
  - class: The named lithology class (e.g., "sedimentary").
  - prop: The proportion of the lithology within the column, calculated from the individual Macrostrat units within the column.
  - lith_id: The unique identification number of the lithology.
- environ: a data.frame containing the environments present within the column, with the following columns:
  - name: The named environment (e.g., "delta plain").
  - type: The named environment type (e.g., "siliciclastic").
  - class: The named environment class (e.g., "marine").
  - prop: The proportion of the environment within the column, calculated from the individual Macrostrat units within the column.
  - environ_id: The unique identification number of the environment.
- econ: a data.frame containing the economic attributes present within the column, with the following columns:
  - name: The named economic attribute (e.g., "gold").
  - type: The named economic attribute type (e.g., "mineral").
  - class: The named economic attribute class (e.g., "precious commodity").
  - prop: The proportion of the economic attribute out of all economic attributes contained within the column, calculated from the individual Macrostrat units within the column.
  - econ_id: The unique identification number of the economic attribute.
- t_units: The total number of units contained within the column.
- t_sections: The total number of sections contained within the column.

If sf is TRUE, an sf object is returned instead, with the same columns plus a "geometry" column that contains the spatial data.

**Developer(s)**

William Gearty

**Reviewer(s)**

Christopher D. Dean

## See Also

Macrostrat data entities: def_columns(), get_age_model(), get_sections(), get_units()

## Examples

```
# Return columns that overlap with a named chronostratigraphic interval
ex1 <- get_columns(interval_name = "Permian")
# Return columns that overlap with a specified age range
ex2 <- get_columns(age_top = 200, age_bottom = 250)
# Return columns that contain a specific stratigraphic unit, in `sf` format
ex3 <- get_columns(strat_name = "mancos", sf = TRUE)
# Return the columns at a specific geographic coordinate
ex4 <- get_columns(lat = 43, lng = -89, adjacents = TRUE)
```

---

# get_eodp *Retrieve core data from ocean drilling programs*

---

## Description

A function to retrieve data from the Extending Ocean Drilling Pursuits (eODP) project, a collation of sedimentary description data from ocean drilling cores. This currently includes cores from the Deep Sea Drilling Project (DSDP), Integrated Ocean Drilling Program (IODP), and Ocean Drilling Program (ODP). Note: eODP data is currently being integrated into Macrostrat's data entities (e.g. columns and units) and should be available via get_columns() and get_units() in the future.

## Usage

```
get_eodp(column_id = NULL, site = NULL, leg = NULL, program = NULL, sf = FALSE)
```

## Arguments

| | |
|---|---|
| column_id | integer. Filter cores by their unique identification number(s). |
| site | character. Filter cores to those from a particular drilling site (e.g., "U1351"). |
| leg | character. Filter cores to those from a particular drilling leg or expedition (e.g., "317"). |
| program | character. Filter cores to those from a particular drilling program ("DSDP", "IODP", or "ODP"). |
| sf | logical. Should the results be returned as an sf object? Defaults to FALSE. |

## Details

More information can be found about the inputs for this function by using def_drilling_sites().

## Value

A `data.frame` containing, for each retrieved core:

- `col_group`: The name of the drilling program and leg/expedition.
- `site_hole`: The name of the drilling site and hole.
- `date_started`: The date on which drilling commenced.
- `ref_id`: The unique identifier of the reference.
- `col_id`: The unique identifier of the Macrostrat column.
- `lat`: Decimal degree latitude of the core.
- `lng`: Decimal degree longitude of the core.
- `top_depth`: A vector describing the height of the top of each unit.
- `bottom_depth`: A vector describing the height of the bottom of each unit.
- `primary_lith`: A vector giving the name of the primary lithology of each unit.
- `lith_id`: A vector giving the unique identifier of the primary lithology of each unit.
- `minor_lith`: A vector giving the name of the primary lithology of each unit.

If `sf` is `TRUE`, an `sf` object is returned instead, with a "geometry" column that contains the spatial data instead of the `lat`/`lng` columns.

## Developer(s)

Bethany Allen

## Reviewer(s)

William Gearty

## References

Sessa JA, Fraass AJ, LeVay LJ, Jamson KM, and Peters SE. (2023). The Extending Ocean Drilling Pursuits (eODP) Project: Synthesizing Scientific Ocean Drilling Data. Geochemistry, Geophysics, Geosystems, 24 (3) e2022GC010655. doi:10.1029/2022GC010655.

## See Also

External data: `def_drilling_sites()`, `def_measurements()`, `get_fossils()`, `get_measurements()`

## Examples

```
# Get data for specific cores
cores <- get_eodp(column_id = c(5081, 5082))
# Get data for all cores at a specific site
cores <- get_eodp(site = "U1351")
# Get data for all cores for a specific leg
cores <- get_eodp(leg = "317")
# Get data for all cores for a specific program
cores <- get_eodp(program = "IODP")
```

`get_fossils` *Retrieve collections from the Paleobiology Database*

## Description

A function to retrieve collections from the Paleobiology Database (PBDB), matched to Macrostrat units.

## Usage

```
get_fossils(
  unit_id = NULL,
  column_id = NULL,
  interval_name = NULL,
  age = NULL,
  age_top = NULL,
  age_bottom = NULL,
  lithology = NULL,
  lithology_id = NULL,
  lithology_type = NULL,
  lithology_class = NULL,
  environ = NULL,
  environ_id = NULL,
  environ_type = NULL,
  environ_class = NULL,
  econ = NULL,
  econ_id = NULL,
  econ_type = NULL,
  econ_class = NULL,
  project_id = NULL,
  strat_name_id = NULL,
  sf = FALSE
)
```

## Arguments

| | |
|---|---|
| `unit_id` | `integer`. Filter PBDB collections to those within one or more unit(s) as specified by their unique identification number(s). |
| `column_id` | `integer`. Filter PBDB collections to those within one or more column(s) as specified by their unique identification number(s). |
| `interval_name` | `character`. Filter PBDB collections to those that overlap with a named chronostratigraphic time interval (e.g., "Permian"). |
| `age` | `numeric`. Filter PBDB collections to those that overlap with the specified numerical age, in millions of years before present. |

| | |
|---|---|
| age_top | numeric. Filter PBDB collections to those that overlap with the age range between the specified numerical age and age_bottom. Should be in millions of years before present. age_bottom must also be specified, and this must be older than age_top. |
| age_bottom | numeric. Filter PBDB collections to those that overlap with the age range between the specified numerical age and age_top. Should be in millions of years before present. age_top must also be specified, and this must be younger than age_bottom. |
| lithology | character. Filter PBDB collections to those containing a named lithology (e.g., "shale", "sandstone"). |
| lithology_id | integer. Filter PBDB collections to those containing one or more lithology(ies) identified by their unique identification number(s). |
| lithology_type | character. Filter PBDB collections to those containing a named lithology type (e.g., "carbonate", "siliciclastic"). |
| lithology_class | character. Filter PBDB collections to those containing a named lithology class (e.g., "sedimentary", "igneous", "metamorphic"). |
| environ | character. Filter PBDB collections to those containing a named environment (e.g., "delta plain", "pond"). |
| environ_id | integer. Filter PBDB collections to those containing one or more environment(s) as specified by their unique identification number(s). |
| environ_type | character. Filter PBDB collections to those containing a named environment type (e.g., "fluvial", "eolian", "glacial"). |
| environ_class | character. Filter PBDB collections to those containing a named environment class (e.g., "marine", "non-marine"). |
| econ | character. Filter PBDB collections to those containing a named economic attribute (e.g., "brick", "ground water", "gold"). |
| econ_id | integer. Filter PBDB collections to those containing one or more economic attribute(s) as specified by their unique identification number(s). |
| econ_type | character. Filter PBDB collections to those containing a named economic attribute type (e.g., "construction", "aquifer", "mineral"). |
| econ_class | character. Filter PBDB collections to those containing a named economic attribute class (e.g., "material", "water", "precious commodity"). |
| project_id | integer. Filter sections to those contained within a Macrostrat project as specified by its unique identification number. |
| strat_name_id | integer. Filter PBDB collections to those containing a unit that matches one or more stratigraphic name(s) as specified by their unique identification number(s). |
| sf | logical. Should the results be returned as an sf object? Defaults to FALSE. |

## Details

More information can be found for the inputs for this function using the definition functions (beginning with def_). Terminology related to the PBDB can be found at https://paleobiodb.org/#/ or in the suggested references below.

## Value

A `dataframe` containing the following columns:

- `collection_no`: The unique identification number of the collection, as assigned in the PBDB.
- `collection_name`: The unique name of the collection, as assigned in the PBDB.
- `t_age`: The top age of the unit containing the collection, estimated using the continuous time age model, in millions of years before present.
- `b_age`: The bottom age of the unit containing the collection, estimated using the continuous time age model, in millions of years before present.
- `pbdb_occs`: The count of PBDB occurrences in the specified PBDB collection.
- `genus_no`: A vector containing the unique identification number for each genus that appears in the collection, corresponding to the genus_no column in the Paleobiology Database.
- `taxon_no`: The count of unique taxa in the specified PBDB collection.
- `unit_id`: The unique identification number of the Macrostrat unit containing the specified PBDB collection.
- `col_id`: The unique identification number of the Macrostrat column containing the specified PBDB collection.
- `refs`: Reference for the source of the data.
- `strat_name_concept_id`: The unique identification number of the stratigraphic name concept containing the specified PBDB collection.

If `sf = TRUE`, an `sf` object is returned instead.

## Developer(s)

Christopher D. Dean

## Reviewer(s)

Lewis A. Jones

## References

Peters, S.E. and McClennen, M. (2016). The Paleobiology Database application programming interface. *Paleobiology*, 42(1), pp. 1–7. doi:10.1017/pab.2015.39.

Uhen, M.D., Allen, B., Behboudi, N., Clapham, M.E., Dunne, E., Hendy, A., Holroyd, P.A., Hopkins, M., Mannion, P., Novack-Gottshall, P. and Pimiento, C. (2023). Paleobiology Database User Guide Version 1.0. *PaleoBios*, 40(11), pp. 1–56. doi:10.5070/P9401160531.

## See Also

External data: `def_drilling_sites()`, `def_measurements()`, `get_eodp()`, `get_measurements()`

## Examples

```
# Get fossils by Macrostrat column ID
ex1 <- get_fossils(column_id = 10)
# Get fossils by Macrostrat unit ID
ex2 <- get_fossils(unit_id = 6279)
# Get fossils by lithology and age
ex3 <- get_fossils(lithology = "sandstone", age_top = 66, age_bottom = 73)
# Get fossils by environment type and age
ex4 <- get_fossils(environ_type = "fluvial", age =  253)
```

---

# get_map_legends *Retrieve geologic map legend data*

---

## Description

A function to retrieve legend components from various geological maps.

## Usage

```
get_map_legends(
  source_id = NULL,
  lithology_id = NULL,
  lithology_type = NULL,
  lithology_class = NULL,
  description = NULL,
  comments = NULL,
  scale = NULL,
  carto = NULL
)
```

## Arguments

| | |
|---|---|
| source_id | integer. Filter legend components to those in sources specified by their unique identification number(s). |
| lithology_id | integer. Filter legend components to those containing one or more lithology(ies) as specified by their unique identification number(s). |
| lithology_type | character. Filter legend components to those containing one or more named lithology type(s) (e.g., "carbonate", "siliciclastic"). |
| lithology_class | character. Filter legend components to those containing one or more named lithology class(es) (e.g., "sedimentary", "igneous", "metamorphic"). |
| description | character. Filter legend components to those containing the given character string in the map unit description field. |
| comments | character. Filter legend components to those containing the given character string in the map unit comments field. |

| | |
|---|---|
| scale | character. Filter legend components to those from maps of specific scale(s) in Macrostrat's system. Options are "tiny" (global), "small" (continental), "medium" (regional), or "large" (local). |
| carto | character. Filter legend components to those of specific scale(s) in Macrostrat's system. Options are "tiny" (global), "small" (continental), "medium" (regional), or "large" (local). |

## Details

Potential Macrostrat map scales are "tiny" (global), "small" (continental), "medium" (regional), or "large" (local).

## Value

A dataframe containing the following columns:

- legend_id: The unique identification number of the map legend.
- source_id: The identification number of the source geologic map (see def_sources()).
- scale: The associated Macrostrat map scale for the legend.
- map_unit_name:The name of the outcrop shape element as defined in the geologic map source.
- strat_name: The geologic name(s) of the outcrop shape element.
- age: The chronostratigraphic bin assigned to the outcrop shape element, as defined in the geologic map source in plain text.
- lith: The lithology of the outcrop shape element as defined in the geologic map source in plain text.
- descrip: Description of the outcrop shape element in plain text.
- comments: Notes assigned to the outcrop shape element.
- t_age: The estimated top age of the outcrop shape element, in millions of years before present.
- b_age: The estimated bottom age of the outcrop shape element, in millions of years before present.
- b_interval: The unique identification number of the bottom time interval matched to the outcrop shape element.
- t_interval: The unique identification number of the top time interval matched to the outcrop shape element.
- strat_name_id: A vector containing the unique identification number(s) for known stratigraphic unit name(s) matched to the outcrop shape element (see def_strat_names()).
- unit_id: A vector containing the unique identification number(s) for known Macrostrat unit(s) matched to the outcrop shape element.
- lith_classes: A vector containing the named lithology class(es) (e.g., "sedimentary") matched to the outcrop shape element.
- lith_types: A vector containing the named lithology type(s) (e.g., "siliciclastic") matched to the outcrop shape element.
- lith_id: A vector containing the unique identification number(s) for known lithology(ies) represented within the outcrop shape element (see def_lithologies()).

- `color`: Recommended color for plotting the outcrop shape element based on the dominant lithology.
- `area`: The area of the outcrop shape element in $km^2$.
- `tiny_area`: The area of the outcrop shape element in $km^2$ at "tiny" Macrostrat map scale.
- `small_area`: The area of the outcrop shape element in $km^2$ at "small" Macrostrat map scale.
- `medium_area`: The area of the outcrop shape element in $km^2$ at "medium" Macrostrat map scale.
- `large_area`: The area of the outcrop shape element in $km^2$ at "large" Macrostrat map scale.

## Developer(s)

Christopher D. Dean

## Reviewer(s)

William Gearty

## See Also

Geologic maps: `def_sources()`, `get_map_outcrop()`, `get_map_points()`, `get_tiles()`

## Examples

```
# Get legend components by ID of the original source map
ex1 <- get_map_legends(source_id = 1)
# Get legend components for any sedimentary lithologies
ex2 <- get_map_legends(lithology_type = "sedimentary")
# Get any legend components that have comments with the word 'breccia'
ex3 <- get_map_legends(comments = "breccia")
```

---

# get_map_outcrop *Retrieve geologic map outcrop shape element data*

---

## Description

A function to retrieve data for geologic map outcrop shape elements from various sources.

## Usage

```
get_map_outcrop(
  map_id = NULL,
  unit_id = NULL,
  strat_name_id = NULL,
  lat = NULL,
  lng = NULL,
  scale = NULL,
  sf = TRUE
)
```

## Arguments

| | |
|---|---|
| map_id | integer. The unique identification number(s) of the map outcrop shape element(s) to return. |
| unit_id | integer. Filter outcrop shape elements to those that match one or more Macrostrat unit(s) as specified by their unique identification number(s). |
| strat_name_id | integer. Filter outcrop shape elements to those that match one or more stratigraphic name(s) as specified by their unique identification number(s). |
| lat | numeric. Return the outcrop shape element(s) at the specified decimal degree latitude. Must also specify lng. |
| lng | numeric. Return the outcrop shape element(s) at the specified decimal degree longitude. Must also specify lat. |
| scale | character. The Macrostrat map scale to use (a vector of scales is also supported). Options are "tiny" (global), "small" (continental), "medium" (regional), or "large" (local). |
| sf | logical. Should the results be returned as an sf object? Defaults to TRUE. If FALSE, a data.frame is returned. |

## Details

More information can be found for the inputs for this function using the definition functions (beginning with def_).

## Value

A data.frame containing the following columns:

- map_id: The identification number of the outcrop shape element.
- source_id: The identification number of the source geologic map.
- name: The name of the outcrop shape element in the original (or modified) source geologic map.
- strat_name: The geologic name(s) of the outcrop shape element.
- lith: The lithology of the outcrop shape element as defined in the geologic map source in plain text.
- descrip: Description of the outcrop shape element in plain text.
- comments: Notes assigned to the outcrop shape element.
- macro_units: A vector containing the unique identification number(s) for known Macrostrat unit(s) matched to the outcrop shape element.
- strat_names: A vector containing the unique identification number(s) for known stratigraphic unit name(s) matched to the outcrop shape element (see def_strat_names()).
- liths: A vector containing the unique identification number(s) of the lithology(ies) represented within the outcrop shape element (see def_lithologies()).
- t_int_id: The identification number of the chronostratigraphic interval containing the top boundary of the outcrop shape element.

- t_int_age: The top age of the chronostratigraphic interval containing the top boundary of the outcrop shape element.
- t_int_name: The name of the chronostratigraphic interval containing the top boundary of the outcrop shape element.
- b_int_id: The identification number of the chronostratigraphic interval containing the bottom boundary of the outcrop shape element.
- b_int_age: The bottom age of the chronostratigraphic interval containing the bottom boundary of the outcrop shape element.
- b_int_name: The name of the chronostratigraphic interval containing the bottom boundary of the outcrop shape element.
- color: Recommended color for plotting the outcrop shape element based on the dominant lithology.
- t_age: The estimated top age (minimum age) of the outcrop shape element, in millions of years before present.
- b_age: The estimated bottom age (maximum age) of the outcrop shape element, in millions of years before present.
- best_int_name: The best/most representative interval name for the outcrop shape element.

If sf is TRUE (the default), an sf object is returned instead, with the same columns plus a "geometry" column that contains the spatial data.

## Developer(s)

William Gearty

## Reviewer(s)

Lewis A. Jones

## See Also

Geologic maps: def_sources(), get_map_legends(), get_map_points(), get_tiles()

## Examples

```
ex1 <- get_map_outcrop(lat = 43, lng = -89.3)
ex2 <- get_map_outcrop(lat = 43, lng = -89.3, scale = "tiny")
```

get_map_points *Retrieve geologic map point shape element data*

## Description

A function to retrieve data for geologic map point shape elements from various sources.

## Usage

```
get_map_points(
  point_id = NULL,
  point_type = NULL,
  min_lat = NULL,
  min_lng = NULL,
  max_lat = NULL,
  max_lng = NULL,
  source_id = NULL,
  sf = TRUE
)
```

## Arguments

| | |
|---|---|
| point_id | integer. The unique identification number(s) of the point(s) to return. |
| point_type | character. Filter points to those of one or more point type(s). Options are "cleavage", "bedding", "axial plane", "fault plane", "foliation", and "joint". Ignored if point_id is supplied. |
| min_lat | integer. The minimum latitude used to define the bounding box of the points to return. Requires min_lng, max_lat, and max_lng. Ignored if point_id is supplied. |
| min_lng | integer. A minimum longitude used to define the bounding box of the points to return. Requires min_lat, max_lat, and max_lng. Ignored if point_id is supplied. |
| max_lat | integer. A maximum latitude used to define the bounding box of the points to return. Requires min_lat, min_lng, and max_lng. Ignored if point_id is supplied. |
| max_lng | integer. A maximum longitude used to define the bounding box of the points to return. Requires min_lat, min_lng, and max_lat. Ignored if point_id is supplied. |
| source_id | integer. The unique identification number(s) of the source(s) to filter the points by. Ignored if point_id is supplied. |
| sf | logical. Should the results be returned as an sf object? Defaults to TRUE. |

## Details

More information can be found relating to the inputs for this function using the definition functions (beginning with def_).

## Value

An sf object containing the following columns:

- point_id: The identification number of the point element.
- strike: The strike value of the point element.
- dip: The dip value of the point element.
- dip_dir: The dip direction of the point element.
- point_type: The type of point element (e.g. "bedding").
- certainty: The certainty of the point element location.
- comments: Notes assigned to the point element.
- source_id: The unique identification number of the source for the point element.
- geometry: The point spatial data.

If sf is TRUE (the default), an sf object is returned, with the a "geometry" column that contains the spatial data. If sf is FALSE, a data.frame object is returned with two additional columns (lng, lat) containing the geographic coordinates of the point elements.

## Developer(s)

Lewis A. Jones

## Reviewer(s)

Bethany Allen

## See Also

Geologic maps: def_sources(), get_map_legends(), get_map_outcrop(), get_tiles()

## Examples

```
# Return a specific point
ex1 <- get_map_points(point_id = 1)
# Return all points within a user-specified bounding box
ex2 <- get_map_points(min_lng = -80, min_lat = 40,
                      max_lng = -70, max_lat = 50)
```

`get_measurements` *Retrieve geological measurements*

## Description

Retrieve a range of measurements relevant to making geological inferences.

## Usage

```
get_measurements(
  measure_id = NULL,
  measurement_id = NULL,
  measurement = NULL,
  measurement_type = NULL,
  measurement_class = NULL,
  measuremeta_id = NULL,
  measure_phase = NULL,
  column_id = NULL,
  section_id = NULL,
  unit_id = NULL,
  interval_name = NULL,
  lithology_id = NULL,
  lithology_type = NULL,
  lithology_class = NULL,
  project_id = NULL,
  sf = FALSE
)
```

## Arguments

| | |
|---|---|
| `measure_id` | `integer`. Filter measurements to those of one or more unique identification number(s). |
| `measurement_id` | `integer`. Filter measurements to those of one or more measurement definition(s) as specified by their unique identification number(s). |
| `measurement` | `character`. Filter measurements to those of one or more measurement definition(s) as specified by their name (e.g., "Pb" or "transport direction"). |
| `measurement_type` | `character`. Filter measurements to those of one or more measurement definition type(s) (e.g., "major elements", "geochronological"). |
| `measurement_class` | `character`. Filter measurements to those of one or more measurement definition class(es) (e.g., "geochemical", "geophysical"). |
| `measuremeta_id` | `integer`. Filter measurements to those of one or more measurement metadata(s) (usually refers to a sample) as specified by their unique identification number(s). |
| `measure_phase` | `character`. Filter measurements by the phase from which the measurement was taken (e.g., "zircon", "whole rock"). |

| | |
|---|---|
| column_id | integer. Filter measurements to those within one or more column(s) as specified by their unique identification number(s). |
| section_id | integer. Filter measurements to those within one or more section(s) as specified by their unique identification number(s). |
| unit_id | integer. Filter measurements to those within one or more unit(s) as specified by their unique identification number(s). |
| interval_name | character. Filter measurements to those from units within or touching a named interval(s) (e.g., "Permian"). |
| lithology_id | integer. Filter measurements to those of one or more lithology(ies) as specified by their unique identification number(s). |
| lithology_type | character. Filter measurements to those of one or more lithology type(s) (e.g., "carbonate", "siliciclastic"). |
| lithology_class | character. Filter measurements to those of one or more lithology class(es) (e.g., "sedimentary", "igneous", "metamorphic"). |
| project_id | integer. Filter measurements by the unique identification number(s) of the Macrostrat project(s). |
| sf | logical. Should the results be returned as an sf object? Defaults to FALSE. |

## Details

More information can be found about the inputs and outputs for this function by using def_measurements().

## Value

A data.frame containing the following columns:

- measurement_id: The unique identification number of the measurement.
- measuremeta_id: The unique identification number of the sample.
- measurement: The name of the type of measurement obtained.
- measure_units: The units of the measurement.
- measure_phase: The phase from which the measurement was taken.
- method: The method used to generate the measurement.
- n: The number of observations or measurements.
- ref_id: The unique identification number of the reference associated with the measurement.
- sample_name: The name of the sample.
- geo_unit: The Macrostrat unit from which the measurement was taken.
- samp_lith: A lithological description of the rock from which the measurement was taken.
- samp_lith_id: The unique identification number of the lithological description of the rock from which the measurement was taken.
- samp_desc: A description of the sample used to generate the measurement.
- samp_age: The geological time interval assigned to the measurement.
- lat: Decimal degree latitude of the measurement.

- lng: Decimal degree longitude of the measurement.
- unit_id: The unique identification number of the Macrostrat unit from which the measurement was taken.
- unit_rel_pos: The relative positive of the sample within the unit.
- col_id: The unique identification number of the Macrostrat column from which the measurement was taken.
- strat_name_id: The unique identification number of the stratigraphic name attributed to the unit from which the measurement was taken.
- match_basis: A terse description of how the measuremeta data was linked to the Macrostrat unit.
- ref: The name of the reference.
- measure_value: The value of the measurement.
- measure_error: The reported error on the measurement value.
- measure_position: The position of the measurement in the Macrostrat column.
- measure_n: The number of measurements used to generate the measure_value; if greater than one, usually used to produce the measure_error.
- sample_no: The sample number for the measurement.
- error_units: The units of the error.

If sf = TRUE, an sf object is returned instead, with the same columns plus a "geometry" column that contains the spatial data.

## Developer(s)

Bethany Allen

## Reviewer(s)

Christopher D. Dean

## See Also

External data: def_drilling_sites(), def_measurements(), get_eodp(), get_fossils()

## Examples

```
# Return measurements based on their specific IDs
ex1 <- get_measurements(measure_id = c(353, 354))
# Return measurements based on the ID of a specific measurement definition
ex2 <- get_measurements(measurement_id = 42)
# Return measurements for a specific measurement type
ex3 <- get_measurements(measurement = "SiO2")
```

`get_paleogeography` *Retrieve paleogeographic geometries*

## Description

A function to retrieve paleogeographic geometries reconstructed to a given a age or interval using the Wright et al. (2013) Global Plate Model.

## Usage

```
get_paleogeography(age = NULL, interval_name = NULL)
```

## Arguments

| | |
|---|---|
| `age` | `numeric`. Return a paleogeographic reconstruction based on a numerical age in millions of years before present (between 0 and 550). |
| `interval_name` | `character`. Return a paleogeographic reconstruction based on a specified time interval. Ignored if `age` is supplied. |

## Details

More information for appropriate interval inputs for this function can be found using the `def_intervals()` function.

## Value

An `sf` object containing the geometries of the paleogeographic map.

## Developer(s)

Lewis A. Jones

## Reviewer(s)

Christopher D. Dean

## References

Wright, N., Zahirovic, S., Müller, R.D., Seton, M. (2013). Towards community-driven paleogeographic reconstructions: integrating open-access paleogeographic and paleobiology data with plate tectonics. Biogeosciences, 10, 1529–1541. doi:10.5194/bg1015292013.

## See Also

Paleogeography: `def_plates()`

## Examples

```
# Get paleogeographic map via age
ex1 <- get_paleogeography(age = 250)
# Get paleogeographic map via interval name
```

---

`get_sections` *Retrieve Macrostrat section data*

---

## Description

A function to retrieve Macrostrat units contained within gap-bound packages (sections).

## Usage

```
get_sections(
  section_id = NULL,
  column_id = NULL,
  unit_id = NULL,
  strat_name = NULL,
  strat_name_id = NULL,
  interval_name = NULL,
  interval_id = NULL,
  age = NULL,
  age_top = NULL,
  age_bottom = NULL,
  lat = NULL,
  lng = NULL,
  lithology = NULL,
  lithology_id = NULL,
  lithology_group = NULL,
  lithology_type = NULL,
  lithology_class = NULL,
  lithology_att = NULL,
  lithology_att_id = NULL,
  lithology_att_type = NULL,
  environ = NULL,
  environ_id = NULL,
  environ_type = NULL,
  environ_class = NULL,
  pbdb_collection_no = NULL,
  econ = NULL,
  econ_id = NULL,
  econ_type = NULL,
  econ_class = NULL,
  project_id = NULL,
  adjacents = FALSE
)
```

## Arguments

| | |
|---|---|
| section_id | integer. Filter sections by their unique identification number(s). |
| column_id | integer. Filter sections to those contained within column(s) as specified by their unique identification number(s). |
| unit_id | integer. Filter sections to those containing unit(s) as specified by their unique identification number(s). |
| strat_name | character. Filter sections to those containing a unit that fuzzy matches a stratigraphic name (e.g., "Hell Creek"). |
| strat_name_id | integer. Filter sections to those containing a unit that matches one or more stratigraphic name(s) as specified by their unique identification number(s). |
| interval_name | character. Filter sections to those that overlap with a named chronostratigraphic time interval (e.g., "Permian"). |
| interval_id | integer. Filter sections to those that overlap with a chronostratigraphic time interval as specified by their unique identification number(s). |
| age | numeric. Filter sections to those that overlap with the specified numerical age, in millions of years before present. |
| age_top | numeric. Filter sections to those that overlap with the age range between the specified numerical age and age_bottom. Should be in millions of years before present. age_bottom must also be specified, and this must be younger than age_bottom. |
| age_bottom | numeric. Filter sections to those that overlap with the age range between the specified numerical age and age_top. Should be in millions of years before present. age_top must also be specified, and this must be older than age_top. |
| lat | numeric. Return the sections at the specified decimal degree latitude. Must also specify lng. |
| lng | numeric. Return the sections at the specified decimal degree longitude. Must also specify lat. |
| lithology | character. Filter sections to those containing a named lithology (e.g., "shale", "sandstone"). |
| lithology_id | integer. Filter sections to those containing one or more lithology(ies) as specified by their unique identification number(s). |
| lithology_group | character. Filter sections to those containing a named lithology group (e.g., "sandstones", "mudrocks", "unconsolidated"). |
| lithology_type | character. Filter sections to those containing a named lithology type (e.g., "carbonate", "siliciclastic"). |
| lithology_class | character. Filter sections to those containing a named lithology class (e.g., "sedimentary", "igneous", "metamorphic"). |
| lithology_att | character. Filter sections to those containing a named lithology attribute (e.g., "fine", "olivine", "poorly washed"). |
| lithology_att_id | integer. Filter sections to those containing one or more lithology attribute(s) as specified by their unique identification number(s). |

`lithology_att_type`
: `character`. Filter sections to those containing a named category of lithology attribute (e.g., "grains", "lithology", "bedform").

`environ`
: `character`. Filter sections to those containing a named environment (e.g., "delta plain", "pond").

`environ_id`
: `integer`. Filter sections to those containing one or more environment(s) as specified by their unique identification number(s).

`environ_type`
: `character`. Filter sections to those containing a named environment type (e.g., "fluvial", "eolian", "glacial").

`environ_class`
: `character`. Filter sections to those containing a named environment class (e.g., "marine", "non-marine").

`pbdb_collection_no`
: `integer`. Filter sections to those containing one or more Paleobiology Database collection(s) as specified by their unique identification number(s).

`econ`
: `character`. Filter sections to those containing a named economic attribute (e.g., "brick", "ground water", "gold").

`econ_id`
: `integer`. Filter sections to those containing one or more economic attribute(s) as specified by their unique identification number(s).

`econ_type`
: `character`. Filter sections to those containing a named economic attribute type (e.g., "construction", "aquifer", "mineral").

`econ_class`
: `character`. Filter sections to those containing a named economic attribute class (e.g., "material", "water", "precious commodity").

`project_id`
: `integer`. Filter sections to those contained within one or more Macrostrat project(s) as specified by their unique identification number(s).

`adjacents`
: `logical`. If `column_id` or `lat/lng` is specified, should all sections that touch the specified column be returned? Defaults to `FALSE`.

## Details

More information can be found for the inputs for this function using the definition functions (beginning with `defs_`).

## Value

A `dataframe` containing the following columns:

- `col_id`: The unique identification number of the Macrostrat column containing the section.
- `col_area`: The area of the associated Macrostrat column in $km^2$.
- `section_id`: The unique identification number of the Macrostrat section.
- `project_id`: The unique identification number of the Macrostrat project.
- `max_thick`: The maximum thickness of the section, in meters.
- `min_thick`: The minimum thickness of the section, in meters.
- `t_age`: The age of the top of the section, estimated using the continuous time age model, in millions of years before present.

- b_age: The age of the bottom of the section, estimated using the continuous time age model, in millions of years before present.
- pbdb_collections: The number of PBDB collections contained within the section.
- lith: a dataframe containing the lithologies present within the section, with the following columns:
  - name: The named lithology (e.g., "sandstone").
  - type: The named lithology type (e.g., "siliciclastic").
  - class: The named lithology class (e.g., "sedimentary").
  - prop: The proportion of the lithology within the section, calculated from the individual Macrostrat units within the section.
  - lith_id: The unique identification number of the lithology.
- environ: a dataframe containing the environments present within the section, with the following columns:
  - name: The named environment (e.g., "delta plain").
  - type: The named environment type (e.g., "siliciclastic").
  - class: The named environment class (e.g., "marine").
  - prop: The proportion of the environment within the section, calculated from the individual Macrostrat units within the section.
  - environ_id: The unique identification number of the environment.
- econ: a dataframe containing the economic attributes present within the section, with the following columns:
  - name: The named economic attribute (e.g., "gold").
  - type: The named economic attribute type (e.g., "mineral").
  - class: The named economic attribute class (e.g., "precious commodity").
  - prop: The proportion of the economic attribute out of all economic attributes contained within the section, calculated from the individual Macrostrat units within the section.
  - econ_id: The unique identification number of the economic attribute.

## Developer(s)

Christopher D. Dean

## Reviewer(s)

Bethany Allen

## See Also

Macrostrat data entities: def_columns(), get_age_model(), get_columns(), get_units()

## Examples

```
# Get sections within a specified column
ex1 <- get_sections(column_id = 10)
# Get sections at a specific geographic coordinate
ex2 <- get_sections(lng = -110.9, lat = 48.4)
```

`get_stats` *Retrieve Macrostrat database statistics*

## Description

A function to retrieve various statistics about each Macrostrat project in the Macrostrat database. Function is called without user-specified arguments.

## Usage

```
get_stats()
```

## Value

A `data.frame` containing the following columns:

- `project_id`: The unique identification number of the Macrostrat project.
- `project`: The name of the Macrostrat project.
- `columns`: The number of Macrostrat columns associated with the project.
- `packages`: The number of Macrostrat packages/sections associated with the project.
- `units`: The number of Macrostrat units associated with the project.
- `pbdb_collections`: The number of Paleobiology Database collections associated with the project.
- `measurements`: The number of measurements associated with the project.
- `t_polys`: The number of spatial polygons associated with the project.

## Developer(s)

William Gearty

## Reviewer(s)

Bethany Allen

## See Also

Macrostrat database metadata: `def_projects()`, `def_references()`, `def_sources()`

## Examples

```
# Return all project statistics
ex1 <- get_stats()
```

get_tiles *Retrieve Macrostrat vectorized tiles*

## Description

A function to retrieve map tile data from the Macrostrat tile server. This includes geologic line features such as faults, anticlines, and moraines, as well as geologic map polygons (similar to `get_map_outcrop()` but for an entire geographic tile). The function retrieves the data in vectorized (`sf`) tile format. Note that the geographic features come from the "carto" Macrostrat scale, which is primarily for visualization purposes, and makes many assumptions about the relative priority of each source map. The layers and scales are seamlessly blended based on the chosen `zoom` level, so no scale-dependent decisions can or should be made.

## Usage

```
get_tiles(zoom = 0, x = NULL, y = NULL, combined = TRUE)
```

## Arguments

| | |
|---|---|
| `zoom` | `integer`. The zoom level of the tile(s) to retrieve. The minimum zoom level is `0`. |
| `x` | `integer`. The x index/indices of the tile(s) to retrieve. If `NULL` (the default), tiles of all x indices will be retrieved and combined. |
| `y` | `integer`. The y index/indices of the tile(s) to retrieve. If `NULL` (the default), tiles of all y indices will be retrieved and combined. |
| `combined` | `logical`. Whether all tiles should be combined. If `TRUE` (the default), the function returns a single list with all tiles combined (as described below). If `FALSE`, the function returns a list of lists, each element of which represents the data for a single tile. |

## Details

The tile indices (`x` and `y`) are zero-indexed, meaning that the first tile in each dimension is `0`. The zoom level is also zero-indexed, meaning that the first (most zoomed out) zoom level is `0`. For a given zoom level, the possible tile indices are `0` to `2 ^ zoom - 1`. Note that retrieving multiple (or all tiles) at higher zoom levels returns large (memory-wise) data objects.

Also note that two (or more) tiles may contain parts of the same geologic outcrop polygon. Since tiles also overlap slightly, these separate outcrop components may also overlap, which may lead to confusing visualizations or invalid downstream analyses. To address this, users are encouraged to union polygons that have the same `map_id` values using the `sf::st_union()` function (either with a combination of `dplyr::group_by()` and `dplyr::summarise()` or with `palaeoverse::group_apply()`, the former of which is significantly faster for a large number of tiles). We have not implemented this here to reduce the number of package dependencies and to reduce the computational time of this function.

## Value

A `list` of length two containing the following elements:

- `lines`: An `sf` object containing the geologic line features such as faults, anticlines, and moraines. It includes 8 variables:
  - `tile`: An identifier of the form "x/y" for the tile that contains the line feature. Not included if `combined = FALSE`.
  - `line_id`: The unique identifier for a line feature.
  - `source_id`: An integer that corresponds to the original map sources (see `def_sources()`).
  - `descrip`: A description of the line.
  - `name`: The name of the line, if available.
  - `direction`: A normalized direction of the line. Not commonly available.
  - `type`: A normalized line type (fault, fold, anticline, etc).
  - `geometry`: The line features as `sf` objects.
- `units`: An `sf` object containing the geologic map polygons. It includes 24 variables:
  - `tile`: An identifier of the form "x/y" for the tile that contains the polygon. Not included if `combined = FALSE`.
  - `map_id`: The unique identifier for a polygon (see [get_map_outcrop[]).
  - `source_id`: An integer that corresponds to the original map sources (see `def_sources()`).
  - `legend_id`: An integer that corresponds to the associated legend component of the original map source (see `get_map_legends()`).
  - `best_age_top`: The best top age that Macrostrat can assign to the unit based on linked resources.
  - `best_age_bottom`: The best bottom age that Macrostrat can assign to the unit based on linked resources.
  - `color`: A hex code associated with the best containing time interval for the map unit.
  - `name`: The name of the polygon. Usually either a stratigraphic name or lithological description.
  - `age`: The age indicated by the map author.
  - `lith`: Lithologies, or lithological description (see `def_lithologies()`).
  - `descrip`: A description of the map unit.
  - `comments`: Comments about the map unit.
  - `t_int_id`: The Macrostrat interval ID associated with the top age of the map unit (see `def_intervals()`).
  - `t_int`: The name of the Macrostrat interval associated with the top age of the map unit (see `def_intervals()`).
  - `b_int_id`: The Macrostrat interval ID associated with the bottom age of the map unit (see `def_intervals()`).
  - `b_int`: The name of the Macrostrat interval associated with the bottom age of the map unit (see `def_intervals()`).
  - `ref_url`: A URL to the original map source.
  - `ref_name`: The name of the original map source.
  - `ref_title`: The title of the original map source.

- ref_authors: The authors of the original map source.
- ref_source: The publication source of the original map source.
- ref_year: The publication year of the original map source.
- ref_isbn: The ISBN of the original map source.
- geometry: The polygon geometries as sf objects.

### Developer(s)

William Gearty

### Reviewer(s)

Lewis A. Jones

### See Also

Geologic maps: def_sources(), get_map_legends(), get_map_outcrop(), get_map_points()

### Examples

```
# get single carto tile of the world
ex1 <- get_tiles()
# get specific tile at a higher zoom
ex2 <- get_tiles(zoom = 6, x = 5, y = 5)
```

---

## get_units *Retrieve Macrostrat unit data*

---

### Description

A function to retrieve Macrostrat unit data matching a user-specified search criteria.

### Usage

```
get_units(
  unit_id = NULL,
  section_id = NULL,
  column_id = NULL,
  strat_name = NULL,
  strat_name_id = NULL,
  interval_name = NULL,
  interval_id = NULL,
  age = NULL,
  age_top = NULL,
  age_bottom = NULL,
  lat = NULL,
  lng = NULL,
```

```
  lithology = NULL,
  lithology_id = NULL,
  lithology_group = NULL,
  lithology_type = NULL,
  lithology_class = NULL,
  lithology_att = NULL,
  lithology_att_id = NULL,
  lithology_att_type = NULL,
  environ = NULL,
  environ_id = NULL,
  environ_type = NULL,
  environ_class = NULL,
  pbdb_collection_no = NULL,
  econ = NULL,
  econ_id = NULL,
  econ_type = NULL,
  econ_class = NULL,
  project_id = NULL,
  adjacents = FALSE,
  sf = FALSE
)
```

## Arguments

| | |
|---|---|
| `unit_id` | `integer`. Filter units by their unique identification number(s). |
| `section_id` | `integer`. Filter units to those contained within section(s) as specified by their unique identification number(s). |
| `column_id` | `integer`. Filter units to those contained within column(s) as specified by their unique identification number(s). |
| `strat_name` | `character`. Filter units to those containing a unit that fuzzy matches a stratigraphic name (e.g., "Hell Creek"). |
| `strat_name_id` | `integer`. Filter units to those that match one or more stratigraphic name(s) as specified by their unique identification number(s). |
| `interval_name` | `character`. Filter units to those that overlap with a named chronostratigraphic time interval (e.g., "Permian"). |
| `interval_id` | `integer`. Filter units to those that overlap with one or more chronostratigraphic time interval(s) as specified by their unique identification number(s). |
| `age` | `numeric`. Filter units to those that overlap with the specified numerical age, in millions of years before present. |
| `age_top` | `numeric`. Filter units to those that overlap with the age range between the specified numerical age and `age_bottom`. Should be in millions of years before present. `age_bottom` must also be specified, and `age_top` must be younger than `age_bottom`. |
| `age_bottom` | `numeric`. Filter units to those that overlap with the age range between the specified numerical age and `age_top`. Should be in millions of years before present. `age_top` must also be specified, and `age_bottom` must be older than `age_top`. |

`lat` numeric. Return the units at the specified decimal degree latitude. Must also specify `lng`.

`lng` numeric. Return the units at the specified decimal degree longitude. Must also specify `lat`.

`lithology` character. Filter units to those containing a named lithology (e.g., "shale", "sandstone").

`lithology_id` integer. Filter units to those containing one or more lithology(ies) as specified by their unique identification number(s).

`lithology_group` character. Filter units to those containing a named lithology group (e.g., "sandstones", "mudrocks", "unconsolidated").

`lithology_type` character. Filter units to those containing a named lithology type (e.g., "carbonate", "siliciclastic").

`lithology_class` character. Filter units to those containing a named lithology class (e.g., "sedimentary", "igneous", "metamorphic").

`lithology_att` character. Filter units to those containing a named lithology attribute (e.g., "fine", "olivine", "poorly washed").

`lithology_att_id` integer. Filter units to those containing one or more lithology attribute(s) as specified by their unique identification number(s).

`lithology_att_type` character. Filter units to those containing a named category of lithology attribute (e.g., "grains", "lithology", "bedform").

`environ` character. Filter units to those containing a named environment (e.g., "delta plain", "pond").

`environ_id` integer. Filter units to those containing one or more environment(s) as specified by their unique identification number(s).

`environ_type` character. Filter units to those containing a named environment type (e.g., "fluvial", "eolian", "glacial").

`environ_class` character. Filter units to those containing a named environment class (e.g., "marine", "non-marine").

`pbdb_collection_no` integer. Filter units to those containing one or more Paleobiology Database collection(s) as specified by their unique identification number(s).

`econ` character. Filter units to those containing a named economic attribute (e.g., "brick", "ground water", "gold").

`econ_id` integer. Filter units to those containing one or more economic attribute(s) as specified by their unique identification number(s).

`econ_type` character. Filter units to those containing a named economic attribute type (e.g., "construction", "aquifer", "mineral").

`econ_class` character. Filter units to those containing a named economic attribute class (e.g., "material", "water", "precious commodity").

| | |
|---|---|
| project_id | integer. Filter units to those contained within one or more Macrostrat project(s) as specified by their unique identification number(s). |
| adjacents | logical. If column_id or lat/lng is specified, should all units that touch the specified column be returned? Defaults to FALSE. |
| sf | logical. Should the results be returned as an sf object? Defaults to FALSE. |

## Details

More information can be found for the inputs for this function using the definition functions (beginning with def_).

## Value

A data.frame containing the following columns:

- unit_id: The unique identification number of the Macrostrat unit.
- section_id: The unique identification number of the Macrostrat section containing the unit.
- col_id: The unique identification number of the Macrostrat column containing the unit.
- project_id: The unique identification number of the Macrostrat project.
- col_area: The area of the associated Macrostrat column in $km^2$.
- unit_name: The name of the Macrostrat unit.
- strat_name_id: The unique Macrostrat stratigraphic name ID.
- Mbr: The lithostratigraphic member.
- Fm: The lithostratigraphic formation.
- Gp: The lithostratigraphic group.
- SGp: The lithostratigraphic supergroup.
- t_age: The age of the top of the unit, estimated using the continuous time age model, in millions of years before present.
- b_age: The age of the bottom of the unit, estimated using the continuous time age model, in millions of years before present.
- max_thick: The maximum thickness of the unit, in meters.
- min_thick: The minimum thickness of the unit, in meters.
- outcrop: Type of exposure ("outcrop", "subsurface", or "both").
- pbdb_collections: The number of Paleobiology Database collections contained within the unit.
- pbdb_occurrences: The number of Paleobiology Database occurrences contained within the unit.
- lith: a dataframe containing the lithologies present within the unit, with the following columns:
  - name: The named lithology (e.g., "sandstone").
  - prop: The proportion of the lithology within the unit, calculated from the individual Macrostrat sub-units within the unit.

  - lith_id: The unique identification number of the lithology.
  - type: The named lithology type (e.g., "siliciclastic").
  - class: The named lithology class (e.g., "sedimentary").
- environ: a dataframe containing the environments present within the unit, with the following columns:
  - class: The named environment class (e.g., "marine").
  - type: The named environment type (e.g., "siliciclastic").
  - name: The named environment (e.g., "delta plain").
  - environ_id: The unique identification number of the environment.
- econ: a dataframe containing the economic attributes present within the unit, with the following columns:
  - type: The named economic attribute type (e.g., "mineral").
  - name: The named economic attribute (e.g., "gold").
  - econ_id: The unique identification number of the economic attribute.
  - class: The named economic attribute class (e.g., "precious commodity").
- measure: a dataframe containing the measure attributes present within the unit, with the following columns:
  - measure_class: The class of measures (e.g., "geochemical").
  - measure_type: the type of measures (e.g., "minor elements").
- notes: Unit specific notes.
- color: Recommended coloring for units based on dominant lithology.
- text_color: Recommended coloring for text based on color.
- t_int_id: The ID of the chronostratigraphic interval containing the top boundary of the unit.
- t_int_name: The name of the time interval represented at the top of the unit.
- t_int_age: The age of the chronostratigraphic interval containing the top boundary of the unit.
- t_prop: Position of continuous time age model top boundary, proportional to reference time interval (t_int_name).
- units_above: The unit_ids of the units contacting the top of the unit.
- b_int_id: The ID of the chronostratigraphic interval containing the bottom boundary of the unit.
- b_int_name: The name of the time interval represented at the bottom of the unit.
- b_int_age: The age of the chronostratigraphic interval containing the bottom boundary of the unit.
- b_prop: Position of continuous time age model bottom boundary, proportional to reference time interval (b_interval).
- units_below: The unit_ids of the units contacting the bottom of the unit.
- strat_name_long: The stratigraphic name associated with the unit (e.g., "Coal Valley Formation").
- refs: The unique identification number(s) of the reference(s) associated with the unit.

- clat: The present day latitude of the centroid of the column to which the unit belongs.
- clng: The present day longitude of the centroid of the column to which the unit belongs.
- t_plat: The paleolatitude of the centroid of the column which the unit belongs to at top age (clat rotated to t_age).
- t_plng: The paleolongitude of the centroid of the column which the unit belongs to at top age (clng rotated to t_age) .
- b_plat: The paleolatitude of the centroid of the column which the unit belongs to at bottom age (clat rotated to b_age).
- b_plng: The paleolongitude of the the centroid of the column which the unit belongs to at bottom age (clng rotated to b_age).

If sf is TRUE, an sf object is returned instead, with a "geometry" column that contains the spatial data instead of the clat/clng columns.

## Developer(s)

Lewis A. Jones

## Reviewer(s)

William Gearty

## See Also

Macrostrat data entities: def_columns(), get_age_model(), get_columns(), get_sections()

## Examples

```
# Get units with a specific stratigraphic name
ex1 <- get_units(strat_name = "Hell Creek")
# Get units at a specific geographic coordinate
ex2 <- get_units(lng = -110.9, lat = 48.4)
```

# Index