Package ‘rATTAINS’

November 3, 2021

Title Access EPA ‘ATTAINS’ Data
Version 0.1.3
Description An R interface to United States Environmental Protection Agency (EPA) Assessment, Total Maximum Daily Load (TMDL) Tracking and Implementation System (‘ATTAINS’) data. ‘ATTAINS’ is the EPA database used to track information provided by states about water quality assessments conducted under federal Clean Water Act requirements. ATTAINS information and API information is available at <https://www.epa.gov/waterdata/attains>.
License MIT + file LICENSE
Encoding UTF-8
RoxygenNote 7.1.1
Imports checkmate, crul, curl, dplyr, fauxpas, fs, hoardr, janitor, magrittr, purrr, rlang, rlist, tibble, tidyjson, tidyr
Suggests vcr (>= 0.6.0), testthat (>= 3.0.0), knitr, rmarkdown, webmockr
Config/testthat/edition 3
BugReports https://github.com/mps9506/rATTAINS/issues
VignetteBuilder knitr
NeedsCompilation no
Author Michael Schramm [aut, cre, cph]
  (<https://orcid.org/0000-0003-1876-6592>)
Maintainer Michael Schramm <mpschramm@gmail.com>
Repository CRAN
Date/Publication 2021-11-03 14:10:02 UTC

R topics documented:

  actions ................................................................. 2
  assessments .......................................................... 4
  assessment_units ..................................................... 7
actions

domain_values ......................................................... 9
huc12_summary .......................................................... 10
plans ................................................................. 11
rATTAINS_caching ...................................................... 12
rATTAINS_options ....................................................... 13
state_summary ........................................................... 14
surveys ................................................................. 15

Index 17

actions Download Actions Data

Description

Provides data about actions (TMDLs, 4B Actions, Alternative Actions, Protection Approach Actions) that have been finalized.

Usage

```
actions(
  action_id = NULL,
  assessment_unit_id = NULL,
  state_code = NULL,
  organization_id = NULL,
  summarize = FALSE,
  parameter_name = NULL,
  pollutant_name = NULL,
  action_type_code = NULL,
  agency_code = NULL,
  pollutant_source_code = NULL,
  action_status_code = NULL,
  completion_date_later_than = NULL,
  completion_date_earlier_than = NULL,
  tmdl_date_later_than = NULL,
  tmdl_date_earlier_then = NULL,
  last_change_later_than_date = NULL,
  last_change_earlier_than_date = NULL,
  return_count_only = FALSE,
  tidy = TRUE,
  ...
)
```

Arguments

```
action_id       (character) Specifies what action to retrieve. multiple values allowed. optional
```
`actions`

- `assessment_unit_id` (character) Filters returned actions to those associated with the specified assessment unit identifier, plus any statewide actions. Multiple values allowed. Optional.
- `state_code` (character) Filters returned actions to those "belonging" to the specified state. Optional.
- `organization_id` (character) Filters returned actions to those "belonging" to specified organizations. Multiple values allowed. Optional.
- `summarize` (logical) If TRUE provides only a count of the assessment units for the action and summary of the pollutants and parameters covered by the action.
- `parameter_name` (character) Filters returned actions to those associated with the specified parameter. Multiple values allowed. Optional.
- `pollutant_name` (character) Filters returned actions to those associated with the specified pollutant. Multiple values allowed. Optional.
- `action_type_code` (character) Filters returned actions to those associated with the specified action type code. Multiple values allowed. Optional.
- `agency_code` (character) Filters returned actions to those with the specified agency code. Multiple values allowed. Optional.
- `pollutant_source_code` (character) Filters returned actions to those matching the specified pollutant source code. Multiple values allowed. Optional.
- `action_status_code` (character) Filters returned actions to those matching the specified action status code. Multiple values allowed. Optional.
- `completion_date_later_than` (character) Filters returned actions to those with a completion date later than the value specified. Must be a character formatted as "YYYY-MM-DD". Optional.
- `completion_date_earlier_than` (character) Filters returned actions to those with a completion date earlier than the value specified. Must be a character formatted as "YYYY-MM-DD". Optional.
- `tmdl_date_later_than` (character) Filters returned actions to those with a TMDL date later than the value specified. Must be a character formatted as "YYYY-MM-DD". Optional.
- `tmdl_date_earlier_than` (character) Filters returned actions to those with a TMDL date earlier than the value specified. Must be a character formatted as "YYYY-MM-DD". Optional.
- `last_change_later_than_date` (character) Filters returned actions to those with a last change date later than the value specified. Can be used with `last_change_earlier_than_date` to return actions changed within a date range. Must be a character formatted as "YYYY-MM-DD". Optional.
last_change_earlier_than_date  
(character) Filters returned actions to those with a last change date earlier than the value specified. Can be used with last_change_later_than_date to return actions changed within a date range. Must be a character formatted as "YYYY-MM-DD". optional

return_count_only  
(logical) If TRUE returns only the count of actions the match the query.

tidy  
(logical) TRUE (default) the function returns a tidied tibble. FALSE the function returns the raw JSON string.

...  
list of curl options passed to `crl::HttpClient()`

Details

One or more of the following arguments must be included: action_id, assessment_unit_id, state_code or organization_id. Multiple values are allowed for indicated arguments and should be included as a comma separated values in the string (eg. organization_id="TCEQMAIN,DCOEE").

Value

If count = TRUE returns a tibble that summarizes the count of actions returned by the query. If count = FALSE returns a list of tibbles including documents and actions identified by the query. If tidy = FALSE the raw JSON string is returned, else the JSON data is parsed and returned as tibbles.

Note

See `domain_values` to search values that can be queried. Data downloaded from the EPA webservice is automatically cached to reduce unnecessary calls to the server. To managed cached files see `rATTAINS_caching`

Examples

```r
## Not run:
## Look up an individual action
actions(action_id = "R8-ND-2018-03")
## Get the JSON instead
actions(action_id = "R8-ND-2018-03", tidy = FALSE)
## End(Not run)
```

Description

Download Assessment Decisions
assessments

Usage

assessments(
  assessment_unit_id = NULL,
  state_code = NULL,
  organization_id = NULL,
  reporting_cycle = NULL,
  use = NULL,
  use_support = NULL,
  parameter = NULL,
  parameter_status_name = NULL,
  probable_source = NULL,
  agency_code = NULL,
  ir_category = NULL,
  state_ir_category_code = NULL,
  multicategory_search = NULL,
  last_change_later_than_date = NULL,
  last_change_earlier_than_date = NULL,
  return_count_only = FALSE,
  exclude_assessments = FALSE,
  tidy = TRUE,
...)

Arguments

assessment_unit_id
  (character) Specify the specific assessment unit assessment data to return. Multiple values can be provided. optional

state_code
  (character) Filters returned assessments to those from the specified state. optional

organization_id
  (character) Filters the returned assessments to those belonging to the specified organization. optional

reporting_cycle
  (character) Filters the returned assessments to those for the specified reporting cycle. The reporting cycle refers to the four-digit year that the reporting cycle ended. Defaults to the current cycle. optional

use
  (character) Filters the returned assessments to those with the specified uses. Multiple values can be provided. optional

use_support
  (character) Filters returned assessments to those fully supporting the specified uses or that are threatened. Multiple values can be provided. Allowable values include "X"= Not Assessed, "I"= Insufficient Information, "F"= Fully Supporting, "N"= Not Supporting, and "T"= Threatened. optional

parameter
  (character) Filters the returned assessments to those with one or more of the specified parameters. Multiple values can be provided. optional
assessments

parameter_status_name
(character) Filters the returned assessments to those with one or more associated parameters meeting the provided value. Valid values are "Meeting Criteria", "Cause", "Observed Effect". Multiple values can be provided. optional

probable_source
(character) Filters the returned assessments to those having the specified probable source. Multiple values can be provided. optional

agency_code
(character) Filters the returned assessments to those by the type of agency responsible for the assessment. Allowed values are "E" = EPA, "S" = State, "T" = Tribal. optional

ir_category
(character) Filters the returned assessments to those having the specified IR category. Multiple values can be provided. optional

state_ir_category_code
(character) Filters the returned assessments to include those having the provided codes.

multicategory_search
(character) Specifies whether to search at multiple levels. If this parameter is set to "Y" then the query applies the EPA IR Category at the Assessment, UseAttainment, and Parameter levels; if the parameter is set to "N" it looks only at the Assessment level.

last_change_later_than_date
(character) Filters the returned assessments to only those last changed after the provided date. Must be a character with format: "yyyy-mm-dd". optional

last_change_earlier_than_date
(character) Filters the returned assessments to only those last changed before the provided date. Must be a character with format: "yyyy-mm-dd". optional

return_count_only
(logical) If TRUE returns only the count of actions the match the query. defaults to FALSE

exclude_assessments
(logical) If TRUE returns only the documents associated with the Assessment cycle instead of the assessment data. Defaults is FALSE.

tidy
(logical) TRUE (default) the function returns a tidied tibble. FALSE the function returns the raw JSON string.

... list of curl options passed to crul::HttpClient()

Details

One or more of the following arguments must be included: action_id, assessment_unit_id, state_code or organization_id. Multiple values are allowed for indicated arguments and should be included as a comma separated values in the string (eg. organization_id="TCEQMAIN,DCOEE").

Value

If count = TRUE returns a tibble that summarizes the count of actions returned by the query. If count = FALSE returns a list of tibbles including documents, use assessment data, and parameters assessment data identified by the query. If tidy = FALSE the raw JSON string is returned, else the JSON data is parsed and returned as tibbles.
Note
See domain_values to search values that can be queried. Data downloaded from the EPA webservice is automatically cached to reduce unnecessary calls to the server. To managed cached files see rATTAINS_caching.

Examples

## Not run:

## Return all assessment decisions with specified parameters
assessments(organization_id = "SDDENR",
probable_source = "GRAZING IN RIPARIAN OR SHORELINE ZONES")

## Returns the raw JSONs instead:
assessments(organization_id = "SDDENR",
probable_source = "GRAZING IN RIPARIAN OR SHORELINE ZONES", tidy = FALSE)

## End(Not run)

---

assessment_units Download Assessment Unit Summary

Description
Provides basic information about the requested assessment units.

Usage

assessment_units(
  assessment_unit_identifier = NULL,
  state_code = NULL,
  organization_id = NULL,
  epa_region = NULL,
  huc = NULL,
  county = NULL,
  assessment_unit_name = NULL,
  last_change_later_than_date = NULL,
  last_change_earlier_than_date = NULL,
  status_indicator = NULL,
  return_count_only = NULL,
  tidy = TRUE,
  ...
)
Arguments

- **assessment_unit_identifier** (character) Filters returned assessment units to one or more specific assessment units. Multiple values can be provided. optional
- **state_code** (character) Filters returned assessment units to only those having a state code matches one in the provided list of states. Multiple values can be provided. optional
- **organization_id** (character) Filters returned assessment units to only those having a matching organization ID. Multiple values can be provided. optional
- **epa_region** (character) Filters returned assessment units to only matching EPA regions. Multiple values can be provided. optional
- **huc** (character) Filters returned assessment units to only those which have a location type of HUC and the location value matches the provided HUC. Multiple values can be provided. optional
- **county** (character) Filters returned assessment units to only those which have a location type of county and matches the provided county. Multiple values can be provided. optional
- **assessment_unit_name** (character) Filters the returned assessment units to matching the provided value.
- **last_change_later_than_date** (character) Filters returned assessment units to those only changed after the provided date. Must be a character with format: "yyyy-mm-dd". optional
- **last_change_earlier_than_date** (character) Filters returned assessment units to those only changed before the provided date. Must be a character with format: "yyyy-mm-dd". optional
- **status_indicator** (character) Filter the returned assessment units to those with specified status. "A" for active, "R" for retired. optional
- **return_count_only** (character) "Y" for yes, "N" for no. Defaults to "N". optional
- **tidy** (logical) TRUE (default) the function returns a tidied tibble. FALSE the function returns the raw JSON string.

... list of curl options passed to `crul::HttpClient()`

Details

One or more of the following arguments must be included: `assessment_unit_identifier`, `state_code` or `organization_id`. Multiple values are allowed for indicated arguments and should be included as a comma separated values in the string (eg. `organization_id="TCEQMAIN,DCOEE"`).

Value

When `tidy = TRUE` a tibble with many variables, some nested, is returned. When `tidy=False` a raw JSON string is returned.
Note

See `domain_values` to search values that can be queried. Data downloaded from the EPA webservice is automatically cached to reduce unnecessary calls to the server. To managed cached files see `rATTAINS_caching`

Examples

```r
## Not run:
## Retrieve data about a single assessment unit
assessment_units(assessment_unit_identifier = "AL03150201-0107-200"
## Retrieve data as a JSON instead
assessment_units(assessment_unit_identifier = "AL03150201-0107-200", tidy = FALSE)
## End(Not run)
```

## Download Domain Values

### Description

Provides information on allowed parameter values in ATTAINS.

### Usage

```r
domain_values(domain_name = NULL, context = NULL, tidy = TRUE, ...)
```

### Arguments

- `domain_name` (character) Specified the domain name to obtain valid parameter values for. Defaults to `NULL` which will a tibble with all the domain names. To return the allowable parameter values for a given domain, the domain should be specified here. optional.
- `context` (character) When specified, the service will return domain_name values alongside the context. optional.
- `tidy` (logical) TRUE (default) the function returns a tidied tibble. FALSE the function returns the raw JSON string.
- `...` list of curl options passed to `crul::HttpClient()`

### Value

If `tidy = FALSE` the raw JSON string is returned, else the JSON data is parsed and returned as a tibble.
Note

Data downloaded from the EPA webservice is automatically cached to reduce unnecessary calls to
the server. To managed cached files see rATTAINS_caching

Examples

```r
## Not run:
## return a tibble with all domain names
domain_values()

## return allowable parameter values for a given domain name and context
domain_values(domain_name="UseName",context="TCEQMAIN")

## return the query as a JSON string instead
domain_values(domain_name="UseName",context="TCEQMAIN", tidy = FALSE)
## End(Not run)
```

Description

Provides summary data for a 12-digit Hydrologic Unit Code (HUC12), based on Assessment Units
in the HUC12. Watershed boundaries may cross state boundaries, so the service may return as-
sessment units from multiple organizations. Returns the assessment units in the HUC12, size and
percentages of assessment units considered Good, Unknown, or Impaired.

Usage

```r
huc12_summary(huc, tidy = TRUE, ...)
```

Arguments

- `huc` (character) Specifies the 12-digit HUC to be summarized. required
- `tidy` (logical) TRUE (default) the function returns a tidied tibble. FALSE the function returns the raw JSON string.
- `...` list of curl options passed to `curl::HttpClient()`

Value

If `tidy = FALSE` the raw JSON string is returned, else the JSON data is parsed and returned as a list
of tibbles that include a list of seven tibbles.
plans

Note

See `domain_values` to search values that can be queried. Data downloaded from the EPA webservice is automatically cached to reduce unnecessary calls to the server. To managed cached files see `rATTAINS_caching`

Examples

```r
## Not run:
## Return a list of tibbles with summary data about a single huc12
x <- huc12_summary(huc = "020700100204")

## Return as a JSON string
x <- huc12_summary(huc = "020700100204", tidy = TRUE)
## End(Not run)
```

plans

Download Plans and Actions by HUC

Description

Returns information about plans or actions (TMDLs, 4B Actions, Alternative Actions, Protective Approach Actions) that have been finalized. This is similar to `actions` but returns data by HUC code and any assessment units covered by a plan or action within the specified HUC.

Usage

```r
plans(huc, organization_id = NULL, summarize = FALSE, tidy = TRUE, ...)
```

Arguments

- **huc** (character) Filters the returned actions by 8-digit or higher HUC. required
- **organization_id** (character). Filters the returned actions by those belonging to the specified organization. Multiple values can be used. optional
- **summarize** (logical) If TRUE the count of assessment units is returned rather than the assessment unit identifiers for each action. Defaults to FALSE.
- **tidy** (logical) TRUE (default) the function returns a list of tibbles. FALSE the function returns the raw JSON string.
- ... list of curl options passed to `curl::HttpClient()`

Details

huc is a required argument. Multiple values are allowed for indicated arguments and should be included as a comma separated values in the string (eg. `organization_id="TCEQMAIN,DCOEE"`).
Value

If count = TRUE returns a tibble that summarizes the count of actions returned by the query. If count = FALSE returns a list of tibbles including documents, use assessment data, and parameters assessment data identified by the query. If tidy = FALSE the raw JSON string is returned, else the JSON data is parsed and returned as a list of tibbles.

Note

See domain_values to search values that can be queried. Data downloaded from the EPA webservice is automatically cached to reduce unnecessary calls to the server. To managed cached files see rATTAINS_caching

Examples

```r
## Not run:
## Query plans by huc
plans(huc = "020700100103")

## return a JSON string instead of list of tibbles
plans(huc = "020700100103", tidy = FALSE)

## End(Not run)
```

Description

Manage data caches

Details

To get the cache directory for a data source, see the method x$cache_path_get()

cache_delete only accepts 1 file name, while cache_delete_all doesn’t accept any names, but deletes all files. For deleting many specific files, use cache_delete in a `lapply()` type call

Note that cached files will continue to be used until they are deleted. You should occasionally delete all cached files.

Value

None
Useful user functions

Assuming x is a HoardClient class object, e.g., actions_cache

- x$cache_path_get() get cache path
- x$cache_path_set() set cache path
- x$list() returns a character vector of full path file names
- x$files() returns file objects with metadata
- x$details() returns files with details
- x$delete() delete specific files
- x$delete_all() delete all files, returns nothing

Caching objects for each data source

- actions(): actions_cache
- assessments(): assessments_cache
- assessment_units(): au_cache
- domain_values(): dv_cache
- huc12_summary(): huc12_cache
- plans(): plans_cache
- state_summary(): state_cache
- surveys(): surveys_cache

rATTAINS_options

Description

rATTAINS options

Usage

rATTAINS_options(cache_downloads = FALSE)

Arguments

cache_downloads

(logical) whether to locally cache downloads. default: FALSE

Details

rATTAINS package level options; stored in an internal package environment rATTAINSenv
state_summary

Value

None

See Also

rATTAINS_caching for managing cached files

Examples

```r
## Not run:
rATTAINS_options(cache_downloads = FALSE)

## End(Not run)
```

---

### state_summary

**Download State Summaries**

#### Description

Provides summary information for assessed uses for an organization (State, Territory or Tribe) and Integrated Reporting Cycle. The Organization ID for the state, territory or tribe is required. If a Reporting Cycle isn’t provided, the service will return the most recent cycle. If a reporting Cycle is provided, the service will return a summary for the requested cycle.

#### Usage

```r
state_summary(organization_id = NULL, reporting_cycle = NULL, tidy = TRUE, ...)
```

#### Arguments

- **organization_id** (character) Restricts results to the specified organization. required
- **reporting_cycle** (character) Filters the returned results to the specified 4 digit reporting cycle year. Typically even numbered years. Will return reporting data for all years prior to and including the reporting cycle by reporting cycle. optional
- **tidy** (logical) TRUE (default) the function returns a tidied tibble. FALSE the function returns the raw JSON string.
- **...** list of curl options passed to `crul::HttpClient()`

#### Value

If tidy = FALSE the raw JSON string is returned, else the JSON data is parsed and returned as a list of tibbles.
surveys

Note

See domain_values to search values that can be queried. Data downloaded from the EPA webservice is automatically cached to reduce unnecessary calls to the server. To managed cached files see rATTAINS_caching.

Examples

```r
## Not run:
## Get a list of tibbles summarizing assessed uses
state_summary(organization_id = "TDECWR", reporting_cycle = "2016")

## Returns the query as a JSON string instead
state_summary(organization_id = "TDECWR", reporting_cycle = "2016", tidy = FALSE)

## End(Not run)
```

surveys

Download State Survey Results

Description

Downloads data about state statistical (probability) survey results.

Usage

```r
surveys(organization_id = NULL, survey_year = NULL, tidy = TRUE, ...)
```

Arguments

- `organization_id` (character) Filters the list to only those “belonging to” one of the specified organizations. Multiple values may be specified. required
- `survey_year` (character) Filters the list to the year the survey was performed. optional.
- `tidy` (logical) TRUE (default) the function returns a tidied tibble. FALSE the function returns the raw JSON string.
- `...` list of curl options passed to `crul::HttpClient()`

Details

Arguments that allow multiple values should be entered as a comma separated string with no spaces (organization_id = "DOEE,21AWIC").

Value

If tidy = FALSE the raw JSON string is returned, else the JSON data is parsed and returned as a list of tibbles.
Note

See `domain_values` to search values that can be queried. Data downloaded from the EPA webservice is automatically cached to reduce unnecessary calls to the server. To manage cached files see `rATTAINS_caching`.

Examples

```r
## Not run:

## return surveys by organization
surveys(organization_id="SDDENR")

## return as a JSON string instead of a list of tibbles
surveys(organization_id="SDDENR", tidy = FALSE)

## End(Not run)
```
Index

* datasets
  * rATTAINS_caching, 12

actions, 2, 11
actions_cache (rATTAINS_caching), 12
assessment_units, 7
assessments, 4
assessments_cache (rATTAINS_caching), 12
au_cache (rATTAINS_caching), 12

crul::HttpClient(), 4, 6, 8–11, 14, 15

domain_values, 4, 7, 9, 11, 12, 15, 16
dv_cache (rATTAINS_caching), 12

huc12_cache (rATTAINS_caching), 12
huc12_summary, 10

lapply(), 12

plans, 11
plans_cache (rATTAINS_caching), 12

rATTAINS_caching, 4, 7, 9–12, 12, 14–16
rATTAINS_options, 13

state_cache (rATTAINS_caching), 12
state_summary, 14
surveys, 15
surveys_cache (rATTAINS_caching), 12