Package ‘ECOTOXr’

October 5, 2021

Type Package

Title Download and Extract Data from US EPA's ECOTOX Database

Version 0.1.1

Date 2021-10-04

Author Pepijn de Vries [aut, cre, dtc]

Maintainer Pepijn de Vries <pepijn.devries@outlook.com>

Description The US EPA ECOTOX database is a freely available database
with a treasure of aquatic and terrestrial ecotoxicological data.
As the online search interface doesn't come with an API, this
package provides the means to easily access and search the database
in R. To this end, all raw tables are downloaded from the EPA website
and stored in a local SQLite database.

Depends R (>= 3.5.0), RSQLite

Imports crayon, dplyr, rappdirs, readr, rvest, stringr, utils

Suggests testthat (>= 3.0.0), webchem

URL <https://github.com/pepijn-devries/ECOTOXr>

BugReports https://github.com/pepijn-devries/ECOTOXr/issues

License GPL (>= 3)

Encoding UTF-8

RoxygenNote 7.1.2

Config/testthat/edition 3

NeedsCompilation no

Repository CRAN

Date/Publication 2021-10-05 08:10:12 UTC

R topics documented:

build_ecotox_sqlite *Build an SQLite database from zip archived tables downloaded from EPA website*

Description

This function is called automatically after `download_ecotox_data`. The database files can also be downloaded manually from the EPA website from which a local database can be build using this function.

Usage

```
build_ecotox_sqlite(source, destination = get_ecotox_path(), write_log = TRUE)
```

Arguments

`source`	A `character` string pointing to the directory path where the text files with the raw tables are located. These can be obtained by extracting the zip archive from `https://cfpub.epa.gov/ecotox/` and look for 'Download ASCII Data'.
`destination`	A `character` string representing the destination path for the SQLite file. By default this is `get_ecotox_path()`.
`write_log`	A `logical` value indicating whether a log file should be written in the destination path `TRUE`. The log contains information on the source and destination path, the version of this package, the creation date, and the operating system on which the database was created.

Details

Raw data downloaded from the EPA website is in itself not very efficient to work with in R. The files are large and would put a large strain on R when loading completely into the system's memory. Instead use this function to build an SQLite database from the tables. That way, the data can be queried without having to load it all into memory.

EPA provides the raw table from the ECOTOX database as text files with pipe-characters ('|') as table column separators. Although not documented, the tables appear not to contain comment or quotation characters. There are records containing the reserved pipe-character that will confuse the table parser. For these records, the pipe-character is replaced with a dash character ('-').

In addition, while reading the tables as text files, this package attempts to decode the text as UTF8. Unfortunately, this process appears to be platform-dependent, and may therefore result in different end-results on different platforms. This problem only seems to occur for characters that are listed as 'control characters' under UTF8. This will have consequences for reproducibility, but only if you build search queries that look for such special characters. It is therefore advised to stick to common (non-accented) alpha-numerical characters in your searches, for the sake of reproducibility.

Value

Returns `NULL` invisibly.

Author(s)

Pepijn de Vries

Examples

```
## Not run:
## This example will only work properly if 'dir' points to an existing directory
## with the raw tables from the ECOTOX database. This function will be called
## automatically after a call to 'download_ecotox_data()'.
test <- check_ecotox_availability()
if (test) {
  files   <- attributes(test)$files[1,]
  dir     <- gsub(".sqlite", "", files$database, fixed = T)
  path    <- files$path
  if (dir.exists(file.path(path, dir))) {
    build_ecotox_sqlite(source = file.path(path, dir), destination = get_ecotox_path())
  }
}

## End(Not run)
```

`check_ecotox_availability` *Check whether a ECOTOX database exists locally*

Description

Tests whether a local copy of the US EPA ECOTOX database exists in `get_ecotox_path`.

Usage

```
check_ecotox_availability(target = get_ecotox_path())
```

Arguments

`target` A `character` string specifying the path where to look for the database file.

Details

When arguments are omitted, this function will look in the default directory (get_ecotox_path). However, it is possible to build a database file elsewhere if necessary.

Value

Returns a logical value indicating whether a copy of the database exists. It also returns a files attribute that lists which copies of the database are found.

Author(s)

Pepijn de Vries

Examples

```
check_ecotox_availability()
```

cite_ecotox *Cite the downloaded copy of the ECOTOX database*

Description

Cite the downloaded copy of the ECOTOX database and this package for reproducible results.

Usage

```
cite_ecotox(path = get_ecotox_path(), version)
```

Arguments

path	A character string with the path to the location of the local database (default is get_ecotox_path()).
version	A character string referring to the release version of the database you wish to locate. It should have the same format as the date in the EPA download link, which is month, day, year, separated by underscores ("%m_%d_%Y"). When missing, the most recent available copy is selected automatically.

Details

When you download a copy of the EPA ECOTOX database using download_ecotox_data(), a BibTex file is stored that registers the database release version and the access (= download) date. Use this function to obtain a citation to that specific download.

In order for others to reproduce your results, it is key to cite the data source as accurately as possible.

Value

Returns a vector of bibentry's, containing a reference to the downloaded database and this package.

Author(s)

Pepijn de Vries

Examples

```
## Not run:
## In order to cite downloaded database and this package:
cite_ecotox()

## End(Not run)
```

---

dbConnectEcotox *Open or close a connection to the local ECOTOX database*

Description

Wrappers for dbConnect and dbDisconnect methods.

Usage

```
dbConnectEcotox(path = get_ecotox_path(), version, ...)

dbDisconnectEcotox(conn, ...)
```

Arguments

path	A character string with the path to the location of the local database (default is get_ecotox_path()).
version	A character string referring to the release version of the database you wish to locate. It should have the same format as the date in the EPA download link, which is month, day, year, separated by underscores ("%m_%d_%Y"). When missing, the most recent available copy is selected automatically.
...	Arguments that are passed to dbConnect method or dbDisconnect method.
conn	An open connection to the ECOTOX database that needs to be closed.

Details

Open or close a connection to the local ECOTOX database. These functions are only required when you want to send custom queries to the database. For most searches the search_ecotox function will be adequate.

Value

A database connection in the form of a DBIConnection-class object. The object is tagged with: a time stamp; the package version used; and the file path of the SQLite database used in the connection. These tags are added as attributes to the object.

Author(s)

Pepijn de Vries

Examples

```
## Not run:
## This will only work when a copy of the database exists:
con <- dbConnectEcotox()

## check if the connection works by listing the tables in the database:
dbListTables(con)

## Let's be a good boy/girl and close the connection to the database when we're done:
dbDisconnectEcotox(con)

## End(Not run)
```

download_ecotox_data *Download and extract ECOTOX database files and compose database*

Description

In order for this package to fully function, a local copy of the ECOTOX database needs to be build. This function will download the required data and build the database.

Usage

```
download_ecotox_data(target = get_ecotox_path(), write_log = TRUE, ask = TRUE)
```

Arguments

target	Target directory where the files will be downloaded and the database compiled. Default is `get_ecotox_path()`.
write_log	A `logical` value indicating whether a log file should be written to the target path `TRUE`.
ask	There are several steps in which files are (potentially) overwritten or deleted. In those cases the user is asked on the command line what to do in those cases. Set this parameter to `FALSE` in order to continue without warning and asking.

Details

This function will attempt to find the latest download url for the ECOTOX database from the EPA website. When found it will attempt to download the zipped archive containing all required data. This data is than extracted and a local copy of the database is build.

Value

Returns `NULL` invisibly.

Known issues

On some machines this function fails to connect to the database download URL from the EPA website due to missing SSL certificates. Unfortunately, there is no easy fix for this in this package. A work around is to download and unzip the file manually using a different machine or browser that is less strict with SSL certificates. You can then call `build_ecotox_sqlite()` and point the `source` location to the manually extracted zip archive.

Author(s)

Pepijn de Vries

Examples

```
## Not run:
download_ecotox_data()

## End(Not run)
```

`ECOTOXr` *Package description*

Description

Everything you need to know when you start using the ECOTOXr package.

Details

The ECOTOXr provides the means to efficiently search, extract and analyse US EPA ECOTOX data, with a focus on reproducible results. Although the package creator/maintainer is confident in the quality of this software, it is the end users sole responsibility to assure the quality of his or her work while using this software. As per the provided license terms the package maintainer is not liable for any damage resulting from its usage. That being said, below we present some tips for generating reproducible results with this package.

How do I get started?

Installing this package is only the first step to get things started. You need to perform the following steps in order to use the package to its full capacity.

- First download a copy of the complete EPA database. This can be done by calling `download_ecotox_data`. This may not always work on all machines as R does not always accept the website SSL certificate from the EPA. In those cases the zipped archive with the database files can be downloaded manually with a different (more forgiving) browser. The files from the zip archive can be extracted to a location of choice.
- Next, an SQLite database needs to be build from the downloaded files. This will be done automatically when you used `download_ecotox_data` in the previous step. When you have manually downloaded the files you can call `build_ecotox_sqlite` to build the database locally.

- When the previous steps have been performed successfully, you can now search the database by calling `search_ecotox`. You can also use `dbConnectEcotox` to open a connection to the database. You can query the database using this connection and any of the methods provided from the DBI or RSQLite packages.

How do I obtain reproducible results?

Each individual user is responsible for evaluating the reproducibility of his or her work. Although this package offers instruments to achieve reproducibility, it is not guaranteed. In order to increase the chances of generating reproducible results, one should adhere at least to the following rules:

- Always use an official release from CRAN, and cite the version used in your analyses (`citation("ECOTOXr")`). Different versions, may produce different end results (although we will strive for backward compatibility).
- Make sure you are working with a clean (unaltered) version of the database. When in doubt, download and build a fresh copy of the database (`download_ecotox_data`). Also cite the (release) version of the downloaded database (`cite_ecotox`), and the system operating system in which the local database was build `get_ecotox_info`). Or, just make sure that you never modify the database (e.g., write data to it, delete data from it, etc.)
- In order to avoid platform dependencies it is advised to only include non-accented alphanumerical characters in search terms. See also search_ecotox and build_ecotox_sqlite.
- When trying to reproduce database extractions from earlier database releases, filter out additions after that specific release. This can be done by adding output fields ’tests.modified_date’, ’tests.created_date’ and ’tests.published_date’ to your search and compare those with the release date of the database you are trying to reproduce results from.

Why isn’t the database included in the package?

This package doesn’t come bundled with a copy of the database which needs to be downloaded the first time the package is used. Why is this? There are several reasons:

- The database is maintained and updated by the US EPA. This process is and should be outside the sphere of influence of the package maintainer.
- Packages on CRAN are not allowed to contain large amounts of data. Publication on CRAN is key to control the quality of this package and therefore outweighs the convenience of having the data bundled with the package.
- The user has full control over the release version of the database that is being used.

Why doesn’t this package search the online ECOTOX database?

Although this is possible, there are several reasons why we opted for creating a local copy:

- The user would be restricted to the search options provided on the website (ECOTOX).
- The online database doesn’t come with an API that would allow for convenient interface.
- The user is not limited by an internet connection and its bandwidth.
- Not all database fields can be retrieved from the online interface.

Author(s)

Pepijn de Vries

References

Official US EPA ECOTOX website: https://cfpub.epa.gov/ecotox/

---

get_ecotox_info *Get information on the local ECOTOX database when available*

Description

Get information on how and when the local ECOTOX database was build.

Usage

```
get_ecotox_info(path = get_ecotox_path(), version)
```

Arguments

path	A character string with the path to the location of the local database (default is get_ecotox_path()).
version	A character string referring to the release version of the database you wish to locate. It should have the same format as the date in the EPA download link, which is month, day, year, separated by underscores ("%m_%d_%Y"). When missing, the most recent available copy is selected automatically.

Details

Get information on how and when the local ECOTOX database was build. This information is retrieved from the log-file that is (optionally) stored with the local database when calling download_ecotox_data or build_ecotox_sqlite.

Value

Returns a vector of characters, containing a information on the selected local ECOTOX database.

Author(s)

Pepijn de Vries

Examples

```
## Not run:
## Show info on the current database (only works when one is downloaded and build):
get_ecotox_info()

## End(Not run)
```

get_ecotox_sqlite_file *The local path to the ECOTOX database (directory or sqlite file)*

Description

Obtain the local path to where the ECOTOX database is (or will be) placed.

Usage

```
get_ecotox_sqlite_file(path = get_ecotox_path(), version)

get_ecotox_path()
```

Arguments

path	When you have a copy of the database somewhere other than the default directory (`get_ecotox_path()`), you can provide the path here.
version	A `character` string referring to the release version of the database you wish to locate. It should have the same format as the date in the EPA download link, which is month, day, year, separated by underscores ("%m_%d_%Y"). When missing, the most recent available copy is selected automatically.

Details

It can be useful to know where the database is located on your disk. This function returns the location as provided by `app_dir`.

Value

Returns a `character` string of the path. `get_ecotox_path` will return the default directory of the database. `get_ecotox_sqlite_file` will return the path to the sqlite file when it exists.

Author(s)

Pepijn de Vries

Examples

```
get_ecotox_path()

## Not run:
## This will only work if a local database exists:
get_ecotox_sqlite_file()

## End(Not run)
```

list_ecotox_fields *List the field names that are available from the ECOTOX database*

Description

List the field names (table headers) that are available from the ECOTOX database

Usage

```
list_ecotox_fields(which = c("default", "full", "all"), include_table = TRUE)
```

Arguments

which	A character string that specifies which fields to return. Can be any of: 'default': returns default output field names; 'all': returns all fields; or 'full': returns all except fields from table 'dose_response_details'.
include_table	A logical value indicating whether the table name should be included as prefix. Default is TRUE.

Details

This can be useful when specifying a search_ecotox, to identify which fields are available from the database, for searching and output.

Value

Returns a vector of type character containing the field names from the ECOTOX database.

Author(s)

Pepijn de Vries

Examples

```
## Fields that are included in search results by default:
list_ecotox_fields("default")

## All fields that are available from the ECOTOX database:
list_ecotox_fields("all")

## All except fields from the table 'dose_response_details'
## that are available from the ECOTOX database:
list_ecotox_fields("all")
```

search_ecotox *Search and retrieve toxicity records from the database*

Description

Create (and execute) an SQL search query based on basic search terms and options. This allows you to search the database, without having to understand SQL.

Usage

```
search_ecotox(
  search,
  output_fields = list_ecotox_fields("default"),
  group_by_results = TRUE,
  ...
)

search_query_ecotox(
  search,
  output_fields = list_ecotox_fields("default"),
  group_by_results = TRUE
)
```

Arguments

`search` A named `list` containing the search terms. The names of the elements should refer to the field (i.e. table header) in which the terms are searched. Use `list_ecotox_fields()` to obtain a list of available field names.

Each element in that list should contain another list with at least one element named 'terms'. This should contain a `vector` of `character` strings with search terms. Optionally, a second element named 'method' can be provided which should be set to either '`contain`' (default, when missing) or '`exact`'. In the first case the query will match any record in the indicated field that contains the search term. In case of '`exact`' it will only return exact matches. Note that searches are not case sensitive, but are picky with special (accented) characters. While building the local database (see build_ecotox_sqlite) such special characters may be treated differently on different operating systems. For the sake of reproducibility, the user is advised to stick with non-accented alpha-numeric characters.

Search terms for a specific field (table header) will be combined with 'or'. Meaning that any record that matches any of the terms are returned. For instance when 'latin_name' 'Daphnia magna' and 'Skeletonema costatum' are searched, results for both species are returned. Search terms across fields (table headers) are combined with 'and', which will narrow the search. For instance if 'chemical_name' 'benzene' is searched in combination with 'latin_name' 'Daphnia magna', only tests where Daphnia magna are exposed to benzene are returned.

When this search behaviour described above is not desirable, the user can either adjust the query manually, or use this function to perform several separate searches and combine the results afterwards.

Beware that some field names are ambiguous and occur in multiple tables (like 'cas_number' and 'code'). When searching such fields, the search result may not be as expected.

`output_fields` A `vector` of `character` strings indicating which field names (table headers) should be included in the output. By default `list_ecotox_fields("default")` is used. Use `list_ecotox_fields("all")` to list all available fields.

`group_by_results`

Ecological test results are generally the most informative element in the ECOTOX database. Therefore, this search function returns a table with unique results in each row.

However, some tables in the database (such as 'chemical_carriers' and 'dose_responses') have a one to many relationship with test results. This means that multiple chemical carriers can be linked to a single test result, similarly, multiple doses can also be linked to a single test result.

By default the search results are grouped by test results. As a result not all doses or chemical carriers may be displayed in the output. Set the `group_by_results` parameter to `FALSE` in order to force SQLite to output all data (all carriers and doses). But beware that test results may be duplicated in those cases.

`...` Arguments passed to `dbConnectEcotox`. You can use this when the database is not located at the default path (`get_ecotox_path()`).

Details

The ECOTOX database is stored locally as an SQLite file, which can be queried with SQL. These functions allow you to automatically generate an SQL query and send it to the database, without having to understand SQL. The function `search_query_ecotox` generates and returns the SQL query (which can be edited by hand if desired). You can also directly call `search_ecotox`, this will first generate the query, send it to the database and retrieve the result.

Although the generated query is not optimized for speed, it should be able to process most common searches within an acceptable time. The time required for retrieving data from a search query depends on the complexity of the query, the size of the query and the speed of your machine. Most queries should be completed within seconds (or several minutes at most) on modern machines. If your search require optimisation for speed, you could try reordering the search fields. You can also edit the query generated with `search_query_ecotox` by hand and retrieve it with `dbGetQuery`.

Note that this package is actively maintained and this function may be revised in future versions. In order to create reproducible results the user must: always work with an official release from CRAN and document the package and database version that are used to generate specific results (see also `cite_ecotox()`).

Value

In case of `search_query_ecotox`, a `character` string containing an SQL query is returned. This query is built based on the provided search terms and options.

In case of `search_ecotox` a `data.frame` is returned based on the search query built with `search_query_ecotox`. The `data.frame` is unmodified as returned by SQLite, meaning that all fields are returned as `characters` (even where the field types are 'date' or 'numeric').

The results are tagged with: a time stamp; the package version used; and the file path of the SQLite database used in the search (when applicable). These tags are added as attributes to the output table or query.

Author(s)

Pepijn de Vries

Examples

```
## Not run:
## let's find the ids of all ecotox tests on species
## where latin names contain either of 2 specific genus names and
## where they were exposed to the chemical benzene
if (check_ecotox_availability()) {
  search <-
    list(
      latin_name    = list(
        terms         = c("Skeletonema", "Daphnia"),
        method        = "contains"
      ),
      chemical_name = list(
        terms         = "benzene",
        method        = "exact"
      )
    )
  ## numbers in result each represent a unique test id from the database
  result <- search_ecotox(search)
  query  <- search_query_ecotox(search)
  cat(query)
} else {
 print("Sorry, you need to use 'download_ecotox_data()' first in order for this to work.")
}

## End(Not run)
```

Index