

Package ‘ExperimentHubData’

January 19, 2026

Type Package

Title Add resources to ExperimentHub

Version 1.36.0

Maintainer Bioconductor Package Maintainer <maintainer@bioconductor.org>

Description Functions to add metadata to ExperimentHub db and resource files to AWS S3 buckets.

License Artistic-2.0

Depends utils, BiocGenerics (>= 0.15.10), S4Vectors, AnnotationHubData (>= 1.21.3)

Imports methods, ExperimentHub, BiocManager, DBI, httr, curl

Suggests GenomeInfoDb, RUnit, knitr, BiocStyle, rmarkdown, HubPub

biocViews Infrastructure

addResources

addResources

Description

Add resource metadata to a local ExperimentHub database

Usage

```
addResources(pathToPackage, fileName=character(), insert = FALSE, ...)
```

Arguments

pathToPackage	Full path to data package including package name.
fileName	Name of single metadata file located in "inst/extdata". If none is provided the function looks for a file named "metadata.csv".
insert	A logical to control if metadata are inserted in the ExperimentHub db. By default this option is FALSE which is a useful state in which to test a new recipe and confirm the metadata fields are correct. When insert = TRUE, the "EXPERIMENT_HUB_SERVER_POST_URL" global option must be set to the http location of the ExperimentHubServer in the global environment or .Rprofile. This option controls Additionally, AWS command line tools must be installed on the local

Examples

```
## Not run:
## Generate metadata for inspection
addResources("/home/vobencha/mypackage", insert=FALSE)
## Inset metadata into ExperimentHub database
addResources("/home/vobencha/mypackage", insert=TRUE)

## End(Not run)
```

ExperimentHubMetadata-class

Class [ExperimentHubMetadata](#) objects and methods

Description

The [ExperimentHubMetadata](#) object is used to represent records in the server data base.

Usage

```
ExperimentHubMetadata(ExperimentHubRoot=NA_character_,
  BiocVersion=BiocManager::version(),
  SourceUrl=NA_character_,
  SourceType=NA_character_,
  SourceVersion=NA_character_,
  SourceLastModifiedDate=as.POSIXct(NA_character_),
  SourceMd5=NA_character_,
  SourceSize=NA_real_,
  DataProvider=NA_character_,
  Title=NA_character_,
  Description=NA_character_,
  Maintainer=NA_character_,
  Species=NA
```

SourceType	character() Form of original data, e.g., BED, FASTA, etc. getValidSourceTypes() list currently acceptable values. If nothing seems appropriate for your data reach out to maintainer@bioconductor.org.
SourceVersion	character(1) Version of original file.
SourceLastModifiedDate	POSIXct() Date when resource was last modified.
SourceMd5	character(1) md5 hash of original file.
SourceSize	numeric(1) Number of bytes in original file.
DataProvider	character(1) Provider of original data, e.g., NCBI, UniProt etc.
Title	character(1) Title for the resource with version or genome build as appropriate. Titles must be unique and not match any existing title in ExperimentHub.
Description	character(1) Description of the resource. May include details such as data type, format, study origin, sequencing technology, treated vs control, number of samples etc.
Species	character(1) Species name. For help on valid species see getSpeciesList, validSpecies, or suggestSpecies.
TaxonomyId	character(1) NCBI code. Taxonomy ID. There are checks for valid taxonomyId given the Species which produce warnings. See GenomeInfoDb::loadTaxonomyDb() for full validation table.
Genome	character(1) Name of genome build.
Tags	'Tags' are search terms used to define a subset of resources in a Hub object, e.g., in a call to query. For ExperimentHub resources, 'Tags' are automatically generated from the 'biocViews' in the DESCRIPTION file of the accompanying software

DispatchClass	character(1). Determines how data are loaded into R. The value for this field should be 'Rda' if the data were serialized with save() and 'Rds' if serialized with saveRDS. The filename should have the appropriate 'rda' or 'rds' extension. A number of dispatch classes are pre-defined in AnnotationHub/R/AnnotationHubResource-class.R with the suffix 'Resource'. For example, if you have sqlite files, the AnnotationHubResource-class.R defines SQLiteFileResource so the Dispatch-Class would be SQLiteFile. Contact maintainer@bioconductor.org if you are not sure which class to use. The function AnnotationHub::DispatchClassList() will output a matrix of currently implemented DispatchClass and brief description of utility. If a predefine class does not seem appropriate contact maintainer@bioconductor.org.
Location_Prefix	character(1) URL location of AWS S3 bucket or web site where resource is located.
Notes	character() Notes about the resource.
PreparerClass	character(1) Used internally.

Details

In practice, instances of this class are generated by a call to addResources or makeExperimentHubMetadata instead of a direct call to the constructor.

addResources is a function used by the Bioconductor Core team when adding new metadata records to the production database. makeExperimentHubMetadata and the low-level helper

Arguments

pathToPackage	Full path to data package including the package name; no trailing slash
fileName	Name of single metadata file located in "inst/extdata". If none is provided the function looks for a file named "metadata.csv".

Details

- `makeExperimentHubMetadata`: Reads the resource metadata in the `metadata.csv` file into a `ExperimentHubMetadata` object. The `ExperimentHubMetadata` is inserted in the ExperimentHub database. Intended for internal use or package authors checking the validity of package metadata.

- Formatting metadata files:

`makeExperimentHubMetadata` reads `.csv` files of metadata located in "inst/extdata". Internal functions perform checks for required columns and data types and can be used by package authors to validate their metadata before submitting the package for review.

A number of dispatch classes are pre-defined in AnnotationHub/R/AnnotationHubResource-class.R with the suffix 'Resource'. For example, if you have sqlite files, the AnnotationHubResource-class.R defines SQLiteFileResource so the DispatchClass would be SQLiteFile. Contact maintainer@bioconductor.org if you are not sure which class to use. The function AnnotationHub::DispatchClassList() will output a matrix of currently implemented DispatchClass and brief description of utility. If a predefined class does not seem appropriate contact maintainer@bioconductor.org. An all purpose DispatchClass is FilePath that instead of trying to load the file into R, will only return the path to the locally downloaded file.

- Location_Prefix: character(1). Do not include this field if data are stored in the Bioconductor AWS S3; it will be generated automatically.
If data will be accessed from a location other than AWS S3 this field should be the base url.
- RDataPath: character(). This field should be the remainder of the path to the resource. The Location_Prefix will be prepended to RDataPath for the full path to the resource. If the resource is stored in Bioconductor's AWS S3 buckets, it should start with the name of the package associated with the metadata and should not start with a leading slash. It should include the resource file name. For strongly associated files, like a bam file and its index file, the two files should be separated with a colon :. This will link a single hub id with the multiple files.
- Tags: character() vector. 'Tags' are search terms used to define a subset of resources in

Examples

```
## makeExperimentHubMetadata() reads data from inst/scripts/<files>.csv
## into ExperimentHubMetadata objects. These objects are used to insert
## metadata into the production database. This function is used internally
## by addResources() and is not intended to be called directly.

## For an example of how this works we can use the GSE62944 ExperimentHub
## package. Download the source tarball from:

# http://www.bioconductor.org/packages/devel/data/experiment/html/GSE62944.html

## and unpack it. Set 'pathToPackage' to point to the downloaded source.
## Then call the function:
## Not run:
makeExperimentHubMetadata("path/to/mypackage")

## End(Not run)
```

Index

* **classes**

ExperimentHubMetadata-class, [3](#)

* **methods**

addResources, [2](#)

makeExperimentHubMetadata, [5](#)

addResources, [2](#), [5](#), [7](#)

AnnotationHubMetadata, [2](#)

class:ExperimentHubMetadata
(ExperimentHubMetadata-class),
[3](#)

ExperimentHubMetadata, [2](#), [3](#), [5–7](#)

ExperimentHubMetadata
(ExperimentHubMetadata-class),
[3](#)

ExperimentHubMetadata-class, [3](#)