

Package ‘AnnotationHub’

October 11, 2016

Type Package

Title Client to access AnnotationHub resources

Version 2.4.2

biocViews Infrastructure, DataImport, GUI, ThirdPartyClient

Maintainer Bioconductor Package Maintainer <maintainer@bioconductor.org>

Description This package provides a client for the Bioconductor AnnotationHub web resource. The AnnotationHub web resource provides a central location where genomic files (e.g., VCF, bed, wig) and other resources from standard locations (e.g., UCSC, Ensembl) can be discovered. The resource includes metadata about each resource, e.g., a textual description, tags, and date of modification. The client creates and manages a local cache of files retrieved by the user, helping with quick and reproducible access.

License Artistic-2.0

Depends BiocGenerics (>= 0.15.10)

Imports utils, methods, grDevices, RSQLite, BiocInstaller,
AnnotationDbi (>= 1.31.19), S4Vectors, interactiveDisplayBase,
httr

Suggests IRanges, GenomicRanges, GenomeInfoDb, VariantAnnotation,
Rsamtools, rtracklayer, BiocStyle, knitr, AnnotationForge,
rBiopaxParser, RUnit, GenomicFeatures, MSnbase, mzR,
Biostrings, SummarizedExperiment

Enhances AnnotationHubData

Collate AnnotationHubOption.R AllGenerics.R Hub-class.R db-utils.R
AnnotationHub-class.R AnnotationHubResource-class.R
AnnotationHubBEDResource-class.R
AnnotationHubProteomicsResource-class.R
AnnotationHubEpigenomeResource-class.R utilities.R sql-utils.R
hub-utils.R cache-utils.R zzz.R

VignetteBuilder knitr

NeedsCompilation yes

Author Martin Morgan [cre],
 Marc Carlson [ctb],
 Dan Tenenbaum [ctb],
 Sonali Arora [ctb]

R topics documented:

AnnotationHub-package	2
AnnotationHub-objects	2
AnnotationHubResource-objects	6
getAnnotationHubOption	7

Index	9
--------------	----------

AnnotationHub-package *Light-weight AnnotationHub 3.0 Client*

Description

Client to Bioconductor AnnotationHub 3.0 for discovery and retrieval of annotation resources.

Author(s)

Martin Morgan mtmorgan@fhcrc.org

Examples

```
packageDescription("AnnotationHub")
```

AnnotationHub-objects *AnnotationHub objects and their related methods and functions*

Description

Use AnnotationHub to interact with Bioconductor's AnnotationHub service. Query the instance to discover and use resources that are of interest, and then easily download and import the resource into R for immediate use.

Use AnnotationHub() to retrieve information about all records in the hub.

Discover records in a hub using mcols(), query(), subset(), [, and display().

Retrieve individual records using [[]]. On first use of a resource, the corresponding files or other hub resources are downloaded from the internet to a local cache. On this and all subsequent uses the files are quickly input from the cache into the R session.

AnnotationHub records can be added (and sometimes removed) at any time. snapshotDate() restricts hub records to those available at the time of the snapshot; use possibleDates() to see possible snapshot dates.

The location of the local cache can be found (and updated) with getAnnotationHubCache and setAnnotationHubCache; removeCache removes all cache resources.

Constructors

```
AnnotationHub(..., hub=getAnnotationHubOption("URL"), cache=getAnnotationHubOption("CACHE"),
  Create an AnnotationHub instance, possibly updating the current database of records.
```

Accessors

In the code snippets below, `x` and `object` are `AnnotationHub` objects.

`hubCache(x)`: Gets the file system location of the local `AnnotationHub` cache.

`hubUrl(x)`: Gets the URL for the online hub.

`length(x)`: Get the number of hub records.

`names(x)`: Get the names (`AnnotationHub` unique identifiers, of the form AH12345) of the hub records.

`fileName(x)`: Get the file path of the hub records as stored in the local cache (`AnnotationHub` files are stored as unique numbers, of the form 12345). NA is returned for those records which have not been cached.

`mcols(x)`: Get the metadata columns describing each record. Columns include:

title Record title, frequently the file name of the object.

dataprovider Original provider of the resource, e.g., Ensembl, UCSC.

species The species for which the record is most relevant, e.g., 'Homo sapiens'.

taxonomyid NCBI taxonomy identifier of the species.

genome Genome build relevant to the record, e.g., hg19.

description Textual description of the resource, frequently automatically generated from file path and other information available when the record was created.

tags Single words added to the record to facilitate identification, e.g., TCGA, Roadmap.

rdataclass The class of the R object used to represent the object when imported into R, e.g., `GRanges`, `VCFFile`.

sourceurl Original URL of the resource.

sourcefile Format of the original resource, e.g., BED file.

`dbconn(x)`: Return an open connection to the underlying SQLite database.

`dbfile(x)`: Return the full path the underlying SQLite database.

`.db_close(conn)`: Close the SQLite connection `conn` returned by `dbconn(x)`.

Subsetting and related operations

In the code snippets below, `x` is an `AnnotationHub` object.

`x$name`: Convenient reference to individual metadata columns, e.g., `x$species`.

`x[i]`: Numerical, logical, or character vector (of `AnnotationHub` names) to subset the hub, e.g., `x[x$species == "Homo sapiens"]`.

`x[[i]]`: Numerical or character scalar to retrieve (if necessary) and import the resource into R.

`query(x, pattern, ignore.case=TRUE, pattern.op= '&')`: Return an `AnnotationHub` subset containing only those elements whose metadata matches `pattern`. Matching uses `pattern` as in `grep` to search the `as.character` representation of each column, performing a logical '&' across columns. e.g., `query(x, c("Homo sapiens", "hg19", "GTF"))`.

`pattern` A character vector of patterns to search (via `grep1`) for in any of the `mcols()` columns.

`ignore.case` A logical(1) vector indicating whether the search should ignore case (TRUE) or not (FALSE).

`pattern.op` Any function of two arguments, describing how matches across pattern elements are to be combined. The default ``&`` requires that only records with *all* elements of `pattern` in their metadata columns are returned.

`subset(x, subset)`: Return the subset of records containing only those elements whose metadata satisfies the *expression* in `subset`. The expression can reference columns of `mcols(x)`, and should return a logical vector of length `length(x)`. e.g., `subset(x, species == "Homo sapiens" & genome="GRCh38")`.

`display(object)`: Open a web browser allowing for easy selection of hub records via interactive tabular display. Return value is the subset of hub records identified while navigating the display.

`recordStatus(hub, record)`: Returns a data.frame of the record id and status. `hub` must be a Hub object and `record` must be a character(1). Can be used to discover why a resource was removed from the hub.

Cache and hub management

In the code snippets below, `x` is an AnnotationHub object.

`snapshotDate(x)` and `snapshotDate(x) <- value`: Gets or sets the date for the snapshot in use. `value` should be one of `possibleDates()`.

`possibleDates(x)`: Lists dates for snapshots that the hub could potentially use.

`cache(x)` and `cache(x) <- NULL`: Adds (downloads) all resources in `x`, or removes all local resources corresponding to the records in `x` from the cache. In this case, `x` would typically be a small subset of AnnotationHub resources.

`hubUrl(x)`: Gets the URL for the online AnnotationHub.

`hubCache(x)`: Gets the file system location of the local AnnotationHub cache.

`removeCache(x)`: Removes local AnnotationHub database and all related resources. After calling this function, the user will have to download any AnnotationHub resources again.

`getAnnotationHubOption()`: TODO: Get cache options "CACHE", "URL", "MAXDOWNLOADS" ...

`setAnnotationHubOption()`: TODO: Set cache options "CACHE", "URL", "MAXDOWNLOADS" ...

Coercion

In the code snippets below, `x` is an AnnotationHub object.

`as.list(x)`: Coerce `x` to a list of hub instances, one entry per element. Primarily for internal use.

`c(x, ...)`: Concatenate one or more sub-hub. Sub-hubs must reference the same AnnotationHub instance. Duplicate entries are removed.

Author(s)

Martin Morgan, Marc Carlson, Sonali Arora, and Dan Tenenbaum

Examples

```
## create an AnnotationHub object
library(AnnotationHub)
ah = AnnotationHub()

## Summary of available records
ah

## Detail for a single record
ah[1]

## and what is the date we are using?
snapshotDate(ah)

## how many resources?
length(ah)

## from which resources, is data available?
head(sort(table(ah$dataprovider), decreasing=TRUE))

## from which species, is data available ?
head(sort(table(ah$species),decreasing=TRUE))

## what web service and local cache does this AnnotationHub point to?
hubUrl(ah)
hubCache(ah)

### Examples ###

## One can search the hub for multiple strings
ahs2 <- query(ah, c("GTF", "77", "Ensembl", "Homo sapiens"))

## information about the file can be retrieved using
ahs2[1]

## one can further extract information from this show method
## like the sourceurl using:
ahs2$sourceurl
ahs2$description
ahs2$title

## We can download a file by name like this (using a list semantic):
gr <- ahs2[[1]]
## And we can also extract it by the names like this:
res <- ah[["AH28812"]]

## the gtf file is returned as a GenomicRanges object and contains
## data about which organism it belongs to, its seqlevels and seqlengths
```

```
seqinfo(gr)

## each GenomicRanges contains a metadata slot which can be used to get
## the name of the hub object and other associated metadata.
metadata(gr)
ah[metadata(gr)$AnnotationHubName]

## And we can also use "[" to restrict the things that are in the
## AnnotationHub object (by position, character, or logical vector).
## Here is a demo of position:
subHub <- ah[1:3]

if(interactive()) {
  ## Display method involves user interaction through web interface
  ah2 <- display(ah)
}

## recordStatus
recordStatus(ah, "TEST")
recordStatus(ah, "AH7220")
```

AnnotationHubResource-objects

AnnotationHubResource objects and their related methods and functions

Description

TODO

Accessors

In the code snippets below, `x` and `object` are `AnnotationHubResource` objects.

`getHub(x)`: Gets the `AnnotationHub` from the hub slot.

`getCache(x)`: Gets the location of the `AnnotationHub` cache from the `AnnotationHub` object in the hub slot.

`getUrl(x)`: Gets the location of the `AnnotationHub` production server from the `AnnotationHub` object in the hub slot.

Author(s)

Bioconductor Core Team

Examples

```
## TODO
```

`getAnnotationHubOption`*Get and set options for default AnnotationHub behavior.*

Description

These functions get or set options for creation of new ‘AnnotationHub’ instances.

Usage

```
getAnnotationHubOption(arg)
setAnnotationHubOption(arg, value)
```

Arguments

<code>arg</code>	The character(1) hub options to set. see ‘Details’ for current options.
<code>value</code>	The value to be assigned to the hub option.

Details

Supported options include:

“**URL**”: character(1). The base URL of the annotation hub. Default: <https://annotationhub.bioconductor.org>

“**CACHE**”: character(1). The location of the hub cache. Default: “.AnnotationHub” in the user home directory.

“**MAX_DOWNLOADS**”: numeric(1). The integer number of downloads allowed before triggering an error. This is to help avoid accidental download of a large number of AnnotationHub members.

“**PROXY**”: request object returned by `httr::use_proxy()`. The request object describes a proxy connection allowing Internet access, usually through a restrictive firewall. Setting this option sends all AnnotationHub requests through the proxy. Default: NULL.

In `setHubOption("PROXY", value)`, `value` can be one of NULL, a request object returned by `httr::use_proxy()`, or a well-formed URL as character(1). The URL can be completely specified by `http://username:password@proxy.dom.com:8080`; `username:password` and port (e.g. `:8080`) are optional.

Default values may also be determined by system and global R environment variables visible *before* the package is loaded. Use options or variables preceeded by “ANNOTATION_HUB_”, e.g., `options(ANNOTATION_HUB_MAX_DOWNLOADS=10)` prior to package load sets the default number of downloads to 10.

Value

The requested or successfully set option.

Author(s)

Martin Morgan mtmorgan@fhcrc.org

Examples

```
getAnnotationHubOption("URL")  
## Not run:  
setAnnotationHubOption("CACHE", "~/myHub")  
  
## End(Not run)
```


Index

- *Topic **classes**
 - AnnotationHub-objects, 2
 - AnnotationHubResource-objects, 6
- *Topic **manip**
 - getAnnotationHubOption, 7
- *Topic **methods**
 - AnnotationHub-objects, 2
 - AnnotationHubResource-objects, 6
- *Topic **package**
 - AnnotationHub-package, 2
 - .Hub (AnnotationHub-objects), 2
 - .db_close (AnnotationHub-objects), 2
 - [, Hub, character, missing-method (AnnotationHub-objects), 2
 - [, Hub, logical, missing-method (AnnotationHub-objects), 2
 - [, Hub, numeric, missing-method (AnnotationHub-objects), 2
 - [<-, Hub, character, missing, Hub-method (AnnotationHub-objects), 2
 - [<-, Hub, logical, missing, Hub-method (AnnotationHub-objects), 2
 - [<-, Hub, numeric, missing, Hub-method (AnnotationHub-objects), 2
 - [[, Hub, character, missing-method (AnnotationHub-objects), 2
 - [[, Hub, numeric, missing-method (AnnotationHub-objects), 2
 - \$, Hub-method (AnnotationHub-objects), 2
- AnnotationHub (AnnotationHub-objects), 2
- AnnotationHub-class (AnnotationHub-objects), 2
- AnnotationHub-objects, 2
- AnnotationHub-package, 2
- AnnotationHubResource-class (AnnotationHubResource-objects), 6
- AnnotationHubResource-objects, 6
- as.list, Hub-method (AnnotationHub-objects), 2
- as.list.Hub (AnnotationHub-objects), 2
- c, Hub-method (AnnotationHub-objects), 2
- cache (AnnotationHub-objects), 2
- cache, AnnotationHub-method (AnnotationHub-objects), 2
- cache, Hub-method (AnnotationHub-objects), 2
- cache<- (AnnotationHub-objects), 2
- cache<-, Hub-method (AnnotationHub-objects), 2
- class:AnnotationHub (AnnotationHub-objects), 2
- class:AnnotationHubResource (AnnotationHubResource-objects), 6
- class:Hub (AnnotationHub-objects), 2
- dbconn, Hub-method (AnnotationHub-objects), 2
- dbfile, Hub-method (AnnotationHub-objects), 2
- display (AnnotationHub-objects), 2
- display, Hub-method (AnnotationHub-objects), 2
- fileName, Hub-method (AnnotationHub-objects), 2
- getAnnotationHubOption, 7
- getHub (AnnotationHubResource-objects), 6
- getHub, AnnotationHubResource-method (AnnotationHubResource-objects), 6
- grepl, 3
- Hub-class (AnnotationHub-objects), 2
- hubCache (AnnotationHub-objects), 2

- hubCache, AnnotationHubResource-method
(AnnotationHubResource-objects),
6
- hubCache, Hub-method
(AnnotationHub-objects), 2
- hubDate (AnnotationHub-objects), 2
- hubDate, Hub-method
(AnnotationHub-objects), 2
- hubUrl (AnnotationHub-objects), 2
- hubUrl, AnnotationHubResource-method
(AnnotationHubResource-objects),
6
- hubUrl, Hub-method
(AnnotationHub-objects), 2

- length, Hub-method
(AnnotationHub-objects), 2

- mcols, Hub-method
(AnnotationHub-objects), 2

- names, Hub-method
(AnnotationHub-objects), 2

- package (AnnotationHub-objects), 2
- package, Hub-method
(AnnotationHub-objects), 2
- possibleDates (AnnotationHub-objects), 2

- query (AnnotationHub-objects), 2
- query, Hub-method
(AnnotationHub-objects), 2

- recordStatus (AnnotationHub-objects), 2
- recordStatus, Hub-method
(AnnotationHub-objects), 2
- removeCache (AnnotationHub-objects), 2

- setAnnotationHubOption
(getAnnotationHubOption), 7
- show, AnnotationHub-method
(AnnotationHub-objects), 2
- show, AnnotationHubResource-method
(AnnotationHub-objects), 2
- show, Hub-method
(AnnotationHub-objects), 2
- snapshotDate (AnnotationHub-objects), 2
- snapshotDate, Hub-method
(AnnotationHub-objects), 2

- snapshotDate<- (AnnotationHub-objects),
2
- snapshotDate<-, Hub-method
(AnnotationHub-objects), 2
- subset, Hub-method
(AnnotationHub-objects), 2