

Package ‘AnnotationHub’

October 8, 2015

Type Package

Title Client to access AnnotationHub resources

Version 2.0.4

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

Imports utils, methods, RSQLite, BiocInstaller, BiocGenerics, S4Vectors, interactiveDisplayBase, httr, AnnotationDbi

Suggests GenomicRanges, VariantAnnotation, Rsamtools, rtracklayer, GenomeInfoDb, BiocStyle, knitr, AnnotationForge, rBiopaxParser, RUnit

Collate AnnotationHub-class.R AnnotationHubResource-class.R AnnotationHubBEDResource-class.R utilities.R option.R sql.R hub.R cache.R display.R zzz.R

VignetteBuilder knitr

NeedsCompilation no

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

R topics documented:

AnnotationHub-package 2

AnnotationHub-objects	2
hubOption	5
Index	7

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

AnnotationHub is the base class for interacting with the AnnotationHub services using Bioconductor. When using the AnnotationHub package, users will create an instance of this class and then subset it down to those resources that are of interest.

Once an AnnotationHub is created users can extract data by using the subset operators (along with an assignment operator to put the contents of one of these resources into a local variable). Basically you just need to look at the contents of the AnnotationHub object using `grep` or a display method until you have decided which records you want to retrieve and then you can use the `[]` operator to retrieve the relevant records. Using the `[]` operator will just subset the AnnotationHub object so that it refers to fewer records.

All AnnotationHub have a snapshot date that by default is set to the most recent one. The `snapshotDate` method indicates which one is in use, but this value can also be set to previous dates. Other dates can be accessed with the `possibleDates` function.

Whenever you download a file from AnnotationHub it will automatically put the data in a local cache for future reference. The location of this cache can be found and even changed with the `hubCache` setters and getters. This cache provides a performance boost for users but it does not mean that you can use this on a plane without needing wireless access.

Usage

```

AnnotationHub(..., hub=hubOption("URL"), cache=hubOption("CACHE"),
  max.downloads=hubOption("MAX_DOWNLOADS"))
cache(x, ..., max.downloads)
cache(x, ...) <- value
hubUrl(x)
hubCache(x)
possibleDates(x)
## S4 method for signature 'AnnotationHub'
mcols(x)
## S4 method for signature 'AnnotationHub'
snapshotDate(x)
## S4 replacement method for signature 'AnnotationHub'
snapshotDate(x) <- value
## S4 method for signature 'AnnotationHub'
query(x, pattern, ignore.case, pattern.op )
## S4 method for signature 'AnnotationHub'
subset(x, subset)
## S4 method for signature 'AnnotationHub'
display(object)
## S4 method for signature 'AnnotationHub'
dbfile(x)
## S4 method for signature 'AnnotationHub'
dbconn(x)

```

Arguments

<code>x, object</code>	The AnnotationHub object.
<code>value</code>	The value to be assigned.
<code>pattern</code>	A character regular expression to query metadata of <code>x</code> .
<code>subset</code>	An expression referring to columns of metadata(<code>x</code>). See Methods for additional detail.
<code>ignore.case</code>	if FALSE, the pattern matching is case sensitive and if TRUE, case is ignored during matching.
<code>hub</code>	The URI where the online hub metadata is stored.
<code>cache</code>	the local cache where the data is stored after being downloaded or updated.
<code>max.downloads</code>	The maximum number of items to try and download without prompting. Used as a safety net to prevent users from trying to cache the entire hub.
<code>pattern.op</code>	if <code>&</code> then records containing all the patterns will be returned and if <code> </code> then records containing any one of or both patterns will be returned
<code>...</code>	Additional arguments.

Methods

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

AnnotationHub(): Constructs an AnnotationHub object.

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

hubUrl(x): Gets the path to the local hub cache.

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

snapshotDate(x) and snapshotDate(x, ...) <- value: Gets or sets the date for the snapshot in use.

cache(x, ...) and cache(x, ...) <- value: Adds (downloads) all resources in x, or (caution!) removes all local resources from the cache.

query(x, pattern, ignore.case, pattern.op): Return an AnnotationHub subset containing only those elements whose metadata matches pattern. Matching uses pattern as in `grep1` to search the `as.character` representation of each column, performing a logical OR across columns. Arguments ... are passed to `grep1` for within-column matching.

subset(x, subset): Return an AnnotationHub subset containing only those elements whose metadata satisfies the *expression* in subset. The expression can reference columns of `metadata(x)`, and should return a logical vector of length `length(x)`.

display(object, ...): Implements the `display` generic from `interactiveDisplay` package. Users can select metadata rows from a web interface, the values in the rows selected by the user become filters applied to the AnnotationHub object.

mcols(x): returns a `DataFrame` with metadata for the AnnotationHub object.

dbfile(x): returns the file path to the AnnotationHub objects metadata database cache.

dbconn(x): returns a database connection for the AnnotationHub objects metadata cache.

Author(s)

Martin Morgan, Marc Carlson and Dan Tenenbaum

Examples

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

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

## how many resources?
length(ah)

## from which resources, is data available?
unique(ah$dataproducer)

## from which species, is data available ?
head(unique(ah$species))

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

```

### Examples ###
### HOW TO : get Inparanoid data wrt ailMe11
## query and subset
ahs <- query(ah, 'inparanoid8')
ahs <- subset(ah, ah$genome=='ailMe11')

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

## We can get the metadata for the file
ahs[1]

## We can extract a result by name like this (using a list semantic):
res <- ahs[[1]]
## And we can also extract it by the names like this:
res <- ahs[["AH6298"]]

## 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)
}

## get the file to the metadata database cache or a DBI db connection.
dbfile(ah)
dbconn(ah)

```

hubOption

Set or get options for default hub behavior.

Description

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

Usage

```

hubOption(arg)
setHubOption(arg, value)

```

Arguments

arg	The character(1) hub options to set. see 'Details' for current options.
value	The (scalar) 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.

Default values may also be determined by system and global R environment variables visible *before* the package is loaded. Use options or variables preceded 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

```
hubOption("URL")
## Not run:
setHubOption("CACHE", "~/myHub")

## End(Not run)
```

Index

- *Topic **classes**
 - AnnotationHub-objects, 2
- *Topic **manip**
 - hubOption, 5
- *Topic **methods**
 - AnnotationHub-objects, 2
- *Topic **package**
 - AnnotationHub-package, 2
- [, AnnotationHub, character, missing-method
(AnnotationHub-objects), 2
- [, AnnotationHub, logical, missing-method
(AnnotationHub-objects), 2
- [, AnnotationHub, numeric, missing-method
(AnnotationHub-objects), 2
- [<-, AnnotationHub, character, missing, AnnotationHub-method
(AnnotationHub-objects), 2
- [<-, AnnotationHub, logical, missing, AnnotationHub-method
(AnnotationHub-objects), 2
- [<-, AnnotationHub, numeric, missing, AnnotationHub-method
(AnnotationHub-objects), 2
- [[, AnnotationHub, character, missing-method
(AnnotationHub-objects), 2
- [[, AnnotationHub, numeric, missing-method
(AnnotationHub-objects), 2
- \$, AnnotationHub-method
(AnnotationHub-objects), 2

- AnnotationHub (AnnotationHub-objects), 2
- AnnotationHub-class
(AnnotationHub-objects), 2
- AnnotationHub-objects, 2
- AnnotationHub-package, 2

- cache (AnnotationHub-objects), 2
- cache<- (AnnotationHub-objects), 2
- class:AnnotationHub
(AnnotationHub-objects), 2

- dbconn, AnnotationHub-method
(AnnotationHub-objects), 2

- dbfile, AnnotationHub-method
(AnnotationHub-objects), 2
- display (AnnotationHub-objects), 2
- display, AnnotationHub-method
(AnnotationHub-objects), 2

- grep1, 4

- hubCache (AnnotationHub-objects), 2
- hubOption, 5
- hubUrl (AnnotationHub-objects), 2

- length, AnnotationHub-method
(AnnotationHub-objects), 2
- mcOis, AnnotationHub-method
(AnnotationHub-objects), 2
- names, AnnotationHub-method
(AnnotationHub-objects), 2
- possibleDates (AnnotationHub-objects), 2
- query (AnnotationHub-objects), 2
- query, AnnotationHub-method
(AnnotationHub-objects), 2

- setHubOption (hubOption), 5
- show, AnnotationHub-method
(AnnotationHub-objects), 2
- show, AnnotationHubResource-method
(AnnotationHub-objects), 2
- snapshotDate (AnnotationHub-objects), 2
- snapshotDate, AnnotationHub-method
(AnnotationHub-objects), 2
- snapshotDate<- (AnnotationHub-objects),
2
- snapshotDate<- , AnnotationHub-method
(AnnotationHub-objects), 2
- subset, AnnotationHub-method
(AnnotationHub-objects), 2