

Package ‘AnnotationHubData’

October 11, 2016

Type Package

Title Transform public data resources into Bioconductor Data Structures

Version 1.2.2

Depends R (>= 3.2.2), methods, S4Vectors (>= 0.7.21), IRanges (>= 2.3.23), GenomicRanges, AnnotationHub

Suggests RUnit, knitr, RMySQL, BiocStyle, grasp2db

Imports GenomicFeatures, Rsamtools, rtracklayer, BiocGenerics, jsonlite, BiocInstaller, httr, AnnotationDbi, Biobase, Biostrings, DBI, GEOquery, GenomeInfoDb, OrganismDbi, RSQLite, rBiopaxParser, AnnotationForge, futile.logger (>= 1.3.0), XML, xml2, curl

Description These recipes convert a wide variety and a growing number of public bioinformatic data sets into easily-used standard Bioconductor data structures.

License Artistic-2.0

LazyLoad yes

biocViews DataImport

VignetteBuilder knitr

Collate Message-class.R ImportPreparer-class.R
makeAnnotationHubResource.R HubMetadata-class.R
AnnotationHubMetadata-class.R utils.R updateResources.R
ahmToJson.R webAccessFunctions.R makeBioPaxImporter.R
makeChEA.R madebSNPVCF.R makeEncodeDCC.R
makeEnsemblGtfToGRanges.R makeEnsemblFasta.R
makeEpigenomeRoadmap.R makeGencodeFasta.R makeGenecodeGFF.R
makeGrasp2Db.R makeGSE62944ToExpressionSet.R makeHaemCode.R
makeInparanoid8ToDbs.R makeNCBIToOrgDbs.R
makeRecentOrgPkgsToDbs.R makePazar.R makeRefNet.R
makeUCSCChain.R makeUCSC2Bit.R makeUCSCTracks.R
trackWithAuxiliaryTableToGRangesRecipe.R
UCSCTrackUpdateChecker.R makeEnsemblTwoBit.R

NeedsCompilation no

Author Martin Morgan [ctb],
 Marc Carlson [ctb],
 Dan Tenenbaum [ctb],
 Sonali Arora [ctb],
 Paul Shannon [ctb],
 Bioconductor Package Maintainer [cre]

Maintainer Bioconductor Package Maintainer <maintainer@bioconductor.org>

R topics documented:

AnnotationHubRecipes-package	2
AnnotationHubMetadata-class	3
flog	6
ImportPreparer-class	6
makeEnsemblFasta	7
makeGencodeFasta	8
updateResources	10
upload_to_S3	12

Index **14**

AnnotationHubRecipes-package

Transform public data resources into Bioconductor Data Structures
 ~~ package title ~~

Description

These recipes convert a wide variety and a growing number of public bioinformatic data sets into easily-used standard Bioconductor data structures.

Details

Package: AnnotationHubRecipes
 Type: Package
 Version: 1.0
 Date: 2012-11-30
 License: Artistic-2.0

This package provides a set of methods which convert bioinformatic data resources into standard Bioconductor data types. For example, a UCSC genome browser track, expressed as a BED file, is converted into a GRanges object. Not every valuable data resource can be transformed quite so easily; some require more elaborate transformation, and hence a more specialized recipe. Every effort is made to limit the number of recipes required. One strategy that helps with the principle of "zero curation": unless absolutely required, the "cooked" version of the data resource produced by

a recipe is a simple and unembellished reflection of the original data in its downloaded form.

Author(s)

Dan Tenenbaum, Paul Shannon

AnnotationHubMetadata-class
Class "AnnotationHubMetadata" and methods

Description

AnnotationHubMetadata is used to represent record(s) in the server data base.

Usage

```
AnnotationHubMetadata(AnnotationHubRoot, SourceUrl, SourceType,
                      SourceVersion, SourceLastModifiedDate, SourceMd5 =
                      NA_character_, SourceSize, DataProvider, Title,
                      Description, Species, TaxonomyId, Genome, Tags,
                      Recipe, RDataClass, RDataDateAdded, RDataPath,
                      Maintainer, ..., BiocVersion = biocVersion(),
                      Coordinate_1_based = TRUE, Notes = NA_character_,
                      DispatchClass, Location_Prefix =
                      "http://s3.amazonaws.com/annotationhub/")

jsonPath(x)
toJson(x)
constructAnnotationHubMetadataFromSourceFilePath(ahroot, originalFile)
constructMetadataFromJsonPath(ahroot, jsonpath)
constructSeqInfo(species, genome)

metadata(x, ...)
hubError(x)
inputFiles(object, ...)
outputFile(object)
ahmToJson(ahm)
deleteResources(id)
getImportPreparerClasses()
makeAnnotationHubResource(objName, makeAnnotationHubMetadataFunction,
                          ..., where)
```

Arguments

AnnotationHubRoot
 character(1) Absolute path to directory structure containing resources to be added to AnnotationHub

SourceUrl	character()	URL where resource(s) can be found
SourceType	character()	which indicates what kind of resource was initially processed. The preference is to name the type of resource if it's a single file type and to name where the resources came from if it is a compound resource. So Typical answers would be like: 'BED', 'FASTA' or 'Inparanoid' etc.
SourceVersion	character(1)	Version of original file
SourceLastModifiedDate	POSIXct()	The date when the source was last modified. Leaving this blank should allow the values to be retrieved for you (if your sourceURL is valid).
SourceMd5	character()	md5 hash of original file
SourceSize	numeric(1)	Number of bytes in original file
DataProvider	character(1)	Where did this resource come from?
Title	character(1)	Title for this resource
Description	character(1)	Description of the resource
Species	character(1)	Species name
TaxonomyId	character(1)	NCBI code
Genome	character(1)	Name of genome build
Tags	character()	Free-form tags
Recipe	character(1)	Name of recipe function
RDataClass	character(1)	Class of derived object (e.g. 'GRanges')
RDataDateAdded	POSIXct()	Date added to AnnotationHub. Used to determine snapshots.
RDataPath	character(1)	file path to serialized form
Maintainer	character(1)	Maintainer name and email address, 'A Maintainer a.maintainer@email.addr '
BiocVersion	character(1)	Under which resource was built
Coordinate_1_based	logical(1)	Do coordinates start with 1 or 0?
DispatchClass	character(1)	string used to indicate which code should be called by the client when the resource is downloaded. This is often the same as the RDataClass. But it is allowed to be a different value so that the client can do something different internally if required.
Location_Prefix	character(1)	This was added for resources where the metadata only is stored and the resource itself comes from a third party web site. The location prefix says the base path where the resource is coming from, and the default value will be from our own site.
Notes	character()	Notes about the resource.
ahm		An instance of class AnnotationHubMetadata.
x		An instance of class AnnotationHubMetadata.
jsonpath	character(1)	full path to a JSON representation of AnnotationHubMetadata-class.

ahroot	A character(1) representing the value of AnnotationHubRoot to be added to the returned instance.
originalFile	Acharacter(1)
object	An AnnotationHubRecipe instance.
species	character(1) The organism, e.g., "Homo sapiens".
genome	character(1) The genome build, e.g., "hg19".
id	An id whose DB record is to be fully deleted (from gamay - not production)
objName	character(1) The name of the preparerClass object that you intend to have be used for dispatch. You can call it whatever you want as long as you don't use an existing preparerClass name
makeAnnotationHubMetadataFunction	function This is not a string, but just the name of the function that makes AHMs out of your resource of choice.
where	the environment in which to store the definition. Default value is sufficient.
...	additional arguments passed to methods

Value

AnnotationHubMetadata returns an instance of the class.

jsonPath returns a character(1)) representation of the full path to the location of the json file associated with this record.

toJson returns the JSON representation of the record.

fromJson returns an instance of the class, as parsed from the JSON file.

Objects from the Class

Objects can be created by calls to the constructor, AnnotationHubMetadata().

Author(s)

Dan Tenenbaum and Marc Carlson

Examples

```
getClass("AnnotationHubMetadata")
```

flog	<i>flog</i>
------	-------------

Description

Write logging message to console and a file.

Usage

```
flog(level, ...)
```

Arguments

level	A character(1) string object.
...	Further arguments.

Details

Writes the message to the console and to a file.

Value

None.

Author(s)

Dan Tenenbaum

ImportPreparer-class *Class ImportPreparer and generic newResources*

Description

The ImportPreparer and derived classes are used for dispatch during data discovery (see [newResources](#)). There is one ImportPreparer class for each data source for [AnnotationHubMetadata](#). `newResources` is a generic function; with methods implemented for each ImportPreparer.

Author(s)

Martin Morgan

See Also

[AnnotationHubMetadata](#).

Examples

```
getImportPreparerClasses()
```

makeEnsemblFasta *Functions to convert Ensembl FASTA files to FaFile and TwoBitFile for inclusion in AnnotationHub.*

Description

Transform an Ensembl FASTA file to a Bioconductor FaFile or ToBitFile.

Usage

```
makeEnsemblFastaToAHM(currentMetadata, baseUrl = "ftp://ftp.ensembl.org/pub/",
                      baseDir = "fasta/", regex,
                      justRunUnitTest = FALSE, BiocVersion = biocVersion())

makeEnsemblTwoBitToAHM(currentMetadata, baseUrl = "ftp://ftp.ensembl.org/pub/",
                       baseDir = "fasta/", regex,
                       justRunUnitTest = FALSE, BiocVersion = biocVersion())

ensemblFastaToFaFile(ahm)

ensemblFastaToTwoBitFile(ahm)
```

Arguments

currentMetadata	Currently not used. Intended to be a list of metadata to filter, i.e., records that do not need to be processed again. Need to remove or fix.
baseUrl	ftp file location.
baseDir	ftp file directory.
regex	Regular expression to match release version, e.g., "*.release-81".
justRunUnitTest	A logical. When TRUE, a small number of records (usually 5) are processed instead of all.
BiocVersion	A character vector of Bioconductor versions the resources should be available for.
ahm	List of AnnotationHubMetadata instances.

Details

TODO

Value

TODO

Author(s)

Bioconductor Core Team

See Also

- [updateResources](#)
- [AnnotationHubMetadata](#)

Examples

```
## updateResources() generates metadata, process records and
## pushes files to AWS S3 buckets. See ?updateResources for details.

## 'regex' is passed to makeEnsemblFastaToFaFile.
## Not run:
meta <- updateResources("/local/path",
                        BiocVersion = c("3.2", "3.3"),
                        preparerClasses = "EnsemblFastaImportPreparer",
                        metadataOnly = TRUE, insert = FALSE,
                        justRunUnitTest = FALSE, regex = ".*release-82")

## End(Not run)
```

makeGencodeFasta

Recipe to add Gencode FASTA resources to AnnotationHub

Description

Create metadata and process raw Gencode FASTA files for inclusion in AnnotationHub

Usage

```
makeGencodeFastaToAHM(currentMetadata,
                      baseUrl="ftp://ftp.sanger.ac.uk/pub/gencode/",
                      species=c("Human", "Mouse"), release,
                      justRunUnitTest=FALSE,
                      BiocVersion=biocVersion())

gencodeFastaToFaFile(ahm)
```

Arguments

currentMetadata	Currently not used. Intended to be a list of metadata to filter, i.e., records that do not need to be processed again. Need to remove or fix.
baseUrl	ftp file location.
species	A character(1) of the species. Currently "Human" and "Mouse" are supported.

release	A character string of the release number.
justRunUnitTest	A logical. When TRUE, a small number of records (usually 5) are processed instead of all.
BiocVersion	A character vector of Bioconductor versions the resources should be available for.
ahm	List of AnnotationHubMetadata instances.

Details

- Documentation: <http://www.gencodegenes.org/releases/>
- File download location: <ftp://ftp.sanger.ac.uk/pub/gencode/>. Gencode_human and Gencode_mouse are used.
- Files downloaded: Code is currently specific for human and mouse. Files chosen for download are described in `AnnotationHubData:::gencodeDescription()`.

Value

makeGencodeFastaAHM returns a list of AnnotationHubMetadata instances. gencodeFastaToFastaFile returns nothing.

Author(s)

Bioconductor Core Team.

See Also

- [updateResources](#)
- [AnnotationHubMetadata](#)

Examples

```
## updateResources() generates metadata, process records and
## pushes files to AWS S3 buckets.

## To run the GencodeFasta recipe specify
## 'preparerClasses = GencodeFastaImportPreparer'. The 'species' and 'release'
## arguments are passed to makeGencodeFastaAHM().
## Not run:
meta <- updateResources("/local/path",
                        BiocVersion = c("3.2", "3.3"),
                        preparerClasses = "GencodeFastaImportPreparer",
                        metadataOnly = TRUE, insert = FALSE,
                        justRunUnitTest = FALSE)

## End(Not run)
```

updateResources	<i>updateResources</i>
-----------------	------------------------

Description

Add new resources to AnnotationHub

Usage

```
updateResources(AnnotationHubRoot, BiocVersion = biocVersion(),
  preparerClasses = getImportPreparerClasses(),
  metadataOnly = TRUE, insert = FALSE,
  justRunUnitTest = FALSE, ...)
```

Arguments

AnnotationHubRoot	Local path where files will be downloaded.
BiocVersion	A character vector of Bioconductor versions the resources should be available for.
preparerClasses	One of the ImportPreparer subclasses defined in getImportPreparer(). This class is used for dispatch during data discovery.
metadataOnly	A logical to specify the processing of metadata only or both metadata and data files. When FALSE, metadata are generated and data files are downloaded, processed and pushed to their final location in S3 buckets. metadata = TRUE produces only metadata and is useful for testing.
insert	A logical to control if metadata are inserted in the AnnotationHub 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 "AH_SERVER_POST_URL" global option must be set to the http location of the AnnotationHubServer in the global environment or .Rprofile. Additionally, AWS command line tools must be installed on the local machine to push files to S3 buckets. See https://aws.amazon.com/cli/ for installation instructions.
justRunUnitTest	A logical. When TRUE, a small number of records (usually 5) are processed instead of all.
...	Arguments passed to other methods such as regex, baseUrl, baseDir.

Details

- updateResources:
updateResources is responsible for creating metadata records and downloading, processing and pushing data files to their final resting place. The

- preparerClasses argument is used in method dispatch to determine which recipe is used. By manipulating the metadataOnly, insert and justRunUnitTest arguments one can flexibly test the metadata for a small number of records with or without downloading and processing the data files.
- global options:
Several recipes look at the "AnnotationHub_Use_Disk" option to determine if the file is written to S3. This is legacy behavior and not clearly documented. If the recipe being run respects this option it must be set before running updateResources,
When insert = TRUE the "AH_SERVER_POST_URL" option must be set to the https location of the AnnotationHub db.

Value

A list of AnnotationHubMetadata objects.

Author(s)

Martin Morgan, Marc Carlson

See Also

- [AnnotationHubMetadata](#)

Examples

```
## Not run:

## -----
## Inspect metadata:
## -----
## A useful first step in testing a new recipe is to generate and
## inspect a small number of metadata records. The combination of
## 'metadataOnly=TRUE', 'insert=FALSE' and 'justRunUnitTest=TRUE'
## generates metadata for the first 5 records and does not download or
## process any data.

meta <- updateResources("/local/path",
                        BiocVersion = c("3.2", "3.3"),
                        preparerClasses = "EnsemblFastaImportPreparer",
                        metadataOnly = TRUE, insert = FALSE,
                        justRunUnitTest = TRUE,
                        regex = ".*release-82")

INFO [2015-11-12 07:58:05] Preparer Class: EnsemblFastaImportPreparer
Ailuropoda_melanoleuca.ailMel1.cdna.all.fa.gz
Ailuropoda_melanoleuca.ailMel1.dna_rm.toplevel.fa.gz
Ailuropoda_melanoleuca.ailMel1.dna_sm.toplevel.fa.gz
Ailuropoda_melanoleuca.ailMel1.dna.toplevel.fa.gz
Ailuropoda_melanoleuca.ailMel1.ncrna.fa.gz
```

```

## The return value is a list of metadata for the first 5 records:

> names(meta)
[1] "FASTA cDNA sequence for Ailuropoda melanoleuca"
[2] "FASTA DNA sequence for Ailuropoda melanoleuca"
[3] "FASTA DNA sequence for Ailuropoda melanoleuca"
[4] "FASTA DNA sequence for Ailuropoda melanoleuca"
[5] "FASTA ncRNA sequence for Ailuropoda melanoleuca"

## Each record is of class AnnotationHubMetadata:

> class(meta[[1]])
[1] "AnnotationHubMetadata"
attr(,"package")
[1] "AnnotationHubData"

## -----
## Insert metadata in the db and process/push data files:
## -----
## This next code chunk creates the metadata and downloads and processes
## the data. If all files are successfully pushed to their
## final resting place then metadata are inserted in the AnnotationHub db.

meta <- updateResources("local/path",
                        BiocVersion = c("3.2", "3.3"),
                        preparerClasses = "EnsemblFastaImportPreparer",
                        metadataOnly = FALSE, insert = TRUE,
                        justRunUnitTest = FALSE,
                        regex = ".*release-81")

## -----
## Recovery helpers:
## -----

## pushResources() and pushMetadata() are both called from updateResources()
## but can be used solo for testing or completing a run that
## terminated unexpectedly.

## Download, process and push to S3 the last 2 files in 'meta':
sub <- meta[length(meta) - 1:length(meta)]
pushResources(sub)

## Insert metadata in the AnotationHub db for the last 2 files in 'meta':

pushMetadata(sub, url = getOption("AH_SERVER_POST_URL"))

## End(Not run)

```

Description

This function is for uploading a file resource to the S3 cloud.

Usage

```
upload_to_S3(file, remotename, bucket, profile, acl="public-read")
```

Arguments

file	The file to upload.
remotename	The name this file should have in S3, including any "keys" that are part of the name. This should not start with a slash (if it does, the leading slash will be removed), but can contain forward slashes.
bucket	Name of the S3 bucket to copy to.
profile	Corresponds to a profile set in the config file for the AWS CLI (see the documentation). If this argument is omitted, the default profile is used.
acl	Should be one of private, public-read, or public-read-write.

Details

Uses the [AWS Command Line Interface](#) to copy a file to Amazon S3. Assumes the CLI is properly configured and that the aws program is in your PATH. The CLI should be configured with the credentials of a user who has permission to upload to the appropriate bucket. It's recommended to use [IAM](#) to set up users with limited permissions.

There is an `RAmazonS3` package but it seems to have issues uploading files to S3.

Value

TRUE on success. If the command fails, the function will exit with an error.

Author(s)

Dan Tenenbaum

Examples

```
## Not run:
upload_to_S3("myfile.txt", "foo/bar/baz/yourfile.txt")
# If this is successful, the file should be accessible at
# http://s3.amazonaws.com/annotationhub/foo/bar/baz/yourfile.txt

## End(Not run)
```

Index

- *Topic **classes**
 - AnnotationHubMetadata-class, 3
 - flog, 6
 - ImportPreparer-class, 6
- *Topic **methods**
 - makeEnsemblFasta, 7
 - makeGencodeFasta, 8
 - updateResources, 10
- *Topic **package**
 - AnnotationHubRecipes-package, 2
- ahmToJson
 - (AnnotationHubMetadata-class), 3
- amazon (upload_to_S3), 13
- AnnotationHubMetadata, 6, 8, 9, 11
- AnnotationHubMetadata
 - (AnnotationHubMetadata-class), 3
- AnnotationHubMetadata-class, 3
- AnnotationHubRecipes
 - (AnnotationHubRecipes-package), 2
- AnnotationHubRecipes-package, 2
- annotationHubRoot
 - (ImportPreparer-class), 6
- AWS (upload_to_S3), 13
- class:AnnotationHubMetadata
 - (AnnotationHubMetadata-class), 3
- class:HubMetadata
 - (AnnotationHubMetadata-class), 3
- constructAnnotationHubMetadataFromSourceFilePath
 - (AnnotationHubMetadata-class), 3
- constructMetadataFromJsonPath
 - (AnnotationHubMetadata-class), 3
- constructSeqInfo
 - (AnnotationHubMetadata-class), 3
- dbSNPVCFImportPreparer
 - (ImportPreparer-class), 6
- dbSNPVCFImportPreparer-class
 - (ImportPreparer-class), 6
- deleteResources
 - (AnnotationHubMetadata-class), 3
- EncodeImportPreparer
 - (ImportPreparer-class), 6
- EncodeImportPreparer-class
 - (ImportPreparer-class), 6
- EnsemblFastaImportPreparer
 - (ImportPreparer-class), 6
- EnsemblFastaImportPreparer-class
 - (ImportPreparer-class), 6
- ensemblFastaToFaFile
 - (makeEnsemblFasta), 7
- ensemblFastaToTwoBitFile
 - (makeEnsemblFasta), 7
- EnsemblGtfImportPreparer
 - (ImportPreparer-class), 6
- EnsemblGtfImportPreparer-class
 - (ImportPreparer-class), 6
- flog, 6
- gencodeFastaToFaFile
 - (makeGencodeFasta), 8
- getImportPreparer
 - (ImportPreparer-class), 6
- getImportPreparerClasses
 - (AnnotationHubMetadata-class), 3
- Grasp2ImportPreparer
 - (ImportPreparer-class), 6

- Grasp2ImportPreparer-class
(ImportPreparer-class), 6
- HaemCodeImportPreparer
(ImportPreparer-class), 6
- HaemCodeImportPreparer-class
(ImportPreparer-class), 6
- hubError (AnnotationHubMetadata-class),
3
- hubError, HubMetadata-method
(AnnotationHubMetadata-class),
3
- hubError, list-method
(AnnotationHubMetadata-class),
3
- hubError<-
(AnnotationHubMetadata-class),
3
- hubError<-, HubMetadata, character-method
(AnnotationHubMetadata-class),
3
- hubError<-, list, character-method
(AnnotationHubMetadata-class),
3
- HubMetadata
(AnnotationHubMetadata-class),
3
- HubMetadata-class
(AnnotationHubMetadata-class),
3
- HubMetadataFromJson
(AnnotationHubMetadata-class),
3
- ImportPreparer-class, 6
- Inparanoid8ImportPreparer
(ImportPreparer-class), 6
- Inparanoid8ImportPreparer-class
(ImportPreparer-class), 6
- inputFiles
(AnnotationHubMetadata-class),
3
- inputFiles, HubMetadata-method
(AnnotationHubMetadata-class),
3
- jsonPath (AnnotationHubMetadata-class),
3
- makeAnnotationHubResource
(AnnotationHubMetadata-class),
3
- makeEnsemblFasta, 7
- makeEnsemblFastaToAHM
(makeEnsemblFasta), 7
- makeEnsemblTwoBitToAHM
(makeEnsemblFasta), 7
- makeGencodeFasta, 8
- makeGencodeFastaToAHM
(makeGencodeFasta), 8
- metadata (AnnotationHubMetadata-class),
3
- metadata, HubMetadata-method
(AnnotationHubMetadata-class),
3
- metadata<-
(AnnotationHubMetadata-class),
3
- metadata<-, HubMetadata, list-method
(AnnotationHubMetadata-class),
3
- metadataList (ImportPreparer-class), 6
- metadataTable (ImportPreparer-class), 6
- NCBIImportPreparer
(ImportPreparer-class), 6
- NCBIImportPreparer-class
(ImportPreparer-class), 6
- newResources, 6
- newResources (ImportPreparer-class), 6
- newResources, dbSNPVCFImportPreparer-method
(ImportPreparer-class), 6
- newResources, EncodeImportPreparer-method
(ImportPreparer-class), 6
- newResources, EnsemblFastaImportPreparer-method
(ImportPreparer-class), 6
- newResources, EnsemblGtfImportPreparer-method
(ImportPreparer-class), 6
- newResources, Grasp2ImportPreparer-method
(ImportPreparer-class), 6
- newResources, HaemCodeImportPreparer-method
(ImportPreparer-class), 6
- newResources, ImportPreparer-method
(ImportPreparer-class), 6
- newResources, Inparanoid8ImportPreparer-method
(ImportPreparer-class), 6
- newResources, NCBIImportPreparer-method
(ImportPreparer-class), 6

newResources,RefNetImportPreparer-method
 (ImportPreparer-class), 6

newResources,UCSCChainPreparer-method
 (ImportPreparer-class), 6

newResources,UCSCFullTrackImportPreparer-method
 (ImportPreparer-class), 6

newResources,UCSCTrackImportPreparer-method
 (ImportPreparer-class), 6

UCSCChainPreparer-class
 (ImportPreparer-class), 6

UCSCTrackImportPreparer
 (ImportPreparer-class), 6

UCSCTrackImportPreparer-class
 (ImportPreparer-class), 6

updateResources, 8, 9, 10

upload_to_S3, 12

outputFile
 (AnnotationHubMetadata-class),
 3

outputFile,HubMetadata-method
 (AnnotationHubMetadata-class),
 3

pushMetadata (updateResources), 10

pushResources (updateResources), 10

recipeName
 (AnnotationHubMetadata-class),
 3

recipeName,HubMetadata-method
 (AnnotationHubMetadata-class),
 3

RefNetImportPreparer
 (ImportPreparer-class), 6

RefNetImportPreparer-class
 (ImportPreparer-class), 6

run (AnnotationHubMetadata-class), 3

run,AnnotationHubMetadata-method
 (AnnotationHubMetadata-class),
 3

runRecipes (updateResources), 10

runRecipes,AnnotationHubMetadata-method
 (updateResources), 10

S3 (upload_to_S3), 13

show (AnnotationHubMetadata-class), 3

show,HubMetadata-method
 (AnnotationHubMetadata-class),
 3

show,ImportPreparer-method
 (ImportPreparer-class), 6

sourceUrls (ImportPreparer-class), 6

toJson (AnnotationHubMetadata-class), 3

UCSCChainPreparer
 (ImportPreparer-class), 6