

AnnotationHub: Access the AnnotationHub Web Service

Introduction

Finding and using public genomics data such as browser or chip-seq tracks; annotation for genes, exons, transcripts; gene ontology and functional gene information; etc. often requires quite a bit of work. Bioconductor has done some of this work already by

1. Finding and curating popular genomic resources
2. Using “recipes” to create R object versions of these resources
3. Make those resources available as a web service that is accessible from R

The `AnnotationHub` server provides easy *R / Bioconductor* access to large collections of publicly available whole genome resources, e.g., ENSEMBL genome fasta or gtf files, UCSC chain resources, ENCODE data tracks at UCSC, etc.

To get started, make sure that you have the `AnnotationHub` package installed:

```
source('https://bioconductor.org/biocLite.R')
biocLite('AnnotationHub')
```

AnnotationHub objects

The *AnnotationHub* package provides a client interface to resources stored at the AnnotationHub web service.

```
library(AnnotationHub)
```

The *AnnotationHub* package is straightforward to use. Create an `AnnotationHub` object

```
ah = AnnotationHub()
```

```
## snapshotDate(): 2018-04-30
```

Now at this point you have already done everything you need in order to start retrieving annotations. For most operations, using the `AnnotationHub` object should feel a lot like working with a familiar `list` or `data.frame`.

Lets take a minute to look at the `show` method for the hub object `ah`

```
ah

## AnnotationHub with 44923 records
## # snapshotDate(): 2018-04-30
## # $dataprovder: BroadInstitute, Ensembl, UCSC, ftp://ftp.ncbi.nlm.nih....
## # $species: Homo sapiens, Mus musculus, Drosophila melanogaster, Bos ta...
## # $rdataclass: GRanges, BigWigFile, FaFile, TwoBitFile, Rle, OrgDb, Cha...
## # additional mcols(): taxonomyid, genome, description,
## #   coordinate_1_based, maintainer, rdatadateadded, preparerclass,
## #   tags, rdatapath, sourceurl, sourcetype
## # retrieve records with, e.g., 'object[["AH2"]]'
##
##           title
## AH2      | Ailuropoda_melanoleuca.ailMel1.69.dna.toplevel.fa
```

```
## AH3      | Ailuropoda_melanoleuca.ailMel1.69.dna_rm.toplevel.fa
## AH4      | Ailuropoda_melanoleuca.ailMel1.69.dna_sm.toplevel.fa
## AH5      | Ailuropoda_melanoleuca.ailMel1.69.ncrna.fa
## AH6      | Ailuropoda_melanoleuca.ailMel1.69.pep.all.fa
## ...      ...
## AH63653  | phastCons46wayPrimates.UCSC.hg19.chrUn_gl000248.rds
## AH63654  | phastCons46wayPrimates.UCSC.hg19.chrUn_gl000249.rds
## AH63655  | phastCons46wayPrimates.UCSC.hg19.chrX.rds
## AH63656  | phastCons46wayPrimates.UCSC.hg19.chrY.rds
## AH63657  | Alternative Splicing Annotation for Homo sapiens (Human)
```

You can see that it gives you an idea about the different types of data that are present inside the hub. You can see where the data is coming from (dataprovder), as well as what species have samples present (species), what kinds of R data objects could be returned (rdataclass). We can take a closer look at all the kinds of data providers that are available by simply looking at the contents of dataprovder as if it were the column of a data.frame object like this:

```
unique(ah$dataprovder)
```

```
## [1] "Ensembl"
## [2] "UCSC"
## [3] "RefNet"
## [4] "Inparanoid8"
## [5] "NHLBI"
## [6] "ChEA"
## [7] "Pazar"
## [8] "NIH Pathway Interaction Database"
## [9] "Haemcode"
## [10] "BroadInstitute"
## [11] "PRIDE"
## [12] "Gencode"
## [13] "CRIBI"
## [14] "Genoscope"
## [15] "MISO, VAST-TOOLS, UCSC"
## [16] "UWashington"
## [17] "Stanford"
## [18] "dbSNP"
## [19] "BioMart"
## [20] "GeneOntology"
## [21] "KEGG"
## [22] "URGI"
## [23] "ftp://ftp.ncbi.nlm.nih.gov/gene/DATA/"
```

In the same way, you can also see data from different species inside the hub by looking at the contents of species like this:

```
head(unique(ah$species))
```

```
## [1] "Ailuropoda melanoleuca" "Anolis carolinensis"
## [3] "Bos taurus"           "Caenorhabditis elegans"
## [5] "Callithrix jacchus"   "Canis familiaris"
```

And this will also work for any of the other types of metadata present. You can learn which kinds of metadata are available by simply hitting the tab key after you type 'ah\$'. In this way you can explore for yourself what kinds of data are present in the hub right from the command line. This interface also allows you to access the hub programatically to extract data that matches a particular set of criteria.

Another valuable types of metadata to pay attention to is the `rdataclass`.

```
head(unique(ah$rdataclass))
```

```
## [1] "FaFile"          "GRanges"          "data.frame"        "Inparanoid8Db"  
## [5] "TwoBitFile"      "ChainFile"
```

The `rdataclass` allows you to see which kinds of R objects the hub will return to you. This kind of information is valuable both as a means to filter results and also as a means to explore and learn about some of the kinds of annotation objects that are widely available for the project. Right now this is a pretty short list, but over time it should grow as we support more of the different kinds of annotation objects via the hub.

Now lets try getting the Chain Files from UCSC using the query and subset methods to selectively pare down the hub based on specific criteria. The query method lets you search rows for specific strings, returning an `AnnotationHub` instance with just the rows matching the query.

From the show method, one can easily see that one of the dataprovider is UCSC and there is a `rdataclass` for `ChainFile`

One can get chain files for *Drosophila melanogaster* from UCSC with:

```
dm <- query(ah, c("ChainFile", "UCSC", "Drosophila melanogaster"))  
dm
```

```
## AnnotationHub with 45 records  
## # snapshotDate(): 2018-04-30  
## # $dataprovider: UCSC  
## # $species: Drosophila melanogaster  
## # $rdataclass: ChainFile  
## # additional mcols(): taxonomyid, genome, description,  
## #   coordinate_1_based, maintainer, rdatadateadded, preparerclass,  
## #   tags, rdatapath, sourceurl, sourcetype  
## # retrieve records with, e.g., 'object[["AH15102"]]'  
##  
##           title  
## AH15102 | dm3ToAnoGam1.over.chain.gz  
## AH15103 | dm3ToApiMel3.over.chain.gz  
## AH15104 | dm3ToDm2.over.chain.gz  
## AH15105 | dm3ToDm6.over.chain.gz  
## AH15106 | dm3ToDp3.over.chain.gz  
## ...     ...  
## AH15142 | dm2ToDroVir3.over.chain.gz  
## AH15143 | dm2ToDroWil1.over.chain.gz  
## AH15144 | dm2ToDroYak1.over.chain.gz  
## AH15145 | dm2ToDroYak2.over.chain.gz  
## AH15146 | dm1ToDm2.over.chain.gz
```

Query has worked and you can now see that the only species present is *Drosophila melanogaster*.

The metadata underlying this hub object can be retrieved by you

```
df <- mcols(dm)
```

By default the show method will only display the first 5 and last 5 rows. There are already thousands of records present in the hub.

```
length(ah)
```

```
## [1] 44923
```

Lets look at another example, where we pull down only Inparanoid8 data from the hub and use subset to return a smaller base object (here we are finding cases where the genome column is set to panda).

```
ahs <- query(ah, c('inparanoid8', 'ailuropoda'))
ahs
```

```
## AnnotationHub with 1 record
## # snapshotDate(): 2018-04-30
## # names(): AH10451
## # $dataprovder: Inparanoid8
## # $species: Ailuropoda melanoleuca
## # $rdataclass: Inparanoid8Db
## # $rdatadateadded: 2014-03-31
## # $title: hom.Ailuropoda_melanoleuca.inp8.sqlite
## # $description: Inparanoid 8 annotations about Ailuropoda melanoleuca
## # $taxonomyid: 9646
## # $genome: inparanoid8 genomes
## # $sourcetype: Inparanoid
## # $sourceurl: http://inparanoid.sbc.su.se/download/current/Orthologs/A....
## # $sourcesize: NA
## # $tags: c("Inparanoid", "Gene", "Homology", "Annotation")
## # retrieve record with 'object[["AH10451"]]'
```

We can also look at the AnnotationHub object in a browser using the `display()` function. We can then filter the AnnotationHub object for `__chainFile__` by either using the Global search field on the top right corner of the page or the in-column search field for 'rdataclass'.

```
d <- display(ah)
```

The data from your data.frame - Mozilla Firefox (on gamay)

The data from your d... x

127.0.0.1:5758

Selected rows: AH14107 AH14109

Return rows to R session

Show 10 entries Search:

idx	dataprovider	species	genome	description	tags	rdataclass	sourcetype
AH14107	UCSC	Homo sapiens	hg38	UCSC liftOver chain file from hg38 to canFam3	liftOver, chain, UCSC, genome, homology	ChainFile	Chain
AH14108	UCSC	Homo sapiens	hg38	UCSC liftOver chain file from hg38 to hg19	liftOver, chain, UCSC, genome, homology	ChainFile	Chain
AH14109	UCSC	Homo sapiens	hg38	UCSC liftOver chain file from hg38 to mm10	liftOver, chain, UCSC, genome, homology	ChainFile	Chain
AH14110	UCSC	Homo sapiens	hg38	UCSC liftOver chain file from hg38 to monDom5	liftOver, chain, UCSC, genome, homology	ChainFile	Chain
AH14111	UCSC	Homo sapiens	hg38	UCSC liftOver chain file from hg38 to panTro4	liftOver, chain, UCSC, genome, homology	ChainFile	Chain
AH14112	UCSC	Homo sapiens	hg38	UCSC liftOver chain file from hg38 to rheMac3	liftOver, chain, UCSC, genome, homology	ChainFile	Chain
AH14113	UCSC	Homo sapiens	hg38	UCSC liftOver chain file from hg38 to rn5	liftOver, chain, UCSC, genome, homology	ChainFile	Chain
AH14114	UCSC	Homo sapiens	hg19	UCSC liftOver chain file from hg19 to allMel1	liftOver, chain, UCSC, genome, homology	ChainFile	Chain
AH14115	UCSC	Homo sapiens	hg19	UCSC liftOver chain file from hg19 to allMis1	liftOver, chain, UCSC, genome, homology	ChainFile	Chain
AH14116	UCSC	Homo sapiens	hg19	UCSC liftOver chain file from hg19 to anoCar1	liftOver, chain, UCSC, genome, homology	ChainFile	Chain

Showing 1 to 10 of 1,113 entries (filtered from 19,268 total entries) Previous 1 2 3 4 5 ... 8 Next

Displaying and filtering the Annotation Hub object in a browser

By default 1000 entries are displayed per page, we can change this using the filter on the top of the page or navigate through different pages using the page scrolling feature at the bottom of the page.

We can also select the rows of interest to us and send them back to the R session using 'Return rows to R session' button ; this sets a filter internally which filters the `AnnotationHub` object. The names of the selected AnnotationHub elements displayed at the top of the page.

Using AnnotationHub to retrieve data

Looking back at our chain file example, if we are interested in the file `dm1ToDm2.over.chain.gz`, we can get its metadata using

```
dm
```

```
## AnnotationHub with 45 records
## # snapshotDate(): 2018-04-30
## # $datapprovider: UCSC
## # $species: Drosophila melanogaster
## # $rdaclass: ChainFile
## # additional mcols(): taxonomyid, genome, description,
## #   coordinate_1_based, maintainer, rdatadateadded, preparerclass,
## #   tags, rdatapath, sourceurl, sourcetype
## # retrieve records with, e.g., 'object[["AH15102"]]'
##
##           title
## AH15102 | dm3ToAnoGam1.over.chain.gz
## AH15103 | dm3ToApiMel3.over.chain.gz
## AH15104 | dm3ToDm2.over.chain.gz
## AH15105 | dm3ToDm6.over.chain.gz
## AH15106 | dm3ToDp3.over.chain.gz
## ...
## AH15142 | dm2ToDroVir3.over.chain.gz
## AH15143 | dm2ToDroWil1.over.chain.gz
## AH15144 | dm2ToDroYak1.over.chain.gz
## AH15145 | dm2ToDroYak2.over.chain.gz
## AH15146 | dm1ToDm2.over.chain.gz
```

```
dm["AH15146"]
```

```
## AnnotationHub with 1 record
## # snapshotDate(): 2018-04-30
## # names(): AH15146
## # $datapprovider: UCSC
## # $species: Drosophila melanogaster
## # $rdaclass: ChainFile
## # $rdatadateadded: 2014-12-15
## # $title: dm1ToDm2.over.chain.gz
## # $description: UCSC liftOver chain file from dm1 to dm2
## # $taxonomyid: 7227
## # $genome: dm1
## # $sourcetype: Chain
## # $sourceurl: http://hgdownload.cse.ucsc.edu/goldenpath/dm1/liftOver/dm...
## # $sourcesize: NA
## # $tags: c("liftOver", "chain", "UCSC", "genome", "homology")
## # retrieve record with 'object[["AH15146"]]'
```

We can download the file using

```
dm[["AH15146"]]
```

```
## require("rtracklayer")
## downloading 0 resources
## loading from cache
```

```
##      '/Users/sdavis2//.AnnotationHub/19241'
```

```
## Chain of length 11
```

```
## names(11): chr2L chr2R chr3L chr3R chr4 chrX chrU chr2h chr3h chrXh chrYh
```

Each file is retrieved from the AnnotationHub server and the file is also cache locally, so that the next time you need to retrieve it, it should download much more quickly.

Accessing Genome-Scale Data

Non-model organism gene annotations

Bioconductor offers pre-built `org.*` annotation packages for model organisms, with their use described in the OrgDb section of the Annotation work flow. Here we discover available OrgDb objects for less-model organisms

```
library(AnnotationHub)
ah <- AnnotationHub()
```

```
## snapshotDate(): 2018-04-30
```

```
query(ah, "OrgDb")
```

```
## AnnotationHub with 1691 records
```

```
## # snapshotDate(): 2018-04-30
```

```
## # $datapvider: ftp://ftp.ncbi.nlm.nih.gov/gene/DATA/
```

```
## # $species: Escherichia coli, 'Caballeronia concitans', 'Chlorella vulg...
```

```
## # $rdataclass: OrgDb
```

```
## # additional mcols(): taxonomyid, genome, description,
```

```
## #   coordinate_1_based, maintainer, rdatadateadded, preparerclass,
```

```
## #   tags, rdatapath, sourceurl, sourcetype
```

```
## # retrieve records with, e.g., 'object[["AH61768"]]'
```

```
##
```

```
##           title
```

```
## AH61768 | org.Ag.eg.db.sqlite
```

```
## AH61769 | org.At.tair.db.sqlite
```

```
## AH61770 | org.Bt.eg.db.sqlite
```

```
## AH61771 | org.Cf.eg.db.sqlite
```

```
## AH61772 | org.Gg.eg.db.sqlite
```

```
## ...      ...
```

```
## AH63468 | org.Salmonella_typhimurium_LT2.eg.sqlite
```

```
## AH63469 | org.Acinetobacter_baumannii.eg.sqlite
```

```
## AH63470 | org.Acinetobacter_genomosp._2.eg.sqlite
```

```
## AH63471 | org.Acinetobacter_genomospecies_2.eg.sqlite
```

```
## AH63472 | org.Bacterium_anitratum.eg.sqlite
```

```
orgdb <- query(ah, "OrgDb")[[1]]
```

```
## downloading 0 resources
```

```
## loading from cache
```

```
##      '/Users/sdavis2//.AnnotationHub/68514'
```

```
## Loading required package: AnnotationDbi
```

```
## Loading required package: Biobase
```

```
## Welcome to Bioconductor
##
## Vignettes contain introductory material; view with
## 'browseVignettes()'. To cite Bioconductor, see
## 'citation("Biobase")', and for packages 'citation("pkgname")'.
##
## Attaching package: 'Biobase'
##
## The following object is masked from 'package:AnnotationHub':
##
## cache
```

The object returned by AnnotationHub is directly usable with the `select()` interface, e.g., to discover the available keytypes for querying the object, the columns that these keytypes can map to, and finally selecting the SYMBOL and GENENAME corresponding to the first 6 ENTREZIDS

```
keytypes(orgdb)
```

```
## [1] "ACCNUM"      "ENSEMBL"      "ENSEMBLPROT"  "ENSEMBLTRANS"
## [5] "ENTREZID"    "ENZYME"       "EVIDENCE"     "EVIDENCEALL"
## [9] "GENENAME"    "GO"           "GOALL"        "ONTOLOGY"
## [13] "ONTOLOGYALL" "PATH"         "PMID"         "REFSEQ"
## [17] "SYMBOL"      "UNIGENE"      "UNIPROT"
```

```
columns(orgdb)
```

```
## [1] "ACCNUM"      "ENSEMBL"      "ENSEMBLPROT"  "ENSEMBLTRANS"
## [5] "ENTREZID"    "ENZYME"       "EVIDENCE"     "EVIDENCEALL"
## [9] "GENENAME"    "GO"           "GOALL"        "ONTOLOGY"
## [13] "ONTOLOGYALL" "PATH"         "PMID"         "REFSEQ"
## [17] "SYMBOL"      "UNIGENE"      "UNIPROT"
```

```
egid <- head(keys(orgdb, "ENTREZID"))
```

```
select(orgdb, egid, c("SYMBOL", "GENENAME"), "ENTREZID")
```

```
## 'select()' returned 1:1 mapping between keys and columns
```

```
## ENTREZID      SYMBOL      GENENAME
## 1  1267437 AgaP_AGAP012606 AGAP012606-PA
## 2  1267439 AgaP_AGAP012559 AGAP012559-PA
## 3  1267440 AgaP_AGAP012558 AGAP012558-PA
## 4  1267447 AgaP_AGAP012586 AGAP012586-PA
## 5  1267450 AgaP_AGAP012834 AGAP012834-PA
## 6  1267459 AgaP_AGAP012589 AGAP012589-PA
```

Roadmap Epigenomics Project

All Roadmap Epigenomics files are hosted here. If one had to download these files on their own, one would navigate through the web interface to find useful files, then use something like the following R code.

```
url <- "http://egg2.wustl.edu/roadmap/data/byFileType/peaks/consolidated/broadPeak/E001-H3K4me1.broadPeak"
filename <- basename(url)
download.file(url, destfile=filename)
if (file.exists(filename))
  data <- import(filename, format="bed")
```


This would have to be repeated for all files, and the onus would lie on the user to identify, download, import, and manage the local disk location of these files.

AnnotationHub reduces this task to just a few lines of *R* code

```
library(AnnotationHub)
ah = AnnotationHub()
```

```
## snapshotDate(): 2018-04-30
```

```
epiFiles <- query(ah, "EpigenomeRoadMap")
```

A look at the value returned by `epiFiles` shows us that 18248 roadmap resources are available via *AnnotationHub*. Additional information about the files is also available, e.g., where the files came from (dataprovder), genome, species, sourceurl, sourcetypes.

```
epiFiles
```

```
## AnnotationHub with 18248 records
## # snapshotDate(): 2018-04-30
## # $dataprovder: BroadInstitute
## # $species: Homo sapiens
## # $rdaclass: BigWigFile, GRanges, data.frame
## # additional mcols(): taxonomyid, genome, description,
## #   coordinate_1_based, maintainer, rdatadateadded, preparerclass,
## #   tags, rdatapath, sourceurl, sourcetype
## # retrieve records with, e.g., 'object[["AH28856"]]'
##
##           title
## AH28856 | E001-H3K4me1.broadPeak.gz
## AH28857 | E001-H3K4me3.broadPeak.gz
## AH28858 | E001-H3K9ac.broadPeak.gz
## AH28859 | E001-H3K9me3.broadPeak.gz
## AH28860 | E001-H3K27me3.broadPeak.gz
## ...
## AH49540 | E058_mCRF_FractionalMethylation.bigwig
## AH49541 | E059_mCRF_FractionalMethylation.bigwig
## AH49542 | E061_mCRF_FractionalMethylation.bigwig
## AH49543 | E081_mCRF_FractionalMethylation.bigwig
## AH49544 | E082_mCRF_FractionalMethylation.bigwig
```

A good sanity check to ensure that we have files only from the Roadmap Epigenomics project is to check that all the files in the returned smaller hub object come from *Homo sapiens* and the hg19 genome

```
unique(epiFiles$species)
```

```
## [1] "Homo sapiens"
```

```
unique(epiFiles$genome)
```

```
## [1] "hg19"
```

Broadly, one can get an idea of the different files from this project looking at the sourcetype

```
table(epiFiles$sourcetype)
```

```
##
##   BED BigWig   GTF   tab   Zip
##  8298   9932     3     1    14
```

To get a more descriptive idea of these different files one can use:

```
sort(table(epiFiles$description), decreasing=TRUE)
```

```
##
##           Bigwig File containing -log10(p-value) signal tracks from EpigenomeRoadMap Proj
##
##           Bigwig File containing fold enrichment signal tracks from EpigenomeRoadMap Proj
##
##           Narrow ChIP-seq peaks for consolidated epigenomes from EpigenomeRoadMap Proj
##
##           Broad ChIP-seq peaks for consolidated epigenomes from EpigenomeRoadMap Proj
##
##           Gapped ChIP-seq peaks for consolidated epigenomes from EpigenomeRoadMap Proj
##
##           Narrow DNasePeaks for consolidated epigenomes from EpigenomeRoadMap Proj
##
##           15 state chromatin segmentations from EpigenomeRoadMap Proj
##
## Broad domains on enrichment for DNase-seq for consolidated epigenomes from EpigenomeRoadMap Proj
##
##           RRBS fractional methylation calls from EpigenomeRoadMap Proj
##
##           Whole genome bisulphite fractional methylation calls from EpigenomeRoadMap Proj
##
##           MeDIP/MRE(mCRF) fractional methylation calls from EpigenomeRoadMap Proj
##
## GencodeV10 gene/transcript coordinates and annotations corresponding to hg19 version of the human genome
##
##           RNA-seq read count matrix for intronic protein-coding RNA elements
##
##           RNA-seq read counts matrix for ribosomal gene expression
##
##           RPKM expression matrix for ribosomal gene expression
##
##           Metadata for EpigenomeRoadMap Project
##
##           RNA-seq read counts matrix for non-coding RNA elements
##
##           RNA-seq read counts matrix for protein coding genes
##
##           RNA-seq read counts matrix for protein coding genes
##
##           RNA-seq read counts matrix for ribosomal genes
##
##           RPKM expression matrix for non-coding RNA elements
##
##           RPKM expression matrix for protein coding genes
##
##           RPKM expression matrix for protein coding genes
##
##           RPKM expression matrix for ribosomal genes
```

The ‘metadata’ provided by the Roadmap Epigenomics Project is also available. Note that the information

displayed about a hub with a single resource is quite different from the information displayed when the hub references more than one resource.

```
metadata.tab <- query(ah , c("EpigenomeRoadMap", "Metadata"))
metadata.tab
```

```
## AnnotationHub with 1 record
## # snapshotDate(): 2018-04-30
## # names(): AH41830
## # $dataprovder: BroadInstitute
## # $species: Homo sapiens
## # $rdataclass: data.frame
## # $rdatadateadded: 2015-05-11
## # $title: EID_metadata.tab
## # $description: Metadata for EpigenomeRoadMap Project
## # $taxonomyid: 9606
## # $genome: hg19
## # $sourcetype: tab
## # $sourceurl: http://egg2.wustl.edu/roadmap/data/byFileType/metadata/EI...
## # $sourcesize: 18035
## # $tags: c("EpigenomeRoadMap", "Metadata")
## # retrieve record with 'object[["AH41830"]]'
```

So far we have been exploring information about resources, without downloading the resource to a local cache and importing it into R. One can retrieve the resource using `[[` as indicated at the end of the `show` method

```
## downloading 0 resources

## loading from cache
##      '/Users/sdavis2//.AnnotationHub/47270'

metadata.tab <- ah[["AH41830"]]
```

```
## downloading 0 resources

## loading from cache
##      '/Users/sdavis2//.AnnotationHub/47270'
```

The `metadata.tab` file is returned as a *data.frame*. The first 6 rows of the first 5 columns are shown here:

```
metadata.tab[1:6, 1:5]
```

	EID	GROUP	COLOR	MNEMONIC
## 1	E001	ESC	#924965	ESC.I3
## 2	E002	ESC	#924965	ESC.WA7
## 3	E003	ESC	#924965	ESC.H1
## 4	E004	ES-deriv	#4178AE	ESDR.H1.BMP4.MESO
## 5	E005	ES-deriv	#4178AE	ESDR.H1.BMP4.TROP
## 6	E006	ES-deriv	#4178AE	ESDR.H1.MSC
##				STD_NAME
## 1				ES-I3 Cells
## 2				ES-WA7 Cells
## 3				H1 Cells
## 4	H1 BMP4 Derived	Mesendoderm	Cultured	Cells
## 5	H1 BMP4 Derived	Trophoblast	Cultured	Cells
## 6	H1 Derived	Mesenchymal	Stem	Cells

One can keep constructing different queries using multiple arguments to trim down these 18248 to get the files one wants. For example, to get the ChIP-Seq files for consolidated epigenomes, one could use

```
bpChIPepi <- query(ah , c("EpigenomeRoadMap", "broadPeak", "chip", "consolidated"))
```

To get all the bigWig signal files, one can query the hub using

```
allBigWigFiles <- query(ah, c("EpigenomeRoadMap", "BigWig"))
```

To access the 15 state chromatin segmentations, one can use

```
seg <- query(ah, c("EpigenomeRoadMap", "segmentations"))
```

If one is interested in getting all the files related to one sample

```
E126 <- query(ah , c("EpigenomeRoadMap", "E126", "H3K4ME2"))
E126
```

```
## AnnotationHub with 6 records
## # snapshotDate(): 2018-04-30
## # $datapprovider: BroadInstitute
## # $species: Homo sapiens
## # $rdataclass: BigWigFile, GRanges
## # additional mcols(): taxonomyid, genome, description,
## #   coordinate_1_based, maintainer, rdatadateadded, preparerclass,
## #   tags, rdatapath, sourceurl, sourcetype
## # retrieve records with, e.g., 'object[["AH29817"]]'
##
##           title
## AH29817 | E126-H3K4me2.broadPeak.gz
## AH30868 | E126-H3K4me2.narrowPeak.gz
## AH31801 | E126-H3K4me2.gappedPeak.gz
## AH32990 | E126-H3K4me2.fc.signal.bigwig
## AH34022 | E126-H3K4me2.pval.signal.bigwig
## AH40177 | E126-H3K4me2.imputed.pval.signal.bigwig
```

Hub resources can also be selected using `$`, `subset()`, and `display()`; see the main *AnnotationHub* vignette for additional detail.

Hub resources are imported as the appropriate *Bioconductor* object for use in further analysis. For example, peak files are returned as *GRanges* objects.

```
## downloading 0 resources
## loading from cache
##      '/Users/sdavis2//.AnnotationHub/35257'
##
## Attaching package: 'Biostrings'
## The following object is masked from 'package:base':
##
##      strsplit
peaks <- E126[['AH29817']]
```

```
## downloading 0 resources
## loading from cache
##      '/Users/sdavis2//.AnnotationHub/35257'
seqinfo(peaks)
```

```
## Seqinfo object with 93 sequences (1 circular) from hg19 genome:
```

```
##      seqnames      seqlengths isCircular genome
##      chr1          249250621      FALSE   hg19
##      chr2          243199373      FALSE   hg19
##      chr3          198022430      FALSE   hg19
##      chr4          191154276      FALSE   hg19
##      chr5          180915260      FALSE   hg19
##      ...           ...           ...     ...
##      chrUn_gl000245 36651        FALSE   hg19
##      chrUn_gl000246 38154        FALSE   hg19
##      chrUn_gl000247 36422        FALSE   hg19
##      chrUn_gl000248 39786        FALSE   hg19
##      chrUn_gl000249 38502        FALSE   hg19
```

BigWig files are returned as *BigWigFile* objects. A *BigWigFile* is a reference to a file on disk; the data in the file can be read in using `rtracklayer::import()`, perhaps querying these large files for particular genomic regions of interest as described on the help page `?import.bw`.

Each record inside *AnnotationHub* is associated with a unique identifier. Most *GRanges* objects returned by *AnnotationHub* contain the unique *AnnotationHub* identifier of the resource from which the *GRanges* is derived. This can come handy when working with the *GRanges* object for a while, and additional information about the object (e.g., the name of the file in the cache, or the original sourceurl for the data underlying the resource) that is being worked with.

```
metadata(peaks)
```

```
## $AnnotationHubName
## [1] "AH29817"
##
## $`File Name`
## [1] "E126-H3K4me2.broadPeak.gz"
##
## $`Data Source`
## [1] "http://egg2.wustl.edu/roadmap/data/byFileType/peaks/consolidated/broadPeak/E126-H3K4me2.broadPeak.gz"
##
## $Provider
## [1] "BroadInstitute"
##
## $Organism
## [1] "Homo sapiens"
##
## $`Taxonomy ID`
## [1] 9606
```

```
ah[metadata(peaks)$AnnotationHubName]$sourceurl
```

```
## [1] "http://egg2.wustl.edu/roadmap/data/byFileType/peaks/consolidated/broadPeak/E126-H3K4me2.broadPeak.gz"
```

Configuring AnnotationHub objects

When you create the *AnnotationHub* object, it will set up the object for you with some default settings. See `?AnnotationHub` for ways to customize the hub source, the local cache, and other instance-specific options, and `?getAnnotationHubOption` to get or set package-global options for use across sessions.

If you look at the object you will see some helpful information about it such as where the data is cached and where online the hub server is set to.

```
ah
```

```
## AnnotationHub with 44923 records
## # snapshotDate(): 2018-04-30
## # $dataProvider: BroadInstitute, Ensembl, UCSC, ftp://ftp.ncbi.nlm.nih....
## # $species: Homo sapiens, Mus musculus, Drosophila melanogaster, Bos ta...
## # $rdataclass: GRanges, BigWigFile, FaFile, TwoBitFile, Rle, OrgDb, Cha...
## # additional mcols(): taxonomyid, genome, description,
## #   coordinate_1_based, maintainer, rdatadateadded, preparerclass,
## #   tags, rdatapath, sourceurl, sourcetype
## # retrieve records with, e.g., 'object[["AH2"]]'
##
##           title
## AH2      | Ailuropoda_melanoleuca.ailMel1.69.dna.toplevel.fa
## AH3      | Ailuropoda_melanoleuca.ailMel1.69.dna_rm.toplevel.fa
## AH4      | Ailuropoda_melanoleuca.ailMel1.69.dna_sm.toplevel.fa
## AH5      | Ailuropoda_melanoleuca.ailMel1.69.ncrna.fa
## AH6      | Ailuropoda_melanoleuca.ailMel1.69.pep.all.fa
## ...      ...
## AH63653 | phastCons46wayPrimates.UCSC.hg19.chrUn_gl000248.rds
## AH63654 | phastCons46wayPrimates.UCSC.hg19.chrUn_gl000249.rds
## AH63655 | phastCons46wayPrimates.UCSC.hg19.chrX.rds
## AH63656 | phastCons46wayPrimates.UCSC.hg19.chrY.rds
## AH63657 | Alternative Splicing Annotation for Homo sapiens (Human)
```

By default the `AnnotationHub` object is set to the latest `snapshotData` and a snapshot version that matches the version of *Bioconductor* that you are using. You can also learn about these data with the appropriate methods.

```
snapshotDate(ah)
```

```
## [1] "2018-04-30"
```

If you are interested in using an older version of a snapshot, you can list previous versions with the `possibleDates()` like this:

```
pd <- possibleDates(ah)
pd
```

```
## [1] "2013-03-19" "2013-03-21" "2013-03-26" "2013-04-04" "2013-04-29"
## [6] "2013-06-24" "2013-06-25" "2013-06-26" "2013-06-27" "2013-10-29"
## [11] "2013-11-20" "2013-12-19" "2014-02-12" "2014-02-13" "2014-03-31"
## [16] "2014-04-27" "2014-05-11" "2014-05-13" "2014-05-14" "2014-05-22"
## [21] "2014-07-02" "2014-07-09" "2014-12-15" "2014-12-24" "2015-01-08"
## [26] "2015-01-14" "2015-03-09" "2015-03-11" "2015-03-12" "2015-03-25"
## [31] "2015-03-26" "2015-05-06" "2015-05-07" "2015-05-08" "2015-05-11"
## [36] "2015-05-14" "2015-05-21" "2015-05-22" "2015-05-26" "2015-07-17"
## [41] "2015-07-27" "2015-07-31" "2015-08-10" "2015-08-13" "2015-08-14"
## [46] "2015-08-17" "2015-08-26" "2015-12-28" "2015-12-29" "2016-01-25"
## [51] "2016-03-07" "2016-05-03" "2016-05-25" "2016-06-06" "2016-07-20"
## [56] "2016-08-15" "2016-10-11" "2016-11-03" "2016-11-08" "2016-11-09"
## [61] "2016-11-13" "2016-11-14" "2016-12-22" "2016-12-28" "2017-01-05"
## [66] "2017-02-07" "2017-04-03" "2017-04-04" "2017-04-05" "2017-04-10"
## [71] "2017-04-11" "2017-04-13" "2017-04-24" "2017-04-25" "2017-05-31"
## [76] "2017-06-06" "2017-06-07" "2017-06-08" "2017-06-29" "2017-07-11"
## [81] "2017-08-28" "2017-08-31" "2017-09-07" "2017-10-18" "2017-10-23"
```

```
## [86] "2017-10-24" "2017-10-27" "2017-11-24" "2017-10-26" "2017-10-20"
## [91] "2017-12-21" "2018-01-18" "2018-02-20" "2018-04-11" "2018-04-13"
## [96] "2018-04-16" "2018-04-19" "2018-04-20" "2018-04-23" "2018-04-30"
```

Set the dates like this:

```
snapshotDate(ah) <- pd[1]
```

Session info

```
sessionInfo()
```

```
## R version 3.5.0 RC (2018-04-16 r74624)
## Platform: x86_64-apple-darwin15.6.0 (64-bit)
## Running under: macOS Sierra 10.12.6
##
## Matrix products: default
## BLAS: /Library/Frameworks/R.framework/Versions/3.5/Resources/lib/libRblas.0.dylib
## LAPACK: /Library/Frameworks/R.framework/Versions/3.5/Resources/lib/libRlapack.dylib
##
## locale:
## [1] en_US.UTF-8/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8
##
## attached base packages:
## [1] stats4      parallel  stats      graphics  grDevices  utils      datasets
## [8] methods     base
##
## other attached packages:
## [1] BSgenome.Hsapiens.UCSC.hg19_1.4.0 BSgenome_1.47.5
## [3] Biostrings_2.47.12                  XVector_0.19.9
## [5] AnnotationDbi_1.41.6                 Biobase_2.39.2
## [7] rtracklayer_1.39.13                 GenomicRanges_1.31.23
## [9] GenomeInfoDb_1.15.5                 IRanges_2.13.29
## [11] S4Vectors_0.17.43                   AnnotationHub_2.12.0
## [13] BiocGenerics_0.25.3                 BiocStyle_2.7.9
##
## loaded via a namespace (and not attached):
## [1] Rcpp_0.12.16                        compiler_3.5.0
## [3] BiocInstaller_1.30.0                later_0.7.2
## [5] zlibbioc_1.25.0                     bitops_1.0-6
## [7] tools_3.5.0                         digest_0.6.15
## [9] bit_1.1-12                          lattice_0.20-35
## [11] RSQLite_2.1.0                       evaluate_0.10.1
## [13] memoise_1.1.0                       pkgconfig_2.0.1
## [15] Matrix_1.2-14                       DelayedArray_0.5.35
## [17] shiny_1.0.5                         DBI_0.8
## [19] curl_3.2                            yaml_2.1.19
## [21] GenomeInfoDbData_1.1.0              httr_1.3.1
## [23] stringr_1.3.1                       knitr_1.20
## [25] grid_3.5.0                          rprojroot_1.3-2
## [27] bit64_0.9-7                         R6_2.2.2
## [29] BiocParallel_1.13.3                 XML_3.98-1.11
## [31] rmarkdown_1.9                      blob_1.1.1
```

```
## [33] magrittr_1.5                matrixStats_0.53.1
## [35] GenomicAlignments_1.15.14   Rsamtools_1.31.3
## [37] backports_1.1.2             promises_1.0.1
## [39] htmltools_0.3.6             SummarizedExperiment_1.9.18
## [41] mime_0.5                     interactiveDisplayBase_1.18.0
## [43] xtable_1.8-2                 httpuv_1.4.3
## [45] stringi_1.2.2                RCurl_1.95-4.10
```