

Package ‘annaffy’

April 15, 2020

Version 1.58.0

Date 2010-12-29

Title Annotation tools for Affymetrix biological metadata

Author Colin A. Smith <colin@colinsmith.org>

Maintainer Colin A. Smith <colin@colinsmith.org>

Depends R (>= 2.5.0), methods, Biobase, GO.db, KEGG.db

Imports AnnotationDbi (>= 0.1.15), DBI

Suggests hgu95av2.db, multtest, tcltk

Description Functions for handling data from Bioconductor Affymetrix annotation data packages. Produces compact HTML and text reports including experimental data and URL links to many online databases. Allows searching biological metadata using various criteria.

License LGPL

LazyLoad yes

biocViews OneChannel, Microarray, Annotation, GO, Pathways, ReportWriting

git_url <https://git.bioconductor.org/packages/annaffy>

git_branch RELEASE_3_10

git_last_commit 5617008

git_last_commit_date 2019-10-29

Date/Publication 2020-04-14

R topics documented:

aaf.handler	2
aafChromLoc	3
aafChromLoc-class	4
aafChromosome	5
aafChromosome-class	5
aafCytoband	6
aafCytoband-class	7
aafDescription	8
aafDescription-class	8

aafExpr	9
aafGenBank	9
aafGenBank-class	10
aafGO	11
aafGO-class	12
aafGOItem-class	13
aafIntensity-class	13
aafList-class	14
aafLocusLink	15
aafLocusLink-class	16
aafPathway	16
aafPathway-class	17
aafPathwayItem-class	18
aafProbe	19
aafProbe-class	19
aafPubMed	20
aafPubMed-class	21
aafSearchGO	21
aafSearchText	22
aafSigned-class	23
aafSymbol	24
aafSymbol-class	24
aafTable	25
aafTable-class	26
aafTableAnn	27
aafTableFrame	28
aafTableInt	28
aafUniGene	29
aafUniGene-class	30
getCSS-methods	30
getHTML-methods	31
getTD-methods	31
getText-methods	32
getURL-methods	32
is.annpkg	33
selectorWidget	33

Index 35

aaf.handler	<i>Handle fetching annotation data columns</i>
-------------	--

Description

Dispatches requests for annotation data to the correct function. Alternatively returns a list of all the columns it supports.

Usage

```
aaf.handler(probeids, chip, name)
```

Arguments

probeids	character vector containing probe ids
chip	name of chip
name	name of the column of data to return

Value

An aafList containing objects of the proper class.

If no arguments are passed, it will return a character vector of the columns currently supported.

Note

Written at the NASA Center for Computational Astrobiology
<http://cca.arc.nasa.gov/>

Author(s)

Colin A. Smith, <annaffy@colinsmith.org>

aafChromLoc

Constructor for aafChromLoc objects

Description

For the given probeids, constructs an aafList of aafChromLoc objects containing annotation data from the chip data package.

Usage

```
aafChromLoc(probeids, chip)
```

Arguments

probeids	character vector containing probe ids
chip	name of the chip data package

Value

An aafList of aafChromLoc objects. NA values are returned as empty objects.

Author(s)

Colin A. Smith, <annaffy@colinsmith.org>

See Also

[aafChromLoc-class](#)

Examples

```
if (require(hgu95av2.db)) {  
  data(aafExpr)  
  probes <- featureNames(aafExpr)  
  locations <- aafChromLoc(probes, "hgu95av2.db")  
  show(locations[6:10])  
}
```

aafChromLoc-class *Class aafChromLoc, a class for gene chromosome locations*

Description

An abstraction for gene chromosome locations from Bioconductor data packages.

Objects from the Class

Objects are generally created by the [aafChromLoc](#) constructor. Objects can also be created manually by calls of the form `new("aafChromLoc", description)`.

Slots

.Data: Object of class integer

Extends

Class integer, from data part.

Methods

No methods defined with class "aafChromLoc" in the signature. See generic implementations of [getText](#), [getURL](#), [getHTML](#), [getTD](#), and [getCSS](#).

Author(s)

Colin A. Smith, <annaffy@colinsmith.org>

See Also

[aafChromLoc](#)

aafChromosome	<i>Constructor for aafChromosome objects</i>
---------------	--

Description

For the given probeids, constructs an aafList of aafChromosome objects containing annotation data from the chip data package.

Usage

```
aafChromosome(probeids, chip)
```

Arguments

probeids	character vector containing probe ids
chip	name of the chip data package

Value

An aafList of aafChromosome objects. NA values are returned as empty objects.

Author(s)

Colin A. Smith, <annaffy@colinsmith.org>

See Also

[aafChromosome-class](#)

Examples

```
if (require(hgu95av2.db)) {  
  data(aafExpr)  
  probes <- featureNames(aafExpr)  
  chromosomes <- aafChromosome(probes, "hgu95av2.db")  
  show(chromosomes[6:10])  
}
```

aafChromosome-class	<i>Class aafChromosome, a class for gene chromosome assignments</i>
---------------------	---

Description

An abstraction for gene gene chromosome assignments from Bioconductor data packages.

Objects from the Class

Objects are generally created by the [aafChromosome](#) constructor. Objects can also be created manually by calls of the form `new("aafChromosome", description)`.

Slots

.Data: Object of class character

Extends

Class character, from data part.

Methods

No methods defined with class "aafChromosome" in the signature. See generic implementations of [getText](#), [getURL](#), [getHTML](#), [getTD](#), and [getCSS](#).

Author(s)

Colin A. Smith, <annaffy@colinsmith.org>

See Also

[aafChromosome](#)

aafCytoband

Constructor for aafCytoband objects

Description

For the given probeids, constructs an aafList of aafCytoband objects containing annotation data from the chip data package.

Usage

```
aafCytoband(probeids, chip)
```

Arguments

probeids	character vector containing probe ids
chip	name of the chip data package

Value

An aafList of aafCytoband objects. NA values are returned as empty objects.

Author(s)

Colin A. Smith, <annaffy@colinsmith.org>

See Also

[aafCytoband-class](#)

Examples

```
if (require(hgu95av2.db)) {  
  data(aafExpr)  
  probes <- featureNames(aafExpr)  
  bands <- aafCytoband(probes, "hgu95av2.db")  
  show(bands[6:10])  
}
```

aafCytoband-class	<i>Class aafCytoband, a class for cytoband data</i>
-------------------	---

Description

An abstraction for cytoband data from Bioconductor data packages.

Objects from the Class

Objects are generally created by the [aafCytoband](#) constructor. Objects can also be created manually by calls of the form `new("aafCytoband", band, genbank)`.

Slots

band: Object of class character containing genomic cytoband

gene: Object of class character containing containing Gene ID

Methods

getText (aafCytoband): Returns text of band.

getURL (aafCytoband): Returns a URL corresponding entry in NCBI's cytoband map viewer.

See generic implementations of [getHTML](#), [getTD](#), and [getCSS](#).

Author(s)

Colin A. Smith, <annaffy@colinsmith.org>

See Also

[aafCytoband](#)

aafDescription *Constructor for aafDescription objects*

Description

For the given probeids, constructs an aafList of aafDescription objects containing annotation data from the chip data package.

Usage

```
aafDescription(probeids, chip)
```

Arguments

probeids character vector containing probe ids
chip name of the chip data package

Value

An aafList of aafDescription objects. NA values are returned as empty objects.

Author(s)

Colin A. Smith, <annaffy@colinsmith.org>

See Also

[aafDescription-class](#)

Examples

```
if (require(hgu95av2.db)) {  
  data(aafExpr)  
  probes <- featureNames(aafExpr)  
  descriptions <- aafDescription(probes, "hgu95av2.db")  
  show(descriptions[6:10])  
}
```

aafDescription-class *Class aafDescription, a class for gene descriptions*

Description

An abstraction for gene description from Bioconductor data packages.

Objects from the Class

Objects are generally created by the [aafDescription](#) constructor. Objects can also be created manually by calls of the form `new("aafDescription", description)`.

Slots

.Data: Object of class character

Extends

Class character, from data part.

Methods

No methods defined with class "aafDescription" in the signature. See generic implementations of [getText](#), [getURL](#), [getHTML](#), and [getTD](#).

Author(s)

Colin A. Smith, <annaffy@colinsmith.org>

See Also

[aafDescription](#)

aafExpr

Sample ExpressionSet used for demonstration purposes

Description

Contains expression values for 250 probe ids with 8 samples. Two covariates are provided. Expression comes from the hgu95av2 chip.

Details

The data is real but anonymized. 250 genes expression values were chosen at random from an existing ExpressionSet. Another 250 probe ids were selected at random and were assigned to the expression values. That way, expression values do not correspond to the true probe ids.

Post-processing was done with `rma()` in `affy` 1.2.23.

aafGenBank

Constructor for aafGenBank objects

Description

For the given probeids, constructs an `aafList` of `aafGenBank` objects containing annotation data from the chip data package.

Usage

`aafGenBank(probeids, chip)`

Arguments

probeids character vector containing probe ids
 chip name of the chip data package

Value

An aafList of aafGenBank objects. NA values are returned as empty objects.

Author(s)

Colin A. Smith, <annaffy@colinsmith.org>

See Also

[aafGenBank-class](#)

Examples

```
if (require(hgu95av2.db)) {
  data(aafExpr)
  probes <- featureNames(aafExpr)
  gbs <- aafGenBank(probes, "hgu95av2.db")
  show(gbs[6:10])
}
```

aafGenBank-class

Class aafGenBank, a class for GenBank accession numbers

Description

An abstraction for GenBank accession numbers from Bioconductor data packages.

Objects from the Class

Objects are generally created by the [aafGenBank](#) constructor. Objects can also be created manually by calls of the form `new("aafGenBank", accnum)`.

Slots

.Data: Object of class character

Extends

Class character, from data part.

Methods

getURL (aafGenBank): Returns a URL to the corresponding entry in NCBI's GenBank database. See generic implementations of [getText](#), [getHTML](#), and [getTD](#).

Author(s)

Colin A. Smith, <annaffy@colinsmith.org>

See Also[aafGenBank](#)

aafGO	<i>Constructor for aafGO objects</i>
-------	--------------------------------------

Description

For the given probeids, constructs an aafList of aafGO objects containing annotation data from the chip data package.

Usage

```
aafGO(probeids, chip)
```

Arguments

probeids	character vector containing probe ids
chip	name of the chip data package

Value

An aafList of aafGO objects. NA values are returned as empty objects.

Author(s)

Colin A. Smith, <annaffy@colinsmith.org>

See Also[aafGO-class](#)**Examples**

```
if (require(hgu95av2.db)) {  
  data(aafExpr)  
  probes <- featureNames(aafExpr)  
  gos <- aafGO(probes, "hgu95av2.db")  
  show(gos[6:10])  
}
```

aafGO-class

Class aafGO, a class for gene ontology ids

Description

An abstraction for gene ontology ids from Bioconductor data packages. This class is actually extends `aafList` and holds `aafGOItem` objects which have the actual annotation data.

Objects from the Class

Objects are generally created by the `aafGO` constructor. Objects can also be created manually by calls of the form `new("aafGO", list(goitems))`.

Slots

`.Data`: Object of class `list`

Extends

Class `aafList`, from data part.

Methods

getText (aafGO): Returns a comma delimited list of the individual `aafGOItem` objects.

getURL (aafGO): Returns a single URL to an AmiGO page which displays all the gene ontology identifiers in an hierarchical listing.

getHTML (aafGO): Returns an HTML representation of each of the individual `aafGOItem` objects, concatenated together.

getTD (aafGO): Returns an HTML table cell representation with the class set to "aafGO".

getCSS (aafGOItem): Returns a line of CSS that indents GOItem paragraphs.

Author(s)

Colin A. Smith, <annaffy@colinsmith.org>

See Also

[aafGO](#), [aafGOItem](#), [aafList](#)

aafGOItem-class	<i>Class aafGOItem, a class for gene ontology id elements</i>
-----------------	---

Description

An abstraction for gene ontology id elements from Bioconductor data packages. Multiple instances of this class are held by the wrapper class aafGO.

Objects from the Class

Objects are generally created by the [aafGO](#) constructor. Objects can also be created manually by calls of the form `new("aafGOItem", id, name, type)`.

Slots

id: Object of class character containing GO id
name: Object of class character containing textual name
type: Object of class character containing GO subtype
evid: Object of class character containing GO evidence code

Methods

getText (aafGOItem): Returns textual representation formatted "id: name".
getURL (aafGOItem): Returns a URL to the corresponding gene ontology entry on AmiGO.
getHTML (aafGOItem): Returns an HTML representation including the URL link, gene ontology name, and rollover subtype.

Author(s)

Colin A. Smith, <annaffy@colinsmith.org>

See Also

[aafGO-class](#), [aafGO](#)

aafIntensity-class	<i>Class aafIntensity, a class for gene expression values</i>
--------------------	---

Description

A class for displaying gene expression values with a green background of differing intensities.

Objects from the Class

Objects are generally created by the [aafTableInt](#) constructor. Objects can also be created manually by calls of the form `new("aafIntensity", intensity)`.

Slots

.Data: Object of class `numeric`

Extends

Class `numeric`, from data part.

Methods

getTD (`aafIntensity`): Returns an HTML table cell with background varying from white to green depending on intensity. Scaling is controlled by two options, `minIntensity` (fully white) and `maxIntensity` (fully green), usually set by `writeHTML`.

See generic implementations of [getText](#), [getURL](#), [getHTML](#), and [getCSS](#).

Author(s)

Colin A. Smith, <annaffy@colinasmith.org>

See Also

[aafTableInt](#)

aafList-class

Class aafList, a specialized subclass of list

Description

A class for lists of annotation data objects.

Objects from the Class

Objects are generally created by any of the annotation data constructors that are also part of this package. Objects can also be created manually by calls of the form `new("aafList", list)`.

Slots

.Data: Object of class `list`

Extends

Class `list`, from data part.

Methods

getText (`aafList`): Returns a character vector containing textual representations of every item.

getURL (`aafList`): Returns a character vector containing single URLs (if possible) of every item.

getHTML (`aafList`): Returns a character vector containing HTML representations of every item.

getTD (`aafList`): Returns a character vector containing HTML table cell representations of every item.

getCSS (`aafList`): Returns `getCSS()` of the first item in the list.

[(`aafList`): Returns a subset of `aafList` as another `aafList` object.

Author(s)

Colin A. Smith, <annaffy@colinsmith.org>

aafLocusLink *Constructor for aafLocusLink objects*

Description

For the given probeids, constructs an aafList of aafLocusLink objects containing annotation data from the chip data package.

Usage

```
aafLocusLink(probeids, chip)
```

Arguments

probeids	character vector containing probe ids
chip	name of the chip data package

Value

An aafList of aafLocusLink objects. NA values are returned as empty objects.

Author(s)

Colin A. Smith, <annaffy@colinsmith.org>

See Also

[aafLocusLink-class](#)

Examples

```
if (require(hgu95av2.db)) {  
  data(aafExpr)  
  probes <- featureNames(aafExpr)  
  lls <- aafLocusLink(probes, "hgu95av2.db")  
  show(lls[6:10])  
}
```

aafLocusLink-class *Class aafLocusLink, a class for LocusLink ids*

Description

An abstraction for LocusLink ids from Bioconductor data packages.

Objects from the Class

Objects are generally created by the [aafLocusLink](#) constructor. Objects can also be created manually by calls of the form `new("aafLocusLink", id)`.

Slots

.Data: Object of class integer

Extends

Class integer, from data part.

Methods

getURL (aafLocusLink): Returns a URL to the corresponding entry in NCBI's LocusLink database. On the rare chance that more than one id is defined, more than one URL will be returned.

See generic implementations of [getText](#), [getHTML](#), and [getTD](#).

Author(s)

Colin A. Smith, <annaffy@colinsmith.org>

See Also

[aafLocusLink](#)

aafPathway *Constructor for aafPathway objects*

Description

For the given probeids, constructs an aafList of aafPathway objects containing annotation data from the chip data package.

Usage

```
aafPathway(probeids, chip)
```

Arguments

probeids	character vector containing probe ids
chip	name of the chip data package

Value

An aafList of aafPathway objects. NA values are returned as empty objects.

Author(s)

Colin A. Smith, <annaffy@colinsmith.org>

See Also

[aafPathway-class](#)

Examples

```
if (require(hgu95av2.db)) {
  data(aafExpr)
  probes <- featureNames(aafExpr)
  pathways <- aafPathway(probes, "hgu95av2.db")
  show(pathways[6:10])
}
```

aafPathway-class

Class aafPathway, a class for KEGG pathway ids

Description

An abstraction for KEGG pathway ids from Bioconductor data packages. This class is actually extends aafList and holds aafPathwayItem objects which have the actual annotation data.

Objects from the Class

Objects are generally created by the [aafPathway](#) constructor. Objects can also be created manually by calls of the form `new("aafPathway", list(pathwayItems))`.

Slots

.Data: Object of class list

Extends

Class aafList, from data part.

Methods

getText (aafGO): Returns a comma delimited list of the individual aafPathwayItem objects.

getURL (aafGO): Returns zero length character vector because this method is not valid for this class.

getHTML (aafGO): Returns an HTML representation of each of the individual aafPathwayItem objects, concatenated together.

getTD (aafGO): Returns an HTML table cell representation with the class set to "aafPathway".

getCSS (aafGO): Returns a line of CSS which intends PathwayItem paragraphs.

Author(s)

Colin A. Smith, <annaffy@colinsmith.org>

See Also

[aafPathway](#), [aafPathwayItem](#), [aafList](#)

aafPathwayItem-class *Class aafPathwayItem, a class for KEGG pathway id elements*

Description

An abstraction for KEGG pathway id elements from Bioconductor data packages. Multiple instances of this class are held by the wrapper class [aafPathway](#).

Objects from the Class

Objects are generally created by the [aafPathway](#) constructor. Objects can also be created manually by calls of the form `new("aafPathwayItem", id, name, enzyme)`.

Slots

id: Object of class character containing KEGG pathway id

name: Object of class character containing textual name

enzyme: Object of class character containing the Enzyme Commission number if applicable

Methods

getText (aafPathwayItem): Returns textual representation formatted "id: name".

getURL (aafPathwayItem): Returns a URL to the corresponding entry in the Kyoto Encyclopedia of Genes and Genomes database. If there is a corresponding EC number, it will be highlighted in red.

getHTML (aafPathwayItem): Returns an HTML representation including the URL link and pathway name.

Author(s)

Colin A. Smith, <annaffy@colinsmith.org>

See Also

[aafPathway-class](#), [aafPathway](#)

aafProbe	<i>Constructor for aafProbe objects</i>
----------	---

Description

For the given probeids, constructs an aafList of aafProbe objects.

Usage

```
aafProbe(probeids)
```

Arguments

probeids character vector containing probe ids

Value

An aafList of aafProbe objects.

Author(s)

Colin A. Smith, <annaffy@colinsmith.org>

See Also

[aafProbe-class](#)

Examples

```
if (require(hgu95av2.db)) {
  data(aafExpr)
  probes <- featureNames(aafExpr)
  probesets <- aafProbe(probes)
  getURL(probesets[6:10])
}
```

aafProbe-class	<i>Class aafProbe, a class for Probe ids</i>
----------------	--

Description

An abstraction for Affymetrix ProbeSet ids.

Objects from the Class

Objects are generally created by the [aafProbe](#) constructor. Objects can also be created manually by calls of the form `new("aafProbe", id)`.

Slots

.Data: Object of class character

Extends

Class character, from data part.

Methods

getURL (aafProbe): Returns a URL to the annotation found in the Affymetrix NetAffx Analysis Center.

See generic implementations of [getText](#), [getHTML](#), and [getTD](#).

Author(s)

Colin A. Smith, <annaffy@colinsmith.org>

See Also

[aafProbe](#)

aafPubMed	<i>Constructor for aafPubMed objects</i>
-----------	--

Description

For the given probeids, constructs a list of aafPubMed objects containing annotation data from the chip data package.

Usage

```
aafPubMed(probeids, chip)
```

Arguments

probeids	character vector containing probe ids
chip	name of the chip data package

Value

An aafList of aafPubMed objects. NA values are returned as empty objects.

Author(s)

Colin A. Smith, <annaffy@colinsmith.org>

See Also

[aafPubMed-class](#)

Examples

```
if (require(hgu95av2.db)) {
  data(aafExpr)
  probes <- featureNames(aafExpr)
  pmids <- aafPubMed(probes, "hgu95av2.db")
  show(pmids[6:10])
}
```

aafPubMed-class	<i>Class aafPubMed, a class for PubMed ids</i>
-----------------	--

Description

An abstraction for LocusLink ids from Bioconductor data packages.

Objects from the Class

Objects are generally created by the [aafPubMed](#) constructor. Objects can also be created manually by calls of the form `new("aafPubMed", id)`.

Slots

.Data: Object of class integer

Extends

Class integer, from data part.

Methods

getURL (aafPubMed): Returns a single URL to the corresponding abstracts in NCBI's PubMed database.

getHTML (aafPubMed): Returns an HTML link along with the number of abstracts.

getTD (aafPubMed): Returns an HTML table cell representation with the class set to "aafPubMed".

getCSS (aafPubMed): Returns a line of CSS which centers the PubMed link.

Author(s)

Colin A. Smith, <annaffy@colinsmith.org>

See Also

[aafPubMed](#)

aafSearchGO	<i>Find probe ids corresponding to GO ids</i>
-------------	---

Description

Searches Gene Ontology ids for corresponding probe ids in a given chip, optionally including descendants.

Usage

```
aafSearchGO(chip, ids, descendants = TRUE, logic = "OR")
```

Arguments

chip	name of the chip data package
ids	numeric or character vector of GO ids
descendents	logical, include GO descendents?
logic	type of logic to use, "AND" or "OR"

Value

A character vector of probe ids matching the search criteria.

Author(s)

Colin A. Smith, <annaffy@colinsmith.org>

See Also

[aafSearchText](#)

aafSearchText	<i>Search metadata annotation text</i>
---------------	--

Description

Searches Bioconductor metadata annotation package text for specific strings or Perl compatible regular expressions.

Usage

```
aafSearchText(chip, colnames, text, logic = "OR")
```

Arguments

chip	name of the chip data package
colnames	character vector of metadata column names to search
text	character vector of strings/regular expressions to match
logic	type of logic to use, "AND" or "OR"

Value

A character vector of probe ids matching the search criteria.

Author(s)

Colin A. Smith, <annaffy@colinsmith.org>

See Also

[aafSearchGO](#)

Examples

```
if (require(hgu95av2.db)) {  
  aafSearchText("hgu95av2.db", "Description", c("kinase", "interferon"))  
  # aafSearchText("hgu95av2.db", c("Gene Ontology", "Pathway"), "ribosome")  
}
```

aafSigned-class	<i>Class aafSigned, a class for signed numerical data</i>
-----------------	---

Description

A class for displaying signed numerical data with different styles depending on the sign.

Objects from the Class

Objects are generally created by the [aafTable](#) constructor. Objects can also be created manually by calls of the form `new("aafSigned", signedval)`.

Slots

.Data: Object of class `numeric`

Extends

Class `numeric`, from data part.

Methods

getTD (aafSigned): Returns an HTML table cell with class differentially set based on sign. `aafSignedPos` is used for positive values. `aafSignedNeg` is used for negative values. `aafSignedZero` is used for zero values.

getCSS (aafSigned): Returns two lines of CSS that set the cell background of positive values light blue and negative values light red.

See generic implementations of [getText](#), [getURL](#), and [getHTML](#).

Author(s)

Colin A. Smith, <annaffy@colinsmith.org>

See Also

[aafTable](#)

`aafSymbol`*Constructor for aafSymbol objects*

Description

For the given probeids, constructs a list of aafSymbol objects containing annotation data from the chip data package.

Usage

```
aafSymbol(probeids, chip)
```

Arguments

probeids	character vector containing probe ids
chip	name of the chip data package

Value

An aafList of aafSymbol objects. NA values are returned as empty objects.

Author(s)

Colin A. Smith, <annaffy@colinsmith.org>

See Also

[aafSymbol-class](#)

Examples

```
if (require(hgu95av2.db)) {  
  data(aafExpr)  
  probes <- featureNames(aafExpr)  
  symbols <- aafSymbol(probes, "hgu95av2.db")  
  show(symbols[6:10])  
}
```

`aafSymbol-class`*Class aafSymbol, a class for gene symbols*

Description

An abstraction for gene symbol from Bioconductor data packages.

Objects from the Class

Objects are generally created by the [aafSymbol](#) constructor. Objects can also be created manually by calls of the form `new("aafSymbol", description)`.

Slots

.Data: Object of class character with gene symbol

Extends

Class character, from data part.

Methods

No methods defined with class "aafSymbol" in the signature. See generic implementations of [getText](#), [getURL](#), [getHTML](#), and [getTD](#).

Author(s)

Colin A. Smith, <annaffy@colinsmith.org>

See Also

[aafSymbol](#)

aafTable

Constructor for aafTable objects

Description

Constructs an aafTable object given vectors, lists, or aafList objects.

Usage

```
aafTable(..., items = list(...), colnames = names(items),
          probeids = character(0), signed = FALSE)
```

Arguments

...	named arguments, one for each column
items	alternatively a named list of the items to be put in the table
colnames	character vector of column names
probeids	character vector of probe ids associated with each row
signed	boolean, should each column be colored based on the sign?

Value

An aafTable object.

Author(s)

Colin A. Smith, <annaffy@colinsmith.org>

See Also

[aafTable-class](#)

aafTable-class *Class aafTable, a tabular microarray data class*

Description

A class for storing and flexible output of microarray data to HTML and text formats.

Objects from the Class

Objects are generally created by any of the annotation table constructors that are also part of this package. Objects can also be created manually by calls of the form `new("aafList", probeids, table)`.

Slots

probeids: Object of class character containing the probe ids for each row of the table.

table: Object of class list containing aafList objects all of the same length, representing the columns of the table. Each item in the list must have a unique name.

Methods

probeids (aafTable): Returns a character vector containing the probe ids for each row of the table.

probeids<- (aafTable): Sets the probe ids for the table rows. Can be set to `character(0)` if unknown or not applicable.

colnames (aafTable): Returns a character vector containing the names of the columns stored in the table.

colnames<- (aafTable): Set the column names for the table. Each must be unique.

dim (aafTable): Returns the dimensions of the table.

merge (aafTable, aafTable, all = FALSE, all.x = all, all.y = all, suffixes = c(".x", ".y")): Merges two tables together, aligning common probe ids if possible. Duplicate column names are given suffixes to make them unique. Returns the merged table.

rbind (aafTable, aafTable, ...): Vertically combines tables by row. Requires that column names be identical and that all tables either have probe ids defined or not.

[Returns a subset of the table based on [row, column]. Indices may be passed as integers or probe ids/column names.

[[Returns the given table column. This also supports recursive subsetting to address columns, then cells, then sub-cells (if applicable). See [Extract](#) for more information.

\\$ Returns the given table column.

saveHTML (aafTable, filename, title = "Bioconductor Affymetrix Probe Listing", colnames = colnames(aafTable), range = 1:dim(aafTable)[1], open = FALSE, widget = FALSE): Saves the table to HTML with the specified filename and title. Both the columns and the range of table rows can be specified. Range can either be specified as a character vector of probe ids or an integer vector of row positions. One can also specify whether to open the resulting file in the browser and whether to use a widget for column selection.

saveText (aafTable, filename, header = TRUE, colnames = colnames(aafTable), range = 1:dim(aafTable)[1], widget = FALSE): Saves the table to tab delimited text with specified filename and optional header. Both the columns and the range of table rows can be specified. Range can either be specified as a character vector of probe ids or an integer vector of row positions. One can also specify whether to use a widget for column selection.

Author(s)

Colin A. Smith, <annaffy@colinsmith.org>

See Also

[aafTable](#), [aafTableFrame](#), [aafTableAnn](#), [aafTableInt](#)

aafTableAnn

Constructor for aafTable objects from annotation data

Description

Constructs an aafTable object given a set of probe ids and desired annotation types.

Usage

```
aafTableAnn(probeids, chip, colnames = aaf.handler(chip = chip), widget = FALSE)
```

Arguments

probeids	character vector of probe ids
chip	name of the data package in which the annotation data is stored
colnames	character vector of annotation types
widget	boolean, use widget to select columns?

Value

An aafTable object.

Author(s)

Colin A. Smith, <annaffy@colinsmith.org>

See Also

[aafTable-class](#)

aafTableFrame	<i>Constructor for aafTable objects from data frames</i>
---------------	--

Description

Constructs an aafTable object given a data frame.

Usage

```
aafTableFrame(frame, colnames = names(frame), probeids = row.names(frame),
              signed = FALSE)
```

Arguments

frame	data frame to be converted to the table
colnames	character vector of column names
probeids	character vector of probe ids associated with each row
signed	boolean, should each column be colored based on the sign?

Value

An aafTable object.

Author(s)

Colin A. Smith, <annaffy@colinsmith.org>

See Also

[aafTable-class](#)

aafTableInt	<i>Constructor for aafTable objects from ExpressionSets</i>
-------------	---

Description

Constructs an aafTable object containing expression values given an ExpressionSet.

In the resulting HTML table, the expression values will have backgrounds with varying intensities of green depending on the expression measure.

Usage

```
aafTableInt(exprSet, colnames = sampleNames(exprSet),
            probeids = featureNames(exprSet))
```

Arguments

exprSet	object of class ExpressionSet
colnames	character vector of column names
probeids	character vector of probe ids associated with each row

Value

An aafTable object.

Author(s)

Colin A. Smith, <annaffy@colinsmith.org>

See Also

[aafTable-class](#), [aafIntensity](#)

aafUniGene

Constructor for aafUniGene objects

Description

For the given probeids, constructs a list of aafUniGene objects containing annotation data from the chip data package.

Usage

```
aafUniGene(probeids, chip)
```

Arguments

probeids	character vector containing probe ids
chip	name of the chip data package

Value

An aafList of aafUniGene objects. NA values are returned as empty objects.

Author(s)

Colin A. Smith, <annaffy@colinsmith.org>

See Also

[aafUniGene-class](#)

Examples

```
if (require(hgu95av2.db)) {  
  data(aafExpr)  
  probes <- featureNames(aafExpr)  
  ugs <- aafUniGene(probes, "hgu95av2.db")  
  show(ugs[6:10])  
}
```

aafUniGene-class *Class aafUniGene, a class for UniGene cluster ids*

Description

An abstraction for UniGene cluster ids from Bioconductor data packages.

Objects from the Class

Objects are generally created by the [aafUniGene](#) constructor. Objects can also be created manually by calls of the form `new("aafUniGene", id)`.

Slots

.Data: Object of class character

Extends

Class character, from data part.

Methods

getURL (aafUniGene): Returns a URLs to the corresponding entry in NCBI's UniGene database. On the rare chance that more than one id is defined, more than one URL will be returned.

getHTML (aafUniGene): Returns an HTML representation with a link to the UniGene database. On the rare chance that more than one id is defined, more than one link will be returned.

Author(s)

Colin A. Smith, <annaffy@colinsmith.org>

See Also

[aafUniGene](#)

getCSS-methods *Methods for function getCSS*

Description

Methods to get relevant stylesheet lines for an object.

Methods

object = ANY Returns an empty character vector.

Note

For information about other implementations of this method, see documentation of the respective class.

See Also

[aafList-class](#), [aafPubMed-class](#), [aafGO-class](#), [aafPathway-class](#), [aafSigned-class](#)

getHTML-methods

Methods for function getHTML

Description

Methods to get an HTML representation of an object.

Methods

object = ANY Returns text of object along with URL link if applicable. If object is floating point, it displays a fixed number of significant digits as specified by the sigfigs option (default 6).

Note

For information about other implementations of this method, see documentation of the respective class.

See Also

[aafList-class](#), [aafPubMed-class](#), [aafGO-class](#), [aafGOItem-class](#), [aafPathway-class](#), [aafPathwayItem-class](#)

getTD-methods

Methods for function getTD

Description

Methods to get an HTML table cell representation of an object.

Methods

object = ANY Returns <td> tag containing HTML representation of object. Sets class attribute to class(object).

Note

For information about other implementations of this method, see documentation of the respective class.

See Also

[aafList-class](#), [aafGO-class](#), [aafPathway-class](#), [aafIntensity-class](#)

getText-methods *Methods for function getText*

Description

Methods to get a textual representation of an object.

Methods

object = ANY Returns a comma delimited list of the elements in list.

Note

For information about other implementations of this method, see documentation of the respective class.

See Also

[aafList-class](#), [aafCytoband-class](#), [aafGO-class](#), [aafGOItem-class](#), [aafPathway-class](#), [aafPathwayItem-class](#)

getURL-methods *Methods for function getURL*

Description

Methods to get a URL link to a web resource for an object.

Methods

object = ANY Returns an empty character vector.

Note

For information about other implementations of this method, see documentation of the respective class.

See Also

[aafList-class](#), [aafGenBank-class](#), [aafLocusLink-class](#), [aafCytoband-class](#), [aafUniGene-class](#), [aafPubMed-class](#), [aafGO-class](#), [aafGOItem-class](#), [aafPathwayItem-class](#)

is.annpkg *Determine if packages contain annotation*

Description

Checks to see that the given packages contain all the necessary annotation environments to be usable by annaffy.

Usage

```
is.annpkg(packages, lib.loc = NULL)
```

Arguments

packages character vector containing package names to check

lib.loc a character vector with path names of R libraries, or NULL. The default value of NULL corresponds to all libraries currently known. If the default is used, the loaded packages are searched before the libraries.

Value

A logical vector indicating whether the packages contain annotation data.

Author(s)

Colin A. Smith, <annaffy@colinsmith.org>

Examples

```
pkgnames <- installed.packages()[,"Package"]
pkgnames <- pkgnames[1:5]
pkgnames[is.annpkg(pkgnames)]
```

selectorWidget *Dialog to select items from a list*

Description

Presents the user with a dialog box to select items from a list.

Usage

```
selectorWidget(options, selected = character(0), title = "Selector",
               ordersel = FALSE, ordernsel = FALSE,
               height = max(10,min(20,length(options))))
```

Arguments

options	vector, options to be selected from
selected	vector, subset of options selected by default
title	character scalar, window title
ordersel	boolean, keep the selected items in order?
ordernsel	boolean, keep the not selected items in order?
height	scalar, height of the two listboxes

Value

A character vector containing the selected items. If a vector of a different class was initially provided, it must be manually coerced back to the correct type.

Author(s)

Colin A. Smith, <annaffy@colinsmith.org>

Index

*Topic **classes**

- aafChromLoc-class, 4
- aafChromosome-class, 5
- aafCytoband-class, 7
- aafDescription-class, 8
- aafGenBank-class, 10
- aafGO-class, 12
- aafGOItem-class, 13
- aafIntensity-class, 13
- aafList-class, 14
- aafLocusLink-class, 16
- aafPathway-class, 17
- aafPathwayItem-class, 18
- aafProbe-class, 19
- aafPubMed-class, 21
- aafSigned-class, 23
- aafSymbol-class, 24
- aafTable-class, 26
- aafUniGene-class, 30

*Topic **datasets**

- aafExpr, 9

*Topic **file**

- aaf.handler, 2
- aafChromLoc, 3
- aafChromosome, 5
- aafCytoband, 6
- aafDescription, 8
- aafGenBank, 9
- aafGO, 11
- aafLocusLink, 15
- aafPathway, 16
- aafProbe, 19
- aafPubMed, 20
- aafSearchGO, 21
- aafSearchText, 22
- aafSymbol, 24
- aafTable, 25
- aafTableAnn, 27
- aafTableFrame, 28
- aafTableInt, 28
- aafUniGene, 29
- is.annpkg, 33
- selectorWidget, 33

*Topic **methods**

- getCSS-methods, 30
- getHTML-methods, 31
- getTD-methods, 31
- getText-methods, 32
- getURL-methods, 32
- [, aafList-method (aafList-class), 14
- [, aafTable-method (aafTable-class), 26
- [[, aafTable-method (aafTable-class), 26
- \$.aafTable (aafTable-class), 26

- aaf.handler, 2
- aafChromLoc, 3, 4
- aafChromLoc-class, 3, 4
- aafChromosome, 5, 5, 6
- aafChromosome-class, 5, 5
- aafCytoband, 6, 7
- aafCytoband-class, 6, 7, 32
- aafDescription, 8, 8, 9
- aafDescription-class, 8, 8
- aafExpr, 9
- aafGenBank, 9, 10, 11
- aafGenBank-class, 10, 10, 32
- aafGO, 11, 12, 13
- aafGO-class, 11, 12, 13, 31, 32
- aafGOItem, 12
- aafGOItem (aafGOItem-class), 13
- aafGOItem-class, 13, 31, 32
- aafIntensity, 29
- aafIntensity (aafIntensity-class), 13
- aafIntensity-class, 13, 31
- aafList, 12, 18
- aafList (aafList-class), 14
- aafList-class, 14, 31, 32
- aafLocusLink, 15, 16
- aafLocusLink-class, 15, 16, 32
- aafPathway, 16, 17, 18
- aafPathway-class, 17, 17, 18, 31, 32
- aafPathwayItem, 18
- aafPathwayItem (aafPathwayItem-class), 18
- aafPathwayItem-class, 18, 31, 32
- aafProbe, 19, 19, 20
- aafProbe-class, 19, 19

- aafPubMed, [20, 21](#)
- aafPubMed-class, [20, 21, 31, 32](#)
- aafSearchGO, [21, 22](#)
- aafSearchText, [22, 22](#)
- aafSigned (aafSigned-class), [23](#)
- aafSigned-class, [23, 31](#)
- aafSymbol, [24, 24, 25](#)
- aafSymbol-class, [24, 24](#)
- aafTable, [23, 25, 27](#)
- aafTable-class, [25, 26, 27–29](#)
- aafTableAnn, [27, 27](#)
- aafTableFrame, [27, 28](#)
- aafTableInt, [13, 14, 27, 28](#)
- aafUniGene, [29, 30](#)
- aafUniGene-class, [29, 30, 32](#)

- colnames (aafTable-class), [26](#)
- colnames, aafTable-method
(aafTable-class), [26](#)
- colnames<- (aafTable-class), [26](#)
- colnames<- , aafTable-method
(aafTable-class), [26](#)

- dim.aafTable (aafTable-class), [26](#)

- Extract, [26](#)

- getCSS, [4, 6, 7, 14](#)
- getCSS (getCSS-methods), [30](#)
- getCSS, aafGO-method (aafGO-class), [12](#)
- getCSS, aafList-method (aafList-class),
[14](#)
- getCSS, aafPathway-method
(aafPathway-class), [17](#)
- getCSS, aafPubMed-method
(aafPubMed-class), [21](#)
- getCSS, aafSigned-method
(aafSigned-class), [23](#)
- getCSS, ANY-method (getCSS-methods), [30](#)
- getCSS-methods, [30](#)
- getHTML, [4, 6, 7, 9, 10, 14, 16, 20, 23, 25](#)
- getHTML (getHTML-methods), [31](#)
- getHTML, aafGO-method (aafGO-class), [12](#)
- getHTML, aafGOItem-method
(aafGOItem-class), [13](#)
- getHTML, aafList-method (aafList-class),
[14](#)
- getHTML, aafPathway-method
(aafPathway-class), [17](#)
- getHTML, aafPathwayItem-method
(aafPathwayItem-class), [18](#)
- getHTML, aafPubMed-method
(aafPubMed-class), [21](#)
- getHTML, aafUniGene-method
(aafUniGene-class), [30](#)
- getHTML, ANY-method (getHTML-methods), [31](#)
- getHTML-methods, [31](#)
- getTD, [4, 6, 7, 9, 10, 16, 20, 25](#)
- getTD (getTD-methods), [31](#)
- getTD, aafGO-method (aafGO-class), [12](#)
- getTD, aafIntensity-method
(aafIntensity-class), [13](#)
- getTD, aafList-method (aafList-class), [14](#)
- getTD, aafPathway-method
(aafPathway-class), [17](#)
- getTD, aafPubMed-method
(aafPubMed-class), [21](#)
- getTD, aafSigned-method
(aafSigned-class), [23](#)
- getTD, ANY-method (getTD-methods), [31](#)
- getTD-methods, [31](#)
- getText, [4, 6, 9, 10, 14, 16, 20, 23, 25](#)
- getText (getText-methods), [32](#)
- getText, aafCytoband-method
(aafCytoband-class), [7](#)
- getText, aafGO-method (aafGO-class), [12](#)
- getText, aafGOItem-method
(aafGOItem-class), [13](#)
- getText, aafList-method (aafList-class),
[14](#)
- getText, aafPathway-method
(aafPathway-class), [17](#)
- getText, aafPathwayItem-method
(aafPathwayItem-class), [18](#)
- getText, ANY-method (getText-methods), [32](#)
- getText-methods, [32](#)
- getURL, [4, 6, 9, 14, 23, 25](#)
- getURL (getURL-methods), [32](#)
- getURL, aafCytoband-method
(aafCytoband-class), [7](#)
- getURL, aafGenBank-method
(aafGenBank-class), [10](#)
- getURL, aafGO-method (aafGO-class), [12](#)
- getURL, aafGOItem-method
(aafGOItem-class), [13](#)
- getURL, aafList-method (aafList-class),
[14](#)
- getURL, aafLocusLink-method
(aafLocusLink-class), [16](#)
- getURL, aafPathway-method
(aafPathway-class), [17](#)
- getURL, aafPathwayItem-method
(aafPathwayItem-class), [18](#)
- getURL, aafProbe-method
(aafProbe-class), [19](#)

getUrl, aafPubMed-method
 (aafPubMed-class), 21

getUrl, aafUniGene-method
 (aafUniGene-class), 30

getUrl, ANY-method (getUrl-methods), 32

getUrl-methods, 32

is.annpkg, 33

merge.aafTable (aafTable-class), 26

probeids (aafTable-class), 26

probeids, aafTable-method
 (aafTable-class), 26

probeids<- (aafTable-class), 26

probeids<- , aafTable-method
 (aafTable-class), 26

rbind.aafTable (aafTable-class), 26

saveHTML (aafTable-class), 26

saveHTML, aafTable-method
 (aafTable-class), 26

saveText (aafTable-class), 26

saveText, aafTable-method
 (aafTable-class), 26

selectorWidget, 33

show, aafCytoband-method
 (aafCytoband-class), 7

show, aafGOItem-method
 (aafGOItem-class), 13

show, aafList-method (aafList-class), 14

show, aafPathwayItem-method
 (aafPathwayItem-class), 18