Package ‘OrganismDbi’

March 26, 2013

Title Software to enable the smooth interfacing of different database packages.

Description The package enables a simple unified interface to several annotation packages each of which has its own schema by taking advantage of the fact that each of these packages implements a select methods.

Version 1.0.3

Author Marc Carlson, Herve Pages, Martin Morgan, Valerie Obenchain

Maintainer Biocore Data Team <maintainer@bioconductor.org>

Depends R (>= 2.14.0), methods, AnnotationDbi (>= 1.16.10),GenomicFeatures

Imports BiocGenerics, graph, RBGL, AnnotationDbi

Suggests Homo.sapiens, Rattus.norvegicus, RUnit

Collate AllClasses.R select.R createOrganismPackage.R test_OrganismDbi_package.R

License Artistic-2.0

biocViews AnnotationData, Infrastructure

R topics documented:

makeOrganismPackage 1
OrganismDb-class 3

Index 5

---

makeOrganismPackage *Making OrganismDb packages from annotation packages.*

Description

makeOrganismPackage is a method that generates a package that will load an appropriate annotationOrganismDb object that will in turn point to existing annotation packages.

Usage

```
makeOrganismPackage (pkgname,
                     graphData,
                     organism,
                     version,
                     maintainer,
                     author,
                     destDir,
                     license="Artistic-2.0")
```

Arguments

pkgname	What is the desired package name. Traditionally, this should be the genus and species separated by a ".". So as an example, "Homo.sapiens" would be the package name for human
graphData	A list of short character vectors. Each character vector in the list is exactly two elements long and represents a join relationship between two packages. The names of these character vectors are the package names and the values are the foreign keys that should be used to connect each package. All foreign keys must be values that can be returned by the cols method for each package in question, and obviously they also must be the same kind of identifier as well.
organism	The name of the organism this package represents
version	What is the version number for this package?
maintainer	Who is the package maintainer? (must include email to be valid)
author	Who is the creator of this package?
destDir	A path where the package source should be assembled.
license	What is the license (and it's version)

Details

The purpose of this method is to create a special package that will depend on existing annotation packages and which will load a special `annotationOrganismDb` object that will allow proper dispatch of special select methods. These methods will allow the user to easily query across multiple annotation resources via information contained by the `annotationOrganismDb` object. Because the end result will be a package that treats all the data mapped together as a single source, the user is encouraged to take extra care to ensure that the different packages used are from the same build etc.

Value

A special package to load an OrganismDb object.

Author(s)

M. Carlson

See Also

OrganismDb

Examples

```
## set up the list with the relevant relationships:
gd <- list(join1 = c(GO.db="GOID", org.Hs.eg.db="GO"),
           join2 = c(org.Hs.eg.db="ENTREZID",
                     TxDb.Hsapiens.UCSC.hg19.knownGene="GENEID"))

## sets up a temporary directory for this example
## (users won't need to do this step)
destination <- tempfile()
dir.create(destination)

## makes an Organism package for human called Homo.sapiens
makeOrganismPackage(pkgname = "Homo.sapiens",
  graphData = gd,
  organism = "Homo sapiens",
  version = "1.0.0",
  maintainer = "Bioconductor Package Maintainer <maintainer@bioconductor.org>",
  author = "Bioconductor Core Team",
  destDir = destination,
  license = "Artistic-2.0")
```

OrganismDb-class *OrganismDb objects*

Description

The OrganismDb class is a container for storing knowledge about existing Annotation packages and the relationships between these resources. The purpose of this object and it's associated methods is to provide a means by which users can conveniently query for data from several different annotation resources at the same time using a familiar interface.

The supporting methods `select`, `cols` and `keys` are used together to extract data from an `OrganismDb` object in a manner that should be consistent with how these are used on the supporting annotation resources.

Methods

In the code snippets below, x is a OrganismDb object. For the metadata and show methods, there is also support for FeatureDb objects.

- `keytypes(x)`: allows the user to discover which keytypes can be passed in to `select` or `keys` and the `keytype` argument.
- `keys(x, keytype)`: returns keys for the database contained in the `OrganismDb` object . By default it will return the "TXNAME" keys for the database, but if used with the `keytype` argument, it will return the keys from that keytype.
- `cols(x)`: shows which kinds of data can be returned for the `OrganismDb` object.
- `select(x, keys, cols, keytype)`: When all the appropriate arguments are specifiedm `select` will retrieve the matching data as a data.frame based on parameters for selected `keys` and `cols` and `keytype` arguments.

Author(s)

Marc Carlson

See Also

makeOrganismPackage

Examples

```
## load a package that creates an OrganismDb
library(Homo.sapiens)
ls(2)
## then the methods can be used on this object.
cols <- cols(Homo.sapiens)[c(7,10,11,12)]
keys <- head(keys(org.Hs.eg.db, "ENTREZID"))
keytype <- "ENTREZID"
res <- select(Homo.sapiens, keys, cols, keytype)
head(res)
```

Index

class:OrganismDb *(*OrganismDb-class*)*, 3
cols,OrganismDb-method
*(*OrganismDb-class*)*, 3

keys,OrganismDb-method
*(*OrganismDb-class*)*, 3
keytypes,OrganismDb-method
*(*OrganismDb-class*)*, 3

makeOrganismPackage, 1, *4*

OrganismDb, *2*
OrganismDb *(*OrganismDb-class*)*, 3
OrganismDb-class, 3

select,OrganismDb-method
*(*OrganismDb-class*)*, 3