org.Pf.plasmo.db

November 6, 2026

`org.Pf.plasmoALIAS2ORF` *Map between Common Gene Symbol Identifiers and ORF*

Description

org.Pf.plasmoALIAS is an R object that provides mappings between common gene symbol identifiers and ORF identifiers.

Details

Each gene symbol maps to a named vector containing the corresponding ORF identifier. The name of the vector corresponds to the gene symbol. Since gene symbols are sometimes redundantly assigned in the literature, users are cautioned that this map may produce multiple matching results for a single gene symbol. Users should map back from the ORF IDs produced to determine which result is the one they want when this happens.

Because of this problem with redundant assigment of gene symbols, is it never advisable to use gene symbols as primary identifiers.

This mapping includes ALL gene symbols including those which are already listed in the SYMBOL map. The SYMBOL map is meant to only list official gene symbols, while the ALIAS maps are meant to store all used symbols.

Mappings were based on data provided by: Plasmo DB http://plasmodb.org/common/downloads/release-28/Pfalciparum3D7/txt With a date stamp from the source of: 23-Mar-2016

References

https://www.ncbi.nlm.nih.gov/entrez/query.fcgi?db=gene

See Also

- `AnnotationDb-class` for use of the `select()` interface.

Examples

```
## select() interface:
## Objects in this package can be accessed using the select() interface
## from the AnnotationDbi package. See ?select for details.

## Bimap interface:
# Convert the object to a list
xx <- as.list(org.Pf.plasmoALIAS2ORF)
# Remove pathway identifiers that do not map to any ORF id
xx <- xx[!is.na(xx)]
if(length(xx) > 0){
    # The ORF identifiers for the first two elements of XX
    xx[1:2]
    # Get the first one
    xx[[1]]
}
```

`org.Pf.plasmo.db` *Bioconductor annotation data package*

Description

Welcome to the org.Pf.plasmo.db annotation Package. This is an organism specific package. The purpose is to provide detailed information about the species abbreviated in the second part of the package name org.Pf.plasmo.db. This package is updated biannually.

Objects in this package are accessed using the `select()` interface. See ?select in the AnnotationDbi package for details.

See Also

- `AnnotationDb-class` for use of `keys()`, `columns()` and `select()`.

Examples

```
## select() interface:
## Objects in this package can be accessed using the select() interface
## from the AnnotationDbi package. See ?select for details.
columns(org.Pf.plasmo.db)

## Bimap interface:
## The 'old style' of interacting with these objects is manipulation as
## bimaps. While this approach is still available we strongly encourage the
## use of select().
ls("package:org.Pf.plasmo.db")
```

org.Pf.plasmoENZYME *Map between ORF IDs and Enzyme Commission (EC) Numbers*

Description

org.Pf.plasmoENZYME is an R object that provides mappings between ORF identifiers and EC numbers.

Details

Each ORF identifier maps to a named vector containing the EC number that corresponds to the enzyme produced by that gene. The name corresponds to the ORF identifier. If this information is unknown, the vector will contain an NA.

Enzyme Commission numbers are assigned by the Nomenclature Committee of the International Union of Biochemistry and Molecular Biology http://www.chem.qmw.ac.uk/iubmb/enzyme/ to allow enzymes to be identified.

An Enzyme Commission number is of the format EC x.y.z.w, where x, y, z, and w are numeric numbers. In org.Pf.plasmoENZYME2ORF, EC is dropped from the Enzyme Commission numbers.

Enzyme Commission numbers have corresponding names that describe the functions of enzymes in such a way that EC x is a more general description than EC x.y that in turn is a more general description than EC x.y.z. The top level EC numbers and names are listed below:

EC 1 oxidoreductases

EC 2 transferases

EC 3 hydrolases

EC 4 lyases

EC 5 isomerases

EC 6 ligases

The EC name for a given EC number can be viewed at http://www.chem.qmul.ac.uk/iupac/jcbn/index.html#6

Mappings between ORF identifiers and enzyme identifiers were obtained using files provided by: KEGG GENOME ftp://ftp.genome.jp/pub/kegg/genomes With a date stamp from the source of: 2011-Mar15

For the reverse map, each EC number maps to a named vector containing the ORF identifier that corresponds to the gene that produces that enzyme. The name of the vector corresponds to the EC number.

References

ftp://ftp.genome.ad.jp/pub/kegg/pathways

See Also

- AnnotationDb-class for use of the select() interface.

Examples

```
## select() interface:
## Objects in this package can be accessed using the select() interface
## from the AnnotationDbi package. See ?select for details.

## Bimap interface:
x <- org.Pf.plasmoENZYME
# Get the ORF identifiers that are mapped to an EC number
mapped_genes <- mappedkeys(x)
# Convert to a list
xx <- as.list(x[mapped_genes])
if(length(xx) > 0) {
  # Get the ENZYME for the first five genes
  xx[1:5]
  # Get the first one
  xx[[1]]
}
# For the reverse map:
# Convert to a list
xx <- as.list(org.Pf.plasmoENZYME2ORF)
if(length(xx) > 0){
   # Gets the ORF identifiers for the first five enzyme
   #commission numbers
   xx[1:5]
   # Get the first one
   xx[[1]]
}
```

---

org.Pf.plasmoGENENAME *Map between ORF IDs and Genes*

Description

org.Pf.plasmoGENENAME is an R object that maps ORF identifiers to the corresponding gene name.

Details

Each ORF identifier maps to a named vector containing the gene name. The vector name corresponds to the ORF identifier. If the gene name is unknown, the vector will contain an NA.

Gene names currently include both the official (validated by a nomenclature committee) and preferred names (interim selected for display) for genes. Efforts are being made to differentiate the two by adding a name to the vector.

Mappings were based on data provided by: Plasmo DB http://plasmodb.org/common/downloads/release-28/Pfalciparum3D7/txt With a date stamp from the source of: 23-Mar-2016

See Also

- AnnotationDb-class for use of the select() interface.

Examples

```
## select() interface:
## Objects in this package can be accessed using the select() interface
## from the AnnotationDbi package. See ?select for details.

## Bimap interface:
x <- org.Pf.plasmoGENENAME
# Get the gene names that are mapped to an ORF identifier
mapped_genes <- mappedkeys(x)
# Convert to a list
xx <- as.list(x[mapped_genes])
if(length(xx) > 0) {
  # Get the GENE NAME for the first five genes
  xx[1:5]
  # Get the first one
  xx[[1]]
}
```

`org.Pf.plasmoGO` *Map between ORF IDs and Gene Ontology (GO)*

Description

org.Pf.plasmoGO is an R object that provides mappings between ORF identifiers and the GO identifiers that they are directly associated with. This mapping and its reverse mapping do NOT associate the child terms from the GO ontology with the gene. Only the directly evidenced terms are represented here.

Details

Each ORF identifier is mapped to a list of lists. The names on the outer list are GO identifiers. Each inner list consists of three named elements: GOID, Ontology, and Evidence.

The GOID element matches the GO identifier named in the outer list and is included for convenience when processing the data using 'lapply'.

The Ontology element indicates which of the three Gene Ontology categories this identifier belongs to. The categories are biological process (BP), cellular component (CC), and molecular function (MF).

The Evidence element contains a code indicating what kind of evidence supports the association of the GO identifier to the ORF id. Some of the evidence codes in use include:

IMP: inferred from mutant phenotype

IGI: inferred from genetic interaction

IPI: inferred from physical interaction

ISS: inferred from sequence similarity

IDA: inferred from direct assay

IEP: inferred from expression pattern

IEA: inferred from electronic annotation

TAS: traceable author statement

NAS: non-traceable author statement

ND: no biological data available

IC: inferred by curator

A more complete listing of evidence codes can be found at:

http://www.geneontology.org/GO.evidence.shtml

Mappings between ORF identifiers and GO information were obtained through their mappings to ORF identifiers. NAs are assigned to ORF identifiers that can not be mapped to any Gene Ontology information. Mappings between Gene Ontology identifiers an Gene Ontology terms and other information are available in a separate data package named GO.

Mappings were based on data provided by: Gene Ontology http://current.geneontology.org/ontology/go-basic.obo With a date stamp from the source of: 2025-07-22

For the reverse map GO2ORF, each GO term maps to a named vector of ORF identifiers. A GO identifier may be mapped to the same ORF identifier more than once but the evidence code can be different. Mappings between Gene Ontology identifiers and Gene Ontology terms and other information are available in a separate data package named GO.

References

ftp://ftp.ncbi.nlm.nih.gov/gene/DATA/

See Also

- org.Pf.plasmoGO2ALLORFS.
- AnnotationDb-class for use of the select() interface.

Examples

```
## select() interface:
## Objects in this package can be accessed using the select() interface
## from the AnnotationDbi package. See ?select for details.

## Bimap interface:
x <- org.Pf.plasmoGO
# Get the ORF identifiers that are mapped to a GO ID
mapped_genes <- mappedkeys(x)
# Convert to a list
xx <- as.list(x[mapped_genes])
if(length(xx) > 0) {
    # Try the first one
    got <- xx[[1]]
    got[[1]][["GOID"]]
    got[[1]][["Ontology"]]
    got[[1]][["Evidence"]]
}
# For the reverse map:
# Convert to a list
xx <- as.list(org.Pf.plasmoGO2ORF)
if(length(xx) > 0){
    # Gets the ORF ids for the top 2nd and 3nd GO identifiers
    goids <- xx[2:3]
    # Gets the ORF ids for the first element of goids
    goids[[1]]
    # Evidence code for the mappings
    names(goids[[1]])
}
```

org.Pf.plasmoGO2ALLORFS *Map Between Gene Ontology (GO) Identifiers and all ORF Identifiers in the subtree*

Description

org.Pf.plasmoGO2ALLORFS is an R object that provides mappings between a given GO identifier and all ORF identifiers annotated at that GO term or one of its children in the GO ontology.

Details

GO consists of three ontologies—molecular function (MF), biological process (BP), and cellular component (CC). All ontologies are structured as directed acyclic graphs (DAGs). Each node in each DAG (tree) is a GO term (id) associated with a named vector of manufacturer identifiers. The name associated with each ORF id corresponds to the evidence code for that GO identifier. This object org.Pf.plasmoGO2ALLORFS maps between a given GO identifier and all ORF identifiers annotated at that GO term or one of its children in the GO ontology.

The evidence code indicates what kind of evidence supports the association between the GO and ORF identifiers. Evidence codes currently in use include:

IMP - inferred from mutant phenotype

IGI - inferred from genetic interaction

IPI - inferred from physical interaction

ISS - inferred from sequence similarity

IDA - inferred from direct assay

IEP - inferred from expression pattern

IEA - inferred from electronic annotation

TAS - traceable author statement

NAS - non-traceable author statement

ND - no biological data available

IC - inferred by curator

A GO identifier may be mapped to the same ORF identifier more than once but the evidence code can be different. Mappings between Gene Ontology identifiers and Gene Ontology terms and other information are available in a separate data package named GO.

Mappings were based on data provided by: Gene Ontology http://current.geneontology.org/ontology/go-basic.obo With a date stamp from the source of: 2025-07-22 and Plasmo DB http://plasmodb.org/common/downloads/rele 28/Pfalciparum3D7/txt With a date stamp from the source of: 23-Mar-2016 For GO2ALL style mappings, the intention is to return all the genes that are the same kind of term as the parent term (based on the idea that they are more specific descriptions of the general term). However because of this intent, not all relationship types will be counted as offspring for this mapping. Only "is a" and "has a" style relationships indicate that the genes from the child terms would be the same kind of thing.

References

ftp://ftp.ncbi.nlm.nih.gov/gene/DATA/

See Also

- AnnotationDb-class for use of the select() interface.

Examples

```
## select() interface:
## Objects in this package can be accessed using the select() interface
## from the AnnotationDbi package. See ?select for details.

## Bimap interface:
# Convert to a list
xx <- as.list(org.Pf.plasmoGO2ALLORFS)
if(length(xx) > 0){
    # Gets the ORF identifiers for the top 2nd and 3nd GO identifiers
    goids <- xx[2:3]
    # Gets all the ORF identifiers for the first element of goids
    goids[[1]]
    # Evidence code for the mappings
    names(goids[[1]])
}
```

---

org.Pf.plasmoMAPCOUNTS

Number of mapped keys for the maps in package org.Pf.plasmo.db

---

Description

org.Pf.plasmoMAPCOUNTS provides the "map count" (i.e. the count of mapped keys) for each map in package org.Pf.plasmo.db.

Details

This "map count" information is precalculated and stored in the package annotation DB. This allows some quality control and is used by the checkMAPCOUNTS function defined in AnnotationDbi to compare and validate different methods (like count.mappedkeys(x) or sum(!is.na(as.list(x)))) for getting the "map count" of a given map.

See Also

- mappedkeys,
- count.mappedkeys,
- checkMAPCOUNTS
- AnnotationDb-class for use of the select() interface.

Examples

```
## select() interface:
## Objects in this package can be accessed using the select() interface
## from the AnnotationDbi package. See ?select for details.

## Bimap interface:
org.Pf.plasmoMAPCOUNTS
```

```
mapnames <- names(org.Pf.plasmoMAPCOUNTS)
org.Pf.plasmoMAPCOUNTS[mapnames[1]]
x <- get(mapnames[1])
sum(!is.na(as.list(x)))
count.mappedkeys(x)   # much faster!

## Check the "map count" of all the maps in package org.Pf.plasmo.db
checkMAPCOUNTS("org.Pf.plasmo.db")
```

---

org.Pf.plasmoORGANISM *The Organism for org.Pf.plasmo*

Description

org.Pf.plasmoORGANISM is an R object that contains a single item: a character string that names the organism for which org.Pf.plasmo was built.

Details

Although the package name is suggestive of the organism for which it was built, org.Pf.plasmoORGANISM provides a simple way to programmatically extract the organism name.

See Also

- AnnotationDb-class for use of the select() interface.

Examples

```
## select() interface:
## Objects in this package can be accessed using the select() interface
## from the AnnotationDbi package. See ?select for details.

## Bimap interface:
org.Pf.plasmoORGANISM
```

---

org.Pf.plasmoPATH *Mappings between ORF identifiers and KEGG pathway identifiers*

Description

KEGG (Kyoto Encyclopedia of Genes and Genomes) maintains pathway data for various organisms. org.Pf.plasmoPATH maps ORF identifiers to the identifiers used by KEGG for pathways

Details

Each KEGG pathway has a name and identifier. Pathway name for a given pathway identifier can be obtained using the KEGG data package that can either be built using AnnBuilder or downloaded from Bioconductor http://www.bioconductor.org.

Graphic presentations of pathways are searchable at url http://www.genome.ad.jp/kegg/pathway.html by using pathway identifiers as keys.

Mappings were based on data provided by: KEGG GENOME ftp://ftp.genome.jp/pub/kegg/genomes With a date stamp from the source of: 2011-Mar15

References

http://www.genome.ad.jp/kegg/

Examples

```
x <- org.Pf.plasmoPATH
# Get the ORF identifiers that are mapped to a KEGG pathway ID
mapped_genes <- mappedkeys(x)
# Convert to a list
xx <- as.list(x[mapped_genes])
if(length(xx) > 0) {
  # Get the PATH for the first five genes
  xx[1:5]
  # Get the first one
  xx[[1]]
}

# For the reverse map:
# Convert the object to a list
xx <- as.list(org.Pf.plasmoPATH2ORF)
# Remove pathway identifiers that do not map to any ORF id
xx <- xx[!is.na(xx)]
if(length(xx) > 0){
    # The ORF identifiers for the first two elements of XX
    xx[1:2]
    # Get the first one
    xx[[1]]
}
```

---

org.Pf.plasmoSYMBOL *Map between ORF Identifiers and Gene Symbols*

Description

org.Pf.plasmoSYMBOL is an R object that provides mappings between ORF identifiers and gene abbreviations.

Details

Each ORF identifier is mapped to the a common abbreviation for the corresponding gene. An NA is reported if there is no known abbreviation for a given gene.

Symbols typically consist of 3 letters that define either a single gene (ABC) or multiple genes (ABC1, ABC2, ABC3). Gene symbols can be used as key words to query public databases such as ORF.

Mappings were based on data provided by: Plasmo DB http://plasmodb.org/common/downloads/release-28/Pfalciparum3D7/txt With a date stamp from the source of: 23-Mar-2016

References

https://www.ncbi.nlm.nih.gov/entrez/query.fcgi?db=gene

Examples

```
x <- org.Pf.plasmoSYMBOL
# Get the gene symbol that are mapped to an ORF identifiers
mapped_genes <- mappedkeys(x)
# Convert to a list
xx <- as.list(x[mapped_genes])
if(length(xx) > 0) {
  # Get the SYMBOL for the first five genes
  xx[1:5]
  # Get the first one
  xx[[1]]
}
# For the reverse map:
x <- org.Pf.plasmoSYMBOL2ORF
# Get the ORF identifiers that are mapped to a gene symbol
mapped_genes <- mappedkeys(x)
# Convert to a list
xx <- as.list(x[mapped_genes])
if(length(xx) > 0) {
  # Get the ORF ID for the first five genes
  xx[1:5]
  # Get the first one
  xx[[1]]
}
```

---

org.Pf.plasmo_dbconn *Collect information about the package annotation DB*

Description

Some convenience functions for getting a connection object to (or collecting information about) the package annotation DB.

Usage

```
org.Pf.plasmo_dbconn()
org.Pf.plasmo_dbfile()
org.Pf.plasmo_dbschema(file="", show.indices=FALSE)
org.Pf.plasmo_dbInfo()
```

Arguments

file	A connection, or a character string naming the file to print to (see the file argument of the cat function for the details).
show.indices	The CREATE INDEX statements are not shown by default. Use show.indices=TRUE to get them.

Details

org.Pf.plasmo_dbconn returns a connection object to the package annotation DB. IMPORTANT: Don't call dbDisconnect on the connection object returned by org.Pf.plasmo_dbconn or you will break all the AnnDbObj objects defined in this package!

org.Pf.plasmo_dbfile returns the path (character string) to the package annotation DB (this is an SQLite file).

org.Pf.plasmo_dbschema prints the schema definition of the package annotation DB.

org.Pf.plasmo_dbInfo prints other information about the package annotation DB.

Value

org.Pf.plasmo_dbconn: a DBIConnection object representing an open connection to the package annotation DB.

org.Pf.plasmo_dbfile: a character string with the path to the package annotation DB.

org.Pf.plasmo_dbschema: none (invisible NULL).

org.Pf.plasmo_dbInfo: none (invisible NULL).

See Also

dbGetQuery, dbConnect, dbconn, dbfile, dbschema, dbInfo

Examples

```
library(DBI)
## Count the number of rows in the "genes" table:
dbGetQuery(org.Pf.plasmo_dbconn(), "SELECT COUNT(*) FROM genes")

org.Pf.plasmo_dbschema()

org.Pf.plasmo_dbInfo()
```

Index

∗ **datasets**
org.Pf.plasmo.db, 2
org.Pf.plasmo_dbconn, 11
org.Pf.plasmoALIAS2ORF, 1
org.Pf.plasmoENZYME, 3
org.Pf.plasmoGENENAME, 4
org.Pf.plasmoGO, 5
org.Pf.plasmoGO2ALLORFS, 7
org.Pf.plasmoMAPCOUNTS, 8
org.Pf.plasmoORGANISM, 9
org.Pf.plasmoPATH, 9
org.Pf.plasmoSYMBOL, 10
∗ **utilities**
org.Pf.plasmo_dbconn, 11

AnnDbObj, *11*

cat, *11*
checkMAPCOUNTS, *8*
count.mappedkeys, *8*

dbconn, *12*
dbConnect, *12*
dbDisconnect, *11*
dbfile, *12*
dbGetQuery, *12*
dbInfo, *12*
dbschema, *12*

mappedkeys, *8*

org.Pf.plasmo *(*org.Pf.plasmo.db*)*, 2
org.Pf.plasmo.db, 2
org.Pf.plasmo_dbconn, 11
org.Pf.plasmo_dbfile
*(*org.Pf.plasmo_dbconn*)*, 11
org.Pf.plasmo_dbInfo
*(*org.Pf.plasmo_dbconn*)*, 11
org.Pf.plasmo_dbschema
*(*org.Pf.plasmo_dbconn*)*, 11
org.Pf.plasmoALIAS2ORF, 1
org.Pf.plasmoENZYME, 3
org.Pf.plasmoENZYME2ORF
*(*org.Pf.plasmoENZYME*)*, 3
org.Pf.plasmoGENENAME, 4
org.Pf.plasmoGO, 5
org.Pf.plasmoGO2ALLORFS, *6*, 7
org.Pf.plasmoGO2ORF *(*org.Pf.plasmoGO*)*, 5
org.Pf.plasmoMAPCOUNTS, 8
org.Pf.plasmoORGANISM, 9
org.Pf.plasmoPATH, 9
org.Pf.plasmoPATH2ORF
*(*org.Pf.plasmoPATH*)*, 9
org.Pf.plasmoSYMBOL, 10
org.Pf.plasmoSYMBOL2ORF
*(*org.Pf.plasmoSYMBOL*)*, 10