RpsiXML

April 20, 2011

`availableXrefs-methods` *Methods for function availableXrefs*

Description

Methods for function `availableXrefs`. See `availableXrefs` for more details

Methods

x = "list" A list of `psimi25Interactor-class` objects, an logical option "intersect" can be set to tell the function whether to return the intersected or the unioned cross ref databases.

x = "psimi25InteractionEntry" Similar to that of "list", "intersect" option is also provided

x = "psimi25Interactor" returns the avaiable cross refences of the given interactor, "intersect" option not available

---

`availableXrefs` *Get cross reference(s) of interactors from PSI-MI 2.5 files*

Description

PSI-MI 2.5 data exchange format encourages data providers to encode cross referenes of the interactors into the file, which prove valuable for annotating the data. These functions help finding out the available cross references of interactors.

Usage

```
availableXrefs(x,...)
xref(object)
```

Arguments

`x` for `availableXrefs`, 'x' can either be an object of `psimi25Interactor-class`, `link{psimi25InteractionEntry}` or a list of `psimi25Interactor-class` objects.

for `xref`, 'object' should be an object of `psimi25Interactor-class`

`object` New interface of RpsiXML uses 'object' uniformly.

`...` intersect, logical option. If set to `TRUE` then only the cross references available for every interactor are returned (intersected), otherwise all the cross references are unioned and returned. See the examples

Details

If `psimi25InteractionEntry` is provided as the parameter of `availableXrefs`, an option named "intersect" can be set to extract only those cross references intersected among all the interactors, namely the ones assigned in every interactor. Please see the examples.

Value

`xref` Returns a matrix with two columns: `db` (external database name) and `id` (external database index)

`availableXrefs`
A vector of characters, the names of external databases referenced in the file

Author(s)

Jitao David Zhang

Examples

```
xmlDir <- system.file("/extdata/psi25files",package="RpsiXML")

hprdxml <- file.path(xmlDir, "hprd_200709_test.xml")
hprdSet <- parsePsimi25Interaction(hprdxml, HPRD.PSIMI25)
hprdInteractors <- interactors(hprdSet)

availableXrefs(hprdSet)
xref(hprdInteractors[[1]])
availableXrefs(hprdInteractors[[1]])
```

`bait` *Extract bait, prey, participant, inhibitor, pubmed, confidence value, interaction type, or neutral component information from an object of psimi25Interaction-class*

Description

The functions return bait/prey UniProt identifier of the given `psimi25Interaction` object.

Usage

```
bait(x,...)
prey(x,...)
participant(x,...)
inhibitor(x,...)
pubmedID(x,...)
confidenceValue(x,...)
neutralComponent(x,...)
```

Arguments

`x`	An object of `psimi25Interaction-class`, see example
`...`	Other parameters to control the identifier returned, not implemented yet

Value

The source database identifier is returned.

Author(s)

Jitao David Zhang <j.zhang@dkfz.de>, Tony Chiang <tchiang@ebi.ac.uk>

References

The UniProt database `http://www.expasy.uniprot.org/`

See Also

`psimi25Interaction-class`

Examples

```
xmlDir <- system.file("/extdata/psi25files",package="RpsiXML")

gridxml <- file.path(xmlDir, "biogrid_200804_test.xml")
gridSet <- parsePsimi25Interaction(gridxml, BIOGRID.PSIMI25)

interExp <- interactions(gridSet)[[1]]
bait(interExp)
prey(interExp)
```

`BIOGRID.PSIMI25` *Supported PSI-MI 2.5 XML data sources*

Description

Objects representing supported PSI-MI 2.5 XML data sources. They are implicitly used in interaction and/or complex parsers. The supporting list will grow as soon as there are new PSI-MI 2.5 compatible data sources available.

For usage see examples of `parsePsimi25Interaction`, `parsePsimi25Complex` and `psimi25XML2Graph`.

Details

If a new repository is added manually by the user, please read the notes in `psimi25Source-class` before coding

References

PSI-MI XML v2.5 data exchange format `http://www.psidev.info/index.php?q=node/60`

BioGRID `http://www.thebiogrid.com`

DIP `http://dip.doe-mbi.ucla.edu/`

HPRD `http://www.hprd.org/`

IntAct `http://www.ebi.ac.uk/intact`

MINT `http://mint.bio.uniroma2.it/mint`

MIPS/CORUM `http://mips.gsf.de/`

MPact `http://mips.gsf.de/genre/proj/mpact`

`buildPCHypergraph` *Build protein complex hypergraph from PSI-MI 2.5 files*

Description

The protein complexes can be represented by hypergraph models, with proteins as nodes and complexes as hypergraphs. This function builds protein complex hypergraph from one or more PSI-MI 2.5 files (complex mode), with the option to split the dataset by organism name or taxonomy ID of the complexes.

Usage

```
buildPCHypergraph(xmlFiles, psimi25source, split.by = c("none", "organismName",
```

Arguments

`xmlFiles` PSI-MI 2.5 files, must be of complex mode.

`psimi25source` PSI-MI 2.5 source indicator, for example INTACT.PSIMI25 for files from IntAct database

`split.by` The qualifier to split the dataset, `none` specifies not to split the dataset, `organismName` and `taxId` splits the dataset according to organism name or taxonomy ID respectively. In the case of splitting, the results are a list of hypergraphs indexed by respective split qualifier

`...` other parameters passed to `parsePsimi25Complex`

Details

See `psimi25Hypergraph-class` for the use of resulting hypergraphs and examples.

Value

In case the dataset is not split (by setting the option `split.by` as `none`, the result is a `psimi25Hypergraph-class` object. If the dataset was split, a list of `psimi25Hypergraph-class` is returned, which is indexed by either the organism name or the taxonomy ID.

Author(s)

Jitao David Zhang <j.zhang@dkfz.de>

See Also

`psimi25Hypergraph-class` for the use of hypergraph objects, `separateXMLDataByExpt` for similar functionality but for interaction mode files

Examples

```
xmlDir <- system.file("/extdata/psi25files",package="RpsiXML")
intactComplexxml <- file.path(xmlDir,"intact_complexSample.xml")

pc2 <- buildPCHypergraph(intactComplexxml, INTACT.PSIMI25,split.by="organismName")

pc2[2]
complexes(pc2[[2]])[1:3]
```

`complexes` *Extract complexes from psimi25ComplexEntry*

Description

Extract complexes from an object of `psimi25ComplexEntry-class`

Usage

```
complexes(x)
```

Arguments

`x`	An object of `psimi25ComplexEntry-class`

Value

A list of `psimi25Complex`

Author(s)

Jitao David Zhang <j.zhang@dkfz.de>, Tony Chiang <tchiang@ebi.ac.uk>

References

PSI-MI 2.5 XML data exchange format `http://www.psidev.info/index.php?q=node/60`

See Also

psimi25Complex-class

Examples

```
## Not run:
xmlDir <- system.file("/extdata/psi25files",package="RpsiXML")

intactComplexxml <- file.path(xmlDir,"intact_complexSample.xml")
intactComplexSet <- parsePsimi25Complex(intactComplexxml,
INTACT.PSIMI25)

complexes(intactComplexSet)
## End(Not run)
```

complexName *Accessor functions for complex*

Description

These functions are used to extract useful information of complex in the form of psimi25Complex-class object.

Usage

```
complexName(x,...)
members(x)
attributesList(x)
```

Arguments

x	An object of psimi25Complex-class
...	Not implemented yet

Details

See examples

Value

complexName	Returns the name of the complex in characters
members	A data frame of protein members building the complex and their information
attributesList	A list of psimi25Attribute objects, recording the attribute name, name accession and value.

Author(s)

Jitao David Zhang <j.zhang@dkfz.de>, Tony Chiang <tchiang@ebi.ac.uk>

See Also

psimi25Complex-class

Examples

```
xmlDir <- system.file("/extdata/psi25files",package="RpsiXML")

intactComplexxml <- file.path(xmlDir,"intact_complexSample.xml")
intactComplexSet <- parsePsimi25Complex(intactComplexxml, INTACT.PSIMI25)
complexSample <- complexes(intactComplexSet)[[2]]

complexName(complexSample)
attributesList(complexSample)
members(complexSample)
```

eListHandler *xmlEventParse handlers for PSI-MI XML documents*

Description

xmlEventParse handlers for PSI-MI XML documents

Usage

```
eListHandler()
iListHandler()
```

Details

A dump() method is supplied to deliver the list.

Value

list; see examples for structure

Author(s)

Vince Carey <stvjc@channing.harvard.edu>

Examples

```
xmlDir <- system.file("/extdata/psi25files",package="RpsiXML")
fn <- file.path(xmlDir, "hprd_200709_test.xml")
XML::xmlEventParse(fn, eListHandler())$dump()
```

getAbstractByPMID *A function to obtain the abstract information via a Pubmed ID*

Description

This function takes a character vector of pubmed IDs and returns a list of pubMedAbst objects indexed by each ID.

Usage

```
getAbstractByPMID(pmID)
```

Arguments

pmID	A chacater vector of pubmed IDs

Value

A list of pubMedAbst objects.

Author(s)

Tony Chiang

Examples

```
xmlDir <- system.file("/extdata/psi25files",package="RpsiXML")
intactxml <- file.path(xmlDir, "intact_2008_test.xml")
x <- parsePsimi25Interaction(intactxml, INTACT.PSIMI25)
y <- interactions(x)[[1]]
getAbstractByPMID(pubmedID(y))
```

graphConverter *maps one type of grap onto another*

Description

The graphConverter function takes a graphNEL object, along with some information about what species that graphNEL is from and what species you wish to convert it into, and then uses the mappings in the inparanoid packages to convert that graph into an equivalent graph from the other species. The hyperGraphConverter performs the same service for either an incidence matrix or a Hypergraph.

Usage

```
graphConverter(graph, srcSpecies, destSpecies,
  srcIDType, destIDType, keepMultGeneMatches=FALSE,
  keepMultProtMatches=FALSE)
hyperGraphConverter(graph, srcSpecies, destSpecies, srcIDType,
  destIDType, mapCols=FALSE, keepMultGeneMatches=FALSE,
  keepMultProtMatches=FALSE)
```

Arguments

`graph` If calling graphConverter. then this is a graphNEL object. Otherwise, it will be an incidence matrix or a hyperGraph

`srcSpecies` The original source species in in paranoid format. In other words, the 3 letters of the genus followed by 2 letters of the species in all caps. Ie. 'HOMSA' is for Homo sapiens etc.

`destSpecies` the destination species in inparanoid format

`srcIDType` The source ID type written exactly as it would be used in a mapping name for an eg package. So for example, 'UNIPROT' is how the uniprot mappings are always written, so we keep that convention here.

`destIDType` the destination ID, written the same way as you would write the srcIDType.

`mapCols` For hyperGraphConverter set to true if the cols are gene names so that they too will be mapped.

`keepMultGeneMatches` Do you want to try and keep the 1st ID in those ambiguous cases where more than one protein is suggested? (You probably want to filter them out - hence the default is FALSE)

`keepMultProtMatches` Do you want to try and keep the 1st ID in those ambiguous cases where more than one protein is suggested? (default = FALSE)

Value

A graphNEL containing as many nodes as it was possible to find matches for.

Author(s)

Marc Carlson

Examples

```
  library(AnnotationDbi)
  directory <- system.file("/extdata/psi25files",
                           package="RpsiXML")
  mintXML <- file.path(directory,
                       "mint_200711_test.xml")
  mintGraph <- separateXMLDataByExpt(xmlFiles=mintXML,
                                     psimi25source = MINT.PSIMI25,
                                     type = "indirect",
                                     directed=TRUE,
                                     abstract=FALSE)

#
#  if(require("hom.Mm.inp.db") & require("org.Mm.eg.db")) {
#    newGraph = graphConverter(mintGraph[[1]], "MUSMU", "HOMSA")
#  }

  ##Get a hypergraph
  xmlDir <- system.file("/extdata/psi25files",package="RpsiXML")
  intactComplexxml <- file.path(xmlDir,"intact_complexSample.xml")
  hyperGraph <- buildPCHypergraph(intactComplexxml, INTACT.PSIMI25)

#  if(require("hom.Hs.inp.db")) {
```

```
#   ##Convert it
#   newHyper = hyperGraphConverter(hyperGraph, "HOMSA", "MUSMU", "UNIPROT",
#   "UNIPROT")
#   }

  ##Get a incidence matrix from ScISI
  if(require(ScISI)) {
    data(ScISIC)
#    if(require("org.Sc.eg.db")) {
#      ##Convert it
#      newScISIC = hyperGraphConverter(ScISIC, "SACCE", "MUSMU",
#                   srcIDType="ORF", destIDType = "EG")
#     }
#   }
```

interactions — *List interactions in the given psimi25InteractionEntry object*

Description

List interactions from an object of the psimi25InteractionEntry-class object

Usage

```
interactions(x)
```

Arguments

x	An object of psimi25InteractionEntry-class, see example

Value

A list of interactions

Author(s)

Jitao David Zhang <j.zhang@dkfz.de>, Tony Chiang <tchiang@ebi.ac.uk>

See Also

psimi25InteractionEntry-class

Examples

```
xmlDir <- system.file("/extdata/psi25files",package="RpsiXML")

hprdxml <- file.path(xmlDir, "hprd_200709_test.xml")
hprdSet <- parsePsimi25Interaction(hprdxml, HPRD.PSIMI25)

interactions(hprdSet)
```

interactionType *Type of the interaction*

Description

Return the interaction type of the `psimi25Interaction`

Usage

```
interactionType(object)
```

Arguments

`object`	An object of `psimi25Interaction-class`

Value

A character string representing the interaction type

Author(s)

Jitao David Zhang <j.zhang@dkfz.de>, Tony Chiang <tchiang@ebi.ac.uk>

See Also

`psimi25Interaction-class`

Examples

```
xmlDir <- system.file("/extdata/psi25files",package="RpsiXML")

hprdxml <- file.path(xmlDir, "hprd_200709_test.xml")
hprdSet <- parsePsimi25Interaction(hprdxml, HPRD.PSIMI25)

interExp <- interactions(hprdSet)[[1]]
interExpTyp <- interactionType(interExp)
```

interactorInfo-methods *Methods for Function interactorInfo in Package ‘RpsiXML’*

Description

See interactorInfo

Methods

- **x = "list"** A list of `psimi25Interactor-class` objects
- **x = "psimi25ComplexEntry"** Object of `psimi25ComplexEntry-class`, parsed from PSI-MI 2.5 complex files
- **x = "psimi25Graph"** Object of `psimi25Graph-class`, parsed by `psimi25XML2Graph` by `psimi25XML2Graph`
- **x = "psimi25InteractionEntry"** Object of `psimi25InteractionEntry-class`, parsed from PSI-MI 2.5 interaction files

---

`interactorInfo` *Interactor info in a matrix*

Description

The function returns the essential information of interactors in a matrix. Xrefs are left out since they have arbitrary numbers of annotation and cannot be summarized into a matrix.

Usage

```
interactorInfo(x)
```

Arguments

`x`	An object which contains psimi25Interactor information, for example objects of `psimi25InteractionEntry-class`, `psimi25Graph-class`, `psimi25Hypergraph-class` or `psimi25ComplexEntry-class` .

Value

A matrix of interactor information, each row represents an interactor The columns are

`sourceDb`	source database
`sourceId`	source database index
`shortLabel`	short label assigned by the source database
`uniprotId`	UniProt ID, NA if not available
`organismName`	the organism of the interactor protein

Author(s)

Jitao David Zhang <j.zhang@dkfz.de>

See Also

`xref`, `availableXrefs`

Examples

```
xmlDir <- system.file("/extdata/psi25files",package="RpsiXML")

hprdxml <- file.path(xmlDir, "hprd_200709_test.xml")
hprdSet <- parsePsimi25Interaction(hprdxml, HPRD.PSIMI25)

hprdInteractorInfo <- interactorInfo(hprdSet)
```

`interactors` *Get information of interactors of the given object*

Description

`interactors` gets the list of interactors stored in the given object. One can also assign a list to replace old interactors.

`numInteractors` returns the length of interactors.

Usage

```
interactors(x)
interactors(x) <- value
numInteractors(x)
```

Arguments

`x`	An object of the subclass of `interactorListBase`
`value`	A list of interactors

Value

The getting method returns a list of interactors. The setting method mutates the object. The count method returns the list length as integer.

Author(s)

Jitao David Zhang <j.zhang@dkfz.de>, Tony Chiang <tchiang@ebi.ac.uk>

See Also

`psimi25InteractionEntry-class`, `psimi25ComplexEntry-class`

Examples

```
xmlDir <- system.file("/extdata/psi25files",package="RpsiXML")

hprdxml <- file.path(xmlDir, "hprd_200709_test.xml")
hprdSet <- parsePsimi25Interaction(hprdxml, HPRD.PSIMI25)
interactors(hprdSet)

## Not run:
intactComplexxml <- file.path(xmlDir,"intact_complexSample.xml")
intactComplexSet <- parsePsimi25Complex(intactComplexxml,
INTACT.PSIMI25)

interactors(intactComplexSet)
numInteractors(intactComplexSet)
## End(Not run)
```

list2Matrix *Converts list into matrix*

Description

Converts a named list representation of a graph into matrix

Usage

```
list2Matrix(namedList, type="interaction")
```

Arguments

namedList	Named list.
type	Character: either interaction or complex

Value

A matrix. A bait-prey matrix if type is "interaction" with baits indexing the columns and prey the rows or a protein complex incidence graph with complexes indexing the columns and proteins, the rows.

If type is interaction, the names of the list are the baits and the entries of the list correspond the prey found by each bait.

If type is complex, the names of the list are the names of the protein complex and the entries are the members of each corresponding complex.

Author(s)

Tony Chiang <tchiang@ebi.ac.uk>

null2na *Turns null or NA into character "NA"*

Description

The functions turns NULL or NA into character "NA"

Usage

```
null2na(x)
```

Arguments

x	A vector

Value

Either the original vector (if not NULL or logical NA) or character NA

Author(s)

Tony Chiang <tchiang@ebi.ac.uk>

Examples

```
null2na(NA)
null2na(NULL)
```

---

numInteractions-methods
Get interaction number of the given object

Description

Get the interaction number of the given object

Methods

x = "psimi25InteractionEntry" Returns the interactio number of the interactionEntry

---

parsePsimi25Interaction
Parsing PSI-MI 2.5 XML documents into interactions

Description

The PSI-MI 2.5 XML format is used widely by many repositories to record protein-protein interaction data as well as protein complex data. This functions parse such files into interactions or complexes.

`parsePsimi25Interaction` is the parser for interaction data and `parsePsimi25Complex` is the parser for complex data.

Usage

```
parsePsimi25Interaction(psimi25file, psimi25source, verbose=TRUE)
parsePsimi25Complex(psimi25file, psimi25source, verbose=FALSE)
```

Arguments

`psimi25file`	character, file name or URL of the XML document
`psimi25source`	A supported data repository source, see also `psimi25Source-class`
`verbose`	logical, whether the parsing state should be displayed verbosely.

Value

psimi25Interaction returns a list of psimi25InteractionEntry objects, each represents one entry in the XML document psimi25Complex returns a psimi25ComplexEntry objects, representing the complex data from one XML document.

Author(s)

Jitao David Zhang <j.zhang@dkfz-heidelberg.de>, Tony Chiang <tchiang@ebi.ac.uk>

References

PSI-MI XML v2.5 data exchange format http://www.psidev.info/index.php?q=node/60

BioGRID http://www.thebiogrid.com

DIP http://dip.doe-mbi.ucla.edu/

HPRD http://www.hprd.org/

IntAct http://www.ebi.ac.uk/intact

MINT http://mint.bio.uniroma2.it/mint

MIPS/CORUM http://mips.gsf.de/

See Also

psimi25Interaction-class, psimi25InteractionEntry-class, psimi25Complex-class psimi25ComplexEntry-class,

Examples

```
# parse interaction data
xmlDir <- system.file("/extdata/psi25files",package="RpsiXML")

gridxml <- file.path(xmlDir, "biogrid_200804_test.xml")
gridSet <- parsePsimi25Interaction(gridxml, BIOGRID.PSIMI25)

intactxml <- file.path(xmlDir, "intact_2008_test.xml")
intactSet <- parsePsimi25Interaction(intactxml, INTACT.PSIMI25, verbose=TRUE)

# parse complex data
intactComplexxml <- file.path(xmlDir,"intact_complexSample.xml")
intactComplexSet <- parsePsimi25Complex(intactComplexxml, INTACT.PSIMI25)
```

---

psimi25Attribute-class

Class "psimi25Attribute"

Description

Persistence of the data structures specified by the PSI-MI 2.5 standard

Objects from the Class

Objects can be created by calls of the form `new("psimi25Attribute", ...)`, or by the constructors.

Slots

`.Data`**:** Object of class `"character"`: value

`name`**:** Object of class `"character"`: name attribute

`nameAc`**:** Object of class `"character"`: nameAc attribute

Extends

Class `"character"`, from data part. Class `"vector"`, by class "character", distance 2.

Methods

iValue `signature(object = "psimi25Attribute")`: ...

iValue<- `signature(object = "psimi25Attribute")`: ...

name `signature(object = "psimi25Attribute")`: ...

name<- `signature(object = "psimi25Attribute", value = "ANY")`: ...

name<- `signature(object = "psimi25Attribute", value = "character")`: ...

nameAc `signature(object = "psimi25Attribute")`: ...

nameAc<- `signature(object = "psimi25Attribute")`: ...

show `signature(object = "psimi25Attribute")`: ...

value<- `signature(object = "psimi25Attribute")`: ...

Author(s)

Jitao David Zhang

References

http://psidev.sourceforge.net/mi/rel25/doc/

`psimi25Complex-class`

Class "psimi25Complex"

Description

A class representing complex data

Objects from the Class

Objects can be created by calls of `parsePsimi25Complex`

Slots

- `sourceDb`: Object of class `"character"`, short label of the source database
- `sourceId`: Object of class `"character"`, complex ID of the source database
- `shortLabel`: Object of class `"character"`, short label of the complex
- `fullName`: Object of class `"character"`, full name of the complex
- `interactorRef`: Object of class `"character"`, reference ID of the interactor
- `organismName`: Object of class `"character"`, orgnism name
- `taxId`: Object of class `"character"`, taxonomy ID
- `members`: Object of class `"data.frame"`, members (UniProt ID if available)
- `attributesList`: A list of attributes of the complex, each as an object of `psimi25Attribute` object

Methods

- **show** `signature(object = "psimi25Complex")`: shows information of the complex
- **sourceDb** `signature(x = "psimi25Complex")`: returns source database
- **sourceId** `signature(x = "psimi25Complex")`: returns source ID
- **attributesList** `signature(x = "psimi25Complex")`: returns attributes of the complex
- **members** `signature(x = "psimi25Complex")`: returns members of the complex
- **complexName** `signature(x = "psimi25Complex")`: returns complex name

Author(s)

Jitao David Zhang <j.zhang@dkfz.de>, Tony Chiang <tchiang@ebi.ac.uk>

See Also

`parsePsimi25Complex`

Examples

```
showClass("psimi25Complex")
```

---

psimi25ComplexEntry-class

Class "psimi25ComplexEntry"

Description

A class representing the complex data extracted from file in PSI-MI 2.5 format.

Objects from the Class

Objects can be created by calls of the form `parsePsimi25Complex`

Slots

- `releaseDate`: Object of class `"character"`, release date
- `interactors`: Object of class `"matrix"`, interactors involved
- `complexes`: Object of class `"list"`, complexes

Methods

- **complexes** `signature(x = "psimi25ComplexEntry")`: returns all complexes
- **interactors** `signature(x = "psimi25ComplexEntry")`: returns information of interactors
- **show** `signature(object = "psimi25ComplexEntry")`: shows information of all the complexes

Author(s)

Jitao David Zhang <j.zhang@dkfz.de>, Tony Chiang <tchiang@ebi.ac.uk>

See Also

`parsePsimi25Complex` `psimi25ComplexEntry`

Examples

```
showClass("psimi25ComplexEntry")
```

`psimi25Experiment-class` *Class "psimi25Experiment"*

Description

Representing an experiment recorded in PSI-MI 2.5 XML files

Objects from the Class

The object is usually only initialized internally

Slots

- `sourceDb`: Object of class `"character"`, source database short label
- `sourceId`: Object of class `"character"`, experiment ID of the source database
- `interactionType`: Object of class `"character"`, interaction type, "Y2H","vv", etc
- `expPubMed`: Object of class `"character"`, PubMed ID of the experiment

Methods

No methods defined with class "psimi25Experiment" in the signature.

Author(s)

Jitao David Zhang <j.zhang@dkfz.de>, Tony Chiang <tchiang@ebi.ac.uk>

Examples

```
showClass("psimi25Experiment")
```

psimi25Graph-class *Class "psimi25Graph" ~~~*

Description

A graph object representing data extracted from PSI-MI 2.5 files

Objects from the Class

Objects can be created by calls of the form `psimi25XML2Graph`

Slots

`interactors`: Object of class `"matrix"`, interactor information in a matrix, Each row represents one interactor. Source IDs are used as row names. Each column represents one annotation. Annotations include: UniProt ID, short label, organism name, and NCBI taxonomy ID. Only those interactors which are the nodes of the psimi25Graph is given.

`abstract`: Object of class `pubMedAbst`

Extends

Class `graphNEL`, directly. Class `graphNEL`, by class "graphNEL", distance 2.

Methods

show `signature(object = "psimi25Graph")`: show method

translateSourceID `signature(r = "psimi25Graph")`: translate the source ID into other IDs

abstract `signature(object="psimi25Graph")`: get the abstract information for the dataset from NCBI

Author(s)

Tony Chiang <tchiang@ebi.ac.uk> , Jitao David Zhang <j.zhang@dkfz.de>

See Also

`psimi25XML2Graph,S4classpsimi25Hypergraph-class`

Examples

```
showClass("psimi25Graph")
```

psimi25Hypergraph2GraphNEL

Convert psimi25Hypergraph to graphNEL

Description

Convert a `psimi25Hypergraph` object to `graphNEL` object, for the purpose of modelling, visualization, etc.

NA nodes will be first removed from the hypergraph, and then the hypergraph is converted to the graphNEL object

Usage

```
psimi25Hypergraph2GraphNEL(x)
```

Arguments

`x`	An object of the class `psimi25Hypergraph`

Value

An object of the class `graphNEL`

Author(s)

Jitao David Zhang <j.zhang@dkfz.de>

Examples

```
## to be implemented
```

psimi25Hypergraph-class

Class "psimi25Hypergraph"

Description

Class to present PSI-MI 2.5 XML data as hypergraph. Proteins are projected as hypergraph nodes and complex composition as hyperedges.

Objects from the Class

Objects can be created by calls of the form `psimi25XML2Graph`

Slots

`interactors`: Object of class `"matrix"`, Object of class `"matrix"`, interactor information in a matrix, Each row represents one interactor. Source IDs are used as row names. Each column represents one annotation. Annotations include: UniProt ID, short label, organism name, and NCBI taxonomy ID. Only those interactors which are the nodes of the psimi25Graph is given.

Extends

Class Hypergraph, directly

Methods

initialize signature(.Object = "psimi25Hypergraph")

show signature(object = "psimi25Hypergraph"): show method,print complex and protein number

interactors signature(object = "psimi25Hypergraph"): list of psimi25Interactor objects, providing full information of complex members

edgeLabel signature(object = "psimi25Hypergraph"): returns complex names (as hyperedge label), as a character vector

hyperedgeNodes signature(object = "Hypergraph"): returns a list of characters: names of the list are complex names and character vector in each list item are the members of that complex

complexes signature(object = "psimi25Hypergraph"): a wrapper of hyperedgeNodes

translateSourceID signature(r = "psimi25Hypergraph"): translate source ID into other IDs

numInteractors signature(r = "psimi25Hypergraph"): returns the number of proteins

interactorInfo signature(r = "psimi25Hypergraph"): returns a data frame containing essential information of the interactors

numEdges signature(r = "psimi25Hypergraph"): returns the number of complexes

revInciMat signature(r = "matrix"): returns the hypergraph built from the incidence matrix

Author(s)

Tony Chiang <tchiang@ebi.ac.uk> , Jitao David Zhang <j.zhang@dkfz.de>

See Also

psimi25XML2Graph,S4classpsimi25Graph-class

Examples

```
xmlDir <- system.file("/extdata/psi25files",package="RpsiXML")
intactComplexxml <- file.path(xmlDir,"intact_complexSample.xml")
pc1 <- buildPCHypergraph(intactComplexxml, INTACT.PSIMI25)

## print number of proteins and complexes (edges)
numNodes(pc1)
## the same as numInteractors(pc1)
numEdges(pc1)

## print proteins (nodes)
nodes(pc1)[1:3]

## print complex names
edgeLabel(pc1)[1:3]
```

```
## print complexes (not so informative with 'hyperedges')
hyperedges(pc1)[1:3]
## better with 'complexes' or 'hyperedgeNodes'
complexes(pc1)[1:3]

## get interactor detailed information
interactors(pc1)[[1]]
```

psimi25Interaction-class *Class "psimi25Interaction"*

Description

A class representing interaction data

Objects from the Class

Objects can be created by calls of the form `parsePsimi25Interaction`

Slots

- `sourceDb`: Object of class `"character"`, source database
- `sourceId`: Object of class `"character"`, source database ID
- `interactionType`: Object of class `"character"`, character, the method used for detecting the interaction ,such as "pull down"
- `expPubMed`: Object of class `"character"`, PubMed ID of the publication that describes the experiment
- `sourceId`: Object of class `"character"`, source database ID of the experiment
- `confidenceValue`: Object of class `"character"`, confidence value of the experimental interaction
- `participant`: Object of class `"character"`, UniProt IDs of the participants. Important when no bait/prey information is available
- `bait`: Object of class `"character"`, UniProt ID of the bait
- `prey`: Object of class `"character"`, UniProt ID(s) of the prey(s)
- `inhibitor`: Object of class `"character"`, UniProt ID of the inhibitor, NA when missing
- `neutralComponent`: Object of class `"character"`, UniProt ID of the neutral components, NA when missing
- `baitUniProt`: An object of class `"character"`
- `preyUniProt`: An object of class `"character"`

Methods

- **interactionType** `signature(x = "psimi25Interaction")`: finds out interaction type
- **show** `signature(object = "psimi25Interaction")`: a print method
- **sourceDb** `signature(x = "psimi25Interaction")`: finds out source database
- **sourceId** `signature(x = "psimi25Interaction")`: returns ID of the source database

bait `signature(x = "psimi25Interaction")`: returns the UniProt ID of the bait

prey `signature(x = "psimi25Interaction")`: returns the UniProt ID(s) of the prey(s)

confidenceValue `signature(x = "psimi25Interaction")`: returns the confidence value of the interaction

Author(s)

Tony Chiang <tchiang@ebi.ac.uk> , Jitao David Zhang <j.zhang@dkfz.de>

See Also

`parsePsimi25Interaction`,`S4CLASSpsimi25Interactor-class`, `S4CLASSpsimi25Experiment-class`, `S4classpsimi25InteractionEntry-class`

Examples

```
showClass("psimi25Interaction")
```

`psimi25InteractionEntry-class`

Class "psimi25InteractionEntry"

Description

A class representing interaction data parsed from PSI-MI 2.5 XML files

Objects from the Class

Objects can be created by calls of the form `parsePsimi25Interaction`

Slots

`organismName`: Object of class `"character"`, the unique organism name(s) of the interactors

`taxId`: Object of class `"character"`, the unique NCBI taxonomy ID(s) of the interactors

`releaseDate`: Object of class `"character"`, character, release date of the data entry, recorded in the `entry` element of the XML file

`interactors`: Object of class `"list"`, a list of `psimi25Interactor-class` objects, each represents one interactor.

`interactions`: Object of class `"list"`, list, a list of `psimi25Interaction-class` objects, each represents one interaction.

Methods

interactions `signature(x = "psimi25InteractionEntry")`: return a list of `psimi25Intearaction` objects, each representing one interaction

interactors `signature(x = "psimi25InteractionEntry")`: a method to find all the interactors within the psimi25InteractionEntry and print them

organismName `signature(x = "psimi25InteractionEntry")`: returns organism names of the interactors

releaseDate `signature(x = "psimi25InteractionEntry")`: returns the release date, serving as a proxy for versioning

show `signature(object = "psimi25InteractionEntry")`: a print method

taxId `signature(x = "psimi25InteractionEntry")`: returns NCBI taxonomy ID

numInteractors `signature(x = "psimi25InteractionEntry")`: returns the number of interactors.

numInteractions `signature(x = "psimi25InteractionEntry")`: returns the number of interactions.

pubmedID `signature(x = "psimi25InteractionEntry")`: returns the (unique) PubMed IDs of the papers reporting the interactions in the entry.

Author(s)

Tony Chiang <tchiang@ebi.ac.uk> , Jitao David Zhang <j.zhang@dkfz.de>

See Also

`parsePsimi25Interaction`, `S4classpsimi25Interaction-class`

Examples

```
showClass("psimi25InteractionEntry")
```

`psimi25Interactor-class`

A class representing interactor from PSI-MI 2.5 XML files

Description

A class representing interactor from PSI-MI 2.5 XML files

Slots

`sourceDb`: Object of class `"character"`, source database

`sourceId`: Object of class `"character"`, ID of the source database

`shortLabel`: Object of class `"character"`, short label of the interactor, if not available the slot will be filled with "fullName" node of the file

`uniprotId`: Object of class `"character"`, UniProt ID of the interactor

`organismName`: Object of class `"character"`, the unique name of the organism name

`taxId`: Object of class `"character"`, the NCBI taxonomy ID

`xref`: Object of class `"environment"`

Methods

show `signature(x = "psimi25Interactor")`: a print method

sourceDb `signature(x = "psimi25Interactor")`: finds out source database

sourceId `signature(x = "psimi25Interactor")`: returns ID of the source database

xref `sigature(x = "psimi25Interactor")`: returns cross-references of the given interactor

Author(s)

Jitao David Zhang <j.zhang@dkfz.de>

Examples

```
showClass("psimi25Interactor")
```

---

psimi25Source-class

Class "psimi25Source"

Description

A class representing data source compatible with PSI-MI 2.5 XML data exchange format

Objects from the Class

Objects can be created by calls of the form `new("psimi25Source", ...)`

Slots

`label`: Object of class `"character"`, a meaningful unique label of the database, does not necessarily have to be the identifier used in the PSI-MI files. The value is advised to be overwritten by the objects

`sourceDb`: Object of class `"character"`, the character string used in the PSI-MI files to identify the source database. The value must be overwritten by the object, depending on the database identifier used in the PSI-MI file.

`uniprotSymbol`: Object of class `"character"`, the identifier of UniProf reference in "primaryRef" or "secondaryRef" attributes of the interactors, telling the parser where to find UniProt symbol of the interactors. The value must be overwritten by the objects.

Methods

sourceDb `signature(x = "psimi25Source")`: returns the source database

uniprot `signature(x = "psimi25Source")`: returns the uniprot identifier of the PSI-MI 2.5 XML file

Note

In case of a new data repository, you can first try creating a new object of the class with `label`, `sourceDb` and `uniprotSymbol` adjusted as needed. Try parsing a function with this new psimi25Source object. In case of warnings or errors, it may mean that the file provided is not strictly compatible with PSI-MI 2.5 data exchange format standard, please then contact the data provider.

Author(s)

Jitao David Zhang <j.zhang@dkfz.de>

See Also

parsePsimi25Interaction, parsePsimi25Complex, psimi25XML2Graph

Examples

```
showClass("psimi25Source")
```

---

psimi25XML2Graph *Convert a vector of PSI-MI 2.5 XML files into graph objects*

Description

The function `psimi25XML2Graph` take a vector of XML 2.5 files from te same data source and generates a graph object based on the type of the files. `psimi25XML2Graph` is a wrapper for `interactionEntry2graph` and `complexEntry2graph`, which transform interactionEntry list and complexEntry list into graphs respectively.

Usage

```
psimi25XML2Graph(psimi25files, psimi25source, type = "interaction",
directed = TRUE, ...)
```

Arguments

`psimi25files` Single file name or a vector of PSI-MI 2.5 XML file names or URLs. In case of splitted data the latter form is preferred. Different datasets or datasets from different sources should not be put into the same vector.

`psimi25source` Source of the PSI-MI 2.5 XML file, see psimi25Source-class

`type` A character string which is either "interaction" or "complex". As the value suggests, use "interaction" if the XML file contains experimental physical data, and "complex" if the file contains curated protein complex membership data.

`directed` Logical, whehter the returned graph object should be directed or undirected

`...` Other parameters passed to parsePsimi25Interaction

Value

If type is "interaction", then a resulting psimi25Graph object is genertated on the aggregation of the XML files. Otherwise if type is "complex,", a resulting psimi25HyperGraph object is generated on the aggregate of the XML files.

Author(s)

Jitao David Zhang, Tony Chiang

See Also

psimi25Source-class, psimi25Graph-class, psimi25Hypergraph-class

Examples

```
xmlDir <- system.file("/extdata/psi25files",package="RpsiXML")

intactxml <- file.path(xmlDir, "intact_2008_test.xml")
intactGraph <- psimi25XML2Graph(intactxml, INTACT.PSIMI25, type="interaction")

intactComplexxml <- file.path(xmlDir,"intact_complexSample.xml")
intactComplexGraph <- psimi25XML2Graph(intactComplexxml, INTACT.PSIMI25, type="complex")
```

`separateXMLDataByExpt` *Convert a vector of PSI-MI 2.5 XML files into graph objects based on pubmedID*

Description

The function `psimi25XML2Graph` take a vector of XML 2.5 files from te same data source and generates a graph object based on the type of the files.

Usage

```
separateXMLDataByExpt(xmlFiles, psimi25source, type = "direct", directed = TRUE,
```

Arguments

`xmlFiles` Single file name or a vector of PSI-MI 2.5 XML file names or URLs.

`type` A character. Currently the user can specify to cull either "direct" interactions or "indirect" interactions.

`psimi25source` Source of the PSI-MI 2.5 XML file, see `psimi25Source-class`

`directed` Logical, whehter the returned graph object should be directed or undirected.

`abstract` Logical; if TRUE, the abstract information will be appended to the graph object.

`...` Other parameters passed to `parsePsimi25Interaction`, for example `verbose=TRUE`

Value

A list of `psimi25Graph-class` are generated indexed by the pubmedID of each bait-prey interaction. WARNING - the abstract information is obtained using the pubmed and buildPubMedAbst functions from the annotate package which warns the user that NCBI may block access to their site. The default is to not obtain the abstract for this reason.

Author(s)

Jitao David Zhang, Tony Chiang

See Also

`psimi25Source-class`, `psimi25Graph-class`, `psimi25Hypergraph-class`

Examples

```
xmlDir <- system.file("/extdata/psi25files",package="RpsiXML")

intactxml <- file.path(xmlDir, "intact_2008_test.xml")
intactGraph <- separateXMLDataByExpt(intactxml, INTACT.PSIMI25, type="indirect")
```

`sourceId-methods` *Extract or set source database name or ID in the source database*

Description

The methods extracts or sets the source database where the object comes from, or its identifier there.

Methods

x = "sourceDbAndId" An object of `sourceDbAndId` (internal), or one of its subclasses

---

`taxId-methods` *Get or Set the NCBI Taxonomy ID or Organism Name*

Description

Get or set the NCBI taxonomy ID or organism name

Methods

x = "organismTaxIdAndName" An `organismTaxIdAndName` object

`translateID-methods` *Translate interactors into other identifiers*

Description

The method finds any annotation IDs of of `psimi25Graph-class` or `psimi25Hypergraph-class` or `psimi25Interactor-class`

Methods

r = "psimi25Graph" An object of `psimi25Graph-class`. It replaces the nodes with translated IDs and returns a new psimi25Graph object back.

r = "psimi25Hypergraph" An object of `psimi25Hypergraph-class`. Since the nodes of psimi25Hypergraph are not mutable, this method accepts a UniProt ID of interactors and returns the translated ID back. NA is also acceptable, which can be used to check interactors without mapping to UniProt

r = "list" A list of `psimi25Interactor-class`, returns a vector of translated identifiers back

r = "psimi25Interactor" An object of `psimi25Interactor-class`, returns translated identifier. Before using the method it is advisable to used xref(x) method to check which cross references are provided.

`translateID` *Finds identifiers of a given object*

Description

see `translateID-methods`

Usage

```
translateID(r, ...)
```

Arguments

`r`	An object of psimi25Graph, psimi25Hypergraph, psimi25Interactor or a list of `psimi25Interactor`
`...`	the symbol of the ID to translate

Details

see `translateID-methods`

Value

The object of the same class as the parameter

Author(s)

Jitao David Zhang <j.zhang@dkfz.de>

See Also

see `translateID-methods`

Examples

```
xmlDir <- system.file("/extdata/psi25files",package="RpsiXML")

hprdxml <- file.path(xmlDir, "hprd_200709_test.xml")
hprdSet <- parsePsimi25Interaction(hprdxml, HPRD.PSIMI25)

it <- interactors(hprdSet)[[1]]
translateID(it, "uniprot")
translateID(it, "entrezgene")

##Not run
intactxml <- file.path(xmlDir, "intact_2008_test.xml")
intactSet <- parsePsimi25Interaction(intactxml, INTACT.PSIMI25)
intactGraph <- psimi25XML2Graph(intactxml, INTACT.PSIMI25)
intactGraphNew <- translateID(intactGraph,"sourceId")## translate the nodes of the graph

intactSetInteractors <- interactors(intactSet)
intactXrefExample <- xref(intactSetInteractors[[1]])
translateID(intactSetInteractors,"intact")
translateID(intactSetInteractors[[1]],"intact")

intactComplexxml <- file.path(xmlDir,"intact_complexSample.xml")
intactComplexSet <- parsePsimi25Complex(intactComplexxml, INTACT.PSIMI25)
intactComplexGraph <- psimi25XML2Graph(intactComplexxml, INTACT.PSIMI25, type="complex")
translateID(intactComplexGraph, "intact", "P49432")
translateID(intactComplexGraph, "intact", NA)
## End(Not run)
```

`uniprot-methods` *Methods for Function uniprot in Package 'RpsiXML'*

Description

if the given parameter is an object of `psimi25Interactor-class`, then the UniProt ID of this protein is returned. In case it is an object of `psimi25Source-class`, the character string representing the path to UniProt is returned.

Methods

x = "psimi25Interactor" An object of `psimi25Interactor-class`

x = "psimi25Source" An object of `psimi25Source-class`

`uniprot` *The UniProt Identifier in the PSI-MI 2.5 XML file*

Description

A character string representing the UniProt identifier in primaryRef or secondaryRef attributes

Usage

```
uniprot(x)
```

Arguments

`x`	An object of `psimi25Source-class`

Value

A character string of the identifier

Author(s)

Tony Chiang <tchiang@ebi.ac.uk>, Jitao David Zhang <j.zhang@dkfz.de>

See Also

`psimi25Source-class`

Examples

```
uniprot(HPRD.PSIMI25)
```

validatePSIMI25 *Validating PSI-MI 2.5 file with MIF25 XML schema*

Description

The function validates given PSI-MI 2.5 file with MIF25 XML schema provided by the Molecular Interactions Workgroup of HUPO Proteomics Standards Initiative

Usage

```
validatePSIMI25(file,
schema = system.file("extdata/schemas/MIF25.xsd", package = "RpsiXML"),
ignore.stderr = TRUE)
```

Arguments

`file` The name or the URL of the file to be validated

`schema` The schema file of PSI-MI 2.5 file format by default

`ignore.stderr`
Whether to print out errors in the console

Details

We advice to set "ignore.stderr=FALSE" for the first time of validating. The error number of the file will be printed in the console. When the number is not 0, one can set "ignore.stderr=TRUE" and see the errors produced during the validation.

Value

The number of errors in validating the file

Note

This function sofar depends on the tool "xmllint" bundled with libxml2 library. We plan to compile it later to run independent of the tool.

Author(s)

Jitao David Zhang <j.zhang@dkfz.de>, Tony Chiang <tchiang@ebi.ac.uk>

References

libxml2 library http://xmlsoft.org/ PSI-MI 2.5 XML schema http://psidev.sourceforge.net/mi/rel25/src/MIF25.xsd

Examples

```
okFile <- system.file("extdata/psi25files/intact_2008_test.xml",
                      package="RpsiXML")
validatePSIMI25(okFile)
```

Index