

Package ‘sybilSBML’

July 20, 2015

Type Package

Title SBML Integration in Package Sybil

Version 2.0.10

Date 2015-07-03

Maintainer C. Jonathan Fritzeimer <clausjonathan.fritzeimer@uni-duesseldorf.de>

Depends R (>= 2.14.2), Matrix, sybil (>= 1.2.4)

Imports methods

Description SBML (Systems Biology Markup Language) integration in sybil. Many constraint based metabolic models are published in SBML format (*.xml). Herewith is the ability to read and check SBML files in sybil provided.

License GPL-3 | file LICENSE

LazyLoad yes

Collate generics.R sbmlPtrClass.R sbmlErrorClass.R sybilSBML.R
uglyHack.R readSBMLmod.R zzz.R

Author C. Jonathan Fritzeimer [cre, ctb],
Gabriel Gelius-Dietrich [aut, ctb],
Deya Alzoubi [ctb]

NeedsCompilation yes

Repository CRAN

Date/Publication 2015-07-20 18:19:57

R topics documented:

closeSBMLfile	2
delSBMLmodel	3
getSBMLCompartList	3
getSBMLErrors	4
getSBMLlevel	5
getSBMLmodel	6
getSBMLmodId	7
getSBMLmodName	8

getSBMLnumCompart	8
getSBMLnumReactions	9
getSBMLnumSpecies	10
getSBMLReactionsList	11
getSBMLSpeciesList	12
getSBMLunitDefinitionsList	14
getSBMLversion	15
openSBMLfile	16
readSBMLmod	16
sbmlError-class	20
sbmlPtr-class	21
validateSBMLdocument	22
versionLibSBML	23

Index 25

closeSBMLfile	<i>Close SBML File</i>
---------------	------------------------

Description

Close SBML file and free all memory associated with the SBML document pointer.

Usage

```
closeSBMLfile(sbmlf)
```

Arguments

sbmlf An object of class [sbmlPtr](#) as returned by [openSBMLfile](#).

Value

Returns NULL invisibly.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Claus Jonathan Fritzeimer <clausjonathan.fritzeimer@uni-duesseldorf.de>

References

Bornstein, B. J., Keating, S. M., Jouraku, A., and Hucka M. (2008) LibSBML: An API Library for SBML. *Bioinformatics* **24**, 880–881.

See Also

[openSBMLfile](#), [sbmlPtr](#)

delSBMLmodel	<i>Delete Pointer to SBML Model</i>
--------------	-------------------------------------

Description

Delete Pointer to SBML Model

Usage

```
delSBMLmodel(sbmlm)
```

Arguments

sbmlm	An object of class sbmlPtr as returned by getSBMLmodel . This is basically a pointer to a SBML model.
-------	---

Value

Returns NULL invisibly.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Claus Jonathan Fritzemeier <clausjonathan.fritzemeier@uni-duesseldorf.de>

References

Bornstein, B. J., Keating, S. M., Jouraku, A., and Hucka M. (2008) LibSBML: An API Library for SBML. *Bioinformatics* **24**, 880–881.

See Also

[getSBMLmodel](#), [sbmlPtr](#)

getSBMLCompartList	<i>Get Compartment List</i>
--------------------	-----------------------------

Description

Retrieve list of compartments included in a SBML model

Usage

```
getSBMLCompartList(sbmlm)
```

Arguments

sbmlm An object of class [sbmlPtr](#) as returned by [getSBMLmodel](#). This is basically a pointer to a SBML model.

Value

If the SBML model contains a ListOfCompartments section, a list is returned:

id character vector containing the compartment id's. If no id is given for a particular compartment, the corresponding value is set to "no_id".

name character vector containing the reaction names. If not available, the value is set to the empty string "".

outside character vector containing the outside argument of the compartments. If not available, the value is set to the empty string "".

All list elements have the same length. If the SBML model does not contain a ListOfCompartments section, NULL is returned.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Claus Jonathan Fritzemeier <clausjonathan.fritzemeier@uni-duesseldorf.de>

References

Bornstein, B. J., Keating, S. M., Jouraku, A., and Hucka M. (2008) LibSBML: An API Library for SBML. *Bioinformatics* **24**, 880–881.

See Also

[getSBMLmodel](#), [sbmlPtr](#)

getSBMLErrors

Get SBML Errors

Description

Retrieve list of errors in a SBML file.

Usage

```
getSBMLErrors(sbmlf)
```

Arguments

sbmlf An object of class [sbmlPtr](#) as returned by [openSBMLfile](#). This is basically a pointer to a SBML document.

Value

If the SBML document contains errors, a list is returned:

infos	a list of infos.
warnings	a list of warnings.
errors	a list of errors.
fatals	a list of fatals.

Each elements of the lists above, is again a list:

id A single integer value containing the error id.

line A single integer value containing the row number of the SBML file producing the error.

column A single integer value containing the column number of the SBML file producing the error.

message A single character string containing the error message.

If the SBML document does not contain errors, TRUE is returned.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Claus Jonathan Fritzemeier <clausjonathan.fritzemeier@uni-duesseldorf.de>

References

Bornstein, B. J., Keating, S. M., Jouraku, A., and Hucka M. (2008) LibSBML: An API Library for SBML. *Bioinformatics* **24**, 880–881.

See Also

[openSBMLfile](#), [validateSBMLdocument](#), [sbmlPtr](#)

getSBMLlevel

Get SBML Level

Description

Retrieve SBML level of SBML file.

Usage

```
getSBMLlevel(sbmlf)
```

Arguments

sbmlf An object of class [sbmlPtr](#) as returned by [openSBMLfile](#). This is basically a pointer to a SBML document.

Value

A single integer value containing the SBML level of the SBML file.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Claus Jonathan Fritzeimer <clausjonathan.fritzeimer@uni-duesseldorf.de>

References

Bornstein, B. J., Keating, S. M., Jouraku, A., and Hucka M. (2008) LibSBML: An API Library for SBML. *Bioinformatics* **24**, 880–881.

See Also

[openSBMLfile](#), [sbmlPtr](#)

getSBMLmodel

Get SBML Model

Description

Retrieve a pointer to a SBML model.

Usage

```
getSBMLmodel(sbmlf, ptrtype = "sbml_mod")
```

Arguments

sbmlf An object of class [sbmlPtr](#) as returned by [openSBMLfile](#).

ptrtype A single character string containing the pointer type.

Value

A object of class [sbmlPtr](#).

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Claus Jonathan Fritzeimer <clausjonathan.fritzeimer@uni-duesseldorf.de>

References

Bornstein, B. J., Keating, S. M., Jouraku, A., and Hucka M. (2008) LibSBML: An API Library for SBML. *Bioinformatics* **24**, 880–881.

See Also

[openSBMLfile](#), [sbmlPtr](#)

getSBMLmodId	<i>Get Model Id</i>
--------------	---------------------

Description

Retrieve model id of a SBML model

Usage

```
getSBMLmodId(sbmlm)
```

Arguments

sbmlm	An object of class sbmlPtr as returned by getSBMLmodel . This is basically a pointer to a SBML model.
-------	---

Value

A single character string containing the model id of the SBML model.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Claus Jonathan Fritzemeier <clausjonathan.fritzemeier@uni-duesseldorf.de>

References

Bornstein, B. J., Keating, S. M., Jouraku, A., and Hucka M. (2008) LibSBML: An API Library for SBML. *Bioinformatics* **24**, 880–881.

See Also

[getSBMLmodel](#), [sbmlPtr](#)

getSBMLmodName *Get Model Name*

Description

Retrieve model name of a SBML model

Usage

```
getSBMLmodName(sbmlm)
```

Arguments

sbmlm An object of class [sbmlPtr](#) as returned by [getSBMLmodel](#). This is basically a pointer to a SBML model.

Value

A single character string containing the model name of the SBML model.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Claus Jonathan Fritzeimer <clausjonathan.fritzeimer@uni-duesseldorf.de>

References

Bornstein, B. J., Keating, S. M., Jouraku, A., and Hucka M. (2008) LibSBML: An API Library for SBML. *Bioinformatics* **24**, 880–881.

See Also

[getSBMLmodel](#), [sbmlPtr](#)

getSBMLnumCompart *Get Number of Compartments*

Description

Retrieve number of compartments of a SBML model.

Usage

```
getSBMLnumCompart(sbmlm)
```


Arguments

sbmlm An object of class [sbmlPtr](#) as returned by [getSBMLmodel](#). This is basically a pointer to a SBML model.

Value

A single integer value containing the number of compartments of a SBML model.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Claus Jonathan Fritzeimer <clausjonathan.fritzeimer@uni-duesseldorf.de>

References

Bornstein, B. J., Keating, S. M., Jouraku, A., and Hucka M. (2008) LibSBML: An API Library for SBML. *Bioinformatics* **24**, 880–881.

See Also

[getSBMLmodel](#), [sbmlPtr](#)

getSBMLnumReactions *Get Number of Reactions*

Description

Retrieve number of reactions of a SBML model.

Usage

```
getSBMLnumReactions(sbmlm)
```

Arguments

sbmlm An object of class [sbmlPtr](#) as returned by [getSBMLmodel](#). This is basically a pointer to a SBML model.

Value

A single integer value containing the number of reactions of a SBML model.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Claus Jonathan Fritzeimer <clausjonathan.fritzeimer@uni-duesseldorf.de>

References

Bornstein, B. J., Keating, S. M., Jouraku, A., and Hucka M. (2008) LibSBML: An API Library for SBML. *Bioinformatics* **24**, 880–881.

See Also

[getSBMLmodel](#), [sbmlPtr](#)

<code>getSBMLnumSpecies</code>	<i>Get Number of Species</i>
--------------------------------	------------------------------

Description

Retrieve number of species of a SBML model.

Usage

```
getSBMLnumSpecies(sbmlm)
```

Arguments

<code>sbmlm</code>	An object of class sbmlPtr as returned by getSBMLmodel . This is basically a pointer to a SBML model.
--------------------	---

Value

A single integer value containing the number of species of a SBML model.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Claus Jonathan Fritzemeier <clausjonathan.fritzemeier@uni-duesseldorf.de>

References

Bornstein, B. J., Keating, S. M., Jouraku, A., and Hucka M. (2008) LibSBML: An API Library for SBML. *Bioinformatics* **24**, 880–881.

See Also

[getSBMLmodel](#), [sbmlPtr](#)

getSBMLReactionsList *Get Reaction List*

Description

Retrieve list of reactions included in a SBML model

Usage

```
getSBMLReactionsList(sbmlm)
```

Arguments

sbmlm An object of class [sbmlPtr](#) as returned by [getSBMLmodel](#). This is basically a pointer to a SBML model.

Value

If the SBML model contains a ListOfReactions section, a list is returned:

id	character vector containing the reaction id's. If no id is given for a particular reaction, the corresponding value is set to "no_id".
name	character vector containing the reaction names. If not available, the value is set to the empty string "".
reversible	logical vector containing the reversible flag of the reactions. If no reversible flag is given for a particular reaction, the corresponding value is set to FALSE.
notes	character vector containing the note strings. If no note is given for a particular reaction, the corresponding value is set to the empty string "".
annotation	character vector containing the annotation strings. If no annotation is given for a particular reaction, the corresponding value is set to the empty string "".
reactants	a list containing the reactions reactants. id character vector containing the species reference id's involved as reactants. If no id is given for a particular species, the corresponding value is set to "no_id". species character vector containing the species id's involved as reactants. If no id is given for a particular species, the corresponding value is set to "no_species". stoichiometry numeric vector containing the stoichiometry of the species. If no value is given for a particular species, the corresponding value is set to 1. If no reactant is given for a particular reaction, the corresponding value is set to NULL.
products	a list containing the reactions products.

id character vector containing the species reference id's involved as products. If no id is given for a particular species, the corresponding value is set to "no_id".

species character vector containing the species id's involved as products. If no id is given for a particular species, the corresponding value is set to "no_species".

stoichiometry numeric vector containing the stoichiometry of the species. If no value is given for a particular species, the corresponding value is set to 1.

If no product is given for a particular reaction, the corresponding value is set to NULL.

kinetic_law a list containing the reactions parameters.

id character vector containing the parameter id's. If no id is given for a particular parameter, the corresponding value is set to "no_id".

value numeric vector containing the parameter values. If no value is given for a particular parameter, the corresponding value is set to 0.

units character vector containing the parameter units. If no unit is given for a particular parameter, the corresponding value is set to the empty string "".

If no parameter is given for a particular reaction, the corresponding value is set to NULL.

All list elements have the same length. If the SBML model does not contain a ListOfReactions section, NULL is returned.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Claus Jonathan Fritzeimer <clausjonathan.fritzeimer@uni-duesseldorf.de>

References

Bornstein, B. J., Keating, S. M., Jouraku, A., and Hucka M. (2008) LibSBML: An API Library for SBML. *Bioinformatics* **24**, 880–881.

See Also

[getSBMLmodel](#), [sbmlPtr](#)

getSBMLSpeciesList *Get Species List*

Description

Retrieve list of species included in a SBML model

Usage

```
getSBMLSpeciesList(sbm1m)
```

Arguments

sbm1m An object of class [sbmlPtr](#) as returned by [getSBMLmodel](#). This is basically a pointer to a SBML model.

Value

If the SBML model contains a ListOfSpecies section, a list is returned:

id	character vector containing the species id's. If no id is given for a particular species, the corresponding value is set to "no_id".
name	character vector containing the species names. If not available, the value is set to the empty string "".
compartment	character vector containing the species compartments. If not available, the value is set to the empty string "".
charge	integer vector containing the charge of the species. If no charge is given for a particular species, the corresponding value is set to 0.
boundaryCondition	logical vector containing the boundaryCondition flag of the species. If no boundaryCondition flag is given for a particular species, the corresponding value is set to FALSE.

All list elements have the same length. If the SBML model does not contain a ListOfSpecies section, NULL is returned.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Claus Jonathan Fritzemeier <clausjonathan.fritzemeier@uni-duesseldorf.de>

References

Bornstein, B. J., Keating, S. M., Jouraku, A., and Hucka M. (2008) LibSBML: An API Library for SBML. *Bioinformatics* **24**, 880–881.

See Also

[getSBMLmodel](#), [sbmlPtr](#)

```
getSBMLunitDefinitionsList
```

Get Unit Definitions List

Description

Retrieve list of unit definitions included in a SBML model

Usage

```
getSBMLunitDefinitionsList(sbmlm)
```

Arguments

sbmlm	An object of class <code>sbmlPtr</code> as returned by <code>getSBMLmodel</code> . This is basically a pointer to a SBML model.
-------	---

Value

If the SBML model contains a `listOfUnitDefinitions` section, a list is returned:

definition_id	character vector containing the unit definition id's. If no id is given for a particular unit definition, the corresponding value is set to "no_id".
---------------	--

definition	a list containing the units.
------------	------------------------------

kind	character vector containing the unit kind's involved as unit definitions. If no kind is given for a particular unit, the corresponding value is set to "no_kind".
-------------	---

scale	integer vector containing the scale. If no scale is given for a particular unit, the corresponding value is set to 0.
--------------	---

exponent	integer vector containing the exponent. If no exponent is given for a particular unit, the corresponding value is set to 1.
-----------------	---

multiplier	numeric vector containing the multiplier. If no multiplier is given for a particular species, the corresponding value is set to 1.
-------------------	--

If no unit is given for a particular unit definition, the corresponding value is set to NULL.

All list elements have the same length. If the SBML model does not contain a `listOfUnitDefinitions` section, NULL is returned.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Claus Jonathan Fritzemeier <clausjonathan.fritzemeier@uni-duesseldorf.de>

References

Bornstein, B. J., Keating, S. M., Jouraku, A., and Hucka M. (2008) LibSBML: An API Library for SBML. *Bioinformatics* **24**, 880–881.

See Also

[getSBMLmodel](#), [sbmlPtr](#)

getSBMLversion	<i>Get SBML Version</i>
----------------	-------------------------

Description

Retrieve SBML version of SBML file.

Usage

```
getSBMLversion(sbmlf)
```

Arguments

sbmlf An object of class [sbmlPtr](#) as returned by [openSBMLfile](#). This is basically a pointer to a SBML document.

Value

A single integer value containing the SBML version of the SBML file.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Claus Jonathan Fritzemeier <clausjonathan.fritzemeier@uni-duesseldorf.de>

References

Bornstein, B. J., Keating, S. M., Jouraku, A., and Hucka M. (2008) LibSBML: An API Library for SBML. *Bioinformatics* **24**, 880–881.

See Also

[openSBMLfile](#), [sbmlPtr](#)

openSBMLfile	<i>Open SBML File</i>
--------------	-----------------------

Description

Retrieve a pointer to a SBML document.

Usage

```
openSBMLfile(fname, ptrtype = "sbml_doc")
```

Arguments

fname	A single character string containing a file name of an SBML file.
ptrtype	A single character string containing the pointer type.

Value

A object of class [sbmlPtr](#).

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Claus Jonathan Fritzeimer <clausjonathan.fritzeimer@uni-duesseldorf.de>

References

Bornstein, B. J., Keating, S. M., Jouraku, A., and Hucka M. (2008) LibSBML: An API Library for SBML. *Bioinformatics* **24**, 880–881.

See Also

[closeSBMLfile](#), [sbmlPtr](#)

readSBMLmod	<i>Reads a Metabolic Network in SBML Format</i>
-------------	---

Description

The function reads a metabolic network in SBML format. The function returns an S4 object of the class [modelorg](#)

Usage

```
readSBMLmod(filename, description,
            def_bnd = SYBIL_SETTINGS("MAXIMUM"),
            validateSBML = FALSE,
            extMetFlag = "b",
            bndCond = TRUE,
            ignoreNoAn = FALSE,
            mergeMet = TRUE,
            balanceReact = TRUE,
            remUnusedMetReact = TRUE,
            singletonMet = FALSE,
            deadEndMet = FALSE,
            remMet = FALSE,
            constrMet = FALSE,
            tol = SYBIL_SETTINGS("TOLERANCE"))
```

Arguments

filename	SBML file containing the model
description	Character vector containing a description of the model. Default: filename.
def_bnd	Single numeric value. Absolute value for upper and lower bounds for reaction constraints – if they are missing in the SBML file. Default: MAXIMUM(SYBIL_SETTINGS).
validateSBML	Boolean: validate the xml file. Default: TRUE.
extMetFlag	A single character string how external metabolites were identified. If the metabolite id ends in "_extMetFlag", the corresponding metabolite is considered to be external and will be removed if bndCond is set to FALSE. Default: "b".
bndCond	Boolean: use the value of SBML tag boundaryCondition in order to identify external metabolites. Default: TRUE.
ignoreNoAn	Boolean: if set to TRUE, any notes and annotation fields in the listOfReactions in an SBML file are ignored. Default: FALSE.
mergeMet	Boolean: if set to TRUE, metabolites used more than once as reactant or product in a particular reaction are added up, see details below. If set to FALSE, the last value is used without warning. Default: TRUE.
balanceReact	Boolean: if set to TRUE, metabolites used as reactant and product in a particular reaction at the same time are balanced, see details below. If set to FALSE the last value is used without warning (reactants before products). Default: TRUE.

remUnusedMetReact	Boolean: if set to TRUE, metabolites and reactions which are not used in the stoichiometric matrix will be removed. A metabolite or a reaction is considered as unused, if the corresponding element of rowSums (metabolites) or colSums (reactions) of the binary version of the stoichiometric matrix is zero, see details below. If set to FALSE, only a warning is given. Default: FALSE.
singletonMet	Boolean: if set to TRUE, metabolites appearing only once in the stoichiometric matrix are identified. Metabolites appear only once, if rowSums of the binary stoichiometric matrix is one in the corresponding row, see details below. Default: FALSE.
deadEndMet	Boolean: if set to TRUE, metabolites which are produced but not consumed, or vice versa are identified, see details below. If both arguments singletonMet and deadEndMet are set to TRUE, the function will first look for singleton metabolites, and exclude them (and the corresponding reactions) from the search list. Afterwards, dead end metabolites are searched only in the smaller model. Default: FALSE.
remMet	Boolean: if set to TRUE, metabolites identified as singleton or dead end metabolites will be removed from the model. Additionally, reactions containing such metabolites will be removed also. Default: FALSE.
constrMet	Boolean: if set to TRUE, reactions containing metabolites identified as singleton or dead end metabolites will be constrained to zero. Default: FALSE.
tol	A single numeric value, giving the smallest positive floating point number unequal to zero, see details below. Default: TOLERANCE(SYBIL_SETTINGS).

Details

The library libSBML is used to read an SBML file and to collect the information in an object of the class `modelorg`.

If a metabolite is used more than once as product or reactant of a particular reaction, it is merged: $a + (2) a$ is converted to $(3) a$ and a warning will be given.

If a metabolite is used first as reactant and then as product of a particular reaction, the reaction is balanced: $(2) b + a \rightarrow b + c$ is converted to $b + a \rightarrow c$

A binary version of the stoichiometric matrix S is constructed via $|S| > tol$.

A binary version of the stoichiometric matrix S is scanned for reactions and metabolites which are not used in S . If there are some, a warning will be given and the corresponding reactions and metabolites will be removed from the model if `remUnusedMetReact` is set to TRUE.

The binary version of the stoichiometric matrix S is scanned for metabolites, which are used only once in S . If there are some, at least a warning will be given. If either `constrMet` or `remMet` is set to TRUE, the binary version of S is scanned for paths of singleton metabolites. If `constrMet` is set to TRUE, reactions containing those metabolites will be constrained to zero; if `remMet` is set to TRUE, the metabolites and the reactions containing those metabolites will be removed from the network.

In order to find path of singleton metabolites a binary version of the stoichiometric matrix S is used. Sums of rows gives the vector of metabolite usage, each element is the number of reactions a metabolite participates. A single metabolite (singleton) is a metabolite with a row sum of zero. All columns in S (reactions) containing singleton metabolites will be set to zero. And again, singleton metabolites will be searched until none are found.

The algorithm to find dead end metabolites works in a quite similar way, but not in the binary version of the stoichiometric matrix. Here, metabolite i is considered as dead end, if it is for example produced by reaction j but not used by any other reaction k .

Value

An S4 object of the class `modelorg`.

Note

The function `readSBMLmod` makes use of the library `libSBML` (<http://www.sbml.org>).

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Claus Jonathan Fritzeimer <clausjonathan.fritzeimer@uni-duesseldorf.de>

References

The BiGG database <http://bigg.ucsd.edu/>.

Bornstein, B. J., Keating, S. M., Jouraku, A., and Hucka M. (2008) LibSBML: An API Library for SBML. *Bioinformatics* **24**, 880–881.

Schellenberger, J., Park, J. O., Conrad, T. C., and Palsson, B. Ø., (2010) BiGG: a Biochemical Genetic and Genomic knowledgebase of large scale metabolic reconstructions. *BMC Bioinformatics* **11**, 213.

Becker, S. A., Feist, A. M., Mo, M. L., Hannum, G., Palsson, B. Ø. and Herrgard, M. J. (2007) Quantitative prediction of cellular metabolism with constraint-based models: the COBRA Toolbox. *Nat Protoc* **2**, 727–738.

Schellenberger, J., Que, R., Fleming, R. M. T., Thiele, I., Orth, J. D., Feist, A. M., Zielinski, D. C., Bordbar, A., Lewis, N. E., Rahmanian, S., Kang, J., Hyduke, D. R. and Palsson, B. Ø. (2011) Quantitative prediction of cellular metabolism with constraint-based models: the COBRA Toolbox v2.0. *Nat Protoc* **6**, 1290–1307.

See Also

[validateSBMLdocument](#)

Examples

```
dir <- system.file(package = "sybilSBML", "extdata")
file <- file.path(dir, "ecoli_core_model.xml")
mod <- readSBMLmod(file, bndCond = FALSE)
```

sbmlError-class	Class "sbmlError"
-----------------	-------------------

Description

Class to administrate Infos, Warnings, Errors and Fatals returned after validation of an SBML file.

Objects from the Class

Objects can be created by calls of the form

```
err <- sbmlError(err, sbmlf).
```

err: an object of class `sbml_error` containing a list of SBML errors.

sbmlf: an object of class `sbmlPtr` containing a pointer to an SBML document object.

This constructor function is only used internally, objects of class `sbmlError` are returned by the functions `validateSBMLdocument` and `getSBMLErrors`.

Slots

`sbmlInfos`: Object of class "list" containing all Infos.

`sbmlWarnings`: Object of class "list" containing all Warnings.

`sbmlErrors`: Object of class "list" containing all Errors.

`sbmlFatals`: Object of class "list" containing all Fatals.

`sbmlFileName`: Object of class "character" containing the file name of the SBML file generating the errors.

`sbmlDocKey`: Object of class "character" containing a single character string functioning as a unique key to a SBML document object.

Methods

getNumErrors signature(object = "sbmlError"): returns a vector of length 5 containing the number of Infos, Warnings, Errors, Fatals and the total number of entries.

length signature(x = "sbmlError"): returns the total number of entries.

printSlot signature(object = "sbmlError", ws = "character"): prints the slot given in argument ws. It can be "Infos", "Warnings", "Errors" or "Fatals".

sbmlDocKey signature(object = "sbmlError"): gets the sbmlDocKey slot.

sbmlErrors signature(object = "sbmlError"): gets the sbmlErrors slot.

sbmlFatals signature(object = "sbmlError"): gets the sbmlFatals slot.

sbmlFileName signature(object = "sbmlError"): gets the sbmlFileName slot.

sbmlInfos signature(object = "sbmlError"): gets the sbmlInfos slot.

sbmlWarnings signature(object = "sbmlError"): gets the sbmlWarnings slot.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Claus Jonathan Fritzemeier <clausjonathan.fritzemeier@uni-duesseldorf.de>

See Also

[validateSBMLdocument](#), [getSBMLErrors](#)

Examples

```
showClass("sbmlError")
```

sbmlPtr-class	Class "sbmlPtr"
---------------	-----------------

Description

Containing SBML pointer objects.

Objects from the Class

Objects can be created by calls of the form

```
doc <- sbmlDocPointer(pointer).
```

pointer: an object of class `sbml_doc_ptr` containing a pointer to a SBML document.

```
doc <- sbmlModPointer(pointer, sbmlDoc).
```

pointer: an object of class `sbml_mod_ptr` containing a pointer to a SBML model.

sbmlDoc: an object of class `sbmlPtr` containing a pointer to a SBML document (returned by `sbmlDocPointer`).

This constructor functions are only used internally, objects of class `sbmlPtr` are returned by the functions [openSBMLfile](#) and [getSBMLmodel](#).

Slots

sbmlPtrType: Object of class "character" containing the type of the pointer.

sbmlPointer: Object of class "externalptr" containing a pointer to a SBML document or model.

sbmlFileName: Object of class "character" containing the file name of the SBML file generating the errors.

sbmlDocKey: Object of class "character" containing a single character string functioning as a unique key to a SBML document object.

Methods

isNULLpointerSBML signature(object = "sbmlPtr"): returns TRUE if sbmlPointer(object) is a NULL pointer, otherwise FALSE.

isSBMLdocpointer signature(object = "sbmlPtr"): returns TRUE if sbmlPointer(object) is a pointer to a SBML document, otherwise FALSE.

isSBMLmodpointer signature(object = "sbmlPtr"): returns TRUE if sbmlPointer(object) is a pointer to a SBML model, otherwise FALSE.

sbmlDocKey signature(object = "sbmlPtr"): gets the sbmlDocKey slot.

sbmlFileName signature(object = "sbmlPtr"): gets the sbmlFileName slot.

sbmlPointer signature(object = "sbmlPtr"): gets the sbmlPointer slot.

sbmlPtrType signature(object = "sbmlPtr"): gets the sbmlPtrType slot.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Claus Jonathan Fritzeimer <clausjonathan.fritzeimer@uni-duesseldorf.de>

See Also

[openSBMLfile](#), [getSBMLmodel](#)

Examples

```
showClass("sbmlError")
```

validateSBMLdocument *Validate SBML*

Description

Validate files written in SBML.

Usage

```
validateSBMLdocument(sbmlf)
```

Arguments

sbmlf A single character string containing a filename of an SBML file, or an object of class [sbmlPtr](#) as returned by [openSBMLfile](#).

Details

The given SBML file is validated by the libSBML function `checkConsistency`. Errors are retrieved by the libSBML function `getError` and returned as list.

Value

If argument `smlf` is an object of class `character`, the function returns `TRUE`, if no infos, warnings, errors or fatals have been found. Otherwise a the value returned is a list.

<code>infos</code>	Infos
<code>warnings</code>	Warnings
<code>errors</code>	Errors
<code>fatals</code>	Fatals

Each entry of one of these lists is again a list containing the message id, line and column number of the SBML file generating the message and the corresponding message text.

If argument `smlf` is an object of class `smlPtr`, the function returns the number of errors found in the SBML document.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Claus Jonathan Fritzeimer <clausjonathan.fritzeimer@uni-duesseldorf.de>

References

Bornstein, B. J., Keating, S. M., Jouraku, A., and Hucka M. (2008) LibSBML: An API Library for SBML. *Bioinformatics* **24**, 880–881.

See Also

[readSBMLmod](#), [openSBMLfile](#), [getSBMLerrors](#), [smlPtr](#)

Examples

```
dir <- system.file(package = "sybilSBML", "extdata")
file <- file.path(dir, "ecoli_core_model.xml")
err <- validateSBMLdocument(file)
```

versionLibSBML

Get libSBML Version

Description

Retrieve libSBML version number.

Usage

```
versionLibSBML()
```

Value

A single character string containing the dotted version of the libSBML version number.

Author(s)

Gabriel Gelius-Dietrich <geliudie@uni-duesseldorf.de>

Maintainer: Claus Jonathan Fritzeimer <clausjonathan.fritzeimer@uni-duesseldorf.de>

References

Bornstein, B. J., Keating, S. M., Jouraku, A., and Hucka M. (2008) LibSBML: An API Library for SBML. *Bioinformatics* **24**, 880–881.

Index

*Topic **IO**

- closeSBMLfile, [2](#)
- delSBMLmodel, [3](#)
- getSBMLCompartList, [3](#)
- getSBMLErrors, [4](#)
- getSBMLlevel, [5](#)
- getSBMLmodel, [6](#)
- getSBMLmodId, [7](#)
- getSBMLmodName, [8](#)
- getSBMLnumCompart, [8](#)
- getSBMLnumReactions, [9](#)
- getSBMLnumSpecies, [10](#)
- getSBMLReactionsList, [11](#)
- getSBMLSpeciesList, [12](#)
- getSBMLunitDefinitionsList, [14](#)
- getSBMLversion, [15](#)
- openSBMLfile, [16](#)
- readSBMLmod, [16](#)
- validateSBMLdocument, [22](#)
- versionLibSBML, [23](#)

*Topic **classes**

- sbmlError-class, [20](#)
- sbmlPtr-class, [21](#)

closeSBMLfile, [2](#), [16](#)

delSBMLmodel, [3](#)

getNumErrors (sbmlError-class), [20](#)

getNumErrors, sbmlError-method
(sbmlError-class), [20](#)

getSBMLCompartList, [3](#)

getSBMLErrors, [4](#), [20](#), [21](#), [23](#)

getSBMLlevel, [5](#)

getSBMLmodel, [3](#), [4](#), [6](#), [7–15](#), [21](#), [22](#)

getSBMLmodId, [7](#)

getSBMLmodName, [8](#)

getSBMLnumCompart, [8](#)

getSBMLnumReactions, [9](#)

getSBMLnumSpecies, [10](#)

getSBMLReactionsList, [11](#)

getSBMLSpeciesList, [12](#)

getSBMLunitDefinitionsList, [14](#)

getSBMLversion, [15](#)

isNULLpointerSBML (sbmlPtr-class), [21](#)

isNULLpointerSBML, sbmlPtr-method
(sbmlPtr-class), [21](#)

isSBMLdocpointer (sbmlPtr-class), [21](#)

isSBMLdocpointer, sbmlPtr-method
(sbmlPtr-class), [21](#)

isSBMLmodpointer (sbmlPtr-class), [21](#)

isSBMLmodpointer, sbmlPtr-method
(sbmlPtr-class), [21](#)

length, sbmlError-method
(sbmlError-class), [20](#)

modelorg, [16](#), [18](#), [19](#)

openSBMLfile, [2](#), [4–7](#), [15](#), [16](#), [21–23](#)

printSlot (sbmlError-class), [20](#)

printSlot, sbmlError, character-method
(sbmlError-class), [20](#)

readSBMLmod, [16](#), [23](#)

rowSums, [18](#)

sbmlDocKey (sbmlPtr-class), [21](#)

sbmlDocKey, sbmlError-method
(sbmlError-class), [20](#)

sbmlDocKey, sbmlPtr-method
(sbmlPtr-class), [21](#)

sbmlDocPointer (sbmlPtr-class), [21](#)

sbmlError (sbmlError-class), [20](#)

sbmlError-class, [20](#)

sbmlErrors (sbmlError-class), [20](#)

sbmlErrors, sbmlError-method
(sbmlError-class), [20](#)

sbmlFatal (sbmlError-class), [20](#)

sbmlFatal, sbmlError-method
 (sbmlError-class), 20
sbmlFileName (sbmlPtr-class), 21
sbmlFileName, sbmlError-method
 (sbmlError-class), 20
sbmlFileName, sbmlPtr-method
 (sbmlPtr-class), 21
sbmlInfos (sbmlError-class), 20
sbmlInfos, sbmlError-method
 (sbmlError-class), 20
sbmlModPointer (sbmlPtr-class), 21
sbmlPointer (sbmlPtr-class), 21
sbmlPointer, sbmlPtr-method
 (sbmlPtr-class), 21
sbmlPtr, 2–16, 20, 22, 23
sbmlPtr-class, 21
sbmlPtrType (sbmlPtr-class), 21
sbmlPtrType, sbmlPtr-method
 (sbmlPtr-class), 21
sbmlWarnings (sbmlError-class), 20
sbmlWarnings, sbmlError-method
 (sbmlError-class), 20

validateSBMLdocument, 5, 19–21, 22
versionLibSBML, 23