Package ‘minval’

April 12, 2018

Type Package
Title MINimal VALidation for Stoichiometric Reactions
Version 0.8-1
URL https://github.com/gibbslab/minval
Maintainer Daniel Osorio <dcosorioh@unal.edu.co>
Suggests testthat, knitr, rmarkdown, sybil, glpkAPI
Description For a given set of stoichiometric reactions, this package
evaluates the mass and charge balance, extracts all reactants, products, orphan
metabolites, metabolite names and compartments. Also are included some options
to characterize and write models in TSV and SBML formats.
License GPL (>= 2)
LazyData TRUE
NeedsCompilation no
Depends R (>= 2.10)
RoxygenNote 6.0.1
VignetteBuilder knitr
Author Daniel Osorio [aut, cre],
Janneth Gonzalez [aut, ths],
Andres Pinzon [aut, ths]
Repository CRAN
Date/Publication 2018-04-12 03:32:58 UTC

R topics documented:

characterizeReactions .................................................. 2
checkBalance ............................................................. 3
compartments ............................................................. 4
downloadChEBI ............................................................ 6
metabolites ............................................................... 7
orphanMetabolites ....................................................... 8
characterizeReactions

Description

For a given set of stoichiometric reactions, this function:

- Counts the number of reactions,
- Computes the relative frequency of each reaction type (transport, exchange and compartmentalized),
- Computes the relative frequency of reactions by compartment,
- Counts the number of unique metabolites,
- Computes the relative frequency of metabolites by compartment.

Usage

characterizeReactions(reactionList, rawOutput = FALSE)

Arguments

reactionList  A set of stoichiometric reaction with the following characteristics:
  - Arrows symbols must be given in the form '=>' or '<='
  - Inverse arrow symbols '<=' or other types as: '==>', '<==>', '<=>' will not be parsed and will lead to errors.
  - Arrow symbols and plus signs (+) must be surrounded by a space character
  - Stoichiometric coefficients must be surrounded by a space character and not by parentheses.
  - Each metabolite must have only one stoichiometric coefficient, substituents must be joined to metabolite name by a hyphen (-) symbol.
  - Exchange reactions have only one metabolite before arrow symbol
  - Compartments must be given between square brackets ([compartment]) joined at the end of metabolite name

Some examples of valid stoichiometric reactions are:
  - H2O[c] + Urea-1-Carboxylate[c] <= 2 CO2[c] + 2 NH3[c]
  - ADP[c] + Phosphoenolpyruvate[c] => ATP[c] + Pyruvate[c]
checkBalance

- CO2[c] <=>

rawOutput A boolean value 'TRUE' or 'FALSE' if raw data should be returned instead of computed values.

Author(s)
Daniel Camilo Osorio <dcosoriho@tamu.edu>

Examples

# Loading a set of stoichiometric reactions
glycolysis <- read.csv(system.file("extdata/glycolysisModel.csv", package = "minval"), sep = "\t")

# Characterizing the reactions
characterizeReactions(reactionList = glycolysis$REACTION)

checkBalance

Evaluate the mass or charge balance for a set of stoichiometric reactions

Description
For a given set of stoichiometric reactions, this function evaluates the mass or charge balance using a reference data. The checkBalance function returns a boolean value 'TRUE' if the reaction is balanced. One of 'mFormula', 'mWeight' or 'mCharge' arguments must be given.

Usage

checkBalance(reactionList, referenceData, ids, mFormula = NULL, mWeight = NULL, mCharge = NULL, woCompartment = TRUE)

Arguments

reactionList A set of stoichiometric reaction with the following characteristics:
- Arrows symbols must be given in the form '=>' or '<=>'
- Inverse arrow symbols '<=' or other types as: '<=>', '<=>', '<=>' will not be parsed and will lead to errors.
- Arrow symbols and plus signs (+) must be surrounded by a space character
- Stoichiometric coefficients must be surrounded by a space character and not by parentheses.
- Each metabolite must have only one stoichiometric coefficient, substituents must be joined to metabolite name by a hyphen (-) symbol.
- Exchange reactions have only one metabolite before arrow symbol
- Compartments must be given between square brackets ([compartment]) joined at the end of metabolite name

Some examples of valid stoichiometric reactions are:
compartments

- \( \text{H}_2\text{O} + \text{Urea-1-Carboxylate} \rightleftharpoons 2\text{CO}_2 + 2\text{NH}_3 \)
- \( \text{ADP} + \text{Phosphoenolpyruvate} \rightarrow \text{ATP} + \text{Pyruvate} \)
- \( \text{CO}_2 \rightarrow \)

referenceData
A chemical table containing data to evaluate the balance

ids
A mandatory ID of metabolite names column in the referenceData

mFormula
An optional ID of molecular formula column in the referenceData

mWeight
An optional ID of molecular weight column in the referenceData

mCharge
An optional ID of net charge column in the referenceData

woCompartment
A boolean value 'TRUE' or 'FALSE' to indicate if compartment label should be removed of stoichiometric reactions

Value
This function returns a boolean value 'TRUE' if reaction is balanced.

Author(s)
Daniel Camilo Osorio <dcosorioh@tamu.edu>

Examples

```r
# Loading a set of stoichiometric reactions
glycolysis <- read.csv(system.file("extdata/glycolysisModel.csv", package = "minval"), sep='\t')

# Loading external chemical information
chemicalData <- read.csv2(system.file("extdata", "chemData.csv", package = "minval"))
head(chemicalData)

# Evaluating mass balance
checkBalance(
  reactionList = glycolysis$REACTION,
  referenceData = chemicalData,
  ids = "NAME",
  mFormula = "FORMULA"
)
```

compartments
Extract the compartments associated to metabolites of a set of stoichiometric reactions.

Description
For a given set of stoichiometric reactions, this function identifies the compartments associated to each involved metabolite and return a vector with the list of unique compartments identified.
Usage

compartments(reactionList, uniques = TRUE)

Arguments

reactionList  A set of stoichiometric reaction with the following characteristics:
  • Arrows symbols must be given in the form ‘=>’ or ‘<=’
  • Inverse arrow symbols ‘<=’ or other types as: ‘-->’, ‘<=>’, ‘->’ will not be parsed and will lead to errors.
  • Arrow symbols and plus signs (+) must be surrounded by a space character
  • Stoichiometric coefficients must be surrounded by a space character and not by parentheses.
  • Each metabolite must have only one stoichiometric coefficient, substituents must be joined to metabolite name by a hyphen (-) symbol.
  • Exchange reactions have only one metabolite before arrow symbol
  • Compartments must be given between square brackets ([compartment]) joined at the end of metabolite name

Some examples of valid stoichiometric reactions are:
  • H2O[c] + Urea-1-Carboxylate[c] <= 2 CO2[c] + 2 NH3[c]
  • ADP[c] + Phosphoenolpyruvate[c] => ATP[c] + Pyruvate[c]
  • CO2[c] <=

uniques  A boolean value ‘TRUE’ or ‘FALSE’ if uniques must be returned

Value

A vector with the list of compartments identified for the metabolites of a set of stoichiometric reactions.

Author(s)

Daniel Camilo Osorio <dcosorioh@tamu.edu>

Examples

# Loading a set of stoichiometric reactions
glycolysis <- read.csv(system.file("extdata/glycolysisModel.csv",package = "minval"), sep = '\t')

# Extract unique compartments
compartments(reactionList = glycolysis$REACTION)

# Extract all compartments
compartments(reactionList = glycolysis$REACTION, unique = FALSE)

# Extract compartments of metabolites
compartments(reactionList = "H2O[e]")
downloadChEBI  

**Download the ChEBI database**

**Description**

This function downloads the compounds, formulas, masses and charges from the selected release of the ChEBI database. The ChEBI database (Chemical Entities of Biological Interest), is a database and ontology of molecular entities focused on 'small' chemical compounds.

**Usage**

```r
downloadChEBI(release = "latest", woAssociations = FALSE)
```

**Arguments**

- `release` A character string with the release number of the ChEBI database version to be downloaded, by default the 'latest' release is downloaded.
- `woAssociations` A logical value 'TRUE' or 'FALSE' if a light version of the ChEBI database without associations should be returned.

**Value**

A data.frame with the following data associated to the ChEBI compounds:

- 'ID': The unique identifier
- 'CheBI': The name recommended for use in biological databases
- 'KEGG': The associated name(s) in the KEGG database
- 'IUPAC': The name(s) generated according to recommendations of IUPAC
- 'MetaCyc': The associated name(s) in the MetaCyc database
- 'ChEMBL': The associated name(s) in the ChEMBL database
- 'FORMULA': The molecular formula
- 'MASS': The molecular mass
- 'MONOISOTOPIC': The molecular monoisotopic mass
- 'CHARGE': The molecular net charge

If `woAssociations` is 'TRUE' a data.frame with the following data is returned:

- 'NAME': The name(s) associated in several biological databases
- 'FORMULA': The molecular formula
- 'MASS': The molecular mass
- 'MONOISOTOPIC': The molecular monoisotopic mass
- 'CHARGE': The molecular net charge


metabolites

Author(s)
Daniel Camilo Osorio <dcosorioh@tamu.edu>

See Also
The ChEBI database webpage: https://www.ebi.ac.uk/chebi/

Examples

```r
# Not run:
# Download ChEBI database with associations
ChEBI <- downloadChEBI(release = '142')

# Download ChEBI database without associations
ChEBI <- downloadChEBI(release = '142', woAssociations = TRUE)
```

metabolites

Identify the list of metabolites for a set of stoichiometric reactions

Description
This function identifies the list of metabolites for a set of stoichiometric reactions. If 'woCompartment' is 'TRUE' compartment label is removed. If 'uniques' is 'TRUE', list of uniques is returned.

Usage

```r
metabolites(reactionList, woCompartment = FALSE, uniques = TRUE)
```

Arguments

- `reactionList` A set of stoichiometric reaction with the following characteristics:
  - Arrows symbols must be given in the form '&gt;' or '&lt;='
  - Inverse arrow symbols '&lt;' or other types as: '&lt;&gt;', '&lt;=' or '&gt;' will not be parsed and will lead to errors.
  - Arrow symbols and plus signs (+) must be surrounded by a space character
  - Stoichiometric coefficients must be surrounded by a space character and not by parentheses.
  - Each metabolite must have only one stoichiometric coefficient, substituents must be joined to metabolite name by a hyphen (-) symbol.
  - Exchange reactions have only one metabolite before arrow symbol
  - Compartments must be given between square brackets ([compartment]) joined at the end of metabolite name

Some examples of valid stoichiometric reactions are:

- `H2O[c] + Urea-1-Carboxylate[c] &lt;&gt; 2 CO2[c] + 2 NH3[c]`
orphanMetabolites

- ADP[c] + Phosphoenolpyruvate[c] => ATP[c] + Pyruvate[c]
- CO2[c] <=>

woCompartment A boolean value 'TRUE' or 'FALSE' to indicate if compartment label should be removed
uniques A boolean value 'TRUE' or 'FALSE' to indicate if uniques must be returned

Value
A list of metabolites for a set of stoichiometric reactions

Author(s)
Daniel Camilo Osorio <dcosorioh@tamu.edu>

Examples

# Extract metabolites of a stoichiometric reaction
metabolites(reactionList = "ADP[c] + Phosphoenolpyruvate[c] => ATP[c] + Pyruvate[c]"

# Loading a set of stoichiometric reactions
glycolysis <- read.csv(system.file("extdata/glycolysisModel.csv",package = "minval"), sep='\t')

# Extract unique metabolites
metabolites(reactionList = glycolysis$REACTION)

# Extract unique metabolites without compartments
metabolites(reactionList = glycolysis$REACTION, woCompartment = TRUE)

# Extract all metabolites
metabolites(reactionList = glycolysis$REACTION, uniques = FALSE)

Description
This function identifies the orphan metabolites (metabolites not produced or not consumed in any other reaction or just involved in one reaction) for a set of stoichiometric reactions.

Usage

orphanMetabolites(reactionList, actingAs = NULL, byCompartment = FALSE)
orphanReactants(reactionList, byCompartment = FALSE)
orphanProducts(reactionList, byCompartment = FALSE)
Arguments

**reactionList** A set of stoichiometric reaction with the following characteristics:

- Arrows symbols must be given in the form '=>' or '<='
- Inverse arrow symbols '<=' or other types as: '=>', '<=>', '->' will not be parsed and will lead to errors.
- Arrow symbols and plus signs (+) must be surrounded by a space character
- Stoichiometric coefficients must be surrounded by a space character and not by parentheses.
- Each metabolite must have only one stoichiometric coefficient, substituents must be joined to metabolite name by a hyphen (-) symbol.
- Exchange reactions have only one metabolite before arrow symbol
- Compartments must be given between square brackets ([compartment]) joined at the end of metabolite name

Some examples of valid stoichiometric reactions are:

- `H2O[c] + Urea-1-Carboxylate[c] <=> 2 CO2[c] + 2 NH3[c]`
- `ADP[c] + Phosphoenolpyruvate[c] => ATP[c] + Pyruvate[c]`
- `CO2[c] <=`

**actingAs** A text string that specifies the type of metabolite to be returned; only 'reactant' and 'product' are supported.

**byCompartment** A boolean value 'TRUE' or 'FALSE' to indicate if orphan reactants should be reported by compartment

Value

If byCompartment == FALSE, a vector with orphan reactants is returned, in opposite case a list is returned. If actingAs == 'reactant', metabolites not produced in any other reaction or just are involved in one reaction are returned. If actingAs == 'products', metabolites not consumed in any other reaction or just are involved in one reaction are returned.

Functions

- **orphanReactants**: Identify the orphan reactants of a set of stoichiometric reactions
- **orphanProducts**: Identify the orphan products of a set of stoichiometric reactions

Author(s)

Daniel Camilo Osorio <dcosorioh@tamu.edu>

Examples

```r
# Loading a set of stoichiometric reactions
glycolysis <- read.csv(system.file("extdata/glycolysisModel.csv", package = "minval"), sep="\t")

# Identify orphan metabolites
orphanMetabolites(reactionList = glycolysis$REACTION)
```
# Identify orphan reactants
orphanReactants(reactionList = glycolysis$REACTION)

# Identify orphan products
orphanProducts(reactionList = glycolysis$REACTION)

# Identify orphan metabolites by compartment
orphanMetabolites(reactionList = glycolysis$REACTION, byCompartment = TRUE)

---

**products**

*Identify the products of a stoichiometric reaction*

**Description**

This function identifies the products for a set of stoichiometric reactions.

**Usage**

`products(reactionList)`

**Arguments**

- `reactionList` A set of stoichiometric reaction with the following characteristics:
  - Arrows symbols must be given in the form `=>' or `<=>`
  - Inverse arrow symbols '<=' or other types as: '->', '<=>', '-' will not be parsed and will lead to errors.
  - Arrow symbols and plus signs (+) must be surrounded by a space character
  - Stoichiometric coefficients must be surrounded by a space character and not by parentheses.
  - Each metabolite must have only one stoichiometric coefficient, substituents must be joined to metabolite name by a hyphen (-) symbol.
  - Exchange reactions have only one metabolite before arrow symbol
  - Compartments must be given between square brackets ([compartment]) joined at the end of metabolite name

Some examples of valid stoichiometric reactions are:

- `H2O[c] + Urea-1-Carboxylate[c] <==> 2 CO2[c] + 2 NH3[c]`
- `ADP[c] + Phosphoenolpyruvate[c] => ATP[c] + Pyruvate[c]`
- `CO2[c] <==>`

**Value**

A vector with the identified products in the reaction, or a list with the identified products in each reaction if a set of stoichiometric reactions was given.

**Author(s)**

Daniel Camilo Osorio <dcosorioh@tamu.edu>
Reactants

Identify the reactants of a stoichiometric reaction

Description

This function identifies the reactants for a set of stoichiometric reactions.

Usage

reactants(reactionList)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>reactionList</td>
<td>A set of stoichiometric reaction with the following characteristics:</td>
</tr>
<tr>
<td></td>
<td>• Arrows symbols must be given in the form '=&gt;' or '&lt;=&gt;'</td>
</tr>
<tr>
<td></td>
<td>• Inverse arrow symbols '&lt;=' or other types as: '-&gt;&gt;', '&lt;=&gt;', '-&gt;' will</td>
</tr>
<tr>
<td></td>
<td>not be parsed and will lead to errors.</td>
</tr>
<tr>
<td></td>
<td>• Arrow symbols and plus signs (+) must be surrounded by a space character</td>
</tr>
<tr>
<td></td>
<td>• Stoichiometric coefficients must be surrounded by a space character and</td>
</tr>
<tr>
<td></td>
<td>not by parentheses.</td>
</tr>
<tr>
<td></td>
<td>• Each metabolite must have only one stoichiometric coefficient, substituents</td>
</tr>
<tr>
<td></td>
<td>must be joined to metabolite name by a hyphen (-) symbol.</td>
</tr>
<tr>
<td></td>
<td>• Exchange reactions have only one metabolite before arrow symbol</td>
</tr>
<tr>
<td></td>
<td>• Compartments must be given between square brackets ([compartment]) joined</td>
</tr>
<tr>
<td></td>
<td>at the end of metabolite name.</td>
</tr>
</tbody>
</table>

Some examples of valid stoichiometric reactions are:

- H2O[c] + Urea-1-Carboxylate[c] <=> 2 CO2[c] + 2 NH3[c]
- ADP[c] + Phosphoenolpyruvate[c] => ATP[c] + Pyruvate[c]
- CO2[c] <=>

Value

A vector with the identified reactants in the reaction, or a list with the identified reactants in each reaction if a set of stoichiometric reactions was given.

Author(s)

Daniel Camilo Osorio <dcosorioh@tamu.edu>
stoichiometricMatrix  

Build the stoichiometric matrix for a set of stoichiometric reactions

Description

A set of stoichiometric reactions is often represented in a more compact form called the stoichiometry matrix. If a metabolic network has $n$ reactions and $m$ participating metabolites, then the stoichiometry matrix will have correspondingly $m$ rows and $n$ columns. Values in the stoichiometric matrix represent the metabolites coefficients in each reaction.

Usage

stoichiometricMatrix(reactionList)

Arguments

reactionList  
A set of stoichiometric reaction with the following characteristics:

- Arrows symbols must be given in the form `=>` or `<=`
- Inverse arrow symbols `<=` or other types as: `-->`, `<==>`, `->` will not be parsed and will lead to errors.
- Arrow symbols and plus signs (+) must be surrounded by a space character
- Stoichiometric coefficients must be surrounded by a space character and not by parentheses.
- Each metabolite must have only one stoichiometric coefficient, substituents must be joined to metabolite name by a hyphen (-) symbol.
- Exchange reactions have only one metabolite before arrow symbol
- Compartments must be given between square brackets ([compartment]) joined at the end of metabolite name

Some examples of valid stoichiometric reactions are:

- $\text{H}_2\text{O}[c] + \text{Urea-1-Carboxylate}[c] \Rightarrow 2 \text{CO}_2[c] + 2 \text{NH}_3[c]$
- $\text{ADP}[c] + \text{Phosphoenolpyruvate}[c] \Rightarrow \text{ATP}[c] + \text{Pyruvate}[c]$
- $\text{CO}_2[c] \Leftrightarrow$

Value

The stoichiometric matrix for a given set of stoichiometric reactions

Author(s)

Daniel Camilo Osorio <dcosori@tamu.edu>
validateSyntax

Examples

# Loading a set of stoichiometric reactions
glycolysis <- read.csv(system.file("extdata/glycolysisModel.csv", package = "minval"), sep='\t')

# Building the stoichiometric matrix
stoichiometricMatrix(reactionList = glycolysis$REACTION)

validateSyntax Evaluate if a stoichiometric reaction has a valid syntax

Description

For a set of given stoichiometric reactions, this function makes the following syntactic evaluations for each reaction:

- Evaluates if the reaction contain more than one coefficient by metabolite
- Evaluates if the reaction contain metabolite coefficients between parenthesis
- Evaluates if the reaction contain arrow symbol between spaces
- Evaluates if the reaction contain not allowed arrow symbols
- Evaluates if the reaction contain metabolites name separated by a plus symbol between spaces
- Evaluates if the reaction contain substituents separated of the metabolite names

Usage

validateSyntax(reactionList)

Arguments

reactionList A set of stoichiometric reaction with the following characteristics:
- Arrows symbols must be given in the form ‘=>’ or ‘<=’
- Inverse arrow symbols ‘<=’ or other types as: ‘<=>’, ‘<>=>’, ‘<=>’ will not be parsed and will lead to errors.
- Arrow symbols and plus signs (+) must be surrounded by a space character
- Stoichiometric coefficients must be surrounded by a space character and not by parentheses.
- Each metabolite must have only one stoichiometric coefficient, substituents must be joined to metabolite name by a hyphen (-) symbol.
- Exchange reactions have only one metabolite before arrow symbol
- Compartments must be given between square brackets ([compartment]) joined at the end of metabolite name

Some examples of valid stoichiometric reactions are:
- H2O[c] + Urea-1-Carboxylate[c] <= 2 CO2[c] + 2 NH3[c]
- ADP[c] + Phosphoenolpyruvate[c] => ATP[c] + Pyruvate[c]
- CO2[c] <=
writeSBMLmod

Value

A boolean value ‘TRUE’ if reaction has a valid syntax.

Author(s)

Daniel Camilo Osorio <dcosorioh@tamu.edu>

Examples

# Evaluate the syntax for a single reaction
validateSyntax(reactionList = "ADP[c] + Phosphoenolpyruvate[c] => ATP[c] + Pyruvate[c]"")

# Loading a set of stoichiometric reactions
glycolysis <- read.csv(system.file("extdata/glycolysisModel.csv",package = "minval"), sep = '\t')

# Evaluating the syntax for a set of stoichiometric reactions
validateSyntax(reactionList = glycolysis$REACTION)

writeSBMLmod Write a model in SBML format

Description

This function converts a data.frame or a modelOrg object to a valid SBML file. The Systems Biology Markup Language (SBML) is a representation format, based on XML, for communicating and storing computational models of biological processes.

Usage

writeSBMLmod(modelData, modelID = "model", outputFile, boundary = "b")

Arguments

modelData A modelOrg or a data.frame object. If a data.frame is given, it must contain following mandatory colnames:

• 'ID': A list of single character strings containing the reaction abbreviations, Entries in the field abbreviation are used as reaction ids, so they must be unique.

• 'REACTION': A set of stoichiometric reaction with the following characteristics:
  – Arrows symbols must be given in the form '=>' or '<='
  – Inverse arrow symbols '<=' or other types as: '->', '<=>', '->' will not be parsed and will lead to errors.
  – Arrow symbols and plus signs (+) must be surrounded by a space character
  – Stoichiometric coefficients must be surrounded by a space character and not by parentheses.
Each metabolite must have only one stoichiometric coefficient, substituents must be joined to metabolite name by a hyphen (\(\sim\)) symbol.

Exchange reactions must have only one metabolite before arrow symbol.

Compartments must be given between square brackets ([compartment]) joined at the end of metabolite name.

Some examples of valid stoichiometric reactions are:

- \(\text{H}_2\text{O}[c] + \text{Urea-1-Carboxylate}[c] \leftrightarrow 2 \text{CO}_2[c] + 2 \text{NH}_3[c]\)
- \(\text{ADP}[c] + \text{Phosphoenolpyruvate}[c] \rightarrow \text{ATP}[c] + \text{Pyruvate}[c]\)
- \(\text{CO}_2[c] \leftrightarrow\)

- 'GPR': A set of genes joined by boolean operators as AND or OR, rules may be nested by parenthesis. (optional: column can be empty).
- 'LOWER.BOUND': A list of numeric values containing the lower bounds of the reaction rates. If not set, zero is used for an irreversible reaction and -1000 for a reversible reaction. (optional: column can be empty).
- 'UPPER.BOUND': A list of numeric values containing the upper bounds of the reaction rates. If not set, 1000 is used by default. (optional: column can be empty).
- 'OBJECTIVE': A list of numeric values containing objective values (-1, 0 or 1) for each reaction (optional: column can be empty).

Value

A valid SBML file

Author(s)

Daniel Camilo Osorio <dcosorioh@tamu.edu>

Examples

```r
# Loading a metabolic model
glycolysis <- read.csv(system.file("extdata/glycolysisModel.csv",package = "minval"), sep='\t')

# Writing a model in SBML format
writeSBMLmod(modelData = glycolysis,modelID = "Glycolysis",outputFile = "glycolysis.xml")
```

Not run:

```
# Writing a modelOrg object in a SBML format
# Loading the sybil R package
library(sybil)

# Loading the data
data("Ec_core")
```
writeTSVmod

Write a model in TSV format for the 'sybil' R package

Description

This function converts a data.frame or a modelOrg model to TSV format for the 'sybil' R package. TSV format require three '.TSV' output files ('_react.tsv', '_met.tsv', '_desc.tsv').

Usage

writeTSVmod(modelData, modelID = "model", outputFILE, boundary = "b")

Arguments

modelData A modelOrg or a data.frame object. If a data.frame is given, it must contain following mandatory colnames:
- 'ID': A list of single character strings containing the reaction abbreviations, Entries in the field abbreviation are used as reaction ids, so they must be unique.
- 'REACTION': A set of stoichiometric reaction with the following characteristics:
  - Arrows symbols must be given in the form '=>' or '<='
  - Inverse arrow symbols '<=' or other types as: '->', '<=->', '<->' will not be parsed and will lead to errors.
  - Arrow symbols and plus signs (+) must be surrounded by a space character
  - Stoichiometric coefficients must be surrounded by a space character and not by parentheses.
  - Each metabolite must have only one stoichiometric coefficient, substituents must be joined to metabolite name by a hyphen (-) symbol.
  - Exchange reactions must have only one metabolite before arrow symbol
  - Compartments must be given between square brackets ([compartment]) joined at the end of metabolite name

Some examples of valid stoichiometric reactions are:
- H2O[c] + Urea-1-Carboxylate[c] <=> 2 CO2[c] + 2 NH3[c]
- ADP[c] + Phosphoenolpyruvate[c] => ATP[c] + Pyruvate[c]
- CO2[c] <=>
modelID A single character string giving the modelID
outputFile A writable path for the three `.TSV' output files.
boundary A single character string specifying the compartment to be used as boundary

Value
A set of three `.TSV' files in a valid format to the 'sybil' R package.

Author(s)
Daniel Camilo Osorio <dcosorioh@tamu.edu>

Examples
```r
#' # Loading a metabolic model
glycolysis <- read.csv(system.file("extdata/glycolysisModel.csv", package = "minval"), sep='\t')

# Writing a model in TSV format
writeTSVmod(modelData = glycolysis, modelID = "Glycolysis", outputFile = "glycolysis")

## Not run:
## Writing a modelOrg object in a SBML format
## Loading the sybil R package
library(sybil)

## Loading the data
data("Ec_core")

## Writing the modelOrg object in a SBML format
writeTSVmod(modelData = Ec_core, modelID = "E.coli", outputFile = "eColi")

## End(Not run)
```
Index

characterizeReactions, 2
checkBalance, 3
compartments, 4
downloadChEBI, 6
metabolites, 7
orphanMetabolites, 8
orphanProducts (orphanMetabolites), 8
orphanReactants (orphanMetabolites), 8
products, 10
reactants, 11
stoichiometricMatrix, 12
validateSyntax, 13
writeSbsonml, 14
writeTSVmod, 16