Package ‘epanetReader’

May 26, 2017

Type Package
Title Read Epanet Files into R
Version 0.5.1
Date 2017-05-26
Author Bradley J. Eck
Maintainer Bradley Eck <brad@bradeck.net>
Depends R (>= 3.0.0), graphics, utils
Suggests testthat, Kmisc (>= 0.5.0)
Description Reads water network simulation data in 'Epanet' text-based '.inp' and '.rpt' formats into R. Also reads results from 'Epanet-msx'. Provides basic summary information and plots. The README file has a quick introduction. See <http://www2.epa.gov/water-research/epanet> for more information on the 'Epanet' software for modeling hydraulic and water quality behavior of water piping systems.
License MIT + file LICENSE
LazyData true
RoxygenNote 6.0.1
NeedsCompilation no
Repository CRAN
Date/Publication 2017-05-26 13:30:25 UTC

R topics documented:

  binBreaker ................................................. 2
epanetDefaultOptions ..................................... 3
epanetReader .............................................. 3
expandedLinkTable ....................................... 4
Net1 .......................................................... 4
Net1rpt ..................................................... 5
plot.epanet.inp .......................................... 5
plot.epanet.rpt .......................................... 6
binBreaker

Description

Generate break points for use with cut() and range labels based on sample max and min

Usage

binBreaker(x, nbin)

Arguments

x vector to find cuts for
nbin number of bins

Details

Helpful in making labels use the actual max and min rather than the +/- 1

Value

list with elements Breaks and Labels
epanetDefaultOptions  Epanet Default Options

Description

A list of Epanet’s default options

Usage

epanetDefaultOptions()

Details

Provides a named list in the form of OPTION = default_value where the values are taken from pages 152-154 of the manual.

References


Examples

epanetDefaultOptions()

epanetReader  Read text files from Epanet into R

Description

epanetReader is a package for reading water network information and simulation results in Epanet’s .inp and .rpt files into R. See functions read.inp, read.rpt, and read.msxrpt for some examples.

Author(s)

Bradley J. Eck
expandedLinkTable  

Expanded Link Table

Description
Create an expandedLinkTable object by adding node coordinates to a data frame of pipes, pumps, or valves.

Usage
expandedLinkTable(Links, Coordinates)

Arguments
- Links: data frame of Pipes, Pumps or Valves of from epanet.inp
- Coordinates: table of epanet.inp

Value
an expandedLinkTable object

Examples
x <- expandedLinkTable(Net1$Pipes, Net1$Coordinates)
print(x)
plot(x)

---

Net1  

Epanet’s Net1 Example

Description
A dataset created by reading the Net1.inp file distributed with Epanet using this package’s read.inp() function.

Usage
Net1

Format
An object of class epanet.inp created by read.inp.

Source
http://www.epa.gov/sites/production/files/2014-06/en2setup_0.exe
**Net1rpt**

**Epanet’s Net1 Example**

**Description**

A dataset created by reading the Net1.inp file distributed with Epanet using this package’s read.inp() function.

**Usage**

Net1rpt

**Format**

An object of class epanet.rpt created by read.inp.

**plot.epanet.inp**

*Plot Method for epanet.inp*

**Description**

Make a plot of the network using base graphics

**Usage**

```r
## S3 method for class 'epanet.inp'
plot(x, plot.junctions = TRUE,
     legend.locn = "topright", plot.labels = FALSE, link.lwd = 3,
     link.col = "black", ...)
```

**Arguments**

- `x` object of class epanet.inp
- `plot.junctions` logical indicating whether to plot junctions
- `legend.locn` character string passed to legend() specifying the location of the legend on the plot
- `plot.labels` logical indicating whether to plot the labels using text()
- `link.lwd` value of lwd passed to segments()
- `link.col` value of col passed to segments()
- `...` other arguments passed to plot()
Details

Implements the generic plot function for S3 objects of class epanet.inp. The plot is built from base graphics by creating a blank plot and then calling the helper functions plotInpLinks(), plotInpNodes(), plotElementsLegend().

Examples

plot(Net1)

Description

Plots simulation results for a single time step in map form

Usage

```r
## S3 method for class 'epanet.rpt'
plot(x, inp, Timestep = "0:00:00", juncQty = "Demand",
     linkQty = "Velocity", legend1.locn = "topright",
     legend2.locn = "topleft", ...)
```

Arguments

- `x`: epanet.rpt object
- `inp`: epanet.inp object associated with `x`
- `Timestep`: string indicating the time to plot
- `juncQty`: string specifying which column of `x$nodeResults` (Demand, Head, Pressure, Chlorine, etc.) to show by circle size at network junctions or NULL if no quantity should be plotted at the junctions
- `linkQty`: string specifying which column of `x$linkResults` (Flow, Velocity, Headloss) to show by line width on network links or NULL if no quantity should be plotted for the links.
- `legend1.locn`: string passed to legend() for placing legend of network elements
- `legend2.locn`: string passed to legend() for placing legend of junction and link quantities
- `...`: further arguments passed to plot

Details

juncQty plots and values for Junctions only; Tanks and Reservoirs are not included. In contrast, linkQty is scaled over all of the link types: Pipes, Pumps & Valves. These choices aim at a map showing demand at junctions and velocity in links.
Description

Plots a sparkline table of Epanet-msx results

Usage

```r
## S3 method for class 'epanetmsx.rpt'
plot(x, elementType = "Nodes", ...)
```

Arguments

- `x`: epanetmsx.rpt object
- `elementType`: character indicating whether results for "nodes" or links" should be plotted
- `...`: further arguments passed to plotSparklineTable

See Also

plotSparklineTable

plot.expandedLinkTable

Description

plot an expanded link table

Usage

```r
## S3 method for class 'expandedLinkTable'
plot(x, add = FALSE, label = FALSE,
     linewidths = 3, color = "black", ...)
```

Arguments

- `x`: object of type expandedLinkTable
- `add`: logical indicating whether to add to the currently active plot. add=FALSE creates a new plot.
- `label`: logical indicating if the links should be labeled at the mid points
- `linewidths`: passed to lwd argument in segments()
- `color`: passed to col argument in segments()
- `...`: further arguments passed to segments()
Details

An implementation of the generic plot function for expandedLinkTable objects. Links are drawn using segments(). Useful for building up network plots.

plot.sparklineTable

Description

Plot a sparkline

Usage

## S3 method for class 'sparkline'
plot(x, ...)

Arguments

x             sparkline object
...
  further arguments passed to plot.default

Details

Implementation of the generic plot function for a single sparkline object. The primarily used to build up plots of a sparklineTable

See Also

sparkline

plot.sparklineTable

Description

Plot Sparkline Table

Usage

## S3 method for class 'sparklineTable'
plot(x, ...)

Arguments

x             object of class sparklineTable
...
  further arguments passed to par
plotElementsLegend  

**Plot Legend of Network Elements**

**Description**
Add legend of network elements to the active plot

**Usage**
```r
plotElementsLegend(legend.locn)
```

**Arguments**
- `legend.locn` keyword for location of legend. See details of legend() function.

**Details**
Helper function for adding a legend to the active plot. Uses plot characters 16, 15, 8 and 25 for Tanks, Reservoirs, Pumps and Valves for compatibility with plotInpNodes()

**Examples**
```r
## make a new blank plot
plot(c(0,1), c(0,1), type = 'n')
## add the nodes, including junctions
plotElementsLegend('topright')
```

plotInpLinks  

**Plot .inp Links**

**Description**
Add lines for pipes, pumps and valves from an epanet.inp object to an existing plot

**Usage**
```r
plotInpLinks(x, lwd = 3, col = "black")
```

**Arguments**
- `x` epanet.inp object
- `lwd` width of lines
- `col` color of lines
plotInpNodes

Details

Helper function for building up a plot of the network by adding links to an existing plot.

Examples

```r
## make a new blank plot
plot( range(Net1$Coordinates$X), range(Net1$Coordinates$Y), type = 'n'
## add the links
plotInpLinks(Net1)
```

plotInpNodes  

Plot Node Elements

Description

Adds node elements from epanet.inp object to an existing plot

Usage

```r
plotInpNodes(x, plot.junctions)
```

Arguments

- `x` epanet.inp object
- `plot.junctions` logical indicating whether to plot junctions

Details

Helper function for building up a network plot. Tanks and Reservoirs are shown using plot characters (pch) 16 and 15. Junctions, if plotted, appear as pch = ".".

Examples

```r
## make a new blank plot
plot( range(Net1$Coordinates$X), range(Net1$Coordinates$Y), type = 'n'
## add the nodes, including junctions
plotInpNodes(Net1, TRUE )
```
plotSparklineTable  

Description

Generate a table of sparkline plots

Usage

plotSparklineTable(df, row.var, col.vars, xvar = NULL, xrange.labels = NULL)

Arguments

df  
data.frame of values to plot.
row.var  
variable for rows of the table
col.vars  
variables for columns of the table
xvar  
optional name of variable for horizontal axis of sparkline plots
xrange.labels  
optional vector of length 2 with labels for the first and last quantities plotted on x-axis, often a date and/or time

Details

Generates a table of 'sparkline' plots of data in df. Rows the table correspond to different values of row.var. The table's first column gives the value of row.var. The remaining columns contain sparkline plots for the values of col.vars. When xvar is not provided values are plotted against their index in the extracted vector. The starting and ending values are labeled. Uses layout() function to arrange plots.

References


See Also

yaletoolkit and sparkTable packages

Examples

plotSparklineTable( Orange, row.var = 'Tree', col.vars = c('age', 'circumference'))  
plotSparklineTable( Loblolly, row.var = 'Seed', col.vars = 'height')  
## specify the x variable if you have it, especially if it differs
plotSparklineTable(Theoph, row.var = 'Subject', col.vars = 'conc')  
## a warning is normally issued when the ranges of xvar differ
suppressWarnings( plotSparklineTable(Theoph, row.var = 'Subject', col.vars = 'conc', xvar = 'Time'))
print.summary.epanet.rpt

Print rpt summary

Description

The function prints a summary of simulation results contained in the rpt file.

Usage

```r
# S3 method for class 'summary.epanet.rpt'
print(x, ...)
```

Arguments

- `x`: a summary.epanet.rpt object
- `...`: further arguments passed to print

print.summary.epanetmsx.rpt

Print msx rpt summary

Description

The function prints a summary of multi-species simulation results contained in the report file.

Usage

```r
# S3 method for class 'summary.epanetmsx.rpt'
print(x, ...)
```

Arguments

- `x`: a summary.epanetmsx.rpt object
- `...`: further arguments passed to print
Description
Read an Epanet .inp file into R

Usage
read.inp(file)

Arguments
file the name of the file to read

Details
This function reads a text file in Epanet’s .inp format and returns an S3 object with entries for sections of the .inp file. Sections of the .inp file that are implemented appear in the Value section.

Fields for node or link ID are stored as characters not factors or integers. However, some fields are stored as factors to allow more informative summaries. Examples include valve type and pipe status.

Sections that are absent from the .inp file are NULL in the list.

Columns of data.frames use the headings exported by the Epanet GUI.

The [OPTIONS] section in the .inp file is used to update a list of Epanet’s default options. In this way if an option such as units is not specified by the .inp file, the units that would be used by default are provided.

In the [PATTERNS] and [CURVES] sections, integers used as names of list elements are backquoted according to the default behavior in R. So if the .inp file has a pattern ”1” this pattern will appear as element ‘1’ in the list that is returned. A warning is issued in this case.

Value
Returns an epanet.inp S3 object with elements of the following names and types corresponding to sections of the .inp file. Sections missing from the .inp file have a value of NULL.

<table>
<thead>
<tr>
<th>Title</th>
<th>character</th>
</tr>
</thead>
<tbody>
<tr>
<td>Junctions</td>
<td>data.frame</td>
</tr>
<tr>
<td>Tanks</td>
<td>data.frame</td>
</tr>
<tr>
<td>Reservoirs</td>
<td>data.frame</td>
</tr>
<tr>
<td>Pipes</td>
<td>data.frame</td>
</tr>
<tr>
<td>Pumps</td>
<td>data.frame</td>
</tr>
<tr>
<td>Valves</td>
<td>data.frame</td>
</tr>
<tr>
<td>Demands</td>
<td>data.frame</td>
</tr>
</tbody>
</table>
References


http://nepis.epa.gov/Adobe/PDF/P1007WWU.pdf

Examples

```r
# path to Net1.inp example file included with this package
inp <- file.path(find.package("epanetReader"), "extdata","Net1.inp")

#read the network file into R
n1 <- read.inp(inp)
summary(n1)
names(n1)
summary(n1$Junctions)
summary(n1$Pipes)
plot(n1)
```
Description

reads an Epanet-msx .rpt file into R

Usage

read.msxrpt(file)

Arguments

file the name of the file to read

Details

Specify the needed outputs from an Epanet-msx simulation in the [REPORT] section of the .msx file to create reports for reading with this function.

The function returns an S3 object (list) with a data.frame for node results and data.frame for link results. These data.frames contain results from all the time periods to facilitate time series plots.

Value

Returns an epanetmsx.rpt S3 object.

print(nodeResults) data.frame
print(linkResults) data.frame

References


Examples

# path to example file included with this package
msr <- file.path( find.package("epanetReader"), "extdata","example.rpt")

# read the results into R
x <- read.msxrpt(msr)
names(x)
summary(x)
plot(x)
**Description**

reads an Epanet .rpt file into R

**Usage**

```r
read.rpt(file)
```

**Arguments**

- `file` the name of the file to read

**Details**

Add lines "Page 0", "Links All" and "Nodes All" to the [REPORT] section of the .inp file so that the Epanet simulation outputs information to read with this function.

In contrast to the treatment of .inp files, data from .rpt files is stored using a slightly different structure than the .rpt file. The function returns an object (list) with a data.frame for node results and data.frame for link results. These two data frames contain results from all the time periods to facilitate time series plots.

**Value**

Returns an epanet.rpt S3 object with two data.frame elements.

- `nodeResults` data.frame
- `linkResults` data.frame

**References**


http://nepis.epa.gov/Adobe/PDF/P1007WWU.pdf

**Examples**

```r
# path to Net1.rpt example file included with this package
rpt <- file.path( find.package("epanetReader"), "extdata","Net1.rpt")
nlr <- read.rpt(rpt)
summary(nlr)
names(nlr)

#Results for a chosen time period can be retrieved using the subset function.
subset(nlr$nodeResults, Timestamp == "0:00:00")
```
read_lines_wrapper

Description

Wrapper function for different implementations of readlines functions

Usage

read_lines_wrapper(file)

Arguments

file the name of the file to read

Details

calls Kmisc::readlines if available and base::readLines otherwise

Value

character vector where each entry corresponds to a line in the file.

sparkline Sparkline

Description

Create sparkline object by extracting from a data frame

Usage

sparkline(df, id.var, ID, yvar, xvar)
Arguments

df  data.frame from which data for the sparkline is extracted
idNvar  variable in df with IDs
ID  value in id.var on which to extract
yvar  name of variable for the y values in the sparkline
xvar  optional name of variable for horizontal axis of sparkline plots

Details

Creates an object with info for a single sparkline by extracting from a data.frame. The function
works on data.frames with one column of ID variables and possibly several columns of other vari-
bles. The main use is as a helper function for building up a sparklineTable.

Examples

```r
## look at the names in the built-in data set Theoph
names(Theoph)
## make sparkline object for the concentration over time in subject 2
sl <- sparkline(df= Theoph, id.var = 'Subject', ID = 2, yvar='conc', xvar = 'Time')
plot(sl)
```

Description

Create S3 object of data for table of sparklines

Usage

```r
sparklineTable(df, rowNvar, colNvars, xvar = NULL, xrangeNlabels = NULL)
```

Arguments

- df  data.frame of values to plot.
- rowNvar  variable for rows of the table
- colNvars  variables for columns of the table
- xvar  optional name of variable for horizontal axis of sparkline plots
- xrangeNlabels  optional vector of length 2 with labels for the first and last quantities plotted on
  x-axis, often a date and/or time

See Also

plotSparklineTable
**summary.epanet.inp**  
*Summary Method for epanet.inp*

---

**Description**

Summarizes the network by printing the Title of the network and the number of each type of elements.

**Usage**

```r
## S3 method for class 'epanet.inp'
summary(object, ...)
```

**Arguments**

- `object` of class `epanet.inp`
- `...` further arguments

---

**summary.epanet.rpt**  
*Summary of Epanet Simulation Results*

---

**Description**

Provides a basic summary of simulation results

**Usage**

```r
## S3 method for class 'epanet.rpt'
summary(object, ...)
```

**Arguments**

- `object` of `epanet.rpt` class
- `...` further arguments passed to `summary()`

**Details**

Summary of pipe results shows positive and negative values of flow but only positive values of velocity as in the `rpt` file.
Summary of Epanet-msx Simulation Results

Description
Provides a basic summary of simulation results

Usage
## S3 method for class 'epanetmsx.rpt'
summary(object, ...)

Arguments
- object: of epanetmsx.rpt class
- ...: further arguments passed to summary()

Write .inp file

Description
Write an epanet.inp object to a file

Usage
write.inp(x, file)

Arguments
- x: epanet.inp object to write
- file: the name of the file where object is written

Details
Writes an epanet.inp object to a file suitable for simulation with EPANET.

Value
nothing

Examples
write.inp(Net1, "Net1-fromR.inp")
all.equal(Net1, n1)
Index

binBreaker, 2
epanet.inp (read.inp), 13
epanetDefaultOptions, 3
epanetmsx.rpt (read.msxrpt), 15
epanetReader, 3
epanetReader-package (epanetReader), 3
expandedLinkTable, 4
Net1, 4
Net1rpt, 5
plot.epanet.inp, 5
plot.epanet.rpt, 6
plot.epanetmsx.rpt, 7
plot.expandedLinkTable, 7
plot.sparkline, 8
plot.sparklineTable, 8
plotElementsLegend, 9
plotInpLinks, 9
plotInpNodes, 10
plotSparklineTable, 11
print.summary.epanet.rpt, 12
print.summary.epanetmsx.rpt, 12
read.inp, 3-5, 13
read.msxrpt, 3, 15
read.rpt, 3, 16
read_lines_wrapper, 17
sparkline, 17
sparklineTable, J8, 18
summary.epanet.inp, 19
summary.epanet.rpt, 19
summary.epanetmsx.rpt, 20
write.inp, 20