

Package ‘readmet’

July 23, 2025

Type Package
Title Read some less Popular Formats Used in Meteorology
Version 1.7.1
Date 2024-03-01
Author Clemens Druee, Universitaet Trier
Maintainer Clemens Druee <druee@uni-trier.de>
Description Contains tools for reading and writing data from or to files in the formats: akterm, dmna, Scintec Format-1, and Campbell Scientific TOA5.
License GPL-3
NeedsCompilation no
Repository CRAN
Date/Publication 2024-03-01 14:30:02 UTC

Contents

dmna.axes	2
dmna.grid	3
dmna.header	4
read.akterm	4
read.dmna	6
read.scintec1	7
read.toa5	8
readmet	9
scintec1.comments	10
scintec1.header	11
scintec1.nonprofile	12
scintec1.profile	13
scintec1.variables	14
write.arcgrid	15
write.dmna	16
Index	18

`dmna.axes`*read grid positions from a file in DMNA-format*

Description

Files in DMNA format contain gridded data. The header provides information on how the data are organized. This routine expands the positions of the grid planes along each dimension used in the file. The values are put out in columns x, y, z for each respective axis.

Usage

```
dmna.axes(file, debug = FALSE)
```

Arguments

<code>file</code>	name (and path if necessary) of the file to read
<code>debug</code>	if TRUE shwon debugging information

Value

returns a data.frame with the following columns:

<code>x</code>	grid positions along the x axis
<code>y</code>	grid positions along the y axis
<code>z</code>	grid level heights

Note

Prior to version 1.2.4, this function did not arrays with more than two dimensions, or multiple variables in files of more than one dimension. The number of dimensions is currently limited to three.

Author(s)

Clemens Druee

References

AUSTAL 3.1 model reference (by Ingenieurbuero Janicke)

Examples

```
ax <- dmna.axes(system.file("extdata", "example-grid.dmna", package="readmet"))
ax$x
```

dmna.grid	<i>read (horizontal) grid definition from a file in DMNA-format</i>
-----------	---

Description

Files in DMNA format contain gridded data. The header provides information on the grid definition. This function extracts the horizontal grid definition and provides a list that may be supplied directly to function write.arcgrid. DMNA is used for example by the German national dispersion model AUSTAL.

Usage

```
dmna.grid(file)
```

Arguments

file	name (and path if necessary) of the file to read
------	--

Value

returns a list of character. List entries are:

xlen	number of cells in x direction
ylen	number of cells in y direction
xll	easting of the lower left (i.e. southwest) corner
yll	northing of the lower left (i.e. southwest) corner
delta	grid node spacing

Note

This function was added in version 1.3.0.

Author(s)

Clemens Druce

References

AUSTAL 3.1 model reference (by Ingenieurbuero Janicke)

Examples

```
dmna.grid(system.file("extdata", "example-grid.dmna", package="readmet"))
```

dmna.header	<i>read header information from a file in DMNA-format</i>
-------------	---

Description

Files in DMNA format contain gridded data. The header provides information on how the data are organized and other user-specified meta data. DMNA is used for example by the German national dispersion model AUSTAL2000.

Usage

```
dmna.header(file)
```

Arguments

file	name (and path if necessary) of the file to read
------	--

Value

list(character)

Author(s)

Clemens Druce

References

AUSTAL2000 model reference (by Ingenieurbuero Janicke)

Examples

```
hdr <- dmna.header(system.file("extdata", "example-grid.dmna", package="readmet"))
hdr$idnt
```

read.akterm	<i>function to read contents of a file in AKTERM-format</i>
-------------	---

Description

reads weather data time series from a AKTERM format file that is used for example by the German national dispersion model AUSTAL

Usage

```
read.akterm(file)
```

Arguments

file name (and path if necessary) of the file to read

Value

returns a data.frame with the following columns:

1. "Time" is the time of observation as POSIXct.
2. "STA" is the WMO number of the observing station
3. "QDD" is the quality byte for DD
4. "QFF" is the quality byte for FF
5. "DD" is the wind direction clockwise in degrees from north
6. "FF" is the wind speed in m/s
7. "QQ1" is the quality byte for KM
8. "KM" is the Klug/Manier stability class
9. "QQ2" is the quality byte for HM
10. "HM" is unused
11. "QQ3" is the overall quality byte.

Note that DD and FF are always in degrees and m/s, independent from the value of the quality byte. Values where the quality byte is 9 are always NA. I.e. the quality byte is given for informational purposes only.

Note

Prior to version 1.2.4, this function did not arrays with more than two dimensions, or multiple variables in files of more than one dimension. Prior to version 1.5.0 timeseries were not implemented.

Author(s)

Clemens Druce

References

AUSTAL 3.1 model reference (by Ingenieurbuero Janicke)

Examples

```
series <- read.akterm(system.file("extdata", "example-series.akterm", package="readmet"))
plot(series$Time, series$FF, type="l")
```

read.dmna	<i>function to read contents of a file in DMNA-format</i>
-----------	---

Description

reads gridded one to two dimensional data from a DMNA format that used for example by the German national dispersion model AUSTAL

Usage

```
read.dmna(file, val=1, debug=FALSE)
```

Arguments

file	name (and path if necessary) of the file to read
val	name (number of name of the variable to be extracted. Only needed in case the file contains gridded data and more than one variable)
debug	if TRUE shwon debugging information

Value

returns an object whos type depends on the data containt in the file.

If the file contains a timeseries, a data.frame is returned that contains each variable in the file as a column. Column names are taken from the format description embedded in the file. If a column for time (name "te") is present, times are returned as POSIXct.

If the file contains gridded data an array of numeric is returned. The number of dimensions of the array is the lowest needed to hold the data. If the file contains mor than one variable, the function will return only one variable with each call. Choose the varibale by providing paramter val (name or number of variable).

Note

Prior to version 1.2.4, this function did not arrays with more than two dimensions, or multiple variables in files of more than one dimension. Prior to version 1.5.0 timeseries were not implemented.

Author(s)

Clemens Druee

References

AUSTAL 3.1 model reference (by Ingenieurbuero Janicke)

Examples

```
so2 <- read.dmna(system.file("extdata", "example-grid.dmna", package="readmet"))
ax <- dmna.axes(system.file("extdata", "example-grid.dmna", package="readmet"))
filled.contour(ax$x, ax$y, so2)

monitor <- read.dmna(system.file("extdata", "example-monitor.dmna", package="readmet"))
plot(monitor$te, monitor$X2, type="l")

series <- read.dmna(system.file("extdata", "example-series.dmna", package="readmet"))
plot(series$te, series$ua, type="l")
```

read.scintec1	<i>function to read the contents of a file in Scintec Format-1</i>
---------------	--

Description

reads output data (extension ".mnd") from Scintec atmospheric measurement systems into a list that contains the variable names as names and the values as either matrix or vector.

Usage

```
read.scintec1(files)
```

Arguments

files	array containing names (and path if necessary) of the files to read
-------	---

Value

list(different types)	
vector	length = n (number of times contained in the file) each vector contains a non-profile variable (see variable list) special variable name "time" contains the times as POSIXct
matrix	dim(n,m) where n is number of times contained in the file and m is the number of levels i.e. rows represents timeseries of at a fixed level, columns represent profiles at a fixed time

Author(s)

Clemens Druee

References

Scintec APRun software manual

Examples

```
mnd <- read.scintec1(system.file("extdata", "example.mnd", package="readmet"))
mnd$CT2
```

read.toa5

function to read contents of a file in TOA5-format

Description

reads the contents of a Campbell Scientific table-orientad ASCII format 5 (TOA5) into a data frame, columns are named after the TOA5 variable names

Usage

```
read.toa5(file)
```

Arguments

file name (and path if necessary) of the file to read

Value

data.frame

Author(s)

Clemens Druce

References

Campbell Scientific CR3000 data logger manual

Examples

```
dat <- read.toa5(system.file("extdata", "example.dat", package="readmet"))
dat$time <- as.POSIXct(dat$TIMESTAMP)
plot (dat$time, dat$AirTC_Avg)
```


scintec1.header	read the header of a file in Scintec Format-1 format
scintec1.nonprofiles	read the single-level data from a file in Scintec Format-1 format
scintec1.profile	read the multi-level (profile) variables from a Scintec Format-1 file
scintec1.variables	read the variable descriptions from a file in Scintec Format-1 format

Author(s)

Clemens Druée, Umweltmeteorologie, Universitaet Trier, Germany <druée@uni-trier.de>

scintec1.comments	<i>read the comment fields of a file in Scintec Format-1 format</i>
-------------------	---

Description

get comment entries from a Scintec atmospheric profiler data format "Format-1" (extension ".mnd")

Usage

```
scintec1.comments(file, header = list())
```

Arguments

file	name (and path if necessary) of the file to read
header	optionally, instead of reading the header from the file again, the output of a previous call to scintec1.header on the same file can be supplied via this argument.

Value

named list	each entry corresponds to one comment field
------------	---

Author(s)

Clemens Druée

References

Scintec APRun software manual

Examples

```
hdr <- read.scintec1(system.file("extdata", "example.mnd", package="readmet"))
hdr$`Serial Number`
```

scintec1.header	<i>read the header of a file in Scintec Format-1 format</i>
-----------------	---

Description

get timeseries of non-profile variables from a Scintec atmospheric profiler data format "Format-1" (extension ".mnd")

Usage

```
scintec1.header(file)
```

Arguments

file	name (and path if necessary) of the file to read
------	--

Value

named list; each entry corresponds to one header field:

starttime	starting time of the measurement as POSIXct object
filecount	running number of files produced during the current measurement
instrument	model type of the device
commentlines	lines of text containing the comment list
variables	lines of text containing the list of the variables in encoded form
heightlevels	number of height of levels where data are produced

Author(s)

Clemens Druce

References

Scintec APRun software manual

Examples

```
hdr <- read.scintec1(system.file("extdata", "example.mnd", package="readmet"))
hdr$instrument
```

scintec1.nonprofile *read the single-level variables from a file in Scintec Format-1 format*

Description

get timeseries of non-profile variables from a Scintec atmospheric profiler data format "Format-1" (extension ".mnd")

Usage

```
scintec1.nonprofile(file, header = list(), vars = list())
```

Arguments

file	name (and path if necessary) of the file to read
header	optionally, instead of reading the header from the file again, the output of a previous call to <code>scintec1.header</code> on the same file can be supplied via this argument.
vars	optionally, instead of reading the header from the file again, the output of a previous call to <code>scintec1.variables</code> on the same file can be supplied via this argument.

Value

list(vector); length(vector) = n (number of times contained in the file)
each vector contains a non-profile variable (see variable list) special variable name "time" contains the times as POSIXct

Author(s)

Clemens Druce

References

Scintec APRun software manual

Examples

```
dat <- scintec1.nonprofile(system.file("extdata", "example.mnd", package="readmet"))  
head(dat)
```

scintec1.profile	<i>read the multi-level (profile) variables from a Scintec Format-1 file</i>
------------------	--

Description

get profile variables from a Scintec atmospheric profiler data format "Format-1" (extension ".mnd")

Usage

```
scintec1.profile(file, header = list(), vars = list())
```

Arguments

file	name (and path if necessary) of the file to read
header	optionally, instead of reading the header from the file again, the output of a previous call to <code>scintec1.header</code> on the same file can be supplied via this argument.
vars	optionally, instead of reading the header from the file again, the output of a previous call to <code>scintec1.variables</code> on the same file can be supplied via this argument.

Value

`list(matrix); dim(matrix)=c(n,m)`
where n is number of times contained in the file and m is the number of levels
i.e. rows represents timeseries of at a fixed level, columns represent profiles at a fixed time

Author(s)

Clemens Druce

References

Scintec APRun software manual

Examples

```
dat <- scintec1.profile(system.file("extdata", "example.mnd", package="readmet"))  
dat
```

scintec1.variables	<i>read the variable descriptions from a file in Scintec Format-1 format</i>
--------------------	--

Description

get timeseries of non-profile variables from a Scintec atmospheric profiler data format "Format-1" (extension ".mnd")

Usage

```
scintec1.variables(file, header = list())
```

Arguments

file	name (and path if necessary) of the file to read
header	optionally, instead of reading the header from the file again, the output of a previous call to scintec1.header on the same file can be supplied via this argument.

Value

data.frame; each entry corresponds to one variable. The columns are:

label	Name of variable
symbol	short name; corresponds to list name in <code>scintec1.profile</code> , <code>scintec1.nonprofile</code> , and <code>read.scintec1</code>
unit	physical unit
type	code describing for example averaging, profile/nonprofile, measured, derived or assimilated. See Scintec Software Manual
error.mask	See Scintec Software Manual
gap.value	value in files representing missing values of this variable

Author(s)

Clemens Druce

References

Scintec APRun software manual

Examples

```
vars <- scintec1.variables(system.file("extdata", "example.mnd", package="readmet"))
head(vars)
```

write.arcgrid	<i>function to write data (read from DMNA file) into an ESRI ArcInfo gridded ASCII file</i>
---------------	---

Description

This function writes a 2D matrix into an ESRI ArcInfo gridded ASCII file that can be easily imported into most geographic information systems (GIS)

Usage

```
write.arcgrid(z, file, xlen, ylen, xll, yll, delta, grid, naval=-9999)
```

Arguments

z	2D matrix containing the data
file	name (including path, if needed) of the file to write to
xlen, ylen	number of data along x and y axis, resp.
xll, yll	position of the lower left (i.e. southwest) corner
delta	grid spacing
grid	a list containing grid parameters. Instead of providing xlen, ylen, xll, yll, and delta individually, containing these values may be provided. For example, such a list is returned when calling dmna.grid
naval	value written to file instead of NA and +/-Inf

Details

The standard plotting functions for R plot columns along the x axis and rows along the y axis. Hence the matrix is rotated 90 degrees left (compared to [write.table](#) or [write.csv](#)) to yield the same orientation of the data when plotted in R and in GIS.

Value

nothing.

Note

This function was added in version 1.3.0.

Author(s)

Clemens Druce

References

Wikipedia entry on Esri grid : https://en.wikipedia.org/wiki/Esri_grid

See Also

[read.dmna](#), [dmna.grid](#)

Examples

```
# read data and grid info
infile <- system.file("extdata", "example-grid.dmna", package="readmet")
so2 <- read.dmna(infile)
grid <- dmna.grid(infile)
# write file
write.arcgrid(so2, file = "myfile.grid", grid = grid)
# show head of file
writeLines(readLines("myfile.grid", n=7))
# delete file
file.remove("myfile.grid")
```

write.dmna

function to write data to a file in DMNA-format

Description

writes gridded on-e, two or three-dimensional data or timeseries to a DMNA format that is used for example by the German national dispersion model AUSTAL

Usage

```
write.dmna(filename, values, axes=NULL, name=NULL, types=NULL, vldf="V", debug=FALSE)
```

Arguments

filename	name (and path if necessary) of the file to write
values	matrix or list(matrix) or data.frame. Matrix or list(matrix) implies writing gridded data. list(matrix) must be named using the variable names. data.frame implies writing a timeseries and must contain a column named "te" containing time as POSIXct.
axes	data.frame. Required for gridded data. Must contain columns "x" and column "y" if values are two-dimensional and additionally "sk" (or "z") if data are three-dimensional. The spacing of all values in "x" and "y" must be identical.
name	character string. Variable name. Required if values is of class matrix. Is ignored else.
types	named list. Variable type for each variable. If 'values' is a list, names in 'types' must match names in 'values'. If 'values' is a matrix, a name in 'types' must match 'name'. Types are: • 'd': integer number format • 'f': floating point format (suitable for numbers 0.1 ... 99999.) • 'e': exponential format

	• 't': timestamp
vldf	character string. specifies, where values are located in the model grid. "V" denotes volume average, "P" volumne center point values, or "" volume edge values.
debug	boolean. ignored. Is kept for compatibility

Value

returns nothing

Note

Introduced in version 1.6.0.

Author(s)

Clemens Druce

References

AUSTAL 3.1 model reference (by Ingenieurbuero Janicke)

Examples

```
#create data and write file
len <- 25
val <- list(random = matrix(runif(len * len),nrow = len))
ax <- list(x = 1:len,
          y = 1:len)
write.dmna("myfile.dmna", val, ax)

# show first lines
writeLines(readLines('myfile.dmna', n=12))

# delete file
unlink("myfile.dmna")
```

Index

- * **AUSTAL2000**
 - dmna.header, 4
- * **AUSTAL**
 - dmna.axes, 2
 - dmna.grid, 3
 - read.akterm, 4
 - read.dmna, 6
 - write.dmna, 16
- * **ArcInfo**
 - write.arcgrid, 15
- * **DMNA**
 - dmna.axes, 2
 - dmna.grid, 3
 - dmna.header, 4
 - read.akterm, 4
 - read.dmna, 6
 - readmet, 9
 - write.dmna, 16
- * **ESRI**
 - write.arcgrid, 15
- * **IO**
 - dmna.axes, 2
 - dmna.grid, 3
 - dmna.header, 4
 - read.akterm, 4
 - read.dmna, 6
 - read.scintec1, 7
 - readmet, 9
 - scintec1.comments, 10
 - scintec1.header, 11
 - scintec1.nonprofile, 12
 - scintec1.profile, 13
 - scintec1.variables, 14
 - write.arcgrid, 15
 - write.dmna, 16
- * **Scintec**
 - read.scintec1, 7
 - readmet, 9
 - scintec1.comments, 10
 - scintec1.header, 11
 - scintec1.nonprofile, 12
 - scintec1.profile, 13
 - scintec1.variables, 14
- * **TAO5**
 - read.toa5, 8
- * **TOA5**
 - readmet, 9
- * **file**
 - dmna.axes, 2
 - dmna.grid, 3
 - dmna.header, 4
 - read.akterm, 4
 - read.dmna, 6
 - read.scintec1, 7
 - readmet, 9
 - scintec1.comments, 10
 - scintec1.header, 11
 - scintec1.nonprofile, 12
 - scintec1.profile, 13
 - scintec1.variables, 14
 - write.arcgrid, 15
 - write.dmna, 16
- * **format**
 - read.toa5, 8
- * **package**
 - readmet, 9
- * **read**
 - read.toa5, 8
- dmna.axes, 2
- dmna.grid, 3, 15, 16
- dmna.header, 4
- read.akterm, 4
- read.dmna, 6, 16
- read.scintec1, 7
- read.toa5, 8
- readmet, 9
- readmet-package (readmet), 9

scintec1.comments, [10](#)
scintec1.header, [10](#), [11](#), [12–14](#)
scintec1.nonprofile, [12](#)
scintec1.profile, [13](#)
scintec1.variables, [12](#), [13](#), [14](#)

write.arcgrid, [15](#)
write.csv, [15](#)
write.dmna, [16](#)
write.table, [15](#)