Package 'readmet'

March 1, 2024

R topics documented:

---

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

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

debug   if TRUE shwon debugging information

Value

returns a data.frame with the following columns:

x   grid positions along the x axis

y   grid positions along the y axis

z   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 *read (horizontal) grid definition from a file in DMNA-format*

Description

Files in DMNA format contain gridded data. The header provides information on the grid definition. This function extracts the horizontal grid definition ans 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 Druee

References

AUSTAL 3.1 model reference (by Ingenieurbuero Janicke)

Examples

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

---

dmna.header	*read header information from a file in DMNA-format*

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 Druee

References

AUSTAL2000 model reference (by Ingenieurbuero Janicke)

Examples

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

---

read.akterm	*function to read contents of a file in AKTERM-format*

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 obeservation as POSIXct.
2. "STA" is the WMO number of the observing statin
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 unsused
11. "QQ3" is the overall quality byte.

Note that DD and FF are always in degrees and m/s, indepenent 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 Druee

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   *function to read contents of a file in DMNA-format*

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   *function to read the contents of a file in Scintec Format-1*

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   lenght = n (number of times contained in the file)
each vevtor contains a non-profile variable (see variable list)
special varaible 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 vribale names

Usage

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

Arguments

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

Value

data.frame

Author(s)

Clemens Druee

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)
```

---

readmet   *Read some less Popular Formats Used in Meteorology*

Description

Contains tools for reading and writing data from or to files in dmna, Scintec Format-1, Campbell Scientific TOA5 formats.

Details

Package:	readmet
Type:	Package
Version:	1.6.9
Date:	2023-02-11
License:	GPL-3

Currently supported:

DMNA	Self describing gridded ASCII by Ing.Buero Janicke/Umweltbundesamt
	format description in e.g. AUSTAL dispersion model description https://www.umweltbundesamt.de/sites/de
Scintec Format-1	Format used to store atmospheric data from wind profilers, SODAR, RASS and Scintillometer devices by
	format description in the device manuals, available to customers upon registration from https://www.scintec
TOA5	Table oriented ACSII format #5 by Campbell Scientific
	format description in the device manuals, e.g. CR3000 data logger https://s.campbellsci.com/documents/us/

Available functions:

Function	Format
read.akterm	function to read contents of a file in AKTERM-format
read.dmna	function to read contents of a file in DMNA-format
dmna.axes	read grid positions from a file in DMNA-format
dmna.header	read header information from a file in DMNA-format
dmna.grid	read (horizontal) grid definition from a file in DMNA-format
write.dmna	function to write data to a file in DMNA-format
write.arcgrid	function to write data (read from DMNA file) into an ESRI ArcInfo gridded ASCII file
read.toa5	function to read contents of a file in TOA5-format
read.scintec1	function to read the contents of a file in Scintec Format-1
scintec1.comments	read the comment fields of a file in Scintec Format-1 format
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 Druee, Umweltmeteorologie, Universitaet Trier, Germany <druee@uni-trier.de>

scintec1.comments   *read the comment fields of a file in Scintec Format-1 format*

Description

get comment entries from a 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 Druee

References

Scintec APRun software manual

Examples

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

scintec1.header   *read the header of a file in Scintec Format-1 format*

Description

get timeseries of non-profile variables from a 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 coment 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 Druee

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 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 scintec1.header 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 scintec1.variables on the same file can be supplied via this argument.

Value

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

Author(s)

Clemens Druee

References

Scintec APRun software manual

Examples

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

---

scintec1.profile   *read the multi-level (profile) variables from a Scintec Format-1 file*

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 scintec1.header 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 scintec1.variables 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 Druee

References

Scintec APRun software manual

Examples

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

---

scintec1.variables   *read the variable descriptions from a file in Scintec Format-1 format*

Description

get timeseries of non-profile variables from a 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 scintec1.profile,scintec1.nonprofile, and read.scintec1
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 Druee

References

Scintec APRun software manual

Examples

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

---

write.arcgrid	*function to write data (read from DMNA file) into an ESRI ArcInfo gridded ASCII file*

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 tha data
file	name (incuding path, if needed) of the file to write to
xlen,ylen	number of data alon 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 way 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 Druee

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 POXIXct.
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 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 Druee

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