R documentation

of ‘rmap.as.Rd’

March 17, 2017

as.rmap *Create raster image from R objects or GDAL raster files.*

Description

as.rmap converts R **base** objects matrix, array, numeric, data.frame list, **sp** objects SpatialGridDataFrame, SpatialPixelsDataFrame and SpatialPointsDataFrame, **raster** objects raster, stack and brick, and GDAL raster files (using functions from **rgdal** package) to rmapRaster object.

Usage

```
as.rmap(obj, ...)
as_rmap(obj, ...)
```

Arguments

obj	R object for coercion
...	Depending on class of obj, arguments are passed to repsective functions.

Details

as_rmap is a synonym to as.rmap.

This is a high-level function to create rmapRaster objects. The follwed classes of R objects are implemented:

‘Data Class’	‘Appropriate method’
array	rmap.new
matrix	rmap.new
numeric	rmap.new
data.frame	allocate
SpatialPointsDataFrame (**sp**)	allocate
SpatialPixelsDataFrame (**sp**)	allocate
SpatialGridDataFrame (**sp**)	rmap.new
list of rmapRaster objects	unlist
list *(general)*	Items `$x` and `$y` are required, If lengths of `$x` and `$y` are equal to dim of data, t
ggmap (**ggmap**)	rmap.new.

raster (**raster**)	rmap.new.
brick (**raster**)	rmap.new.
stack (**raster**)	rmap.new.
character (GDAL supported file name)	gdal.read.

Generally, allocate is used for objects with non-regular grid, and rmap.new is used for regular grids. The raster grid is defined from object properties or from sessional grid.

Color tables are supported for GDAL file names and **raster** objects (raster, brick, stack).

For ENVI *.hdr Labelled Raster Files there are alternatives:

1. Read object with GDAL (gdal.read);
2. Read object without GDAL (envi.read).

Value

Object of class rmapRaster

Author(s)

Nikita Platonov <platonov@sevin.ru>

Examples

```
session.grid(NULL)
a1 <- as.rmap(volcano)
print(a1)
display(a1)

session.grid(NULL)
b <- rmap.dummy(mul=1/16,bandname=format(Sys.Date()+seq(3)-1,"%A"))
print(b)

c1 <- as.list(b[1])
str(c1)
b1a <- as.rmap(c1)
print(c(original=b[1],imported=b1a))
session.grid(NULL)
b1b <- as.rmap(c1$z)
print(c(projection=rmap.proj(b1b)))
print(b1b)

c2 <- as.data.frame(b)
str(c2)
session.grid(NULL)
b2a <- as.rmap(c2)
print(b2a)

session.grid(NULL)
attr(c2,"proj") <- NULL
b2b <- as.rmap(c2)
print(b2b)
print(rmap.grid(b2b))

c3 <- unclass(as.matrix(b))
str(c3)
```

```
session.grid(b)
b3a <- as.rmap(c3)
print(b3a)
print(rmap.grid(b3a))
session.grid(NULL)
b3b <- as.rmap(c3)
print(b3b)
print(rmap.grid(b3b))

c4 <- as.array(b)
str(c4)
session.grid(b)
b4a <- as.rmap(c4)
print(b4a)
print(rmap.grid(b4a))
session.grid(NULL)
b4b <- as.rmap(c4)
print(b4b)
print(rmap.grid(b4b))

n <- 20
c5 <- data.frame(y=runif(n,min=1000000,max=5000000)
                ,x=runif(n,min=-3000000,max=1000000)
                ,value=runif(n,min=0,max=10))
print(head(c5))
session.grid(b)
b5a <- as.rmap(c5)
print(b5a)
display(b5a)
session.grid(NULL)
b5b <- as.rmap(c5)
print(b5b)
display(b5b)

b6 <- as.rmap(system.file("pictures/erdas_spnad83.tif",package="rgdal"))
print(b6)
display(b6,coast=FALSE,col="orange")

## Not run:
require(raster)
r <- brick(system.file("external/rlogo.gri",package="raster"))
print(r)
b7 <- as.rmap(r)
rmap.proj(b7) <- ""
print(b7)
display.rgb(b7)
## End(Not run)
```

Index