Package 'RAFPALdb'

March 22, 2016

Title R-AFPALdb package.

Version 0.0.0.9000

Description Package dedicated to the interaction with and extraction of data stored in AFAPLdb.

Depends R (>= 3.2.2)

License ``Use it at your own risk'' license.

LazyData true

RoxygenNote 5.0.1.9000

Imports RMySQL,
   rjson,
   sp,
   ncdf4,
   raster,
   ade4,
   maptools

R topics documented:

Connect	*Connection to the database*

Description

This function allows you to connect to AFPALdb.

Usage

```
Connect(user = -1, pass = -1, host = -1)
```

Arguments

user	User.
pass	Password.
host	Adresse of the server. Default 'localhost'.

Examples

```
Connect(user="root",pass="root123")
```

---

correlationPlot	*Generate a correlation plot.*

Description

This figure measures the correlation between two records at different periods and with different time windows.

Usage

```
correlationPlot(x1, x2, xmn, xmx, xstep, ymn = 0, ymx = xmx, ystep,
  Contour = FALSE, exportPDF = TRUE, savePDF = paste(getwd(),
  "/CorrelationPlot.pdf", sep = ""), title = "CorrelationPlot",
  xlab = "Age", ylab = "Width")
```

Arguments

x1	Record to be compared. x1 must be a two-column matrix/data frame with time in the first column /and the variable to compare.
x2	Idem.
xmn	Date of beginning of the comparison.
xmx	Date of end of the comparison.
xstep	Temporal interval between two measures.
ymn	Minimum size of window to be measured.
ymx	Maximum size of window to be measured. Default is a global measure of the correlation of the two records.
ystep	Steps of the correlation window.
Contour	Boolean. If TRUE, contours at every 0.1 point of correlation will be added to the figure.
exportPDF	Boolean. If TRUE, the graphic will be saved in the file named 'savePDF'.
savePDF	Address and Name of the file. Default if 'CorrelationPlot.pdf' in the working directory.
title	Title to display on the graphic.
xlab	Label of the x-axis.
ylab	Label of the y-axis.

Examples

```
d1=getData(Citation_Key="Chase_etal_2013",Record_Name="SWP-1-1",Proxy.Uncer=FALSE,Chrono.Uncer=FALSE,Proxy=
d2=getData(Citation_Key="Chase_etal_2013",Record_Name="SWP-1-1",Proxy.Uncer=FALSE,Chrono.Uncer=FALSE,Proxy=
correlationPlot(d1,d2,3000,8000,100,ymn=0,ystep=100,Contour=TRUE)
```

---

digitizeR   *DigitizeR software*

Description

Starts the digitizeR software.

Usage

```
digitizeR()
```

Examples

```
digitizeR()
```

Disconnect   *Disconnect the user from the database*

Description

This function disconnects the AFPALdb.

Usage

```
Disconnect()
```

Examples

```
Disconnect()
```

---

exportMonteCarloInterpolation
                                Export interpolated reconstructions into .csv and .pdf files.

Description

Export interpolated reconstructions into .csv and .pdf files.

Usage

```
exportMonteCarloInterpolation(dat, saveData = getwd())
```

Arguments

dat   The classic Monte-Carlo object generated by Interpolation.init().

saveData   Folder where to save the outputs (in the working folder by default).

Examples

```
exportMonteCarloInterpolation(Pella.init)
```

---

exportMonteCarloStacking

Export stacked reconstructions into .csv and .pdf files.

Description

Export stacked reconstructions into .csv and .pdf files.

Usage

```
exportMonteCarloStacking(dat, saveData = getwd(), vline = seq(min(dat$XX),
  max(dat$XX), length.out = 11), XLAB = "Age")
```

Arguments

dat	The classic Monte-Carlo object generated by Interpolation.init().
saveData	Folder where to save the outputs (in the working folder by default).
vline	List of position from where to draw a vertical dashed lines (the total xrange is divided in 10 equal intervals by default).
XLAB	Label of the x-axis.

Examples

```
exportMonteCarloStacking(Pella.init,XLAB="Age (cal BP)")
```

---

extractTRACE

Extract data from TRACE.

Description

Extract data from TRACE.

Usage

```
extractTRACE(gridcells, var, month = 1, decadal = TRUE, layer = "",
  exportCSV = FALSE, saveCSV = paste(getwd(), "/Extract_", var, ".csv", sep
  = ""), tmin = -50, tmax = 22000)
```

Arguments

gridcells	Coordinates of the pixels of interest (lon1, lat1, lon2, lat2, ...).
var	Variable of interest (TS/PRECC/PRECT/AI/MAP/MAT/PET/TSMN/TSMX/SOLIN).
month	A vector of values comprised between 1 and 12.
decadal	Boolean to select between annual (FALSE) or averaged by decade (TRUE, default) timeseries.
layer	The layer to extract from the netcdf file.
exportCSV	Boolean to export the results in a csv.
saveCSV	Address of the file. Default is "Extract_var.csv" in the working directory.
tmin	Subset of the 22,000 years of simulation.
tmax	Subset of the 22,000 years of simulation.

Examples

```
TS=extractTRACE(c(0,0,5,5),"TS",month=4:6,exportCSV=TRUE)
```

---

getAgeList   *Get the Age table.*

Description

This function allows you to get all the entries from the AFPALdb.C14 table.

Usage

```
getAgeList(ORDERBY = "LabCode")
```

Arguments

ORDERBY   Set of variables (separated by a comma) to order the results by. Results sorted
                 by 'LabCode' by default.

Examples

```
Age=getAgeList()
Age=getAgeList("LabCode")
```

---

getAltitude   *Get the Altitude table.*

Description

This function allows you to get all the entries from the AFPALdb.Altitude table.

Usage

```
getAltitude()
```

Examples

```
altitude=getAltitude()
```

---

getBio   *Get the bio table.*

Description

This function allows you to get all the entries from the AFPALdb.bio table.

Usage

```
getBio()
```

Examples

```
bio=getBio()
```

---

getBiome   *Get the Biome table.*

Description

This function allows you to get all the entries from the AFPALdb.Biome table.

Usage

```
getBiome()
```

Examples

```
biome=getBiome()
```

---

getChronoAgeList   *Get the ChronoC14 table.*

Description

This function allows you to get all the entries from the AFPALdb.ChronoC14 table.

Usage

```
getChronoAgeList(ORDERBY = "Chrono_ID,LabCode")
```

Arguments

ORDERBY   Set of variables (separated by a comma) to order the results by. Results sorted
               by 'Chrono_ID x LabCode' by default.

Examples

```
chronocAge=getChronoAgeList()
chronocAge=getChronoAgeList("LabCode")
```

---

getChronoDataList   *Get the ChronoData table.*

Description

This function allows you to get all the entries from the AFPALdb.ChronoData table.

Usage

```
getChronoDataList(ORDERBY = "Chrono_ID,Dataset_ID")
```

Arguments

ORDERBY   Set of variables (separated by a comma) to order the results by. Results sorted
                 by 'Chrono_ID x Dataset_ID' by default.

Examples

```
chronodata=getChronoDataList()
chronodata=getChronoDataList("Dataset_ID")
```

---

getChronoFromDataID   *Extract datasets from the database*

Description

This function allows you to extract the chronology of a dataset identified by its ID.

Usage

```
getChronoFromDataID(Dataset_ID, Uncertainties = FALSE)
```

Arguments

Dataset_ID   ID of the dataset that needs its little chronology.

Uncertainties   Boolean value for the extraction of chronological uncertainties.

Examples

```
chrono1=getChronoFromDataID(49)
chrono2=getChronoFromDataID(49,TRUE)
```

---

getChronologyList *Get the Chronology table.*

Description

This function allows you to get all the entries from the AFPALdb.Chronology table, except the data.

Usage

```
getChronologyList(ORDERBY = "Chrono_ID")
```

Arguments

ORDERBY   Set of variables (separated by a comma) to order the results by. Results sorted
               by 'Chrono_ID' by default.

Examples

```
chronology=getChronologyList()
chronology=getChronologyList("Record_Name")
```

---

getColorGradient *Open a SSH tunnel with the server to open a distant file.*

Description

This functions opens a SSH tunnel with the server to open a distant file.

Usage

```
getColorGradient(name = "", reverse = FALSE)
```

Arguments

name   Name of the gradient. If empty or mispelled, the user will have to select from
               the list of available gradients.

reverse   Boolean to reverse the color gradient.

Examples

```
getColorGradient()
```

---

getData   *Extract chronologies from the database*

Description

This function allows you to extract JSON objects (chronology containers) from AFPALdb.

This function allows you to extract JSON objects (data containers) from AFPALdb.

Usage

```
getData(Site_Name = "", Record_Name = "", Citation_Key = "", Proxy = "",
  Proxy.Uncer = FALSE, Dataset_ID = -1, Chronology = TRUE,
  Chrono.Uncer = FALSE)

getData(Site_Name = "", Record_Name = "", Citation_Key = "", Proxy = "",
  Proxy.Uncer = FALSE, Dataset_ID = -1, Chronology = TRUE,
  Chrono.Uncer = FALSE)
```

Arguments

Site_Name	Name of the site.
Record_Name	Name of the specific record requested.
Citation_Key	Reference associated to the dataset requested.
Proxy	Type of proxy of dataset requested.
Dataset_ID	ID of the requested dataset.
Chrono_ID	ID of the requested dataset.
Site_Name	Name of the site.
Record_Name	Name of the specific record requested.
Citation_Key	Reference associated to the dataset requested.
Proxy	Type of proxy of dataset requested.

Examples

```
d0=getData() # Returns NULL
d1=getData(Site_Name="Seweweekspoort")
d2=getData(Proxy="d13C")
d3=getData(Citation_Key="Chase_etal_2013")
d4=getData(Citation_Key="Chase_etal_2013",Record_Name="SWP-1-1")
d0=getData() # Returns NULL
d1=getData(Site_Name="Seweweekspoort")
d2=getData(Proxy="d13C")
d3=getData(Citation_Key="Chase_etal_2013")
d4=getData(Citation_Key="Chase_etal_2013",Record_Name="SWP-1-1",Proxy.Uncer=TRUE,Chrono.Uncer=TRUE)
```

getDatasetFromReference

Get a list of datasets associated to a given reference.

Description

This function allows you to get a list of datasets associated to a given reference. This function DOES NOT return data, just information about the dataset.

Usage

```
getDatasetFromReference(Citation_Key, fullRef = FALSE,
  ORDERBY = "Record_Name")
```

Arguments

Citation_Key	Citation key.
fullRef	Boolean. TRUE returns all the details of the site.
ORDERBY	Field name to sort the output table. Default 'Record_Name'.

Examples

```
getDatasetFromReference("Chase_etal_2013")
getDatasetFromReference("Tierney_etal_2008",ORDERBY="Proxy")
```

getDatasetFromSite   *Get a list of datasets associated to a given site.*

Description

This function allows you to get a list of datasets associated to a given site. This function DOES NOT return data, just information about the dataset.

Usage

```
getDatasetFromSite(Site_ID = -1, Site_Name = "", Record_Name = "",
  fullRef = FALSE, ORDERBY = "Record_Name")
```

Arguments

fullRef	Boolean. TRUE returns all the details of the site.
ORDERBY	Field name to sort the output table. Default 'Record_Name'.
Citation_Key	Citation key.

Examples

```
getDatasetFromSite(Site_ID=1)
getDatasetFromSite(Site_Name="Seweweekspoort")
getDatasetFromSite(Record_Name="SWP-1-1")
```

---

getDatasetList *Get the Dataset table.*

Description

This function allows you to get all the entries from the AFPALdb.Reference table, except the data.

Usage

```
getDatasetList(ORDERBY = "Dataset_ID")
```

Arguments

ORDERBY   Set of variables (separated by a comma) to order the results by. Results sorted by 'Dataset_ID' by default.

Examples

```
dataset=getDatasetList()
dataset=getDatasetList("Proxy")
dataset=getDatasetList("Record_Name,Citation_Key")
```

---

getDataSSH *Open a SSH tunnel with the server to open a file.*

Description

This functions opens a SSH tunnel with the server to open a file.

Usage

```
getDataSSH(file, header = TRUE, sep = ",")
```

Arguments

file   Complete address of the file.

header   Boolean (used in read.table).

sep   Column separator (used in read.table).

Examples

```
head(getDataSSH("/home/emmanuel.chevalier/Bureau/*.csv"))
```

---

getDistrib *Get the distrib table.*

Description

This function allows you to get all the entries from the AFPALdb.distrib table.

Usage

```
getDistrib()
```

Examples

```
distrib=getDistrib()
```

---

getInsolation *Get monthly insolation values.*

Description

This function allows to extract monthly insolation values at 500 years resolution between present and 100,000 years ago.

Usage

```
getInsolation(lat, month, from = 0, to = 1e+05)
```

Arguments

lat	Latitude of the desired insolation. Integer between -90 and 90.
month	Month of the desired insolation. Can be either an integer between 1 and 12 or a string among jan/feb/mar/apr/may/jun/jul/aug/sep/oct/nov/dcm.
from	Beginning of the desired period of time. Default is '0'.
to	End of the desired period of time. Default is '100,000'.

Examples

```
getInsolation(65,"aug",0,35000)
getInsolation(-30,"dec",10000,0000)
```

---

getPET *Get the PET table.*

Description

This function allows you to get all the entries from the AFPALdb.PET table.

Usage

```
getPET()
```

Examples

```
pet=getPET()
```

---

getPrecip *Get the precip table.*

Description

This function allows you to get all the entries from the AFPALdb.precip table.

Usage

```
getPrecip()
```

Examples

```
precip=getPrecip()
```

---

getReferenceFromDataID

Get the reference associated to a dataset.

Description

This function allows you to get the reference associated to a dataset.

Usage

```
getReferenceFromDataID(Dataset_ID, fullRef = FALSE)
```

Arguments

fullRef	Boolean. TRUE returns all the details of the reference.
Region	Name of the region.

Examples

```
getReferenceFromDataID(1)
getReferenceFromDataID(2,TRUE)
```

getReferenceFromRegion

Get a list of studies performed in a given Region.

Description

This function allows you to get a list of studies performed in a given Region.

Usage

```
getReferenceFromRegion(Region, fullRef = FALSE, ORDERBY = "Citation_Key")
```

Arguments

Region	Name of the region.
fullRef	Boolean. TRUE returns all the details of the reference.
ORDERBY	Field name to sort the output table. Default 'Citation_Key'.

Examples

```
getReferenceFromRegion("SRZ")
getReferenceFromRegion("WRZ",TRUE,ORDERBY="Site_Name")
```

getReferenceFromSite   *Get a list of references associated to a given site.*

Description

This function allows you to get a list of references associated to a given site.

Usage

```
getReferenceFromSite(Site_ID = -1, Site_Name = "", Record_Name = "",
  fullRef = FALSE, ORDERBY = "Citation_Key")
```

Arguments

Site_ID	ID of the site.
Site_Name	Name of the site.
Record_Name	Name of the record.
fullRef	Boolean. TRUE returns all the details of the reference.
ORDERBY	Field name to sort the output table. Default 'Citation_Key'.

Examples

```
d1=getReferenceFromSite(Site_Name="Seweweekspoort")
d2=getReferenceFromSite(Site_ID=1,fullRef=FALSE,ORDERBY="Journal")
d3=getReferenceFromSite(Record_Name="SWP-1-1",fullRef=TRUE)
```

---

getReferenceFromXY   *Get a list of studies performed in a given area.*

Description

This function allows you to get a list of studies performed in a given area.

Usage

```
getReferenceFromXY(xmn, xmx, ymn, ymx, fullRef = FALSE,
  ORDERBY = "Citation_Key")
```

Arguments

xmn	Lower bound of the longitudinal range.
xmx	Upper bound of the longitudinal range.
ymn	Lower bound of the latitudinal range.
ymx	Upper bound of the latitudinal range.
fullRef	Boolean. TRUE returns all the details of the reference.
ORDERBY	Field name to sort the output table. Default 'Citation_Key'.

Examples

```
getReferenceFromXY(20,30,-30,-20,ORDERBY="Site_Name")
getReferenceFromXY(25,30,-30,-20,fullRef=TRUE)
getReferenceFromXY(25,20,-30,-20)
```

---

getReferenceList   *Get the Reference table.*

Description

This function allows you to get all the entries from the AFPALdb.Reference table.

Usage

```
getReferenceList(ORDERBY = "Citation_Key")
```

Arguments

ORDERBY	Set of variables (separated by a comma) to order the results by. Results sorted by 'Citation_Key' by default.

Examples

```
ref=getReferenceList()
ref=getReferenceList("Citation_Key")
```

---

getRefSiteList *Get the RefSite table.*

Description

This function allows you to get all the entries from the AFPALdb.RefSite table.

Usage

```
getRefSiteList(ORDERBY = "Citation_Key,Record_Name")
```

Arguments

ORDERBY   Set of variables (separated by a comma) to order the results by. Results sorted
                by 'Citation_Key x Record_Name' by default.

Examples

```
refsite=getRefSiteList()
refsite=getRefSiteList("Record_Name")
```

---

getSiteFromDataID *Get the site associated to a dataset.*

Description

This function allows you to get the site associated to a dataset.

Usage

```
getSiteFromDataID(Dataset_ID, fullRef = FALSE)
```

Arguments

fullRef   Boolean. TRUE returns all the details of the site.

Region   Name of the region.

Examples

```
getSiteFromDataID(1)
getSiteFromDataID(2,TRUE)
```

getSiteFromReference   *Get a list of references associated to a given site.*

Description

This function allows you to get a list of references associated to a given site.

Usage

```
getSiteFromReference(Citation_Key, fullRef = FALSE, ORDERBY = "Site_Name")
```

Arguments

Citation_Key	Citation key.
fullRef	Boolean. TRUE returns all the details of the site.
ORDERBY	Field name to sort the output table. Default 'Site_Name'.

Examples

```
getSiteFromReference("Chase_etal_2013")
getSiteFromReference("Tierney_etal_2008",TRUE)
```

getSiteFromRegion   *Get a list of sites part of a given Region.*

Description

This function allows you to get a list of sites part of a given Region.

Usage

```
getSiteFromRegion(Region, fullRef = FALSE, ORDERBY = "Site_Name")
```

Arguments

Region	Name of the region.
fullRef	Boolean. TRUE returns all the details of the reference.
ORDERBY	Field name to sort the output table. Default 'Site_Name'.

Examples

```
getSiteFromRegion("SRZ")
getSiteFromRegion("WRZ",TRUE)
```

---

getSiteFromXY *Get a list of studies performed in a given area.*

Description

This function allows you to get a list of studies performed in a given area.

Usage

```
getSiteFromXY(xmn, xmx, ymn, ymx, fullRef = FALSE, ORDERBY = "Site_Name")
```

Arguments

xmn	Lower bound of the longitudinal range.
xmx	Upper bound of the longitudinal range.
ymn	Lower bound of the latitudinal range.
ymx	Upper bound of the latitudinal range.
fullRef	Boolean. TRUE returns all the details of the site.
ORDERBY	Field name to sort the output table. Default 'Site_Name'.

Examples

```
getSiteFromXY(20,30,-30,-20)
getSiteFromXY(25,30,-30,-20)
getSiteFromXY(25,20,-30,-20,ORDERBY="Lat")
```

---

getSiteList *Get the Site table.*

Description

This function allows you to get all the entries from the AFPALdb.Site table.

Usage

```
getSiteList(ORDERBY = "Site_Name")
```

Arguments

ORDERBY	Set of variables (separated by a comma) to order the results by. Results sorted by 'Site_Name' by default.

Examples

```
sites=getSiteList()
sites=getSiteList(ORDERBY="Site_ID")
sites=getSiteList("Site_Name")
```

`getSiteRegionList`   *Get the SiteRegion table.*

Description

This function allows you to get all the entries from the AFPALdb.SiteRegion table.

Usage

```
getSiteRegionList(ORDERBY = "Region")
```

Arguments

ORDERBY   Set of variables (separated by a comma) to order the results by. Results sorted
                 by 'Region' by default.

Examples

```
siteregion=getSiteRegionList()
siteregion=getSiteRegionList("SRZ")
```

`getTaxalist`   *Get the Taxalist table.*

Description

This function allows you to get all the entries from the AFPALdb.Taxalist table.

Usage

```
getTaxalist()
```

Examples

```
taxalist=getTaxalist()
```

`getTmpr_max`   *Get the tmpr_max table.*

Description

This function allows you to get all the entries from the AFPALdb.tmpr_max table.

Usage

```
getTmpr_max()
```

Examples

```
tmpr_max=getTmpr_max()
```

---

getTmpr_mean *Get the tmpr_mean table.*

Description

This function allows you to get all the entries from the AFPALdb.tmpr_mean table.

Usage

```
getTmpr_mean()
```

Examples

```
tmpr_mean=getTmpr_mean()
```

---

getTmpr_min *Get the tmpr_min table.*

Description

This function allows you to get all the entries from the AFPALdb.tmpr_min table.

Usage

```
getTmpr_min()
```

Examples

```
tmpr_min=getTmpr_min()
```

Interpolation.init *Function used to initiate the Monte-Carlo process. It will extract the data from FPALdb and group them into a Monte-Carlo object.*

Description

Function used to initiate the Monte-Carlo process. It will extract the data from FPALdb and group them into a Monte-Carlo object.

Usage

```
Interpolation.init(sites, variable, time.range, time.resol, NREP = 50000,
  Citation_Keys = rep("", length(sites)))
```

Arguments

sites	List of sites to be used in the process.
variable	Variable of interest.
time.range	Minimum and maximum values of th period of interest. Does not necessary encompass the total temporal range of the records.
time.resol	Resolution of the record.
NREP	Number of scenarios to generate (default 50,000).
Citation_Keys	Number of scenarios to generate (default 50,000).

Examples

```
Pella.init=Interpolation.init(sites=c("PEL-1-1","PEL-1-4a"),variable="ReconAI",time.range=c(0,50000),time.r
```

---

makeTransparent *Add transparency to a color.*

Description

Add transparency to a color.

Usage

```
makeTransparent(color, alpha = 0.5)
```

Arguments

color	List of colors.
alpha	Degree of transparency (0: full transparency, 1: no transparency).

Examples

```
plot(1:20,1:20,col=getColorGradient("Rd2Bl20"),pch=15,cex=5)
points(1:20,1:20+2,col=makeTransparent(getColorGradient("Rd2Bl20"),alpha=0.5),pch=15,cex=5)
```

MonteCarloCentring *Automatically calculates an optimum centring value for the stacking.*

Description

Automatically calculates an optimum centring value for the stacking.

Usage

```
MonteCarloCentring(dat, ref, overlap)
```

Arguments

dat	The classic Monte-Carlo object generated by Interpolation.init().
ref	String that indicates the sequence of reference.
overlap	Vector that indicates for each site the period to consider for the overlap, e.g. c("site1",0,10000,"site2",3000,11000,"site3",350,8000)

Examples

```
Pella.init=MonteCarloCentring(Pella.init,"PEL-1-1",c("PEL-1-4a",100,1200))
```

MonteCarloInterpolation

Interpolate individually each reconstruction.

Description

Interpolate individually each reconstruction.

Usage

```
MonteCarloInterpolation(dat, exportData = FALSE, saveData = getwd())
```

Arguments

dat	The classic Monte-Carlo object generated by Interpolation.init().
exportData	Boolean (default FALSE) to save the individual interpolations.
saveData	Folder where to save the outputs (in the working folder by default).

Examples

```
Pella.init=MonteCarloInterpolation(Pella.init,exportData=TRUE)
```

MonteCarloStacking   *Perform the Monte-Carlo stacking.*

Description

Perform the Monte-Carlo stacking.

Usage

```
MonteCarloStacking(dat, exportData = FALSE, saveData = getwd())
```

Arguments

dat	The classic Monte-Carlo object generated by Interpolation.init().
exportData	Boolean (default FALSE) to save the individual interpolations.
saveData	Folder where to save the outputs (in the working folder by default).
centring	A vector of values to center data. Using the output of MonteCarloCentring() is recommended.

Examples

```
Pella.init=MonteCarloStacking(Pella.init,exportData=TRUE)
```

---

openArticle   *Function that opens pdf files from a reference or an author.*

Description

Function that opens pdf files from a reference or an author.

Usage

```
openArticle(Citation_Key = "", Author = "")
```

Arguments

Citation_Key   Key of the reference.

Author   Name of the author. Selection on the terminal possible if many references correspond to the name.

Examples

```
openArticle("Baker_etal_2014")
openArticle(Author="Scott")
```

---

openTRACE   *Open a TRACE file*

Description

Open a TRACE file.

Usage

```
openTRACE(var, month = 1, decadal = TRUE, layer = "")
```

Arguments

var   Variable of interest (TS/PRECC/PRECT/AI/MAP/MAT/PET/TSMN/TSMX/SOLIN).

month   A value between 1 and 12.

decadal   Boolean to select between annual (FALSE) or averaged by decade (TRUE, default) timeseries.

layer   The layer to extract from the netcdf file.

Examples

```
TS=openTRACE("TS",month=4,decadal=TRUE)
```

---

plotCountries *Add countries to a plot.*

Description

Add countries to a plot.

Usage

```
plotCountries(EXT = c(-180, 180, -90, 90), add = FALSE)
```

Arguments

EXT   Extension of the plot(xmin, xmax, ymin, ymax).

add   Boolean. If TRUE (default), add countries borders to an existing plot.

Examples

```
plotCountries(EXT=c(0,40,-40,0),add=FALSE)
```

---

plotCountriesByName *Add countries to a plot.*

Description

Add countries to a plot.

Usage

```
plotCountriesByName(countries = c("South Africa"), add = FALSE)
```

Arguments

countries   List of countries to be plotted.

add   Boolean. If TRUE (default), add countries borders to an existing plot.

Examples

```
plotCountriesByName(countries=c("South Africa","Lesotho","Namibia","Botswana","Swaziland"),add=FALSE)
```

---

plotTRACE   *Extract data from TRACE.*

Description

Extract data from TRACE.

Usage

```
plotTRACE(dat, exportPDF = FALSE, savePDF = paste(getwd(), "/plotTRACE.pdf",
  sep = ""), add.contours = TRUE, add.countries = TRUE, WIDTH = 5.51,
  HEIGHT = 5.51, legend = "bottom", EXT = c(-180, 180, -90, 90),
  COL = getColorGradient("Rd2Bl20"))
```

Arguments

dat	Matrix to be plotted.
exportPDF	Boolean. TRUE to generate a pdf of the figure.
savePDF	Address and name of the pdf. Default is "plotTRACE.pdf" in the working directory.
add.contours	Boolean to add contours on the figure.
WIDTH	The width of the figure in inches.
HEIGHT	The height of the figure in inches.
legend	Position of the legend (top/right/bottom/left/none).
EXT	Extension of the plot(xmin, xmax, ymin, ymax).
COL	Color gradient.
add.contries	Boolean to add country borders on the figure.

Examples

```
plotTRACE(dat=openTRACE("TS",month=4,decadal=TRUE))
```

---

removeSample   *Remove a sample from a Monte-Carlo object.*

Description

Remove a sample from a Monte-Carlo object.

Usage

```
removeSample(dat, n, site)
```

Arguments

dat	The classic Monte-Carlo object generated by Interpolation.init().
n	ID of the sample(s) to remove.
site	Name of the Record to remove samples from.

Examples

```
Pella.init=removeSample(Pella.init,21,"PEL-1-1")
```

---

verticalPlot   *Generate a correlation plot.*

Description

This figure measures the correlation between two records at different periods and with different time windows.

Usage

```
verticalPlot(dat, x.range = range(unlist(lapply(dat, function(z)
  return(range(z[, 1], na.rm = TRUE))))), xstep = 1000, reverse = rep(FALSE,
  length(dat)), ystep = unlist(lapply(dat, function(z)
  return(round(diff(range(z[, 2]))/6, 1)))), adjust = TRUE,
  relativeHeight = rep(1, length(dat)), COL = sample(colors(), length(dat)),
  CEX = 1, exportPDF = TRUE, savePDF = paste(getwd(), "/VerticalPlot.pdf",
  sep = ""), xlab = "Age", WIDTH = 3.54, HEIGHT = 9)
```

Arguments

dat	A list of two-columns matrixes that contains the data to plot.
x.range	Min and max values to plot on the x-axis. Default values are calculated from the min and max values from 'dat'.
xstep	Distance between vertical lines.
reverse	Vector of boolean values that indicates whether the record should be plotted upside down. Default is FALSE.
ystep	Vector that contains the distance between tickmarks on the y-axes. Dafault is 1/6th of the total range.
adjust	Set to TRUE (default) to reduce the white space between the curves.
relativeHeight	Vector that indicates the relative height each curve should add. By default, all curves have the same space allocated.
COL	Vector of colors. Random selection of color by default.
CEX	Size of the text. Default to 1.
exportPDF	Boolean to indicate whether the figure should be save in a pdf.
savePDF	Adress and name of the pdf. Default is "VarticalPlot.pdf" in the working directory.
xlab	Label of the x-axis.
WIDTH	Width of the pdf in inches (Default is 3.54in=90mm).
HEIGHT	Height of the pdf in inches (Default is 9in=22-230mm).

Examples

```
d1=getData(Citation_Key="Chase_etal_2013",Record_Name="SWP-1-1",Proxy.Uncer=FALSE,Chrono.Uncer=FALSE,Proxy=
d2=getData(Citation_Key="Chase_etal_2013",Record_Name="SWP-1-1",Proxy.Uncer=FALSE,Chrono.Uncer=FALSE,Proxy=
d3=getData(Citation_Key="Chase_etal_2013",Record_Name="SWP-1-5",Proxy.Uncer=FALSE,Chrono.Uncer=FALSE,Proxy=
d4=getData(Citation_Key="Chase_etal_2013",Record_Name="SWP-1-5",Proxy.Uncer=FALSE,Chrono.Uncer=FALSE,Proxy=
dat=list("SWP11-d15N"=d1,"SWP11-d13C"=d2,"SWP15-d13C"=d3,"SWP15-d15N"=d4)
verticalPlot(dat,adjust=TRUE,reverse=c(TRUE,FALSE,FALSE,TRUE),x.range=c(0,8000),xstep=1000,xlab="Age (Cal B
```

Index

29