Package ‘stacomiR’

October 14, 2022

Version 0.6.0.7
Date 2022-07-16
Title Fish Migration Monitoring
Description Graphical outputs and treatment for a database of fish pass monitoring. It is a part of the 'STACOMI' open source project developed in France by the French Office for Biodiversity institute to centralize data obtained by fish pass monitoring. This version is available in French and English. See <http://stacomir.r-forge.r-project.org/> for more information on ‘STACOMI’.
License GPL (>= 2)
URL http://stacomir.r-forge.r-project.org/
BugReports https://github.com/MarionLegrandLogrami/stacomiR/issues
Collate 'create_generic.R' 'data.R' 'fun_table_per_dis.R'
    'fun_write_monthly.R' 'fungraph.R' 'fungraph_glasseel.R'
    'funschema.R' 'funstat.R' 'funtable.R' 'ref_choice.R'
    'ref_coe.R' 'ref_dc.R' 'ref_df.R' 'ref_env.R' 'ref_horodate.R'
    'ref_list.R' 'ref_par.R' 'ref_parqual.R' 'ref_parquan.R'
    'ref_period.R' 'ref_stage.R' 'ref_taxa.R' 'ref_textbox.R'
    'ref_timestep.R' 'ref_timestep_daily.R' 'ref_year.R'
    'report_env.R' 'report_ge_weight.R' 'report_ope.R'
    'report_mig.R' 'report_sample_char.R' 'report_mig_char.R'
    'report_mig_multi.R' 'report_mig_env.R'
    'report_mig_interannual.R' 'report_sea_age.R'
    'report_silver_eel.R' 'report_species.R' 'setAs.R' 'stacomi.R'
    'stacomiR-package.R'
LazyLoad yes
LazyData true
Depends R (>= 4.0.0), methods, stacomirtools (>= 0.6.0.1)
Imports magrittr, intervals, RColorBrewer, stringr, RPostgres,
ggplot2, reshape2, graphics, utils, stats, lattice, grDevices,
Hmisc (>= 4.1-1), lubridate, dplyr, xtable, mgcv, rlang, pool,
withr, scales
Suggests testthat, viridis, knitr, rmarkdown

Author Cedric Briand [aut, cre],
Marion Legrand [aut],
Beaulaton Laurent [ctb],
Boulenger Clarisse [ctb],
Lafage Denis [ctb],
Grall Sebastien [ctb]

Maintainer Cedric Briand <cedric.briand00@gmail.com>

RoxygenNote 7.2.0

NeedsCompilation no

VignetteBuilder knitr

Repository CRAN

Date/Publication 2022-07-18 09:20:02 UTC

R topics documented:

barplot.report_annual-method ........................................ 6
calculate .......................................................... 7
calculate.report_ge_weight-method ................................. 7
calculate.report_mig-method ....................................... 8
calculate.report_mig_char-method ................................ 9
calculate.report_mig_env-method .................................. 9
calculate.report_mig_interannual-method ......................... 10
calculate.report_mig_mult-method ............................... 11
calculate.report_sample_char-method ............................. 12
calculate.report_sea_age-method ................................. 12
calculate.report_silver_eel-method ................................. 13
calculate.report_species-method ................................ 14
calculate ......................................................... 14
charge.............................................................. 14
charge.report_mig-method ......................................... 25
charge.report_mig_env-method .................................... 25
charge.report_mig_mult-method ................................. 26
charge.complement ................................................. 27
topics documented:

charge_complement,ref_parqual-method ........................................ 27
charge_with_filter,ref_par-method ............................................ 28
charge_with_filter,ref_parqual-method ........................................ 29
charge_with_filter,ref_parquan-method ....................................... 30
charge_with_filter,ref_stage-method .......................................... 31
charge_with_filter,ref_taxa-method ........................................... 32
choice_c,ref_choice-method .................................................... 32
choice_c,ref_dc-method .......................................................... 33
choice_c,ref_df-method .......................................................... 34
choice_c,ref_env-method .......................................................... 35
choice_c,ref_horodate-method .................................................. 36
choice_c,ref_list-method ......................................................... 36
choice_c,ref_par-method .......................................................... 37
choice_c,ref_stage-method ....................................................... 38
choice_c,ref_taxa-method .......................................................... 39
choice_c,ref_textbox-method ..................................................... 40
choice_c,ref_timestep_daily-method ............................................ 40
choice_c,ref_year-method ......................................................... 41
choice_c,report_annual-method .................................................. 42
choice_c,report_dc-method ....................................................... 43
choice_c,report_df-method ....................................................... 44
choice_c,report_env-method ..................................................... 45
choice_c,report_ge_weight-method ............................................. 45
choice_c,report_mig-method ...................................................... 46
choice_c,report_mig_char-method .............................................. 47
choice_c,report_mig_env-method ............................................... 48
choice_c,report_mig_interannual-method .................................... 49
choice_c,report_mig_mult-method .............................................. 50
choice_c,report_sample_char-method ......................................... 51
choice_c,report_sea_age-method ............................................... 52
choice_c,report_silver_eel-method ............................................. 53
choice_c,report_species-method ............................................... 54
coeff_durif ......................................................................... 55
colortable ......................................................................... 56
connect,report_annual-method .................................................. 57
connect,report_dc-method ....................................................... 57
connect,report_df-method ....................................................... 58
connect,report_env-method ..................................................... 59
connect,report_ge_weight-method ............................................. 59
connect,report_mig_char-method .............................................. 60
connect,report_mig_env-method ............................................... 61
connect,report_mig_interannual-method .................................... 61
connect,report_mig_mult-method .............................................. 62
connect,report_ope-method ..................................................... 63
connect,report_sample_char-method ......................................... 63
connect,report_sea_age-method ............................................... 64
connect,report_silver_eel-method ............................................. 64
connect,report_species-method ............................................... 65
## R topics documented:

- `envir_stacomi` ................................. 66
- `fungraph` .................................. 66
- `fungraph_glasseel` ......................... 68
- `funstat` .................................... 69
- `funtable` ................................... 70
- `fun_aggreg_for_plot` ....................... 71
- `fun_char_spe` ................................ 71
- `fun_date_extraction` ....................... 72
- `fun_report_mig_interannual` ........... 73
- `fun_report_mig_mult` ....................... 73
- `fun_report_mig_mult_overlaps` ........... 74
- `fun_schema` .................................. 75
- `fun_stage_durif` ........................... 75
- `fun_table_per_dis` ......................... 76
- `fun_weight_conversion` .................... 77
- `fun_write_monthly` ......................... 77
- `getvalue` ................................... 78
- `getvalue,ref_period-method` ............. 78
- `graphdate` ................................... 79
- `model` ....................................... 80
- `model,report_ge_weight-method` ......... 80
- `plot,report_annual,missing-method` .... 81
- `plot,report_dc,missing-method` .......... 82
- `plot,report_df,missing-method` .......... 83
- `plot,report_env,missing-method` ....... 84
- `plot,report_ge_weight,missing-method` 85
- `plot,report_mig,ANY-method` ............ 86
- `plot,report_mig_char,missing-method` 87
- `plot,report_mig_env,missing-method` ... 88
- `plot,report_mig_interannual,missing-method` 88
- `plot,report_mig_mult,missing-method` 89
- `plot,report_sample_char,missing-method` 90
- `plot,report_sea_age,missing-method` 91
- `plot,report_silver_eel,missing-method` 92
- `plot,report_species,missing-method` 93
- `print,report_dc-method` .................. 93
- `print,report_df-method` .................. 94
- `print,report_mig_mult-method` .......... 95
- `print,report_sample_char-method` ....... 95
- `print,report_sea_age-method` .......... 96
- `print,report_silver_eel-method` ....... 96
- `ref_choice-class` ......................... 97
- `ref_coe-class` ................................ 97
- `ref_dc-class` .................................. 98
- `ref_df-class` ................................ 99
- `ref_env-class` ............................... 99
- `ref_horodate-class` ....................... 100
- `ref_par-class` ............................... 100
R topics documented:

ref_parqual-class .............................................. 101
ref_parquan-class .............................................. 101
ref_period-class ................................................. 102
ref_stage-class .................................................. 102
ref_taxa-class ................................................... 103
ref_textbox-class ............................................... 104
ref_timestep-class ............................................... 104
ref_timestepChar-class .......................................... 105
ref_timestep_daily-class ........................................ 105
ref_year-class .................................................... 106
report_annual-class ............................................. 107
report_dc-class .................................................. 109
report_df-class .................................................. 110
report_env-class ................................................ 112
report_ge_weight-class ......................................... 113
report_mig-class ................................................. 116
report_mig_char-class ......................................... 118
report_mig_env-class .......................................... 120
report_mig_interannual-class ................................. 122
report_mig_mult-class ......................................... 126
report_ope-class ................................................ 129
report_sample_char-class ...................................... 129
report_sea_age-class .......................................... 132
report_silver_eel-class ....................................... 134
report_species-class .......................................... 136
r_ann .............................................................. 138
r_ann_adour ...................................................... 138
r_dc ............................................................... 139
r_df ............................................................... 140
r_env ............................................................. 140
r_gew ............................................................. 141
r_mig ............................................................. 142
r_mig_char ...................................................... 143
r_mig_dc ........................................................ 143
r_mig_df ........................................................ 144
r_mig_env ...................................................... 144
r_mig_interannual .............................................. 145
r_mig_interannual_vichy ....................................... 145
r_mig_mult ...................................................... 146
r_mig_mult_dc .................................................. 147
r_mig_mult_df .................................................. 148
r_mig_mult_ope ................................................ 148
r_mig_ope ........................................................ 149
r_sample_char ................................................... 149
r_seaa ............................................................ 150
r_silver .......................................................... 151
setasqualitative ................................................. 151
setasqualitative,report_mig_char-method ..................... 152
**Description**

`barplot` method for object `report_annual-class`

**Usage**

```r
## S4 method for signature 'report_annual'
barplot(height, legend.text = NULL, ...)  
```

**Arguments**

- `height` An object of class `report_annual`
- `legend.text` See `barplot` help
- `...` additional arguments passed to `barplot`

**Value**

No return value, called for side effects
calcule

Author(s)
Cedric Briand <cedric.briand@eptb-vilaine.fr>

See Also
report_annual-class for examples

calcule

Generic method for calculations

Description
Generic method for calculations

Usage
calcule(object, ...)

Arguments
object Object
...
Additional parms

Author(s)
cedric.briand

calcule, report_ge_weight-method

Calcule method for report_ge_weight

Description
Calcule method for report_ge_weight

Usage
## S4 method for signature 'report_ge_weight'
calcule(object, silent = FALSE)

Arguments
object An object of class report_ge_weight-class
silent Boolean, if TRUE, information messages are not displayed, only warnings and errors
Value

An object of class `report_ge_weight-class` with `@calcdata["data"]` (essentially a selection of columns and renaming from `@data`) and `coe` daily coefficients extracted from the database `@calcdata["coe"]` and prepared for graphs.

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>

calcule.report_mig-method

Transforms migration per period to daily migrations, and performs the conversion from weights to number is data are stored as weights (glass eel).

Description

The calculation must be launched once data are filled by the connect method. Currently the negative argument has no effect.

Usage

```r
## S4 method for signature 'report_mig'
calcule(object, negative = FALSE, silent = FALSE)
```

Arguments

- `object`: An object of class `report_mig-class`
- `negative`: a boolean indicating if a separate sum must be done for positive and negative values, if true, positive and negative counts return different rows
- `silent`: Boolean, if TRUE, information messages are not displayed, only warnings and errors

Value

`report_mig` with `calcdata` slot filled. It is a list with one element per counting device containing

- `method`: In the case of instantaneous periods (video counting) the sum of daily values is done by the `fun_report_mig_mult` method and the value indicated in method is "sum". If any migration monitoring period is longer than a day, then the migration is split using the `fun_report_mig_mult_overlaps` function and the value indicated in the method is "overlaps" as the latter method uses the overlap package to split migration period.
- `data`: the calculated data.
- `contient_poids`: A boolean which indicates, in the case of glass eel, that the function `fun_weight_conversion` has been run to convert the weights to numbers using the weight to number coefficients in the database (see link `report_ge_weight`).
- `negative`: A parameter indicating if negative migration (downstream in the case of upstream migration devices) have been converted to positive numbers, not developed yet
**Note**

The class `report_mig` does not handle escapement rates nor 'devenir' i.e. the destination of the fishes.

---

**Description**

Merges the content of the list elements 'parqual' and 'parquan' in the data slot, and creates a single dataframe with one line per qualitative and quantitative pair. This methods allow to cross one quantity (e.g. length) with a qualitative parameter (e.g. sex).

**Usage**

```r
## S4 method for signature 'report_mig_char'
calcule(object, silent = FALSE)
```

**Arguments**

- `object` An object of class `report_mig_char-class`
- `silent` Boolean default FALSE, if TRUE information messages not displayed

**Value**

An object of class `report_mig_char-class` with slot @calcdata filled

---

**Description**

Calculations for migration in the class `report_mig_env-class`

**Usage**

```r
## S4 method for signature 'report_mig_env'
calcule(object, silent = FALSE)
```

**Arguments**

- `object` An object of class `report_mig_env-class`
- `silent` Boolean default FALSE, if TRUE information messages not displayed
calcule,report_mig_interannual-method

**Value**

`report_mig_env-class` with data in slot `r_mig_env@report_mig_mult@caldata`  

---

**Description**

Performs the calculation of seasonal coefficients for the `plot(plot.type="seasonal")` method. The numbers are split according to the period chosen, one of "day","week","month","2 weeks". French labels are also accepted as arguments. Once this is done, the seasonality of the migration is displayed using the day when the first fish was seen, then the days (or period) corresponding to 5, 50, 95, and 100 percent of the migration. The duration of 90

**Usage**

```r
## S4 method for signature 'report_mig_interannual'
calculate(object, silent = FALSE, timesplit = "mois")
```

**Arguments**

- `object` An object of class `report_mig_interannual-class`
- `silent` Boolean, if TRUE, information messages are not displayed, only warnings and errors
- `timesplit` One of "day","week","month","2 weeks", "jour","semaine","quinzaine","mois"

**Value**

An object of class `report_mig_interannual-class` with caldata slot filled.

**Note**

The class `report_mig_interannual` does not handle escapement rates nor 'devenir' i.e. the destination of the fishes.

**Author(s)**

Marion Legrand
calcule, report_mig_mult-method

#' Transforms migration per period to daily migrations, and performs the conversion from weights to number if data are stored as weights (glass eel). This calculation is performed in a loop for all dc.

Description

The calculation must be launched once data are filled by the connect method. Currently the negative argument has no effect.

Usage

## S4 method for signature 'report_mig_mult'
calcule(object, negative = FALSE, silent = FALSE)

Arguments

object An object of class report_mig_mult-class
negative a boolean indicating if a separate sum must be done for positive and negative values, if true, positive and negative counts return different rows
silent Default FALSE, should messages be stopped

Value

report_mig_mult with a list in slot calcdat. For each dc one will find a list with the following elements

- **method** In the case of instantaneous periods (video counting) the sum of daily values is done by the fun_report_mig_mult method and the value indicated in method is 'sum'. If any migration monitoring period is longer than a day, then the migration is split using the fun_report_mig_mult_overlaps function and the value indicated in the method is 'overlaps' as the latter method uses the overlap package to split migration period.

- **data** the calculated data. If weight are present, the columns display weight or numbers, the total number is 'Effectif_total' and corresponds to the addition of numbers and numbers converted from weight, the total weight is 'Poids_total'+'poids_depuis_effectifs' and corresponds to weighed glass eel plus glass eel number converted in weights. CALCULE corresponds to calculated number, MESURE to measured numbers, EXPERT to punctual expertise of migration (for instance measured in other path, or known migration of fishes passing the dam but not actually counted, PONCTUEL to fishes counted by visual identification but not by the counting apparatus (in case of technical problem for instance)

- **contient_poids** A boolean which indicates, in the case of glass eel, that the function fun_weight_conversion has been run to convert the weights to numbers using the weight to number coefficients in the database (see linkreport_ge_weight).

- **negative** A parameter indicating if negative migration (downstream in the case of upstream migration devices) have been converted to positive numbers, not developed yet.
Note

The class does not handle escapement rates, though structurally those are present in the database. If you want to use those you will have to do the calculation manually from the data in report_mig_mult@data.

calcule,report_sample_char-method

Calculation for report_sample_char

Description

In that class, most treatments are done in the query, this method checks that data are available and fills information for year, month, two weeks, week, doy

Usage

## S4 method for signature 'report_sample_char'
calcule(object, silent = FALSE)

Arguments

object An object of class report_sample_char-class
silent Boolean, if TRUE, information messages are not displayed, only warnings and errors

Value

An object of class report_sample_char-class with slot @data filled

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>

calcule,report_sea_age-method

Split data according to the limits set in the limit1hm, and limit2hm arguments of the report_sea_age-class.

Description

If no value are provided in the limit1hm slot, an error is returned, if no value is provided in the limit2hm slot a default upper value for salmon size is taken to ensure all salmon are either of age 1 or 2, but no age 3 are returned
Usage

```r
## S4 method for signature 'report_sea_age'
calculate(object, silent)
```

Arguments

- `object`: An object of class `report_sea_age-class`
- `silent`: Default FALSE, if TRUE the program should no display messages

Value

An object of class `report_sea_age-class` with calculated data in slot calldata

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>

---

Description

This `calculate` method for `report_silver_eel`, will transform data from long (one line per size characteristic, size, weight, eye diameter, pectoral fin measurement, lateral line and constrast) to wide format (one line per silver eel). It will also calculate Durif silvering index and Pankhurst and Fulton’s K.

Usage

```r
## S4 method for signature 'report_silver_eel'
calculate(object, silent)
```

Arguments

- `object`: An object of class `report_silver_eel-class`
- `silent`: Boolean, if TRUE, information messages are not displayed, only warnings and errors

Value

An object of class `report_silver_eel-class` with slot calldata filled, as a list for each counting device

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>
Description

The number will be split according to the split argument passed to the class, e.g. per year or month or week. Data from different DC will be grouped. Counts are given per taxa, unless there are several stages, in which case the counts correspond to taxa + stage.

Usage

```r
## S4 method for signature 'report_species'
calcule(object, silent = FALSE)
```

Arguments

- **object**: An object of class `report_species-class`
- **silent**: Boolean, if TRUE, information messages are not displayed, only warnings and errors

Value

An object of class `report_species-class` with caldata slot filled.

charge

*Generic method to load referentials*

Description

Generic method to load referentials

Usage

```r
charge(object, ...)
```

Arguments

- **object**: Object
- **...**: Additional parm

Author(s)

cedric.briand
Description

Loading method for Refchoice referential objects

Usage

## S4 method for signature 'ref_choice'
charge(object, vecteur, label, selected)

Arguments

- **object**: An object of class ref_choice
- **vecteur**: A vector of name, see example code.
- **label**: Labels for the choices
- **selected**: An integer indicating which object is selected at launch

Value

An S4 object of class ref_choice-class

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>

See Also

Other referential objects: ref_choice-class, ref_coe-class, ref_dc-class, ref_df-class, ref_horodate-class, ref_list-class, ref_par-class, ref_parqual-class, ref_parquan-class, ref_period-class, ref_stage-class, ref_taxa-class, ref_year-class

Examples

## Not run:
object=new("ref_choice")
charge(object, vecteur=c("oui","non"), label="essai", selected=as.integer(1))

## End(Not run)
charge, ref_coe-method  
loads the coefficients for the period defined in class

Description

The slots datedebut and datefin have to be filled before using charge

Usage

## S4 method for signature 'ref_coe'
charge(object)

Arguments

object  
An object of class ref_coe-class

Value

An object of class ref_coe-class

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>

Examples

## Not run:
object<- new('ref_coe')
object@datedebut<-strptime('01/01/1996',format='%d/%m/%Y')
object@datefin<-strptime('01/01/1997',format='%d/%m/%Y')
charge(object)

## End(Not run)

charge, ref_dc-method  
Method to load the counting devices of the control station

Description

Method to load the counting devices of the control station

Usage

## S4 method for signature 'ref_dc'
charge(object)
charge,ref_df-method

Arguments

object An object of class ref_dc-class

Value

an object of class ref_dc with data loaded

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>

Description

Loading method for DF referential objects

Usage

```r
## S4 method for signature 'ref_df'
charge(object)
```

Arguments

object An object of class ref_df-class

Value

An object of class ref_df with df loaded

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>

Examples

```r
## Not run:
object=new('ref_df')
charge(object)

## End(Not run)
```
charge,ref_env-method  Loading method for ref_env referential object

Description
Loading method for ref_env referential object

Usage
## S4 method for signature 'ref_env'
charge(object)

Arguments
object An object of class ref_env-class

Value
An S4 object of class ref_env with data loaded from the database

Author(s)
Cedric Briand <cedric.briand@eptb-vilaine.fr>

Examples
## Not run:
object=new('ref_env')
charge(object)

## End(Not run)

charge,ref_list-method  Loading method for ref_list referential objects

Description
Loading method for ref_list referential objects

Usage
## S4 method for signature 'ref_list'
charge(object, listechoice, label)
Arguments

object An object of class ref_list-class
listechoice A character vector setting the possible values in which the user can select
label A label for refiste

Value

An S4 object of class ref_list-class
An S4 object of class ref_list-class

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>

Examples

## Not run:
object=new('ref_list')
charge(object)

## End(Not run)

Description

Loading method for ref_par referential objects

Usage

## S4 method for signature 'ref_par'
charge(object)

Arguments

object An object of class ref_par-class

Value

An S4 object of class ref_par
An S4 object of class ref_par-class

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>
Examples

```r
## Not run:
object=new('ref_par')
charge(object)

## End(Not run)
```

Description

Loading method for Reparqual referential objects

Usage

```r
## S4 method for signature 'ref_parqual'
charge(object)
```

Arguments

- `object`: An object of class `ref_parqual-class`

Value

An S4 object of class `ref_parqual`

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>

Examples

```r
## Not run:
object=new('ref_parqual')
charge(object)

## End(Not run)
```
Description

Loading method for Reparquan referential objects

Usage

## S4 method for signature 'ref_parquan'
charge(object)

Arguments

object An object of class ref_parquan-class

Value

An S4 object of class ref_parquan-class with data loaded

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>

Examples

## Not run:
object=new('ref_parquan')
charge(object)

## End(Not run)

Description

Loading method for ref_stage referential objects

Usage

## S4 method for signature 'ref_stage'
charge(object)
Arguments

object An object of class \texttt{ref\_stage-class}

Value

An S4 object of class \texttt{ref\_stage-class} with all stages available in the database

Author(s)

Cedric Briand \texttt{<cedric.briand@eptb-vilaine.fr>}

Examples

\begin{verbatim}
## Not run:
  object=new('ref\_stage')
  charge(object)

## End(Not run)
\end{verbatim}

---

\textit{charge,ref\_taxa-method}

\textit{Loading method for \texttt{ref\_taxa} referential objects}

Description

Loading method for \texttt{ref\_taxa} referential objects

Usage

\begin{verbatim}
## S4 method for signature 'ref\_taxa'
charge(object)
\end{verbatim}

Arguments

object An object of class \texttt{ref\_taxa-class}

Value

An S4 object of class \texttt{ref\_taxa}

An S4 object of class \texttt{ref\_taxa-class} with all taxa loaded from the database

Author(s)

Cedric Briand \texttt{<cedric.briand@eptb-vilaine.fr>}

---
charge,ref_textbox-method

Loading method for ref_textbox referential objects

Description

Loading method for ref_textbox referential objects

Usage

## S4 method for signature 'ref_textbox'
charge(object, title, label)

Arguments

- **object**: An object of class ref_textbox-class
- **title**: A title for the frame
- **label**: A label for the TextBox

Value

An S4 object of class ref_textbox-class

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>

Examples

## Not run:
object=new('ref_textbox')
charge(object)
## End(Not run)
Loading method for ref_year referential objects

Description

Selects year available either in the bjo table if report_object==report_mig_interannual) or in the t_operation_ope table

Usage

```r
## S4 method for signature 'ref_year'
charge(object, objectreport = "report_ge_weight")
```

Arguments

- `object` An object of class `ref_year-class`
- `objectreport` The object report, default `report_ge_weight` other possible value `report_mig_interannual`

Value

- `object` An object of class `ref_year-class` with slot data filled with the available years for the corresponding report

Author(s)

Cedric Briand &lt;cedric.briand@eptb-vilaine.fr&gt;

Examples

```r
## Not run:
object=new("ref_year")
charge(object)
validObject(annee)
showMethods("charge")

## End(Not run)
```
charge, report_mig-method

 Loads additional data on migration control operations, df (fishway) dc (counting device).

Description

this method creates additional classes in envir_stacomi for later use in plot (operations, DF operation, DC operation). So unlike in most report classes where the charge method is only used by the graphical interface, it is necessary to run charge for report_mig.

Usage

## S4 method for signature 'report_mig'
charge(object, silent = FALSE)

Arguments

object An object of class report_mig-class
silent Should the program be returning messages

Value

An object of class report_mig-class with slots filled from values assigned in envir_stacomi environment

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>

charge, report_mig_env-method

charge method for report_mig_env class

Description

#’ Unique the other report classes where the charge method is only used by the graphical interface to collect and test objects in the environment envir_stacomi, and see if the right choices have been made in the graphical interface, this methods runs the charge,report_mig_mult-method and needs to be called from the command line (see examples)

Usage

## S4 method for signature 'report_mig_env'
charge(object, silent = FALSE)
charge,report_mig_mult-method

Arguments

object An object of class report_mig_env-class
silent Should the function remain silent (boolean)

Value

An object of class report_mig_env-class with data set from values assigned in envir_stacomi environment

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>

Description

Unique the other report classes where the charge method is only used by the graphical interface to collect and test objects in the environment envir_stacomi, and see if the right choices have been made in the graphical interface, this method is used to load data on migration control operations fishway operations, and counting devices operations as data from those are displayed in the main plots.

Usage

## S4 method for signature 'report_mig_mult'
charge(object, silent = FALSE)

Arguments

object An object of class report_mig_mult-class
silent Default FALSE, if TRUE the program should no display messages

Value

An object of class report_mig_mult-class with slots filled from values assigned in envir_stacomi environment

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>
charge_complement

**Generic method to load additional data**

**Description**

Generic method to load additional data

**Usage**

charge_complement(object, ...)

**Arguments**

- object: Object
- ...: Additional parms

**Author(s)**

cedric.briand

---

charge_complement,ref_parqual-method

*Loads an additional dataset this method is loaded to obtain the possible values of a qualitative parameter. Here data only contains one line*

**Description**

Loads an additional dataset this method is loaded to obtain the possible values of a qualitative parameter. Here data only contains one line

**Usage**

```r
## S4 method for signature 'ref_parqual'
charge_complement(object)
```

**Arguments**

- object: An object of class `ref_parqual-class`

**Value**

An S4 object of class `ref_parqual-class` with the valqual slot filled

**Author(s)**

Cedric Briand <cedric.briand@eptb-vilaine.fr>
Examples

```r
## Not run:
object=new('ref_par')
object<-charge(object)
charge_complement(object)

## End(Not run)
```

### Description

Loading method for `ref_par` referential objects searching only those parameters existing for a DC, a Taxa, and a stage.

### Usage

```r
## S4 method for signature 'ref_par'
charge_with_filter(object, dc_selected, taxa_selected, stage_selected)
```

### Arguments

- **object**: An object of class `ref_par-class`
- **dc_selected**: A counting device selected for the report
- **taxa_selected**: The taxa selected for the report
- **stage_selected**: The stage selected for the report

### Value

An S4 object of class `ref_par-class`

### Author(s)

Cedric Briand &lt;cedric.briand@eptb-vilaine.fr&gt;

### Examples

```r
## Not run:
object=new('ref_par')
charge_with_filter(object,dc_selected=6,taxa_selected=2038,stage_selected=c('AGJ','CIV'))

## End(Not run)
```
Description

Loading method for Reparqual referential objects searching only those parameters existing for a DC, a Taxon, and a stage

Usage

```r
## S4 method for signature 'ref_parqual'
charge_with_filter(object, dc_selected, taxa_selected, stage_selected)
```

Arguments

- `object`: An object of class `ref_parqual-class`
- `dc_selected`: The dc set in the report object
- `taxa_selected`: The taxa set in the report object
- `stage_selected`: The stage set in the report object

Value

An S4 object of class `ref_parqual-class`

Author(s)

Cedric Briand &lt;cedric.briand@eptb-vilaine.fr&gt;

Examples

```r
## Not run:
dc_selected=6
taxa_selected=2038
stage_selected='AGJ'
object=new('ref_parqual')
charge_with_filter(object,dc_selected,taxa_selected,stage_selected)

## End(Not run)
```
Loading method for Reparquan referential objects searching only those parameters existing for a DC (counting device), a Taxon, and a stage

Description

Loading method for Reparquan referential objects searching only those parameters existing for a DC (counting device), a Taxon, and a stage

Usage

```r
## S4 method for signature 'ref_parquan'
charge_with_filter(object, dc_selected, taxa_selected, stage_selected)
```

Arguments

- `object`: An object of class `ref_parquan-class`
- `dc_selected`: The dc set in the report object
- `taxa_selected`: The taxa set in the report object
- `stage_selected`: The stage set in the report object

Value

An S4 object of class `ref_parquan-class` with data loaded showing available parameters for one DC

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>

Examples

```r
## Not run:
dc_selected=6
taxa_selected=2038
stage_selected='AGJ'
object=new('ref_parquan')
charge_with_filter(object,dc_selected,taxa_selected,stage_selected)
## End(Not run)
```
## Description

Loading method for ref_stage referential objects searching only those stages existing for a DC and a Taxon.

## Usage

```r
## S4 method for signature 'ref_stage'
charge_with_filter(object, dc_selected, taxa_selected)
```

## Arguments

- `object`: An object of class `ref_stage-class`
- `dc_selected`: The selected counting device
- `taxa_selected`: The selected species

## Value

An S4 object of class `ref_stage-class` listing all stages available for one DC and one taxon

## Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>

## Examples

```r
## Not run:
dc_selected=6
taxa_selected=2038
object=new('ref_stage')
charge_with_filter(object,dc_selected,taxa_selected)
```

## End(Not run)
**charge_with_filter, ref_taxa-method**

*Loading method for ref_taxa referential objects searching only taxa existing for a DC*

---

**Description**

Loading method for ref_taxa referential objects searching only taxa existing for a DC

**Usage**

```r
## S4 method for signature 'ref_taxa'
charge_with_filter(object, dc_selected)
```

**Arguments**

- `object`: An object of class `ref_taxa-class`
- `dc_selected`: A counting device selected, only taxa attached to this dc are selected

**Value**

An S4 object of class `ref_taxa-class` with all taxa present on a DC (counting device)

**Author(s)**

Cedric Briand <cedric.briand@eptb-vilaine.fr>

**Examples**

```r
## Not run:
dc_selected=6
object=new('ref_taxa')
charge_with_filter(object, dc_selected=dc_selected)
## End(Not run)
```

---

**choice_c, ref_choice-method**

*Choice_c method for ref_choice referential objects*

---

**Description**

Choice_c method for ref_choice referential objects

**Usage**

```r
## S4 method for signature 'ref_choice'
choice_c(object, selectedvalue)
```
Arguments

object An object of class ref_choice-class

selectedvalue the value selected in the combo

Value

An S4 object of class ref_choice-class

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>

Examples

```r
## Not run:
object=new('ref_list')
object<-charge(object, vecteur=c('1','2'), label='please choose')
object<-choice_c(object)

## End(Not run)
```

Description

the choice_c method is intended to have the same behaviour as choice (which creates a widget in the graphical interface) but from the command line. The parameters for dc are transformed to integer as the ref_dc only takes integer in the dc slots. The method also loads the stations and ouvrages (dams) associated with the counting device (dc). The values passed to the choice_c method are then checked with the setValidty method. Finally, if an objectreport is passed as a parameter, the method will do a charge_with_filter to select only the taxa present in the counting devices

Usage

```r
## S4 method for signature 'ref_dc'
choice_c(object, dc)
```

Arguments

object an object of class ref_dc

dc a character vector of dc chosen

Value

An object of class ref_dc with dc selected
Author(s)
Cedric Briand <cedric.briand@eptb-vilaine.fr>

Examples
## Not run:
win=gwindow()
group=ggroup(container=win,horizontal=FALSE)
object=new('ref_dc')
object<-charge(object)
objectreport=new('report_mig_mult')
choice_c(object=object,objectreport=objectreport,dc=1)

## End(Not run)

choice_c,ref_df-method
Command line interface to choose a fishway

Description

the choice_c method is intended to have the same behaviour as choice (which creates a widget in the graphical interface) but from the command line. The parameters for dF are transformed to integer as the ref_df only takes integer in the df slots. DF are third in hierarchy in the stacomi database Station>ouvrage>DF>DC>operation. This class is only used in the report_df class.

Usage

## S4 method for signature 'ref_df'
choice_c(object, df)

Arguments

object an object of class ref_df-class
df a character vector of df chosen

Value

An object of class ref_df with df selected

Author(s)
Cedric Briand <cedric.briand@eptb-vilaine.fr>
Examples

```r
## Not run:
win=gwindow()
group=ggroup(container=win,horizontal=FALSE)
object=new('ref_df')
object<-charge(object)
objectreport=new('report_mig_mult')
choice_c(object=object,objectreport=objectreport,dc=1)

## End(Not run)
```

Description

The `choice_c` method is intended to have the same behaviour as `choice` (which creates a widget in the graphical interface) but from the command line.

Usage

```r
## S4 method for signature 'ref_env'
choice_c(object, stationMesure)
```

Arguments

- `object`: an object of class `ref_env`
- `stationMesure`: a character vector of the monitoring station code (corresponds to `stm_libelle` in the `tj_stationmesure_stm` table)

Value

An object of class `ref_env-class` with the monitoring station selected

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>
Description

Choice_c method for ref_horodate

Usage

```r
## S4 method for signature 'ref_horodate'
choice_c(
  object,
  nomassign = "horodate",
  funoutlabel = "nous avons le choix dans la date\n",
  horodate,
  silent = FALSE
)
```

Arguments

- **object**: An object of class `ref_horodate-class`
- **nomassign**: The name assigned in environment `envir_stacomi`
- **funoutlabel**: Text displayed by the interface
- **horodate**: The horodate to set, formats "%d/%m/%Y %H:%M:%s", "%d/%m/%y %H:%M:%s", "%Y-%m-%d %H:%M:%s" formats can also be passed with the date set to the minute %d/%m/%Y %H:%M or the day %d/%m/%Y ... are accepted. The choice_c method assigns and
- **silent**: Default FALSE, should messages be displayed

Value

An object of class `ref_horodate-class` with slot `horodate` set, and assigns an object of class `POSIXt` with name `nomassign` in `envir_stacomi`
Usage

```r
## S4 method for signature 'ref_list'
choice_c(object, selectedvalue)
```

Arguments

- `object`: An object of class `ref_list-class`
- `selectedvalue`: the value selected in the combo

Value

An S4 object of class `ref_list-class`

Note

the choice method assigns an object of class refList named ref_list in the environment envir_stacomi

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>

Examples

```r
## Not run:
object=new('ref_list')
object<-charge(object,vecteur=c('1','2'),label='please choose')
object<-choice_c(object)
## End(Not run)
```

Description

Command line interface to select a parameter

Usage

```r
## S4 method for signature 'ref_par'
choice_c(object, par, silent = FALSE)
```

Arguments

- `object`: an object of class `ref_par-class`
- `par`: A character vector of par
- `silent`: Default FALSE but not used there
choice_c,ref_stage-method

Value

An object of class ref_par-class

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>

Description

the choice_c method is intended to have the same behaviour as choice (which creates a widget in
the graphical interface) but from the command line. The values passed to the choice_c method for
stage is the code. Any numeric value will be discarded

Usage

## S4 method for signature 'ref_stage'
choice_c(object, stage, silent = FALSE)

Arguments

object An object of class ref_stage-class
stage the vector of stages chosen
silent Boolean, if TRUE, information messages are not displayed

Value

An S4 object of class ref_stage-class with the stage selected in the data slot

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>

Examples

## Not run:
object=new('ref_stage')
object<-charge(object)
## End(Not run)
Description

the choice_c method is intended to have the same behaviour as choice (which creates a widget in the graphical interface) but from the command line. The values passed to the choice_c method for taxa can be either numeric (2038 = Anguilla anguilla) or character.

Usage

## S4 method for signature 'ref_taxa'
choice_c(object, taxa)

Arguments

object                  An object from the class ref_taxa

 taxa                   The vector of taxa, can be either code (numeric) or latin name

Value

An S4 object of class ref_taxa-class with data filtered according to the taxa

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>

Examples

## Not run:
object=new('ref_taxa')
object<-charge(object)
objectreport=new('report_mig_mult')
choice_c(object=object,'Anguilla anguilla')

## End(Not run)
**Description**

Choice_c method for ref_textbox referential objects

**Usage**

```
## S4 method for signature 'ref_textbox'
choice_c(object, value, nomassign = "ref_textbox")
```

**Arguments**

- **object**: An object of class `ref_textbox-class`
- **value**: The value to set
- **nomassign**: The name with which the object will be assigned in `envir_stacom`

**Value**

An S4 object of class `ref_textbox-class` label selected

**Author(s)**

Cedric Briand <cedric.briand@eptb-vilaine.fr>

---

**Description**

the choice_c method is intended to have the same behaviour as choice (which creates a widget in the graphical interface) but from the command line.

**Usage**

```
## S4 method for signature 'ref_timestep_daily'
choice_c(object, datedebut, datefin)
```
choice_c,ref_year-method

Arguments

object An object of class ref_timestep_daily-class
datedebut A character (format '15/01/1996' or '1996-01-15' or '15-01-1996'), or POSIXct object
datefin A character

Value

An S4 object of class ref_timestep_daily-class with date selected

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>

Examples

## Not run:
object=new('ref_dc')
object<-charge(object)
choice_c(object=object,datedebut='2012-01-01',datefin='2013-01-01')

## End(Not run)

choice_c,ref_year-method

choice_c method for ref_year referential from the command line

Description

The choice_c method will issue a warning if the year is not present in the database Allows the selection of year and the assignment in environment envir_stacomi

Usage

## S4 method for signature 'ref_year'
choice_c(
  object,
  annee,
  nomassign = "ref_year",
  funoutlabel = gettext("Year selected\n", domain = "R-stacomiR"),
  silent = FALSE
)
Arguments

object An object of class ref_year-class
annee The year to select, either as a character or as a numeric
nomassign The name to be assigned in envir_stacomi
funoutlabel The label that appears in funout
silent Stops messages from being displayed if silent=TRUE, default FALSE

Value

object An object of class ref_year-class with year selected

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>

Examples

```r
## Not run:
object=new("ref_year")
object<-charge(object)
win=gwindow(title="test ref_year")
    group=ggroup(container=win,horizontal=FALSE)
    choice(object,nomassign="ref_year",funoutlabel="essai",titleFrame="essai ref_year",preselect=1)
    dispose(win)

## End(Not run)
```

choice_c,report_annual-method

command line interface for report_annual-class

description

The choice_c method fills in the data slot for classes ref_dc-class, ref_taxa-class, ref_stage-class and two slots of ref_year-class

Usage

```r
## S4 method for signature 'report_annual'
choice_c(object, dc, taxa, stage, start_year, end_year, silent = FALSE)
```
Arguments

- **object**: An object of class `report_annual-class`
- **dc**: A numeric or integer, the code of the dc, coerced to integer, see `choice_c.ref_dc-method`
- **taxa**: Either a species name in latin or the SANDRE code for species (ie 2038=Anguilla anguilla), it should match the ref.tr_taxon_tax referential table in the stacom database, see `choice_c.ref_taxa-method`
- **stage**: A stage code matching the ref.tr_stade_developpement_std table in the stacom database, see `choice_c.ref_stage-method`
- **start_year**: The starting the first year, passed as character or integer
- **end_year**: The finishing year
- **silent**: Boolean, if TRUE, information messages are not displayed

Value

An object of class `report_annual-class` with data selected

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>

Description

The `choice_c` method fills in the data slot for ref_dc, and then uses the `choice_c` methods of these object to "select" the data.

Usage

```r
## S4 method for signature 'report_dc'
choice_c(object, dc, horodatedebut, horodatefin, silent = FALSE)
```

Arguments

- **object**: An object of class `ref_dc-class`
- **dc**: The dc to set
- **horodatedebut**: A POSIXt or Date or character to fix the date of beginning of the report
- **horodatefin**: A POSIXt or Date or character to fix the last date of the report
- **silent**: Should program be silent or display messages
choice_c,report_df-method

Value
An object of class ref_dc-class with data selected

Author(s)
Cedric Briand <cedric.briand@eptb-vilaine.fr>

command line interface for report_df class

Description
The choice_c method fills in the data slot for ref_df, and then uses the choice_c methods of these object to "select" the data.

Usage
## S4 method for signature 'report_df'
choice_c(object, df, horodatedebut, horodatefin, silent = FALSE)

Arguments

object An object of class ref_df-class
df The df to set
horodatedebut A POSIXt or Date or character to fix the date of beginning of the report
horodatefin A POSIXt or Date or character to fix the last date of the report
silent Should program be silent or display messages

Value
An object of class ref_df-class with data selected

Author(s)
Cedric Briand <cedric.briand@eptb-vilaine.fr>
Description

The choice_c method fills in the data slot for ref_env-class by running the charge method of this object. It then runs the choice method on this object. It also applies the choice method for objects of class ref_horodate-class.

Usage

```r
## S4 method for signature 'report_env'
choice_c(object, stationMesure, datedebut, datefin, silent = FALSE)
```

Arguments

- `object`: An object of class report_env-class.
- `stationMesure`: A character, the code of the monitoring station, which records environmental parameters.
- `datedebut`: The starting date as a character, formats like `%Y-%m-%d` or `%d-%m-%Y` can be used as input.
- `datefin`: The finishing date of the report, for this class this will be used to calculate the number of daily steps.
- `silent`: Boolean default FALSE, if TRUE information messages not displayed.

Value

An object of class report_env-class with data selected.

Author(s)

Cedric Briand <cedric.briand@epbt-vilaine.fr>
Usage

```r
## S4 method for signature 'report_ge_weight'
choice_c(object, dc, start_year, end_year, selectedvalue, silent = FALSE)
```

Arguments

- `object`: An object of class `report_ge_weight-class`
- `dc`: A numeric or integer, the code of the dc, coerced to integer, see `choice_c,ref_dc-method`
- `start_year`: The starting the first year, passed as character or integer
- `end_year`: The finishing year, must be > start_year (minimum one year in August to the next in August)
- `selectedvalue`: A character to select and object in the `ref_list-class`
- `silent`: Boolean, if TRUE, information messages are not displayed

Value

An object of class `report_ge_weight-class` with data selected. The `choice_c` method fills in the data slot for classes `ref_dc-class ref_year-class ref_coe-class ref_list-class`

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>

choice_c,report_mig-method

command line interface for report_mig class

Description

The `choice_c` method fills in the data slot for `ref_dc`, `ref_taxa`, `ref_stage`, and `ref_ref_timestep_daily` and then uses the `choice_c` methods of these objects to select the data.

Usage

```r
## S4 method for signature 'report_mig'
choice_c(object, dc, taxa, stage, datedebut, datefin)
```

Arguments

- `object`: An object of class `report_mig-class`
- `dc`: A numeric or integer, the code of the dc, coerced to integer, see `choice_c,ref_dc-method`
- `taxa`: Either a species name in Latin or the SANDRE code for species (ie 2038=Anguilla anguilla), these should match the `ref.tr_taxon_tax` referential table in the stacomi database, see `choice_c,ref_taxa-method`
stage A stage code matching the ref.tr_statedevelopment_std table in the stacomi database see choice_c,ref_stage-method
datedebut The starting date as a character, formats like %Y-%m-%d or %d-%m-%Y can be used as input
datefin The finishing date of the report, for this class this will be used to calculate the number of daily steps.

Value
An object of class report_mig-class with data selected

Author(s)
Cedric Briand <cedric.briand@eptb-vilaine.fr>

command line interface for report_mig_char class

choose_c,report_mig_char-method
S4 method for signature 'report_mig_char'

choose_c(
  object,
  dc,
  taxa,
  stage,
  parquan = NULL,
  parqual = NULL,
  horodatedebut,
  horodatefin,
  echantillon = c("with", "without"),
  silent = FALSE
)

Arguments

object An object of class report_mig_char-class
dc A numeric or integer, the code of the dc, coerced to integer,see choice_c,ref_dc-method
taxa '2220=Salmo salar', can be a vector with several values these should match the ref.tr_taxon_tax referential table in the stacomi database, see choice_c,ref_taxa-method
The stages selected, can be a vector with several values

- **parquan**: Quantitative parameter
- **parqual**: Qualitative parameter
- **horodatedebut**: The starting date as a character, formats like `%Y-%m-%d` or `%d-%m-%Y` can be used as input
- **horodatefin**: The finishing date of the report, for this class this will be used to calculate the number of daily steps
- **echantillon**: 'with' can be 'without', checking without modifies the query in the connect method so that subsamples are not allowed
- **silent**: Default FALSE, if TRUE the program should no display messages

### Value

An object of class `report_sea_age-class` The choice_c method fills in the data slot for classes `ref_dc-class`, `ref_taxa-class`, `ref_stage-class`, `ref_par-class` and two slots of `ref_horodate-class` and then uses the choice_c methods of these object to select the data.

### Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>
**Arguments**

- **object** An object of class `report_env-class`
- **dc** A numeric or integer, the code of the dc, coerced to integer, see `choice_c.ref_dc-method`
- **taxa** Either a species name in latin or the SANDRE code for species (ie 2038=Anguilla anguilla), it should match the ref.tr_taxon_tax referential table in the stacomii database, see `choice_c.ref_taxa-method`
- **stage** 'AGJ=Yellow eel', 'AGG=Silver eel', 'CIV=glass eel'
- **stationMesure** A character, the code of the monitoring station, which records environmental parameters, see `choice_c.ref_env-method`
- **datedebut** The starting date as a character, formats like `%Y-%m-%d` or `%d-%m-%Y` can be used as input
- **datefin** The finishing date of the report, for this class this will be used to calculate the number of daily steps.
- **silent** Boolean default FALSE, if TRUE information messages not displayed.

**Value**

An object of class `report_env-class` with data selected

**Author(s)**

Cedric Briand <cedric.briand@eptb-vilaine.fr>
stage  A stage code matching the ref.tr_stadedeveloppement_std table in the stacomi database, see choice_c,ref_stage-method
start_year  The starting the first year, passed as character or integer
end_year  the finishing year
silent  Boolean, if TRUE, information messages are not displayed

Value
An object of class report_mig_interannual-class with data selected. The choice_c method fills in the data slot for classes ref_dc-class, ref_taxa-class, ref_stage-class and two slots of ref_year-class

Author(s)
Cedric Briand <cedric.briand@eptb-vilaine.fr>

Description
The choice_c method fills in the data slot for ref_dc, ref_taxa, ref_stage and then uses the choice_c methods of these object to 'select' the data.

Usage
```r
## S4 method for signature 'report_mig_mult'
choice_c(object, dc, taxa, stage, datedebut, datefin, silent = FALSE)
```

Arguments
- **object**  An object of class report_mig-class
- **dc**  A numeric or integer, the code of the dc, coerced to integer, see choice_c,ref_dc-method
- **taxa**  Either a species name in latin or the SANDRE code for species (ie 2038=Anguilla anguilla), these should match the ref.tr_taxon_tax referential table in the stacomi database, see choice_c,ref_taxa-method
- **stage**  A stage code matching the ref.tr_stadedeveloppement_std table in the stacomi database see choice_c,ref_stage-method
- **datedebut**  The starting date as a character, formats like %Y-%m-%d or %d-%m-%Y can be used as input
- **datefin**  The finishing date of the report, for this class this will be used to calculate the number of daily steps.
- **silent**  Should messages be hidet default FALSE
Value

An object of class `report_mig_mult-class` with data selected

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>

---

Description

```
#' The choice_c method fills in the data slot for classes ref_dc-class, ref_taxa-class, ref_stage-class,
ref_par-class and two slots of ref_horodate-class and then uses the choice_c methods of these object
to select the data.
```

Usage

```r
## S4 method for signature 'report_sample_char'
choice_c(
  object,
  dc,
  taxa,
  stage,
  par,
  horodatedebut,
  horodatefin,
  silent = FALSE
)
```

Arguments

- `object` An object of class `report_sample_char-class`
- `dc` A numeric or integer, the code of the dc, coerced to integer, see `choice_c,ref_dc-method`
- `taxa` Either a species name in latin or the SANDRE code for species (ie 2038=Anguilla anguilla), these should match the ref.tr_taxon_tax referential table in the stacomi database, see `choice_c,ref_taxa-method`
- `stage` A stage code matching the ref.tr_statedeveloppement_std table in the stacomi database, see `choice_c,ref_stage-method`
- `par` A parameter matching th ref.tg_parametre_par table in the stacomi database, see `choice_c,ref_par-method`
- `horodatedebut` The starting date as a character, formats like %Y-%m-%d or %d-%m-%Y can be used as input
The finishing date of the report, for this class this will be used to calculate the number of daily steps.

Boolean, if TRUE, information messages are not displayed

An object of class report_mig-class with data selected

Cedric Briand <cedric.briand@eptb-vilaine.fr>
choice_c,report_silver_eel-method

par Parameters chosen for the report are measured body size (1786), measured fork length (1785), video size (C001) and number of year at sea (A124)

horodatedebut The starting date as a character, formats like %Y-%m-%d or %d-%m-%Y can be used as input

horodatefin The finishing date of the report, for this class this will be used to calculate the number of daily steps.

limit1hm Size limit of a salmon for an one sea winter fish

limit2hm Size limit of a salmon for a two sea winter fish

silent Default FALSE, if TRUE the program should no display messages

Value

An object of class report_sea_age-class

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>

choice_c,report_silver_eel-method

command line interface for report_silver_eel class

Description

#' The choice_c method fills in the data slot for classes ref_dc-class, ref_taxa-class, ref_stage-class, ref_par-class and two slots of ref_horodate-class and then uses the choice_c methods of these object to select the data.

Usage

## S4 method for signature 'report_silver_eel'
choice_c(
  object,
  dc,
  taxa = 2038,
  stage = "AGG",
  par = c("1786", "CCCC", "BBBB", "CONT", "LINP", "A111", "PECT"),
  horodatedebut,
  horodatefin,
  silent = FALSE
)
Arguments

- **object**: An object of class `report_silver_eel-class`
- **dc**: A numeric or integer, the code of the dc, coerced to integer, see `choice_c.ref_dc-method`
- **taxa**: '2038=Anguilla anguilla’, these should match the ref.tr_taxon_tax referential table in the stacomi database, see `choice_c.ref_taxa-method`
- **stage**: 'AGG'
- **par**: Parameters chosen for the report are body size (1786), vertical eye diameter (BBBB), horizontal eye diameter (CCCC), body contrast (CONT), presence of punctuation on the lateral line (LINP), length of the pectoral fin (PECT)
- **horodatedebut**: The starting date as a character, formats like %Y-%m-%d or %d-%m-%Y can be used as input
- **horodatefin**: The finishing date of the report, for this class this will be used to calculate the number of daily steps.
- **silent**: Boolean, if TRUE, information messages are not displayed

Value

An object of class `report_mig-class`

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>

Description

command line interface for `report_species-class`

Usage

```r
# S4 method for signature 'report_species'
choice_c(
  object,
  dc,
  taxa = "all",
  split = "none",
  start_year,
  end_year,
  silent = FALSE
)
```
**Arguments**

- **object**: An object of class `report_species-class`
- **dc**: A numeric or integer, the code of the dc, coerced to integer, see `choice_c.ref_dc-method`
- **taxa**: Either 'all' (default) or a species name in latin or the SANDRE code for species (ie 2038=Anguilla anguilla), it should match the ref.tr_taxon_tax referential table in the stacomi database, see `choice_c.ref_taxa-method`
- **split**: one of c('none','week','month','year')
- **start_year**: The starting the first year, passed as character or integer
- **end_year**: the finishing year
- **silent**: Boolean, if TRUE, information messages are not displayed

**Value**

An object of class `report_species-class` with data selected

**Author(s)**

Cedric Briand <cedric.briand@eptb-vilaine.fr>

---

**Description**

Classification scores are calculated by multiplying the metrics $BL = \text{body length}$, $W = \text{weight}$, $MD = \text{mean eye diameter (Dv+Dh)/2}$, and $FL = \text{length of the pectoral fin}$, with each parameter $p$ as $S=\text{Constant}+BL*p(Bl)+W*p(W)...$ The stage chosen is the one achieving the highest score

**Usage**

`coef_durif`

**Format**

An object of class `matrix` (inherits from `array`) with 5 rows and 6 columns.

**References**

colortable  

Builds a table with colors to merge with a dataframe for later use in ggplot. An initial check will be done on the name of the color vector. A data frame is built. It contains a column color which is a factor. The factor order match the order of the vector (not the alphabetical order of the colors).

Description

Builds a table with colors to merge with a dataframe for later use in ggplot. An initial check will be done on the name of the color vector. A data frame is built. It contains a column color which is a factor. The factor order match the order of the vector (not the alphabetical order of the colors).

Usage

```r
colortable(
  color = NULL,
  vec,
  palette = "Set2",
  color_function = c("brewer.pal", "gray.colors", "random")
)
```

Arguments

- `color`: Either null (default) or a named vector of colors, the names should correspond to the values of vec.
- `vec`: The vector to match the color with, if a named vector or color is supplied the names should match.
- `palette`: the name of the RColorBrewer palette, defaults to "Set2", ignored for other color gradient functions and if a named vector of colors is provided.
- `color_function`: the name of the function used to brew the colors, one for "brewer.pal", "gray.colors", "random", default to "brewer.pal", this argument is ignored if a named vector of color is passed.

Value

A dataframe with two columns, the vector (name) and the color (color) as a reordered factor.

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>
connect,report_annual-method
connect method for report_annual class this method performs the sum over the year attention this function does not count subsamples.

Description
connect method for report_annual class this method performs the sum over the year attention this function does not count subsamples.

Usage
## S4 method for signature 'report_annual'
connect(object, silent = FALSE)

Arguments
object An object of class report_annual-class
silent Stops messages from being displayed if silent=TRUE, default FALSE

Value
An instantiated object with values filled with user choice
An object of class report_annual-class including a dataframe with column effectif, comprising the sum of report_mig counts

Author(s)
Cedric Briand <cedric.briand@eptb-vilaine.fr>

connect,report_dc-method
connect method for report_dc

Description
loads the working periods and type of arrest or disfunction of the DC

Usage
## S4 method for signature 'report_dc'
connect(object, silent = FALSE)
Arguments

object  An object of class report_dc-class
silent  boolean, default FALSE, if TRUE messages are not displayed

Value

An object of class report_dc-class with slot data filled from the database

Author(s)

cedric.briand
**connect,report_env-method**

*connect method for report_env class*

**Description**

connect method for report_env class

**Usage**

```r
## S4 method for signature 'report_env'
connect(object, silent = FALSE)
```

**Arguments**

- `object`: An object of class `report_env-class`
- `silent`: Default FALSE, if TRUE the program should no display messages

**Value**

An object of class `report_env-class` with slot data filled from the database

**Author(s)**

Cedric Briand <cedric.briand@eptb-vilaine.fr>

---

**connect,report_ge_weight-method**

*connect method for report_Poids_moyen*

**Description**

The connect method adapts queries according to user choices, mean weight w is calculated as car_valeur_quantitatif/lot_effectif. These coefficients are stored in the database, and the connect method loads them from the table using the ref_coe-class

**Usage**

```r
## S4 method for signature 'report_ge_weight'
connect(object, silent = TRUE)
```

**Arguments**

- `object`: An object of class `report_ge_weight-class`
- `silent`: Should the method be silent
Value

An object of class `report_ge_weight-class` with slots data and coe filled from the database

Note

dates for the request are from august to august (a glass eel season)

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>

Description

uses the `report_mig_mult` method

Usage

```r
## S4 method for signature 'report_mig_char'
connect(object, silent = FALSE)
```

Arguments

- `object` An object of class `report_mig_char-class`
- `silent` Boolean default FALSE, if TRUE information messages not displayed

Value

An object of class `report_mig_char-class` with list in `@data$parquan` and `@data$parqual` filled in from the database
connect, report_mig_env-method

connect method for report_mig_env class

Description

connect method for report_mig_env class

Usage

## S4 method for signature 'report_mig_env'
connect(object, silent = FALSE)

Arguments

object
An object of class report_mig_env-class

silent
Default FALSE, if TRUE the program should no display messages

Value

an object of report_mig_env class

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>

connect, report_mig_interannual-method

connect method for report_mig_interannual

Description

This method will check if the data in the t_reportjournalier_bjo table has no missing data, if missing the program will load missing data. As a second step, the program will check if the numbers in the table t_reportjournalier_bjo differ from those in the database, and propose to re-run the report_mig (which has a write_database method to write daily reports) for those years.

Usage

## S4 method for signature 'report_mig_interannual'
connect(object, silent = FALSE, check = TRUE)
connect,report_mig_mult-method

Arguments

- **object**: An object of class `report_mig_interannual-class`
- **silent**: Stops messages from being displayed if silent=TRUE, default FALSE
- **check**: Checks that data are corresponding between report_annual and report_mig

Value

- `report_mig_interannual` an instantiated object with values filled with user choice

Note

We expect different results between daily reports from the `t_reportjournalier_bjo` table and the annual sums from `report_annual` for glass eels as those may have been weighted and not only counted. The `t_reportjournalier_bjo` table used by `report_mig_interannual` contains the sum of glass eel numbers converted from weights and those directly counted. The `report_annual` does not.

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>

Description

This method loads data from the database for `report_mig` but also fills the table of conversion coefficient, if the taxa is eel. It also calls connect method for `report_df-class, report_dc-class` and `report_ope-class` associated with the report and used by the `fungraph` and `fungraph_glasseseel` functions. As a side effect it assigns objects `report_dc-class, report_df-class` and `report_ope-class` in environment `envir_stacomi`

Usage

```r
## S4 method for signature 'report_mig_mult'
connect(object, silent = FALSE)
```

Arguments

- **object**: An object of class `report_mig_mult-class`
- **silent**: Boolean, if TRUE messages are not displayed

Value

- An object of class `report_mig_mult-class` with slot @data filled from the database
### connect,report_ope-method

**connect method for report_ope**

#### Description

connect method for report_ope

#### Usage

```r
## S4 method for signature 'report_ope'
connect(object, silent = FALSE)
```

#### Arguments

- **object**
  - An object of class `report_ope-class` load data from the operation table, one dataset per DC
- **silent**
  - Boolean, TRUE removes messages.

#### Value

An object of class `report_ope-class` with slot @data filled

#### Author(s)

cedric.briand

---

### connect,report_sample_char-method

**connect method for report_sample_char**

#### Description

connect method for report_sample_char

#### Usage

```r
## S4 method for signature 'report_sample_char'
connect(object, silent = FALSE)
```

#### Arguments

- **object**
  - An object of class `report_sample_char-class`
- **silent**
  - Boolean if TRUE messages are not displayed
connect, report_silver_eel-method

Description

connect method for report_silver_eel

Usage

## S4 method for signature 'report_silver_eel'
connect(object, silent = FALSE)

Arguments

object An object of class report_silver_eel-class
silent Default FALSE, if TRUE the program should no display messages

Value

An object of class report_silver_eel-class with slot data @data filled

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>

connect, report_sea_age-method

connect method for report_sea_age

Description

connect method for report_sea_age

Usage

## S4 method for signature 'report_sea_age'
connect(object, silent = FALSE)

Arguments

object An object of class report_sea_age-class
silent Default FALSE, if TRUE the program should no display messages

Value

An object of class report_sea_age-class with slot data @data filled

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>
**connect,report_species-method**

**Arguments**

object  
An object of class `report_silver_eel-class`

silent  
Boolean if TRUE messages are not displayed

**Value**

An object of class `report_silver_eel-class` with slot data @data filled

**Author(s)**

Cedric Briand <cedric.briand@eptb-vilaine.fr>

---

**connect,report_species-method**

*connect method for report_species*

---

**Description**

connect method for report_species

**Usage**

```r
## S4 method for signature 'report_species'
connect(object, silent = FALSE)
```

**Arguments**

object  
An object of class `report_species`

silent  
Boolean, if TRUE, information messages are not displayed

**Value**

An object of class `report_species-class` with slot data filled with slot data @data filled

**Author(s)**

Cedric Briand <cedric.briand@eptb-vilaine.fr>
envir_stacomi

Environment where most objects from the package are stored and then loaded by the charge method

Description

envir_stacomi envir_stacomi <- new.env(parent = baseenv()) is the environment where most object created by the interface are stored

This is where the graphical interface stores its objects try ls(envir=envir_stacomi)

This is where the graphical interface stores its objects try ls(envir=envir_stacomi)

Usage

envir_stacomi

envir_stacomi

envir_stacomi

Format

An object of class environment of length 0.

An object of class environment of length 0.

An object of class environment of length 0.

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>

fungraph

Function for report_mig graphs including numbers DF DC operations

Description

This graph is for species other than glass eel

Usage

fungraph(
  report_mig,
  tableau,
  time.sequence,
  taxa,
  stage,
fungraph

    dc = NULL,
    silent,
    color = NULL,
    color_ope = NULL,
    ...
)

Arguments

report_mig An object of class report_mig
tableau A data frame with the with the following columns: No.pas,debut_pas,fin_pas,
          ope_dic_identifiant,lot_tax_code,lot_std_code,type_de_quantite,MESURE,CALCULE,
          EXPERT,PONCTUEL,Effectif_total,taux_d_echappement,coe_valeur_coefficient
time.sequence A vector POSIX

taxa The species
stage The stage
dc The DC
silent Message displayed or not
color Default NULL, a vector of color in the following order, working, stopped, 1...5
types of operation for the fishway or DC, measured, calculated, expert, direct
observation. If null will be set to brewer.pal(12,"Paired")[c(8,10,4,6,1,2,3,5,7)]
color_ope Default NULL, a vector of color for the operations. Default to brewer.pal(4,"Paired")
... additional parameters passed to matplot, main, ylab, ylim, lty, pch, bty, cex.main,
it is currently not a good idea to change xlim (numbers are wrong, the month plot
covers all month, and legend placement is wrong

Value

No return value, called for side effects

Note

this function is intended to be called from the plot method in report_mig_mult and report_mig

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>
fungraph_glasseel  

Graph function for glass eel migration. Differs from fungraph as it does not draw the ggplot graph for month

Description

This graph will also plot numbers and bars according to whether the glass eel have been counted through weight or numbers

Usage

fungraph_glasseel(
  report_mig,
  table,
  time.sequence,
  taxa,
  stage,
  dc = null,
  silent,
  color = NULL,
  color_ope = NULL,
  ...
)

Arguments

report_mig  an object of class report_mig-class or an object of class report_mig_mult-class  
table  a data frame with the results  
time.sequence  a vector POSIXt  
taxa  the species  
stage  the stage  
dc  the counting device, default to null, only necessary for report_mig_mult-class  
silent  Message displayed or not  
color  Default NULL, a vector of length 11 of color in the following order, numbers, weight, working, stopped, 1...5 types of operation, the 2 latest colors are not used but kept for consistency with fungraph for the fishway, if null will be set to brewer.pal(12,"Paired“)[c(4,6,1,2,3,5,7,8,10,11,12)]  
color_ope  Default NULL, a vector of color for the operations. Default to brewer.pal(4,"Paired")  
...  additional parameters passed to plot, main, ylab, cex.main, font.main, type, xlim, ylim, lty, bty, pch it is not possible to change xlim

Value

No return value, called for side effects
funstat

Author(s)
Cedric Briand <cedric.briand@eptb-vilaine.fr>

funstat Function to calculate statistics per month

Description
Function to calculate statistics per month

Usage
funstat(tableau, time.sequence, taxa, stage, DC, silent)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>tableau</td>
<td>A table with the following columns: No.pas,debut_pas,fin_pas,ope_dic_identifiant,lot_tax_code,lot_std_code,type_de_quantite,MESURE,CALCULE,EXPERT,PONCTUEL,Effectif_total,taux_d_echappement,coe_valeur_coefficient</td>
</tr>
<tr>
<td>time.sequence</td>
<td>Passed from report_mig or report_mig_mult</td>
</tr>
<tr>
<td>taxa</td>
<td>Taxa</td>
</tr>
<tr>
<td>stage</td>
<td>The Stage</td>
</tr>
<tr>
<td>DC</td>
<td>The counting device</td>
</tr>
<tr>
<td>silent</td>
<td>Message displayed or not</td>
</tr>
</tbody>
</table>

Value
No return value, called for side effects

Note
this function is intended to be called from within the summary method

Author(s)
Cedric Briand <cedric.briand@eptb-vilaine.fr>
funtable  

function to print and save statistics in .csv and .html formats for report_mig and report_mig_mult class

Description

function to print and save statistics in .csv and .html formats for report_mig and report_mig_mult class

Usage

funtable(tableau, time.sequence, taxa, stage, DC, resum, silent)

Arguments

tableau A table with the following columns: No.pas,debut_pas,fin_pas, ope_dic_identifiant,lot_tax_code,lot_std_code,type_de_quantite,MESURE,CALCULE,EXPERT,PONCTUEL,Effectif_total,taux_d_echappement,coe_valeur_coefficient
time.sequence Passed from report_mig or report_mig_mult
taxa Taxa
stage The Stage
DC The counting device
resum A summary table generated by funstat
silent If TRUE, all messages turned off (except warnings and errors)

Value

No return value, called for side effects

Note

this function is intended to be called from within the summary method

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>
fun_aggreg_for_plot  Calculates a data.frame where all components within the list calcdata are aggregated and formatted for plot

Description
Calculates a data.frame where all components within the list calcdata are aggregated and formatted for plot

Usage
fun_aggreg_for_plot(object)

Arguments
object  An object of class report_mig_mult-class

Value
A data.frame

Author(s)
Cedric Briand <cedric.briand@eptb-vilaine.fr>

fun_char_spe  function used to remove special non utf8 character which cause the gtk interface to crash

Description
function used to remove special non utf8 character which cause the gtk interface to crash

Usage
fun_char_spe(text)

Arguments
text  a text string which might contain no utf8 characters

Value
text

Author(s)
Cedric Briand <cedric.briand@eptb-vilaine.fr>
fun_date_extraction

This function extracts temporal characteristics from a dataframe

Description

This function extracts temporal characteristics from a dataframe

Usage

fun_date_extraction(
  data,
  nom_coldt,
  annee = TRUE,
  mois = TRUE,
  quinzaine = FALSE,
  semaine = TRUE,
  semaine_std = FALSE,
  jour_an = FALSE,
  jour_mois = TRUE,
  heure = FALSE
)

Arguments

data a data frame containing a Date or POSIXt column
nom_coldt the name of the column containing date or POSIXt entry to be processed
annee logical do you want a column describing year to be added to the dataframe
mois logical, add column with month
quinzaine logical, add column with 15 days
semaine logical, add column with weeks
semaine_std logical, add column with standard weeks (using isoweek from lubridate)
jour_an logical, add column with day of year
jour_mois logical, add column with day of month
heure logical, add column with hour

Value

The dataframe with date column filled

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>
fun_report_mig_interannual

statistics per time period

Description

function called for report_mig_mult objects renames columns replaces nulls, and calculates reports with time period larger than day

Usage

fun_report_mig_interannual(dat, annee = NULL, timesplit = NULL)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dat</td>
<td>a data frame with columns (&quot;bjo_annee&quot;,&quot;bjo_jour&quot;,&quot;bjo_labelquantite&quot;,&quot;bjo_valeur&quot;)</td>
</tr>
<tr>
<td>annee</td>
<td>The year to exclude from the historical series (it will be plotted against the historical series)</td>
</tr>
<tr>
<td>timesplit</td>
<td>&quot;week&quot; &quot;2 week&quot; &quot;month&quot; as provided to seq.POSIXt, default NULL</td>
</tr>
</tbody>
</table>

Value

a data frame with mean, max, and min calculated for each timesplit

fun_report_mig_mult

Calculate daily migration by simple repartition

Description

Function to calculate daily migration from migration monitoring whose length is less than one day, typically video recording whose period are instant events.

Usage

fun_report_mig_mult(time.sequence, datasub, negative = FALSE)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>time.sequence</td>
<td>the time sequence to be filled in with new data</td>
</tr>
<tr>
<td>datasub</td>
<td>the initial dataset</td>
</tr>
<tr>
<td>negative</td>
<td>'boolean', default FALSE, TRUE indicates a separate sum for negative and positive migrations</td>
</tr>
</tbody>
</table>
Description
Function to calculate daily migration from migration monitoring whose length is more than one day, this calculation relies on the (false) assumption that migration is evenly spread over time.

Usage
fun_report_mig_mult_overlaps(time.sequence, datasub, negative = FALSE)

Arguments
- time.sequence: the time sequence to be filled in with new data
- datasub: the initial dataset
- negative: 'boolean', default FALSE, TRUE indicates a separate sum for negative and positive migrations to time.sequence period and summed over the new sequence. A migration operation spanning several days will be converted to 'daily' values assuming that the migration was regular over time. The function returns one row per taxa, stages, counting device. It does not account for the destination of taxa. It returns separate rows for quantities and numbers. Several columns are according to the type of measure (MESURE, CALCULE, PONCTUEL or EXPERT).

Value
A data.frame with daily migrations

Author(s)
Cedric Briand <cedric.briand@eptb-vilaine.fr>

See Also
calcule.report_mig_mult-method
fun_schema

**fun_schema**\n
*Creates a list of available schemas in the db*

**Description**

Creates a list of available schemas in the db

**Usage**

`fun_schema()`

**Value**

A table with data providers with org_code, the user of each schema, and org_description the description of the schema

---

fun_stage_durif

**fun_stage_durif**\n
*Function to calculate the stages from Durif*

**Description**

Function to calculate the stages from Durif

**Usage**

`fun_stage_durif(data)`

**Arguments**

data\n
A dataset with columns BL, W, Dv, Dh, FL corresponding to body length (mm), Weight (g), vertical eye diameter (mm), vertical eye diameter (mm), and pectoral fin length (mm)

**Value**

A data.frame with durif stages per individual

**Author(s)**

Laurent Beaulaton <laurent.beaulaton@onema.fr>
fun_table_per_dis

functions called in DF and DC

Description

functions called in DF and DC

Usage

fun_table_per_dis(
  typeperiode,
  tempsdebut,
  tempsfin,
  libelle,
  color,
  date = TRUE
)

Arguments

typeperiode ref.tr_typearretdisp_tar(per_tar_code) the code of the period (see table ref.tr_typearretdisp_tar)
tempsdebut ref.tr_typearretdisp_tar(per_date_debut) starting timestamp of the period
tempsfin The postgres column ref.tr_typearretdisp_tar(per_date_fin) ending timestamp of the period
libelle The postgres column ref.tr_typearretdisp_tar(libelle )description of the period
color A named vector of color matching libelle.
date Boolean, should the function return a POSIXt or date value

Value

A list

Note

returns either POSIXt or date if date=TRUE

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>
**fun_weight_conversion**

returns a table where weights and number are calculated from number and weights respectively. Performs a query to collect the conversion coefficients.

**Description**

returns a table where weights and number are calculated from number and weights respectively. Performs a query to collect the conversion coefficients.

**Usage**

fun_weight_conversion(tableau, time.sequence, silent)

**Arguments**

- **tableau**: Table issued from report_mig
- **time.sequence**: Time sequence from report_mig
- **silent**: If silent=TRUE do not display messages

**Value**

tableau, the data frame with weight converted to numbers

**Author(s)**

Cedric Briand <cedric.briand@eptb-vilaine.fr>

---

**fun_write_monthly**

This writes monthly data in `t_reportmensuel_mens` table

**Description**

This writes monthly data in `t_reportmensuel_mens` table

**Usage**

fun_write_monthly(report_mig, resum, silent)

**Arguments**

- **report_mig**: an object of class `report_mig`
- **resum**: data frame with summary per month
- **silent**: Suppresses messages
Value
No return value, called for side effects

Note
This function is launched by fun_write_daily, the resum dataset is created by the funstat function

---

### getvalue

Generic method `getvalue`

#### Description

Generic method `getvalue`

#### Usage

`getvalue(object, ...)`

#### Arguments

- `object`: Object
- `...`: Additional parms

#### Author(s)

cedric.briand

---

### getvalue, ref_period-method

*Returns the POSIXt value of a given name*

#### Description

*Returns the POSIXt value of a given name*

#### Usage

```r
## S4 method for signature 'ref_period'
getvalue(object, id)
```

#### Arguments

- `object`: An object of class `ref_period-class`
- `id`: one of ‘jour’, ‘semaine’, ‘quinzaine’, ‘mois’
graphdate

Value

A character to be used in seq.POSIXt

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>

Examples

## Not run:
getvalue(new('ref_period'), 'quinzaine')

## End(Not run)

Description

function used for some lattice graphs with dates

Usage

graphdate(vectordate)

Arguments

vectordate date or POSIXt

Value

vectordate (without class)

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>
model, report_ge_weight-method

Description

model method for report_ge_weight' this method uses samples collected over the season to model the variation in weight of glass eel or yellow eels.

Usage

## S4 method for signature 'report_ge_weight'
model(object, model.type = "seasonal", silent = FALSE)

Arguments

- object: An object of class report_ge_weight-class
- model.type: default 'seasonal', 'seasonal1','seasonal2','manual'.
- silent: Default FALSE, if TRUE the program should no display messages
Details

Depending on model.type several models are produced

- model.type='seasonal'. The simplest model uses a seasonal variation, it is fitted with a sine wave curve allowing a cyclic variation \( w \sim a \cos(2\pi d'/T)/365)+b \) with a period T. The modified day \( d' \) used in this model is set at 1 the 1st of august doy = \( d' + d0; d0 = 212, \) doy=julian days

- model.type='seasonal1'. A time component is introduced in the model, which allows for a long term variation along with the seasonal variation. This long term variation is fitted with a gam, the time variable is set at zero at the beginning of the first day of observed values. The seasonal variation is modeled on the same modified julian time as model.type='seasonal' but here we use a cyclic cubic spline cc, which allows to return at the value of d0=0 at d=365. This model was considered as the best to model size variations by Diaz & Briand in prep. but using a large set of values over years.

- model.type='seasonal2'. The seasonal trend in the previous model is now modeled with a sine curve similar to the sine curve used in seasonal. The formula for this is \( \sin(\omega vt) + \cos(\omega vt) \), where vt is the time index variable \( \omega \) is a constant that describes how the index variable relates to the full period (here, \( 2\pi/365 = 0.0172 \)). The model is written as following \( w \cos(0.0172 * \text{doy}) + \sin(0.0172 * \text{doy}) + s(\text{time}). \)

- model.type='manual'. The dataset don (the raw data), coe (the coefficients already present in the database, and newcoe the dataset to make the predictions from, are written to the environment envir_stacomi. please see example for further description on how to fit your own model, build the table of coefficients, and write it to the database.

Value

An object of class report_ge_weight-class with @calcdata[['import_coe']] filled.

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>
Arguments

x  An object of class report_annual-class
plot.type  Default point
silent  Stops displaying the messages.
  • plot.type="point": ggplot+geom_point'

Value

No return value, called for side effects

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>

See Also

report_mig_interannual-class for examples

Description

• plot.type=1A barplot of the operation time per month
• plot.type=2Barchat giving the time per type of operation
• plot.type=2Rectangle plots drawn along a line
• plot.type=4Plots per day drawn over the period to show the operation of a df, days in x, hours in y

Usage

## S4 method for signature 'report_dc,missing'
plot(
  x,
  plot.type = 1,
  silent = FALSE,
  main = NULL,
  color_type_oper = c("Fonc normal" = "#76BEBE", "Arr ponctuel" = "#FF6700",
  "Arr maint" = "#9E0142", "Dysfonc" = "#EE1874", "Non connu" = "#999999"),
  color_etat = c("TRUE" = "#0F313A", "FALSE" = "#CEB99A")
)
Arguments

x  An object of class report_dc-class.
plot.type  1 to 4, barplot, barchart, rectangle plot and box showing details of daily operation, a plot with adjacent rectangles.
silent  Stops displaying the messages default to FALSE
main  The title of the graph, if NULL a default title will be plotted with the number of the DF.
color_type_oper  Named vector of color for the graph, must match type operation default to c("Fonc normal" = "#76BEBE", "Arr ponctuel" = "#FF6700", "Arr maint" = "#9E0142", "Dysfonc" = "#EE1874", "Non connu" = "#999999").
color_etat  Named vector state value (must match the names "TRUE", "FALSE").

Value

Nothing but prints the different plots.

Note

The program cuts periods which overlap between two month. The splitting of different periods into month is assigned to the envir_stacomi environment.

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>

plot,report_df,missing-method

Different plots for report_df

Description

- plot.type=1 A barplot of the operation time per month
- plot.type=2 Barchat giving the time per type of operation
- plot.type=2 Rectangle plots drawn along a line
- plot.type=4 Plots per day drawn over the period to show the operation of a df, days in x, hours in y
Usage

```r
## S4 method for signature 'report_df,missing'
plot(
  x,
  plot.type = 1,
  silent = FALSE,
  main = NULL,
  color_type_oper = c("Fonc normal" = "#1B9E77", "Arr ponctuel" = "#E6AB02",
                      "Arr maint" = "#9E0142", Dysfonc = "#E41A1C", "Non connu" = "#999999"),
  color_etat = c("TRUE" = "chartreuse3", "FALSE" = "orangered3")
)
```

Arguments

- **x**: An object of class `report_df-class`
- **plot.type**: 1 to 4.
- **silent**: Stops displaying the messages.
- **main**: The title of the graph, if NULL a default title will be plotted with the number of the DF.
- **color_type_oper**: Named vector of color for the graph, must match type operation default to `c("Fonc normal" = "#1B9E77", "Arr ponctuel" = "#E6AB02", "Arr maint" = "#9E0142", Dysfonc = "#E41A1C", "Non connu" = "#999999").`
- **color_etat**: Named vector state value (must match the names "TRUE", "FALSE").

Value

Nothing but prints the different plots.

Note

The program cuts periods which overlap between two month. The splitting of different periods into month is assigned to the `envir_stacomi` environment.

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>

Description

Plot method for `report_env`
#### Usage

```r
## S4 method for signature 'report_env,missing'
plot(x, silent = FALSE)
```

#### Arguments

- **x**
  - An object of class `report_env-class`
- **silent**
  - Stops displaying the messages

#### Value

Nothing, called for its side effect of plotting data

#### Author(s)

Cedric Briand &lt;cedric.briand@eptb-vilaine.fr&gt;

---

### Description

Plot method for `report_ge_weight`

#### Usage

```r
## S4 method for signature 'report_ge_weight,missing'
plot(x, plot.type = 1, silent = FALSE)
```

#### Arguments

- **x**
  - An object of class `report_ge_weight-class`
- **plot.type**
  - Default '1'. '1' plot of mean weight of glass eel against the mean date of operation (halfway between start, and end of operation). The ggplot 'p' can be accessed from `envir_stacomi` using `get('p', envir_stacomi)`. '2' standard plot of current coefficient. '3' same as '1' but with size according to number.
- **silent**
  - Stops displaying the messages

#### Value

Nothing, called for its side effect of plotting data

#### Note

the model method provides plots for the fitted models
plot, report_mig, ANY-method

Plots of various type for report_mig.

Description

- plot.type="standard" calls fungraph and fungraph_glasseel functions to plot as many "report_mig" as needed, the function will test for the existence of data for one dc, one taxa, and one stage
- plot.type="step" creates Cumulated graphs for report_mig_mult. Data are summed per day for different dc taxa and stages
- plot.type="multiple" Method to overlay graphs for report_mig_mult (multiple dc/taxa/stage in the same plot)

Usage

```r
## S4 method for signature 'report_mig,ANY'
plot(
  x,
  y,
  plot.type = "standard",
  color = NULL,
  color_ope = NULL,
  silent = FALSE,
  ...
)
```

Arguments

- `x` An object of class report_mig
- `y` From the formals but missing
- `plot.type` One of "standard", "step". Default to standard the standard report_mig with dc and operation displayed, can also be step or multiple
- `color` Default NULL, argument passed for the plot.type="standard" method. A vector of color in the following order: (1) working, (2) stopped, (3:7) 1...5 types of operation, (8:11) numbers, weight, NULL, NULL (if glass eel), (8:11) measured, calculated, expert, direct observation for other taxa. If null will be set to brewer.pal(12,"Paired")[c(8,10,4,6,1,2,3,5,7)]
- `color_ope` Default NULL, argument passed for the plot.type="standard" method. A vector of color for the operations. Default to brewer.pal(4,"Paired")
- `silent` Stops displaying the messages.
- `...` Additional arguments passed to matplot or plot if plot.type="standard", see ... in fungraph_glasseel and fungraph
Value

Nothing, called for its side effect

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>

Description

plot method for report_mig_char

Usage

## S4 method for signature 'report_mig_char,missing'
plot(x, color_parm = NULL, plot.type = "qual", silent = FALSE, ...)

Arguments

- **x**: An object of class report_mig_char
- **color_parm**: A named vector for the colors of either parameters (if plot.type=quant) or levels for parameters (if plot.type=qual).
- **plot.type**: One of 'qual', 'quant' 'crossed' default to qual
- **silent**: Boolean default FALSE, if TRUE information messages not displayed
- **...**: Additional parameters

Value

Nothing, called for its side effect of plotting data

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>
### plot,report_mig_env,missing-method

*Plot method for report_mig_env*

**Description**

Plot method for report_mig_env

**Usage**

```r
## S4 method for signature 'report_mig_env,missing'
plot(x, color_station = NULL, color_dc = NULL, silent = FALSE)
```

**Arguments**

- `x`: An object of class `report_mig_env`
- `color_station`: A named vector of station color (e.g. `c("temp_gabion"="red","coef_maree"="blue","phases_lune"="green")`)
  
  Default: null
- `color_dc`: A named vector giving the color for each dc default null (e.g. `c("5"="#4D4D4D","6"="#E6E6E6","12"=#E6E6E6")`)
- `silent`: Stops displaying the messages.

**Value**

Nothing, called for its side effect of plotting

**Author(s)**

Cedric Briand <cedric.briand@eptb-vilaine.fr>

### plot,report_mig_interannual,missing-method

*Plot method for report_mig_interannual*

**Description**

Several of these plots are scaled against the same year, i.e. the comparison is based on year 2000, meaning that day 1 would correspond to the first date of 2000, which is also a saturday, the last day of the week.

**Usage**

```r
## S4 method for signature 'report_mig_interannual,missing'
plot(x, plot.type = "standard", timesplit = "mois", silent = FALSE)
```
Arguments

x An object of class report_mig_interannual-class
plot.type Default standard
timesplit Used for plot.type barchart or dotplot, Default mois (month) other possible values are semaine (week), quinzaime (2 weeks), English values within parenthesis are also accepted.
silent Stops displaying the messages.
  • plot.type="line": one line per daily report_mig
  • plot.type="standard": the current year is displayed against a ribbon of historical values"
  • plot.type="density": creates density plot to compare seasonality, data computed by 15 days period
  • plot.type="step" : creates step plots to compare seasonality, the year chosen in the interface is the latest if silent=TRUE, or it can be selected in the droplist. It is highlighted against the other with a dotted line
  • plot.type="barchart": comparison of daily migration of one year against periodic migration for the other years available in the chronicle, different periods can be chosen with argument timesplit
  • plot.type="pointrange": Pointrange graphs, different periods can be chosen with argument timesplit
  • plot.type="seasonal": plot to display summary statistics about the migration period

Value

Nothing, called for its side effect of plotting

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>

plot,report_mig_mult,missing-method

Plots of various type for report_mig_mult

Description

• plot.type='standard' calls fungraph and fungraph_glasseel functions to plot as many 'report_mig' as needed, the function will test for the existence of data for one dc, one taxa, and one stage
• plot.type='step' creates Cumulated graphs for report_mig_mult. Data are summed per day for different dc taxa and stages
• plot.type='multiple' Method to overlay graphs for report_mig_mult (multiple dc/taxa/stage in the same plot)
Usage

```r
## S4 method for signature 'report_mig_mult,missing'
plot(
  x,
  plot.type = "standard",
  color = NULL,
  color_ope = NULL,
  silent = FALSE,
  ...
)
```

Arguments

- **x**: An object of class `report_mig_mult`
- **plot.type**: One of ‘standard’, ‘step’, ‘multiple’. Default to standard the standard `report_mig` with dc and operation displayed, can also be step or multiple
- **color**: Default NULL, argument passed for the `plot.type='standard'` method. A vector of color in the following order: (1) working, (2) stopped, (3:7) 1...5 types of operation, (8:11) numbers, weight, NULL, NULL (if glass eel), (8:11) measured, calculated, expert, direct observation for other taxa. If null will be set to `brewer.pal(12,'Paired')[c(8,10,4,6,1,2,3,5,7)]`
- **color_ope**: Default NULL, argument passed for the `plot.type='standard'` method. A vector of color for the operations. Default to `brewer.pal(4,'Paired')`
- **silent**: Stops most messages from being displayed
- **...**: Additional arguments passed to `matplot` or `plot` if `plot.type='standard'`, see ... in `fungraph_glassee` and `fungraph`

Value

Nothing, called for its side effect of plotting

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>

---

Description

Plots of various type for `reportcarlot`

Usage

```r
## S4 method for signature 'report_sample_char,missing'
plot(x, plot.type = "1", silent = FALSE)
```
plot.report_sea_age,missing-method

Arguments

- **x** An object of class `report_sample_char`
- **plot.type** One of '1', 'violin plot'. Default to 1, can also be 2 boxplot or 3 points.
- **silent** Stops displaying the messages

Value

Nothing, called for its side effect, plotting

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>

Description

Plots of various type for `report_sea_age`

Usage

```r
# S4 method for signature 'report_sea_age,missing'
plot(x, plot.type = "1", silent = FALSE)
```

Arguments

- **x** An object of class `report_sea_age-class`
- **plot.type** Default "1"
  - plot.type="1" density plot by sea age
  - plot.type="2" Density plot by sea age and dc
- **silent** Default FALSE, if TRUE the program should no display messages.

Value

Nothing, called for its side effect of plotting

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>
Description

Plots of various type for report_silver_eel

Usage

```r
## S4 method for signature 'report_silver_eel,missing'
plot(x, plot.type = c("1", "2", "3", "4"), silent = FALSE)
```

Arguments

- `x`: An object of class `report_silver_eel-class`
- `plot.type`: Default "1"
  - `plot.type="1"` Lattice plot of Durif’s stages according to Body Length and Eye Index (average of vertical and horizontal diameters). If several DC are provided then a comparison of data per dc is provided
  - `plot.type="2"` Lattice plot giving a comparison of Durif’s stage proportion over time, if several DC are provided an annual comparison is proposed, if only one DC is provided then the migration is split into month.
  - `plot.type="3"` Series of graphs showing mean Fulton’s coefficient, Pankhurst eye index, along with a size weight analysis and regression using robust regression (rlm more robust to the presence of outliers).
  - `plot.type="4"` Lattice cloud plot of Pankurst~ Body Length ~ weight)
- `silent`: Stops displaying the messages

Value

A lattice `xy.plot` if `plot.type =1`, a lattice barchart if `plot.type=2`, nothing but plots a series of graphs in a single plot if `plot.type=3`, a lattice cloud object if `plot.type=4`

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>
plot,report_species,missing-method

Plot method for report_species

Description
Plot method for report_species

Usage
## S4 method for signature 'report_species,missing'
plot(x, plot.type = "pie", color = NULL, silent = FALSE)

Arguments
x
An object of class report_species-class

plot.type
Default pie #'

• plot.type="pie": A pie'
• plot.type="barchart": A barchart

color
Default NULL, a vector of colors of length corresponding to the number of
taxa-stage different values, use unique(bilesp@calcdata$taxa_stage) to get that
number. The color applies to both pie and barchart plots

silent
Stops displaying the messages

Value
Nothing, called for producing plots

Author(s)
Cedric Briand <cedric.briand@eptb-vilaine.fr>

print,report_dc-method

Method to print the command line of the object.

Description
Method to print the command line of the object.

Usage
## S4 method for signature 'report_dc'
print(x, ...)

Arguments

- `x` An object of class `report_df`
- `...` Additional parameters passed to `print`

Value

Nothing, called for its side effect

Author(s)

`cedric.briand`

---

**Method to print the command line of the object**

Description

Method to print the command line of the object

Usage

```r
## S4 method for signature 'report_df'
print(x, ...)
```

Arguments

- `x` An object of class `report_df`
- `...` Additional parameters passed to `print`

Value

Nothing, called for its side effect of printing data

Author(s)

`cedric.briand`
print.report_mig_mult-method

Method to print the command line of the object

Description

Method to print the command line of the object

Usage

```r
## S4 method for signature 'report_mig_mult'
print(x, ...)
```

Arguments

- `x` An object of class report_mig_mult
- `...` Additional parameters passed to print

Author(s)

cedric.briand

print.report_sample_char-method

Method to print the command line of the object

Description

Method to print the command line of the object

Usage

```r
## S4 method for signature 'report_sample_char'
print(x, ...)
```

Arguments

- `x` An object of class report_sample_char
- `...` Additional parameters passed to print

Author(s)

cedric.briand
Method to print the command line of the object

## S4 method for signature 'report_sea_age'

```r
print(x, ...)
```

### Arguments

- `x`: An object of class `report_sea_age`
- `...`: Additional parameters passed to `print`

### Author(s)

cedric.briand

Method to print the command line of the object

## S4 method for signature 'report_silver_eel'

```r
print(x, ...)
```

### Arguments

- `x`: An object of class `report_silver_eel`
- `...`: Additional parameters passed to `print`

### Value

NULL, prints data in the console

### Author(s)

cedric.briand
ref_choice-class

Description
ref_choice referential class allows to choose within several values with radiobuttons interface

Slots
listechoice  A character vector giving possible choices
label  A character, title of the box giving the possible choices
selected  An Integer the initial selected value (as an index), first=1 used in gradio

Objects from the Class
Objects can be created by calls of the form new('ref_choice', listechoice=character(), label=character(), selected=integer()).

Author(s)
cedric.briand@eptb-vilaine.fr

See Also
Other referential objects: charge, ref_choice-method, ref_coe-class, ref_dc-class, ref_df-class, ref_horodate-class, ref_list-class, ref_par-class, ref_parqual-class, ref_parquan-class, ref_period-class, ref_stage-class, ref_taxa-class, ref_year-class

ref_coe-class

Description
Enables to load conversion coefficients quantity-number. This class only exists to load the data with its method charge. It is not used directly as component of the graphical interface, as the year is already loaded in the different report objects

Slots
data  A data.frame
datedebut  A 'POSIXlt'
datefin  A 'POSIXlt'
Objects from the Class

Objects can be created by calls of the form `new('ref_coe')`.

Note

Class loading coefficient of conversion between quantity (weights or volumes of glass eel) and numbers between a starting and finishing date

Author(s)

cedric.briand@eptb-vilaine.fr

See Also

Other referential objects: `charge`, `ref_choice-method`, `ref_choice-class`, `ref_dc-class`, `ref_df-class`, `ref_horodate-class`, `ref_list-class`, `ref_par-class`, `ref_parqual-class`, `ref_parquan-class`, `ref_period-class`, `ref_stage-class`, `ref_taxa-class`, `ref_year-class`

---

### ref_dc-class

**Class 'ref_dc'**

**Description**

Description of a control device.

**Slots**

- `dc_selected` Object of class 'integer', The selected device
- `ouvrage` Object of class 'integer', the attached dam
- `station` Object of class 'character', the attached migration monitoring station, this is necessary to join the table of escapements calculated at the station level.
- `data` Object of class 'data.frame' data pertaining to the control device

Objects from the Class

Objects can be created by calls of the form `new('ref_dc', dc_selected=integer(), ouvrage=integer(), data=data.frame())`.

Author(s)

cedric.briand@eptb-vilaine.fr

See Also

Other referential objects: `charge`, `ref_choice-method`, `ref_choice-class`, `ref_coe-class`, `ref_df-class`, `ref_horodate-class`, `ref_list-class`, `ref_par-class`, `ref_parqual-class`, `ref_parquan-class`, `ref_period-class`, `ref_stage-class`, `ref_taxa-class`, `ref_year-class`
**ref_df-class**

Class `ref_df`  

**Description**  
Representation of a fishway, contains description data of all fishways from the database along with the selected fishways (df) (integer) Objects from the Class: Objects can be created by calls of the form `new('ref_df', df_selected=integer(), ouvrage=integer(), data=data.frame())`.

**Arguments**
- `df_selected`  
  Object of class 'integer' The identifier of the fishway  
- `ouvrage`  
  Object of class 'integer' The attached dam  
- `data`  
  Object of class 'data.frame' Data concerning the fishway

**Author(s)**  
cedric.briand@eptb-vilaine.fr

**See Also**

Other referential objects: `charge`, `ref_choice-method`, `ref_choice-class`, `ref_coe-class`, `ref_dc-class`, `ref_horodate-class`, `ref_list-class`, `ref_par-class`, `ref_parqual-class`, `ref_parquan-class`, `ref_period-class`, `ref_stage-class`, `ref_taxa-class`, `ref_year-class`

---

**ref_env-class**

Class `ref_env`  

**Description**  
Enables to load measure stations and to select one of them

**Slots**
- `dataframe`  
  Data concerning the measure station  
- `env_selected`  
  The selected measure station

**Objects from the Class**

Objects can be created by calls of the form `new('ref_env', ...)`.  

**Author(s)**  
cedric.briand@eptb-vilaine.fr
ref_horodate-class  

Class ref_horodate

Description
choice of date with method to show current and previous year

Slots
horodate a "POSIXt"

Objects from the Class
Objects can be created by calls of the form new("ref_horodate", ...{}).

Author(s)
Cedric Briand <cedric.briand@eptb-vilaine.fr>

See Also
Other referential objects: charge, ref_choice-method, ref_choice-class, ref_coe-class, ref_dc-class, ref_df-class, ref_list-class, ref_par-class, ref_parqual-class, ref_parquan-class, ref_period-class, ref_stage-class, ref_taxa-class, ref_year-class

ref_par-class  

Class 'ref_par'

Description
Class enabling to load the list of parameters and select one of them

Slots
data A data.frame
par_selected A character vector corresponding to par_code
data='data.frame' the list of parameters

Objects from the Class
Objects can be created by calls of the form

Author(s)
cedric.briand@eptb-vilaine.fr
ref_parqual-class

See Also

Other referential objects: charge, ref_choice-method, ref_choice-class, ref_coe-class, ref_dc-class, ref_df-class, ref_horodate-class, ref_list-class, ref_par-class, ref_parqual-class, ref_parquan-class, ref_period-class, ref_stage-class, ref_taxa-class, ref_year-class

----

ref_parqual-class  Class 'ref_parqual'

Description

Class enabling to load the list of qualitative parameters and to select one of them. It inherits from 'ref_par', uses its 'choice' method

Slots

valqual='data.frame' the list of qualitative parameters

Author(s)

cedric.briand@epth-vilaine.fr

See Also

Other referential objects: charge, ref_choice-method, ref_choice-class, ref_coe-class, ref_dc-class, ref_df-class, ref_horodate-class, ref_list-class, ref_par-class, ref_parqual-class, ref_parquan-class, ref_period-class, ref_stage-class, ref_taxa-class, ref_year-class

----

ref_parquan-class  Class 'ref_parquan'

Description

Class enabling to load the list of quantitative parameters and to select one of them. It inherits from 'ref_par', uses its 'choice' method

Author(s)

cedric.briand@epth-vilaine.fr

See Also

Other referential objects: charge, ref_choice-method, ref_choice-class, ref_coe-class, ref_dc-class, ref_df-class, ref_horodate-class, ref_list-class, ref_par-class, ref_parqual-class, ref_parquan-class, ref_period-class, ref_stage-class, ref_taxa-class, ref_year-class
ref_period-class

Class 'ref_period' referential class

Description

ref_period referential class to choose a period

Slots

data = 'data.frame' providing correspondance between period and their English names

Objects from the Class

Objects can be created by calls of the form new('ref_period', ...).

Note

pgval are used by seq.POSIXt

See Also

Other referential objects: charge, ref_choice-method, ref_choice-class, ref_coe-class, ref_dc-class, ref_df-class, ref_horodate-class, ref_list-class, ref_par-class, ref_parqual-class, ref_parquan-class, ref_stage-class, ref_taxa-class, ref_year-class

ref_stage-class

Class 'ref_stage'

Description

Representation of a fish phase

Slots

data A data frame containing data loaded from the database by either charge or charge_with_filter methods
stage_selected Contains the code 'tax_code' of the stage selected by choice_c() method

Objects from the Class

Objects can be created by calls of the form new('ref_stage', data='data.frame').

list('data') Object of class 'data.frame' ~ The phases available in the database
: Object of class 'data.frame' ~ The phases available in the database
ref_taxa-class

Author(s)

cedric.briand@eptb-vilaine.fr

See Also

Other referential objects: charge, ref_choice-method, ref_choice-class, ref_coe-class, ref_dc-class, ref_df-class, ref_horodate-class, ref_list-class, ref_par-class, ref_parqual-class, ref_parquan-class, ref_period-class, ref_taxa-class, ref_year-class

ref_taxa-class  Class 'ref_taxa'

Description

Loading and selection of fish species. This class is a referential class, and it is integrated into refreport objects.

Slots

data  A 'data.frame' of species available in the database
taxa_selected  Contains the code 'tax_code' of the taxa selected by choice_c() method

Objects from the Class

Objects can be created by calls of the form new('ref_taxa', ...).

Author(s)

cedric.briand@eptb-vilaine.fr

See Also

Other referential objects: charge, ref_choice-method, ref_choice-class, ref_coe-class, ref_dc-class, ref_df-class, ref_horodate-class, ref_list-class, ref_par-class, ref_parqual-class, ref_parquan-class, ref_period-class, ref_stage-class, ref_year-class
Description
allows to put a value within a glabel

Slots
  title='character'  the title of the box giving the possible choices
  labels  the logical parameters choice
  checked  a vector

Author(s)
cedric.briand@eptb-vilaine.fr

Objects from the Class
Objects can be created by calls of the form new("ref_timestep",dateDebut="POSIXt",step_duration=numeric(),nb_step=numeric(),nocurrent_step=integer()).

  list("dateDebut")  Object of class "POSIXt" Starting date
  : Object of class "POSIXt" Starting date
  list("step_duration")  Object of class "numeric" Step length
  : Object of class "numeric" Step length
  list("nb_step")  Object of class "numeric" Number of steps
  : Object of class "numeric" Number of steps
  list("nocurrent_step")  Object of class "integer" Number of the current step
  : Object of class "integer" Number of the current step

Author(s)
cedric.briand@eptb-vilaine.fr

See Also
ref_timestep_daily
ref_timestepChar-class

Class "ref_timestepChar"

Description

Character to represent a ref_timestep

Objects from the Class

Objects can be created by calls of the form `new("ref_timestepChar", ...{})`

Author(s)

cedric.briand@eptb-vilaine.fr

See Also

ref_timestep

Examples

`showClass("ref_timestepChar")`

ref_timestep_daily-class

Class 'ref_timestep_daily'

Description

Representation of a ref_timestep object with a step length equal to one day. It receives an inheritance from ref_timestep

Details

validity_ref_timestep_daily
Objects from the Class

Objects can be created by calls of the form `new('ref_timestep_daily', dateDebut='POSIXt', step=numeric())`.

- `list('dateDebut')` Object of class 'POSIXt' Starting date
  - Object of class 'POSIXt' Starting date
- `list('step_duration')` Object of class 'numeric' Step length
  - Object of class 'numeric' Step length
- `list('nb_step')` Object of class 'numeric' Number of steps
  - Object of class 'numeric' Number of steps
- `list('nocurrent_step')` Object of class 'integer' Number of the current step
  - Object of class 'integer' Number of the current step

Author(s)

cedric.briand@eptb-vilaine.fr

See Also

- `ref_timestep`

---

**ref_year-class**  
*Year reference class*

Description

Class used to select one or several years

Slots

- `data` A data.frame with the list of possible years selected as numerics
- `year_selected` A numeric vector

Objects from the Class

Objects can be created by calls of the form `new('ref_year', data=data.frame(), year_selected=numeric())`.

Author(s)

cedric.briand@eptb-vilaine.fr

See Also

Other referential objects: charge, ref_choice-method, ref_choice-class, ref_coe-class, ref_dc-class, ref_df-class, ref_horodate-class, ref_list-class, ref_par-class, ref_parqual-class, ref_parquan-class, ref_period-class, ref_stage-class, ref_taxa-class
Description

This class displays annual migration counts, for several counting device, taxa or stages.

Slots

dc Object of class ref_dc-class, the counting device, multiple values allowed
data Object of class "data.frame" data for report lot
taxa An object of class ref_taxa-class, multiple values allowed
stage An object of class ref_stage-class, multiple values allowed
start_year Object of class ref_year-class. ref_year allows to choose year of beginning
end_year Object of class ref_year-class ref_year allows to choose last year of the report

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>

See Also

Other report Objects: report_dc-class, report_df-class, report_env-class, report_ge_weight-class,
report_mig-class, report_mig_char-class, report_mig_env-class, report_mig_interannual-class,
report_mig_mult-class, report_sample_char-class, report_sea_age-class, report_silver_eel-class,
report_species-class

Examples

# launching stacomi without database for demo
stacomi(database_expected=FALSE)
# the following piece of script will load the Arzal dataset and connected to iav postgres schema
# it requires a working database
# prompt for user and password but you can set appropriate options for host, port and dbname
## Not run:
stacomi(database_expected=TRUE, sch='iav')
if (interactive()){
  if (!exists("user")){
    user <- readline(prompt="Enter user: ")
    password <- readline(prompt="Enter password: ")
  }
}
options(
stacomiR.dbname = "bd_contmig_nat",
stacomiR.host ="localhost",
stacomiR.port = "5432",
stacomiR.user = user,
r_ann<-new("report_annual")
  r_ann<-choice_c(r_ann,
    dc=c(5,6,12),
    taxa=c("Anguilla anguilla"),
    stage=c("AGJ","AGG"),
    start_year="1996",
    end_year="2015",
    silent=FALSE)
  r_ann<-connect(r_ann)

## End(Not run)
# the following dataset has been generated by the previous code
data(r_ann)
xtr_ann<-stacomiR::xtable(r_ann,
    dc_name=c("Passe bassins","Piege anguille RG","Piege anguille RD"),
    tax_name="Anguille",
    std_name=c("Arg."","Jaun."))

# below not run but one can create a file as following
## Not run:
  path=file.path(path.expand(get("datawd",envir=envir_stacomi)),
    paste(paste(r_ann@dc@dc_selected,collapse="+"),"_",
    paste(r_ann@taxa@taxa_selected,collapse="+"),"_",
    paste(r_ann@stage@stage_selected,collapse="+"),"_",
    r_ann@start_year@year_selected,":",
    r_ann@end_year@year_selected,".html",sep=""),fsep="/")

# here you can add an argument file=path
print(xtr_ann,type="html")

# the following uses the "addtorow" argument which creates nice column headings,
# format.args creates a thousand separator
# again this will need to be saved in a file using the file argument
print(xtr_ann,
  add.to.row=get("addtorow",envir_stacomi),
  include.rownames = TRUE,
  include.colnames = FALSE,
  format.args = list(big.mark = " ", decimal.mark = ",")
)

# barplot transforms the data, further arguments can be passed as to barplot
barplot(r_ann)
barplot(r_ann,
    args.legend=list(x="topleft",bty = "n"),
    col=c("CA803E","1A9266","E10168","005327","FF9194"))

# An example with custom arguments for legend.text (overriding plot defaults)
data(r_ann_adour)
  if (requireNamespace("RColorBrewer", quietly = TRUE)){
    lesdc<-r_ann_adour@dc@data$dc_code[r_ann_adour@dc@data$dc%in%r_ann_adour@dc@dc_selected]
    barplot(r_ann_adour,
      legend.text=lesdc,
      args.legend=list(x="topleft",bty = "n"),
      col=RColorBrewer:::brewer.pal(9,"Spectral"),
      main="Custom Legend Text")
  }
Beside TRUE)
}
plot(r_ann_adour)
## End(Not run)

---

**report_dc-class**

Class "report_dc" report du fonctionnement du dispositif de comptage

**Description**

The counting device is not always working. It may be stopped either following a monitoring protocol, or due to malfunction of the device, this class allows to draw graphics allowing an overview of the device operation

**Slots**

data A data frame
dc An object of class ref_dc-class
horodatedebut An object of class ref_horodate-class
horodatefin An object of class ref_horodate-class

**Objects from the Class**

Objects can be created by calls of the form new("report_dc", ...).

**Author(s)**

Cedric Briand <cedric.briand@eptb-vilaine.fr>

**See Also**

Other report Objects: report_annual-class, report_df-class, report_env-class, report_ge_weight-class, report_mig-class, report_mig_char-class, report_mig_env-class, report_mig_interannual-class, report_mig_mult-class, report_sample_char-class, report_sea_age-class, report_silver_eel-class, report_species-class

**Examples**

```r
# An example that will work only if the database is present
# and the program installed and comprises the schema iav
# prompt for user and password but you can set appropriate options for host, port and dbname
## Not run:
if (interactive()){  
  if (!exists("user")){  
    user <- readline(prompt="Enter user: ")
```
password <- readline(prompt="Enter password: ")

options(
  stacomiR.dbname = "bd_contmig_nat",
  stacomiR.host = "localhost",
  stacomiR.port = "5432",
  stacomiR.user = user,
  stacomiR.password = password
)

stacomi(TRUE,sch="iav")

r_dc=new("report_dc")
r_dc<-choice_c(r_dc,
  5,
  horodatedebut="2000-01-01",
  horodatefin="2015-12-31",
  silent=TRUE)
r_dc<-connect(r_dc)

## End(Not run)

# this dataset has been loaded by the previous lines
###########################################################
# Without connexion to the database (use dataset r_dc)
# this option allows to launch the program without the interface to display
# some of the program features.
staconi(database_expected=FALSE)
data("r_dc")
plot(r_dc,plot.type="1")
plot(r_dc,plot.type="2")
plot(r_dc,plot.type="3",main="trial title")
plot(r_dc,plot.type="4",main="trial title")
# the following will write in the datawd folder
summary(r_dc)

### report_df-class

**Report on fishway operation**

**Description**

Fishways (DF) are of various nature, from very simple eel ladders fed by water discharged from the river, to more complex fishways with levels adjusted by the opening of various gates and regulators.
The objective of this class is to provide an assessment of the working status of a fishway throughout the year. A number of fishes ascending a fishway has meaning only if we know that the fishway is operational, and that the counting operated on the fishway has remained operational. In the database the operation of the fishway (DF) and counting device (DC) is aggregated in one table (t_periodefonctdispositif_per). The column per_etat_fonctionnement indicates whether the fishway is operational (with a boolean) and the column per_tar_code indicates the status of either the fishway or DC. In the database four types of operation are set, "1"=normal operation, "2"=Device stopped in normal operation (ie lift ascending, high tide...), "3"="Stopped for maintenance or other problem", "4"="Works but not fully operational,i.e.flow problem, flood, clogged with debris...", "5"="Not known")

Slots

data  A data frame
df An object of class ref_df-class
horodatedebut  An object of class ref_horodate-class
horodatefin  An object of class ref_horodate-class

Objects from the Class

Objects can be created by calls of the form new("report_df").

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>

See Also

Other report Objects: report_annual-class, report_dc-class, report_env-class, report_ge_weight-class, report_mig-class, report_mig_char-class, report_mig_env-class, report_mig_interannual-class, report_mig_mult-class, report_sample_char-class, report_sea_age-class, report_silver_eel-class, report_species-class

Examples

stacomi(
  database_expected=FALSE)
# An example that will work with the database installed only and schema iav in the database
# prompt for user and password but you can set appropriate options for host, port and dbname

## Not run:
stacomi(
  database_expected=TRUE, sch='iav')
if (interactive()){
  if (!exists("user")){
    user <- readline(prompt="Enter user: ")
    password <- readline(prompt="Enter password: ")
  }
}
options(}
report_env-class

class report_env simple output of one or several environmental conditions...

Description

Annual overview of environmental conditions. This class enables to draw some plot, but will mostly be used to build joined graphs crossing the information from report_mig_mult-class and report_mig_env-class.

Slots

- horodatedebut ref_horodate-class
- horodatefin ref_horodate-class
- stationMesure ref_env-class
- data data.frame

---

r_df = new("report_df")
r_df <- choice_c(r_df, 1,
    horodatedebut = "2015-01-01",
    horodatefin = "2015-12-31",
    silent = TRUE)
Sys.setenv(TZ = "GMT")
# the times at Arzal are recorded continuously
# they are converted to date when a time appears while the hour is changing
# hence the following
r_df <- connect(r_df)

## End(Not run)
data("r_df")
plot(r_df, plot.type = "4")
# the following examples work but take a while to compute
## Not run:
    plot(r_df, plot.type = "1")
    plot(r_df, plot.type = "2", main = "A nice title")
    plot(r_df, plot.type = "3", main = "A nice title")

## End(Not run)
Author(s)

cedric.briand@epib-vilaine.fr

See Also

Other report Objects: report_annual-class, report_dc-class, report_df-class, report_ge_weight-class, report_mig-class, report_mig_char-class, report_mig_env-class, report_mig_interannual-class, report_mig_mult-class, report_sample_char-class, report_sea_age-class, report_silver_eel-class, report_species-class

Examples

```r
stacomi(
database_expected=FALSE)
## Not run:
if (interactive()){
  if (!exists("user")){
    user <- readline(prompt="Enter user: ")
    password <- readline(prompt="Enter password: ")
  }
}
options(
stacomiR.dbname = "bd_contmig_nat",
stacomiR.host ="localhost",
stacomiR.port = "5432",
stacomiR.user = user,
stacomiR.user = password
)
  r_env<-new("report_env")
  r_env<-choice_c(r_env,
    stationMesure=c("temp_gabion","coef_maree"),
    datedebut="2008-01-01",
    datefin="2008-12-31",
    silent=FALSE)
  r_env<-connect(r_env)

## End(Not run)

data("r_env")
plot(r_env,silent=TRUE)
```

---

**report_ge_weight-class**

*Trend of wet weight in glass eel*
Description

In trapping ladders, glass eel are seldom counted, as they are too tiny to handle and too numerous to count. The usual operation is to weight them, or to use a bucket to measure their volume. These weights or volumes will later need to be converted to numbers. The glass eel weight may follow a seasonal pattern. It’s the case for Anguilla anguilla glass eel in the Bay of Biscay. Weights can be modelled using sine wave curves, or more complex gam models. This class has a model method to try those models, which can also be used to extract coefficients manually to manually test more complex models. Some plots are provided to display the coefficients stored in the database, and the model results. A parameter provided in the graphical interface or in the command line (slot liste) takes values '1', '>1', 'tous' which mean respectively use only individual sample of glass eels, or use 'group weights' which can be more close to the real weight of glass eel during counts as glass eel are not completely drained from their water during handling to preserve their mucus. The list choice 'tous' means that both individual and group weights are selected.

Slots

data A 'data.frame' data for report lot
calcdatas A list containing two processed data frames, data and coe
dc Object of class ref_dc-class, the counting device
start_year Object of class ref_year-class, ref_year allows to choose the year of beginning
end_year Object of class ref_year-class ref_year allows to choose last year of the report
coe Object of class ref_coe-class class loading coefficient of conversion between quantity (weights or volumes of glass eel) and numbers
liste Object of class ref_list-class ref_list referential class choose within a list, here the choice is whether subsamples or not. Subsamples in the stacom database are samples with a non null value for parent sample. Migration counts are never made on subsamples but those can be integrated to calculate mean weights.

Note

In this class some tools are available to import glass eel measurement from experimental fishing in the estuary. For the charge method dates for the request are from august to august (a glass eel season)

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>

See Also

Other report Objects: report_annual-class, report_dc-class, report_df-class, report_env-class, report_mig-class, report_mig_char-class, report_mig_env-class, report_mig_interannual-class, report_mig_mult-class, report_sample_char-class, report_sea_age-class, report_silver_eel-class, report_species-class
Examples

```r
require(stacomiR)
# launching stacomi without selecting the scheme or interface
fastacomi(
  database_expected=FALSE, sch='iav')
# this requires a working database with the schema iav
# prompt for user and password but you can set appropriate options for host, port and dbname
## Not run:
fastacomi(
  database_expected=TRUE, sch='iav')
if (interactive()){
  if (!exists("user")){
    user <- readline(prompt="Enter user: ")
    password <- readline(prompt="Enter password: ")
  }
}
options(
  stacomiR.dbname = "bd_contmig_nat",
  stacomiR.host = "localhost",
  stacomiR.port = "5432",
  stacomiR.user = user,
  stacomiR.user = password
)
# create an instance of the class
r_gew <- new("report_ge_weight")
r_gew@liste <- charge(object=r_gew@liste,listechoice=c("=1",">1","tous"),label="")
# here I'm using weights when number are larger than 1 i.e. wet weight
# always choose a date from one year to the next eg 2010 to 2011
# as the dates are from august to august
r_gew <- choice_c(r_gew,dc=c(6),
  start_year="2009",
  end_year="2015",
  selectedvalue=">1",
  silent=FALSE)
r_gew <- connect(r_gew)
r_gew <- calcule(r_gew)
## End(Not run)
# load the dataset generated by previous lines
data("r_gew")
# the calculation will fill the slot calcdato

# A ggplot showing the trend in weight
plot(r_gew, plot.type=1)
# A plot showing both the data and the trend as recorded in the database
plot(r_gew, plot.type=2)
# Same as plot.type=1 but with size according to size of the sample,
# usefull for wet weights where weight are recorded on a number of glass eel
plot(r_gew, plot.type=3)
## Not run:
# First model with nls, see Guerault and Desaunay (1993)
```
report_mig-class

Migration report for one DC, one species and one stage

Description

This class performs a migration summary. A migration monitoring operation can correspond to a single horodate (in the case of some video monitoring operation) or comprise a period which does not necessarily span a full day. The daily migration is calculated by splitting the operation between days, and the migration is either grouped or split according to the length of the different time spans.

Slots

dc Object of class ref_dc-class: the control device
taxa Object of class ref_taxa-class: the species
stage Object of class ref_stage-class: the stage of the fish	
timestep Object of class ref_timestep_daily-class: the time step constrained to daily value and 365 days
data Object of class data.frame with data filled in from the connect method
calcd"ata A "list" of calculated daily data, one per dc, filled in by the calcule method
coef_conversion A data.frame of daily weight to number conversion coefficients, filled in by the connect method if any weight are found in the data slot.
time.sequence Object of class POSIXct: a time sequence of days generated by the calcule method

Note

In practise, the report_mig class uses methods (calcule, connect...) from the more elaborate report_mig_mult-class

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>

See Also

Other report Objects: report_annual-class, report_dc-class, report_df-class, report_env-class, report_ge_weight-class, report_mig_char-class, report_mig_env-class, report_mig_interannual-class, report_mig_mult-class, report_sample_char-class, report_sea_age-class, report_silver_eel-class, report_species-class
Examples

```r
stacomi(database_expected=FALSE)
# If you have a working database
# the following line of code will create the r_mig dataset from the iav (default)
# schema in the database
## Not run:
stacomi(database_expected=TRUE)
if (interactive()){
  if (!exists("user")){
    user <- readline(prompt="Enter user: ")
    password <- readline(prompt="Enter password: ")
  }
}
options(
stacomiR.dbname = "bd_contmig_nat",
stacomiR.host = "localhost",
stacomiR.port = "5432",
stacomiR.user = user,
stacomiR.user = password
)
stacomi(
  database_expected=TRUE)
  r_mig=new("report_mig")
  r_mig=choice_c(r_mig,
dc=5,
taxa=c("Liza ramada"),
stage=c("IND"),
datedebut="2015-01-01",
datefin="2015-12-31")
  r_mig<-charge(r_mig)
  # launching charge will also load classes associated with the report
  # e.g. report_ope, report_df, report_dc
  r_mig<-connect(r_mig)
  # calculations
  r_mig<-calcule(r_mig,silent=TRUE)
}
## End(Not run)

# loading data
# use the following to get the raw data loaded by the connect method
# not shown there as the database and program might not be installed
# All three classes report... were created by the charge and connect method
# of report_mig_mult
# in the previous example

data("r_mig")
data("r_mig_ope")
assign("report_ope", r_mig_ope, envir=envir_stacomi)
data("r_mig_df")
assign("report_df", r_mig_df, envir=envir_stacomi)
```
data("r_mig_dc")
assign("report_dc",r_mig_dc,envir=envir_stacomi)

# Individual plot for all DC (standard), taxa and stage where data present
# silent argument to stop all messages
plot(r_mig,plot.type="standard",silent=TRUE)
# cumulated migration at the station (all stages and DC grouped)
plot(r_mig,plot.type="step")

# data will be written in the data directory specified in datawd argument to stacomi default "~" file

## Not run:
summary(r_mig,silent=TRUE)
## End(Not run)

# this will write the daily report for later in in the reportnMigrationInterannuelle-class
## Not run:
write_database(r_mig,silent=TRUE,dbname="bd_contmig_nat",host="localhost",port=5432)
## End(Not run)

---

**report_mig_char-class**  
Migration report along with quantitative and qualitative characteristics

### Description

Migration along with qualitative or quantitative characteristics or both (e.g.) weight of eels according to the size class per period of time, weight of fish according to gender, number of fish per age class. This class does not split migration evenly over time period. So, unlike calculations made in class report_mig and report_mig_mult the whole time span of the migration operation is not considered, only the date of beginning of the operation is used to perform calculations.

### Slots

- **calcdata** A 'list' of calculated data, filled in by the calcule method
- **data** A data.frame inherited from report_sample_char-class
- **dc** An object of class ref_dc-class inherited from report_sample_char-class
- **taxa** An object of class ref_taxa-class inherited from report_sample_char-class
- **stage** An object of class ref_stage-class inherited from report_sample_char-class
- **horodatedebut** An object of class ref_horodate-class inherited from report_sample_char-class
- **horodatefin** An object of class ref_horodate-class inherited from report_sample_char-class
- **par** An object of class ref_par-class inherited from report_sample_char-class
- **echantillon** An object of class ref_choice-class, vector of choice
- **parquan** An object of class ref_parquan-class, quantitative parameter
- **parqual** An object of class ref_parqual-class, qualitative parameter
Objects from the Class

Objects can be created by calls of the form `new('report_mig_char', ...)`. They are loaded by the interface using `interface_report_mig_char` function.

Note

The main difference between this class and `report_sample_char-class` is that this class allows to select (or not) the samples, and that it handles quantitative and qualitative parameters separately.

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>

See Also

Other report Objects: `report_annual-class, report_dc-class, report_df-class, report_env-class, report_ge_weight-class, report_mig-class, report_mig_env-class, report_mig_interannual-class, report_mig_mult-class, report_sample_char-class, report_sea_age-class, report_silver_eel-class, report_species-class`

Examples

```r
require(stacomiR)

stacomi(
  database_expected=FALSE, sch='logrami')
# this requires a database with the schema logrami
# prompt for user and password but you can set appropriate options for host, port and dbname
## Not run:
stacomi(database_expected=TRUE, sch='logrami')
if (interactive()){
  if (!exists("user")){
    user <- readline(prompt="Enter user: ")
    password <- readline(prompt="Enter password: ")
  }
}
options(
  stacomiR.dbname = "bd_contmig_nat",
  stacomiR.host = "localhost",
  stacomiR.port = "5432",
  stacomiR.user = user,
  stacomiR.user = password
)

r_mig_char <- new("report_mig_char")
# here parqual is not in the list
# so this is equivalent to parqual=NULL
# default for echantillon is "with"
rc.mig_char <- choice_c(r_mig_char,
  dc=c(107,108,101),
  taxa=c("Salmo salar"),
  stage=c("5","11","BEC","BER","IND"),
)`
parquan=c('C001','1786','1785'),
horodatedebut="2012-01-01",
horodatefin="2012-12-31",
silent=FALSE)
# r_mig_char<-charge(r_mig_char) not necessary there
r_mig_char <- connect(r_mig_char)

## End(Not run)
# load the dataset generated by previous lines
data("r_mig_char")

r_mig_char<-calcule(r_mig_char, silent=TRUE)
plot(r_mig_char,plot.type="quant", silent=TRUE)
# one quantitative parameter found, manual choice of color
plot(r_mig_char,plot.type="quant",color_parm=c("C001"="red"), silent=TRUE)
# age will be plotted as a qualitative variable
# here we split size data according to the limit known between different ages from
# scale reading in the Loire
r_mig_char <- setasqualitative(r_mig_char,par='C001',
breaks=c(0,675,850,2000),
labels=c("age 1","age 2","age 3"))
plot(r_mig_char,plot.type="qual", silent=TRUE)
plot(r_mig_char,plot.type="crossed")
plot(r_mig_char,plot.type="crossed",
    color_parm=c("age 1"="#379ec6","age 2"="#173957","age 3"="#b09953"))
xt<-xtable(r_mig_char)
# use method print.xtable to get the output

---

**report_mig_env-class**

**Class** "report_mig_env"

**Description**

Enables to compute an annual overview of fish migration and environmental conditions in the same chart. Environmental conditions may trigger migration events, variation in flow or temperatures can be plotted along migration to check graphically for a possible relation. To enable this, environmental conditions are loaded from an "environmental monitoring station", which records environmental parameters and is attached to a migratory station in the database. This class enables both continuous output (temperature -flow) as well as discrete parameters (qualitative = moon phase, type of operation of a gate, opening of a gate...) which will be displayed on the graph. Values are scaled so that single plot can display migration numbers and environmental parameters. Environmental parameters when stored at a time scale lower that a day are averaged per day, unless they are qualitative parameters, in which case only the first event of the day is displayed on the annual plot.
report_mig_env-class

Slots

report_mig_mult report_mig_mult-class
report_env report_env-class

Author(s)

cedric.briand@eptb-vilaine.fr marion.legrand@logrami.fr
Cedric Briand <cedric.briand@eptb-vilaine.fr>

See Also

Other report Objects: report_annual-class, report_dc-class, report_df-class, report_env-class, report_ge_weight-class, report_mig-class, report_mig_char-class, report_mig_interannual-class, report_mig_mult-class, report_sample_char-class, report_sea_age-class, report_silver_eel-class, report_species-class

Other report Objects: report_annual-class, report_dc-class, report_df-class, report_env-class, report_ge_weight-class, report_mig-class, report_mig_char-class, report_mig_interannual-class, report_mig_mult-class, report_sample_char-class, report_sea_age-class, report_silver_eel-class, report_species-class

Examples

require(stacomiR)
staconi(
  database_expected=FALSE)
  # the following will load the data provided the user has access to the database
  # with data in the iav example scheme.
  # prompt for user and password but you can set appropriate options for host, port and dbname
  ## Not run:
  staconi(
  database_expected=TRUE)
  if (interactive()){
    if (!exists("user")){
      user <- readline(prompt="Enter user: ")
      password <- readline(prompt="Enter password: ")
    }
  }
  options(
    stacomiR.dbname = "bd_contmig_nat",
    stacomiR.host = "localhost",
    stacomiR.port = "5432",
    stacomiR.user = user,
    stacomiR.user = password
  )
  r_mig_env<-(new("report_mig_env")
  r_mig_env<-(choice_c(r_mig_env,
    dc=c(5,6,12),
    taxa=c("Anguilla anguilla"),
    stage=c("AGJ","AGG","CIV"),
    stationMesure=c("temp_gabion","coef_maree","phases_lune"),
  )

report_mig_interannual-class

Class "report_mig_interannual"

Description

When daily report are written in the `t_reportjournalier_bjo` table by the `report_mig-class` they can be used by this class to display interannual comparisons of migration. When running its connect method, this class will run the `report_mig-class` for each year where data are missing, or where the annual sum in the `t_reportjournalier_bjo` table differs from the counts generated by the `report_annual-class`: rows have been changed in the database. Different charts are produced with different period grouping. See `write_database,report_mig-method` for details about how this method inserts data in the `t_reportjournalier_bjo` table.

Slots

dc  An object of class `ref_dc-class`, the counting device
data  A data.frame data loaded from the daily migration table `t_bilanmigrationjournalier_bjo`
taxa  An object of class `ref_taxa-class`
stage  An object of class `ref_stage-class`
start_year  An object of class `ref_year-class`. ref_year allows to choose year of beginning
end_year  An object of class `ref_year-class` ref_year allows to choose last year of the report
calcd data  A list of calculated data, filled in by the calcule method

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>
See Also

Other report Objects: report_annual-class, report_dc-class, report_df-class, report_env-class, report_ge_weight-class, report_mig-class, report_mig_char-class, report_mig_env-class, report_mig_mult-class, report_sample_char-class, report_sea_age-class, report_silver_eel-class, report_species-class

Examples

```r
require(stacomiR)
# launching stacomi without selecting the scheme or interface
stacomi(
database_expected=FALSE, sch='pmp')
# If you have connection to the database with the pmp scheme
# prompt for user and password but you can set appropriate options for host, port and dbname
## Not run:
stacomi(database_expected=TRUE, sch="pmp")
if (interactive()){
  if (!exists("user")){
    user <- readline(prompt="Enter user: ")
    password <- readline(prompt="Enter password: ")
  }
}
options(
stacomiR.dbname = "bd_contmig_nat",
stacomiR.host ="localhost",
stacomiR.port = "5432",
stacomiR.user = user,
stacomiR.user = password)

# (longest historical dataset available
# in France for eel ... ) this suppose you have access to the pmp schema...
# a glimpse of the dataset is still available in the r_mig_interannual dataset
# loaded in the package...
r_mig_interannual <- new("report_mig_interannual")
r_mig_interannual <- choice_c(r_mig_interannual, dc=c(16),
taxa=c("Anguilla anguilla"),
stage=c("PANG"),
start_year="1990",
end_year="2015",
silent=TRUE)
r_mig_interannual <- charge(r_mig_interannual)
r_mig_interannual <- connect(r_mig_interannual, check=TRUE)
r_mig_interannual <- calcule(r_mig_interannual, silent=TRUE)
## End(Not run)

#############otherwise use this ######################
# load the dataset generated by previous lines
data("r_mig_interannual")
```

"""
# the first plot is of little interest, it allows to see what data
# are available... simple lines
# For irregular operations like those reported at the enfrenaux eel ladder....
plot(r_mig_interannual, plot.type="line", silent=TRUE)

# a plot to show the seasonality, this graph may be misleading if the
# migration is not monitored all year round. Note the y unit is not very informative
# you need to have the viridis package loaded to run this example
plot(r_mig_interannual, plot.type="density", silent=TRUE)
## Not run:
if (requireNamespace("ggplot2", quietly = TRUE)&
    requireNamespace("viridis", quietly = TRUE)){
g<-get("g", envir=envir_stacomi)
g+
ggplot2::scale_fill_manual(values=viridis::viridis(22)) +
ggplot2::ggtitle("Saisonnalite de la migration aux Enfrenaux")
}

#############################################
# the standard plot is showing daily values
#############################################
plot(r_mig_interannual, plot.type="standard", silent=TRUE)
# Manual edition of the graph produced
if (requireNamespace("ggplot2", quietly = TRUE)){
g1<-get("g1", envir=envir_stacomi)
g1<-g1+ggplot2::ggtitle("Les Enfrenaux") +
ggplot2::scale_fill_manual(name="Source", values=c("purple", 
#0A0C01"), values = c("purple", 
labels = c("historical set", "2015 values")) +
ggplot2::scale_colour_manual(name="Source", values="#B8EA00", 
labels = c("historical mean")) +
ggplot2::ylab("Nombre d’anguilles")
print(g1)
}

#########################################################
# Another graph to show a "manual" processing of the data
# and their extraction from the data slot
#########################################################
if (requireNamespace("ggplot2", quietly = TRUE) &
    requireNamespace("viridis", quietly = TRUE)){
    dat<-fun_date_extraction(r_mig_interannual@data, # data to import
"bjo_jour", # name of the column where dates are found
annee=FALSE,
mois=TRUE,
semaine=TRUE,
jour_mois=FALSE)
    # sum per month
    res<-dplyr::select(dat, bjo_valeur, bjo_annee, semaine)
    res<-dplyr::group_by(res, bjo_annee, semaine)
    res<-dplyr::summarize(res, effectif=sum(bjo_valeur))
    ggplot2::ggplot(res, ggplot2::aes(x = semaine, y = bjo_annee, fill=effectif)) +
ggplot2::geom_tile(colour="black") +
ggplot2::coord_fixed() +
viridis::scale_fill_viridis(begin=0, option="D") +
ggplot2::theme_bw() +
ggplot2::theme(panel.background= ggplot2::element_rect(fill = "#9360A9"),
"bjo_annee", "semaine")
print(g1)
panel.grid.major=ggplot2::element_line(colour="#C1DB39"),
panel.grid.minor=ggplot2::element_line(colour="#7DD632")+
ggplot2::ylab("year")+ggplot2::xlab("week")+
ggplot2::ggtitle("Historical trend at Les Enfrenaux Eel trap")

#################################################################
# barchart with different splitting periods
# the migration is displayed against seasonal data
# extracted from all other years loaded in the report
# available arguments for timesplit are "quinzaine" and "mois" and "semaine"
# with the silent=TRUE argument, it's always the latest year that is selected,
# otherwise the user is prompted with a choice, to select the year he wants
# to compare will all others...
plot(r_mig_interannual,plot.type="barchart",timesplit="quinzaine",silent=TRUE)
# Comparison with historical values. Each year and 2 weeks values
# is a point on the graph...
plot(r_mig_interannual,plot.type="pointrange",timesplit="mois",silent=TRUE)

# Step plot
# different years shown in the graph
# the current year (or the selected year if silent=FALSE)
# is displayed with a dotted line
plot(r_mig_interannual,plot.type="step",silent=TRUE)
if (requireNamespace("ggplot2", quietly = TRUE)&
  requireNamespace("viridis", quietly = TRUE)){
g<-get("g",envir=envir_stacomi) + ggplot2::theme_minimal()
g+viridis::scale_color_viridis(discrete=TRUE)+
ggplot2::ggtitle("Cumulated migration step plot
at Les Enfrenaux eel trap")
}

# Plots for seasonality of the salmon migration
# using a Loire river dataset (Vichy fishway)
# the following show how data are processed to get
# statistics for seasonal migration, daily values
r_mig_interannual_vichy<-calcule(r_mig_interannual_vichy,timesplit="jour",silent=TRUE)
#r_mig_interannual_vichy@calcdata #check this to see the results
# statistics for seasonal migration, weekly values
r_mig_interannual_vichy<-calcule(r_mig_interannual_vichy,timesplit="semaine",silent=TRUE)
#r_mig_interannual_vichy@calcdata

# the plot method also runs the calcule method
plot(r_mig_interannual_vichy,plot.type="seasonal",timesplit="semaine",silent=TRUE)
plot(r_mig_interannual_vichy,plot.type="seasonal",timesplit="mois",silent=TRUE)
plot(r_mig_interannual_vichy,plot.type="seasonal",timesplit="jour",silent=TRUE)
# Plots for seasonality using another Loire river dataset
# with the migration of Lampreys (Petromyzon marinus)
# recorded at the Descarte DF (Vienne)

# run this only if you are connected to the logrami dataset
stacomi(database_expected = TRUE, sch = 'logrami')

bmi_des<-new("report_mig_interannual")
bmi_des<-choice_c(bmi_des,
dc=c(23),
taxa=c("Petromyzon marinus"),
stage=c("5"),
start_year="2007",
end_year="2014",
silent=TRUE)
bmi_des<-connect(bmi_des)
bmi_des<-calcule(bmi_des,timesplit="semaine")
plot(bmi_des,plot.type="seasonal",timesplit="semaine")
plot(bmi_des,plot.type="seasonal",timesplit="jour")
plot(bmi_des,plot.type="seasonal",timesplit="mois")

## End(Not run)

---

### Description

Migration counts for several Fish counting devices (DC), several taxa and several stages. This migration count can be built either by the graphical interface or from the command line (see examples).

### Slots

- `dc` An object of class `ref_dc-class`
- `taxa` An object of class `ref_taxa-class`
- `stage` An object of class `ref_stage-class`
- `timestep` An object of class `ref_timestep_daily-class`
- `data` A data.frame containing raw data filled by the connect method
- `calcdata` A 'list' of calculated daily data, one per dc, filled in by the calcule method
- `coef_conversion` A data frame of daily weight to number conversion coefficients, filled in by the connect method if any weight are found in the data slot.
- `time.sequence` A POSIXt time sequence
**Note**

A Migration report comes from a migration monitoring: the fishes are monitored in a section of river, this section is called a control station (station). Most often, there is a dam, one or several fishways (DF) which comprise one or several counting devices (DC). On each counting device, the migration is recorded. It can be either an instant recording (video control) or the use of traps. Operations are monitoring operations during a period. For each operation, several species of fishes can be recorded (samples). In the case of migratory fishes the stage of development is important as it may indicate generic migrations, to and fro, between the river and the sea.

Hence a Multiple Migration report is built from several one or several counting devices (DC), one or several Taxa (Taxon), one or several stages (stage). The migration can be also recorded not as numbers, but in the case of glass eels, as weight, which will be later transformed to number, from daily conversion coefficients. The methods in this class test whether the counts are numbers or another type of quantity. This class makes different calculations than report_mig, it does not handle escapement coefficients, it uses quantities other than numbers if necessary (only used for glass eel in the connect method).

**Author(s)**

Cedric Briand <cedric.briand@eptb-vilaine.fr>

**See Also**

Other report Objects: report_annual-class, report_dc-class, report_df-class, report_env-class, report_ge_weight-class, report_mig-class, report_mig_char-class, report_mig_env-class, report_mig_interannual-class, report_sample_char-class, report_sea_age-class, report_silver_eel-class, report_species-class

**Examples**

```r
library(stacomiR)

stacomi(database_expected=FALSE)
## launches the application in the command line
## here an example of loading
## the following lines will only run if you have the program installed
## and the iav scheme available in the database
## this example generates the r_mig_mult dataset
# prompt for user and password but you can set appropriate options for host, port and dbname
## Not run:
## stacomi(
##   database_expected=TRUE)
if (interactive()){
  if (!exists("user")){
    user <- readline(prompt="Enter user: ")
    password <- readline(prompt="Enter password: ")
  }
}
options(stacomiR.dbname = "bd_contmig_nat",
        stacomiR.host ="localhost",
```
staconiR.port = "5432",
staconiR.user = user,
staconiR.user = password
)

r_mig_mult <- new("report_mig_mult")
r_mig_mult <- choice_c(r_mig_mult,
dc=c(5,6,12),
taxa=c("Anguilla anguilla"),
stage=c("AGG","AGJ","CIV"),datedebut="2011-01-01",datefin="2011-12-31")
r_mig_mult <- charge(r_mig_mult)
# launching charge will also load classes associated with the report
# e.g. report_ope, report_df, report_dc
r_mig_mult <- connect(r_mig_mult)
# calculations
r_mig_mult <- calcule(r_mig_mult,silent=TRUE)

## End(Not run)

# Use this as example if you don’t have a connexion to the database
data("r_mig_mult")
# The following re-create the object at the time of loading
# All three classes were created by the charge and connect
# method of report_mig_mult in the previous example
data("r_mig_mult_ope")
assign("report_ope",r_mig_mult_ope,envir=envir_stacomi)
data("r_mig_mult_df")
assign("report_df",r_mig_mult_df,envir=envir_stacomi)
data("r_mig_mult_dc")
assign("report_dc",r_mig_mult_dc,envir=envir_stacomi)
# use the following to get the raw data loaded by the connect method
# not shown there as the database and program might not be installed

#Individual plot for all DC, taxa and stage where data present

## Not run:
plot(r_mig_mult,plot.type="standard",silent=TRUE)
# colors in the following order (glass eel)
# working, stopped, 1...5 types of operation,numbers, weight, 2 unused colors
# for yellow eel and other taxa
# stopped, 1...5 types of operation, ponctuel, expert, calcule, mesure, working,
plot(r_mig_mult,plot.type="standard",
color=c("#DEF76B","#9590B5","#9A0D6A","#781A74","#BFD6E","#FFC26E",
"#A66F24","#801274","#6C3E00","#DC7ED8","#8A4A123"),
silent=TRUE)

#For the following plot, beware, all stages and DC are grouped. This can make sense
# for instance if you want to display the cumulated migration for one species
# in several counting devices located on the same dam...
plot(r_mig_mult,plot.type="step",silent=TRUE)
# Combined plot for ggplot2
plot(r_mig_mult,plot.type="multiple",silent=TRUE)
# Data will be written in the data directory specified in
# the datwd argument tostawmi, default "~"
summary(r_mig_mult,silent=FALSE)

## End(Not run)

---

**report_ope-class**

*Report on operations*

**Description**

Operations are monitoring operations generated between two dates. In the case of video monitoring or similar, they can be instantaneous.

**Objects from the Class**

Objects can be created by calls of the form `new("report_ope")`.

**Author(s)**

Cedric Briand <cedric.briand@eptb-vilaine.fr>

---

**report_sample_char-class**

*Class 'report_sample_char'*

**Description**

The `report_sample_char` class is used to load and display sample characteristics, which can be either continuous or discrete variable, for instance, it can be used to analyze size or sex structure during a given period.

**Slots**

data A data frame
dc An object of class `ref_dc-class`: the control devices
taxa An object of class `ref_taxa-class`: the species
stage An object of class `ref_stage-class`: the stages of the fish
par An object of class `ref_par-class`: the parameters used
horodatedebut An object of class `ref_horodate-class`
horodatefin An object of class `ref_horodate-class`
Objects from the Class

Objects can be created by calls of the form `new('report_sample_char', ...)`

Note

This class is displayed by interface_report_sample_char, in the database, the class calls the content of the view vue_lot_ope_car

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>

See Also

Other report Objects: `report_annual-class`, `report_dc-class`, `report_df-class`, `report_env-class`, `report_ge_weight-class`, `report_mig-class`, `report_mig_char-class`, `report_mig_env-class`, `report_mig_interannual-class`, `report_mig_mult-class`, `report_sea_age-class`, `report_silver_eel-class`, `report_species-class`

Examples

```r
# launching stacomi without connection to the database
stacomi(database_expected=FALSE)
# If you have a working database
# the following line of code will create the r_sample_char
data set from the iav (default) schema in the database
## Not run:
stacomi(database_expected=TRUE) # uses default option sch = 'iav'
# prompt for user and password, you can set these in the options,
# including dbname and host
if (interactive()){
  if (!exists("user")){
    user <- readline(prompt="Enter user: ")
    password <- readline(prompt="Enter password: ")
  }
}
options(
stacomiR.dbname = "bd_contmig_nat",
stacomiR.host = "localhost",
stacomiR.port = "5432",
stacomiR.user = user,
stacomiR.password = password
)

# create an instance of the class
r_sample_char <- new("report_sample_char")
# the following will load data for size,
# parameters 1786 (total size) C001 (size at video control)
# dc 5 and 6 are fishways located on the Arzal dam
# two stages are selected
r_sample_char <- choice_c(r_sample_char, dc=c(5,6),)
```
taxa=c("Anguilla anguilla"),
stage=c("AGJ","CIV"),
par=c(1785,1786,1787,"C001"),
horodatedebut="2013-01-01",
horodatefin="2013-12-31",
silent=FALSE)

# two warning produced, ignored if silent=TRUE
r_sample_char <- connect(r_sample_char)
r_sample_char <- calcule(r_sample_char,silent=TRUE)

## End(Not run)
# load the dataset generated by previous lines
data("r_sample_char")

# A "violin" plot
plot(r_sample_char,plot.type="1",silent=TRUE)

# get the plot from envir_stacomi to change labels for name
# if you use require(ggplot2) the :: argument is not needed
# e.g. write require(ggplot2);g<-get("g",envir=envir_stacomi)
# g+xlab("size")+ylab("year")
if (requireNamespace("ggplot2", quietly = TRUE)){
  g<-get("g",envir=envir_stacomi)
  g+ggplot2::xlab("size")+ggplot2::ylab("year")
}

# A boxplot per month
plot(r_sample_char,plot.type="2",silent=TRUE)

# A xyplot
plot(r_sample_char,plot.type="3",silent=TRUE)

## Not run:

#####################################
# an example graph created manually from data
#####################################
# two variables one on DC, one on stage
# passing dc information to the stage variable
r_sample_char@data$std_libelle[r_sample_char@data$sope_dic_identifiant==5]<-
  "Yellow eel (vert. slot fishway)"

r_sample_char@data$std_libelle[r_sample_char@data$std_libelle=="Anguille jaune"]<-
  "Yellow eel (ramp)"

r_sample_char@data$std_libelle[r_sample_char@data$std_libelle=="civelle"]<-
  "Glass eel (ramp)"

# creating a boxplot with custom output : an example
# again if you use require(ggplot2) the :: argument is not needed
if (requireNamespace("ggplot2", quietly = TRUE)){
  g<-ggplot2::ggplot(r_sample_char@data)+
  ggplot2::geom_boxplot(ggplot2::aes(x=annee,
    y = car_valeur_quantitatif,
    fill = std_libelle))+
  ggplot2::xlab("size")+ggplot2::ylab("year")+
  ggplot2::scale_fill_manual("stage & fishway",
    name="Stage & Fishway",
    labels=c("Anguille jaune", "Yellow eel (vert. slot fishway)","Yellow eel (ramp)", "Glass eel (ramp)"),
    values=c("Anguille jaune", "Yellow eel (vert. slot fishway)","Yellow eel (ramp)", "Glass eel (ramp)"))
  g
}

if (requireNamespace("ggplot2", quietly = TRUE)){
  g<-ggplot2::ggplot(r_sample_char@data)+
  ggplot2::geom_boxplot(ggplot2::aes(x=annee,
    y = car_valeur_quantitatif,
    fill = std_libelle))+
  ggplot2::xlab("size")+ggplot2::ylab("year")+
  ggplot2::scale_fill_manual("stage & fishway",
    name="Stage & Fishway",
    labels=c("Anguille jaune", "Yellow eel (vert. slot fishway)","Yellow eel (ramp)", "Glass eel (ramp)"),
    values=c("Anguille jaune", "Yellow eel (vert. slot fishway)","Yellow eel (ramp)", "Glass eel (ramp)"))
  g
}
values=c("Yellow eel (vert. slot fishway)"="blue","Yellow eel (ramp)"="turquoise3","Glass eel (ramp)"="Cyan"))+

ggplot2::theme_bw()
print(g)

# get a simple summary using Hmisc::describe
summary(r_sample_char)
# get the command line to create the object using choice_c
# when the graphical interface has been used
print(r_sample_char)

## End(Not run)

---

**report_sea_age-class**  
*Class “report_sea_age”*

**Description**

the report_sea_age class is used to dispatch adult salmons to age class according to their size and to basin dependent limits set by the user. Once checked with graphs and summary statistics, the results are to be written to the database.

**Slots**

data A data frame with data generated from the database
calcdata A list of dc with processed data. This lists consists of two elements  
  - (1) data A dataset with age set to be used by the plot and summary methods  
  - (2) tj_caracteristitiquelot_car A dataset to import into the database
dc Object of class ref_dc-class: the control devices
taxa Object of class ref_taxa-class: the species
stage Object of class ref_stage-class: the stages of the fish
par Object of class ref_par-class: the parameters used
horodatedebut An object of class ref_horodate-class
horodatefin An object of class ref_horodate-class
limit1hm The size limit, in mm between 1 sea winter fishes and 2 sea winter fishes
limit2hm The size limit, in mm between 2 sea winter fishes and 3 sea winter fishes

**Objects from the Class**

Objects can be created by calls of the form new("report_sea_age", ...)
**Note**

This class is displayed by interface `report_sea_age`

**Author(s)**

Cedric Briand <cedric.briand@eptb-vilaine.fr>

**See Also**

Other report Objects: `report_annual-class`, `report_dc-class`, `report_df-class`, `report_env-class`, `report_ge_weight-class`, `report_mig-class`, `report_mig_char-class`, `report_mig_env-class`, `report_mig_interannual-class`, `report_mig_mult-class`, `report_sample_char-class`, `report_silver_eel-class`, `report_species-class`

**Examples**

```r
require(stacomiR)
staconi(
  database_expected=FALSE)
# If you have a working database
# the following line of code will create the r_seaa dataset
# from the logrami schema in the database
## Not run:
staconi(database_expected=TRUE, sch='logrami')
# overrides the default option sch = 'iav'
# prompt for user and password, you can set these in the options,
# including dbname and host
if (interactive()){
  if (!exists("user")){
    user <- readline(prompt="Enter user: ")
    password <- readline(prompt="Enter password: ")
  }
}
options(
  stacomiR.dbname = "bd_contmig_nat",
  stacomiR.host ="localhost",
  stacomiR.port = "5432",
  stacomiR.user = user,
  stacomiR.user = password
)
# create an instance of the class
r_seaa<-new("report_sea_age")
  r_seaa<-choice_c(r_seaa,
    dc=c(107,108,101),
    horodatedebut="2012-01-01",
    horodatefin="2012-12-31",
    limit1hm=675,
    limit2hm=875,
    silent=FALSE
  )
  r_seaa<-connect(r_seaa)
  r_seaa<-calcule(r_seaa)
```
## End(Not run)
# load the dataset generated by previous lines
# Salmons from the loire on two dams
data("r_seaa")
# the calculation will fill the slot calcdata

# stages are in r_seaa@calcdata[["6"]][,"stage"]
# look at data structure using str(r_seaa@calcdata[["6"]])

# plot data to confirm the split by limits is correct
plot(r_seaa, plot.type=1)

# if there are several dc, data it split by dc
plot(r_seaa, plot.type=2)
## Not run:
# print a summary statistic, and save the output in a list for later use
stats<-summary(r_seaa)

write_database(r_seaa)
## End(Not run)

---

**report_silver_eel-class**

*Class* "report_silver_eel"

---

**Description**

The *report_silver_eel* class is used to calculate various statistics about the silver eel run. It comprises calculation of various maturation index such as Durif’s stages and Pankhurst eye index. The objective is to provide standardized output to the stations monitoring the silver eel run.

**Slots**

data A data frame with data generated from the database

calcdat A list of dc with processed data. Each dc contains a data frame with

- (1) qualitative data on body contrast (CONT), presence of punctuation on the lateral line (LINP)
- (2) quantitative data "BL" Body length,"W" weight,"Dv" vertical eye diameter,"Dh" horizontal eye diameter,"Fl" pectoral fin length
- (3) calculated durif stages, Pankhurst’s index, Fulton’s body weight coefficient K_ful
- (4) other columns containing data pertaining to the sample and the control operation: lot_identifiant, ope_identifiant, ope_dic_identifiant, ope_date_debut, ope_date_fin, dev_code (destination code of fish), dev_libelle (text for destination of fish)

dc Object of class ref_dc-class: the control devices
taxa  An object of class ref_taxa-class: the species  
stage  An object of class ref_stage-class: the stages of the fish  
par    An object of class ref_par-class: the parameters used  
horodatedebut An object of class ref_horodate-class  
horodatefin  An object of class ref_horodate-class

**Objects from the Class**

Objects can be created by calls of the form `new("report_silver_eel", ...)`

**Note**

This class is displayed by interface_report_silver_eel

**Author(s)**

Cedric Briand <cedric.briand@eptb-vilaine.fr>

**See Also**

Other report Objects: report_annual-class, report_dc-class, report_df-class, report_env-class, report_ge_weight-class, report_mig-class, report_mig_char-class, report_mig_env-class, report_mig_interannual-class, report_mig_mult-class, report_sample_char-class, report_sea_age-class, report_species-class

**Examples**

```r
# launching stacomi without selecting the scheme or interface
stacomi(database_expected=FALSE)
# the following script will load data from the two Anguillere monitored in the Somme

# If you have a working database
# the following line of code will create the r_silver dataset
# from the "fd80." schema in the database
## Not run:
stacomi(database_expected=TRUE, sch="fd80.") # overrides the default option sch = 'iav'
# prompt for user and password, you can set these in the options,
# including dbname and host
if (interactive()){  
  if (!exists("user")){  
    user <- readline(prompt="Enter user: ")  
    password <- readline(prompt="Enter password: ")  
  }  
}  
options(  
stacomiR.dbname = "bd_contmig_nat",  
stacomiR.host ="localhost",  
stacomiR.port = "5432",  
stacomiR.user = user,  
stacomiR.user = password  
```
#create an instance of the class
r_silver<-new("report_silver_eel")
r_silver<-choice_c(r_silver,
dc=c(2,6),
horodatedebut="2010-09-01",
horodatefin="2016-10-04",
silent=FALSE)
r_silver<-connect(r_silver)

## End(Not run)
# load the dataset generated by previous lines
data("r_silver")
# the calculation will fill the slot caldata
r_silver<-calcule(r_silver)
# stages are in r_silver@caldata[["6"]][,"stage"]
#look at data structure using str(r_silver@caldata[["6"]])

# standard plot as drawn by Laurent Beaulaton (Analyse des donnees d'argenture acquises en France)
# showing Durif's stage according to size and eye diameter
plot(r_silver, plot.type=1)

# number per month or year and Durif's stage (year if number of dc >1)
plot(r_silver, plot.type="2")

# plot showing fulton's coefficient, and size weight graphs
# inspired from Acou et al., 2009
# Differential Production and Condition Indices of Premigrant
# Eels in Two Small Atlantic Coastal Catchments
# of France
plot(r_silver, plot.type="3")

# get a list of summary data and print output to screen
plot(r_silver, plot.type="4")
# print a summary statistic, and save the output in a list for later use
stats<-summary(r_silver)

---

report_species-class  Counts of number per taxa/stages

**Description**

This class is used to make the assessment of all species, and their number. It is intended as a simple way to check what fishes are present (taxa + development stage). It was altered to include ref_taxa, to allow excluding some of the most numerous taxa from reports. The taxa is reported unless a taxa has several stages, in which case the different stages for the taxa will be reported. Using the split arguments the calc method of the class will count numbers, subsamples are not accounted for in the Overview. The split argument currently takes values year or month. The class is intended to be used over long periods e.g years. The plot method writes either an histogram or a pie chart of number per year/week/month.
Slots

dc  an object of class ref_dc-class

taxa Object of class ref_taxa-class: the species

start_year Object of class ref_year-class

end_year Object of class ref_year-class

data data.frame

calcdata data.frame with data processed by the calc method

split Object of class ref_list-class ref_list referential class choose within a list

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>

See Also

Other report Objects: report_annual-class, report_dc-class, report_df-class, report_env-class,

report_ge_weight-class, report_mig-class, report_mig_char-class, report_mig_env-class,

report_mig_interannual-class, report_mig_mult-class, report_sample_char-class, report_sea_age-class,

report_silver_eel-class

Examples

# launching stacomi without selecting the scheme or interface
stacomi( database_expected=FALSE)

# the following script will load data
# from the two Anguillere monitored in the Somme
# If you have a working database
# the following line of code will create the bilesp dataset from the "iav." 
# schema in the database

## Not run:
bilesp<-new("report_species")
# split is one of "none", "year", "week", "month"
bilesp<-choice_c(bilesp,
dc=c(5,6,12),
split="year",
start_year="2008",
end_year="2012",
silent=FALSE)
bilesp <- connect(bilesp)
bilesp <- calcule(bilesp)
plot(bilesp, plot.type="pie", silent=FALSE)
plot(bilesp, plot.type="barplot", silent=FALSE)
bilesp <- choice_c(bilesp,
dc=c(5,6,12),
split="month",
start_year="2015",
end_year="2016",
silent=FALSE)
```
bilesp <- charge(bilesp)
bilesp <- connect(bilesp)
plot(bilesp, plot.type="pie", silent=FALSE)
plot(bilesp, plot.type="barplot", silent=FALSE)
#length(unique(bilesp@calcdata$taxa_stage)) # 15
# here creating a vector of length 15 with nice blending colours
if (requireNamespace("grDevices", quietly = TRUE)) {
  mycolorrampblue <-
grDevices::colorRampPalette(c("#395B74", "#010F19"))
mycolorrampyellow <-
grDevices::colorRampPalette(c("#B59C53", "#271D00"))
mycolorrampred <-
grDevices::colorRampPalette(c("#B56F53", "#270B00"))
  color<-c(mycolorrampblue(5),
            mycolorrampyellow(5),
            mycolorrampred(5))
  plot(bilesp,plot.type="barplot",color=color,silent=TRUE)
}
summary(bilesp)
```

---

### r_ann

**Annual migration of yellow and silver eel for three fishways / counting devices at the Arzal dam (data from 1995 to 2016)**

**Description**

The dataset corresponds to the three fishways located on the Arzal dam, filled with annual data.

**Usage**

`r_ann`

**Format**

An object of class `report_annual-class` with data slot loaded.

### r_ann_adour

**Annual migration of salmon in the Adour and tributaries**

**Description**

The dataset corresponds to the fishways DC=33:40 of the Adour for adult migrant salmons from 1996 to 2005 (annual counts). It has been kindly provided as an example set by the Migradour association.
Usage
r_ann_adour

Format
An object of class report_annual-class with data slot loaded.

r_dc  Counting Device (DC) operation from 2000 to 2015 at the Arzal dam (Vilaine, France)

Description
This data corresponds to the data collected at the vertical slot fishway camera from 2000 to 2015. It represents an object of class report_dc-class with data loaded

Usage
r_dc

Format
An object of class report_dc with 4 slots:

data  A dataframe with 544 obs. of 7 variables
  per_dis_identifiant  The number of the DC
  per_date_debut  Starting time a POSIXct
  per_date_fin  Ending time a POSIXct
  ope_dic_identifiant  DC id
  per_commentaires  A comment
  per_etat_fonctionnement  Integer 1= working, 0 not working
  per_tar_code  The type of operation ('1'=normal operation, '2'=Device stopped in normal operation (e.g. the trap is disactivated for the duration of the fish sorting and counting by operators), '3'=Stopped for maintenance or other problem', '4'=Works but not fully operational, i.e. the camera is not working properly because of high turbidity...', '5'=Not known')
  libelle  label corresponding to per_tar_code

df  the ref_dc object with 3 slots filled with data corresponding to the iav postgres schema
horodatedebut  the ref_horodate with horodate set for starting date
horodatefin  the ref_horodate with horodate set for ending date
Overview of the fishway operation at Arzal in (Vilaine France).

**Description**

This dataset corresponds to the data collected at the vertical slot fishway in 2015, the fishway is working daily with a cycle depending on tide. This dataset is used to show an example of detailed output for an object of class `report_df-class` with data loaded.

**Usage**

```r
r_df
```

**Format**

An object of class `report_df` with 4 slots:

- **data**: A dataframe with 4261 obs. of 7 variables
  - `per_dis_identifiant`: The number of the DF
  - `per_date_debut`: Starting time a POSIXct
  - `per_date_fin`: Ending time a POSIXct
  - `ope_dic_identifiant`: DF id
  - `per_commentaires`: A comment
  - `per_etat_fonctionnement`: Integer 1= working, 0 not working
  - `per_tar_code`: The type of operation (‘1’=normal operation, ‘2’=Device stopped in normal operation (ie lift ascending, high tide...), ‘3’=Stopped for maintenance or other problem’, ‘4’=‘Works but not fully operational,i.e.flow problem, flood, clogged with debris...’, ‘5’=‘Not known’)
  - `libelle`: label corresponding to `per_tar_code`
- **df**: the `ref_df` object with 3 slots filled with data corresponding to the iav postgres schema
  - `horodatedebut`: the `ref_horodate` with horodate set for starting date
  - `horodatefin`: the `ref_horodate` with horodate set for ending date

---

**r_env**

An object of class `report_env` with data loaded

**Description**

The dataset corresponds to the daily temperatures and moon phases in Arzal (Vilaine estuary, France). This environmental station is used to analyze conditions in which fish migrated at Arzal dam.
r_gew

Usage

r_env

Format

An object of class report_env-class with data slot loaded:

- **stationMesure**: the ref_env object with 5 slots filled with data corresponding to the iav postgres schema
- **horodatedebut**: object of class ref_horodate-class: the start date selected
- **horodatefin**: object of class ref_horodate-class: the end date selected
- **data**: A dataframe with 723 rows and 6 variables
  - **env_date_debut**: start date
  - **env_date_fin**: end date
  - **env_methode_obtention**: method of data collection, measured, calculated...
  - **env_val_identifiant**: the value of the parameter if qualitative
  - **env_valeur_quantitatif**: the value of the parameter if quantitative
  - **env_stm_identifiant**: station id

---

r_gew

*Wet weight of glass eel from the trapping ladder (Arzal, Vilaine France)*

Description

Data correspond to glass eel collected in the Vilaine at the trapping ladder (Arzal, France). The years selected are 2009 to 2012, the query used in the report_ge_weight-class loads from 2008-08-01 to 2012-08-01. Glass eel are too numerous to be counted. They are weighted and in the stacomi database, a table with daily coefficients (in N glass eel/g) to transform weight into number. The weight is called a 'wet weight' as we don't want to drain any of the mucus in glass eel when weighting them. Samples of 50 to 200 glass eel are weighted and then counted to provide an idea of the seasonal evolution of wet weight.

Usage

r_gew

Format

An object of class report_ge_weight of length 1.
Description

This dataset corresponds to the data collected at the vertical slot fishway in 2015, video recording of the thin lipped mullet Liza ramada migration

Usage

r_mig

Format

An object of class report_mig with 8 slots:

- **dc** the ref_dc object with 4 slots filled with data corresponding to the iav postgres schema
- **taxa** the ref_taxa the taxa selected
- **stage** the ref_stage the stage selected
- **timestep** the ref_timestep_daily calculated for all 2015
- **data** A dataframe with 10304 rows and 11 variables
  - **ope_identifiant** operation id
  - **lot_identifiant** sample id
  - **lot_identifiant** sample id
  - **ope_dic_identifiant** dc id
  - **lot_tax_code** species id
  - **lot_std_code** stage id
  - **value** the value
  - **type_de_quantite** either effectif (number) or poids (weights)
  - **lot_dev_code** destination of the fishes
  - **lot_methode_obtention** method of data collection, measured, calculated...
- **coef_conversion** A data frame with 0 observations : no quantity are reported for video recording of mullets, only numbers
- **time.sequence** A time sequence generated for the report, used internally
**r_mig_char**

Qualitative and quantitative parameters describing Salmon migration at Decize (Loire)

Description

The dataset corresponds to the characteristics (qualitative and quantitative) of salmo salar migrating at Decize (Loire river) and Vichy (Allier river) counting device in 2012. It has been loaded as an example for the report_mig_char-class and kindly provided by Loire Grands Migrateurs (LOGRAIM).

Usage

r_mig_char

Format

An object of class report_mig_char-class with data slot loaded:

- **calcd**ata slot to be filled with the calcule method
- **data** A list of 2 elements
  - **parqual** values of all the qualitative parameters
  - **parquan** values of all the quantitative parameters
- **dc** the ref_dc : the control devices selected
- **taxa** the ref_taxa : Salmo salar selected
- **stage** the ref_stage : the stages selected
- **par** an object of class ref_par-class: the parameters used
- **horodatedebut** an object of class ref_horodate-class : the start date selected
- **horodatefin** an object of class ref_horodate-class : the end date selected

**r_mig_dc**

Counting device operation for the video recording (Arzal dam, Vilaine, France).

Description

This dataset corresponds to the data collected in the vertical slot fishway for the video recording operation. It is loaded along with r_mig to demonstrate the use of the report_mig-class when the database is not loaded

Usage

r_mig_dc
**r_mig_df**

*Fishway operation for the vertical slot fishway (Arzal dam, Vilaine, France).*

**Format**

An object of class `report_df-class`

**Description**

This dataset corresponds to the data collected at in the vertical slot fishway it is loaded along with `r_mig` and used to demonstrate the `report_mig-class` when the database is not installed.

**Usage**

`r_mig_df`

**r_mig_env**

An object of class `report_mig_env` with data loaded

**Description**

The dataset correspond to data loaded for the Arzal dam (Vilaine) in 2008, two quantitative parameters (temperature and tide coefficient) and a qualitative parameter (moon phase) are loaded.

**Usage**

`r_mig_env`

**Format**

An object of class `report_env-class` with data slot loaded:

- `report_mig_mult` An object of class `report_mig_mult-class`
- `report_env` An object of class `report_env-class`
**r_mig_interannual**

Daily glass eel and elver migration from 1984 to 2016 in the Sevre Niortaise

**Description**

The first eel trapping ladder in France was built by Antoine Legault and the team from Rennes in the Sevre Niortaise, Marais Poitevin. Also refurbished several times since 1984 it has been operational at the same location and provides one of the longest series of eel migration. For this reason, the dataset has been loaded as an example for the report_mig_interannual-class. It has been kindly provided by the parc du Marais Poitevin. The stage corresponds to small eels (elvers) less than 150 mmm stage name 'PANG'.

**Usage**

r_mig_interannual

**Format**

An object of class report_mig_interannual-class with data loaded.

---

**r_mig_interannual_vichy**

Seasonality of salmon migration at the Vichy counting station (Loire)

**Description**

This data corresponds to the data collected at the Vichy fishway between 1997 and 2012, video recording of the Salmo salar upstream migration. This dataset has been kindly provided by Loire Grands Migrateurs.

**Usage**

r_mig_interannual_vichy

**Format**

An object of class report_mig_interannual-class with 7 slots:

- dc the ref_dc object with 4 slots filled with data corresponding to the iav postgres schema
- taxa the ref_taxa the taxa selected
- stage the ref_stage the stage selected
- start_year the ref_timestep_daily calculated for all 2015
- end_year the ref_timestep_daily calculated for all 2015
The data corresponds to data collected from three fishways and correspond to the migration station at Arzal (Vilaine estuary, France) in 2011 for three continental stages of eel (Anguilla anguilla) : glass eel, yellow eel and silver eel.

**Usage**

```
> r_mig_mult
```

**Format**

An object of class report_mig_mult with slots:

- **dc** the ref_dc object filled with data
- **taxa** the ref_taxa object filled in with data corresponding to dc
- **stage** the ref_stage object filled in with data corresponding to dc, and taxa
- **timestep** the ref_timestep_daily calculated for all 2011

**Data** A dataframe with 400 rows and 11 variables

- **ope_identifiant** operation id
- **lot_identifiant** sample id
- **ope_dic_identifiant** dc id
- **lot_tax_code** species id
- **lot_std_code** stage id
- **value** the value
- **type_de_quantite** either effectif (number) or poids (weights)
- **lot_dev_code** destination of the fishes
- **lot_methode_obtention** method of data collection, measured, calculated...
**r_mig_mult_dc**

- **calcdatal** slot to be filled with the calcule method
- **coef_conversion** A data frame with 364 observations with daily coefficients to convert from weight to numbers
- **time.sequence** A time sequence generated for the report, used internally by the object

---

**r_mig_mult_dc**

*Counting device operation for three different counting device in Arzal (Vilaine, France)*

---

**Description**

This dataset corresponds to data collected at three different control devices. This object is of class `report_dc-class` with data loaded it is loaded along with `r_mig_mult` and used in demonstration for the `report_mig_mult-class`

**Usage**

`r_mig_mult_dc`

**Format**

An object of class `report_dc` with 4 slots

- **data** A dataframe with 25 rows and 7 variables
  - **per_dis_identifiant** the df or dc unique id
  - **per_date_debut** the starting date of the counting device operation `POSIXct`
  - **per_date_fin** the ending date of the counting device operation `POSIXct`
  - **per_commentaires** comments on the counting device operation
  - **per_etat_fonctionnement** Boolean, is the counting device working?
  - **lot_std_code** stage id
  - **per_tar_code** The type of operation for the DC, 1 normal operation, 2 device stopped in normal operation (the stop is considered as normal, e.g. you don’t monitor video if a cage has been placed to trap fishes), 3 stopped for maintenance or other problem, 4 the DC is working but not well (escapement in a tank, high turbidity preventing video counting...), 5 unknown operation.
  - **libelle** The label for the type or operation
- **dc** the `ref_dc` the DC with 4 slots
  - **dc_selected** the selected device
  - **ouvrage** the dam
  - **station** the monitoring station, a section of river
  - **data** A dataset of all dc present in the database with 10 observations
- **horodatedebut** the beginning date, a `ref_horodate-class`
- **horodatefin** the ending date, a `ref_horodate-class`
**Description**

This dataset corresponds to the data collected at three different fishways it is loaded along with *r_mig_mult* and used in demonstration for the *report_mig_mult*-class.

**Usage**

*r_mig_mult_df*

**Format**

An object of class *report_df*-class

**Description**

This dataset corresponds to the data collected at three different control devices. It is an object of class *report_ope*-class with data loaded. It is loaded along with *r_mig_mult*.

**Usage**

*r_mig_mult_ope*

**Format**

An object of class *report_ope*
**r_mig_ope**

An object of class `report_ope-class` with data loaded

### Description

This dataset corresponds to the data collected at the vertical slot fishway in Arzal (Vilaine river estuary, France). The operation of the fishway is dependent on tide and is recorded every 10 minutes. This dataset has to be loaded along with `r_mig` to demonstrate the use of the `report_mig-class`

### Usage

`r_mig_ope`

### Format

An object of class `report_ope`

---

**r_sample_char**

Size of yellow and glass eel at the Arzal dam (Vilaine, France) in the fishway and main eel trapping ladder.

### Description

This dataset corresponds to the data collected at two different control devices at the Arzal control station (see example in `report_sample_char-class`), all body size parameters (total size, size converted from pixel in video control) are used in example

### Usage

`r_sample_char`

### Format

An object of class `report_sample_char-class`
Description

This dataset corresponds to the data collected at Vichy (left and right bank fishways) and Decize-Saint Leger des Vignes fishways (respectively on the Allier and Loire river, France) in 2012 on the size structure of Salmo salar. It has been kindly provided by the Loire Grands Migrateurs (LOGRAMI) association.

Usage

r_seaa

Format

An object of class report_sea_age-class with 8 slots:

dc the ref_dc : the control devices selected
taxa the ref_taxa : Salmo salar selected
stage the ref_stage : the stages selected
par Object of class ref_par-class: the parameters used
horodatedebut object of class ref_horodate-class : the start date selected
horodatefin object of class ref_horodate-class : the end date selected
limit1hm The size limit, in mm between 1 sea winter fishes and 2 sea winter fishes
limit2hm The size limit, in mm between 2 sea winter fishes and 3 sea winter fishes
data A dataframe with 898 rows and 20 variables

  ope_identifiant operation id
  lot_identifiant sample id
  ope_dic_identifiant dc id
  ope_date_debut start date
  ope_date_fin end date
  lot_effectif number of fishes
  lot_tax_code species id
  lot_std_code stages id
  tax_nom_latin species latin names
  std_libelle stages names
  dev_code destination of the fishes id
  dev_libelle destination of the fishes names
  par_nom parameter name
  car_par_code parameter id
  car_methode_obtention method of data collection, measured, calculated...
car_valeur_quantitatif the value of the parameter
Description

The dataset corresponds to the silver eel traps (anguilleres) for 2015-2016. This dataset has been kindly provided by the Federation de Peche de la Somme, given the upstream location of the trap, most individuals are female.

Usage

r_silver

Format

An object of class report_silver_eel-class with data slot loaded.

setasqualitative

Generic method to transform quantitative par into a qualitative one

Description

Generic method to transform quantitative par into a qualitative one

Usage

setasqualitative(object, ...)

Arguments

object  Object
...  Additional parms

Author(s)

cedric.briand
setasqualitative, report_mig_char-method

*Turns a continuous parameter into discrete values*

**Description**

The parm name becomes "parm_discrete". New values are created in the `data[['parqual']]` slot of the report and the parqual slot is updated.

**Usage**

```r
## S4 method for signature 'report_mig_char'
setasqualitative(object, par, silent = FALSE, ...)
```

**Arguments**

- **object**: An object of class `ref_parquan-class`
- **par**: The code of a quantitative parameter
- **silent**: Default FALSE, if TRUE the program should not display messages
- **...**: Additional parms to the cut method `cut`

**Value**

An object of class `ref_parquan-class` with lines removed from `r@data[['parquan']]` and added (after transformation to qualitative values) in `r@data[['parqal']]`

**Author(s)**

Cedric Briand <cedric.briand@eptb-vilaine.fr>

---

**split_per_day**

*Create a dataframe suitable for charts per 24h and day*

**Description**

This functions takes a data frame with a column with starting time and another with ending time. If the period extends over midnight, it will be split into new lines, starting and ending at midnight.

**Usage**

```r
split_per_day(data, horodatedebut, horodatefin)
```
Arguments

data  The dataframe
horodatedebut  The beginning time
horodatefin  The ending time

Value

A data frame with four new columns, Hmin (hour min), Hmax (hmax), xmin (day) and xmax (next day), and new rows

Author(s)

cedric.briand

Examples

datatemp<-structure(list(per_dis_identifiant = c(1L, 1L, 1L),
   per_date_debut = structure(c(1420056600,
      1420071000, 1420081200), class = c("POSIXct", "POSIXt"), tzone = ""),
   per_date_fin = structure(c(1420071000, 1420081200, 1421000000
      ), class = c("POSIXct", "POSIXt"), tzone = ""), per_commentaires = c("fonct calcul",
      "fonct calcul"), per_etat_fonctionnement = c(1L,
      0L, 0L), per_tar_code = 1:3, libelle = c("Fonc normal", "Arr ponctuel",
      "Arr maint")), .Names = c("per_dis_identifiant", "per_date_debut",
      "per_date_fin", "per_commentaires", "per_etat_fonctionnement",
      "per_tar_code", "libelle"), row.names = c(NA, 3L), class = "data.frame")

e = strsplit(per_date_debut, ",")

datatemp$Hmin = unlist(lapply(e, function(x) as.POSIXct(x[1], format="%H:%M:
      %S", tz="")))
datatemp$Hmax = unlist(lapply(e, function(x) as.POSIXct(x[2], format="%H:
      %M:%S", tz="")))
datatemp$xmin = unlist(lapply(e, function(x) as.POSIXt(x[1], tz="")))
datatemp$xmax = unlist(lapply(e, function(x) as.POSIXt(x[2], tz="")))

newdf<-split_per_day(data=datatemp,horodatedebut="per_date_debut",
   horodatefin="per_date_fin")
summary,report_dc-method

Value

Nothing, called for its side effect of loading

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>

Examples

```r
require(stacomiR)
#launch stacomi
## Not run:
stacomi(database_expected=TRUE, datawd='~',sch= "iav")

## End(Not run)
# launch stacomi without connection to the database
stacomi(database_expected=FALSE)
# launch stacomi with options
options(
stacomiR.dbname = "bd_contmig_nat",
stacomiR.host = readline(prompt = "Enter host: "),
stacomiR.port = "5432",
stacomiR.user = readline(prompt = "Enter user: "),
stacomiR.password = readline(prompt = "Enter password: "))
```

Description

summary for report_dc, write csv and html output, and prints summary statistics

Usage

```r
## S4 method for signature 'report_dc'
summary(object, silent = FALSE, ...)
```

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>object</td>
<td>An object of class <code>report_dc-class</code></td>
</tr>
<tr>
<td>silent</td>
<td>Should the program stay silent or display messages, default FALSE</td>
</tr>
<tr>
<td>...</td>
<td>Additional parameters (not used there)</td>
</tr>
</tbody>
</table>
Value

Nothing, called for its side effect of writing html, csv files and printing summary

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>

Description

summary for report_df, write csv and html output, and prints summary statistics

Usage

```r
## S4 method for signature 'report_df'
summary(object, silent = FALSE, ...)
```

Arguments

- `object` An object of class `report_df-class`
- `silent` Should the program stay silent or display messages, default FALSE
- `...` Additional parameters (not used there)

Value

Nothing, called for its side effect of writing html, csv files and printing summary

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>
**summary,report_mig-method**

*summary for report_mig* calls functions funstat and funtable to create migration overviews and generate csv and html output in the user data directory.

---

**Description**

*summary for report_mig* calls functions funstat and funtable to create migration overviews and generate csv and html output in the user data directory.

**Usage**

```r
## S4 method for signature 'report_mig'
summary(object, silent = FALSE, ...)
```

**Arguments**

- `object`: An object of class *report_mig-class*
- `silent`: Should the program stay silent or display messages, default FALSE
- `...`: Additional parameters (not used there)

**Value**

Nothing, calls the `summary.report_mig_mult` method

**Author(s)**

Cedric Briand <cedric.briand@eptb-vilaine.fr>

---

**summary,report_mig_char-method**

*summary for report_mig_char*

---

**Description**

*summary for report_mig_char*

**Usage**

```r
## S4 method for signature 'report_mig_char'
summary(object, silent = FALSE, ...)
```
**Arguments**

- **object**  
  An object of class `report_mig_char-class`
- **silent**  
  Should the program stay silent or display messages, default FALSE
- ...  
  Additional parameters

**Value**

A table with the summary

**Author(s)**

Cedric Briand <cedric.briand@eptb-vilaine.fr>

**Description**

Summary for `report_mig_interannual` provides summary statistics for the latest year (if silent=TRUE), or the year selected in the interface, if silent=FALSE. Mean, min and max are historical statistics with the selected year excluded from the historical dataset.

**Usage**

```r
## S4 method for signature 'report_mig_interannual'
summary(object, year_choice = NULL, silent = FALSE, ...)
```

**Arguments**

- **object**  
  An object of class `report_mig_interannual-class`
- **year_choice**  
  The year chosen to calculate statistics which will be displayed beside the historical series,
- **silent**  
  Should the program stay silent or display messages, default FALSE
- ...  
  Additional parameters (not used there)

**Value**

A list, one element per DC

**Author(s)**

Cedric Briand <cedric.briand@eptb-vilaine.fr>
summary,report_mig_mult-method

summary for report_mig_mult calls functions funstat and funtable to create migration overviews and generate csv and html output in the user data directory

Description

summary for report_mig_mult calls functions funstat and funtable to create migration overviews and generate csv and html output in the user data directory

Usage

## S4 method for signature 'report_mig_mult'
summary(object, silent = FALSE, ...)

Arguments

object An object of class `report_mig_mult-class`
silent Should the program stay silent or display messages, default FALSE
... Additional parameters (not used there)

Value

Nothing, runs funstat and funtable method for each DC

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>

summary,report_sample_char-method

summary for report_sample_char

Description

summary for report_sample_char

Usage

## S4 method for signature 'report_sample_char'
summary(object, silent = FALSE, ...)
summary,report_sea_age-method

Arguments

object An object of class report_sample_char-class
silent Should the program stay silent or display messages, default FALSE
... Additional parameters

Value

Nothing, called for its side effect of printing a summary

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>
summary,report_silver_eel-method

summary for report_silver_eel

**Description**

summary for report_silver_eel

**Usage**

```r
## S4 method for signature 'report_silver_eel'
summary(object, silent = FALSE, ...)
```

**Arguments**

- `object` An object of class `report_silver_eel-class`
- `silent` Should the program stay silent or display messages, default FALSE
- `...` Additional parameters

**Value**

A list per DC with statistic for Durif stages, Pankhurst, MD Eye diameter, BL body length and weight W

**Author(s)**

Cedric Briand <cedric.briand@eptb-vilaine.fr>

summary,report_species-method

summary for report_species

**Description**

generate csv and html output in the user data directory

**Usage**

```r
## S4 method for signature 'report_species'
summary(object, silent = FALSE)
```

**Arguments**

- `object` An object of class `report_species-class`
- `silent` Should the program stay silent or display messages, default FALSE
**supprime**

**Value**
nothing, but writes summary in `get("datawd", envir = envir_stacomi)`, and prints output

**Author(s)**
Cedric Briand <cedric.briand@eptb-vilaine.fr>

---

**supprime**

*Generic method to delete entries from the database*

**Description**
Generic method to delete entries from the database

**Usage**
supprime(object, ...)

**Arguments**
- **object**: Object
- **...**: Additional parms

**Author(s)**
cedric.briand

**See Also**

---

**supprime, ref_coe-method**

*supprime method for 'ref_coe' class*

**Description**
supprime method for 'ref_coe' class

**Usage**
## S4 method for signature 'ref_coe'
supprime(object, tax, std, silent = FALSE)
supprime_report_mig_interannual-method

Arguments

  object  An object of class ref_coe-class
  tax     '2038=Anguilla anguilla'
  std     'CIV=civelle'
  silent  Default FALSE, if TRUE the program should no display messages

Value

  Nothing, called for side effect

Author(s)

  Cedric Briand <cedric.briand@eptb-vilaine.fr>

supprime_report_mig_interannual-method

supprime method for report_mig_interannual class, deletes values in table t_bilanmigrationjournalier_bjo

Description

  supprime method for report_mig_interannual class, deletes values in table t_bilanmigrationjournalier_bjo

Usage

  ## S4 method for signature 'report_mig_interannual'
supprime(object)

Arguments

  object  An object of class report_mig_interannual-class

Value

  nothing, called for its side effect, removing lines from the database

Author(s)

  Cedric Briand <cedric.briand@eptb-vilaine.fr>
**supprime.report_sea_age-method**

Description

Supprime method for report_mig_interannual class

Usage

```r
## S4 method for signature 'report_sea_age'
supprime(object, silent = FALSE)
```

Arguments

- **object**
  - An object of class `report_sea_age-class`
- **silent**
  - Default FALSE, if TRUE the program should no display messages

Value

Nothing, called for its side effect of deleting data in the database

Author(s)

Cedric Briand <cedric.briand@epbt-vilaine.fr>

---

**vector_to_listsql**  

Transforms a vector into a string called within an sql command e.g. c('A','B','C') => in ('A','B','C')

Description

Transforms a vector into a string called within an sql command e.g. c(A,B,C) => in ('A','B','C')

Usage

```
vector_to_listsql(vect)
```

Arguments

- **vect**
  - a character vector

Value

A list of value
write_database

Generic method write_database

Usage

write_database(object, ...)

Arguments

object Object
...

Author(s)

cedric.briand

write_database,report_ge_weight-method

Method to write data to the stacomi database for report_ge_weight-class

Description

Data will be written in tj_coefficientconversion_coe table, if the class retrieves some data from the database, those will be deleted first.

Usage

## S4 method for signature 'report_ge_weight'
write_database(object, silent = FALSE)

Arguments

object An object of class report_ge_weight-class
silent Boolean, if TRUE, information messages are not displayed

Value

Nothing, called for its side effect of writing to the database
write_database, report_mig-method

Command line method to write the daily and monthly counts to the t_bilanmigrationjournalier_bjo table

Description

Daily values are needed to compare migrations from year to year, by the class report_mig_interannual-class. They are added by by this function.

Usage

## S4 method for signature 'report_mig'
write_database(object, silent = TRUE, check_for_bjo = TRUE)

Arguments

object an object of class report_mig
silent : TRUE to avoid messages
check_for_bjo : do you want to check if data are already present in the bjo table, and delete them, this param was added otherwise connect method when called from report_mig_interannual runs in loops

Value

Nothing, just writes data into the database

Note

the user is asked whether or not he wants to overwrite data only when silent is FALSE, if no data are present in the database, the import is done anyway.

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>

Examples

## Not run:
stacomi(database_expected=FALSE)
data("r_mig")
r_mig<-calcule(r_mig)
write_database(report_mig=r_mig,silent=FALSE)

## End(Not run)
Command line method to write the characteristic "sea age" (car_par_code='A124') into the tj_caracteristiquelot_car table in the user's scheme.

Description

The sea age characteristic is calculated from the measured or calculated size of salmon and with a size/age rule defined by the user.

Usage

```r
## S4 method for signature 'report_sea_age'
write_database(object, silent = TRUE)
```

Arguments

- `object`: an object of class `report_sea_age-class`
- `silent`: Default FALSE, if TRUE the program should no display messages.

Value

Nothing, called for its side effect of writing data to the database

Author(s)

Cedric Briand <cedric.briand@eptb-vilaine.fr>

xtable function for `report_annual-class` create an xtable objet but also assigns an add.to.column argument in `envir_stacomi`, for later use by the `print.xtable` method.

Description

xtable function for `report_annual-class` create an xtable objet but also assigns an add.to.column argument in `envir_stacomi`, for later use by the `print.xtable` method.
Usage

```r
## S4 method for signature 'report_annual'
xtable(
x, 
caption = NULL,
label = NULL,
align = NULL,
digits = 0,
display = NULL,
auto = FALSE,
dc_name = NULL,
tax_name = NULL,
std_name = NULL
)
```

Arguments

- `x`, an object of class "report_annual"
- `caption`, see `xtable`
- `label`, see `xtable`
- `align`, see `xtable`, overridden if NULL
- `digits`, default 0
- `display`, see `xtable`
- `auto`, see `xtable`
- `dc_name`, A string indicating the names of the DC, in the order of `x@dc@dc_selected` if not provided DC codes are used.
- `tax_name`, A string indicating the names of the taxa, if not provided Latin names are used
- `std_name`, A string indicating the stages names, if not provided then `std_libelle` are used

Value

A `xtable` for annual report

Description

`xtable` function for `report_mig_char-class` create an `xtable` object to be later used by the `print.xtable` method.
Usage

```r
## S4 method for signature 'report_mig_char'
xtable(
  x,
  caption = NULL,
  label = NULL,
  align = NULL,
  digits = NULL,
  display = NULL,
  auto = FALSE,
  ...
)
```

Arguments

- **x**: an object of class 'report_mig_char'
- **caption**: see xtable
- **label**: see xtable
- **align**: see xtable, overridden if NULL
- **digits**: see xtable
- **display**: see xtable
- **auto**: see xtable
- **...**: Additional parameters

Value

A xtable
Index

* classes
  ref_coe-class, 97
  ref_dc-class, 98
  ref_env-class, 99
  ref_par-class, 100
  ref_parquan-class, 101
  ref_period-class, 102
  ref_stage-class, 102
  ref_timestep_daily-class, 105
  ref_timestepChar-class, 105
  ref_year-class, 106
  report_annual-class, 107
  report_dc-class, 109
  report_df-class, 110
  report_env-class, 112
  report_ge_weight-class, 113
  report_mig-class, 116
  report_mig_char-class, 118
  report_mig_env-class, 120
  report_mig_interannual-class, 122
  report_mig_mult-class, 126
  report_ope-class, 129
  report_sample_char-class, 129
  report_sea_age-class, 132
  report_silver_eel-class, 134
  report_species-class, 136

* datasets
  coef_durif, 55
  envir_stacomi, 66
  r_gew, 141

* data
  r_ann, 138
  r_ann_adour, 138
  r_dc, 139
  r_df, 140
  r_env, 140
  r_mig, 142
  r_mig_char, 143
  r_mig_dc, 143
  r_mig_df, 144
  r_mig_env, 144
  r_mig_interannual, 145
  r_mig_interannual_vichy, 145
  r_mig_mult, 146
  r_mig_mult_dc, 147
  r_mig_mult_df, 148
  r_mig_mult_ope, 148
  r_mig_ope, 149
  r_sample_char, 149
  r_seaa, 150
  r_silver, 151

* environment
  envir_stacomi, 66

* referential objects
  charge, ref_choice-method, 15
  ref_choice-class, 97
  ref_coe-class, 97
  ref_dc-class, 98
  ref_df-class, 99
  ref_horodate-class, 100
  ref_par-class, 100
  ref_parqual-class, 101
  ref_parquan-class, 101
  ref_period-class, 102
  ref_stage-class, 102
  ref_taxa-class, 103
  ref_year-class, 106

* report Objects
  report_annual-class, 107
  report_dc-class, 109
  report_df-class, 110
  report_env-class, 112
  report_ge_weight-class, 113
  report_mig-class, 116
  report_mig_char-class, 118
  report_mig_env-class, 120
  report_mig_interannual-class, 122
  report_mig_mult-class, 126
  report_ope-class, 129
  report_sample_char-class, 129
  report_sea_age-class, 132
  report_silver_eel-class, 134
  report_species-class, 136
  report_annual-class, 107
  report_dc-class, 109
  report_df-class, 110
  report_env-class, 112
  report_ge_weight-class, 113
  report_mig-class, 116
  report_mig_char-class, 118
  report_mig_env-class, 120
  report_mig_interannual-class, 122
  report_mig_mult-class, 126
REPORT SAMPLE CHAR-CLASS, 129
REPORT SEA AGE-CLASS, 132
REPORT SILVER EEL-CLASS, 134
REPORT SPECIES-CLASS, 136

* REPORT OBJECT
  REF_Timestep-class, 104
  REPORT_MIG_CHAR-CLASS, 118
  REPORT_OPE-CLASS, 129
' (CONNECT, REPORT_MIG_MULT-METHOD), 62

BARPLOT, REPORT_ANNUAL-METHOD, 6
BARPLOT.REPORT_ANNUAL
  (BARPLOT, REPORT_ANNUAL-METHOD), 6

CALCULE, 7
CALCULE, REPORT_GE_WEIGHT-METHOD, 7
CALCULE, REPORT_MIG-METHOD, 8
CALCULE, REPORT_MIG_CHAR-METHOD, 9
CALCULE, REPORT_MIG_ENV-METHOD, 9
CALCULE, REPORT_MIG_INTERANNUAL-METHOD, 10
CALCULE, REPORT_MIG_MULT-METHOD, 11
CALCULE, REPORT_SAMPLE_CHAR-METHOD, 12
CALCULE, REPORT SEA AGE-METHOD, 12
CALCULE, REPORT SILVER EEL-METHOD, 13
CALCULE, REPORT SPECIES-METHOD, 14
CALCULE.REPORT_GE_WEIGHT, 161
CALCULE.REPORT_GE_WEIGHT
  (CALCULE, REPORT_GE_WEIGHT-METHOD), 7
CALCULE.REPORT_MIG
  (CALCULE, REPORT_MIG-METHOD), 8
CALCULE.REPORT_MIG_CHAR, 161
CALCULE.REPORT_MIG_CHAR
  (CALCULE, REPORT_MIG_CHAR-METHOD), 9
CALCULE.REPORT_MIG_ENV, 161
CALCULE.REPORT_MIG_ENV
  (CALCULE, REPORT_MIG_ENV-METHOD), 9
CALCULE.REPORT_MIG_INTERANNUAL, 161
CALCULE.REPORT_MIG_INTERANNUAL
  (CALCULE, REPORT_MIG_INTERANNUAL-METHOD) CHARGE.REPORT_MIG_MULT
  (CALCULE, REPORT_MIG_MULT-METHOD), 11
CALCULE.REPORT_MIG_MULT, 161
CALCULE.REPORT_MIG_MULT
  (CALCULE, REPORT_MIG_MULT-METHOD), 11

CALCULE.REPORT_SAMPLE_CHAR, 161
CALCULE.REPORT_SAMPLE_CHAR
  (CALCULE, REPORT_SAMPLE_CHAR-METHOD), 12
CALCULE.REPORT SEA AGE, 161
CALCULE.REPORT SEA AGE
  (CALCULE, REPORT SEA AGE-METHOD), 12
CALCULE.REPORT SILVER EEL, 161
CALCULE.REPORT SILVER EEL
  (CALCULE, REPORT SILVER EEL-METHOD), 13
CALCULE.REPORT SPECIES, 161
CALCULE.REPORT SPECIES
  (CALCULE, REPORT SPECIES-METHOD), 14
CHARGE, 14
CHARGE, REF_CHOICE-METHOD, 15
CHARGE, REF_COE-METHOD, 16
CHARGE, REF_DC-METHOD, 16
CHARGE, REF_DF-METHOD, 17
CHARGE, REF_ENV-METHOD, 18
CHARGE, REF_LIST-METHOD, 18
CHARGE, REF_PAR-METHOD, 19
CHARGE, REF_PARQUAL-METHOD, 20
CHARGE, REF_PARQUAN-METHOD, 21
CHARGE, REF_STAGE-METHOD, 21
CHARGE, REF_TAXA-METHOD, 22
CHARGE, REF_TEXTBOX-METHOD, 23
CHARGE, REF_YEAR-METHOD, 24
CHARGE, REPORT_MIG-METHOD, 25
CHARGE, REPORT_MIG_ENV-METHOD, 25
CHARGE, REPORT_MIG_MULT-METHOD, 25, 26
CHARGE, REF_LIST
  (CHARGE, REF_LIST-METHOD), 18
CHARGE, REF_PAR (CHARGE, REF_PAR-METHOD), 19
CHARGE, REPORT_MIG
  (CHARGE, REPORT_MIG-METHOD), 25
CHARGE, REPORT_MIG_ENV
  (CHARGE, REPORT_MIG_ENV-METHOD), 25
CHARGE, REPORT_MIG_MULT
  (CHARGE, REPORT_MIG_MULT-METHOD), 26
CHARGE_COMPLEMENT, 27
CHARGE_COMPLEMENT, REF_PARQUAL-METHOD, 27
INDEX

charge_with_filter,ref_par-method, 28
charge_with_filter,ref_parqual-method, 29
charge_with_filter,ref_parquan-method, 30
charge_with_filter,ref_stage-method, 31
charge_with_filter,ref_taxa-method, 32
charge_with_filter.ref_par (charge_with_filter,ref_par-method), 28
choice_c,ref_choice-method, 32
choice_c,ref_dc-method, 33, 43, 46, 47, 49–52, 54, 55
choice_c,ref_df-method, 34
choice_c,ref_env-method, 35, 45, 49
choice_c,ref_horodate-method, 36
choice_c,ref_list-method, 36
choice_c,ref_par-method, 37, 51
choice_c,ref_stage-method, 38, 43, 47, 50, 51
choice_c,ref_taxa-method, 39, 43, 46, 47, 49–52, 54, 55
choice_c,refTextbox-method, 40
choice_c,ref_timestep_daily-method, 40
choice_c,ref_year-method, 41
choice_c,report_annual-method, 42
choice_c,report_dc-method, 43
choice_c,report_df-method, 44
choice_c,report_env-method, 45
choice_c,report_ge_weight-method, 45
choice_c,report_mig-method, 46
choice_c,report_mig_char-method, 47
choice_c,report_mig_env-method, 48
choice_c,report_mig_interannual-method, 49
choice_c,report_mig_mult-method, 50
choice_c,report_sample_char-method, 51
choice_c,report_sea_age-method, 52
choice_c,report_silver_eel-method, 53
choice_c,report_species-method, 54
choice_c.ref_horodate (choice_c,ref_horodate-method), 36
choice_c.ref_list (choice_c,ref_list-method), 36
choice_c.ref_par (choice_c,ref_par-method), 37
choice_c.report_annual (choice_c,report_annual-method), 42
choice_c.report_dc (choice_c,report_dc-method), 43
choice_c.report_df (choice_c,report_df-method), 44
choice_c.report_env (choice_c,report_env-method), 45
choice_c.report_ge_weight (choice_c,report_ge_weight-method), 45
choice_c.report_mig_char (choice_c,report_mig_char-method), 47
choice_c.report_mig_env (choice_c,report_mig_env-method), 48
choice_c.report_mig_interannual (choice_c,report_mig_interannual-method), 49
choice_c.report_mig_mult (choice_c,report_mig_mult-method), 50
choice_c.report_sample_char (choice_c,report_sample_char-method), 51
choice_c.report_sea_age (choice_c,report_sea_age-method), 52
choice_c.report_silver_eel (choice_c,report_silver_eel-method), 53
choice_c.report_species (choice_c,report_species-method), 54
coef_durif, 55
colortable, 56
connect,report_annual-method, 57
connect,report_dc-method, 57
connect,report_df-method, 58
connect,report_env-method, 59
connect,report_ge_weight-method, 59
connect,report_mig_char-method, 60
connect, report_mig_env-method, 61
connect, report_mig_interannual-method, 61
connect, report_mig_mult-method, 62
connect, report_ope-method, 63
connect, report_sample_char-method, 63
connect, report_sea_age-method, 64
connect, report_silver_eel-method, 64
connect, report_species-method, 65
connect, report_annual
  (connect, report_annual-method), 57
connect, report_dc
  (connect, report_dc-method), 57
connect, report_df
  (connect, report_df-method), 58
connect, report_env
  (connect, report_env-method), 59
connect, report_ge_weight
  (connect, report_ge_weight-method), 59
connect, report_mig_char
  (connect, report_mig_char-method), 60
connect, report_mig_interannual
  (connect, report_mig_interannual-method), 61
connect, report_mig_mult
  (connect, report_mig_mult-method), 62
connect, report_ope
  (connect, report_ope-method), 63
connect, report_sample_char
  (connect, report_sample_char-method), 63
connect, report_sea_age
  (connect, report_sea_age-method), 64
connect, report_silver_eel
  (connect, report_silver_eel-method), 64
connect, report_species
  (connect, report_species-method), 65
cut, 152
envir_stacomi, 66
fun_aggreg_for_plot, 71
fun_char_spe, 71
fun_date_extraction, 72
fun_report_mig_interannual, 73
fun_report_mig_mult, 8, 11, 73
fun_report_mig_mult_overlaps, 8, 11, 74,
  74
fun_schema, 75
fun_stage_durif, 75
fun_table_per_dis, 76
fun_weight_conversion, 8, 11, 77
fun_write_monthly, 77
fungraph, 62, 66, 86, 89, 90
fungraph_glasseel, 62, 68, 86, 89, 90
funstat, 69, 78
funtab, 70
getvalue, 78
getvalue, ref_period-method, 78
graphdate, 79
model, 80
model, report_ge_weight-method, 80
model, report_ge_weight
  (model, report_ge_weight-method), 80
plot, report_annual, missing-method, 81
plot, report_dc, missing-method, 82
plot, report_df, missing-method, 83
plot, report_env, missing-method, 84
plot, report_ge_weight, missing-method, 85
plot, report_mig, ANY-method, 86
plot, report_mig_char, missing-method, 87
plot, report_mig_env, missing-method, 88
plot, report_mig_interannual, missing-method, 88
plot, report_mig_mult, missing-method, 89
plot, report_sample_char, missing-method, 90
plot, report_sea_age, missing-method, 91
plot, report_silver_eel, missing-method, 92
plot, report_species, missing-method, 93
plot, report_annual
  (plot, report_annual, missing-method), 81
plot.report_dc
  (plot,report_dc,missing-method), 82
plot.report_df
  (plot,report_df,missing-method), 83
plot.report_env
  (plot,report_env,missing-method), 84
plot.report_ge_weight
  (plot,report_ge_weight,missing-method), 85
plot.report_mig
  (plot,report_mig,ANY-method), 86
plot.report_mig_char
  (plot,report_mig_char,missing-method), 87
plot.report_mig_env
  (plot,report_mig_env,missing-method), 88
plot.report_mig_interannual
  (plot,report_mig_interannual,missing-method), 88
plot.report_mig_mult
  (plot,report_mig_mult,missing-method), 89
plot.report_sample_char
  (plot,report_sample_char,missing-method), 90
plot.report_sea_age
  (plot,report_sea_age,missing-method), 91
plot.report_silver_eel
  (plot,report_silver_eel,missing-method), 92
plot.reportreport_species
  (plot,report_species,missing-method), 93
print,report_dc-method, 93
print,report_df-method, 94
print,report_mig_mult-method, 95
print,report_sample_char-method, 95
print,report_sea_age-method, 96
print,report_silver_eel-method, 96
print.report_dc
  (print,report_dc-method), 93
print.report_df
  (print,report_df-method), 93
print.report_mig_mult
  (print,report_mig_mult-method), 95
print.report_sample_char
  (print,report_sample_char-method), 95
print.report_sea_age
  (print,report_sea_age-method), 96
print.report_silver_eel
  (print,report_silver_eel-method), 96
r_ann
  138
r_ann_adour
  138
r_dc
  139
r_df
  140
r_env
  140
r_gew
  141
r_mig
  142, 143, 144, 149
r_mig_char
  143
r_mig_dc
  143
r_mig_df
  144
r_mig_env
  144
r_mig_interannual
  145
r_mig_interannual_vichy
  145
r_mig_mult
  146, 147, 148
r_mig_mult_dc
  147
r_mig_mult_df
  148
r_mig_mult_ope
  148
r_mig_ope
  149
r_sample_char
  149
r_sea
  150
r_silver
  151
ref_choice-class
  15, 33, 97, 118
ref_coe-class
  16, 46, 59, 97, 162
ref_dc-class
ref_df-class
  17, 34, 44, 99
ref_env-class
  18, 35, 45, 99, 112
ref_horodate-class
  36, 45, 48, 51–53, 100, 112, 118, 129, 135, 147
ref_list-class
  19, 37, 46
ref_par-class
  19, 28, 37, 38, 48, 51–53, 100, 118, 129, 132, 135, 143, 150
ref_parqual-class
  20, 27, 29, 101, 118
ref_parquan-class
  21, 30, 101, 118, 152
ref_period-class
  78, 102
INDEX

summary.report_mig_env
  (connect,report_mig_env-method), 61
summary.report_mig_interannual
  (summary,report_mig_interannual-method), 157
summary.report_mig_mult, 156
summary.report_mig_mult
  (summary,report_mig_mult-method), 158
summary.report_sample_char
  (summary,report_sample_char-method), 158
summary.report_sea_age
  (summary,report_sea_age-method), 159
summary.report_silver_eel
  (summary,report_silver_eel-method), 160
summary.report_species
  (summary,report_species-method), 160
supprime, 161
supprime,ref_coe-method, 161
supprime,report_mig_interannual-method, 162
supprime,report_sea_age-method, 163
supprime,report_mig_interannual
  (supprime,report_mig_interannual-method), 162
supprime,report_sea_age
  (supprime,report_sea_age-method), 163

vector_to_listsql, 163
write_database, 164
write_database,report_ge_weight-method, 164
write_database,report_mig-method, 122, 165
write_database,report_sea_age-method, 166
write_database.report_ge_weight
  (write_database,report_ge_weight-method), 164
write_database.report_mig
  (write_database,report_mig-method), 165
write_database.report_sea_age
  (write_database,report_sea_age-method), 166
xtable,report_annual-method, 166
xtable,report_mig_char-method, 167
xtable.report_annual
  (xtable,report_annual-method), 166
xtable.report_mig_char
  (xtable,report_mig_char-method), 167