

Package ‘qoma.smuggler’

August 30, 2018

Title Transport Data and Commands Across the 'FAME' / 'R' Border

Version 0.0.1

Description Transport data and commands across the 'FAME' <<https://fame.sungard.com/support.html>> / 'R' border. A set of utilities for: reading 'FAME' databases into 'R'; writing 'R' data into 'FAME' databases; executing 'FAME' commands in 'R' environment; and, executing 'R' commands from the 'FAME' environment.

Depends R (>= 3.5)

Imports rhli (>= 0.0.2), lubridate(>= 1.7.4), tibble(>= 1.4.2), methods

License AGPL

Encoding UTF-8

LazyData true

Suggests testthat

RoxygenNote 6.1.0

URL <https://github.com/qomaio/r-smuggler/>

BugReports <https://github.com/qomaio/r-smuggler/issues>

NeedsCompilation no

Author Kevin Keane [aut, cre],
Qoma LLC [cph]

Maintainer Kevin Keane <krkeane@qoma.io>

Repository CRAN

Date/Publication 2018-08-30 16:20:06 UTC

R topics documented:

close_hli	2
List-class	2
meta_to_string	3
newEntry	4

open_hli	4
print_catalog	5
print_stack	5
read_fame	6
to_fame_range	6
to_lubridate_index	7
write_fame	7

Index	9
--------------	----------

close_hli	<i>Close the FAME host language interface.</i>
-----------	--

Description

Close the FAME host language interface.

Usage

close_hli()

List-class	<i>Mutable list</i>
------------	---------------------

Description

Mutable list

Methods

get(objnam = NULL) Get an element of the List
 get_data(objnam = NULL) Get data
 get_meta(objnam = NULL) Get meta data
 initialize(l0 = list()) Initialize a List.
 put(key, value) Put a key,value pair into the List

Examples

```

# set FAME monthly date range January 2018 to December 2018
rng <- to_fame_range(rhli::HMONTH,"18m1","18m12")
# convert to equivalent lubridate date column
tbl <- to_lubridate_index(rng)
# generate N(0,1) random observations
nobs <- rng[3]-rng[2]+1
tbl['x'] <- rnorm(nobs)
# construct List entry containing data and FAME metadata
mydb <- List()
entry <- newEntry(tbl$x,
  desc = "N(0,1)",
  docu = "R generated N(0,1) time series.",
  range = rng,obse = rhli::HOBSUM )
# put key='x',value=entry in List
mydb$put('x',entry)
# display contents of List
print_catalog(mydb)
# retrieve value for key 'x' from List
mydb$get('x')

```

meta_to_string

Get a meta data string for an object

Description

Get a meta data string for an object

Usage

```
meta_to_string(fameMeta, objnam)
```

Arguments

fameMeta	a list containing fame object meta data
objnam	object name

Value

string containing meta data

Examples

```

dbname <- file.path(Sys.getenv("FAME"),"util","driecon")
famedb <- read_fame(dbname)
meta <- famedb$get_meta('GDP')
cat(meta_to_string(famedb$get('GDP')$meta, objnam))

```

newEntry	<i>Construct a List entry with FAME data and metadata</i>
----------	---

Description

Construct a List entry with FAME data and metadata

Usage

```
newEntry(data, desc = NULL, docu = NULL, class = rhli::HSERIE,
         range = NULL, type = rhli::HPRECN, basis = rhli::HBSBUS,
         obse = rhli::HOBEND)
```

Arguments

data	data value(s) to store
desc	description
docu	documentation
class	object class HLI code
range	FAME range of object data (if series)
type	object type HLI code
basis	object basis HLI code
obse	object observed attribute

Value

FAME database object (data and metadata as nested R list)

Examples

```
entry <- newEntry('String contents',class=rhli::HSCALA,type=rhli::HSTRNG)
```

open_hli	<i>Open the FAME host language interface.</i>
----------	---

Description

Open the FAME host language interface.

Usage

```
open_hli()
```

print_catalog	<i>Print a catalog</i>
---------------	------------------------

Description

Print a catalog

Usage

```
print_catalog(famedata, list.len = -1)
```

Arguments

famedata	List containing FAME data to display
list.len	maximum entries to display

Examples

```
myStuff <- List()
entry <- newEntry('String contents',class=rhli::HSCALA,type=rhli::HSTRNG)
myStuff$put('str',entry)
print_catalog(myStuff)
```

print_stack	<i>Display version information</i>
-------------	------------------------------------

Description

Display version information

Usage

```
print_stack()
```

read_fame	<i>Read a FAME database into an R list</i>
-----------	--

Description

Read a FAME database into an R list

Usage

```
read_fame(dbname_, wilnam_ = "?", fame_range_ = NULL)
```

Arguments

dbname_	FAME database filename
wilnam_	object name wildcard
fame_range_	FAME range to limit data retrieval

Value

List containing FAME objectname, FAME objectdata pairs

Examples

```
dbname <- file.path(Sys.getenv("FAME"), "util", "driecon")  
famedb <- read_fame(dbname)
```

to_fame_range	<i>Construct a date range</i>
---------------	-------------------------------

Description

Construct a date range

Usage

```
to_fame_range(freq_, start_str_, end_str_)
```

Arguments

freq_	FAME frequency HLI code
start_str_	FAME start date literal (e.g. "2018M12")
end_str_	FAME end date literal

Value

date range

Examples

```
range <- to_fame_range(rhli::HANDEC,"1993","2002")
```

to_lubridate_index *Create a lubridate index*

Description

Create a lubridate index

Usage

```
to_lubridate_index(rng)
```

Arguments

rng FAME range

Value

tibble with lubridate date column

Examples

```
rng <- to_fame_range(rhli::HANDEC,"1993","2002")  
tbl <- to_lubridate_index(rng)
```

write_fame *Write FAME db*

Description

Write FAME db

Usage

```
write_fame(dbname_, container)
```

Arguments

dbname_	FAME database filename
container	List with data to write

Examples

```
mydb <- List()
# construct an entry for FAME scalar string
entry <- newEntry('String contents',class=rhli::HSCALA,type=rhli::HSTRNG)
mydb$put('str',entry)
dbfile <- file.path(tempdir(),'tmp.db')
write_fame(dbfile,mydb)
```


Index

`close_hli`, [2](#)

`List` (`List-class`), [2](#)

`List-class`, [2](#)

`meta_to_string`, [3](#)

`newEntry`, [4](#)

`open_hli`, [4](#)

`print_catalog`, [5](#)

`print_stack`, [5](#)

`read_fame`, [6](#)

`to_fame_range`, [6](#)

`to_lubridate_index`, [7](#)

`write_fame`, [7](#)