The « antaresRead » package

Read in R data from an Antares study

Cheat sheet by François Guillem
francois.guillem@rte-france.com

0. Installation

```
# Install the dependencies
install.packages(c("plyr", "data.table",
                   "lubridate"))
# Install the package
install.packages("path/antaresReadXXX.zip",
                 repos = NULL,
                 type = "win.binary")
```

1. Initialization

```
library(antaresRead)
```

Select a simulation :

```
# Façon interactive
setSimulationPath()
# Façon programmatique
setsimulationPath("chemin_etude", simulation)
```

Parameter « `simulation` » accepts the following values:

- Name of the simulation
- Index of the simulation :
  - 1 : oldest simulation
  - -1 : latest simulation
  - …
- 0 or « input » if one only wants to read input data of an Antares study

2. Read the results

```
readAntares(...)
```

All parameters are optional. They control what data the function reads:

Elements to retrieve:

- `areas` = names of areas to retrieve
- `links` = names of links to retrieve
- `clusters` = names of areas with clusters to retrieve
- `districts` = name of sets of areas to retrieve

Add additional data:

- `misc = TRUE` to add miscellaneous productions
- `thermalAvailabilities = TRUE` to add the available capacity of thermal clusters
- `hydroStorage = TRUE` to add the expected hydraulic storage power in the month
- `hydroStorageMaxPower = TRUE` to add the maximal capacity of hydraulic storage production
- `reserve = TRUE` to add reserves
- `linkCapacity = TRUE` to add technical features of the links
- `mustRun = TRUE` to add productions in partial and complete must run mode

Scenarios Monte-Carlo:

- `synthesis = FALSE` to get results of the Monte-Carlo scenarios instead of the synthetic results
- `mcYears = ...` to get only some scenarios

Others:

- `select = ...` to select only some columns of the results
- `timeStep = ...` to change the time step (by default hourly time step)

Fonctions that help filling the parameters

```
getAreas(...)
getLinks(...)
getAreas(..., withClustersOnly = TRUE)
getDistricts(...)
showAliases()
```

3. Use the results

readAntares returns a data.table or a list of data.tables if different types of objects are retrieved.

- Filtrer data:

```
mydata[area == "fr" & month == "JUL"]
```

- Select and compute columns:

```
mydata[, .(area, month, load2 = LOAD^2)]
```

- Compute aggregated statistics:

```
mydata[, .(total = sum(LOAD)), by = .(month)]
```

- Filter then aggregate data:

```
mydata[area == "fr",
       .(total = sum(LOAD)),
       by = .(month)]
```

More info about data.table:

```
help(package = "data.table")
```

Other functions

- `readInputTs`: read input time series of an Antares study
- `readLayout`: read the coordinates and colors of the nodes in the graphical user interface of Antares
- `copyToClipboard`: copy a table in the clipboard in order to paste it in Excel
- `readClusterDesc`: read the characteristics of the thermal clusters (capacity, number of units, …)
- `viewAntares`: display in the data-viewer an object created by readAntares or readInputTS
- `readBindingConstraints`: read binding constraints
- `simOptions`: get the default options or the options used to create an object