Package 'ROpenFLUID'

November 15, 2012

Type Package

Title Package for using OpenFLUID within the GNU R environment

Version 1.7.2

Date 2012-11-15

Author Jean-Christophe Fabre <fabrejc@supagro.inra.fr>

Maintainer Jean-Christophe Fabre <fabrejc@supagro.inra.fr>

Description This package allows to load, parameterize, run and analyse OpenFLUID simulations within the GNU R environment

URL http://www.umr-lisah.fr/openfluid

License GPL-3 + file LICENSE

Depends R (>= 2.6.2), RUnit

LazyLoad yes

R topics documented:

`OpenFLUID.addExtraFunctionsPaths`
Adds paths to search for simulation functions

Description

Adds paths to search for simulation functions

Usage

```
OpenFLUID.addExtraFunctionsPaths(paths)
```

Arguments

`paths`	the semicolon separated paths to add

See Also

`OpenFLUID.getExtraFunctionsPaths`

Examples

```
## Not run:
OpenFLUID.addExtraFunctionsPaths("/first/path/to/add")
OpenFLUID.addExtraFunctionsPaths("/second/path/to/add:/third/path/to/add")

## End(Not run)
```

OpenFLUID.createInputData

Creates an inputdata for alla spatial units of a class, initialized with a default value

Description

Creates an inputdata for alla spatial units of a class, initialized with a default value

Usage

```
OpenFLUID.createInputData(ofblob, unitclass, idataname,
  idataval)
```

Arguments

ofblob	the simulation definition blob
unitclass	the unit class
idataname	the inpudata name
idataval	the default inpudata value for alla units

See Also

OpenFLUID.getInputData

OpenFLUID.setInputData

Examples

```
## Not run:
OpenFLUID.createInputData(ofsim,"SU","area",1.0)
OpenFLUID.createInputData(ofsim,"SU","code","NONE")

## End(Not run)
```

OpenFLUID.getDeltaT

Returns the simulation time step

Description

Returns the simulation time step

Usage

```
OpenFLUID.getDeltaT(ofblob)
```

Arguments

`ofblob`	the simulation definition blob

Value

the time step value in seconds

See Also

`OpenFLUID.setDeltaT`

Examples

```
## Not run:
deltat = OpenFLUID.getDeltaT(ofsim)

## End(Not run)
```

`OpenFLUID.getExtraFunctionsPaths` *Returns the added paths to search for simulation functions*

Description

Returns the added paths to search for simulation functions

Usage

```
OpenFLUID.getExtraFunctionsPaths()
```

Value

a vector of paths

See Also

`OpenFLUID.addExtraFunctionsPaths`

Examples

```
## Not run:
paths = OpenFLUID.getExtraFunctionsPaths()

## End(Not run)
```

`OpenFLUID.getFunctionParam`
Returns a function parameter value

Description

Returns a function parameter value

Usage

```
OpenFLUID.getFunctionParam(ofblob, funcid, paramname)
```

Arguments

`ofblob`	the simulation definition blob
`funcid`	the simulation function id
`paramname`	the name of the parameter

Value

the parameter value

See Also

`OpenFLUID.setFunctionParam`

Examples

```
## Not run:
val = OpenFLUID.getFunctionParam(ofsim,"my.function","coeff")

## End(Not run)
```

`OpenFLUID.getGeneratorParam`
Returns a generator parameter value

Description

Returns a generator parameter value

Usage

```
OpenFLUID.getGeneratorParam(ofblob, unitclass, varname,
  paramname)
```

Arguments

ofblob	the simulation definition blob
unitclass	the unit class to which the generator is applied
varname	the variable name to which the generator is applied
paramname	the name of the parameter

Value

the parameter value

See Also

OpenFLUID.setGeneratorParam

Examples

```
## Not run:
val = OpenFLUID.getGeneratorParam(ofsim,"SU","var.flux","fixedvalue")

## End(Not run)
```

OpenFLUID.getInputData

Returns an inputdata value for a given spatial unit

Description

Returns an inputdata value for a given spatial unit

Usage

```
OpenFLUID.getInputData(ofblob, unitclass, unitid,
  idataname)
```

Arguments

ofblob	the simulation definition blob
unitclass	the unit class
unitid	the unit ID
idataname	the name of the inputdata

Value

the inputdata value

See Also

`OpenFLUID.createInputData`

`OpenFLUID.setInputData`

Examples

```
## Not run:
val = OpenFLUID.getInputData(ofsim,"SU",18,"length")

## End(Not run)
```

`OpenFLUID.getModelGlobalParam`

Returns a model global parameter value

Description

Returns a model global parameter value

Usage

```
OpenFLUID.getModelGlobalParam(ofblob, paramname)
```

Arguments

`ofblob`	the simulation definition blob
`paramname`	the name of the parameter

Value

the parameter value

See Also

`OpenFLUID.setModelGlobalParam`

Examples

```
## Not run:
val = OpenFLUID.getModelGlobalParam(ofsim,"gvalue")

## End(Not run)
```

`OpenFLUID.getPeriodBeginDate`

Returns the simulation period begin date

Description

Returns the simulation period begin date

Usage

```
OpenFLUID.getPeriodBeginDate(ofblob)
```

Arguments

`ofblob`	the simulation definition blob

Value

the begin date as an ISO datetime string (%Y-%m-%d %H:%M:%S)

See Also

`OpenFLUID.setPeriodBeginDate`

`OpenFLUID.getPeriodEndDate`

`OpenFLUID.setPeriodEndDate`

Examples

```
## Not run:
bdate = OpenFLUID.getPeriodBeginDate(ofsim)

## End(Not run)
```

`OpenFLUID.getPeriodEndDate`

Returns the simulation period end date

Description

Returns the simulation period end date

Usage

```
OpenFLUID.getPeriodEndDate(ofblob)
```

Arguments

ofblob	the simulation definition blob

Value

the end date as an ISO datetime string (%Y-%m-%d %H:%M:%S)

See Also

OpenFLUID.setPeriodEndDate

OpenFLUID.getPeriodBeginDate

OpenFLUID.setPeriodBeginDate

Examples

```
## Not run:
edate = OpenFLUID.getPeriodEndDate(ofsim)

## End(Not run)
```

OpenFLUID.getUnitsClasses

Returns the existing units classes

Description

Returns the existing units classes

Usage

```
OpenFLUID.getUnitsClasses(ofblob)
```

Arguments

ofblob	the simulation definition blob

Value

a vector of units classes

See Also

OpenFLUID.getUnitsIDs

Examples

```
## Not run:
cls = OpenFLUID.getUnitsClasses(ofsim)

## End(Not run)
```

`OpenFLUID.getUnitsIDs`
Returns the existing units IDs for a given units class

Description

Returns the existing units IDs for a given units class

Usage

```
OpenFLUID.getUnitsIDs(ofblob, unitclass)
```

Arguments

`ofblob`	the simulation definition blob
`unitclass`	the units class

Value

a vector of units IDs

See Also

`OpenFLUID.getUnitsClasses`

Examples

```
## Not run:
ids = OpenFLUID.getUnitsIDs(ofsim,"SU")

## End(Not run)
```

`OpenFLUID.getVersion`

Returns the OpenFLUID version

Description

Returns the OpenFLUID version

Usage

```
OpenFLUID.getVersion()
```

Value

the OpenFLUID version number

Examples

```
## Not run:
v = OpenFLUID.getVersion()

## End(Not run)
```

`OpenFLUID.loadResult`

Loads results as a dataframe, giving output dataset informations

Description

Loads results as a dataframe, giving output dataset informations

Usage

```
OpenFLUID.loadResult(ofblob, unitclass, unitid, suffix)
```

Arguments

`ofblob`	the simulation definition blob
`unitclass`	the unit class
`unitid`	the unit ID
`suffix`	the output dataset suffix

Value

a dataframe containing the simulation results

See Also

`OpenFLUID.loadResultFile`

Examples

```
## Not run:
resSU18 = OpenFLUID.loadResult(ofsim,"SU",18,"full")
resRS1 = OpenFLUID.loadResult(ofsim,"RS",1,"waterlevel")

## End(Not run)
```

OpenFLUID.loadResultFile

Loads results as a dataframe, giving output file name

Description

Loads results as a dataframe, giving output file name

Usage

```
OpenFLUID.loadResultFile(filepath)
```

Arguments

`filepath`	the full path of file to load

Value

a dataframe containing the simulation results

See Also

`OpenFLUID.loadResult`

Examples

```
## Not run:
resSU18 = OpenFLUID.loadResultFile("/path/to/output/SU18_full.out")
resRS1 = OpenFLUID.loadResultFile("/path/to/output/RS1_waterlevel.out")

## End(Not run)
```

OpenFLUID.openDataset

Opens a dataset and returns a simulation definition blob

Description

Opens a dataset and returns a simulation definition blob

Usage

```
OpenFLUID.openDataset(path)
```

Arguments

path	the full path of the dataset to open

Value

a simulation definition blob

See Also

OpenFLUID.openProject

OpenFLUID.runSimulation

Examples

```
## Not run:
ofsim = OpenFLUID.openDataset("/path/to/dataset")

## End(Not run)
```

OpenFLUID.openProject

Opens a project and returns a simulation definition blob

Description

Opens a project and returns a simulation definition blob

Usage

```
OpenFLUID.openProject(path)
```

Arguments

path	the full project to open

Value

a simulation definition blob

See Also

`OpenFLUID.openDataset`

`OpenFLUID.runProject`

Examples

```
## Not run:
ofsim = OpenFLUID.openProject("/path/to/project")

## End(Not run)
```

`OpenFLUID.printSimulationInfo`

Prints informations to screen about simulation definition blob

---

Description

Prints informations to screen about simulation definition blob

Usage

```
OpenFLUID.printSimulationInfo(ofblob)
```

Arguments

`ofblob`	the simulation definition blob

Examples

```
## Not run:
OpenFLUID.printSimulationInfo(ofsim)

## End(Not run)
```

`OpenFLUID.runProject`
Runs a project and returns a simulation definition blob

Description

Runs a project and returns a simulation definition blob

Usage

```
OpenFLUID.runProject(path)
```

Arguments

`path`	the full path of the dataset to open

Value

a simulation definition blob

See Also

`OpenFLUID.runSimulation`
`OpenFLUID.openProject`

Examples

```
## Not run:
ofsim = OpenFLUID.runProject("/path/to/dataset")

## End(Not run)
```

`OpenFLUID.runSimulation`
Runs a simulation from a simulation definition blob

Description

Runs a simulation from a simulation definition blob

Usage

```
OpenFLUID.runSimulation(ofblob)
```

Arguments

`ofblob`	the simulation definition blob

See Also

`OpenFLUID.runProject`

`OpenFLUID.openProject`

`OpenFLUID.openDataset`

Examples

```
## Not run:
OpenFLUID.runSimulation(ofsim)

## End(Not run)
```

`OpenFLUID.setCurrentOutputDir`
Sets the current output directory for simulations

Description

Sets the current output directory for simulations

Usage

```
OpenFLUID.setCurrentOutputDir(path)
```

Arguments

`path`	the output directory path

Examples

```
## Not run:
OpenFLUID.setCurrentOutputDir("/path/to/output")

## End(Not run)
```

OpenFLUID.setDeltaT

Sets the simulation time step

Description

Sets the simulation time step

Usage

```
OpenFLUID.setDeltaT(ofblob, deltat)
```

Arguments

ofblob	the simulation definition blob
deltat	the time step value in seconds

See Also

OpenFLUID.getDeltaT

Examples

```
## Not run:
OpenFLUID.setDeltaT(60)
OpenFLUID.setDeltaT(86400)

## End(Not run)
```

OpenFLUID.setFunctionParam

Sets a function parameter value

Description

Sets a function parameter value

Usage

```
OpenFLUID.setFunctionParam(ofblob, funcid, paramname,
  paramval)
```

Arguments

ofblob	the simulation definition blob
funcid	the simulation function id
paramname	the name of the parameter
paramval	the parameter value

See Also

OpenFLUID.getFunctionParam

Examples

```
## Not run:
OpenFLUID.setFunctionParam(ofsim,"my.function","coeff",3)

## End(Not run)
```

OpenFLUID.setGeneratorParam

Sets a generator parameter value

Description

Sets a generator parameter value

Usage

```
OpenFLUID.setGeneratorParam(ofblob, unitclass, varname,
  paramname, paramval)
```

Arguments

ofblob	the simulation definition blob
unitclass	the unit class to which the generator is applied
varname	the variable name to which the generator is applied
paramname	the name of the parameter
paramval	the value of the parameter

See Also

OpenFLUID.getGeneratorParam

Examples

```
## Not run:
OpenFLUID.setGeneratorParam(ofsim,"SU","var.flux","fixedvalue",12.3)

## End(Not run)
```

`OpenFLUID.setInputData`

Sets an inputdata value for a given spatial unit

Description

Sets an inputdata value for a given spatial unit

Usage

```
OpenFLUID.setInputData(ofblob, unitclass, unitid,
  idataname, idataval)
```

Arguments

`ofblob`	the simulation definition blob
`unitclass`	the unit class
`unitid`	the unit ID
`idataname`	the name of the inputdata
`idataval`	the value of the inputdata

See Also

`OpenFLUID.createInputData`

`OpenFLUID.getInputData`

Examples

```
## Not run:
OpenFLUID.setInputData(ofsim,"SU",18,"length",12.3)
OpenFLUID.setInputData(ofsim,"SU",18,"CODE","ABC")

## End(Not run)
```

`OpenFLUID.setModelGlobalParam`

Sets a model global parameter value

Description

Sets a model global parameter value

Usage

```
OpenFLUID.setModelGlobalParam(ofblob, paramname,
  paramval)
```

Arguments

ofblob	the simulation definition blob
paramname	the name of the parameter
paramval	the value of the parameter

See Also

OpenFLUID.getModelGlobalParam

Examples

```
## Not run:
OpenFLUID.setModelGlobalParam(ofsim,"gvalue",37.2)

## End(Not run)
```

OpenFLUID.setPeriodBeginDate

Sets the simulation period begin date

Description

Sets the simulation period begin date

Usage

```
OpenFLUID.setPeriodBeginDate(ofblob, begindate)
```

Arguments

ofblob	the simulation definition blob
begindate	the begin date as an ISO datetime string (%Y-%m-%d %H:%M:%S)

See Also

OpenFLUID.getPeriodBeginDate

OpenFLUID.setPeriodEndDate

OpenFLUID.getPeriodEndDate

Examples

```
## Not run:
OpenFLUID.setPeriodBeginDate(ofsim,"1997-06-05 04:00:00")

## End(Not run)
```

`OpenFLUID.setPeriodEndDate` *Sets the simulation period end date*

Description

Sets the simulation period end date

Usage

```
OpenFLUID.setPeriodEndDate(ofblob, enddate)
```

Arguments

`ofblob`	the simulation definition blob
`enddate`	the end date as an ISO datetime string (%Y-%m-%d %H:%M:%S)

See Also

`OpenFLUID.getPeriodEndDate`

`OpenFLUID.setPeriodBeginDate`

`OpenFLUID.getPeriodBeginDate`

Examples

```
## Not run:
OpenFLUID.setPeriodEndDate(ofsim,"1997-06-05 16:07:17")

## End(Not run)
```

`ROpenFLUID` *Package for using OpenFLUID within the GNU R environment*

Description

This package allows to load, parameterize, run and analyse OpenFLUID simulations within the GNU R environment

Details

Package:	ROpenFLUID
Type:	Package
Version:	1.7.2
Date:	2012-11-15
License:	GPLv3 with special exception
LazyLoad:	yes

Author(s)

Jean-Christophe Fabre <fabrejc@supagro.inra.fr>

Examples

```
## Not run:
# load OpenFLUID library
library("ROpenFLUID")

# add optional paths to search for simulation functions
OpenFLUID.addExtraFunctionsPaths("/path/to/simfuncs")

# open an input dataset
ofsim = OpenFLUID.openDataset("/path/to/dataset")

# set the output dir
OpenFLUID.setCurrentOutputDir("/path/to/output")

# run the simulation
OpenFLUID.runSimulation(ofsim)

## End(Not run)
```