

# Package ‘ROpenFLUID’

May 14, 2014

**Type** Package

**Title** Package for using OpenFLUID within the GNU R environment

**Version** 2.0.0-4

**Date** 2014-05-14

**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 analyze OpenFLUID simulations within the GNU R environment

**URL** <http://www.openfluid-project.org>

**License** GPL-3 + file LICENSE

**Depends** R (>= 2.6.2), RUnit

**LazyLoad** yes

## R topics documented:

OpenFLUID.addExtraObserversPaths . . . . .	1
OpenFLUID.addExtraSimulatorsPaths . . . . .	2
OpenFLUID.addVariablesExportAsCSV . . . . .	3
OpenFLUID.createAttribute . . . . .	3
OpenFLUID.getAttribute . . . . .	4
OpenFLUID.getDefaultDeltaT . . . . .	5
OpenFLUID.getExtraObserversPaths . . . . .	6
OpenFLUID.getExtraSimulatorsPaths . . . . .	6
OpenFLUID.getGeneratorParam . . . . .	7
OpenFLUID.getModelGlobalParam . . . . .	8
OpenFLUID.getObserverParam . . . . .	8
OpenFLUID.getObserversPaths . . . . .	9
OpenFLUID.getPeriodBeginDate . . . . .	10
OpenFLUID.getPeriodEndDate . . . . .	11

OpenFLUID.getSimulatorParam . . . . .	11
OpenFLUID.getSimulatorsPaths . . . . .	12
OpenFLUID.getUnitsClasses . . . . .	13
OpenFLUID.getUnitsIDs . . . . .	14
OpenFLUID.getVersion . . . . .	14
OpenFLUID.loadResult . . . . .	15
OpenFLUID.loadResultFile . . . . .	16
OpenFLUID.openDataset . . . . .	16
OpenFLUID.openProject . . . . .	17
OpenFLUID.printSimulationInfo . . . . .	18
OpenFLUID.removeAttribute . . . . .	18
OpenFLUID.removeModelGlobalParam . . . . .	19
OpenFLUID.removeObserverParam . . . . .	20
OpenFLUID.removeSimulatorParam . . . . .	20
OpenFLUID.resetExtraObserversPaths . . . . .	21
OpenFLUID.resetExtraSimulatorsPaths . . . . .	22
OpenFLUID.runProject . . . . .	22
OpenFLUID.runSimulation . . . . .	23
OpenFLUID.setAttribute . . . . .	24
OpenFLUID.setCurrentOutputDir . . . . .	25
OpenFLUID.setDefaultDeltaT . . . . .	25
OpenFLUID.setGeneratorParam . . . . .	26
OpenFLUID.setModelGlobalParam . . . . .	27
OpenFLUID.setObserverParam . . . . .	27
OpenFLUID.setPeriodBeginDate . . . . .	28
OpenFLUID.setPeriodEndDate . . . . .	29
OpenFLUID.setSimulatorParam . . . . .	29
ROpenFLUID . . . . .	30

---

OpenFLUID.addExtraObserversPaths

*Adds paths to search for observers*

---

### **Description**

Adds paths to search for observers

### **Usage**

OpenFLUID.addExtraObserversPaths (paths)

### **Arguments**

paths            the colon separated paths to add

**See Also**

```
OPENFLUID.getObserversPaths  
OPENFLUID.getExtraObserversPaths  
OPENFLUID.resetExtraObserversPaths
```

**Examples**

```
## Not run:  
OPENFLUID.addExtraObserversPaths("/first/path/to/add")  
OPENFLUID.addExtraObserversPaths("/second/path/to/add:/third/path/to/add")  
  
## End(Not run)
```

---

```
OpenFLUID.addExtraSimulatorsPaths  
Adds paths to search for simulators
```

---

**Description**

Adds paths to search for simulators

**Usage**

```
OpenFLUID.addExtraSimulatorsPaths(paths)
```

**Arguments**

paths            the colon separated paths to add

**See Also**

```
OpenFLUID.getSimulatorsPaths  
OpenFLUID.getExtraSimulatorsPaths  
OPENFLUID.resetExtraSimulatorsPaths
```

**Examples**

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

```
OpenFLUID.addVariablesExportAsCSV
```

*Adds export of simulation variables as CSV files for a given units class*

---

### Description

Adds export of simulation variables as CSV files for a given units class

### Usage

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

### Arguments

<code>ofblob</code>	the simulation definition blob
<code>unitclass</code>	the units class to add for simulation variables export

### See Also

```
OpenFLUID.loadResult
```

### Examples

```
## Not run:  
OpenFLUID.addVariablesExportAsCSV("TU")  
OpenFLUID.addVariablesExportAsCSV("RS")  
  
## End(Not run)
```

---

```
OpenFLUID.createAttribute
```

*Creates an attribute for all spatial units of a class, initialized with a default value*

---

### Description

Creates an attribute for all spatial units of a class, initialized with a default value

### Usage

```
OpenFLUID.createAttribute(ofblob, unitclass, attrname,  
attrval)
```

**Arguments**

ofblob	the simulation definition blob
unitclass	the unit class
attrname	the attribute name
attrval	the default attribute value for alla units

**See Also**

OpenFLUID.getAttribute  
 OpenFLUID.setAttribute  
 OpenFLUID.removeAttribute

**Examples**

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

## End(Not run)
```

---

OpenFLUID.getAttribute

*Returns an attribute value for a given spatial unit*

---

**Description**

Returns an attribute value for a given spatial unit

**Usage**

```
OpenFLUID.getAttribute(ofblob, unitclass, unitid,
  attrname)
```

**Arguments**

ofblob	the simulation definition blob
unitclass	the unit class
unitid	the unit ID
attrname	the name of the attribute

**Value**

the attribute value

**See Also**

```
OpenFLUID.createAttribute  
OpenFLUID.setAttribute  
OpenFLUID.removeAttribute
```

**Examples**

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

---

```
OpenFLUID.getDefaultDeltaT  
Returns the simulation time step
```

---

**Description**

Returns the simulation time step

**Usage**

```
OpenFLUID.getDefaultDeltaT(ofblob)
```

**Arguments**

ofblob            the simulation definition blob

**Value**

the time step value in seconds

**See Also**

```
OpenFLUID.setDefaultDeltaT
```

**Examples**

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

---

```
OpenFLUID.getExtraObserversPaths
```

*Returns the added paths to search for observers*

---

**Description**

Returns the added paths to search for observers

**Usage**

```
OpenFLUID.getExtraObserversPaths ()
```

**Value**

a vector of paths

**See Also**

```
OpenFLUID.addExtraObserversPaths  
OpenFLUID.getObserversPaths  
OPENFLUID.resetExtraObserversPaths
```

**Examples**

```
## Not run:  
paths = OpenFLUID.getExtraObserversPaths ()  
  
## End (Not run)
```

---

```
OpenFLUID.getExtraSimulatorsPaths
```

*Returns the added paths to search for simulators*

---

**Description**

Returns the added paths to search for simulators

**Usage**

```
OpenFLUID.getExtraSimulatorsPaths ()
```

**Value**

a vector of paths

**See Also**

```
OpenFLUID.addExtraSimulatorsPaths
OpenFLUID.getSimulatorsPaths
OPENFLUID.resetExtraSimulatorsPaths
```

**Examples**

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

## 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.getModelGlobalParam
```

*Returns a model global parameter value*

---

### Description

Returns a model global parameter value

### Usage

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

### Arguments

<code>ofblob</code>	the simulation definition blob
<code>paramname</code>	the name of the parameter

### Value

the parameter value

### See Also

```
OpenFLUID.setModelGlobalParam  
OpenFLUID.removeModelGlobalParam
```

### Examples

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

---

```
OpenFLUID.getObserverParam
```

*Returns an observer parameter value*

---

### Description

Returns an observer parameter value

### Usage

```
OpenFLUID.getObserverParam(ofblob, obsid, paramname)
```

**Arguments**

ofblob	the simulation definition blob
obsid	the observer ID
paramname	the name of the parameter

**Value**

the parameter value

**See Also**

OpenFLUID.setObserverParam  
OpenFLUID.removeObserverParam

**Examples**

```
## Not run:  
val = OpenFLUID.getObserverParam(ofsim, "my.observer", "value")  
  
## End(Not run)
```

---

```
OpenFLUID.getObserversPaths
```

*Returns the paths to search for observers*

---

**Description**

Returns the paths to search for observers

**Usage**

```
OpenFLUID.getObserversPaths()
```

**Value**

a vector of paths

**See Also**

OpenFLUID.addExtraObserversPaths  
OpenFLUID.getExtraObserversPaths  
OPENFLUID.resetExtraObserversPaths

**Examples**

```
## Not run:  
paths = OpenFLUID.getObserversPaths()  
  
## 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.getSimulatorParam

*Returns a simulator parameter value*

---

### **Description**

Returns a simulator parameter value

### **Usage**

```
OpenFLUID.getSimulatorParam(ofblob, simid, paramname)
```

**Arguments**

ofblob	the simulation definition blob
simid	the simulator ID
paramname	the name of the parameter

**Value**

the parameter value

**See Also**

OpenFLUID.setSimulatorParam  
OpenFLUID.removeSimulatorParam

**Examples**

```
## Not run:  
val = OpenFLUID.getSimulatorParam(ofsim, "my.simulator", "coeff")  
  
## End(Not run)
```

---

```
OpenFLUID.getSimulatorsPaths
```

*Returns the paths to search for simulators*

---

**Description**

Returns the paths to search for simulators

**Usage**

```
OpenFLUID.getSimulatorsPaths()
```

**Value**

a vector of paths

**See Also**

OpenFLUID.addExtraSimulatorsPaths  
OpenFLUID.getExtraSimulatorsPaths  
OPENFLUID.resetExtraSimulatorsPaths

**Examples**

```
## Not run:  
paths = OpenFLUID.getSimulatorsPaths()  
  
## 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 dataset informations*

---

**Description**

Loads results as a dataframe, giving dataset informations

**Usage**

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

**Arguments**

ofblob	the simulation definition blob
unitclass	the unit class
unitid	the unit ID
varname	the variable name

**Value**

a dataframe containing the simulation results

**See Also**

OpenFLUID.loadResultFile

**Examples**

```
## Not run:  
resSU18 = OpenFLUID.loadResult(ofsim, "SU", 18, "runoff")  
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.removeAttribute
```

*Removes an attribute value for a given spatial unit*

---

**Description**

Removes an attribute value for a given spatial unit

**Usage**

```
OpenFLUID.removeAttribute (ofblob, unitclass, attrname)
```

**Arguments**

ofblob            the simulation definition blob  
unitclass        the unit class  
attrname         the name of the attribute

**See Also**

```
OpenFLUID.createAttribute  
OpenFLUID.getAttribute  
OpenFLUID.setAttribute
```

**Examples**

```
## Not run:  
OpenFLUID.removeAttribute(ofsim, "SU", "length")  
  
## End(Not run)
```

---

OpenFLUID.removeModelGlobalParam  
*Removes a model global parameter value*

---

**Description**

Removes a model global parameter value

**Usage**

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

**Arguments**

ofblob	the simulation definition blob
paramname	the name of the parameter

**See Also**

```
OpenFLUID.getModelGlobalParam  
OpenFLUID.setModelGlobalParam
```

**Examples**

```
## Not run:  
OpenFLUID.removeModelGlobalParam(ofsim, "gvalue")  
  
## End(Not run)
```

---

```
OpenFLUID.removeObserverParam
```

*Removes a observer parameter*

---

**Description**

Removes a observer parameter

**Usage**

```
OpenFLUID.removeObserverParam(ofblob, obsid, paramname)
```

**Arguments**

ofblob	the simulation definition blob
obsid	the simulation observer id
paramname	the name of the parameter

**See Also**

```
OpenFLUID.getObserverParam  
OpenFLUID.setObserverParam
```

**Examples**

```
## Not run:  
OpenFLUID.removeObserverParam(ofsim, "my.observer", "value")  
  
## End(Not run)
```

---

```
OpenFLUID.removeSimulatorParam
```

*Removes a simulator parameter*

---

**Description**

Removes a simulator parameter

**Usage**

```
OpenFLUID.removeSimulatorParam(ofblob, simid, paramname)
```

**Arguments**

ofblob	the simulation definition blob
simid	the simulation simulator id
paramname	the name of the parameter

**See Also**

OpenFLUID.getSimulatorParam  
OpenFLUID.setSimulatorParam

**Examples**

```
## Not run:  
OpenFLUID.removeSimulatorParam(ofsim, "my.simulator", "coeff")  
  
## End(Not run)
```

---

```
OpenFLUID.resetExtraObserversPaths
```

*Resets list of added paths to search for observers*

---

**Description**

Resets list of added paths to search for observers

**Usage**

```
OpenFLUID.resetExtraObserversPaths()
```

**See Also**

OPENFLUID.addExtraObserversPaths  
OPENFLUID.getObserversPaths  
OPENFLUID.getExtraObserversPaths

**Examples**

```
## Not run:  
OpenFLUID.resetExtraObserversPaths()  
  
## End(Not run)
```

---

```
OpenFLUID.resetExtraSimulatorsPaths
```

*Resets list of added paths to search for simulators*

---

**Description**

Resets list of added paths to search for simulators

**Usage**

```
OpenFLUID.resetExtraSimulatorsPaths ()
```

**See Also**

```
OPENFLUID.addExtraSimulatorsPaths  
OPENFLUID.getSimulatorsPaths  
OPENFLUID.getExtraSimulatorsPaths
```

**Examples**

```
## Not run:  
OpenFLUID.resetExtraSimulatorsPaths ()  
  
## 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.setAttribute`*Sets an attribute value for a given spatial unit*

---

**Description**

Sets an attribute value for a given spatial unit

**Usage**

```
OpenFLUID.setAttribute(ofblob, unitclass, unitid,  
                        attrname, attrval)
```

**Arguments**

<code>ofblob</code>	the simulation definition blob
<code>unitclass</code>	the unit class
<code>unitid</code>	the unit ID
<code>attrname</code>	the name of the attribute
<code>attrval</code>	the value of the attribute

**See Also**`OpenFLUID.createAttribute``OpenFLUID.getAttribute``OpenFLUID.removeAttribute`**Examples**

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

*Sets the simulation time step*

---

**Description**

Sets the simulation time step

**Usage**

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

**Arguments**

ofblob                  the simulation definition blob

deltat                  the time step value in seconds

**See Also**

```
OpenFLUID.getDefaultDeltaT
```

**Examples**

```
## Not run:  
OpenFLUID.setDefaultDeltaT(60)  
OpenFLUID.setDefaultDeltaT(86400)  
  
## 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.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  
OpenFLUID.removeModelGlobalParam
```

**Examples**

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

---

```
OpenFLUID.setObserverParam
```

*Sets an observer parameter value*

---

**Description**

Sets an observer parameter value

**Usage**

```
OpenFLUID.setObserverParam(ofblob, obsid, paramname,  
                             paramval)
```

**Arguments**

ofblob	the simulation definition blob
obsid	the simulation observer id
paramname	the name of the parameter
paramval	the parameter value

**See Also**

OpenFLUID.getObserverParam  
 OpenFLUID.removeObserverParam

**Examples**

```
## Not run:
OpenFLUID.setObserverParam(ofsim, "my.observer", "value", 3)

## 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)
```

---

```
OpenFLUID.setSimulatorParam
```

*Sets a simulator parameter value*

---

### Description

Sets a simulator parameter value

### Usage

```
OpenFLUID.setSimulatorParam(ofblob, simid, paramname,  
                             paramval)
```

**Arguments**

ofblob	the simulation definition blob
simid	the simulation simulator id
paramname	the name of the parameter
paramval	the parameter value

**See Also**

```
OpenFLUID.getSimulatorParam  
OpenFLUID.removeSimulatorParam
```

**Examples**

```
## Not run:  
OpenFLUID.setSimulatorParam(ofsim, "my.simulator", "coeff", 3)  
  
## End(Not run)
```

---

ROpenFLUID

*Package for using OpenFLUID within the GNU R environment*

---

**Description**

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

**Details**

Package:	ROpenFLUID
Type:	Package
Version:	
Date:	
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 simulators
OpenFLUID.addExtraSimulatorsPaths("/path/to/simulators")

# 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)
```