A free and open-source

OpenMOLE is developed by the Complex Systems community. It relies on the 4 pilars of the open-source movement, meaning that the code can freely be executed, studied, modified and redistributed. This choice is part of the open science trend that we defend and which promotes the sharing and the reusabilty of code in scientific publications.

As it is usually the case in the open-source movement, OpenMOLE invites anyone to contribute to the project by:

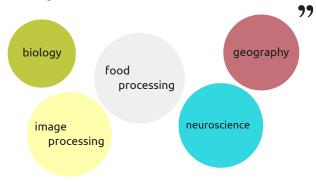
- proposing helpful enhancements or use cases (shared with the community on the website),
- proposing development features,
- reporting bugs,
- developing plugins.

Access to

- the source code and the bug tracker (www.openmole.org)
- the user mailing list: users@list.openmole.org
- the development mailing list: devs@list.openmole.org

Property of the contract of th

Already used in various scientific fields



Short term release cycle (3 months)

Information, downloads, documentation: www.openmole.org





Project supported by

• the "Institut des Systèmes Complexes - Paris Île-de-France"

• the ERC project Geodivercity

• BioEmergences



***** îledeFrance











99



Open ModeL Experiment



Romain Reuillon et al, OpenMOLE, a workflow engine specifically tailored for the distributed exploration of simulation models, Future Generation Computer Systems, vol 29, num 8, pp. 1981-1990, 2013.



Open MOdeL Experiment

A generic workflow platform for distributed experiments on complex-system models

__arge scale computing

- Compute transparently on multi-core machines, clusters, and grids: no particular knowledge is required to access massively parallel environments
- Embed your own programs in less than 5 clicks (C, C++, Java, Scala, Scilab, R, Python, Netlogo, ...)
- Design your distributed process using a powerful workflow approach (loops, alternative processing, massive parallelism)
- Prototype your workflow locally and take advantage of the parallelism with no extra effort: the Zero-deployment approach does not require any installation step on remote machines
- Use OpenMOLE as a scalable computing engine for your web portal (ex: bioemergences.eu is powered by OpenMOLE)



- Embed your model
- Design your model exploration:
 - > design of experiments,
- > stochastic model replication,
- > calibration,
- > sensitivity analysis,
- > ...
- Distribute the numerous executions of your model to explore it:
- > on your multi-core computer // speed up around 4
- > on a multi-core server // speed up around 50
- > on a cluster // speed up arround 200
- > on a grid // speed up > 1000
- Reuse your experiments on several models





Subscribe to the newsletter : users@list.openmole.org

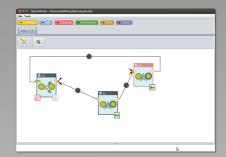


OpenMOLE: for whom?

• Are you a computer scientist?
Script your workflow!

```
val i1 = Prototype[Int]("i1")
val i2 = Prototype[Int]("i2")
val j = Prototype[Int]("i2")
val hello = GroovyTask("hello", "j = Model.compute(i1, i2)")
hello addInput i1
hello addInput i2
hello addUnput j
hello addUnput i2
hello addUnput j
hello addUnput i3
hello addUnput i4
hello addUnput i4
hello addUnput i5
hello addUnput i5
hello addUnput i6
hello by 10 on blomed) toExecution
ex.start
```

Do you make typos, or do you prefer graphical applications?
Use the click and run Application!



•Do you need large scale computing facilities for your web site Use the OpenMOLE web-server!

