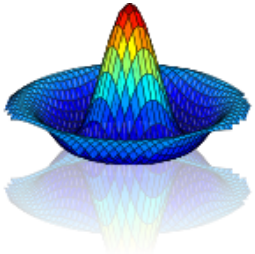
SimExplorer / OpenMOLE

Et l'étude de vos modèles de calcul ne connaîtra plus de limites

Nicolas Dumoulin

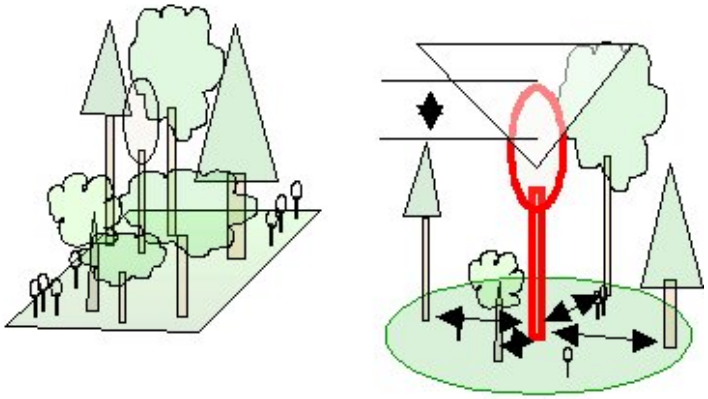
Cemagref — LISC, Aubière, France

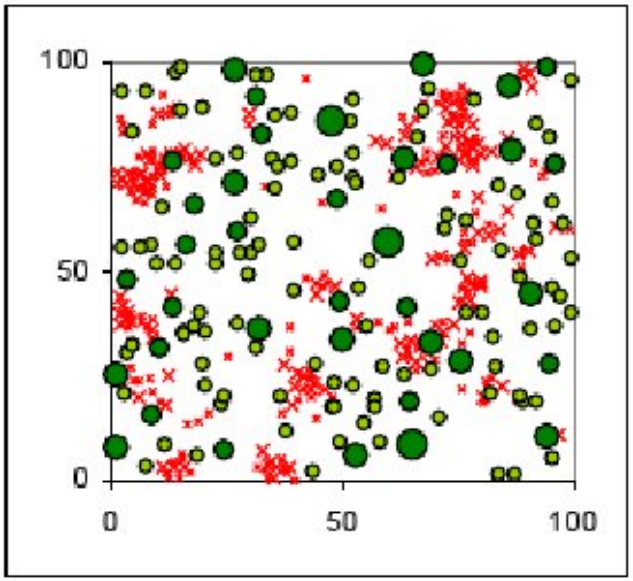
Giens, le 8 Juin 2010



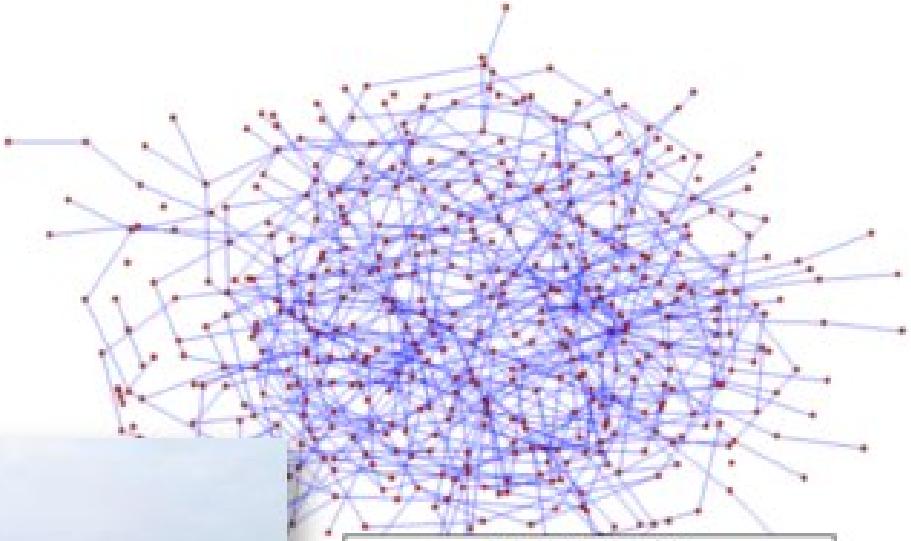
MEXICO

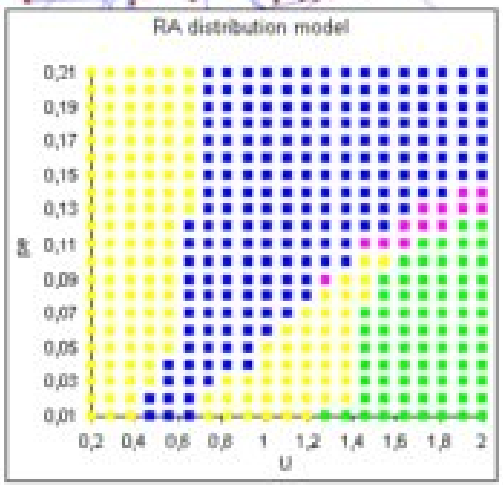
Un système complexe à modéliser ...



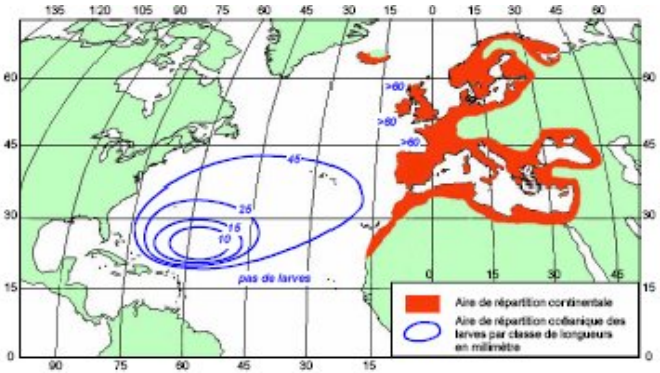


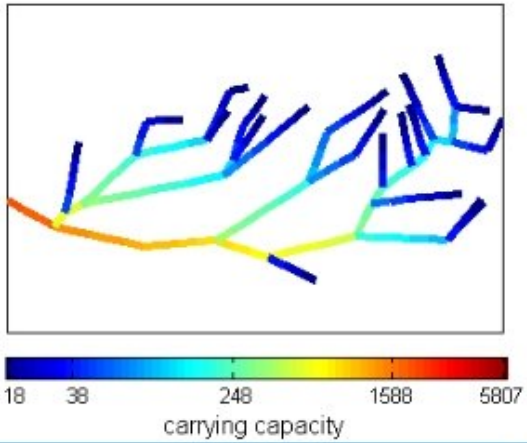
Un système complexe à modéliser ...

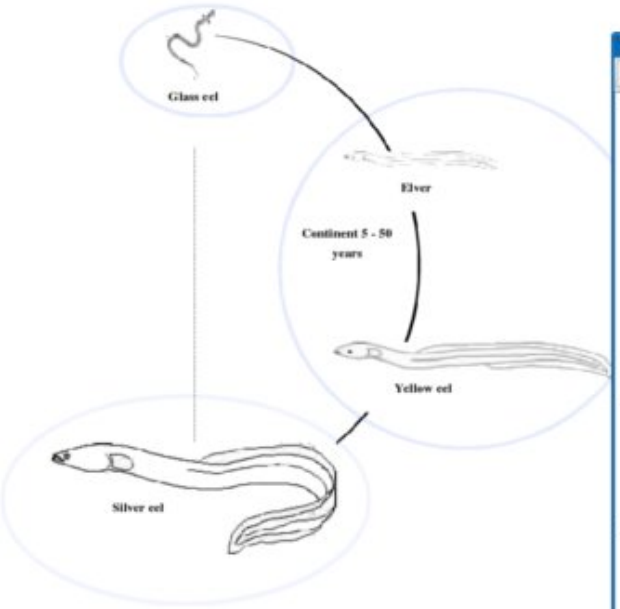


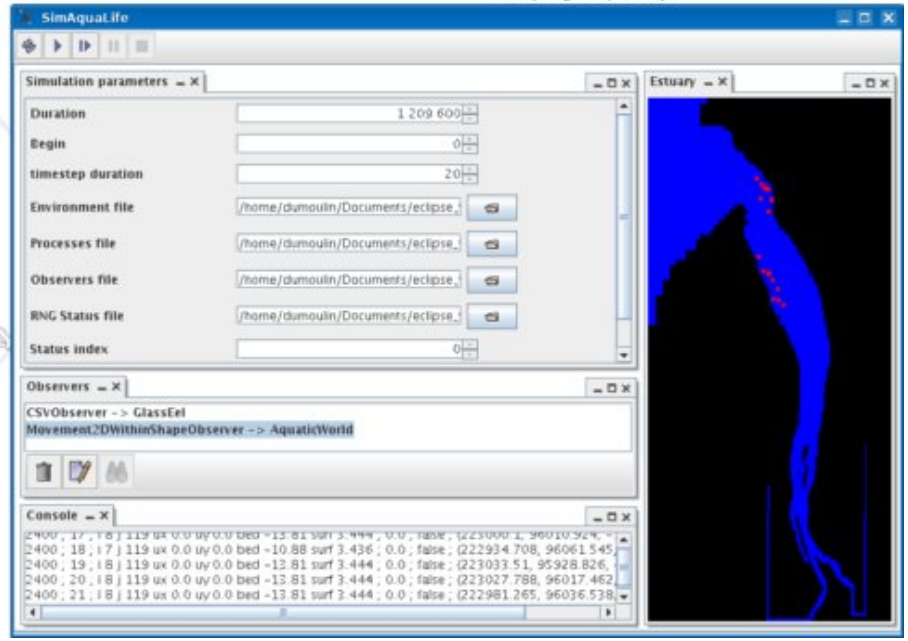


Un système complexe à modéliser ...

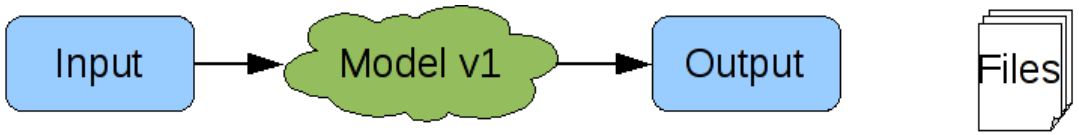






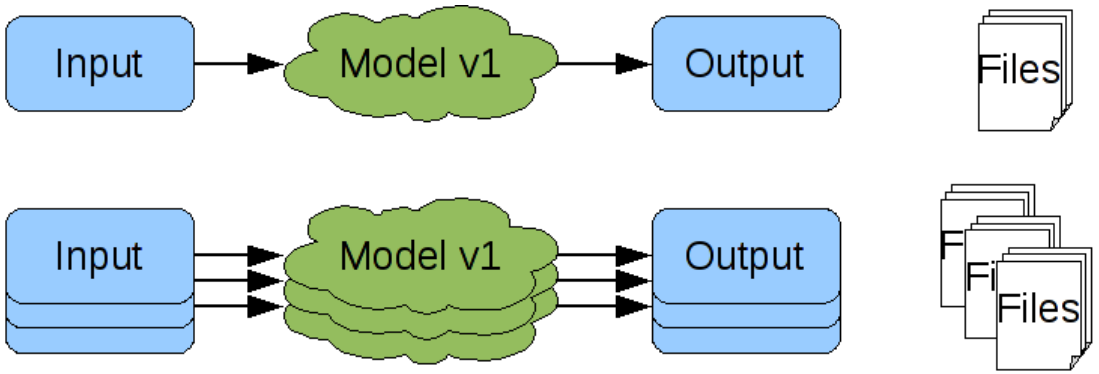


On improvise ?



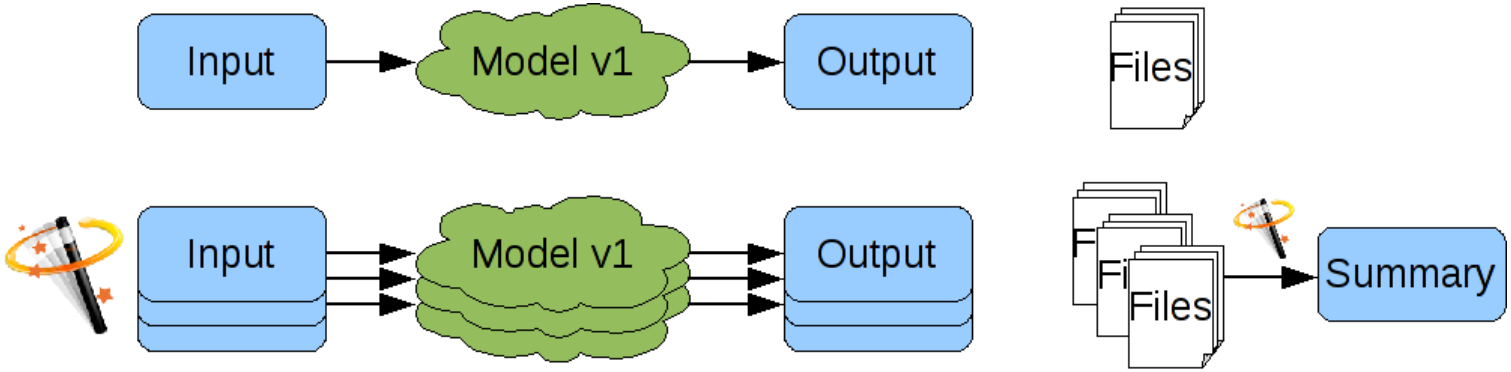
The simulation needs input and gives output

On improvise ?



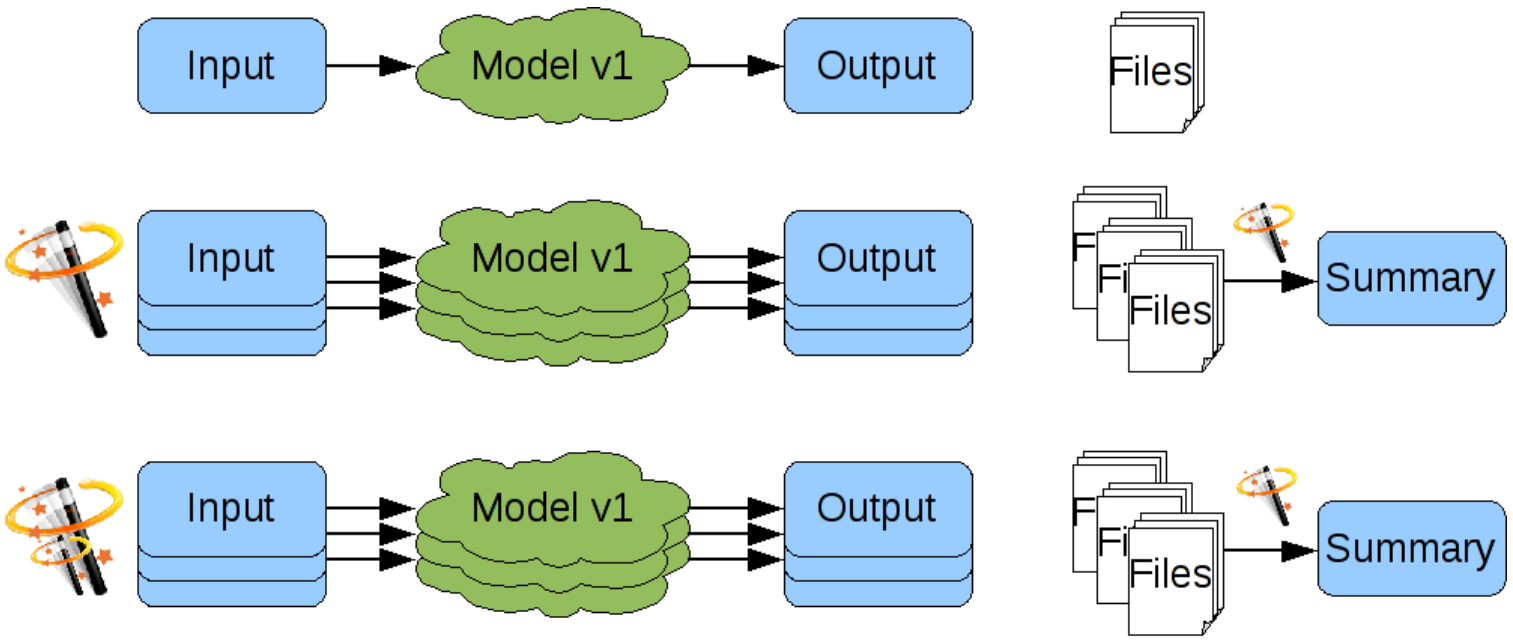
To explore the model, we need to launch several simulations

On improvise ?



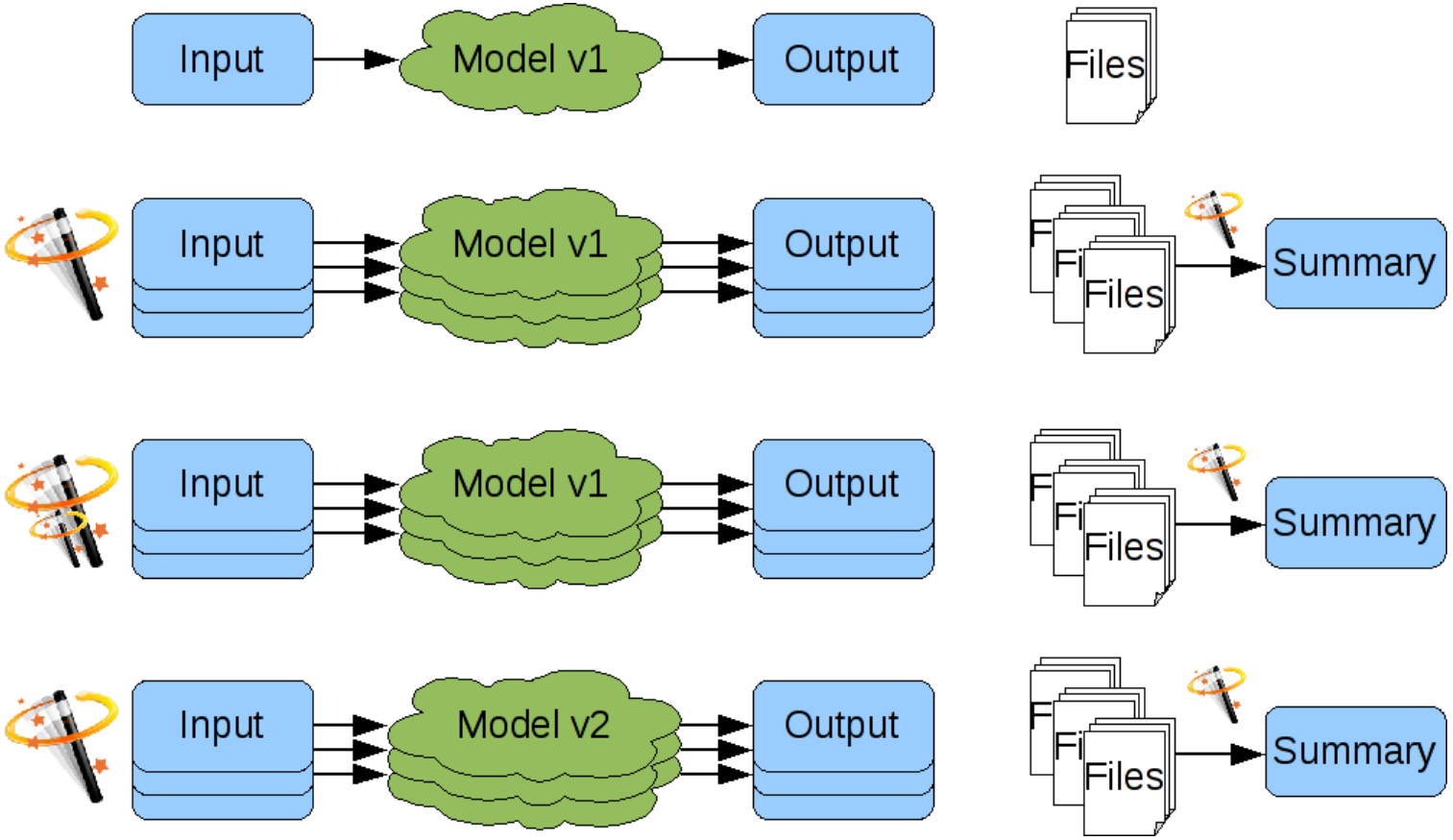
Design of experiments literature helps to build scenarii

On improvise ?



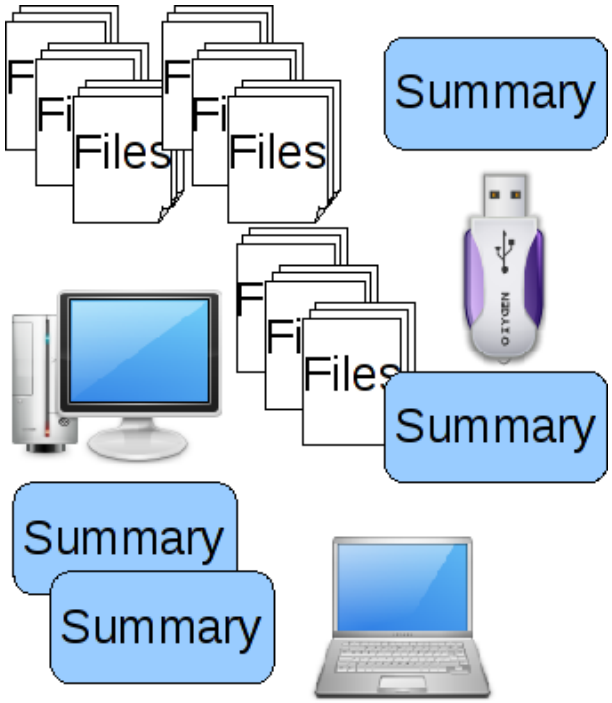
An exploration may bring to design a new exploration (more accurate)

On improvise ?



Analysing the model behaviour may bring to conclude to modelling errors

Les problèmes ne sont pas loin



- Which version of the model was used ?
- How to reproduce or modify this exploration ?
- What was the parameters of the model for these simulations ?
- What was the conclusion of this exploration ?

Nécessité d'explorer l'espace des paramètres

1. Analyses de sensibilité (pour détecter les variables significatives)
2. Détection de trajectoires typiques
3. Calibration (paramètres minimisant une erreur)
4. Calcul de surfaces de réponses (paramètres pour lesquels le modèle satisfait certaines propriétés)

- Nécessité de développer pour chaque modèle une application *ad hoc* pour l'explorer !

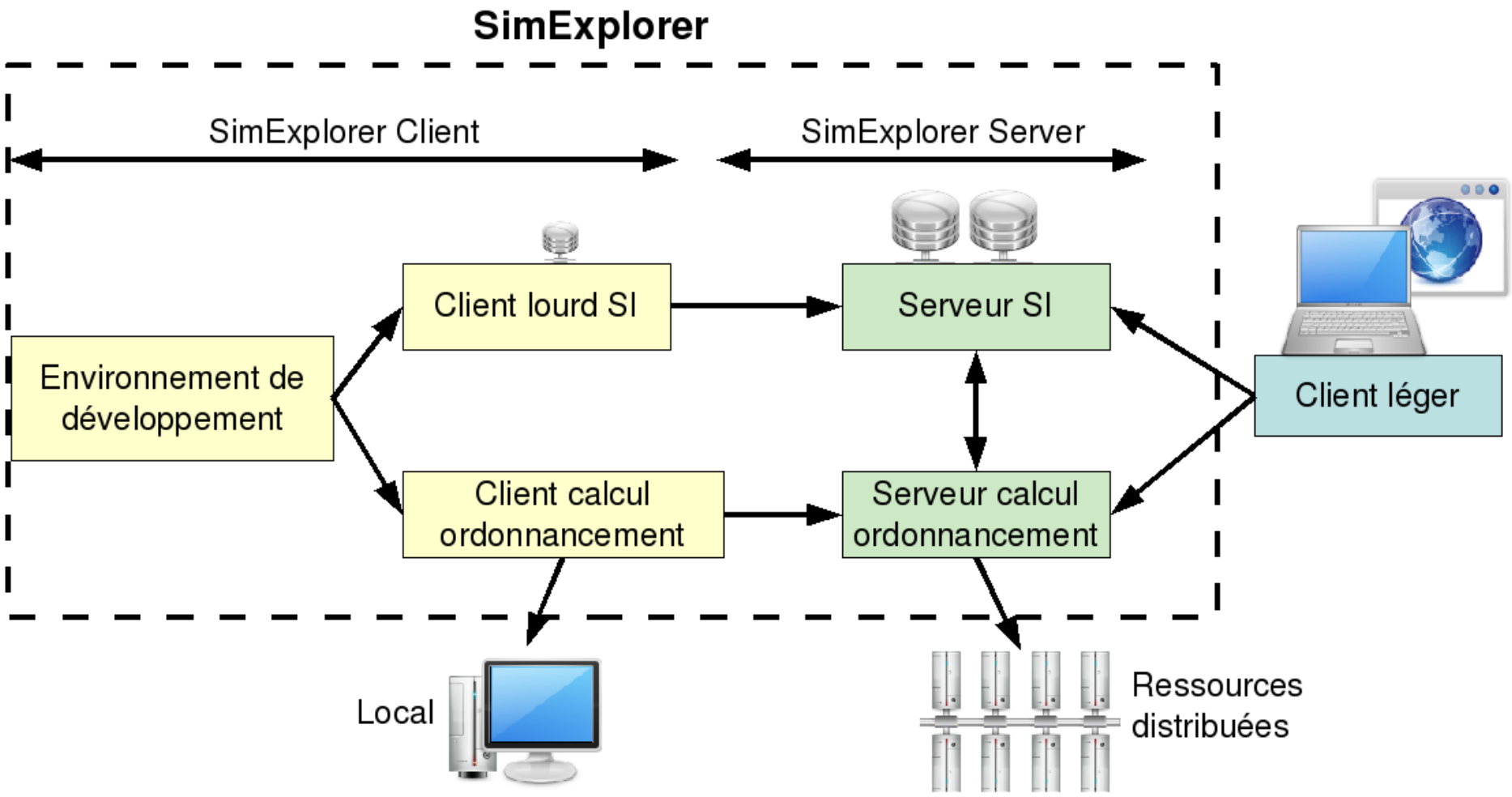
Besoin d'un outil pour

- Éviter de développer un outil *ad-hoc* pour chaque modèle
- Assister la conception de scénarios d'exploration
- Faciliter l'utilisation de clusters ou grilles
- Faciliter l'accès aux bibliothèques de plans d'expériences et d'analyse statistique
- Gérer la reproductibilité / traçabilité des expériences (qualité scientifique)

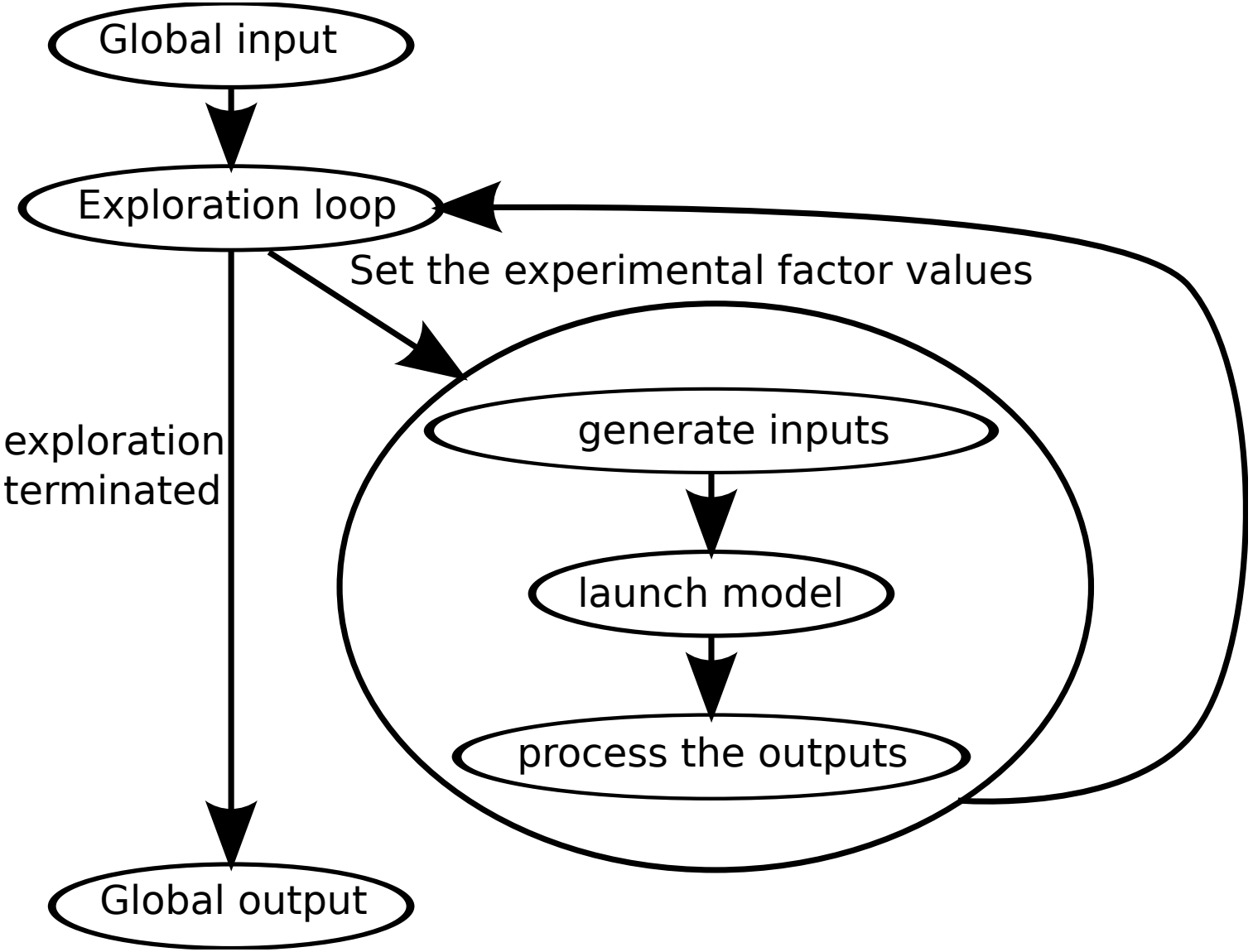
Projet SimExplorer

- Financement :
  - Investissement interne depuis plusieurs années
  - Projet région LifeGrid (financement FEDER)
  - Projet Européen PATRES
- Collaborations :
  - Réseau national Mexico
  - LIMOS (fédération TIMS)
  - Insitut des systèmes complexes Paris Île de France.

Architecture du logiciel



Exploration séquentielle



Exemple d'utilisation : IDE

SimExplorer 0.2.RC8

File Exploration View Window Help

Applications
- bacteria
- Exploration_loop
  - Input_generation
    - GroovyProcessor
    - TemplateFileInputGen
  - Model_launcher
    - SystemExecLauncher
  - Output_processing
    - GroovyProcessor
- Final_processing

a - Properties

Properties	
Type	Double
Domain	RangeDouble
MetaData	
description	
abbreviation	
units	

a

Editor

Here, you can define the plan method used to sample your factors set. Select a method, and configure it (if any)

CentralCompositeDesign

Central Composite Design (CCD) options

Replication

Replication of factorial points	1
Replicates of axial (star) points	1
number of center points	4

Alpha

rotatable	0.0
sphericale	0.0
practicale	0.0
face centered	0.0
other	

Output

Output | IDE Log

Exploration completed !

OutputStructure Window

outputStructure	Type
outputStructure	Complex

Input Structure

inputStructure	Type
inputStructure	Complex
bi	Double
di	Double
alphapi	Double
seed	Integer
Ni	Integer
Npi	Integer
sigma2i	Double

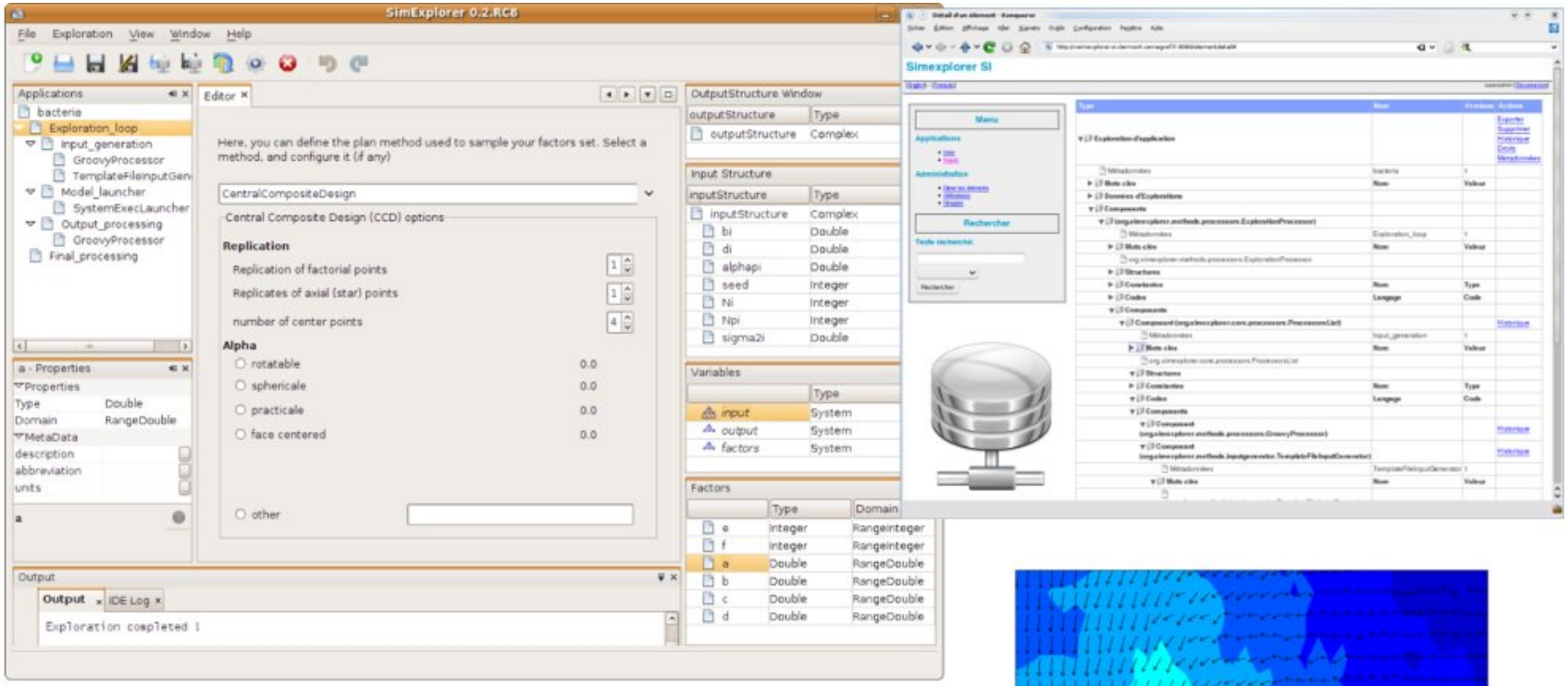
Variables

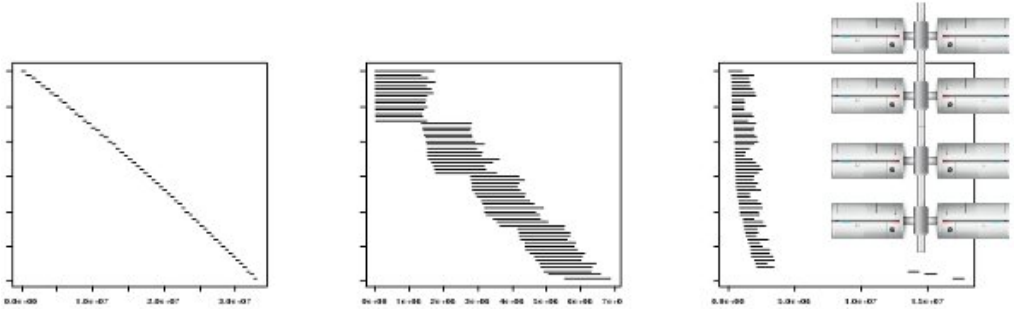
	Type
input	System
output	System
factors	System

Factors

	Type	Domain
e	Integer	RangeInteger
f	Integer	RangeInteger
a	Double	RangeDouble
b	Double	RangeDouble
c	Double	RangeDouble
d	Double	RangeDouble

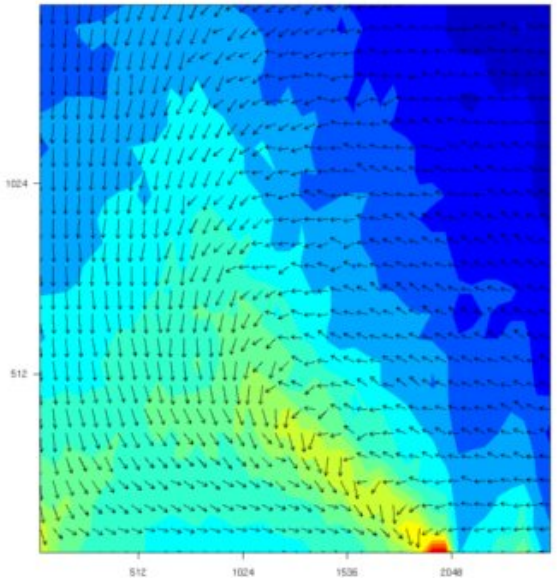
Exemple d'utilisation





(a) Machine locale (b) Cluster de 15 nœuds (c) Grille Auvergrid

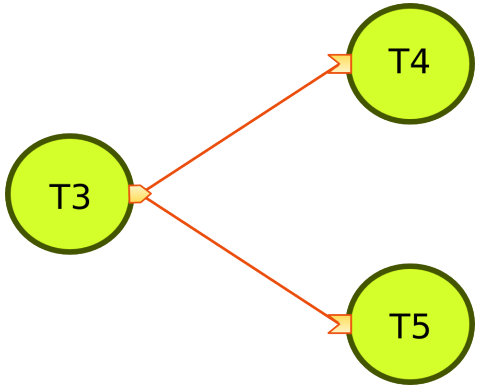
FIG. 9: *Diagrammes de Gantt de l'exploration du modèle de test (60 tâches).*



Naissance d'OpenMOLE

- Conception séquentielle limitée difficilement adaptable ou extensible
- Remise à plat des concepts pour aboutir à une approche par chaîne de traitement modulaire (*Workflow*)
- Relation séquentielle entre des tâches

- Représentation parallèle



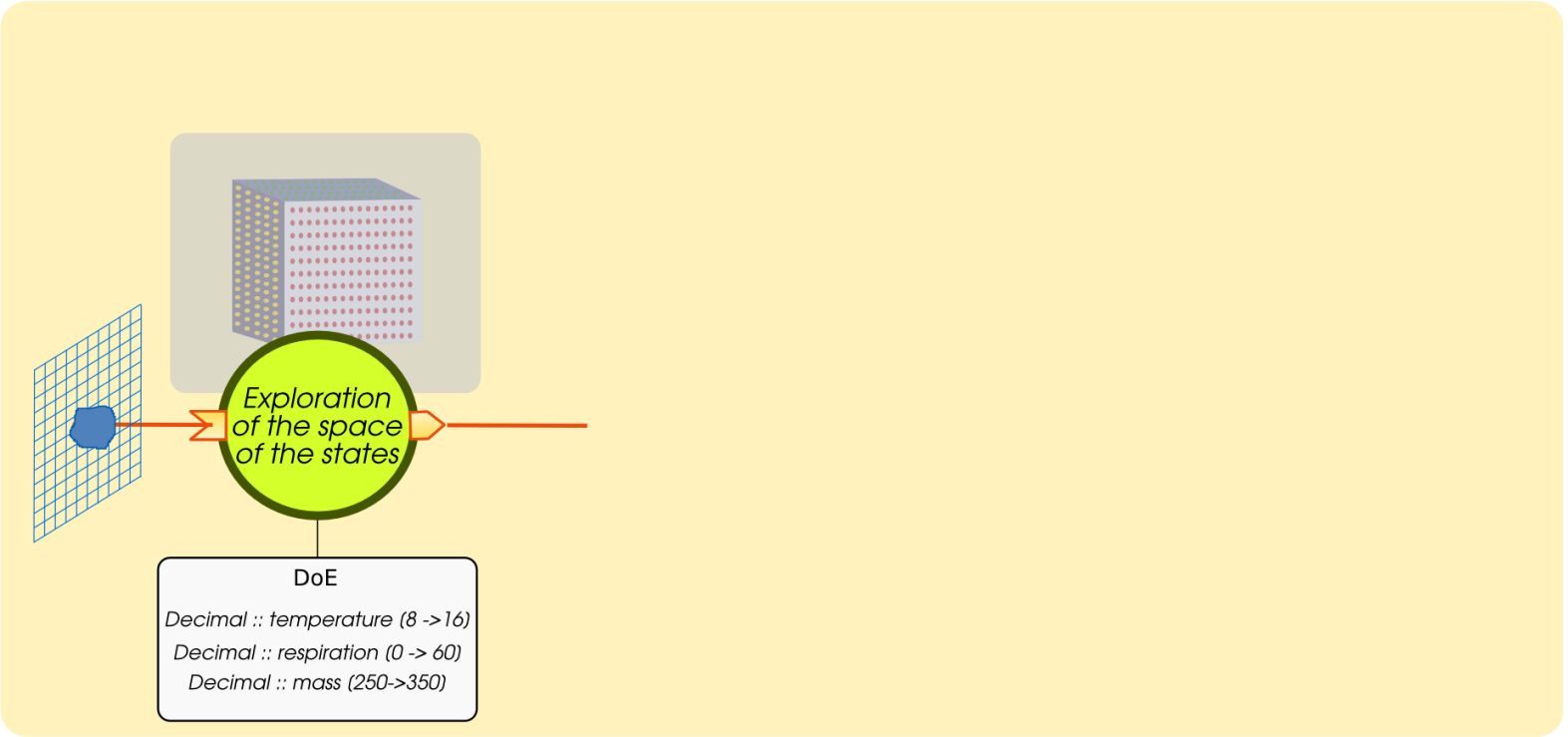
Objectif : Rendre les tâches indépendantes de leur environnement d'exécution

Les concepts d'OpenMOLE

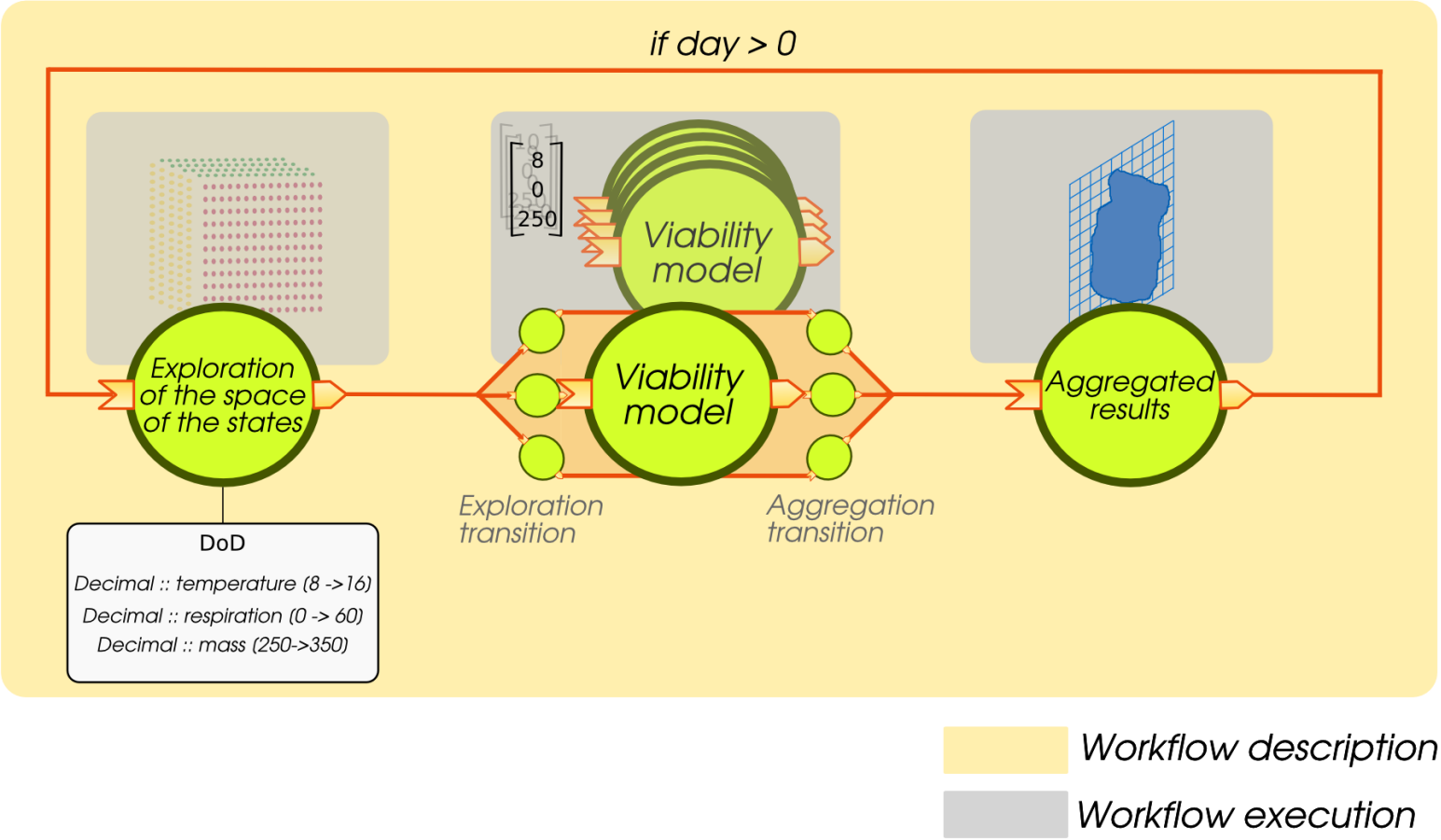
- Les **tâches** sont assimilables à une fonction et définit ses variables et fichiers d'**entrées/sorties**
- La **capsule** accueille une tâche et peut être reliée à une autre capsule à l'aide de **transition**
- Les transitions peuvent être conditionnelles
- L'ensemble constitue un *workflow*
- L'exécution se fait en désignant la capsule de départ, transformant le **worflow** en **MOLE**
- Un gestionnaire de plugin permet d'étendre et spécialiser ces concepts

Application à la viabilité

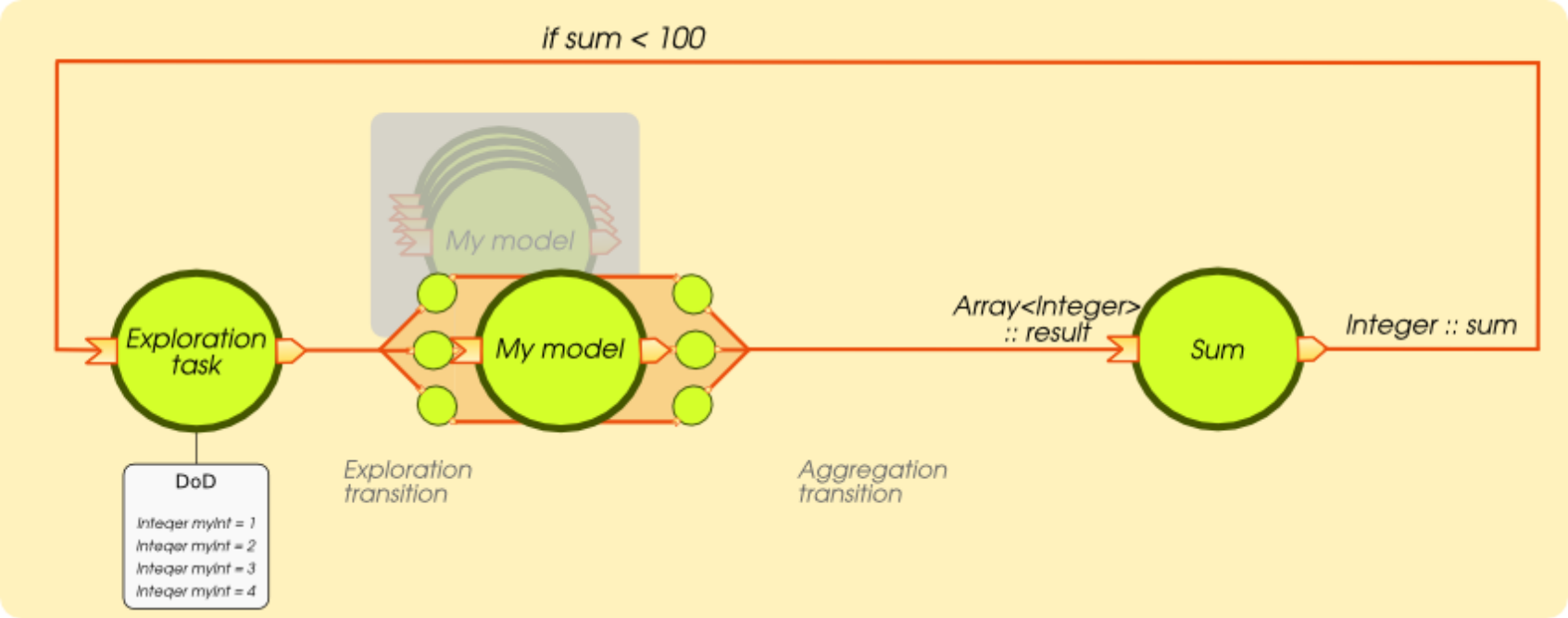
Application à la viabilité



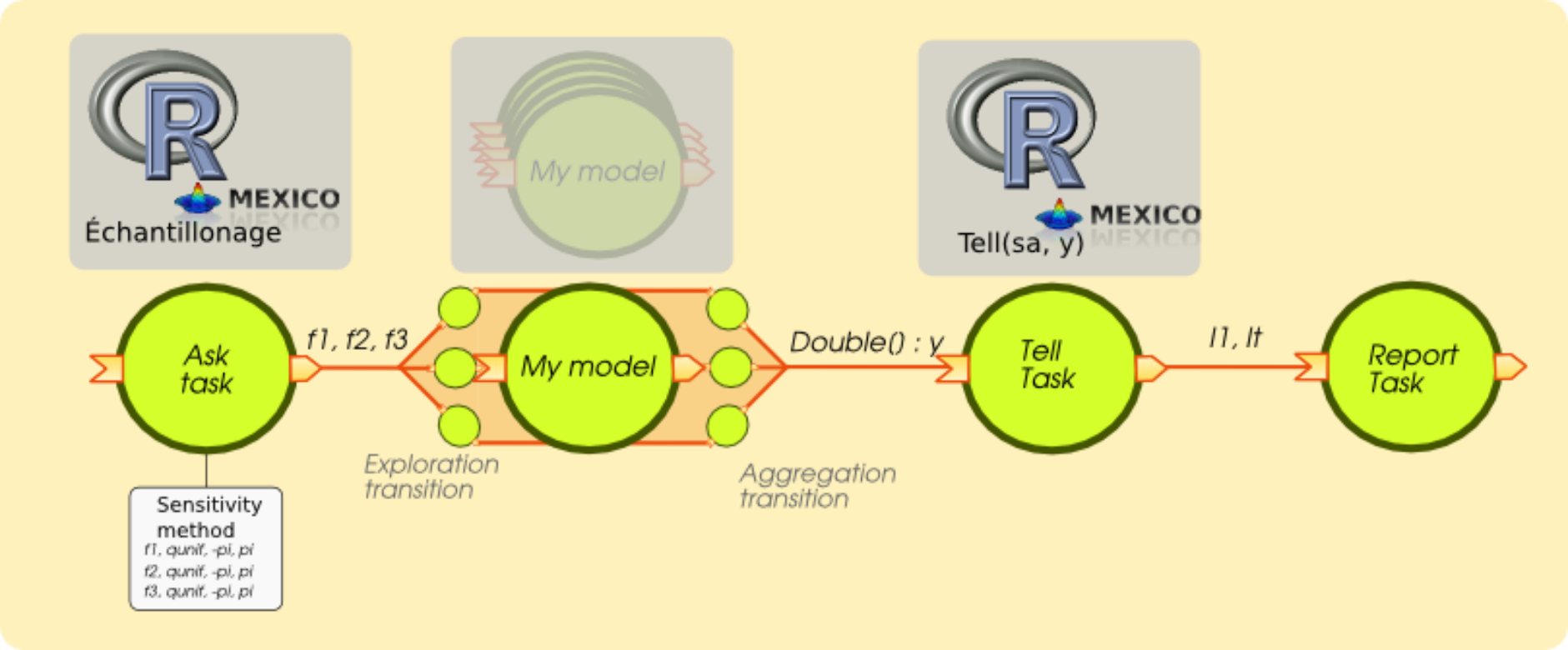
Application à la viabilité



Workflow d'exploration



Workflow pour sensitivity



Workflow pour sensitivity

```
import org.openmole.plugin.task.groovy.GroovyTask
import org.simexplorer.openmole.plugin.task.sensitivity.*

sensitivity.addFactor("f1", Double, new RFunctionDomain("qunif","-pi","pi"))
sensitivity.addFactor("f2", Double, new RFunctionDomain("qunif","-pi","pi"))
sensitivity.addFactor("f3", Double, new RFunctionDomain("qunif","-pi","pi"))

modelTask = new GroovyTask("model")
modelTask.setCode('sleep(3)\ny = Math.sin(f1) + 7.0 * Math.pow(Math.sin(f2), 2) + 0.1 * →
    ↪ Math.pow(f3, 4) * Math.sin(f1)')
sensitivity.setModelTask(modelTask)

reportTask = new GroovyTask("report")
reportTask.setCode('println "First order = ${I1}\nTotal order = ${It}"')
sensitivity.setReportTask(reportTask)

execution = sensitivity.fast99(1000)
execution.start()
```

Workflow pour sensitivity sur une grille

```
import org.openmole.plugin.task.groovy.GroovyTask
import org.simexplorer.openmole.plugin.task.sensitivity.*

sensitivity.addFactor("f1", Double, new RFunctionDomain("qunif","-pi","pi"))
sensitivity.addFactor("f2", Double, new RFunctionDomain("qunif","-pi","pi"))
sensitivity.addFactor("f3", Double, new RFunctionDomain("qunif","-pi","pi"))

modelTask = new GroovyTask("model")
modelTask.setCode('sleep(3)\ny = Math.sin(f1) + 7.0 * Math.pow(Math.sin(f2), 2) + 0.1 * →
    ↪ Math.pow(f3, 4) * Math.sin(f1)')
sensitivity.setModelTask(modelTask)

reportTask = new GroovyTask("report")
reportTask.setCode('println "First order = ${I1}\nTotal order = ${It}"')
sensitivity.setReportTask(reportTask)

import org.openmole.plugin.environment.glite.GliteEnvironmentDescription
sensitivity.setModelEnvironment(new GliteEnvironment("vo.iscpif.fr",
    "voms://grid12.lal.in2p3.fr:20013/O=GRID-FR/C=FR/O=CNRS/OU=LAL/CN=grid12.lal.in2p3. →
        ↪ fr",
    "ldap://topbdii.grif.fr:2170")
)

execution = sensitivity.fast99(1000)
execution.start()
```

Conclusion

- Plateforme flexible, ouverte et collaborative
- SimExplorer et OpenMOLE diffusés en GPL
- Applications 100% portables (bientôt)
- Accès aux ressources de calcul distribuées

http ://www.simexplorer.org

http ://www.openmole.org