

Standard Description of Big Data Neuroimaging Experiments



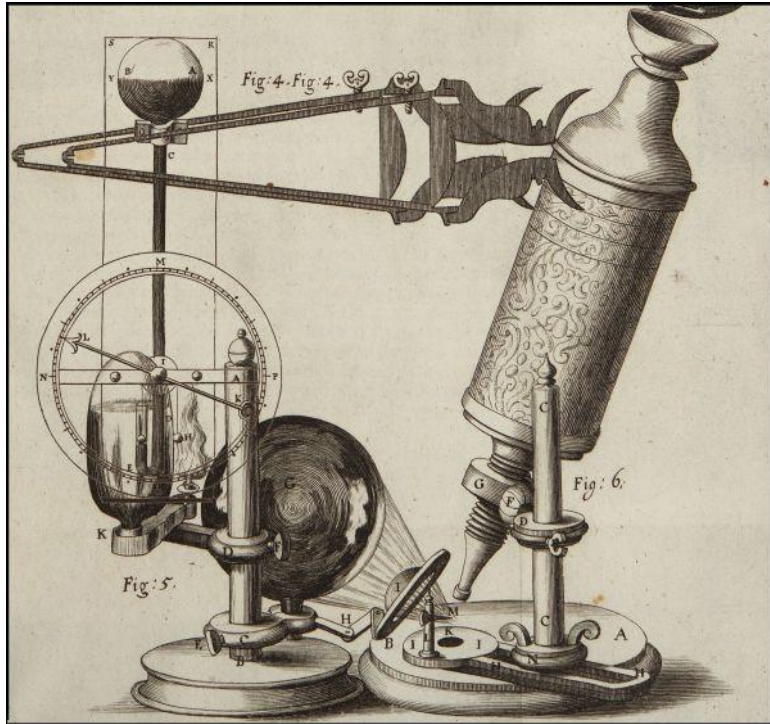
Bridging the gap



A brief history

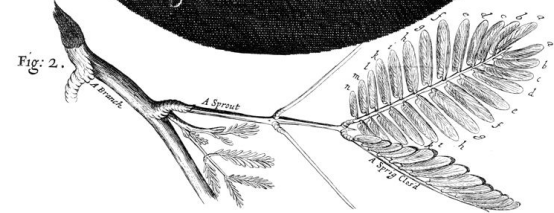
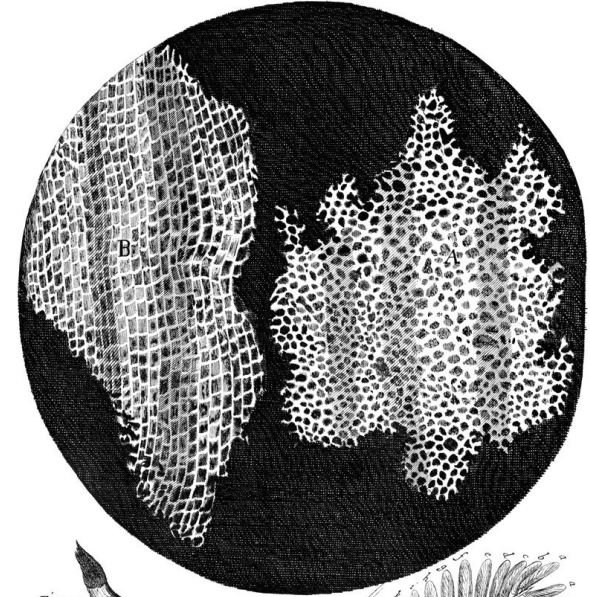


A brief history

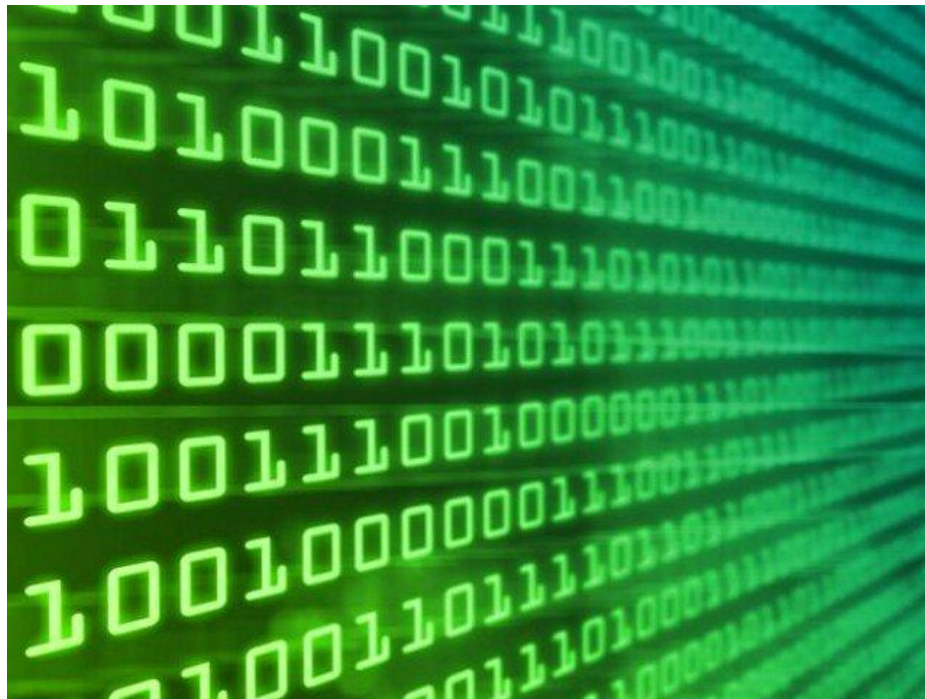


Schem. XI.

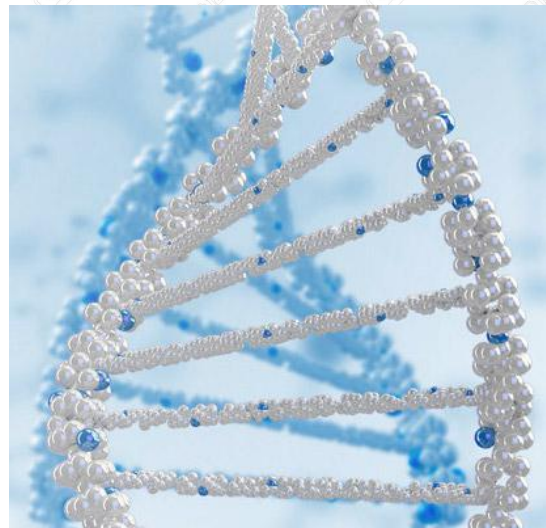
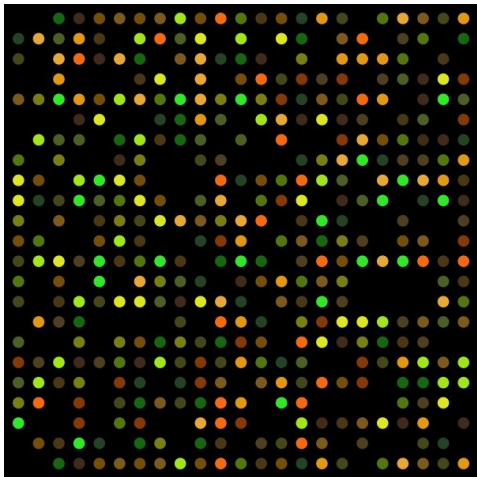
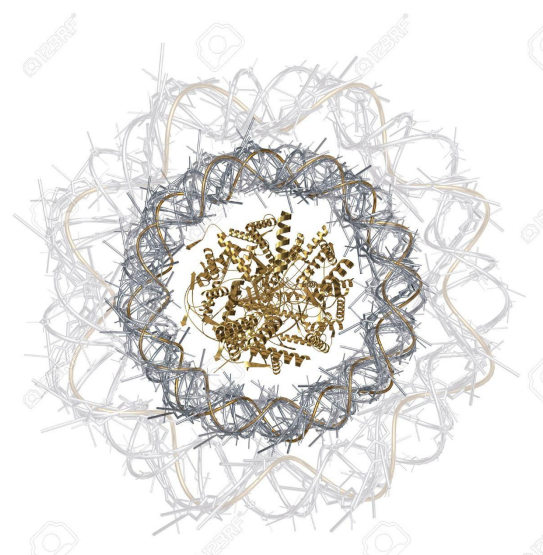
Fig. 1.



A brief history



A brief history



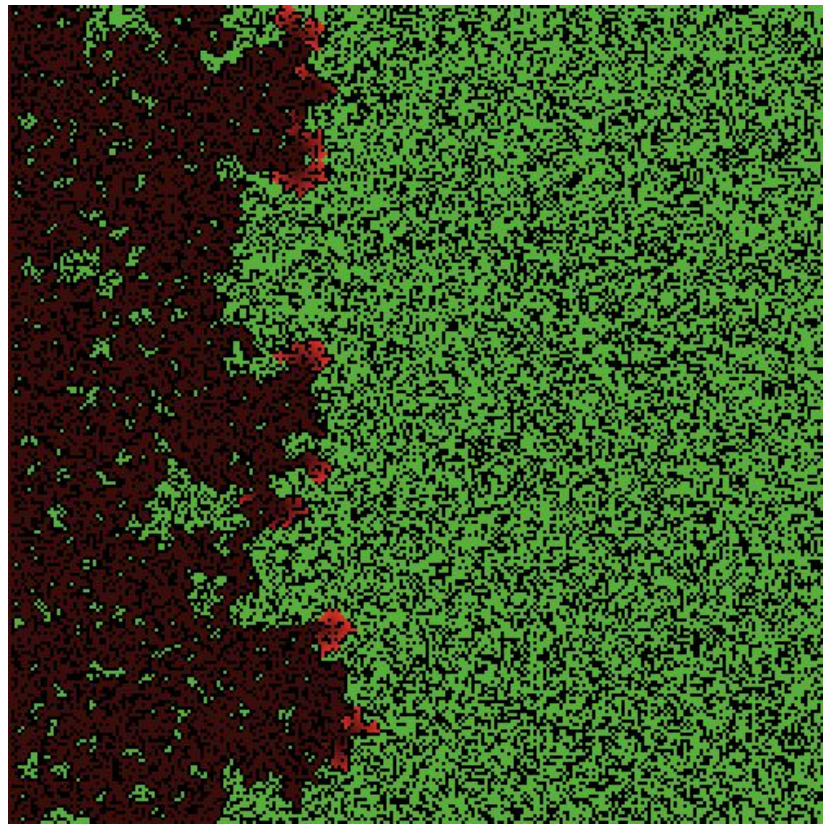
Workflow systems

Encapsulate processes

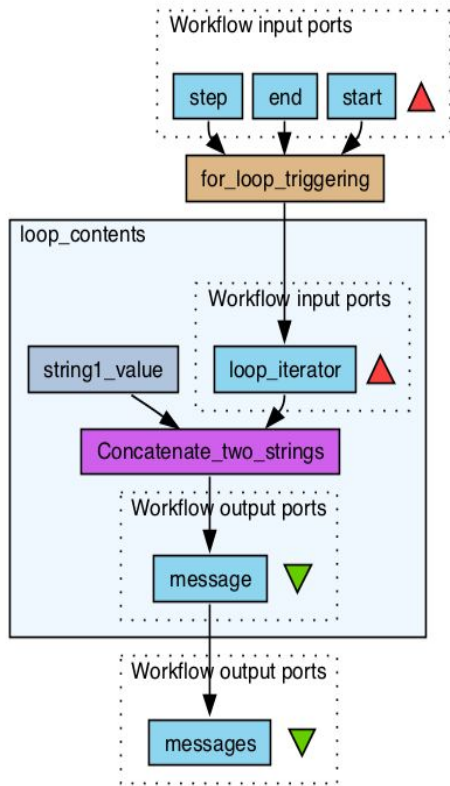
Remote execution

Data provenance

Parameter space exploration



Workflow systems



```
val i = Val[Double]
val res = Val[Double]
```

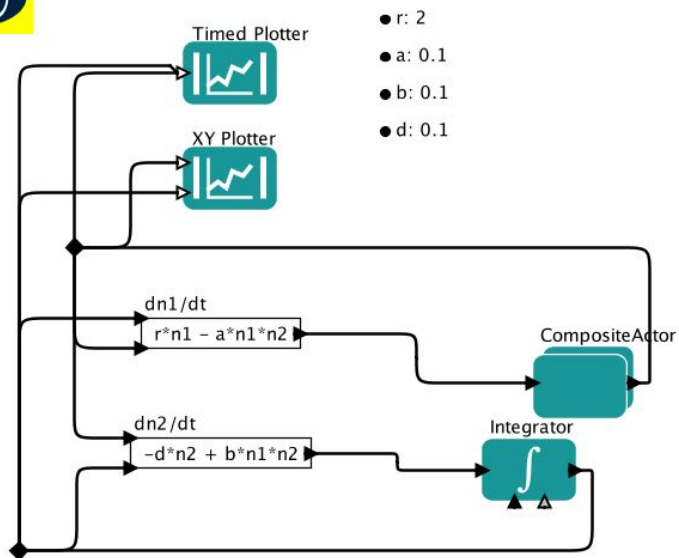
```
val exploration = ExplorationTask(i in (0.0 to 10.0 by 1.0))
```

```
val model =
  ScalaTask("val res = i * 2") set (
    inputs += i,
    outputs += (i, res)
  )
```

```
val env = LocalEnvironment(5)
```

```
exploration -< (model on env hook ToStringHook())
```

Continuous Director



Common Workflow Language



Unifies workflow systems under one language

Provides its own schema language for validation

Supports many essential features for workflows

OpenMOLE



The diagram illustrates the OpenMOLE architecture. At the top left is the 'OpenMOLE' title. Below it is a thick green horizontal bar. A large white rectangle with a thin black border occupies the center. To the left of this rectangle are the labels 'Language' and 'Remote Execution'. To the right are 'Workflow Editor' and 'Client Program'. Inside the rectangle is the label 'Engine'.

Language

Workflow
Editor

Engine

Remote
Execution

Client
Program

OpenMOLE



Advantages:

- model experiments
- remote execution

Disadvantages:

- no data provenance tracking

OpenMOLE DSL



Scala

Tasks

Samplings

Hooks

Environments

Goal



Comparison



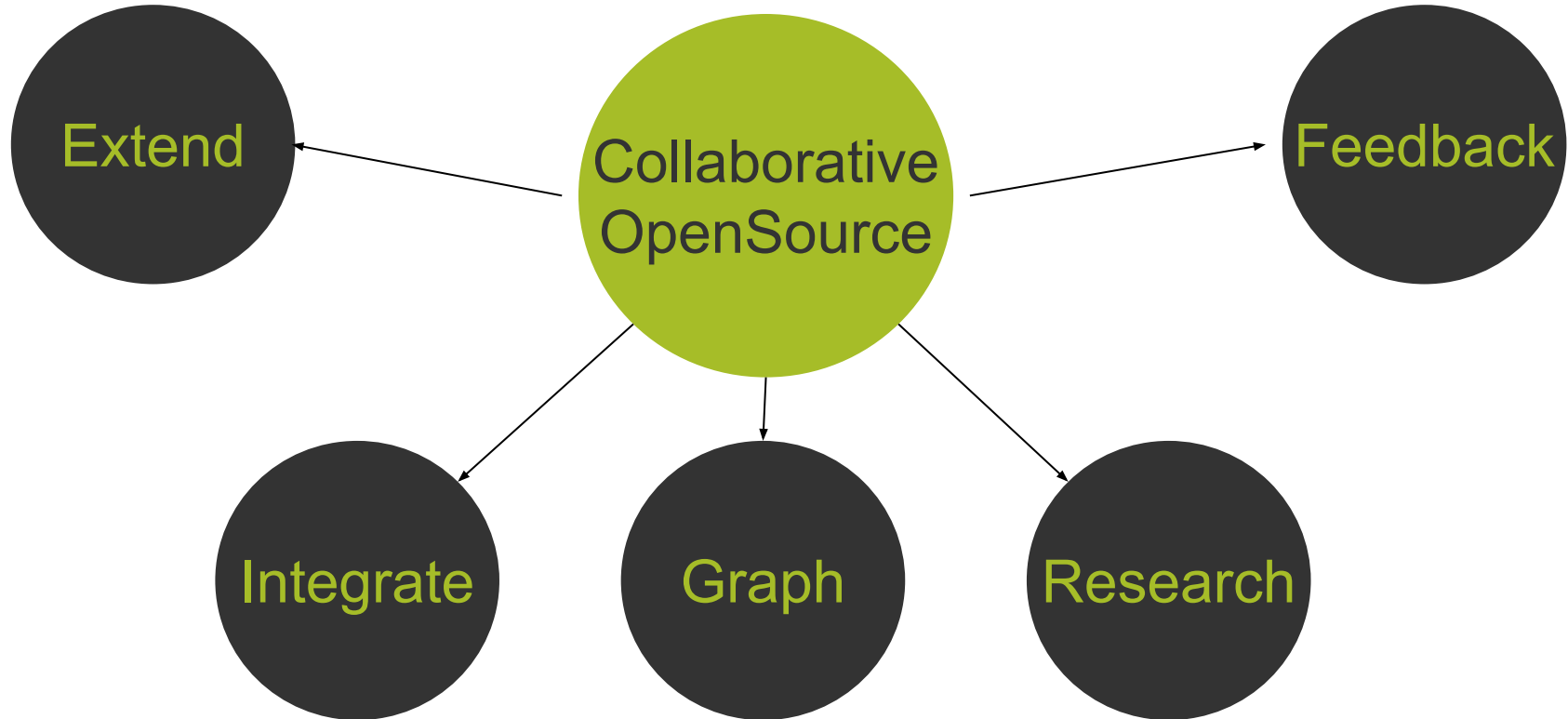
- Steps
- Labels
- Types
- Natural parallelism
- Exploration
- Aggregation
- Inline Javascript expressions
- Docker
- Shell
- User control over HW resource allocation

Design and Implementation



DEMO

So what's next?



Contributions



Analysed existing workflow platforms

Comparison between CWL and the OpenMOLE DSL

Extendable and modularised approach

Parser and Translator (basic implementation)

Prototype for graphical representation

Thank you!



Questions?

