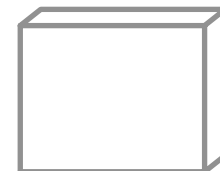
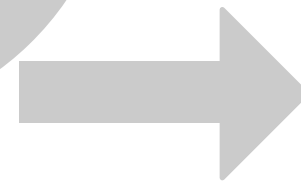
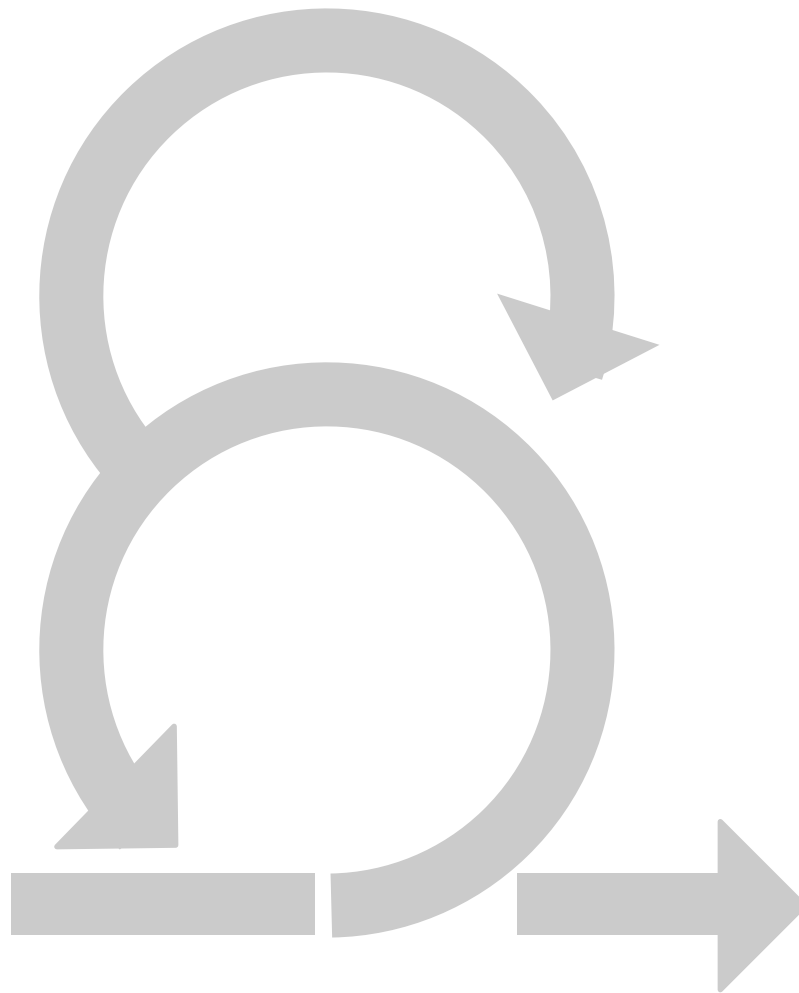
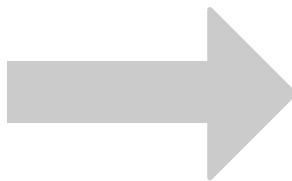
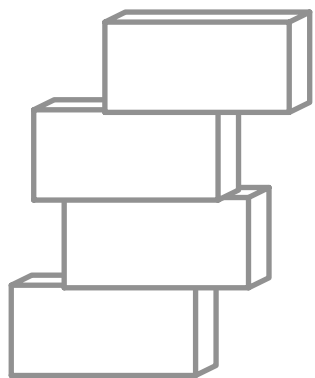


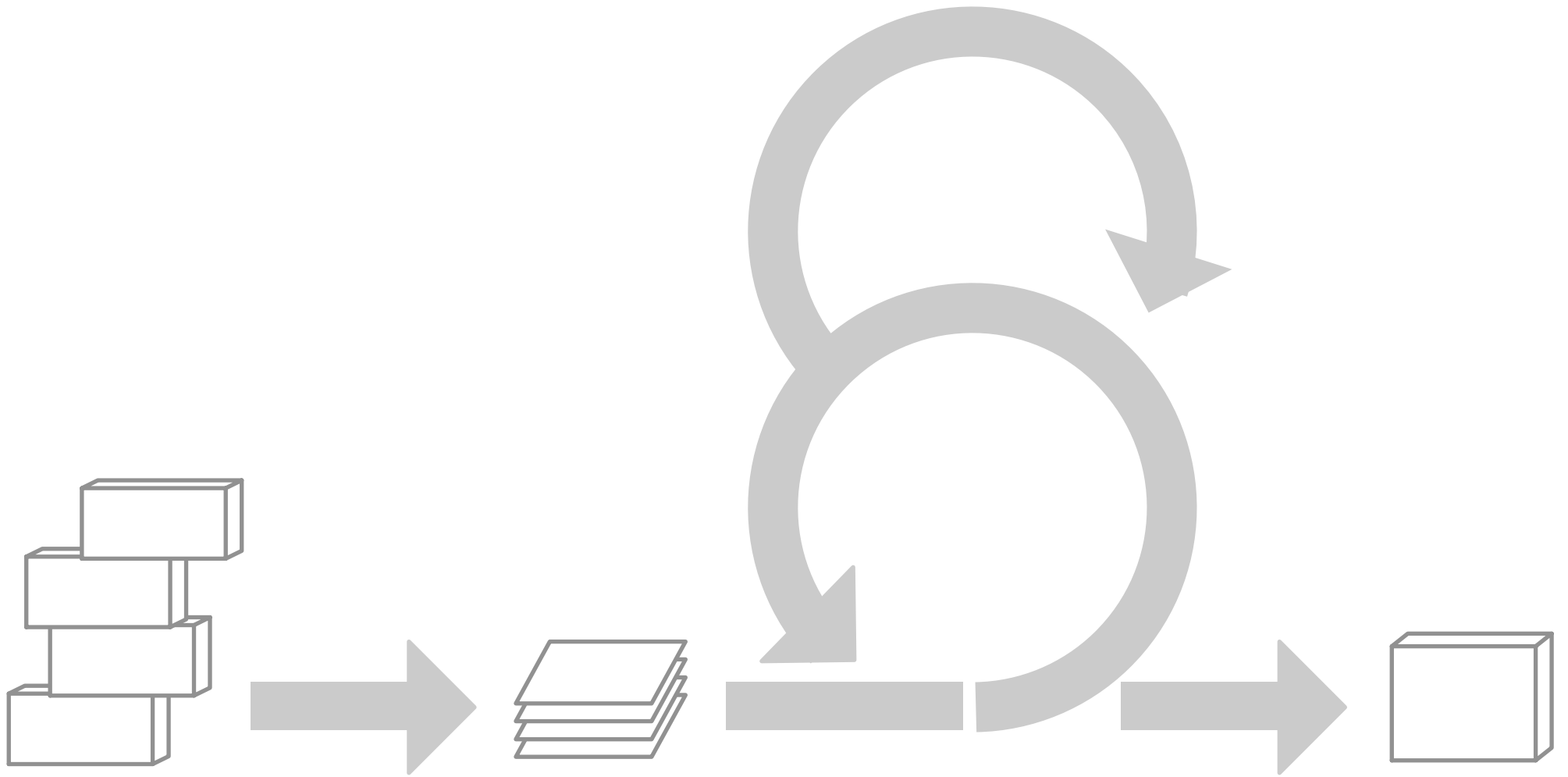
Analyses Logicielles Dédiées

Stéphane Ducasse
<http://rmod.inria.lille.fr>
RMOD / synectique
INRIA

Contrôle processus industriels ?

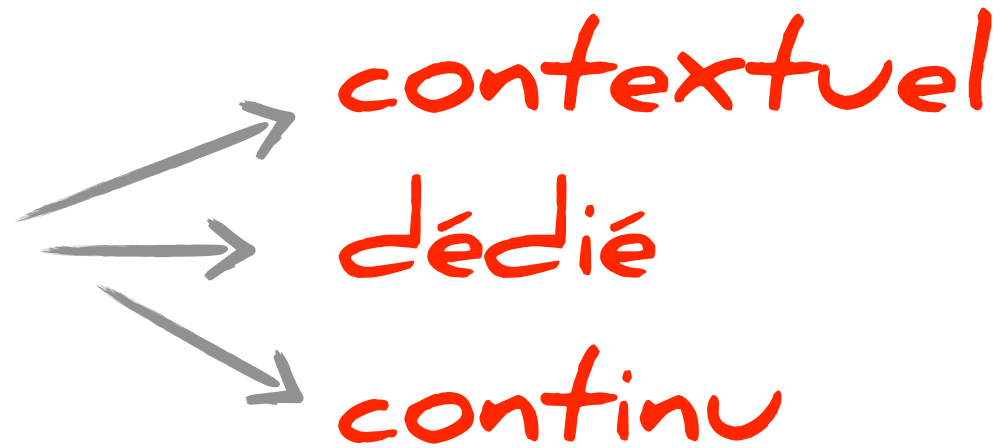


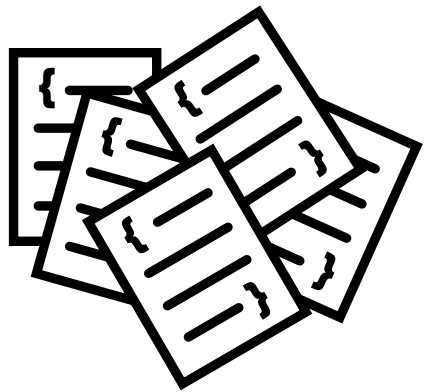




du retour **est la clef**

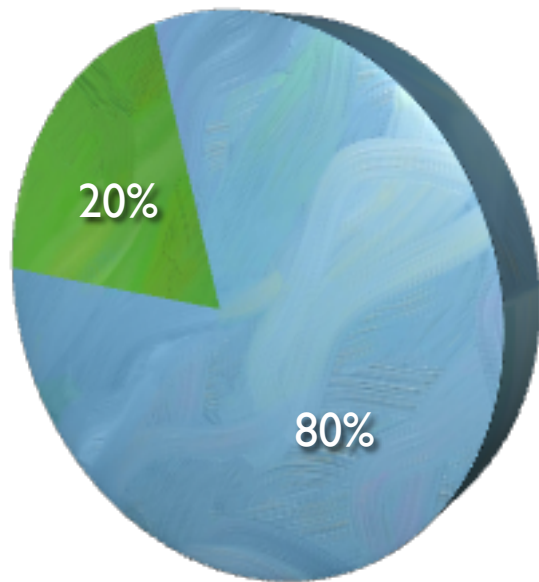
Mais du retour





= *complexe,* **large**

Analyse de Coût

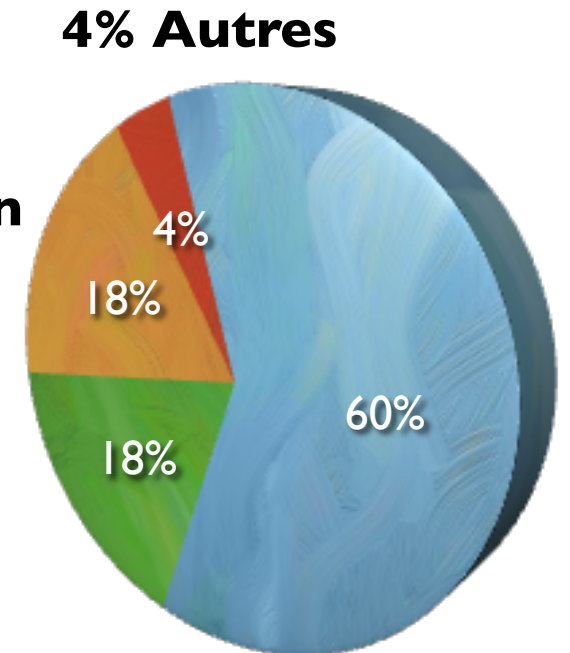


Entre **50%** and **80%** du cout *global est passé dans l'évolution [1992]*



18% Adaptation

18% Bugs



4% Autres

60% Nouvelles fonctions

Software is a living entity...

Early decisions were certainly good at that time

But the context *changes*

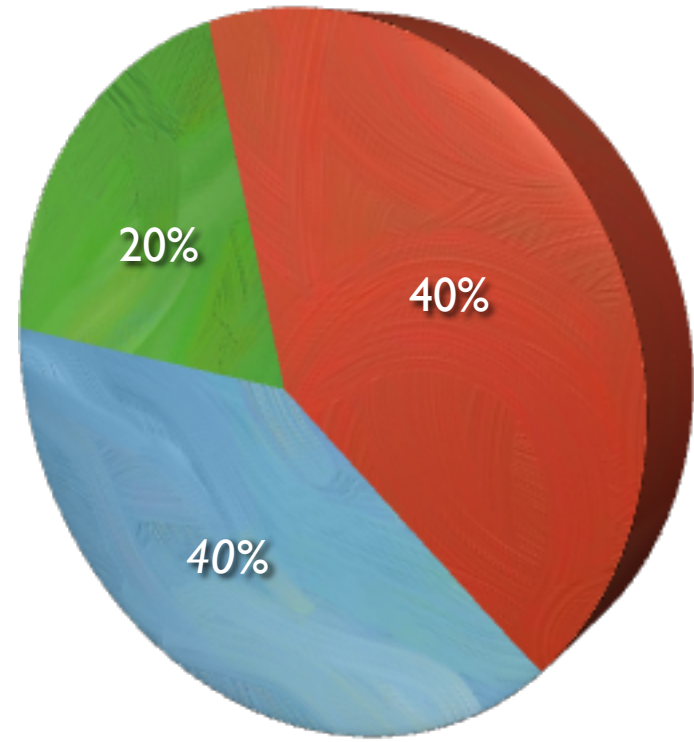
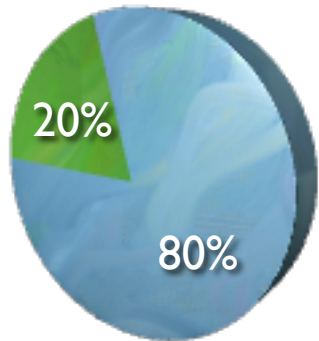
Customers *change*

Technology *changes*

People *change*



50% du temps de developpement est passé à lire le code !



Entre **50%** and **80%** du cout **global est passé dans l'évolution**

On perd énormément de temps avec des pratiques inadaptées et inefficaces

Des outils dédiés
pour nous aider !

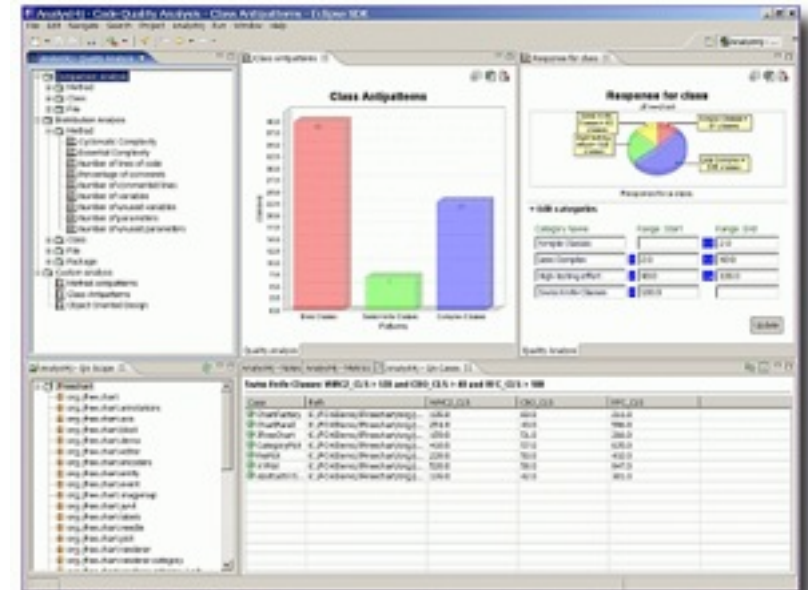


Des outils pour du
contrôle et du
retour !



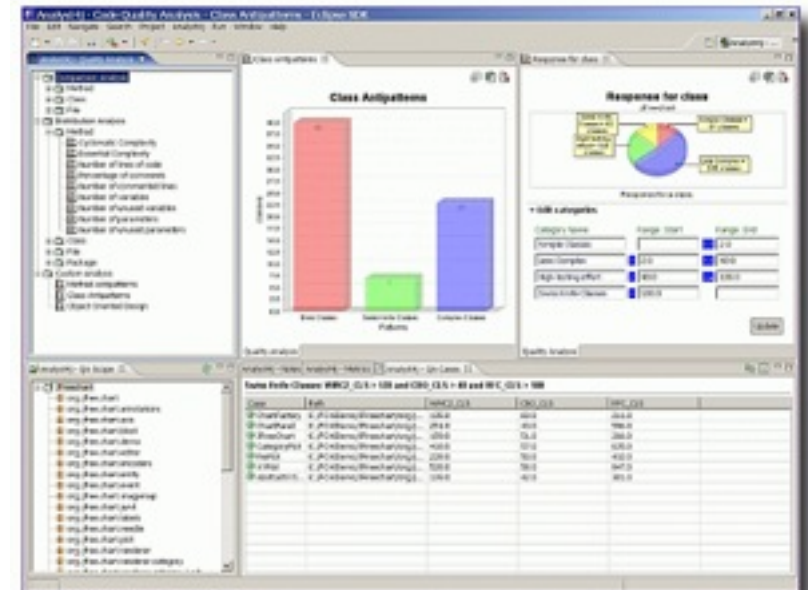
manuel
dédié

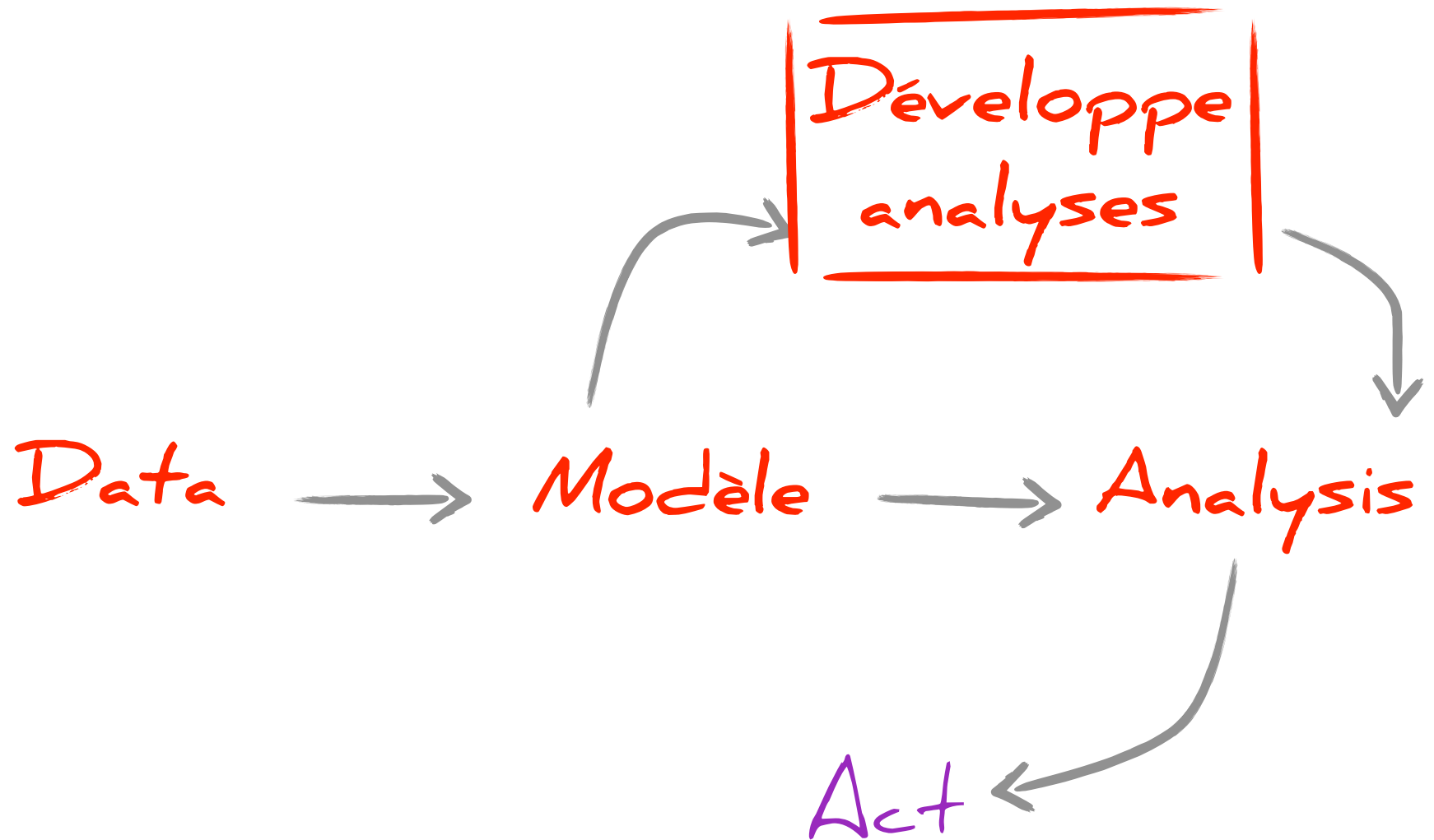
automatique
générique



~~manuel~~
dédié

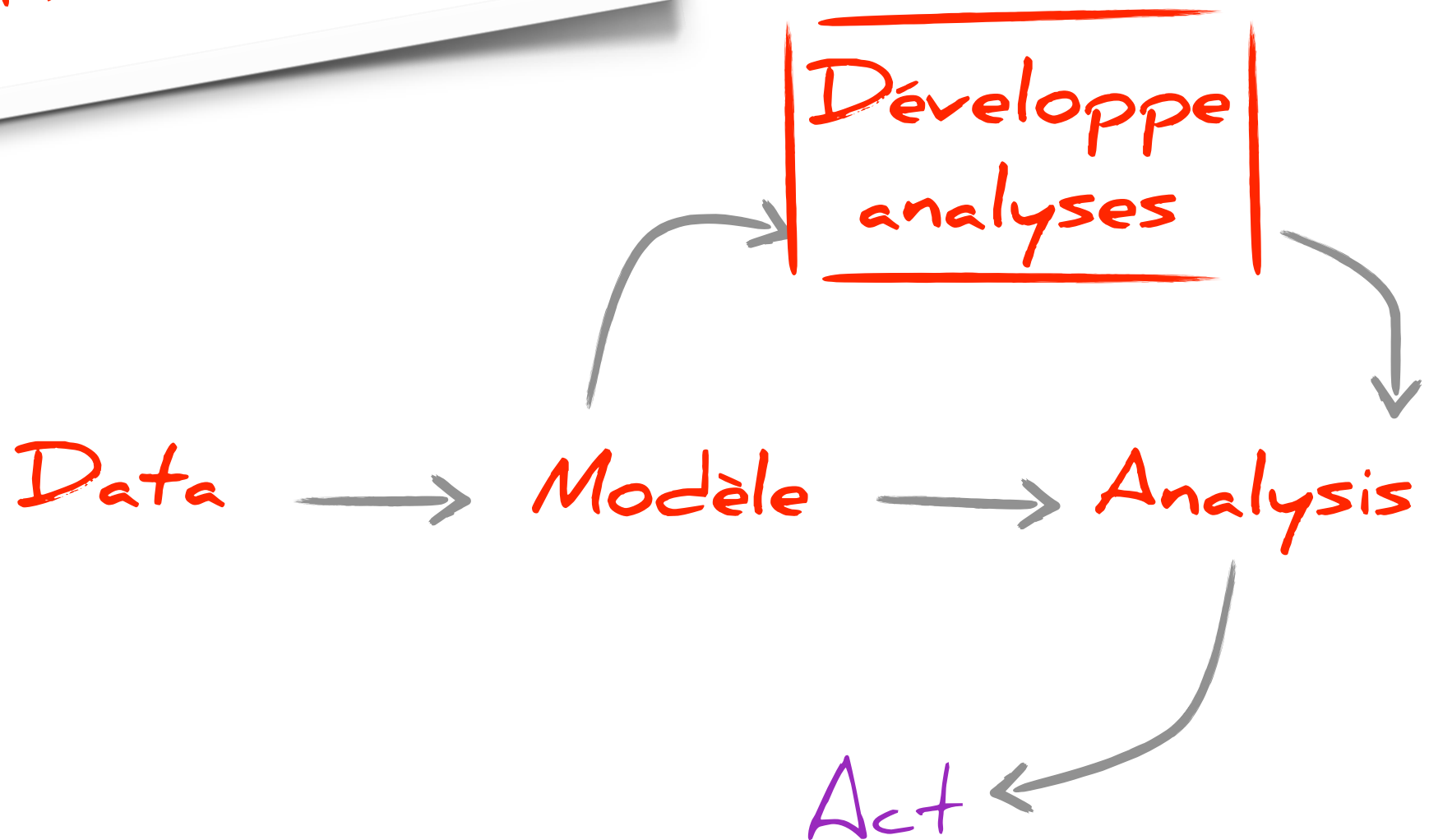
automatique
~~générique~~





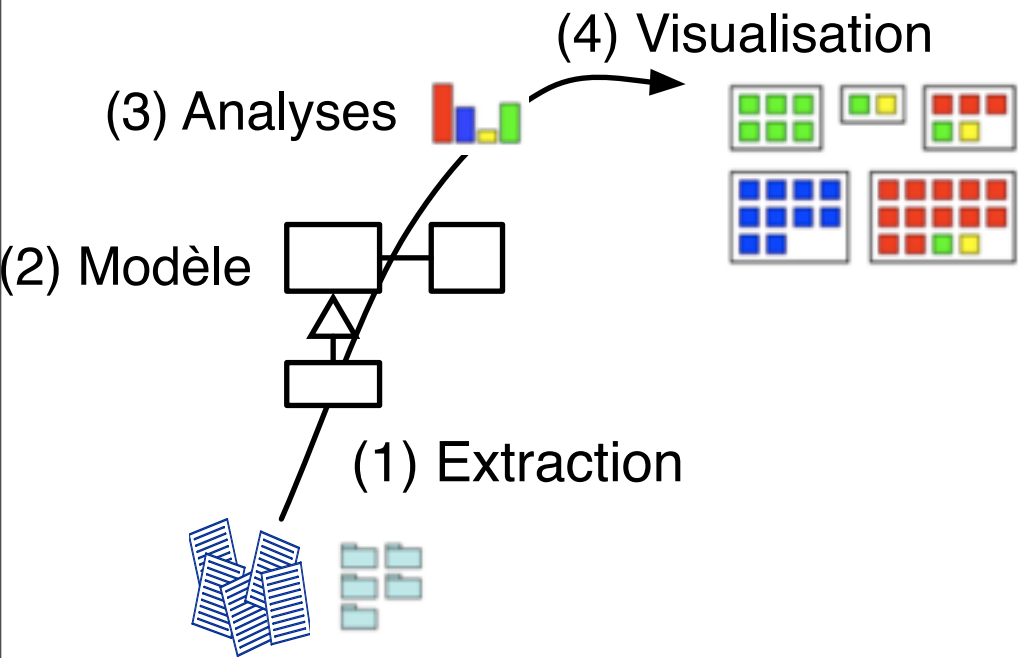
une analyse doit amener à une prise de décision

Outils Dediés



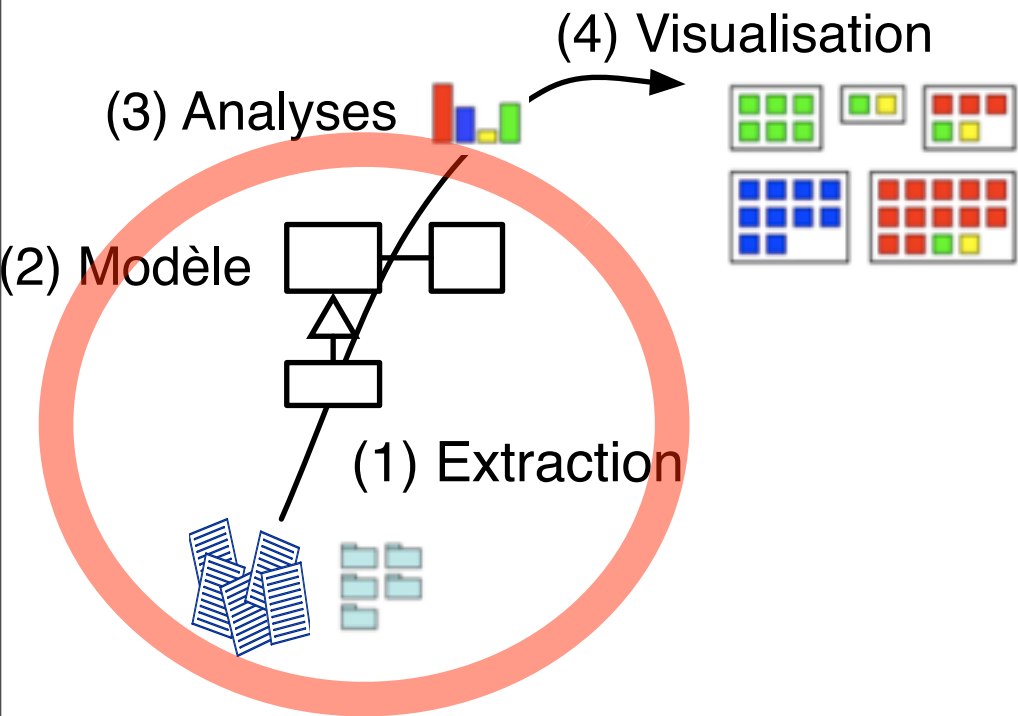
une analyse doit amener à une prise de décision

Exemple : qui est derrière le package X ?

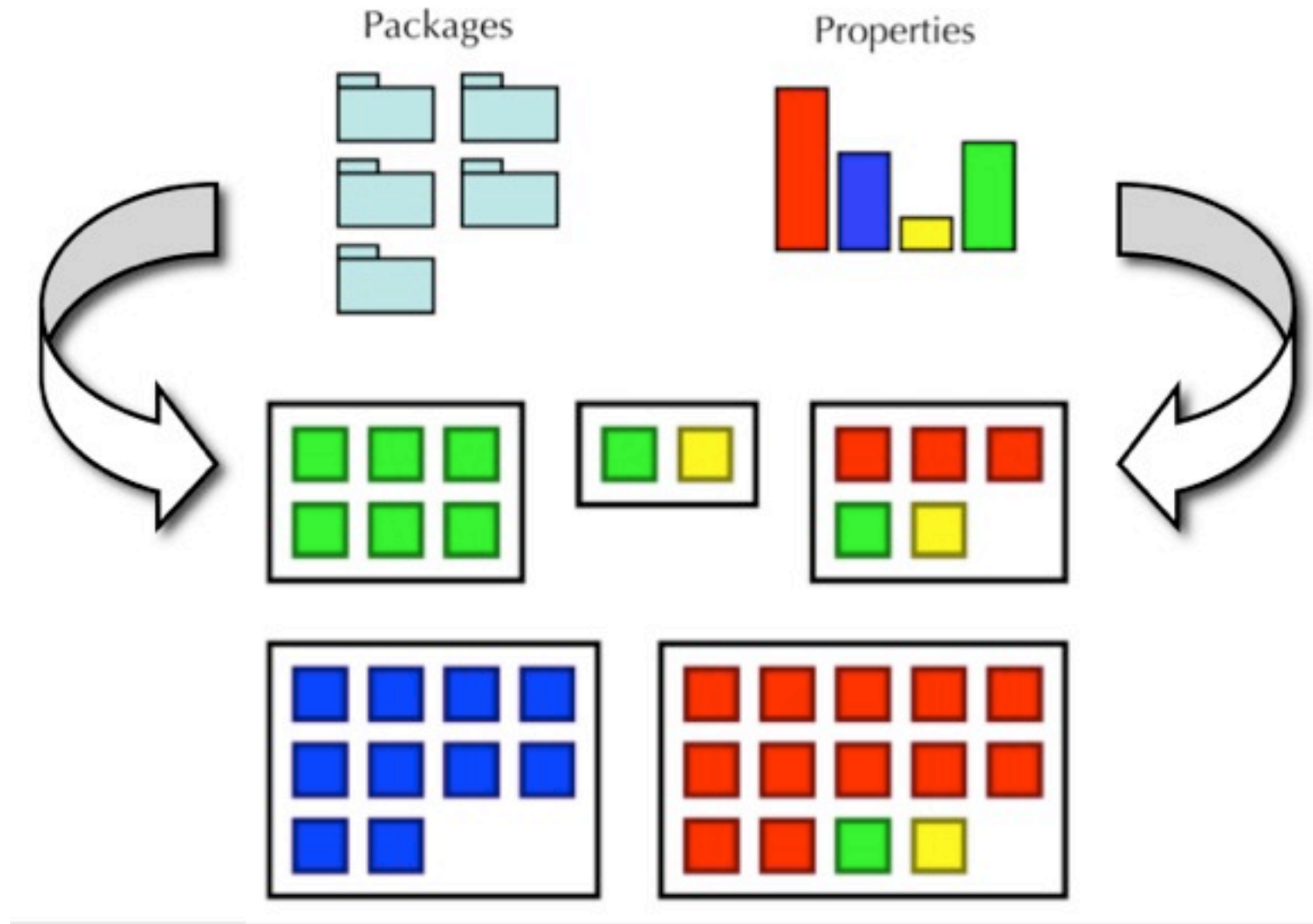


Définition d'un modèle de propriétés

Extraction de données (CVS...)

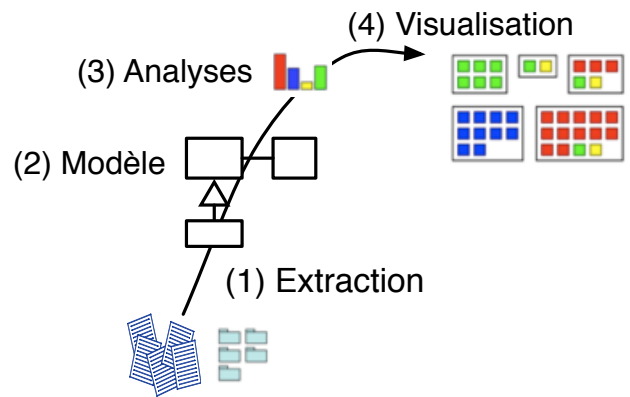


(3) Techniques de cartographies



Carte de Distribution

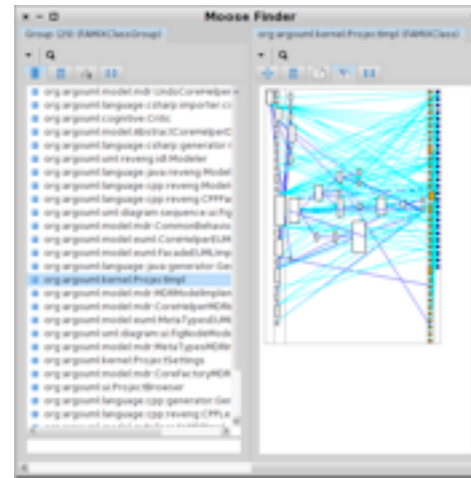
JBoss en un clin d'oeil



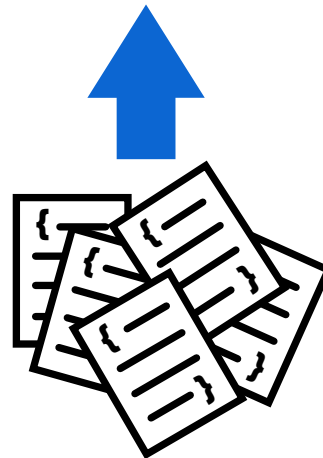
classes select: #isGod

McCabe = 21

LOC = 753,000



Syn'Tool Suite



Métriques logiciels (best of)

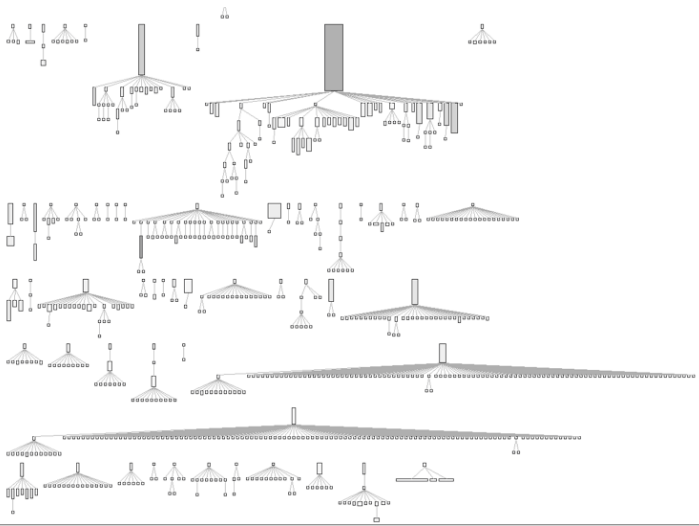
Modèles de qualité

ISO 9126, Squalé (PSA-AirFrance)

Adaptation rapide

Définition spécifique au business

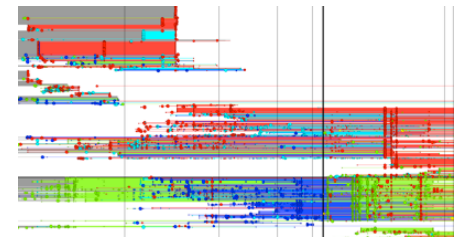
Cartes et visualisations dédiées



System Complexity



Carte de Distribution



Queries dans un contexte

The screenshot shows the Moose Panel interface. On the left, a list of 70 model classes is displayed under the heading "All model classes (70) (FAMIXClassGroup)". On the right, a list of 16 selected classes is shown under "Group (16) (FAMIXClassGroup)". At the bottom of the panel, a search bar contains the query: `each numberOfLinesOfCode > 100`. This search bar and its contents are circled in red.

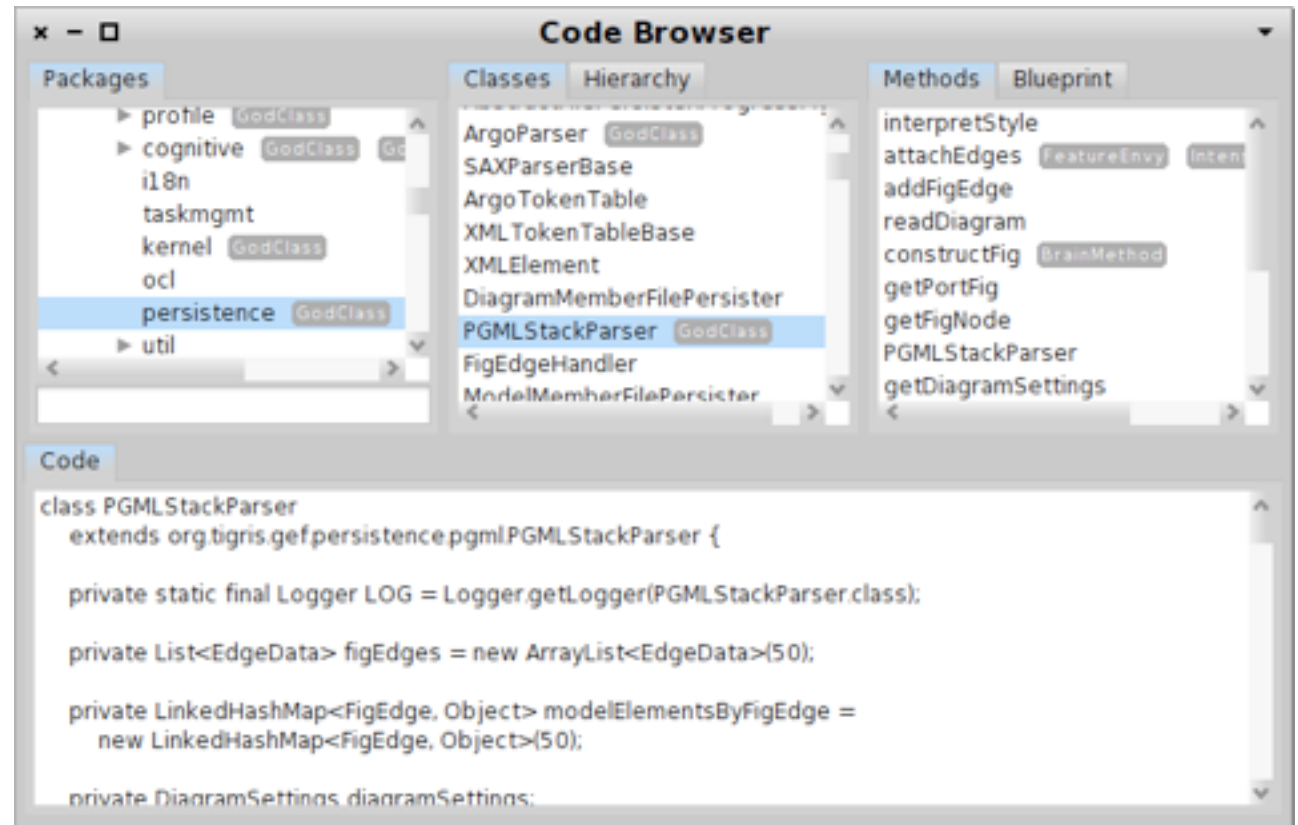
The screenshot shows the System Complexity Locator interface. It displays a title "System Complexity with 16 selected classes" and a list of classes: RBProgramNodeVisitor, RBScanner, RBPParser, RBVariableBinding, RBToken, RBLexicalScope, and RBProgramNode. Below the list, there are complexity diagrams for each class, represented by red vertical bars and tree structures. A grey arrow points from the Moose Panel to this interface. At the bottom, a status bar indicates "Group subtracted with Group (7 FAMIXClasses)".

Outils spécifiques

riches

compacts

meilleur focus



Aggrégation de données / ponts entre outils

parseur combineur
modulaire

Exemple : corrélérer les bugs et la
couverture de tests



Analyses dédiées



The screenshot shows the Mondrian Easel software interface. The main window displays a class hierarchy diagram with a root node and several child nodes, each represented by a rectangle. The diagram is rendered in a tree layout. Below the diagram, there is a configuration panel with the following text:

```
view interaction menu: #mooseMenu.  
view shape rectangle  
  height: #numberOfMethods;  
  width: #numberOfAttributes;  
  linearFillColor: #numberOfLinesOfCode within: classGroup.  
view nodes: classGroup.  
view edgesFrom: #superclass.  
view treeLayout
```

At the bottom of the configuration panel, there is a button labeled "Generate View".

Outils spécifiques

The screenshot shows the Code Browser IDE interface. It is divided into three main sections: Packages, Classes, and Methods. The Packages section on the left shows a tree view with 'persistence' selected. The Classes section in the middle shows a list of classes, with 'PGMLStackParser' selected. The Methods section on the right shows a list of methods, with 'interpretStyle' selected. Below these sections is a Code editor showing the source code for the selected class.

```
class PGMLStackParser
  extends org.tigris.gef.persistence.pgml.PGMLStackParser {

  private static final Logger LOG = Logger.getLogger(PGMLStackParser.class);

  private List<EdgeData> figEdges = new ArrayList<EdgeData>(50);

  private LinkedHashMap<FigEdge, Object> modelElementsByFigEdge =
    new LinkedHashMap<FigEdge, Object>(50);

  private DiagramSettings diagramSettings;
```

```

b := GLMTabulator new.
b column: #namespaces;
column: #classes;
column: #methods.
b transmit to: #namespaces;
andShow: [:a |
  a tree
    display: [ :model |
      model allNamespaces
      select: #isRoot ];
    children: #childScopes;
    format: #name ].
b transmit to: #classes;
from: #namespaces;
andShow: [:a |
  a list
    display: #classes;
    format: #name ].
b transmit to: #methods;
from: #classes;
andShow: [:a |
  a list
    display: #methods;
    format: #signature ].

```

```

b transmit
  toOutsidePort: #class;
  from: #classes.
b transmit to: #methods;
from: #methods.
B := GLMTabulator new.
B title: 'Code Browser'.
B row: #nav;
row: #details.
B transmit to: #nav;
andShow: [:a |
  a custom: b ].
B transmit to: #details;
from: #nav port: #class;
andShow: [:a |
  a text
    display: #sourceText ].
B transmit to: #details;
from: #nav port: #method;
andShow: [:a |
  a text
    display: #sourceText ].

```

Intéressé par vos problèmes

- extraction de règles
- cout
- impact change
- service oriented architecture
- migration
-