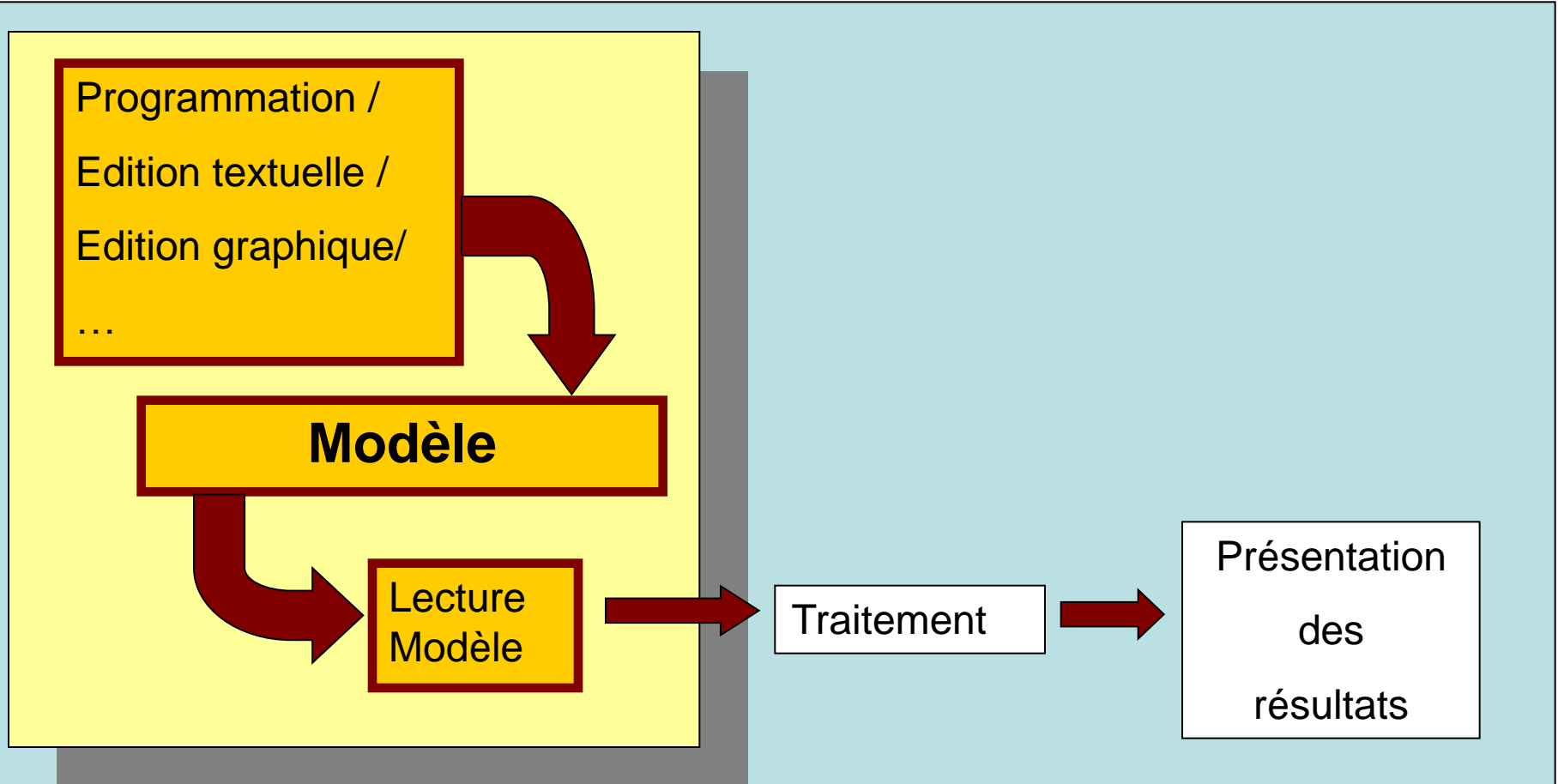


Modélisation sous Eclipse



Solutions apportées par Eclipse ?

Introduction

<http://www.eclipse.org/>

eclipse
INDIGO

Ganymede	2008	3.4
Galileo	2009	3.5
Helios	2010	3.6
Indigo	2011	3.7

- 2001 Début
- 2004 Fondation indépendante non basée sur le profit
- Open source software

~ 1 million downloads/month

Des milliers de membres contributeurs (<http://www.eclipse.org/membership/>)

Des centaines de plug-ins

Eclipse 3.7 Indigo (<http://www.eclipse.org/>)

eclipse-modeling-indigo-win32.zip



Eclipse Modeling Tools, 271 MB

Downloaded 40,184 Times

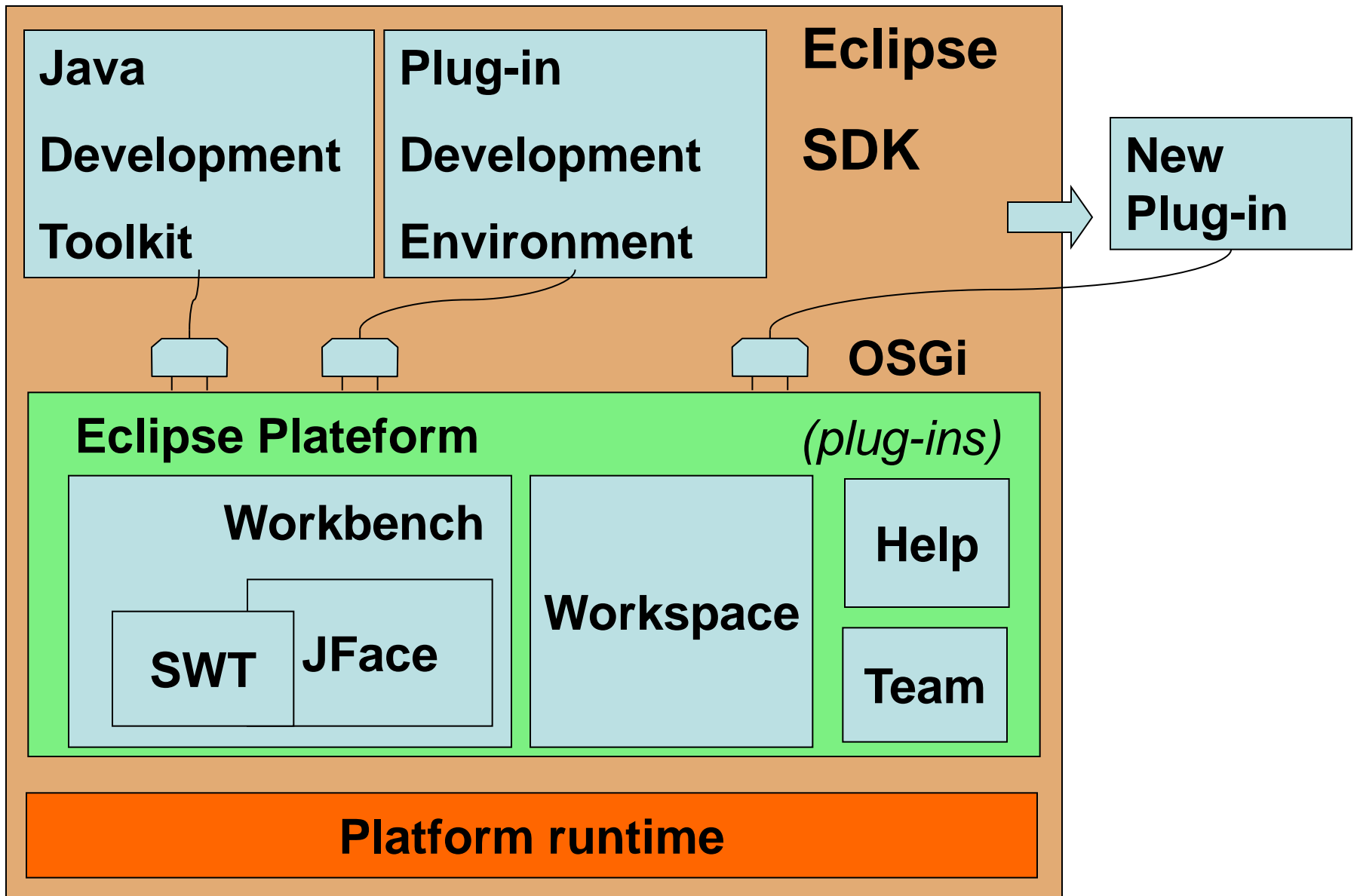
[Details](#)



Windows 32 Bit

Windows 64 Bit

Architecture d'Eclipse



Distribution des exemples

Répertoire de construction des exemples: E:\INDIGO

Workspace: E:\INDIGO\EMT\WORKSPACES\GenerationEditeurGraphique

Distribution: fichier « INDIGO.zip » à dézipper:

Installation.pdf

IntroductionEclipse.pdf

Installation.pdf

PrésentationEclipse.ppt (

REPERTOIRE PAR THEME

DOCUMENTATION

{ exemple fichier: *.pdf }

WORKSPACES (vide)

INSTALLATION (vide)

IDE - Environnement de développement intégré

IDE Eclipse regroupe un ensemble d'outils pour le développement de logiciel

Editeur de texte

- Coloration syntaxique
- Complétion
- Indentation

Compilateur

Débogueur

Gestion de projet

Gestion de développement en équipe

Help

....

IDE - Notre premier programme: Salut!

The screenshot displays the Eclipse IDE interface. The title bar reads "Java - monProjet/src/monPackage/maClasse.java - Eclipse". The menu bar includes "File", "Edit", "Source", "Refactor", "Navigate", "Search", "Project", "Run", "Window", and "Help". The toolbar contains various icons for file operations, running, and debugging. The Package Explorer on the left shows a project structure with "monProjet" containing "src", "monPackage", "maClasse.java", and "maClasse". The main editor window shows the following Java code:

```
package monPackage;

public class maClasse {
    public static void main(String[] args) {
        System.out.println("Salut!");
    }
}
```

The Console window at the bottom shows the output: "<terminated> maClasse [Java Application] C:\Program Files\Java\jre6\bin\javaw.exe (S Salut!". The status bar at the bottom indicates "Writable", "Smart Insert", and "3 : 24".

IDE – Programmation modèle Graphe

Graphe

liste Nœuds

liste d'arcs

Ma
solution!

Nœud

Nœud(nom, graphe)

...

Arc

Arc (NomOrigine, NomExtremite, graphe)

...

```
public class G1 extends Graphe {
```

```
    public G1() {
        new Noeud("n0", this);
        new Noeud("n1", this);
        new Noeud("n2", this);
        new Arc("n0", "n1", this);
        new Arc("n0", "n2", this);
    }
}
```

Console

<terminated> TestMonGraphe [Ja

```
*** graphe ***
    noeud: n0
    noeud: n1
    noeud: n2
    arc: ( n0 -> n1 )
    arc: ( n0 -> n2 )
```

PLUGINS – RCP - Construction

1. Choisir la perspective: Plug-in Développement

3. Créer un projet: Plug-in Project

Choix plugin / Rich Client Application

4. Ajouter extension

Plug-in
Manifest
Editor

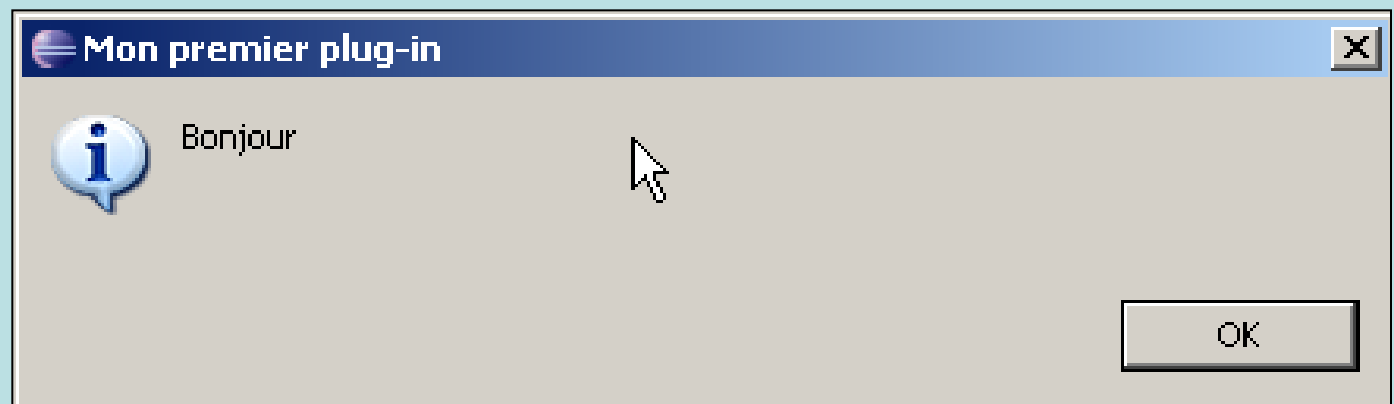
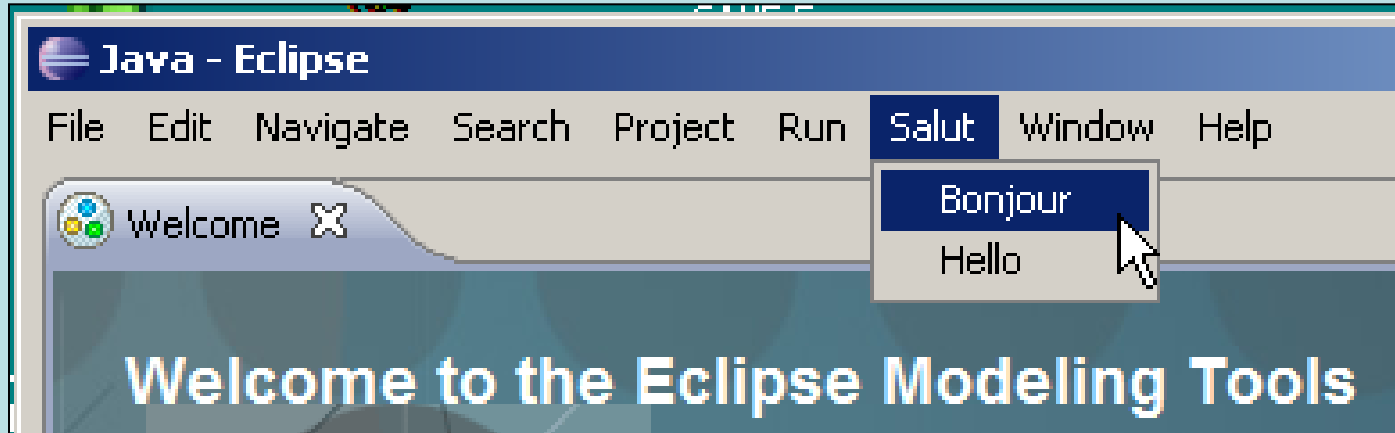
MANIFEST.MF
Plugin.xml

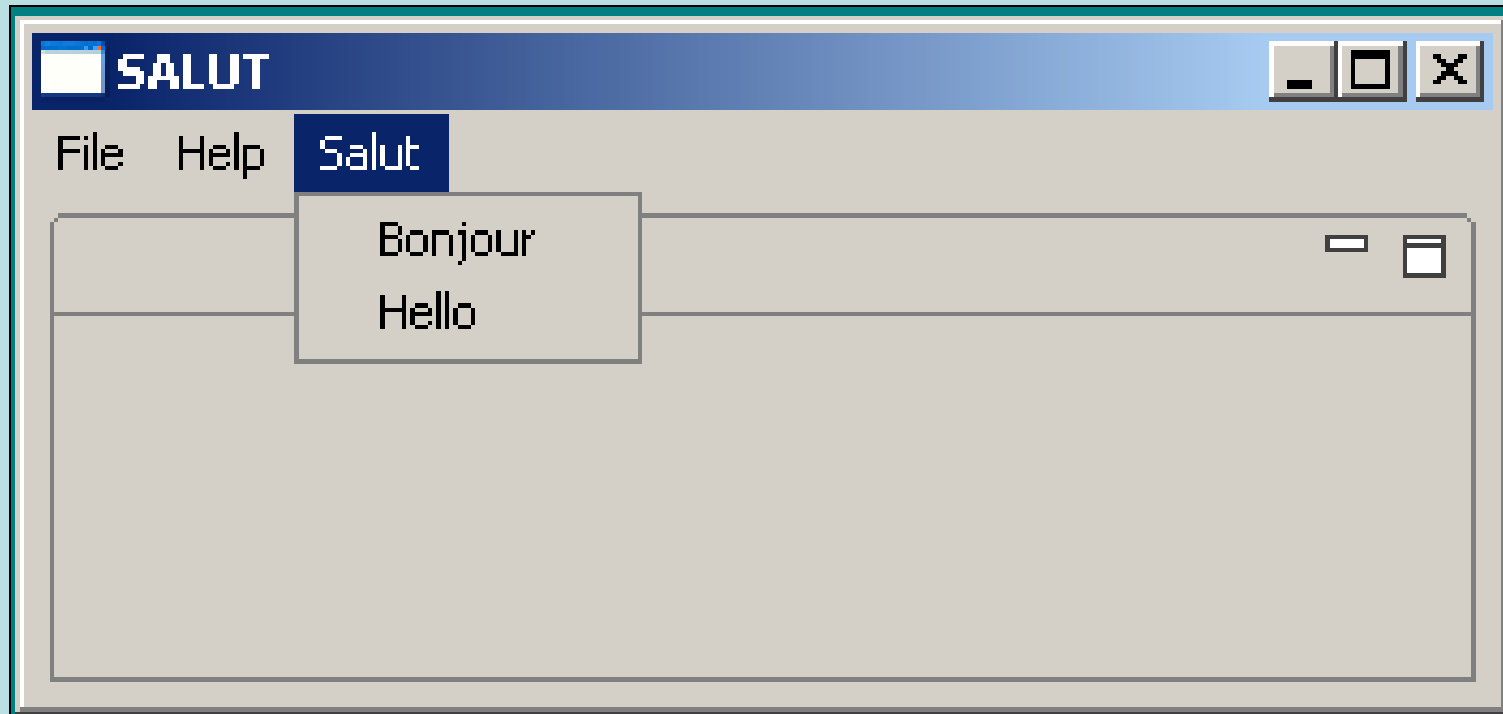
5. Test

OK

**6. Déploiement plug-in dans la plateforme Eclipse
/ déploiement de l'application**

PLUGINS RCP – Premier Plug-in





Utilisation Viewers JFace

Liste d

```
start
n1
n2
end
```

Table

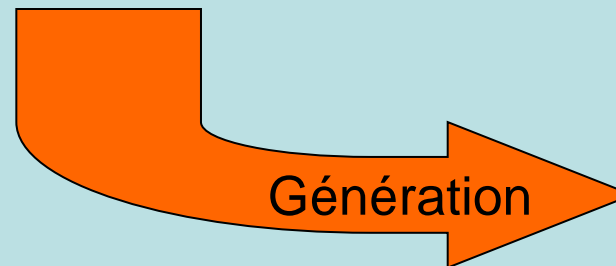
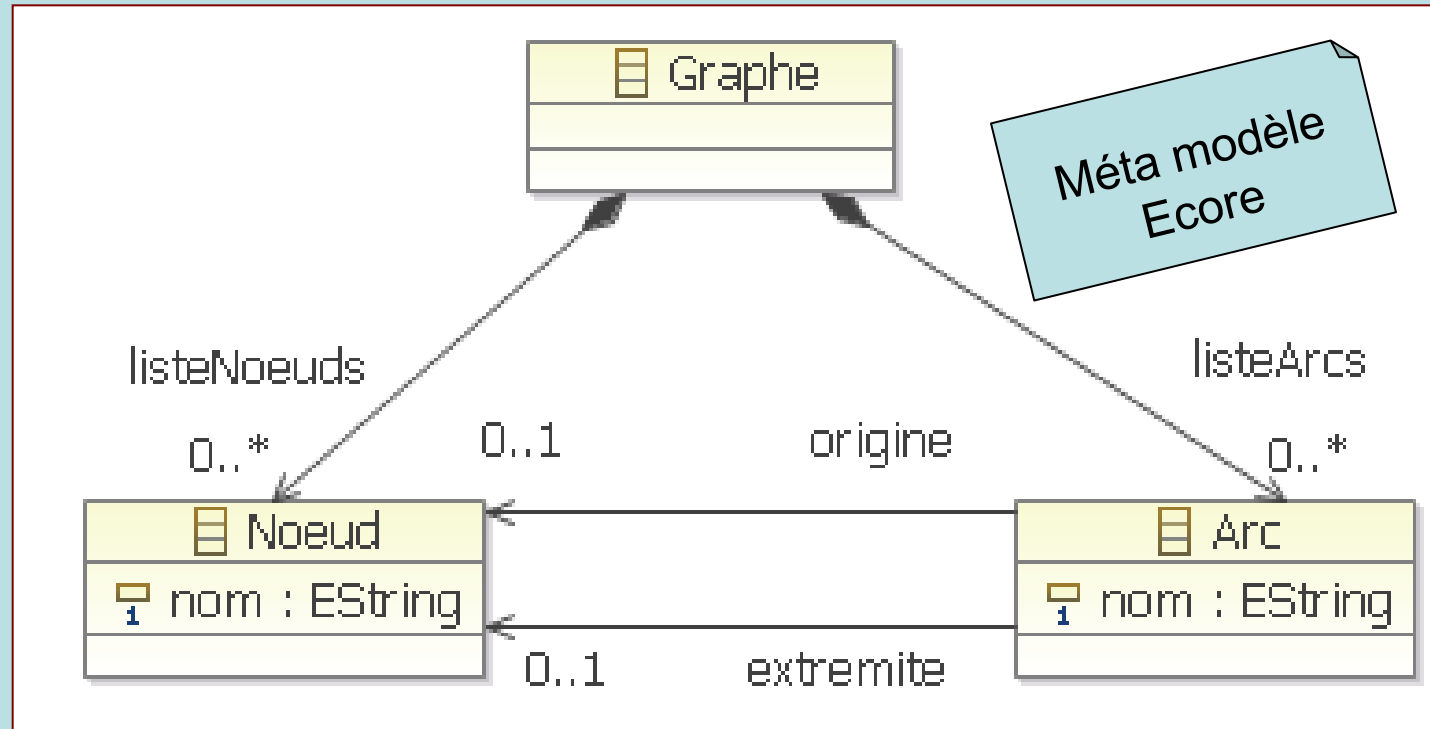
origine arc	extremite arc
start	n1
start	n2
n1	end
n2	end
end	start

Le graphe

```
*** graphe ***
noeud: start
noeud: n1
noeud: n2
noeud: end
arc: ( start -> n1 )
arc: ( start -> n2 )
arc: ( n1 -> end )
arc: ( n2 -> end )
arc: ( end -> start )
```

Arbre des

```
[-] start
  [-] n1
    [-] end
      [+] start
```



Code
modèle

```
GrapheFactory factory = new GrapheFactoryImpl();
graphe = factory.createGraphe();
Noeud n1 = factory.createNoeud();
n1.setNom("N1");
graphe.getListeNoeuds().add(n1);
Noeud n2 = factory.createNoeud();
n2.setNom("N2");
graphe.getListeNoeuds().add(n2);
Arc arc1 = factory.createArc();
arc1.setNom("ARC1");
arc1.setOrigine(n1);
arc1.setExtremite(n2);
graphe.getListeArcs().add(arc1);
```

Utilisation
Code modèle

EMT – Génération éditeur arborescent

The screenshot displays the EMT (Éditeur Modélisateur de Types) interface for editing a graph. The main window is titled "platform:/resource/testEditArborescent/G2.graphe".

Tree View: Shows a hierarchical structure:

- Graphe
 - Noeud N1
 - Noeud N2
 - Arc A1

 A blue callout box labeled "Editeur" points to the tree view.

View Model: A panel on the right displays the current model state:

- Fichier: G2.graphe
- Liste des noeuds: N1, N2
- Liste des arcs: A1(N2->N2)

 A green callout box labeled "Affichage modèle" points to this panel.

Properties Table: A table below the tree view shows the properties of the selected element (Arc A1):

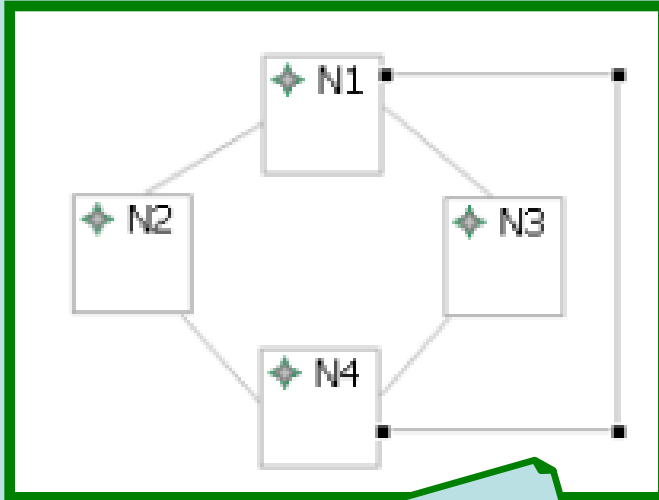
Property	Value
Extremite	◆ Noeud N2
Nom	☰ A1
Origine	◆ Noeud N2

Toolbar: At the bottom, a toolbar includes:

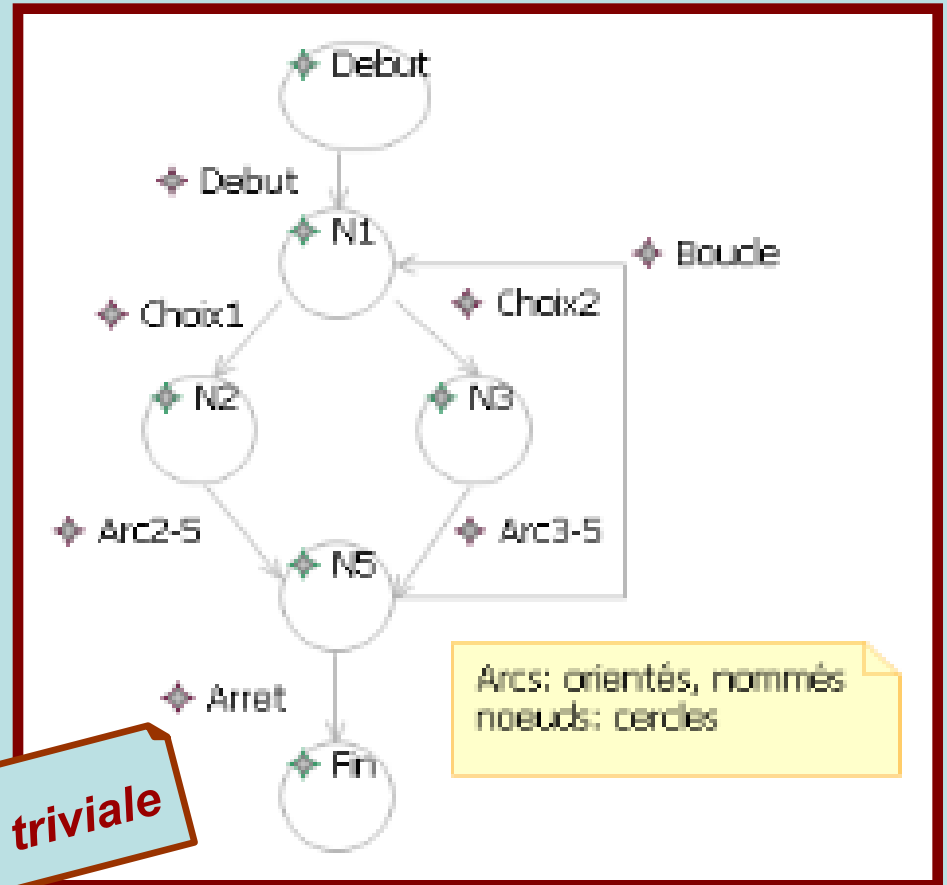
- Graphe (selected)
- New Child
- Undo (Ctrl+Z)
- Creation Noeud (highlighted with a red callout box)
- Noeud
- Arc

EMT – Génération éditeur graphique

Un tableau de bord nous guide dans la génération de l'éditeur (GMF Dashboard)



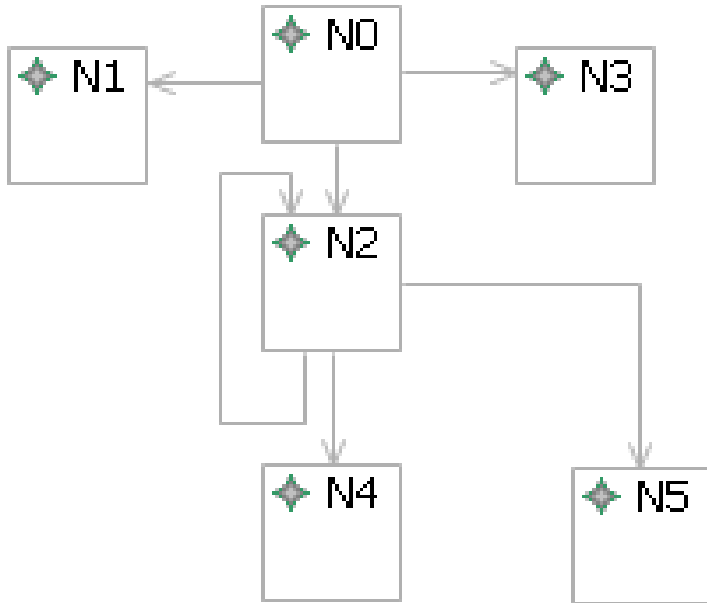
Génération simple



Arcs: orientés, nommés
nœuds: cercles

Génération non triviale

EMT – Expression de contrainte



Contrainte:

« Pour tout nœud du graphe le nombre d'arcs ayant ce nœud pour origine doit être inférieur ou égal à deux »

(Object Constraint Language)

Validation:

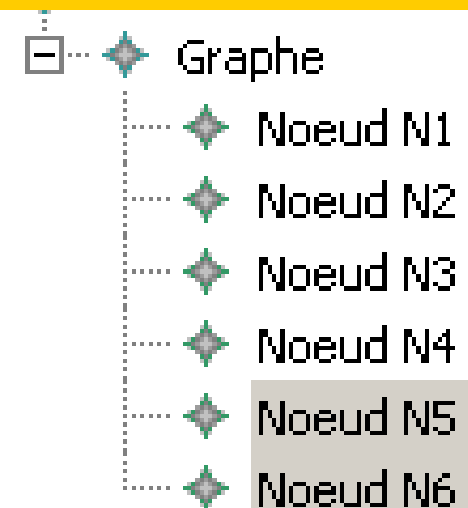
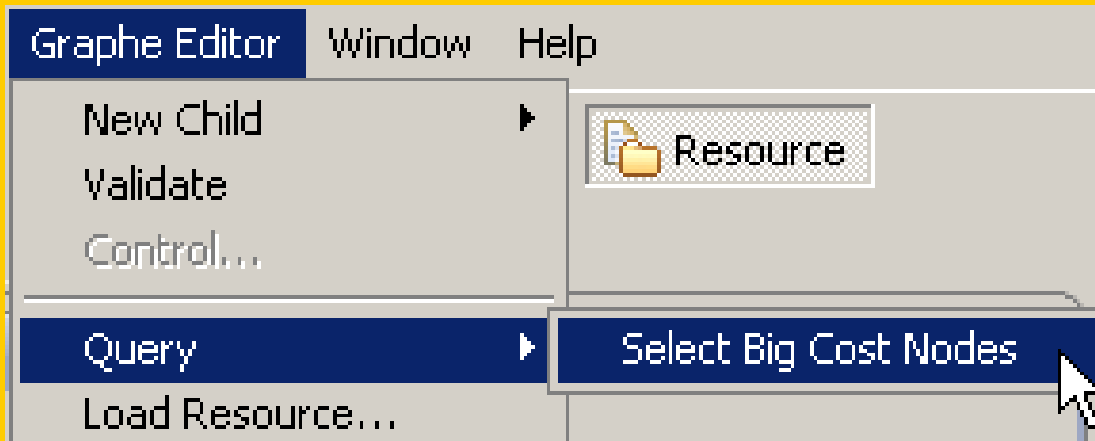
- ❌ The 'toManyOutput' constraint is violated on 'Noeud N2'
- ❌ The 'toManyOutput' constraint is violated on 'Noeud N0'

EMT – Requête sur le modèle édité

```
<graphe:Graphe . . . >
  <listeNoeuds nom="N1" cout="1"/>
  <listeNoeuds nom="N2" cout="2"/>
  <listeNoeuds nom="N3" cout="3"/>
  <listeNoeuds nom="N4" cout="4"/>
  <listeNoeuds nom="N5" cout="5"/>
  <listeNoeuds nom="N6" cout="6"/>
</graphe:Graphe>
```

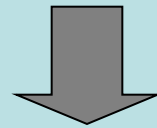
Query:

«Sélectionner tous les noeuds dont l'attribut "coût" est supérieur ou égal à 5 »



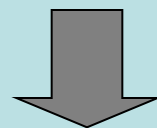
XText - Langage spécifique de description de graphe

Grammaire du langage spécifique



Génération

```
Noeud{nom N1}  
Noeud{nom N2}  
Arc{nom A1 origine N1 extremite N2}
```

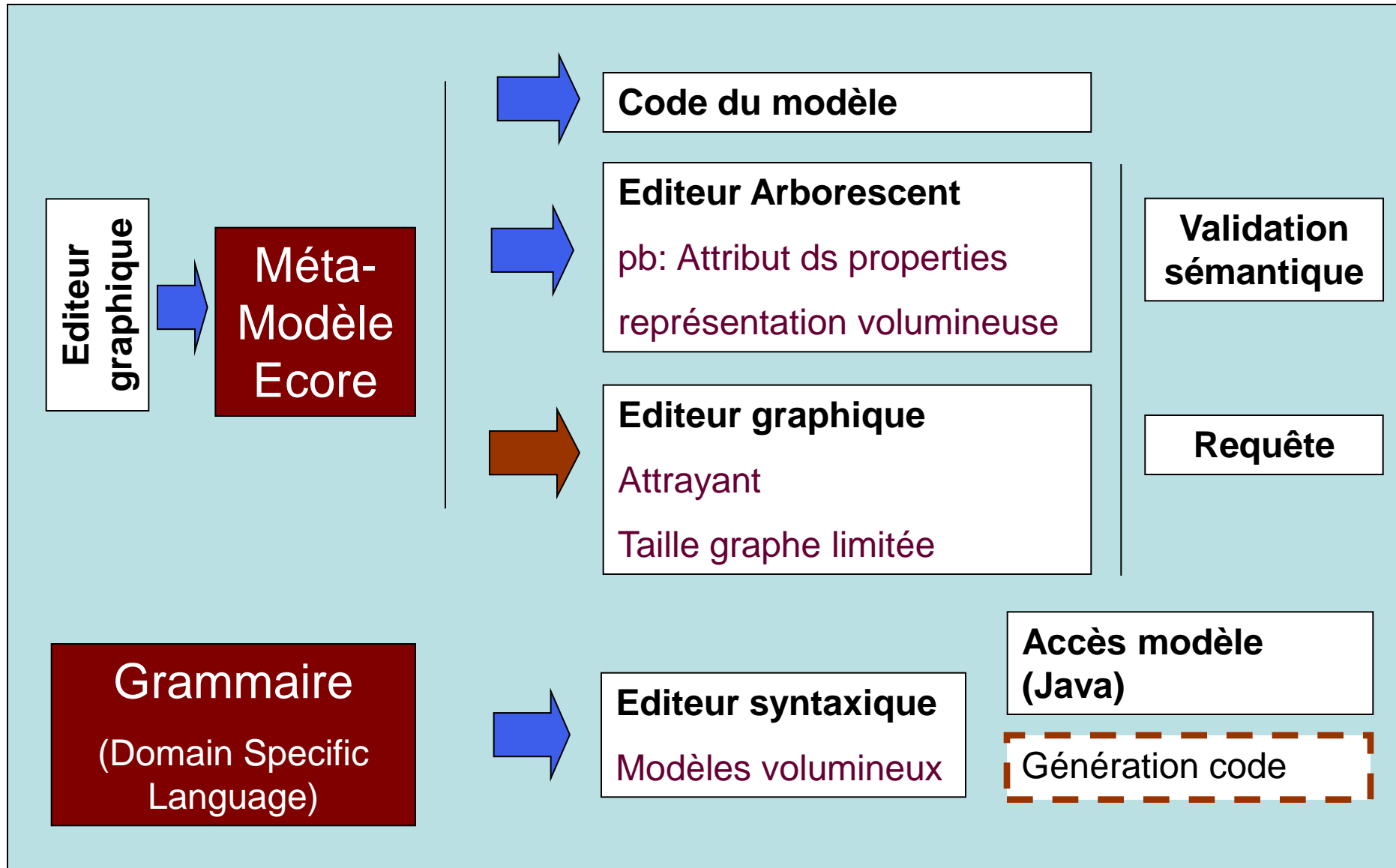


Accès au modèle édité



Modèle: /testEditionGraphe/simple.graphe
A1(N1 -> N2)

Conclusions



Ou trouver des informations?

Fenêtre Welcome

Help > Help Contents

Pour une extension donnée:

Dans le « plugin Manifest Editor », onglet « Extensions »:

- à la création de l'extension, bouton « Add... »
Le wizard « Extension Point Selection » fournit une description et éventuellement un « template »
- Bouton « [Show extension point description](#) »

The Eclipse Series: informit.com/series/eclipse

- eclipse Plug-ins (third Edition)
- eclipse Modeling Framework (Second Edition)
- eclipse Rich Client Platform (Second Edition)
- eclipse Modeling Project

<http://www.vogella.de/eclipse.html>