

# Projet HMIN 306: Extraction de workflow

Auteurs:

Loïc Ortole  
Mariam MAIGA

@git : [https://github.com/HighX97/HMIN306\\_ExtractAppWorkFlow](https://github.com/HighX97/HMIN306_ExtractAppWorkFlow)

# Sommaire

---

- ❖ Définitions
- ❖ Processus d'identification
- ❖ Solution

# Définitions

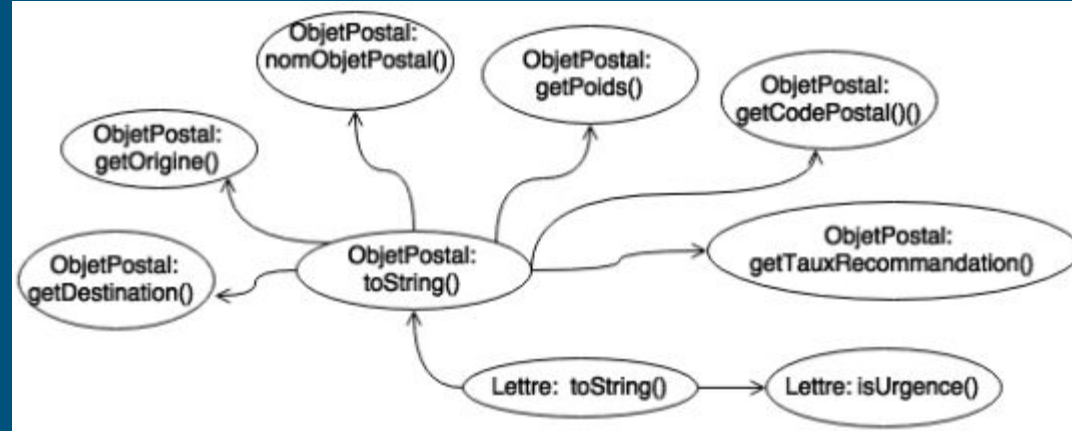
---

- ❖ Tâches
  - Tâche primitive
  - Tâche composite
  - Point d'entrée
- ❖ Flots
  - Flot de contrôle
  - Flot de données

# Définitions

## ❖ Tâches

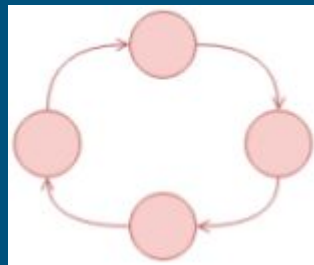
- Tâche primitive
- Tâche composite
- Point d'entrée



Une tâche représente la méthode correspondante au point d'entrée d'un sous-graphe connexe sans arcs sortants.

# Définitions

- ❖ Tâches
  - **Tâche primitive**
  - Tâche composite
  - Point d'entrée

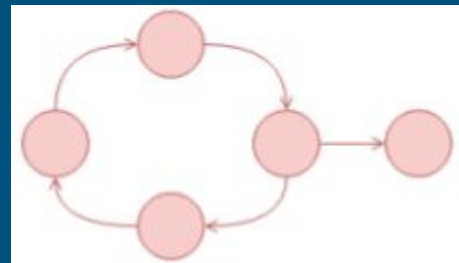


Une tâche primitive est la méthode correspondante au point d'entrée d' un sous-graphe minimal sans arcs sortants.

Donc, ce sous-graphe peut être un sommet sans arcs sortants ou bien un circuit. Par ailleurs, si un sommet d'un circuit possède un arc sortant vers un autre sommet qui ne fait pas partie du circuit, alors ce dernier n'est pas considéré comme une tâche primitive

# Définitions

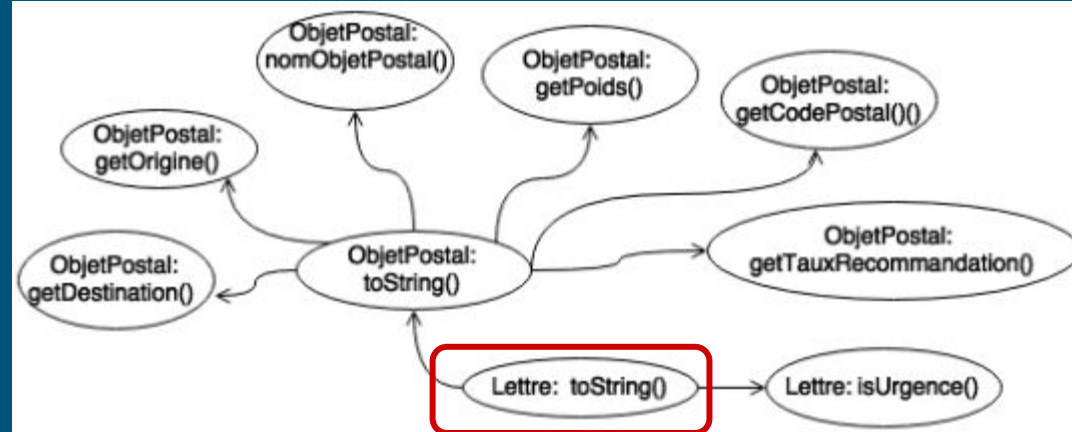
- ❖ Tâches
  - Tâche primitive
  - **Tâche composite**
  - Point d'entrée



Une tâche composite est une tâche qui peut inclure d'autres tâches composites ou primitives. Donc, c'est une méthode qui contient au moins une invocation de méthode.

# Définitions

- ❖ Tâches
  - Tâche primitive
  - Tâche composite
  - **Point d'entrée**

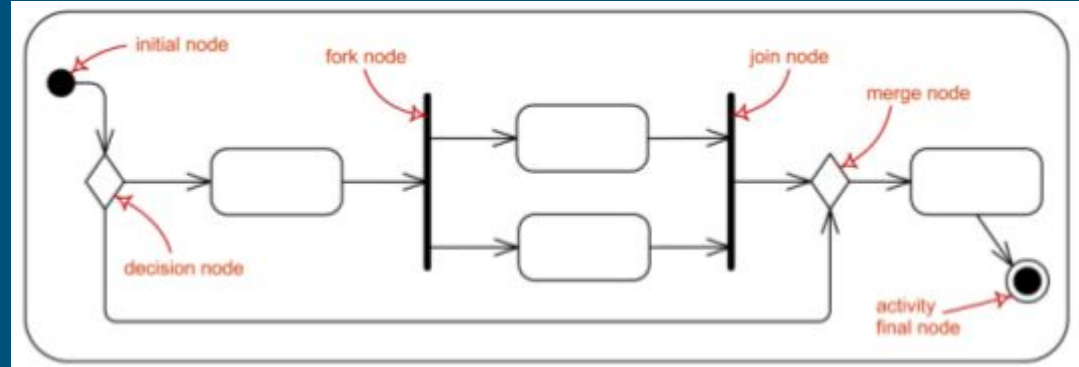


Un point d'entrée est un sommet sans arcs entrants.

Justification: C'est un sommet représentant une méthode qui n'est appelée par aucune autre méthode, c'est elle qui peut faire appel à d'autres méthodes.

# Définitions

- ❖ Flots
  - **Flot de contrôle**
  - Flot de données



Un flot de contrôle spécifie le mode d'assemblage de tâches dans un work-flow à travers l'utilisation de construction dites structure de contrôle, tel que la séquence, le branchement conditionnel, les boucles, le parallélisme ou la synchronisation.



# Définitions

- ❖ Flots
  - Flot de contrôle
  - **Flot de données**



Un flot de données spécifie les dépendances de données entre les tâches

# Processus d'identification

---

## ❖ Tâches primitives

- Entrées
- Sorties

## ❖ Tâches composites

- Sous tâches d'une tâche composite
- Flot d'une tâche composite

# Processus d'identification

## ❖ Tâches primitives

- **Entrées**
- **Sorties**

## ❖ Entrées d'une tâche

- Attributs static
- Objet receveur
- Paramètres



# Processus d'identification

- ❖ Tâches primitives
  - Entrées
  - **Sorties**



- ❖ Sorties d'une tâche
  - Donnée retournée
  - Entrées modifiées

# Processus d'identification

---

- ❖ Tâches composites
  - **Sous tâches d'une tâche composite**
  - Flot d'une tâche composite

Pour les tâches composites, il est possible de déterminer leurs sous-tâches qui sont les tâches qui représentent les méthodes correspondante au point d'entrée des composantes connexes du graphe obtenu

# Processus d'identification

- ❖ Tâches composites
  - Sous tâches d'une tâche composite
  - **Flot d'une tâche composite**

Pour déterminer le flot de contrôle d'une tâche composite, nous avons besoin d'analyser son code qui peut contenir:

1. Des instructions simples (affectations, déclarations de variable, instructions return ou instructions de contrôle (if, while et for) qui ne contiennent pas d'invocations de méthode
2. Des invocations de méthodes
3. Des instructions de contrôle (contiennent pas d'invocations de méthode).

# Solution

---

Création d'une application JAVA avec 5 packages :

- ❖ astvisitor
- ❖ draw
- ❖ graphe
- ❖ projet
- ❖ test

# Solution

Création d'une application JAVA avec 5 packages :

- ❖ **astvisitor**
- ❖ draw
- ❖ graphe
- ❖ projet
- ❖ test

Ensemble des classes permettant la gestion de l'analyse statique au travers de l'AST Visitor



# Solution

Création d'une application JAVA avec 5 packages :

- ❖ astvisitor
- ❖ **draw**
- ❖ graphe
- ❖ projet
- ❖ test

Exemple de création de graphe à partir d'un framework **JUNG**

# Solution

Création d'une application JAVA avec 5 packages :

- ❖ astvisitor
- ❖ draw
- ❖ **graphe**
- ❖ projet
- ❖ test

Ensemble de classe permettant la gestion de classe structure de données graphe.

# Solution

Création d'une application JAVA avec 5 packages :

- ❖ astvisitor
- ❖ draw
- ❖ graphe
- ❖ **projet**
- ❖ test

Coeur de l'application :

- ❖ Parsing
- ❖ GUI

# Solution

Création d'une application JAVA avec 5 packages :

- ❖ astvisitor
- ❖ draw
- ❖ graphe
- ❖ projet
- ❖ **test**

Projet test permettant la visualisation d'un graph simple