

## COMPTE-RENDU DE RÉUNION

**Réunion du :** Jeudi 12 février 2009

**Objet :** Projet Econet

**Participants :**

**M2 ALMA**

Matthieu Aubry

Tima Camara

Aurélia Couvrant

Jérémie Guidoux

Lydie Thierry

Mathieu Venisse

**Encadrants**

Pascal André

Gilles Ardourel

**Prochaine réunion :** Non définie

### Ordre du jour

Lors de cette réunion, nous sommes d'abord revenus sur la description de la « boîte à outils » dont nous disposons pour extraire les composants d'un projet Java. Enfin, MM. ANDRÉ et ARDOUREL nous ont exposé les différents chapitres à intégrer dans notre mémoire, permettant de définir les travaux préalables à effectuer avant de proposer nos solutions.

## Questions générales sur le projet Econet concernant :

- **Le processus d'extraction**

Nous avons reprécisé le cheminement de l'extraction de nos composants en détaillant certains des algorithmes dont nous disposons à appliquer à un projet Java.

- ◆ **JDT**

Cet algorithme nous permet d'obtenir du code Java annoté ainsi qu'un modèle à composants, pour l'instant primitifs (un seul niveau) à partir d'un code java « pur », i.e. non annoté.

- ◆ **Model from annotations**

Cette « boîte » permet de générer un modèle à composants en lisant les annotations d'un code Java.

- ◆ **Annotation writer from model**

À partir d'un modèle à composants, on obtient un code Java annoté.

- ◆ **Clustering tool**

Ceci permet d'extraire les composants composites en prenant en entrée le résultat de l'algorithme JDT, à savoir un code Java annoté et un modèle à composants.

Une ouverture sur cette partie est présentée dans le document test1.pdf accessible à l'adresse :

<http://www.lina.sciences.univ-nantes.fr/coloss/wiki/doku.php?id=econet:materials:start>

Cette approche consiste à utiliser UML pour obtenir un code Java annoté, mais ceci ne s'intègre pas dans le gros du travail à effectuer.

- ◆ **Distribution analyser**

Il est possible de parcourir des fichiers XML pour obtenir un modèle à composants.

- ◆ **Transformation** (pas détaillé)

- ◆ **Vérifications** (pas détaillé)

En d'autres termes, chaque algorithme prend en entrée une potentielle instance d'un modèle, un programme Java et des informations utilisateurs, permettant une interactivité pour, par exemple, préférer une solution plutôt qu'un autre.

Il est à noter que dans un tel système, à savoir un Système Basé sur des Règles (SBR), trois problèmes se posent à nous :

- définir un ordre d'application des règles logique, afin de ne pas inhiber les effets d'une règle antérieure;
- assurer la cohérence du système;
- assurer la complétude, i.e. couverture de l'ensemble des cas.

## Définition des chapitres du rapport

Il a ensuite été établi une ébauche du plan de notre mémoire. Voici les parties à intégrer au rapport final.

1. Econet et process B

2. JDT

3. Étude de l'existant

➤ plugin JDT

➤ règles déjà définies

➤ SBR pas encore défini

4. Proposition

Cette partie devra comporter l'organisation qui sera suivie dans l'élaboration du projet selon les connaissances acquises en Gestion de Projets : définition du rôles de chacun, planification...

5. Conception / implantation

6. Expérimentation CoCoME et analyse des résultats

Nous devons faire ici un bilan de notre expérimentation.

7. Thèse de Koschke

Nous devons également faire apparaître un chapitre Bibliographie dans notre mémoire. Ainsi, il nous a été proposé d'étudier la thèse de Koschke concernant notre problème.

L'élaboration de ce plan a permis de dégager le travail préalable à effectuer, tâches que nous avons délégué de la manière suivante :

- Econet et process B : Mathieu VÉNISSE;
- JDT et plugin JDT : Tima CAMARA;
- SBR : Aurélia COUVRAND;
- CoCoMe : Matthieu AUBRY;
- Thèse Koschke : Lydie THIERRY et Jérémie GUIDOUX.

Cette répartition peut être amenée à changer suivant le travail que certains points demanderont.