



UFR 919 Informatique – Master Informatique

Spécialité STL – UE 4I501 – DLP

## TME4 — Introduction d'un nouveau trait

Jacques Malenfant, Aurélien Moreau, Christian Queinnec

**Objectif :** Ajout du mot-clé `aMoinsQue`

### Buts

- Ajouter un nouveau mot-clé
- Mesurer les conséquences des choix d'implantation

Pour ce TME, vous pouvez choisir d'étendre ILP1 ou ILP2.

## 1 Ajouter la construction `aMoinsQue`

La construction `aMoinsQue` *condition corps* a pour sémantique d'évaluer la condition. Les instructions du corps ne sont évaluées (comme une séquence) que si la valeur de la condition est fausse. On trouve souvent cette construction, notamment en Perl, sous le nom de `unless`.

Si l'on considère le processus menant du fichier XML à l'AST final.

```
1 fichier XML -(1)-> DOM -(2)-> AST -(3)-> AST -(4)-> resultat
```

autre la technique suggérée dans le mémento agissant en (4),  $\Omega$   
<http://www-master.ufr-info-p6.jussieu.fr/site-annuel-courant/dlp>, il existe au moins trois autres façons d'opérer

La variante (1) est un hack moche. La variante (2) est plus propre mais tout comme la première variante, si l'on change de syntaxe d'entrée, il faudra la réécrire. La variante (3) est insensible aux changements de syntaxe et correspond plus aux techniques des compilateurs modernes (qui réécrivent les AST de façon à minimiser le nombre de mot-clés restant à considérer (notion de *core syntax*)).

### 1.1 Travail à réaliser

Vous commencerez par écrire la nouvelle grammaire, nommée `grammar2-tme4.rnc`, définissant `aMoinsQue`. Vous écrirez alors au moins deux programmes de tests (dont le nom sera suffixé par `-2tme4.xml` et une classe `ProcessTest` pour tester que votre extension d'ILP1 (resp. ILP2) passe toujours tous les tests d'ILP1 (resp. ILP2) et les nouveaux tests d'`aMoinsQue`.