

TP5 : Programmation Dynamique

5.1 Utilisation d'une méthode de hook

Nous voulons modifier la classe *Index* afin qu'elle puisse répondre à des méthodes de type *hasTermAndAnotherTerm*

Les règles étant :

- Les méthodes sont préfixées par *has*, suivi d'un terme arbitraire suivi éventuellement du mot clé *And* et un autre terme
- Le nombre de termes n'est pas limité
- Les méthodes renvoient *true* si l'index contient tous les termes compris dans le nom de la méthode

Pour ce faire on utilisera la méthode de hook *methodMissing*

5.2 Implémenter une méthode de *GroovyObject*

Toujours dans la classe *Index*, Implémenter la méthode *getProperty(String propertyName)* de telle sorte que si la propriété n'existe pas la méthode *hasPropertyName* est appelée

5.3 Intercepteurs et *ProxyMetaClass*

- Créer une classe implémentant *Interceptor*.
- Cette classe devra chaîner les 2 intercepteurs fournis par Groovy : *TracingInterceptor* et *BenchmarkInterceptor*.
- Appliquer cet intercepteur à la classe *Indexer* via un *ProxyMetaClass*
- Afficher les informations de ces intercepteurs

5.4 Meta-classe

Ajouter dynamiquement une méthode *export* à l'index qui retourne une String au format CSV de la map interne

5.5 Category

Utiliser une classe annotée pour effectuer la méthode *export* précédente