

Mini-Compte-Rendu

Auteurs: Cormier Levi / Tornambé Damien

Réponses aux deux questions posées

1/ Justifiez le choix de la stratégie de de mapping d'héritage.

```
@Inheritance(strategy = InheritanceType.SINGLE_TABLE)
@Entity
@DiscriminatorColumn(name = "person_type")
public abstract class Person {
    ...
}
```

Nous nous sommes tournés vers la stratégie de la SINGLE_TABLE car:

1. La super classe est abstraite.
2. Elle possède plus de champs que ses sous-classes :
 - La super classe Person possède 5 champs.
 - Les sous-classes User et Librarian ont respectivement 2 et 0 champs.
3. Cela simplifie énormément les requêtes polymorphiques.
4. La hiérarchies n'est pas complexe puisqu'on a ici qu'un seul niveau d'héritage.

2/ Critiquez le modèle de donnée, il contient quelques erreurs de conception.

On a noté plusieurs problèmes vis-à-vis de l'UML.

Tout d'abord, l'association "borrower", initialement noté en one to one, alors qu'un utilisateur est sensé en toute logique pouvoir cumuler plusieurs emprunts.

L'association en one to many entre les tables Borrow et Book qui n'a pas de sens, puisque dans une bibliothèque classique, il est très probable qu'un livre soit en plusieurs exemplaires. Et que par conséquent il puisse apparaître dans plusieurs emprunts. Cette association empêche même un livre d'être ré-emprunté une seconde fois.

Toutefois il y a de grande chance pour que les deux choses que nous venons de citer ne soient pas des erreurs de conception, car on obtenait des erreurs de compilation si on utilisait ces deux associations.

Excepté ça, il y a 3 choses en particulier qui nous dérangent, à savoir:

1/ Un libraire valide un emprunt mais a aucun moment on ne lui permet à lui aussi d'emprunter un livre en tant que libraire. Notamment avec le choix qui s'impose de la SINGLE_TABLE comme stratégie d'héritage,

on se retrouverait avec deux ID pour la même personne, un associé avec le type discriminant d'un user et l'autre avec celui d'un libraire pour qu'il puisse emprunter des livres.

2/ Les cardinalités pour les manager font que tout libraire possède un manager. Hors, si on prend le directeur de la librairie, personne n'est sensé le manager.

3/ Un auteur jusqu'à preuve du contraire est une personne. Du coup lors du premier TP, l'association entre Book et Author ne posait pas de problème. Mais maintenant il faudrait faire hériter Author de la table Person.