

GENIE INFORMATIQUE

INF4215 – Introduction à l'IA



**POLYTECHNIQUE
MONTREAL**

**LE GÉNIE
EN PREMIÈRE CLASSE**

Rapport de TP

LAB 2 – Base de Connaissance

Thomas Bille 1754864

Jean-Marie Robin 1753255

22/03/2015

Réflexion :

Nous avons fonctionné par propriétés. Chaque personne est dotée de propriétés.

4 propriétés obligatoires : vivant - genre - fictif et le métier. Ensuite des questions qui sont liées au métier pour mieux séparer les personnes.

Par exemple :

Eugénie Bouchard :

- vivante
- féminine
- réel
- sportif
- tennis

Le tennis est une propriété liée au sport.

Pour les objets, nous avons commencé par les répartir selon les matériaux qui les composent, puis selon l'endroit où on les trouve le plus souvent. Cela permet d'avoir une base commune pour tous les objets.

De la même manière que pour les personnes nous avons ensuite attribué des propriétés plus spécifiques en fonction des objets.

Notre projet est intéressant mais peut poser problème. Ceci pourrait venir de doublons qui auraient les mêmes propriétés qu'un autre. Ici nous nous basons principalement sur la liste donnée.

Une amélioration envisageable serait de rapprocher notre hiérarchie d'un arbre où chaque nœud n'a que deux fils. Dans notre cas, une question est posée pour chaque métier, ce n'est pas très efficace, et cela engendre un grand nombre de questions si le métier est situé en fin de liste. En limitant le nombre de membres de cette liste par une meilleure hiérarchie, on pourrait donc améliorer grandement la recherche.

Question 1 :

Dans cet exemple on veut forcer l'utilisateur à entrer des paramètres pour la clause "aime". Le programme acceptera toutes les possibilités sauf celles qui ont "mama_burger". Par exemple il est impossible de rentrer "aime(polo,X)" en question à la base de connaissance et la base répondra tout le temps "false" à la requête "aime(polo,b)".

Le coupe-choix permet d'empêcher l'analyse des clauses "aime", et le fail force l'échec, ce qui permet ici de faire comprendre que polo n'aime pas "mama_burger".

Question 2 :

Ceci pourrait servir par exemple à sécuriser une base de connaissance en donnant accès seulement aux connaissances connues. On pourrait imaginer une base de connaissance de mots de passe, où l'utilisateur ne pourrait pas demander tous les mots de passe existants, mais seulement ceux qu'il connaît.