

Modélisation, inférence, base de connaissances I

Exercices

1. Cette semaine nous allons étendre une ontologie similaire à celle de la dernière fois. Pour être sûr que tous partent du même point, chargez le fichier `5-MIBdC.owl` disponible sur le site web.
2. Nous allons tout d'abord ajouter des individus. À l'onglet **Individuals by class**, introduisez les individus **marie** et **jeanne** dans la classe **Femme**.
3. Pour indiquer maintenant qu'un couple d'individus satisfait un rôle, ajouter à **Object property assertions** le fait que **marie estUnDesParentsDe jeanne**.
4. Démarrez ou resynchronisez le raisonneur et vérifiez que **marie** est bien déterminée comme étant une **Mere** et le parent d'une fille.
5. D'un autre côté, **marie** n'est pas déterminée comme étant une **MereMonoparentale**. Pouvez-vous expliquer pourquoi ?
6. À l'onglet **DL Query** (en cochant uniquement **instances**), vous pouvez obtenir tous les individus d'une classe. Expérimentez avec les classes **Personne**, **Femme**, **Mere**, **GrandParent**.
7. Toujours à l'onglet **DL Query** vérifiez, grâce aux requêtes **not MereMonoparentale** et **MereMonoparentale**, que **marie** n'est dans aucune de ces deux classes.
8. Pour le moment, les seules choses que nous avons indiquées au sujet de la classe **MereMonoparentale**, c'est qu'il s'agit d'une sous-classe de **Femme** et de **Parent**. Donc, sauf si on indique qu'un individu est dans **MereMonoparentale**, il n'est pas possible de le déduire. Néanmoins, si un individu n'est pas dans l'une des classe **Femme** ou **Parent**, on aura par conséquence qu'il n'est pas non plus dans **MereMonoparentale**. Vérifiez-le en introduisant un **Homme**, **joseph**, en vérifiant que les classe **Homme** et **Femme** sont disjointes, et en cherchant les instances de **not MereMonoparentale** dans l'onglet **DL Query**.
9. Si on veut aller plus loin et définir une **MereMonoparentale** comme une **Mere** sans conjoint, il faut ajoutez un rôle **conjoint(x, y)** qui représente le fait que x est la/e conjoint/e de y . Ajoutez un tel rôle et modifiez la définition de **MereMonoparentale** pour tenir compte du rôle **conjoint**.
10. Ajoutez maintenant à **marie** le conjoint **joseph** et vérifiez que **marie** est bien dans la classe **not MereMonoparentale**.
11. Pour le moment, rien n'affirme que si **marie** a comme conjoint **joseph**, on a que **joseph** à **marie** comme conjointe ! Ceci est naturel, car un rôle n'est pas nécessairement symétrique. Pour vérifier que ceci est désagréable dans notre cas, introduisez

une classe **PereMonoparental** (et **Pere**, si nécessaire) en la définissant de façon analogue à la **MereMonoparentale** et vérifiez que **joseph** n'est pas dans la classe **not PereMonoparental**.

12. Indiquez maintenant, dans la définition du rôle **conjoint** qu'il s'agit d'une relation symétrique, donc que **conjoint**(x, y) et **conjoint**(y, x) sont équivalents. Vérifiez maintenant que **joseph** est bien dans la classe **not PereMonoparental**.

S'il vous reste du temps, vous pouvez considérer les questions suivantes.

13. Il est aussi possible d'indiquer qu'un rôle est l'inverse d'un autre. Ajoutez donc le rôle **estLEnfantDe** et faites-en l'inverse de **estUnDesParentsDe** (à droite **Inverse Of**).
14. Dans l'onglet **Individuals by class**, vous pouvez maintenant voir, par exemple, que **jeanne estLEnfantDe marie**, comme on s'y attendrait.
15. De plus, si vous définissez une classe **PetitEnfant** comme la sous-classe de **Personne** équivalente à l'expression

estLEnfantDe some (estLEnfantDe some Personne)

vous pouvez vérifier (dans l'onglet **DL Query**) que la classe

estLEnfantDe some (estLEnfantDe some GrandParent)

est équivalente à **PetitEnfant**.