

Ingénierie Des Connaissances

Création d'une ontologie liée au domaines des jeux

Fabien Brisset & Théo Chambon

2014 – 2015



**UNIVERSITÉ
DE MONTPELLIER**

Contexte Applicatif

Une application possible serait, par exemple, un logiciel ou un site internet, qui permettrait à une personne ou un groupe de personnes, de trouver un ensemble de jeux auxquels ils pourraient jouer, selon une liste de critères tels que le nombre de personnes, leur âge, le temps dont ils disposent et le lieu dans lequel ils se trouvent (s'ils sont dans une maison ou en plein air par exemple). Ces jeux peuvent être des jeux de sociétés, de cartes, des jeux vidéo, ou encore des jeux sportifs ou se pratiquant à l'extérieur.

Utilisateurs

Les utilisateurs potentiels sont très nombreux. En effet, n'importe quelle personne ayant envie de se divertir à l'aide d'un jeu, seule ou avec d'autres personnes, pourrait utiliser cette application afin de trouver des idées de jeu.

Utilisation

L'utilisateur peut effectuer une recherche d'un ensemble de jeux en entrant un certain nombre de critères liés aux propriétés du jeu, comme par exemple le nombre de jours, le temps à y passer, ou l'âge des participants, mais aussi des critères liés au contexte dans lequel pratiquer ce jeu, c'est à dire le lieu, par exemple s'il faut que le jeu se pratique à la maison, ou alors à l'extérieur, dans un terrain vague, ou dans un stade.

Les critères sont nombreux, et l'utilisateur peut ainsi affiner sa recherche, ou alors obtenir un ensemble plus large afin d'effectuer ensuite sa sélection lui même.

Concepts Centraux

Personne : Entité. A un âge. Peut être séparé en catégories (Enfants, Adultes, Seniors, ...).

Jeu : Entité. Possède un temps de jeu moyen, un nombre de participants minimum et maximum, se joue dans un lieu, possède un âge minimum et maximum et peut nécessiter un objet pour y jouer.

Lieu : Peut être Intérieur ou Extérieur.

Définition des Propriétés

Les propriétés peuvent être définies par des relations entre un domaine et un co-domaine. Le domaine est un objet et le co-domaine pourra être un second objet dans le cas d'une relation *Object Properties* ou une valeur d'un type primitif dans le cas d'une *Data Properties*.

Dans notre cas, une *Object Properties* sera par exemple le lien entre un Jeu et le Lieu où il se pratique, et les *Data Properties* seront l'ensemble des propriétés d'un jeu (nombres de jours, durée d'une partie, etc...).

Object Properties :

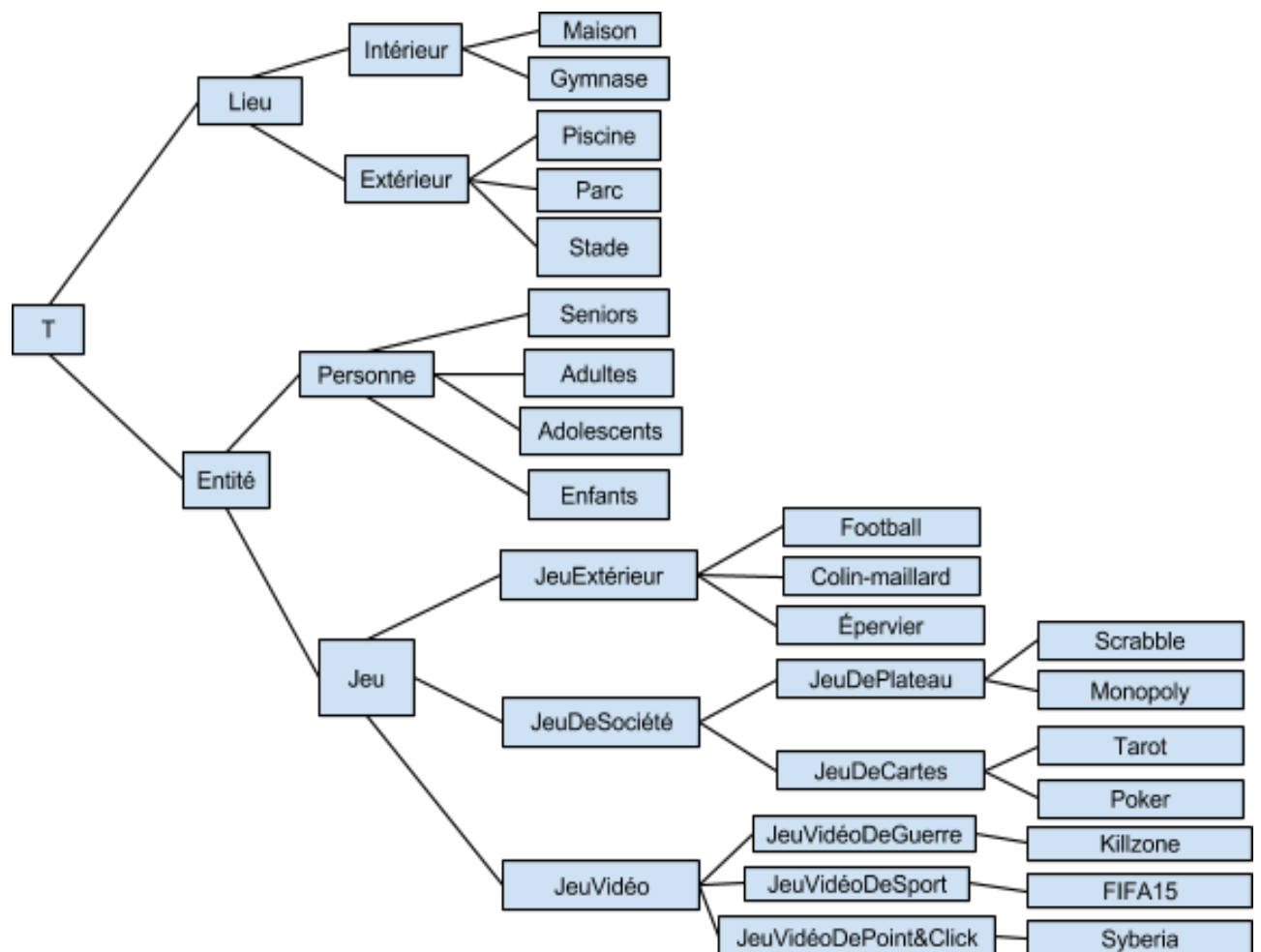
- Jeu hasLieu *Lieu*

Data Properties :

- **Personne** hasAge *Integer*
- **Jeu** hasTempsDeJeuMoyen *Integer*
- **Jeu** hasAgeMinimum *Integer*
- **Jeu** hasNombreMinPersonnes *Integer*
- **Jeu** hasNombreMaxPersonnes *Integer*
- **Jeu** hasLieu *Lieu*
- **JeuVidéo** hasAgeMinimum *Integer*
- **JeuVidéo** hasAgeMaximum *Integer*
- **JeuVidéo** hasPlateForme *String*
- **JeuExtérieur** hasObject *String*

Arborescence Prévisionnelle

Avant de modéliser notre ontologie à l'aide du logiciel Protégé, nous avons représenté l'ensemble de nos concepts, ainsi que certaines instances pouvant servir d'exemples, sous forme d'une arborescence qui nous a permis d'avoir une vision graphique de l'ensemble de nos concepts, et des taxonomies (par exemple les types de Jeux ou les types de Lieux).



Exemple de Requêtes Possibles

Voici quelques exemples de requêtes type, qu'un utilisateur pourra formuler afin d'obtenir l'ensemble de jeux désirés, en langage naturel.

- Nous sommes 4 personnes âgées de 30 à 40 ans, nous sommes en intérieur et avons 4 heures à dépenser, à quoi pouvons-nous jouer ?

Ce qui se traduit donc par : Tous les jeux dont le lieu est de une instance d'Intérieur, dont le nombre minimum de participants et inférieur ou égal à 4, et dont l'âge minimum inférieur ou égal à 30 ans, et dont la durée moyenne est de 4 heures.

Jeu and hasLieu some Interieur and hasNombreMinPersonnes < 4 and hasAgeMinimum < 30 and hasDuree = 4

- Nous sommes 2 personnes, à quoi pouvons-nous jouer ?

Ce qui se traduit donc par : Tous les jeux dont le nombre minimum de participants et inférieur ou égal à 2.

Jeu and hasNombreMinPersonnes <= 2

Il est possible de combiner l'ensemble des critères afin d'obtenir des requêtes de plus en plus spécifiques en fonction du type de jeu recherché. Ces critères sont représentés par les différentes propriétés des objets.

Définition de Classes Complexes

Les classes complexes vont nous permettre, par exemple, de représenter certaines catégories de personnes, comme les adultes, les enfants, les adolescents, et les seniors. Mais ces classes permettent aussi de représenter certaines requêtes typiques et récurrentes, afin de ne pas avoir à refaire la requête à chaque demande.

- **Senior : Personne** and hasAge > 60
- **Adulte : Personne** and hasAge (> 17, < 61)
- **Adolescent : Personne** and hasAge (> 11, < 18)
- **Enfant : Personne** and hasAge < 12
- **JeuDeCourteDurée : Jeu** and hasDureeMoyenne (< 2)
- **JeuAlaMaison : Jeu** and hasLieu Maison
- **JeuxPC : JeuVideo** and hasPlateForme ("PC")

On peut par exemple également imaginer une requête qui serait récurrente, par exemple les parents cherchant des activités à pratiquer à l'extérieur pour 3 enfants ou plus, âgés de 8 ans et plus. On pourrait donc représenter cette requête à l'aide de la classe complexe suivante :

JeuPleinAirMinPers3MinAge8 : **Jeu** and hasLieu Extérieur and
hasNombreMinPersonnes <= 3 and hasAgeMinimum <= 8

Exemples d'Instances

Voici quelques exemples d'instances des classes de cette ontologie.

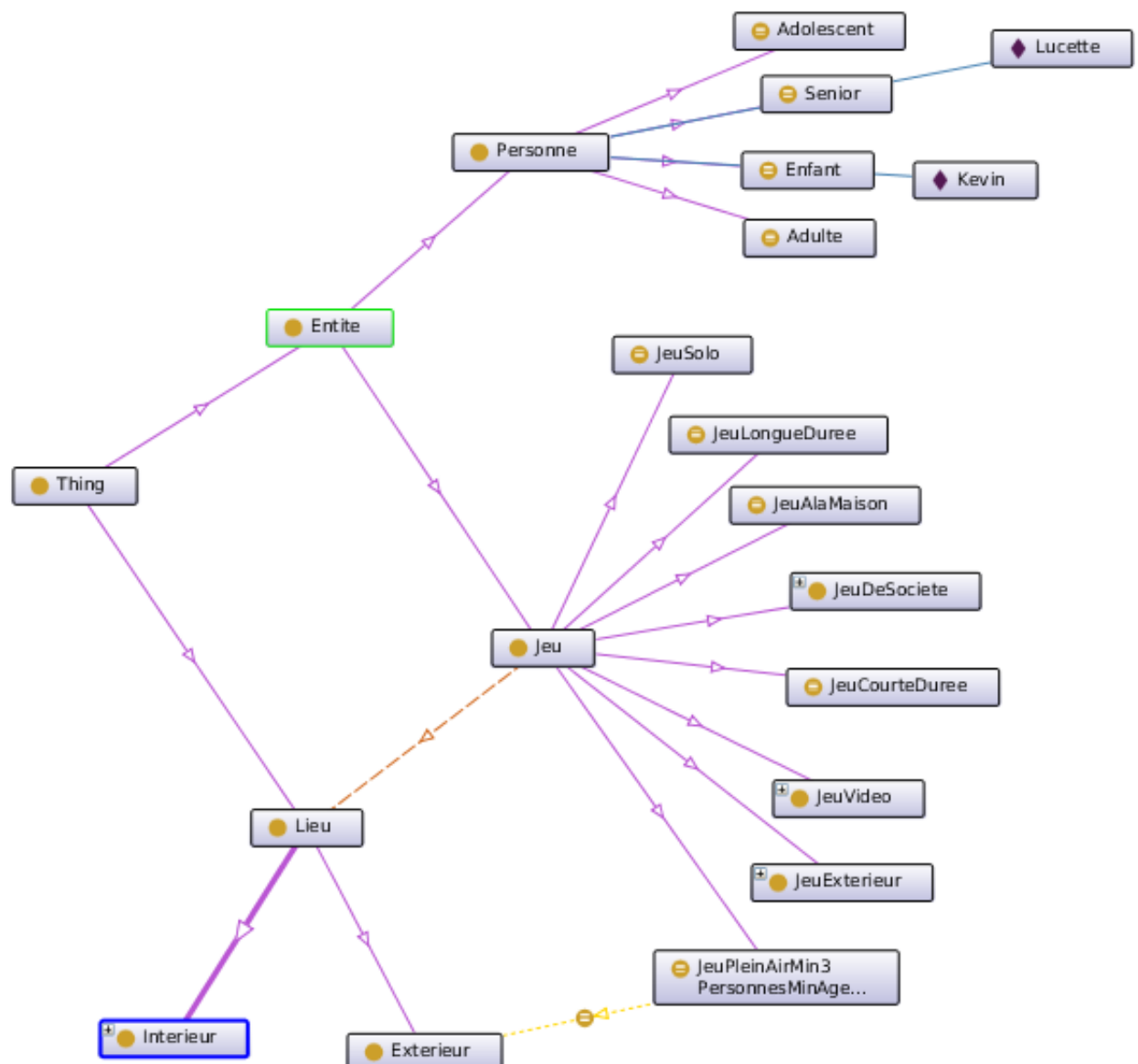
- Maison : Intérieur
- Stade : Extérieur
- Gymnase : Intérieur
- Killzone :
 - **JeuVidéoDeGuerre**
 - AgeMinimum = 16 ans, TempsDeJeuMoyen = 2 heures,
NombreMinPersonnes = 1, NombreMaxPersonnes = 2, PlateForme =
"PS2", Lieu = Maison
- Football :
 - **JeuExtérieur**
 - Objet = "Ballon", TempsDeJeuMoyen = 1, NombreMinPersonne = 2,
AgeMinimum = 5, hasAgeMaximum = 60, NombreMaxPersonnes = 22,
Lieu = Stade

Glossaire

- **Lieu** : Représente un endroit de l'espace ou une entité Jeu pourra se dérouler.
- **Intérieur** : Intérieur est un Lieu se situant entre des murs, par opposition à Extérieur.
- **Extérieur** : Lieu dont l'espace est ouvert, par opposition à Intérieur.
- **Entité** : Chose dont on ne sait rien mais qui existe.
- **Personne** : Être humain, individu, Entité humaine.
- **Enfant** : Jeune personne, âgée de moins de 12 ans
- **Adolescent** : Personne entre l'enfance de l'âge adulte, donc d'un âge situé entre 12 et 18 ans.
- **Adulte** : Personne ayant plus de 18 ans, et moins de 60 ans.
- **Senior** : Personne âgée, c'est à dire ayant plus de 60 ans.
- **Jeu** : Activité de loisirs d'ordre physique ou bien psychique à laquelle on s'adonne pour se divertir.
- **JeuExtérieur** : Jeu se déroulant en dehors du domicile.
- **JeuDeSociété** : Jeu se pratiquant à plusieurs personnes.
- **JeuDePlateau** : Jeu de société dont la zone de jeu est un plateau.
- **JeuDeCartes** : Jeu de société se jouant avec des cartes à jouer.
- **JeuVidéo** : Jeu électronique qui implique une interaction humaine avec une interface utilisateur dans le but de générer un retour visuel sur un dispositif vidéo.
- **JeuVidéoDeGuerre** : Jeu vidéo de tir basé sur la guerre entre plusieurs clans.
- **JeuVidéoDeSport** : Jeu vidéo permettant au joueur de s'adonner à un sport de façon virtuelle.
- **JeuVidéoDePoint&Click** : Jeu vidéo de réflexion et d'aventure dont les actions sont effectuées uniquement à l'aide de la souris.
- **JeuSolo** : Jeu se pratiquant uniquement en solitaire, c'est à dire un seul et unique participant.
- **JeuAlaMaison** : Jeu se pratiquant uniquement dans une maison.
- **JeuLongueDurée** : Jeu qui se pratique pendant une durée importante, fixée ici à plus de 4 heures.
- **JeuCourteDurée** : Jeu qui se pratique pendant une durée réduite, fixée ici inférieure à 2 heures consécutives.

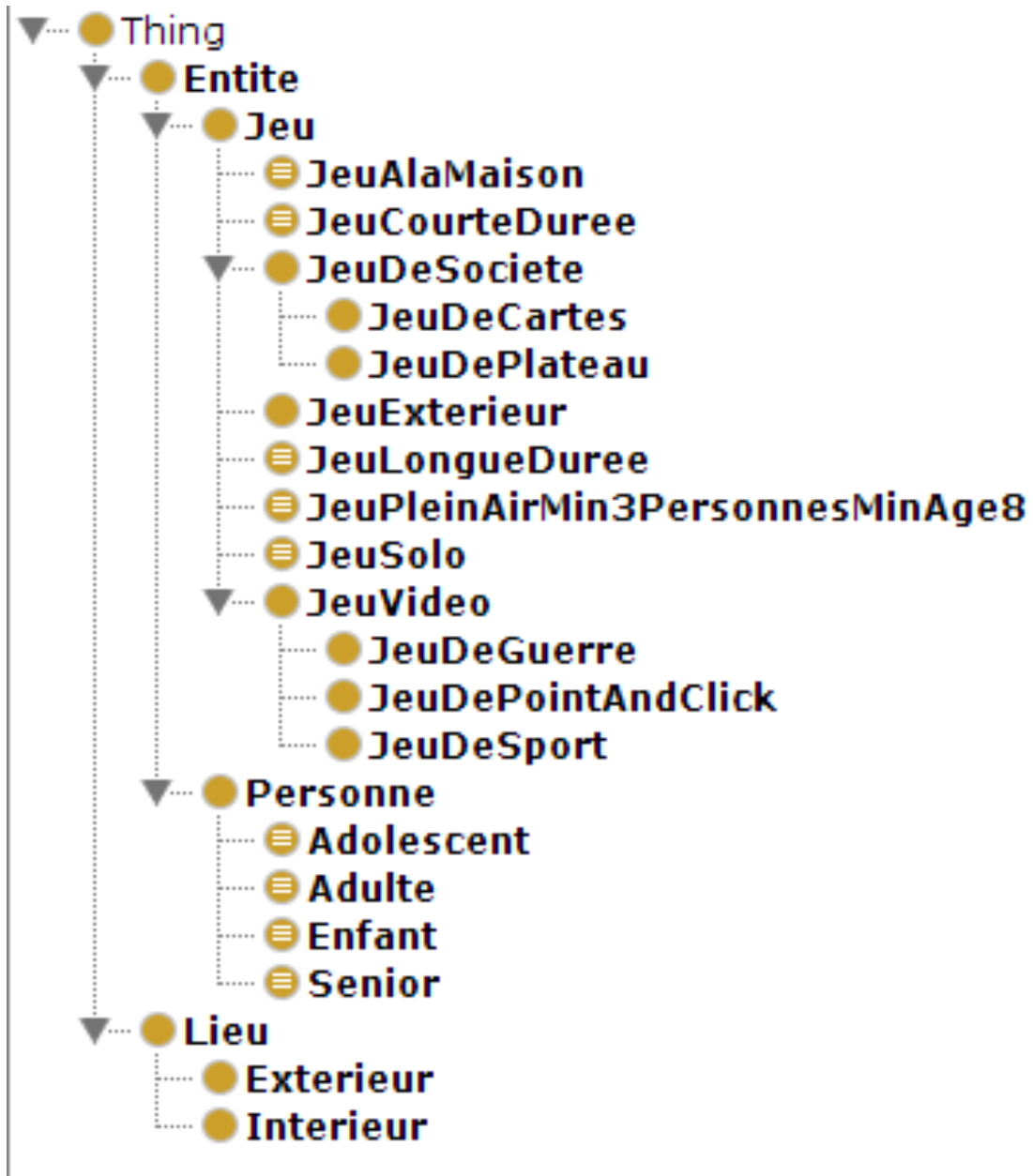
Arborescence Obtenue

Voici l'arborescence que nous avons obtenu, après avoir réalisé notre ontologie sur les jeux à l'aide du logiciel *Protégé*.



Représentation Classique des Classes

Voici la liste des classes obtenues dans le logiciel *Protégé*, sous forme de menu déroulant classique, où l'on peut voir les classes normales ainsi que les classes complexes créées.



Consistance

Afin d'éviter les inconsistances dans notre ontologie, certaines propriétés sont marquées comme fonctionnelles.

En effet, cela permet d'éviter des inconsistances comme par exemple un jeu qui aurait deux durées différentes ou encore une personne qui aurait deux âges différents, ce qui n'a pas de sens.

Cela permet également d'empêcher qu'un jeu ait certaines relations incompatibles, par exemple, un jeu ne pourra pas être dans la classe `JeuDeCourteDurée` et dans la classe `JeuDeLongueDurée`, car il n'a qu'une seule durée et ne peut donc être que dans une seule de ces deux classes.

La consistance de notre ontologie est donc respectée.

Certaines autres propriétés n'ont pas à être marquées comme fonctionnelles, c'est le cas par exemple la relation `JeuVidéo hasPlateForme (String)`. Un jeu vidéo pourra avoir plusieurs plates-formes distinctes (PC et PS3 par exemple).

Conclusion

Nous avons conçu une ontologie en suivant la methontologie, qui répond convenablement aux objectifs que nous nous étions fixés, elle est facilement évolutive et répond aux besoins initiaux.

Nous avons suivis notre arborescence prévisionnelle à quelques détails près ce qui montre que notre analyse prévisionnelle fut bonne et en adéquation avec nos besoins.

Cette ontologie permet donc correctement d'exécuter des requêtes permettant d'obtenir une liste de jeux adaptés en fonctions de certains critères définis.