



TP Construction d'Ontologies

Guohao DAI
November 12, 2022

1. Construction et peuplement guidés de la hiérarchie formelle

In this part of ontology modeling and its statement, we have accomplished several things as bellowed:

- Added entity classes and their corresponding properties as required.
- The *Reasoner* can only infer the relationship of objects in a forward direction, but not in a reverse direction. For example, under normal description, Paris is **locatedIn** France, and France **contains** Paris. Because we don't have an object property that is the inverse of **locatedIn**.
- We can add an example for the *Movie* class and add property such as **takePlaceIn**, **isDirectedBy**, **hasNationality**, **asActor**, etc.
- *Reasoner* can distinguish that *Bérénice Bejo* (female) is an *Actress*, and *Jean Dujardin* (male) is an *Actor*.

2. Construction et peuplement guidés de l'ontologie lourde

We changed and enriched the ontology as needed. After running reasoner, we have accomplished several things as bellowed:

- Added entity classes and their corresponding properties as required.
- Since we added transitive **contains**, the *Reasoner* knows that if a place A is **locatedIn** a place B and this place B is **locatedIn** a place C, then the place A is **locatedIn** the place C. In addition, we also added the properties **locatedIn** and **contains** are *inverse* to each other.
 - ✧ Thus, after *Reasoner* inference, *Paris* is **locatedIn** *France*, but also in *Europe*.
 - ✧ In addition, *Midnight in Paris* **takePlaceIn** *Paris*, in *France*, and in *Europe*.
- We set the *Country* object property **hasCapital** to "hasCapitale exactly 1 City", so if we want to define *Paris* and *Ville Lumière* as the capital of France, we can choose *Ville Lumière* as the **label** for *Paris*.
 - ✧ Thus, after *Reasoner* inference, *France* is a *Country* **locatedIn** *Europe* and **hasCapital** *Paris*. Furthermore, it **contains** *Paris*.

- ✧ Monaco is both a city and a country, which is impossible in our model. Because the classes **Country** and **City** are disjoint, a **Country** cannot be a **City**.
- The *Reasoner* infers that the movie *The Artist* is a *Feature Film* and a *European Film*. In addition, the *Reasoner* infers that *Bérénice Bejo* and *Jean Dujardin* are **asActor** in *The Artist*.
 - ✧ *Reasoner* can differentiate between *Short Films* (*Duration* < 59) and *Feature Films* (*Duration* > 59) based on the *Duration* of the *Movie* instance.
 - ✧ The *Reasoner* can infer whether a *Movie* is *European Movie* based on whether **hasNationality** is in Europe (like *French*).
 - ✧ The *Reasoner* can infer the actor who **playsInMovie** based on the property **asActor**. This property is inverse of **playsInMovie**.
- We set the instance of movie *Midnight in Paris* to **takePlaceIn** Paris, the *Reasoner* can infer from this that the movie also **takePlaceIn** France and Europe.
 - ✧ Because of the property **takePlaceIn** that we specified “*SuperProperty Of (Chain)*” as the following: “*takePlaceIn o locatedIn SuperProperty Of :takePlaceIn*”.