

Pokračujeme s Protégé

Budeme pracovat s již hotovou ontologií, např. Pizza Ontology (můžete ale zkusit i jiné ontologie).

Otevřete Wine ontology pomocí File/Open from URL:

<https://www.w3.org/TR/owl-guide/wine.rdf>

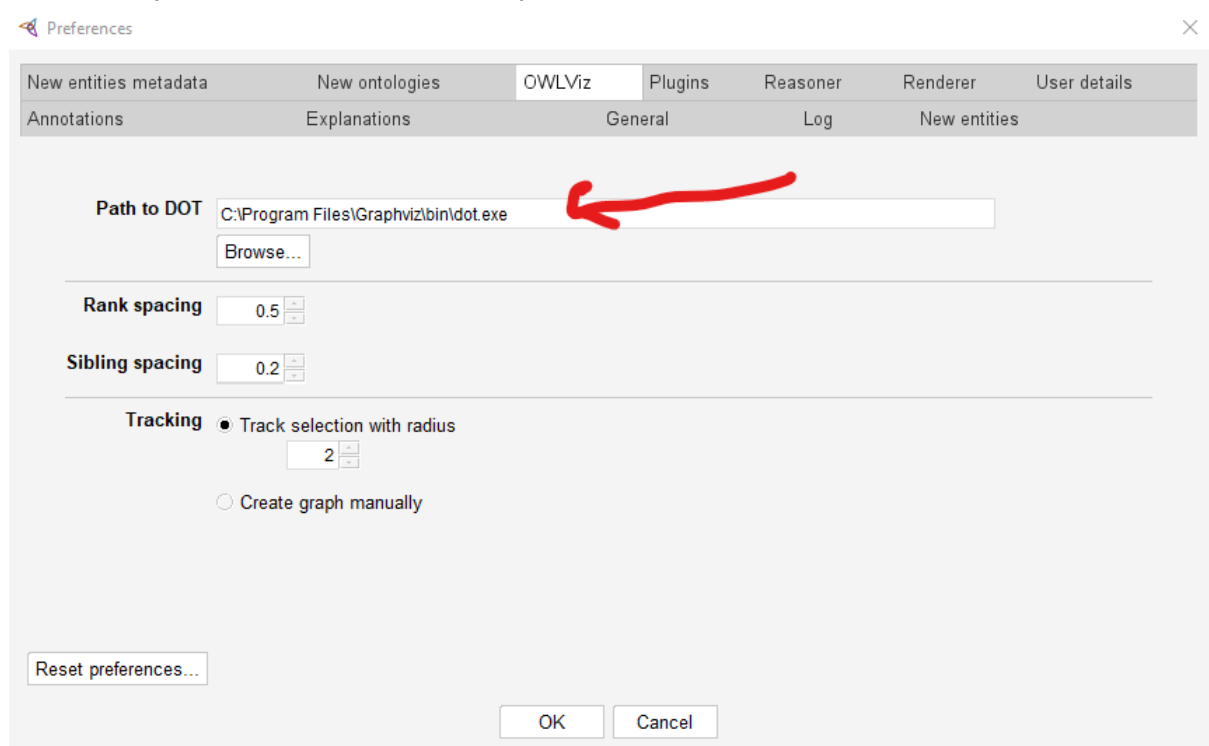
Prohlédněte si ontologii v Protégé a seznamte se s ontologií.

Rozšířené funkce Protégé nejsou na první pohled vidět. Je třeba připnout na hlavní panel tab s aplikací: Window/Tabs - zde vyberte, kterou aplikaci chcete vidět na hlavním panelu.

Vizualizace

Protégé nabízí nejméně dvě možnosti vizualizace: OWLViz a OntoGraf.

Pro OWLViz je nutné mít nainstalovaný GraphViz z <https://graphviz.org/download/>. Dále je potřeba, aby binární soubor GraphViz byl správně nastaven v File/Preferences:

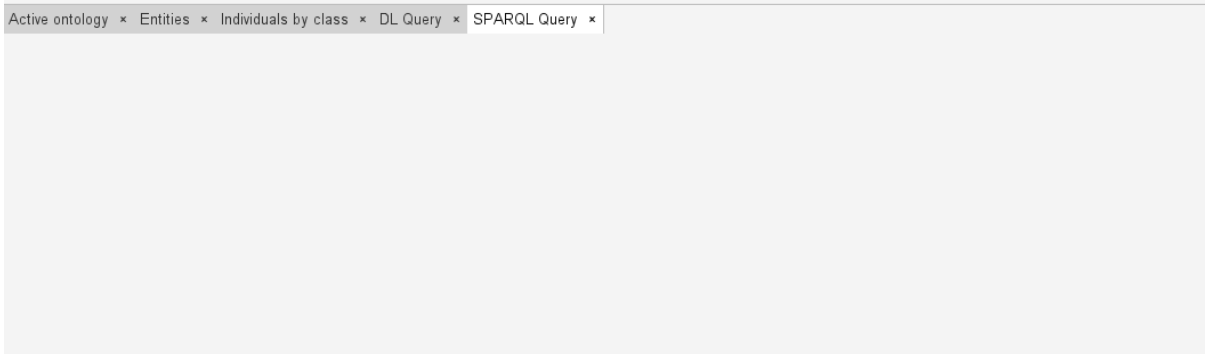


V OntoGraf se může stát, že neuvidíte nic. Pak je potřeba přidat do tabu OntoGraf View (Window/Views/Class views/OntoGraf)

Vyzkoušejte aspoň jednu z vizualizací.

SPARQL

Se SPARQL se v Protégé nepracuje zrovna dobře. Nicméně, i tak si tuto možnost ukážeme. Aktivujte SPARQL Query na hlavním panelu (výběrem z Window/Tabs). Zobrazí se prázdný obsah:



Aby fungovalo okno pro dotazy, přidejte Window/Views/Query Views/Snap SPARQL Query. Pokud se Snap SPARQL nezobrazuje v nabídce, doinstalujte jej přes File/Check for plugins. Pozor, obyčejný View “SPARQL Query” bohužel nefunguje.

Musí být zapnutý reasoner.

Vyzkoušejte dotazy:

PREFIX owl: <<http://www.w3.org/2002/07/owl#>>

PREFIX rdf: <<http://www.w3.org/1999/02/22-rdf-syntax-ns#>>

PREFIX rdfs: <<http://www.w3.org/2000/01/rdf-schema#>>

PREFIX wine: <<http://www.w3.org/TR/2003/PR-owl-guide-20031209/wine#>>

```
SELECT ?wine WHERE {
    ?wine rdf:type ?class.
    ?class rdfs:subClassOf wine:Wine
}
```

```
SELECT ?wine ?region WHERE {
    ?wine rdf:type ?class.
    ?class rdfs:subClassOf wine:Wine .
    ?wine wine:locatedIn ?region .
}
```

SWRL

Nevýhodou použití SWRL v Protégé je, že když pustíme inferenci, může ontologie snadno narůst do obřích rozměrů. Protégé se tím výrazně zpomalí. Když se to stane, je nejjednodušší Protégé restartovat.

Podívejte se, jak velká ontologie Wine je (kolik axiomů, kolik subClassOf).

Vinná réva je ovoce

Najděte libovolnou instance vinné révy (např. CabernetFrancGrape) a prohlédněte si ji.

Přidejte pravidlo, které řekne, že vinná réva (WineGrape) je ovoce (Fruit).
WineGrape(?x) -> Fruit(?x)

Zapněte reasoner (Pellet). Stiskněte postupně všechna tlačítka.

The screenshot shows the Protégé interface with a rule defined: `vin:WineGrape(?x) -> food:Fruit(?x)`. Below the rule, the 'Inferred Axioms' tab is active, displaying the following text:

```
Number of OWL individual declarations exported to rule engine: 130  
Number of OWL individual declarations exported to rule engine: 206  
Number of OWL object property declarations exported to rule engine: 16  
Number of OWL data property declarations exported to rule engine: 1  
Total number of OWL axioms exported to rule engine: 1212  
The transfer took 22 millisecond(s).  
Press the 'Run Drools' button to run the rule engine.  
Successful execution of rule engine.  
Number of inferred axioms: 6590  
The process took 908 millisecond(s).  
Look at the 'Inferred Axioms' tab to see the inferred axioms.  
Press the 'Drools->OWL' button to translate the inferred axioms to OWL knowledge.  
Successfully transferred inferred axioms to OWL model.  
The process took 3970 millisecond(s).
```

At the bottom of the interface, three buttons are visible: 'OWL+SWRL->Drools', 'Run Drools', and 'Drools->OWL'. Red handwritten numbers 1, 2, and 3 are placed above these buttons, indicating the sequence of actions to be performed.

Podívejte se zpět na instanci vinné révy. Můžete zkusit i Explain inference (otazník).

Podívejte se, jak ontologie narostla.

Lehká vinná réva je vinná réva

Přidejte podtřídu třídy WineGrape a pojmenujte ji LightWineGrape.

Přidejte pravidlo, že když se z révy dělá lehké víno, pojmenujeme ji lehká.

`vin:WineGrape(?x) ^ vin:madeIntoWine(?x, ?y) ^ vin:hasBody(?y, vin:Light) -> vin:LightWineGrape(?x)`

Spusťte inferenci, synchronizujte reasoner.

Otestujte, zda CabernetFranc je lehká réva. Co Merlot?

Znovu se podívejte, jak ontologie narostla.

Jak staré je víno?

SWRL má vestavěné funkce (builtIn). Zkuste spočítat, jak staré je víno SaucelitoCanyonZinfandel1998.

`hasVintageYear(?x, ?y) ^ vin:yearValue(?y, ?z) ^ swrlb:subtract(?a, 2023, ?z) -> age(?x, ?a)`

Spusťte inferenci, synchronizujte reasoner. Podívejte se, jestli má SaucelitoCanyonZinfandel1998 vlastnost age (v Property assertions).

