Hands-on session: Protégé

1 Manchester Syntax

Protégé uses OWL and Manchester syntax. Recall that description logic concepts are *classes* and roles are *object properties*. In addition to DL, OWL has *data properties* to connect individuals to data values such as integers, strings, etc., and *annotation properties* that are used to annotate ontologies.

OWL expressions in Manchester syntax	DL counterparts
owl : Thing	$\top$
owl : Nothing	$\bot$
C_1 and C_2	$C_1 \sqcap C_2$
C_1 or C_2	$C_1 \sqcup C_2$
not C	$\neg C$
R some C	$\exists R.C$
R min n C	$\geq nR.C$
R max n C	$\leq nR.C$
R only C	$\forall R.C$
R value a	$\exists R.\{a\}$
inverse R	R^-
C_1 SubClassOf : C_2	$C_1 \sqsubseteq C_2$
C_1 EquivalentTo : C_1	$C_1 \equiv C_2$
C_1 DisjointWith : C_2	$C_1 \sqsubseteq \neg C_2$
R_1 SubObjectPropertyOf : R_2	$R_1 \sqsubseteq R_2$
R_1 InverseOf : R_2	$R_1 \equiv R_2^-$
R Domain : C	$\exists R.\top \sqsubseteq C$
a Types : C	$C(a)$
a Facts : R b	$R(a, b)$
...	

2 Getting Started with the Pizza Ontology

- Download the Pizza ontology:
  `http://protege.stanford.edu/ontologies/pizza/pizza.owl`
- Open it with Protégé (File > Open).
- Identify concepts (in view Entities > Classes) and roles (in view Entities > Object properties).
- Identify the definitions of complex concepts and the relationships between concepts and roles (subconcepts, disjoint concepts, subroles...). Translate some of them into DL syntax. In particular:
  - Write the definition of ThinAndCrispyPizza and CheesyPizza.
  - Is PrinceCarlo a ThinAndCrispyPizza? CheesyPizza? MeatyPizza? InterestingPizza?
  - Is PepperTopping disjoint with FruitTopping?
  - What are the domain and range of hasBase?
  - What is the inverse property of hasIngredient?
  - Which roles are functional? transitive?

- Select and start reasoner (Reasoner > HermiT, Reasoner > Start reasoner). Browse the class hierarchy and observe what new information is now present (indicated by light yellow background).
- In class hierarchy view, change Asserted to Inferred and check how the hierarchy changes.
- Check that your answers to the previous questions are correct.
- Check some inferences explanations by clicking on the "?" next to the inferred relationship.
- Find unsatisfiable concepts, which are subclasses of owl : Nothing, and use the explanations of their equivalence to owl : Nothing to understand what is the modelisation problem.

3 Creating an Ontology

- Create an ontology (File > New) with IRI "http://small-onto" and save it (File > Save as...) in RDF/XML syntax.
- Express the following statements as DL axioms, then add them to your ontology in Protégé.
  - Mammals are animals that produce milk
  - Cats, cows, pigs and platypus are mammals
  - Birds are animals that do not produce milk
  - Birds and platypus produce eggs
  - All animals eat something
  - Cows eat only plants
  - Cats and platypus are carnivorous
  - Pigs eat both plants and meat
  - Herbivorous are exactly the animals that only eat plants
  - Carnivorous are exactly the animals that only eat meat
  - Omnivorous are exactly the animals that eat both plants and meat
  - Meat and plants are disjoint
  - Something that is eaten is food
- Start the reasoner, check ontology consistency and inferences.
- Add the following statements about individuals (in view Entities > Individuals).
  - perry is a platypus
  - garfield is a cat and eats great_lasagna which is meat
  - tweety is a bird
  - babe is a pig and eats beautiful_apple which is a plant
  - grace is a cow and eats delicious_leaf
- Check consistency and inferences.
- Use the query view to ask the following queries (Tick "Query for Instances" to obtain individuals belonging to the class expression you wrote in the Query field). Check their explanations.
  - find all individuals that are plants
  - find all individuals that produce eggs
  - find all individuals that produce both milk and eggs
  - find all individuals that are herbivorous
  - find all individuals that eat some meat