Web Ontology Language (OWL)

Athens 2011

Mikel Egaña Aranguren

3205 Facultad de Informática
Universidad Politécnica de Madrid (UPM)
Campus de Montegancedo
28660 Boadilla del Monte
Spain

http://www.oeg-upm.net

megana@fi.upm.es
http://mikeleganaaranguren.com

House-keeping

Research and fill the wiki
http://delicias.dia.fi.upm.es/athens/index.php/OWL

Idea sharing

Hands-on

Estimated timetable

Yesterday:

OWL basics.

Today:

Introduction to OWL reasoning (15 min.)
Research (OWL reasoning) (30 min.)
Idea sharing (15 min.)
Hands-on (45 min.)
Hands-on idea-sharing (15 min.)

An OWL ontology with individuals and classes is a Knowledge Base

Knowledge Base (KB): Abox + Tbox

TBox (Terminological Box): ~schema (~ classes)

Abox (Assertional Box): ~data (~ individuals)

OWL works under the Open World Assumption (OWA)

Data Base (Closed World Assumption): the information not mentioned is false (Negation as Failure)

Knowledge Base (Open World Assumption): the information not mentioned is unknown (Can be true or false)

!!!

Pedro has spanish nationality

¿Does Pedro have british nationality?

CWA (DB): No

OWA (OWL KB): We don't know (Pedro can have double nationality). Till we assert that Pedro can only have one nationality, OWL will assume he can have more than one

OWA advantage: we can add new knowledge (e.g. New nationalities) easily, we don't have to “change the schema”

OWA is good for settings in which our knowledge will always be incomplete: open systems like the (Semantic) Web

In OWL there is no Unique Name Assumption (UNA)

The fact that two entities have different URIs does not imply that they are different entities

We have to explicitly assert, if we want to, that two entities are different from each other

In the (Semantic) Web, different resources talk about the same entity

!!!

No UNA + OWA:

Building an ontology in OWL is like pruning a space in which by default everything is possible (OWA) and all the entities are the same (!UNA)

Such prunning is performed by adding axioms that limit the possible facts and make entities different to each other

!!!

Reasoning

Reasoning

Reasoning is performed by using a reasoner: a reasoner infers the axioms implied by the axioms we have stated in the ontology

Thus, ther reasoner generates the *inferred* axioms from the *asserted* axioms

The reasoner makes *all* the implied axioms explicit, including the ones that would be missed by a human because of the complexity/size of the ontology

Therefore, a reasoner helps us deal with complex knowledge

OWL offers *sound and complete* reasoning if we don't use OWL full constructs (e.g. make an object property functional and transitive, …)

That is the theory. In practice there can be efficiency problems. Reasoners are improving fast and OWL 2 offers different profiles optimized for different kinds of reassoning

Reasoning can be used to:

Maintain a class hierarchy

Check consistency of the ontology

Clasify an entity against the ontology

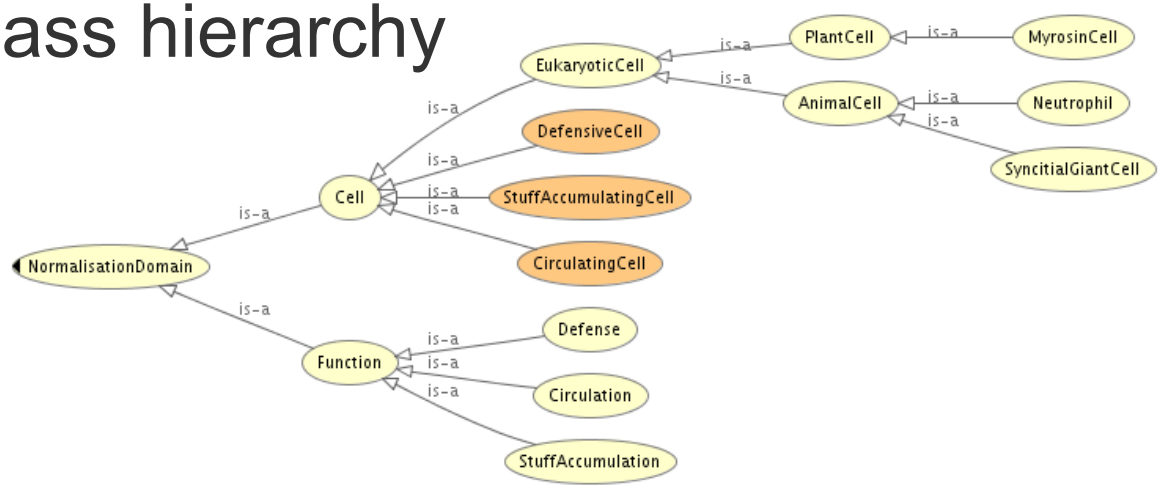
Make queries against the ontology

Use reasoning every time you change your ontology

Be aware of OWA and lack of UNA

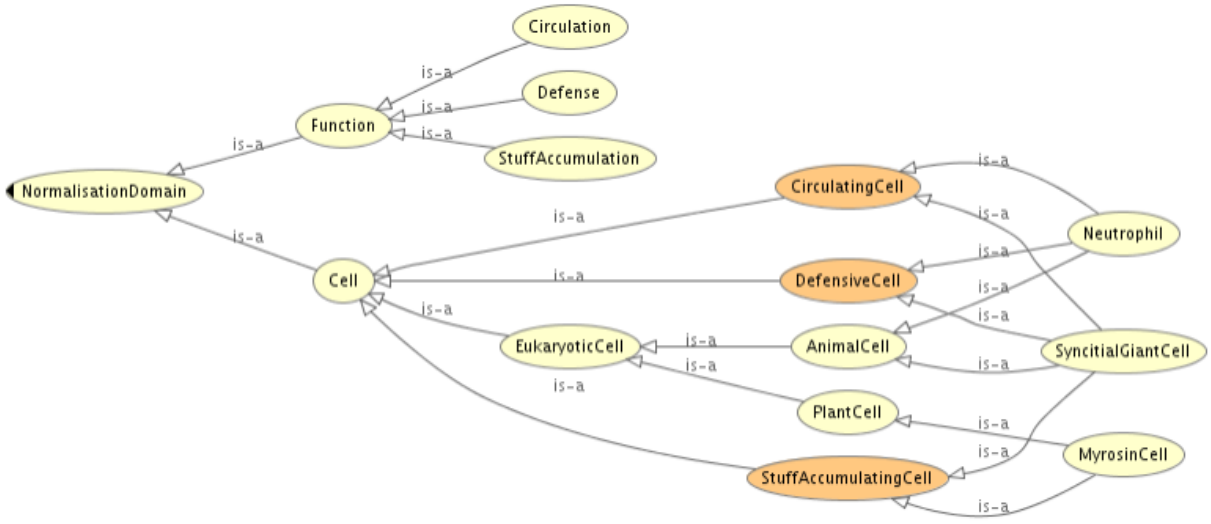
!!!

Maintain a class hierarchy



http://www.gong.manchester.ac.uk/odp/html/Normalisation.html

CLASSIFY



Check consistency of an ontology

Not satisfiable classes cannot have any individual (There is no individual that can satisfy the axioms)

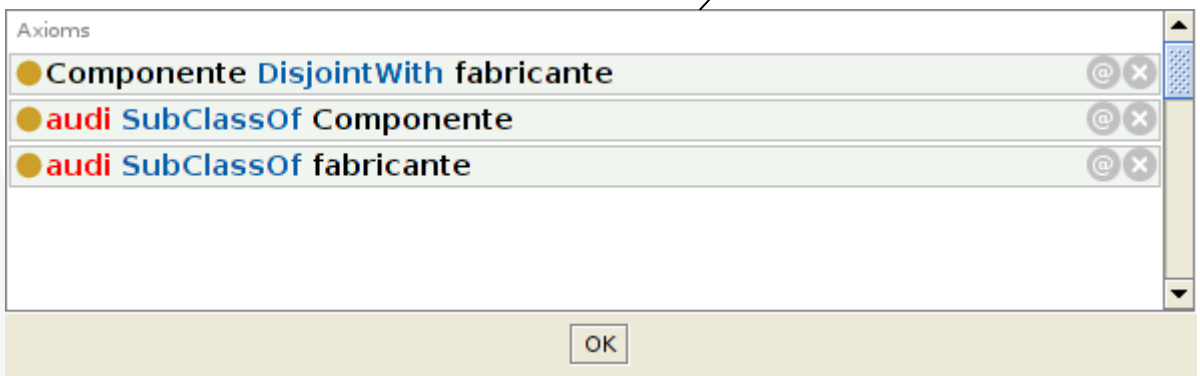
An ontology becomes *inconsistent* if we state that a not satisfiable class has an individual

In an inconsistent ontology, not satisfiable classes are subclasses of owl:Nothing

Automated reasoning cannot be performed in an inconsistent ontology

An inconsistent ontology usually means that we have modelled something wrong

Check consistency of an ontology



Classify new entities against the ontology

Individuals: types

Classes: subClassOf, equivalentTo

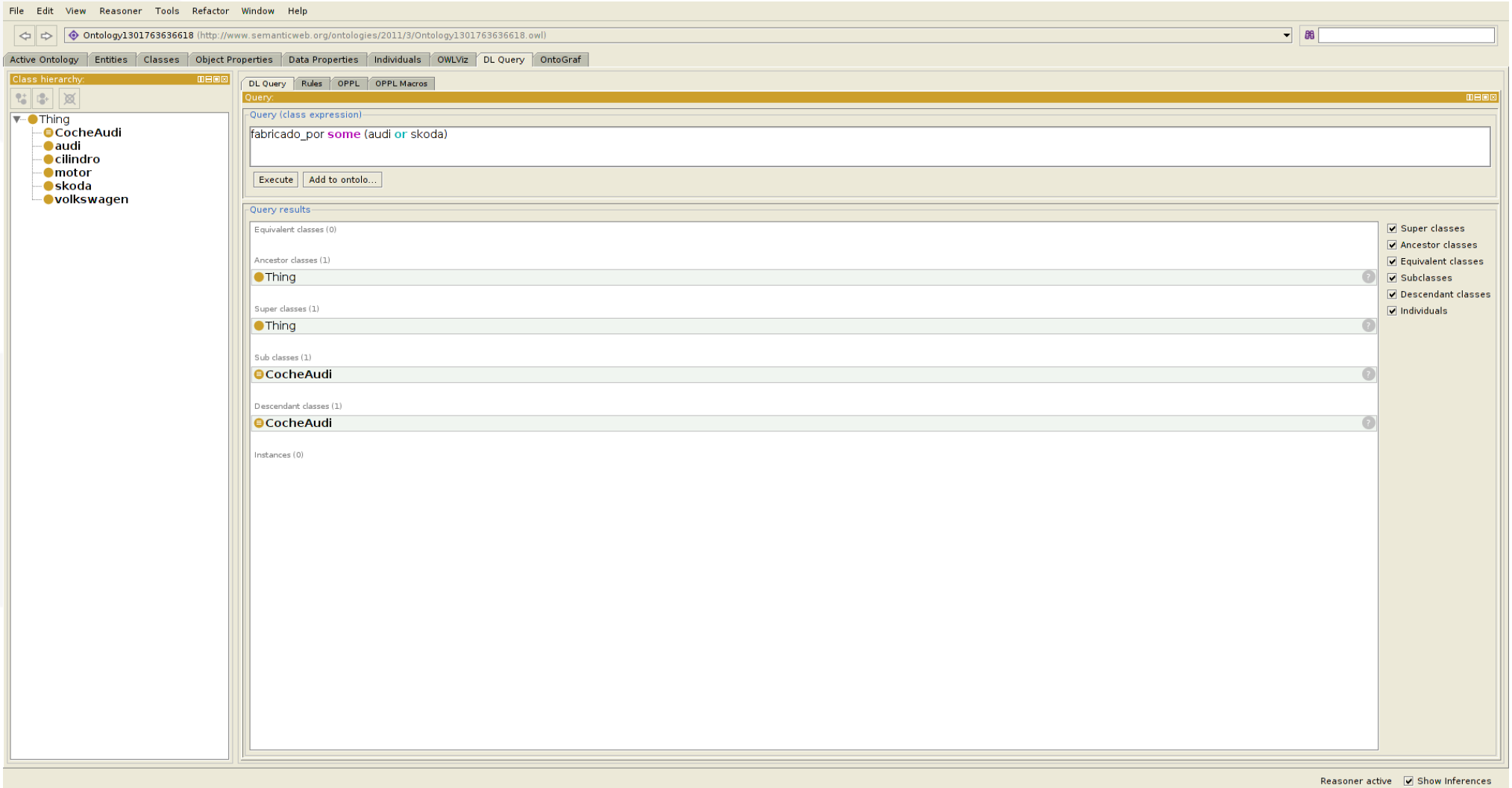
Queries against the ontology

A query is an anonymous class

We ask the reasoner how the entities of the ontology relate to such class (type, subclass, …)

Defined classes can also be regarded as queries

Reasoning


File Edit View Reasoner Tools Refactor Window Help
Ontology1301763636618 (http://www.semanticweb.org/ontologies/2011/3/Ontology1301763636618.owl)
Active Ontology Entities Classes Object Properties Data Properties Individuals OWLViz DL Query OntoGraf
Class hierarchy:
Thing
CocheAudi
audi
cilindro
motor
skoda
volkswagen
DL Query Rules OPPL OPPL Macros
Query:
Query (class expression)
fabricado_por some (audi or skoda)
Execute Add to ontolo...
Query results
Equivalent classes (0)
Ancestor classes (1)
Thing
Super classes (1)
Thing
Sub classes (1)
CocheAudi
Descendant classes (1)
CocheAudi
Instances (0)
Super classes
Ancestor classes
Equivalent classes
Subclasses
Descendant classes
Individuals
Reasoner active
Show Inferences