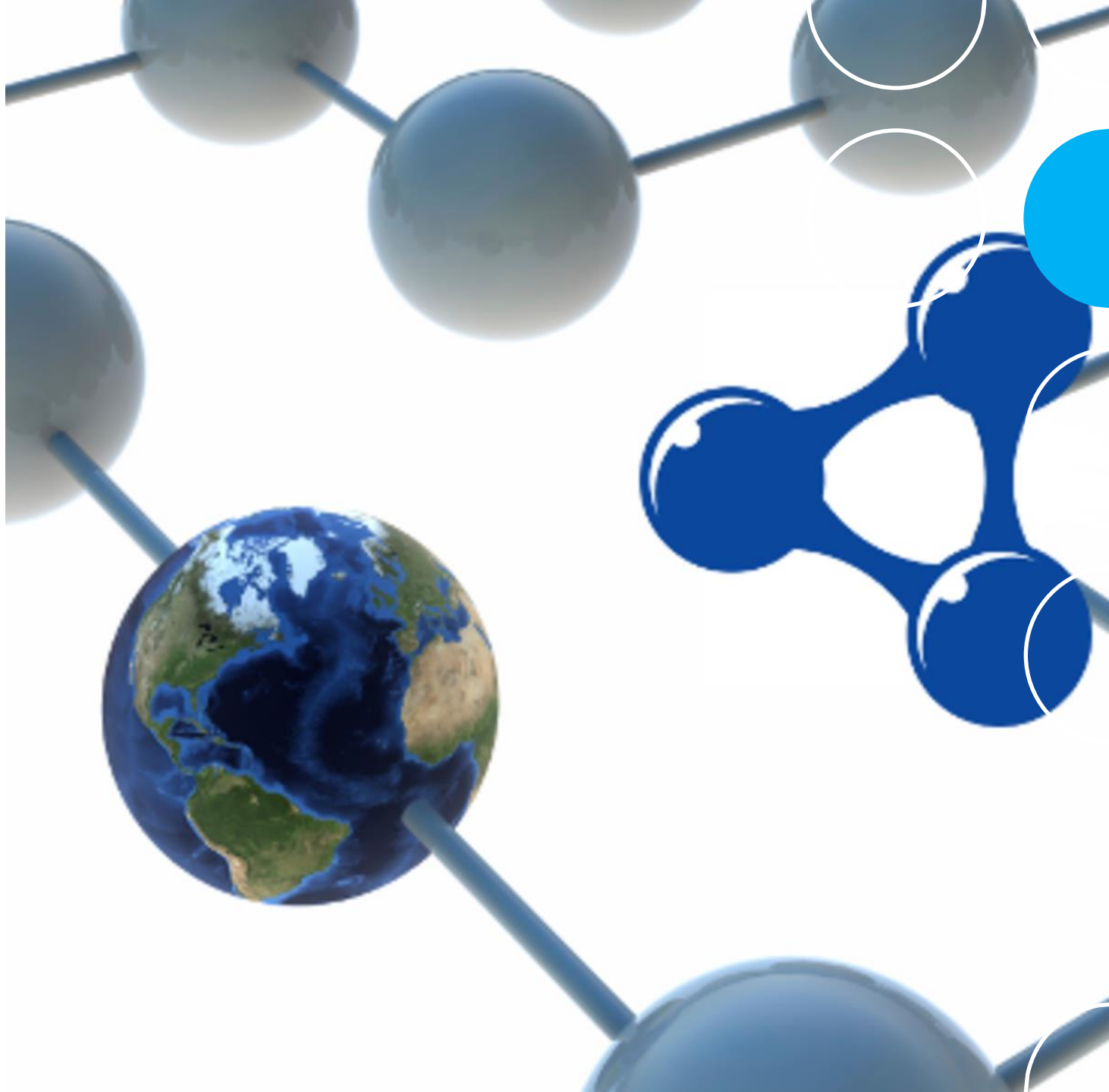


# Semantic Web and Linked Data

Liliana Ferreira  
2022/23

---



# Class 10: Learning Objectives

- The OWL ontology language
    - Properties
    - The Protégé open-source tool.
-

# The Web Ontology Language (OWL) provides concepts for detailed ontologies.

- RDFS captures basic ontological relations but lacks several common and important concepts.
  - cardinality restrictions on properties
  - inverse, symmetric, and transitive properties
  - equality and disjointness
  - ...
- OWL extends RDFS with advanced concepts.
  - RDFS and OWL are used side by side.

# OWL defines additional constraints for individuals, properties, and classes.

## restrictions on individuals

owl:sameAs, owl:differentFrom, ...

## restrictions on properties

owl:ObjectProperty, owl:inverseOf,  
owl:FunctionalProperty, ...

## restrictions on classes

owl:intersectionOf, owl:Restriction, ...

# OWL defines its own version of resources and classes.

**The class of everything is owl:Thing.**

similar to rdfs:Resource

**The class of classes is owl:Class.**

subclass of rdfs:Class

# An IRI uniquely identifies a resource, but one resource can have many IRIs.

- You cannot assume just because 2 IRIs are different they necessarily point to different resources.
  - `ex:Tom` a `ex:Cat`.
  - `ex:Jerry` a `ex:Mouse`.
- You cannot conclude `ex:Tom` and `ex:Jerry` are different.

**An IRI uniquely identifies a resource,  
but one resource can have many IRIs.**

**owl:sameAs** indicates two resources are the same.

**owl:differentFrom** indicates two resources differ.

`ex:Tom owl:differentFrom ex:Jerry.`

**Typical properties can either take a literal or a named node as object.**

**Properties taking only literal values as object are instances of owl:DatatypeProperty.**

```
foaf:givenName a owl:DatatypeProperty.
```

```
foaf:givenName rdfs:range rdfs:Literal.
```



# Typical properties can either take a literal or a named node as object.

## Properties taking only non-literal values as object are instances of owl:ObjectProperty.

foaf:knows a **owl:ObjectProperty**.

foaf:knows rdfs:range \_:NonLiterals.

\_:NonLiterals owl:complementOf rdfs:Literal.

# Inverse properties express a triple in the opposite direction.

One property is the owl:inverseOf another if it asserts the same relation from object to subject.

ex:TimBL **foaf:made** dbr:World\_Wide\_Web.

dbr:World\_Wide\_Web **foaf:maker** ex:TimBL.

foaf:made **owl:inverseOf** foaf:maker

**Inverse properties express a triple  
in the opposite direction.**

**Ontologists typically pick *one* property direction.**

Different ontologies might choose different directions.

`owl:inverseOf` allows to connect such properties.

**A functional property restricts the objects for a given subject to be identical.**

**If any subject can at most have one unique value for some property, it's an owl:FunctionalProperty.**

`ex:Delphine ex:hasBiologicalFather ex:Albert.`

`ex:hasBiologicalFather a`

**`owl:FunctionalProperty.`**

**A functional property restricts the objects for a given subject to be identical.**

**The inverse is owl:InverseFunctionalProperty.**

`ex:Albert ex:isBiologicalFatherOf ex:Delphine.`

`ex:isBiologicalFatherOf a`

**`owl:InverseFunctionalProperty.`**

# Functional properties have strong effects, so you must understand them well.

What is the logical consequence of the following?

`ex:Delphine ex:hasBiologicalFather ex:Albert.`

`ex:Delphine ex:hasBiologicalFather ex:Jacques.`

`ex:hasBiologicalFather a owl:FunctionalProperty.`

# Functional properties have strong effects, so you must understand them well.

It might be counterintuitive, but the conclusion is:

```
ex:Albert owl:sameAs ex:Jacques.
```

To arrive at a contradiction, explicitly define inequality:

```
ex:Albert owl:differentFrom ex:Jacques.
```

**OWL contains similar properties for  
symmetry, reflexivity, and transitivity.**

## **Exercise:**

Consulting W3C OWL Specifications, define these properties and identify some examples of each.



# Sources

- <https://www.w3.org/TR/owl2-primer/>
  - <https://www.w3.org/TR/owl-ref/>
  - [https://www.w3.org/TR/owl2-syntax/#Reflexive\\_Object\\_Properties](https://www.w3.org/TR/owl2-syntax/#Reflexive_Object_Properties)
  - Ruben Verborgh, Web Fundamentals, University of Ghent.
-