# OWL: An Introduction

Valentina Bartalesi Lenzi

Istituto di Scienza e Tecnologie dell'Informazione - Consiglio Nazionale delle Ricerche – Pisa

A.A. 2024-2025

# The OWL Lineage and Family

The **Web Ontology Language** (OWL) was developed within the World Wide Web Consortium (W3C) in a series of subsequent activities carried out by different groups:

- **OWL 1**, the first language version, became a W3C Recommendation in February 2004. It was developed by the Web-Ontology (WebOnt) Working Group, which operated from November 2001 until May 2004
- In December 2006, **OWL 1.1**, an extension of OWL 1, was proposed for including the theoretical advances that had been in the meantime achieved in Description Logics (DLs), namely the DL SROIQ
- The OWL Working Group was created to turn OWL 1.1 into a new W3C recommendation for an updated OWL
  - The Group operated from 2007 until December 2012, and issued the **first edition** of **OWL 2** in October 2009
- **OWL 2**, **second edition**, became a W3C Recommendation in December 2012

# OWL and RDFS

The development of OWL has been motivated by the desire of **enhancing the expressivity allowed by RDF and RDF Schema** languages, e.g. defining a property as transitive or reflexive, or defining axioms of cardinality on classes

# OWL, RDFS and DLs

OWL uses some solutions proposed by RDF, in particular:

- Any OWL ontology uses **IRIs as names**, encoding them as qualified names, the same way RDF does
- Any OWL ontology relies on **RDF datatypes** and **XML Schema datatypes** for datatypes and data values
- To solve the problem of decidability* of an ontology, OWL is based on **Description Logics** (*RDF is based on the First Order Logic that does not guarantee the ontology decidability!*)
- OWL 2 is based on the **DL SROIQ****

*A problem is decidable if there exists an always terminating algorithm which determines, whether or not a solution exists

**for an overview of the different Descriptions Logics see the notes of the course

# OWL Formats

As far as the concrete OWL notation is concerned, there is an **OWL abstract syntax**, similar to the RDF abstract syntax, which can be **serialized** in one of several formats, not all based on XML

Some formats are, for example:

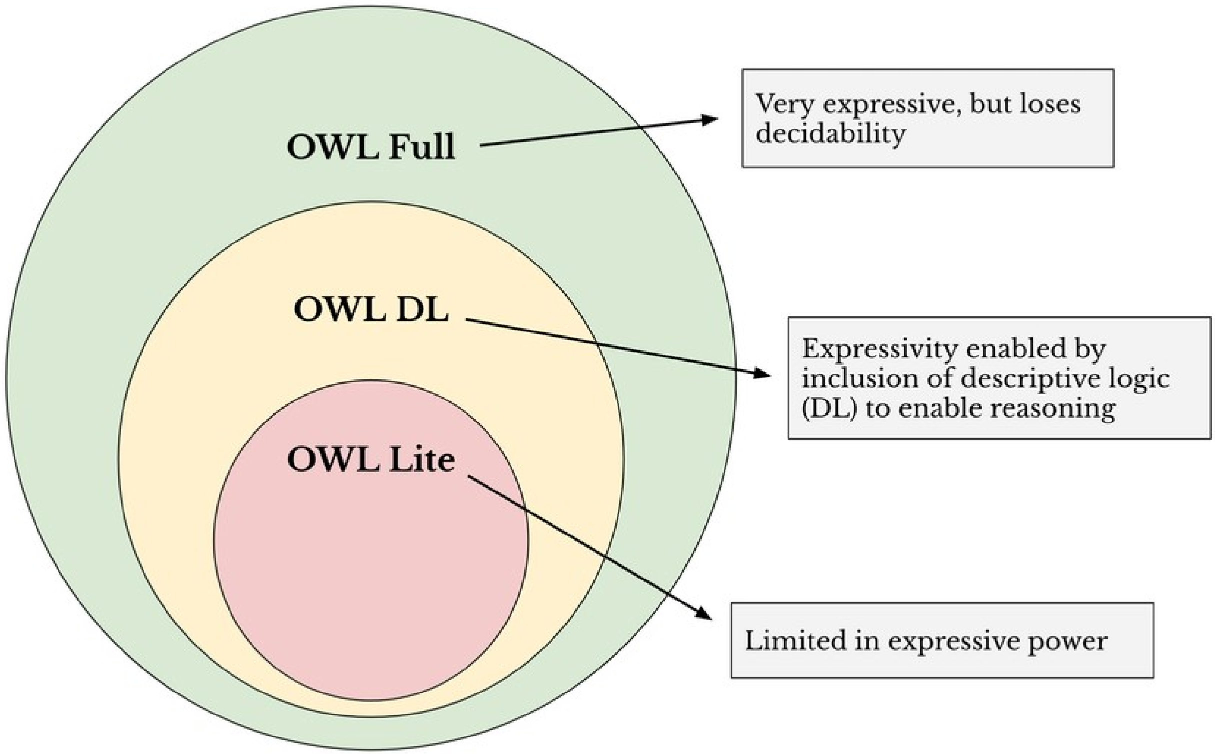
- **RDF/XML**
- **OWL/XML**
- **Turtle** (which we will use)

# OWL variants

Three main variants of **OWL** were defined:

1. **OWL DL**, the language resulting from the encoding of the OWL abstract syntax into a concrete notation. OWL DL <u>is not a semantic extension of RDF</u>
2. **OWL Lite**, a syntactical subset of OWL DL
3. **OWL Full**, the language resulting from <u>extending RDF Schema</u> with the classes and properties needed for encoding the OWL abstract syntax in RDF. OWL Full <u>is a semantic extension of RDF</u>, and the language includes all RDF graphs

# OWL variants



# OWL Semantics

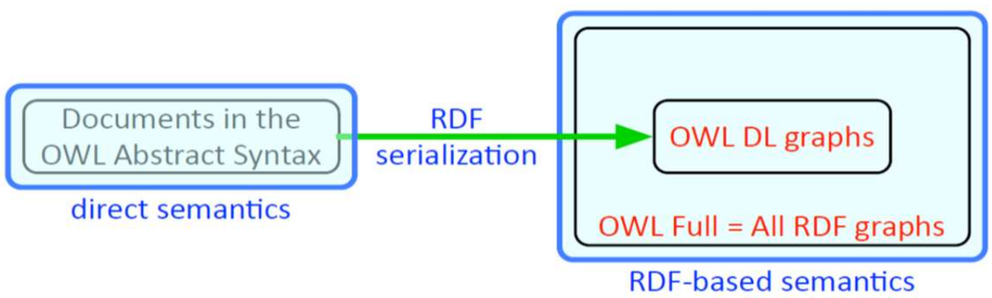
We have therefore two semantics for OWL:

1. the **direct semantics**, defined on the OWL abstract syntax, and based on DLs model theory
2. the **RDF-based semantics**, defined on the encoding of OWL Full in the RDF abstract syntax and based on the RDF model theory (First Order Logic)

# OWL Semantics

- The **direct semantics <u>cannot</u> be applied to OWL Full** because the additional expressive power of OWL Full is meaningless in a DL

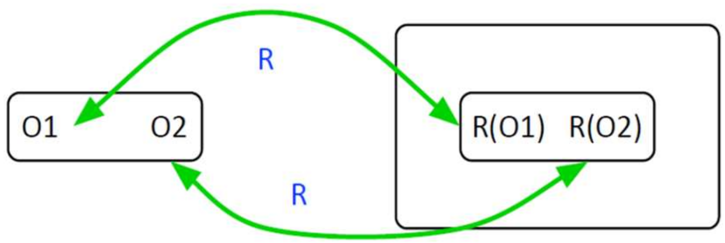
- The **RDF-based semantics <u>can</u> be applied to OWL DL graphs encoded in RDF**. In this case, the two semantics behave in the same way



# OWL Semantics: Theorem

**Theorem (Correspondence between Direct and RDF-based Semantics)**

*Given two OWL DL ontologies O1 and O2, written in the abstract syntax, O1 entails O2 according to the OWL DL model theory if and only if the mapping of O1 into RDF triples entails the mapping of O2 into RDF triples according to the OWL Full model theory.*




O1 entailsD O2 if and only if R(O1) entailsF R(O2)

# OWL 2 DL Profiles

**OWL DL** is **intractable***, and for this reason, OWL 2 DL comes equipped with easier profiles, in particular:

**Three profiles** were defined, also syntactical restrictions of OWL 2 DL, each devised for a specific class of applications:

1. **OWL 2 EL**
2. **OWL 2 QL**
3. **OWL 2 RL**

In contrast, OWL Full retains full compatibility with the RDF syntax, and the OWL abstract syntax is not sufficiently rich in expressing OWL Full ontologies. However, **OWL Full is undecidable****

*Tractable problems are frequently identified with problems that have polynomial-time solutions. Problems that are known to be intractable include those that are EXPTIME-hard.

**A problem is decidable if there exists an always terminating algorithm which determines, whether or not a solution exists

# OWL 2 DL Profiles: Preliminary Definitions

To understand the OWL DL profiles, some definitions are needed:

- **Existential Quantification**. In predicate logic, an existential quantification is a logical constant interpreted as “there exists”, or “there is at least one”. Existential quantification is distinct from universal quantification which asserts that a property holds for *all* members of the domain

- **Ontology Inconsistency.** An ontology is inconsistent if it contains an internal contradiction

- **Class expression subsumption** is the calculus of the class hierarchy

# OWL DL Profiles: Preliminary Definitions

- **Class expression satisfiability.** A class expression CE is satisfiable if exists an interpretation I such that $(\mathrm{CE})^C \neq \emptyset$
- **Instance Checking** is the calculus of the most specific class for an individual
- **Conjunctive Query** is a restricted form of first-order queries using the logical conjunction operator
- **Semantic reasoning** is the ability of a system to infer new facts from existing data based on inference rules or ontologies. A semantic reasoning engine (otherwise known as a semantic reasoner, inference engine, or rules engine) is a software designed to perform reasoning.

# OWL 2 EL Profile

- OWL 2 EL is useful in applications employing ontologies that contain **very large numbers of properties and/or classes**
- OWL 2 EL captures the expressive power used by many such ontologies and it is a subset of OWL 2 for which the **basic reasoning problems** can be performed in time that is **polynomial with respect to the size of the KB**
- Dedicated reasoning algorithms for OWL 2 EL are available and have been demonstrated to be implementable in a **highly scalable way**
- The EL acronym reflects the profile's basis in the **EL family of description logics**: logics that provide only **Existential quantification**

# OWL 2 QL Profile

- OWL 2 QL is aimed at applications that use very **large volumes of instance data**, and where **query answering is the most important reasoning task**
- In OWL 2 QL, conjunctive query answering can be implemented using conventional relational database systems. Using a suitable reasoning technique, sound and complete **conjunctive query answering** can be performed in **LOGSPACE with respect to the size of the data**
- As in OWL 2 EL, **polynomial time algorithms** can be used to implement the **ontology consistency** and **class expression subsumption reasoning problems**
- The QL acronym reflects the fact that query answering in this profile can be implemented by rewriting queries into a **standard relational Query Language**

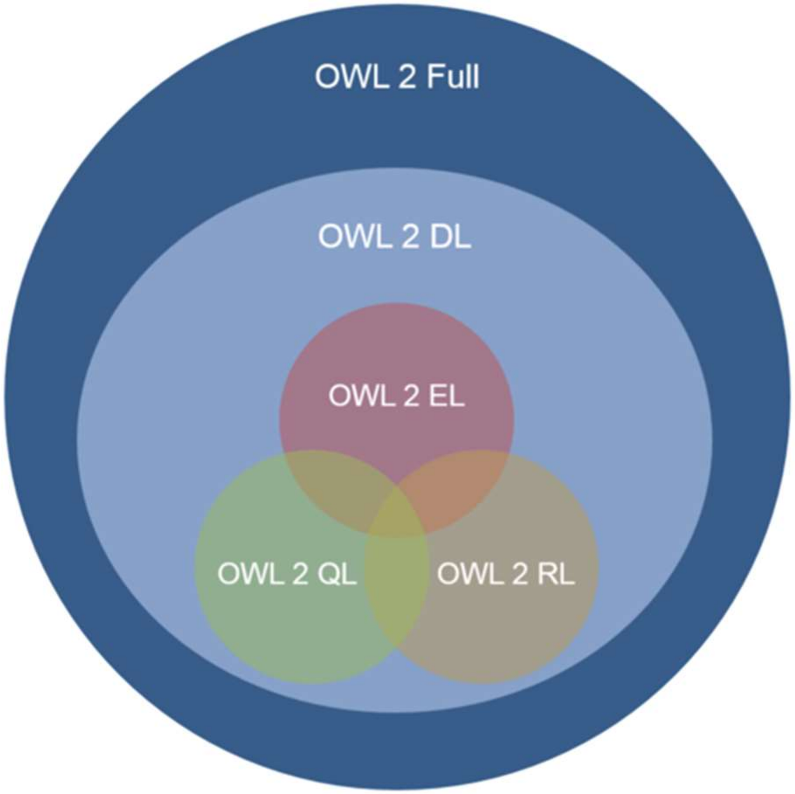
# OWL 2 RL Profile

- OWL 2 RL is aimed at applications that require **scalable reasoning without sacrificing too much expressive power**. It is designed to accommodate OWL 2 applications that can trade the full expressivity of the language for efficiency, as well as RDF(S) applications that need some added expressivity

- OWL 2 RL reasoning systems can be implemented using **rule-based reasoning engines**. The ontology **consistency**, **class expression satisfiability**, **class expression subsumption**, **instance checking**, and **conjunctive query answering** problems can be solved in time that is **polynomial with respect to the size of the KB**

- The RL acronym reflects the fact that reasoning in this profile can be implemented using a standard **Rule Language**

# OWL 2 Profiles

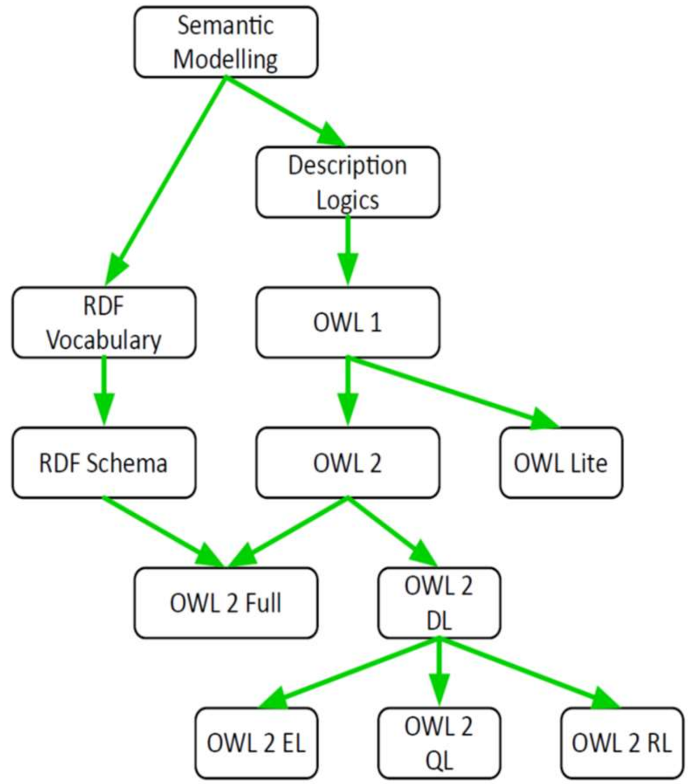


# OWL 2 Variants and Profiles

The OWL family of languages includes a wide variety of languages

They have developed along **two directions** stemming from **semantic modelling** and meeting at **OWL 2 Full**

The variety is aimed at offering useful options on the trade-off between **expressivity** and **tractability**



# Useful Readings


- Richard Cyganiak, David Wood, Markus Lanthaler. RDF 1.1 Concepts and Abstract Syntax.
- Guus Schreiber, Yves Raimond. RDF 1.1 Primer.
- Barwise, J. (1977). An introduction to first-order logic. In Studies in Logic and the Foundations of Mathematics (Vol. 90, pp. 5-46). Elsevier.