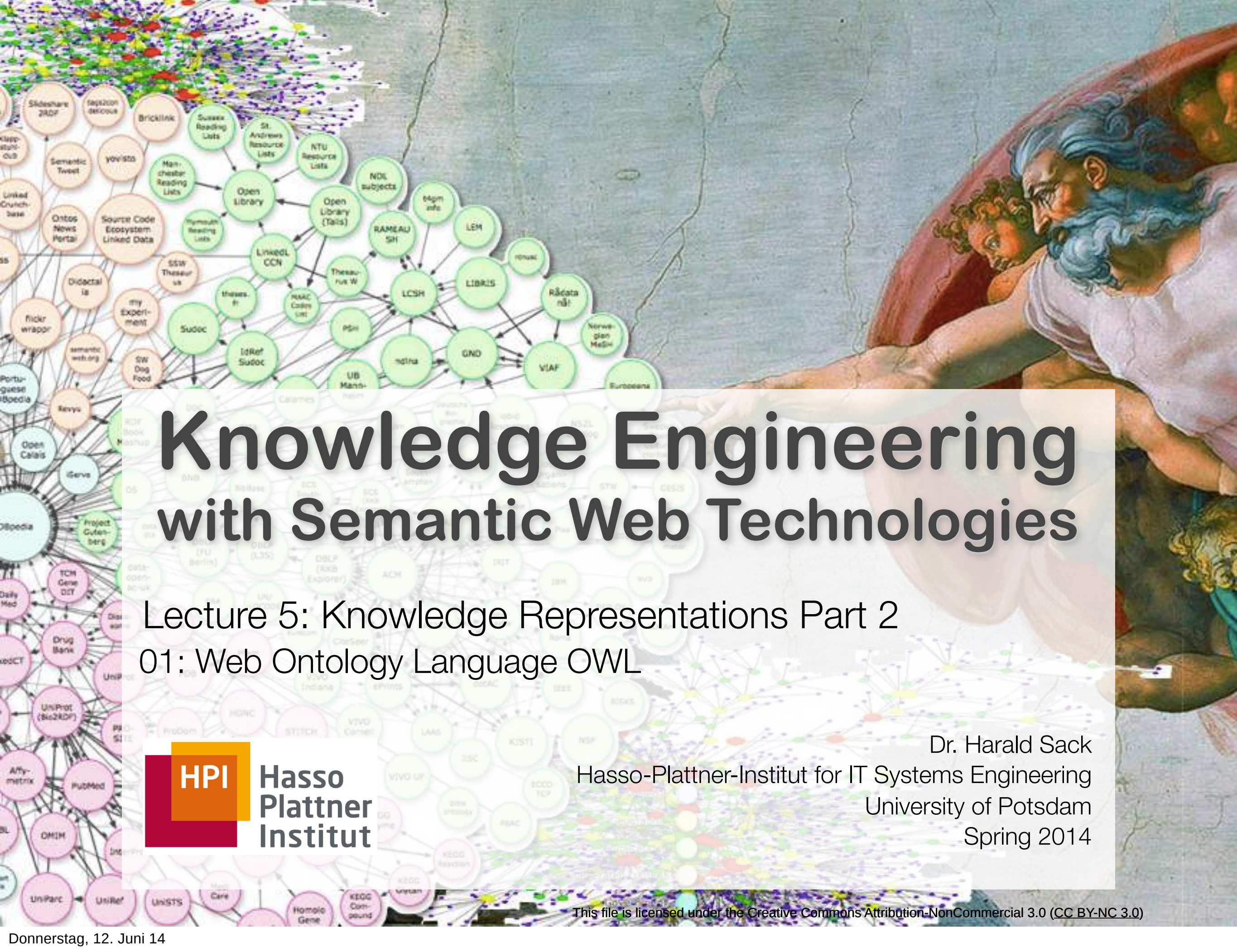




Knowledge Engineering with Semantic Web Technologies

Lecture 5: Knowledge Representations Part 2 01: Web Ontology Language OWL



Dr. Harald Sack
Hasso-Plattner-Institut for IT Systems Engineering
University of Potsdam
Spring 2014



Lecture 5: Knowledge Representations II

Open HPI Course: Knowledge Engineering with Semantic Web Technologies



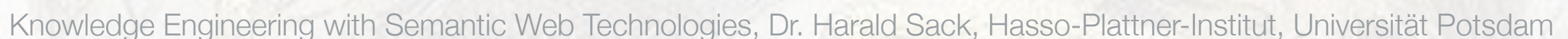
01 Web Ontology Language - OWL

Open HPI Course: Knowledge Engineering with Semantic Web Technologies
Lecture 05: Knowledge Representations Pt.2

OWL – Web Ontology Language

- OWL ($\mathcal{SHOIN}(\mathcal{D})$)- W3C Recommendation since 2004
- OWL 2 ($\mathcal{SHROIQ}(\mathcal{D})$)- W3C Recommendation since 2009
- an OWL Ontology consists of
 - classes / properties / individuals (instances of classes)
- **Open World Assumption**
 - „Absence of information must not be valued as negative information.“
 - E.g.: *sitsNextTo(PersonA, PersonB)*
PersonA may also sit next to another person...
- **No Unique Name Assumption**
 - „Difference must be expressed explicitly“
 - E.g.: *PersonA possibly denotes the same individual as PersonB*

- OWL is a semantic fragment of FOL
- OWL1 exists in different flavors
 - OWL Lite $\subseteq$ OWL1 DL $\subseteq$ OWL1 Full



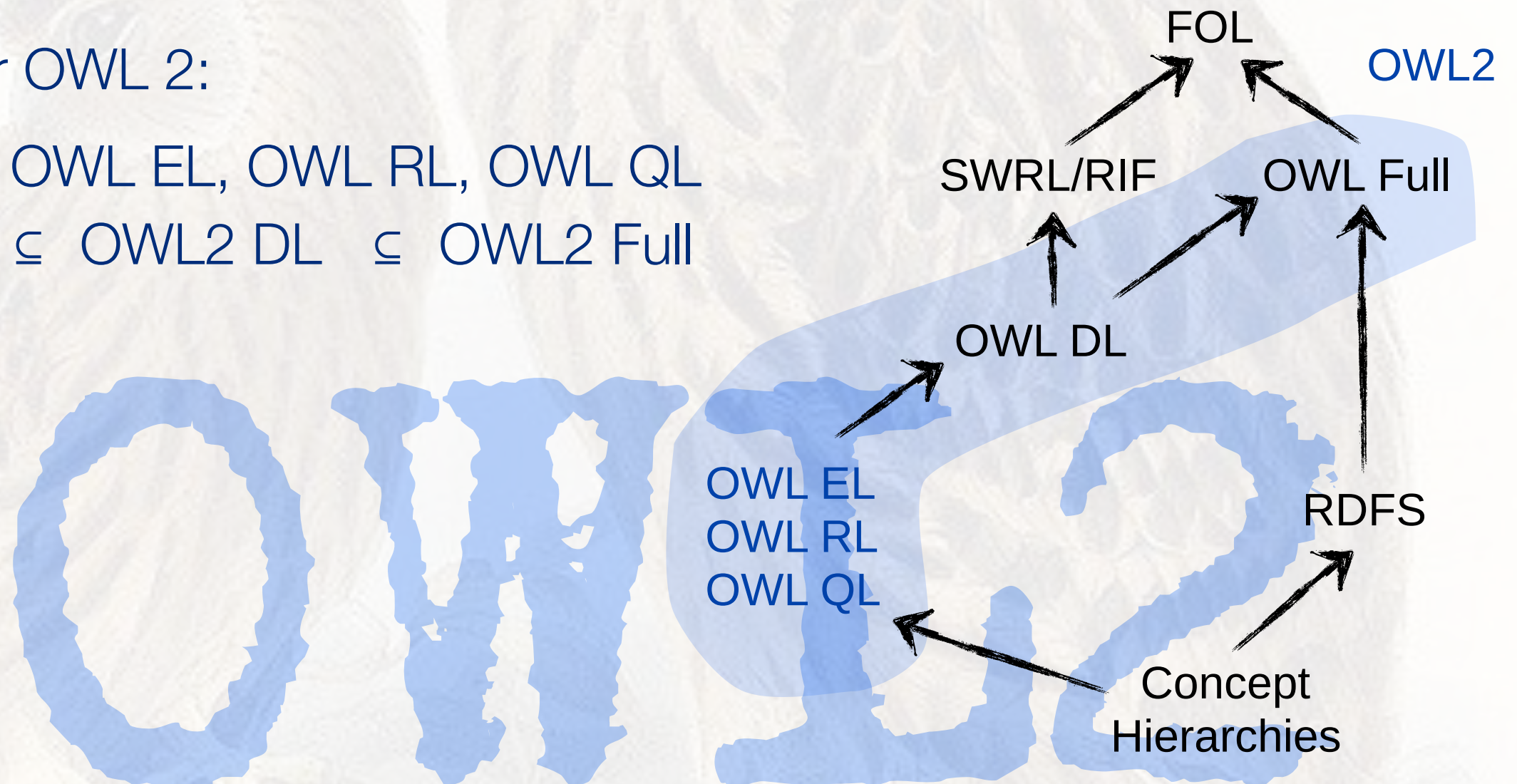
OWL1 DL is based on $\mathcal{SHOIN}(\mathcal{D})$

• Axioms

- **TBox**: subclass relationships $C \sqsubseteq D$ ($\mathcal{H}$)
- **RBox**: subproperty relationships $R \sqsubseteq S$ ($\mathcal{H}$),
inverse properties R^{-} ($\mathcal{I}$), transitivity $\sqsubseteq^+$ ($\mathcal{S}$)
- **ABox**: facts for classes $C(a)$, properties $R(a, b)$,
equality $a=b$, difference $a \neq b$
- **Class constructors**:
 - conjunction $C \sqcap D$, disjunction $C \sqcup D$, negation $\neg C$ of classes
 - property restrictions: universal $\forall R. C$ and existential $\exists R. C$
 - number restrictions: $\leq n R$ and $\geq n R$ ($\mathcal{N}$)
 - closed classes (nominals): $\{a\}$ ($\mathcal{O}$)
- Datatypes ($\mathcal{D}$)

OWL – Web Ontology Language

- OWL is a semantic fragment of FOL
- OWL1 exists in different flavors
 - $\text{OWL Lite} \subseteq \text{OWL1 DL} \subseteq \text{OWL1 Full}$
- for OWL 2:
 - $\text{OWL EL, OWL RL, OWL QL} \subseteq \text{OWL2 DL} \subseteq \text{OWL2 Full}$



OWL2 DL is based on $\mathcal{SHROIQ}(\mathcal{D})$

8

Class Expressions

- Class names A, B
- Conjunction $C \sqcap D$
- Disjunction $C \sqcup D$
- Negation $\neg C$
- Exist. property restriction $\exists R.C$
- Univ. property restriction $\forall R.C$
- Self $\exists S.\text{Self}$
- Greater-than $\geq n\ S.C$
- Less-than $\leq n\ S.C$
- Enumerated classes $\{a\}$

Properties

- Property names R, S, T
- Simple properties S, T
- Inverse properties R^{-}
- Universal property U

Tbox (Class axioms)

- Inclusion $C \sqsubseteq D$
- Equivalence $C \equiv D$

Rbox (Property Axioms)

- Inclusion $R_1 \sqsubseteq R_2$
- General Inclusion $R^{(-)}_1 \circ R^{(-)}_2 \circ \dots \circ R^{(-)}_n \sqsubseteq R$
- Transitivity
- Symmetry
- Reflexivity
- Irreflexivity
- Disjointness

Abox (Facts)

- Class membership $C(a)$
- Property relation $R(a, b)$
- Negated property relation $\neg S(a, b)$
- Equality $a = b$
- Inequality $a \neq b$

OWL Syntax Variants

- OWL2 can be represented in different Syntax variants
 - **Functional Syntax**: substitutes abstract syntax of OWL1
 - **RDF-Syntax**: extension of existing OWL/RDF
 - **XML-Syntax**: independent XML serialisation
 - **Manchester-Syntax**: machine readable syntax,
esp. for ontology editors (e.g. Protégé)
 - **Turtle**: optional
- Functional Syntax is easy to define, no RDF restrictions, more compact
- RDF-Syntax important for compatibility issues
- **Turtle: simple and efficient to write...**

OWL2 - Turtle Syntax - Example

HappyPerson $\equiv \forall \text{hasChild}.\text{HappyPerson} \sqcap \exists \text{hasChild}.\text{HappyPerson}$

```
:HappyPerson a owl:Class ;  
  owl:equivalentClass [  
    a owl:Class ;  
    owl:intersectionOf ([ a owl:Restriction ;  
                           owl:onProperty :hasChild ;  
                           owl:allValuesFrom :HappyPerson ]  
                          [ a owl:Restriction ;  
                           owl:onProperty :hasChild ;  
                           owl:someValuesFrom :HappyPerson ]  
                        )  
  ] .
```

OWL2 - RDF/XML Syntax - Example

HappyPerson $\equiv \forall \text{hasChild}.\text{HappyPerson} \sqcap \exists \text{hasChild}.\text{HappyPerson}$

```
<?xml version="1.0"?>
<rdf:RDF xmlns=...>
<owl:ObjectProperty rdf:about="http://example.org/turtle#hasChild"/>

<owl:Class rdf:about="http://example.org/turtle#HappyPerson">
  <owl:equivalentClass>
    <owl:Class>
      <owl:intersectionOf rdf:parseType="Collection">
        <owl:Restriction>
          <owl:onProperty rdf:resource="http://example.org/turtle#hasChild"/>
          <owl:someValuesFrom rdf:resource="http://example.org/turtle#HappyPerson"/>
        </owl:Restriction>
        <owl:Restriction>
          <owl:onProperty rdf:resource="http://example.org/turtle#hasChild"/>
          <owl:allValuesFrom rdf:resource="http://example.org/turtle#HappyPerson"/>
        </owl:Restriction>
      </owl:intersectionOf>
    </owl:Class>
  </owl:equivalentClass>
</owl:Class>
</rdf:RDF>
```

Standard Namespaces and Prefixes

Prefix name	Prefix IRI
<i>rdf:</i>	<code><http://www.w3.org/1999/02/22-rdf-syntax-ns#></code>
<i>rdfs:</i>	<code><http://www.w3.org/2000/01/rdf-schema#></code>
<i>xsd:</i>	<code><http://www.w3.org/2001/XMLSchema#></code>
<i>owl:</i>	<code><http://www.w3.org/2002/07/owl#></code>

(cf. <http://www.w3.org/TR/owl2-syntax/>)



The screenshot shows the prefix.cc website, which is a namespace lookup tool for RDF developers. The page has a clean, minimalist design with a white background. At the top, the text "prefix.cc" is displayed in a large, bold, black font, followed by the subtitle "namespace lookup for RDF developers" in a smaller, regular black font. Below this, there is a search interface consisting of a text input field containing the prefix "foaf" and a "look up" button to its right. Underneath the search field, a line of text provides examples of how the prefix is used: "examples: foaf foaf:knows dc,foaf rdfs,dc,foaf,geo.sparql http://xmlns.com/foaf/0.1/name". At the bottom of the page, there are links for "popular", "latest", and "about", followed by the "prefix.cc" logo. In the bottom left corner, there are logos for "NUI Galway OE Gaillimh" and "DERI Galway".

<http://prefix.cc/>



02 Classes, Instances, and Relations in OWL

Open HPI Course: Knowledge Engineering with Semantic Web Technologies
Lecture 05: Knowledge Representations Pt.2