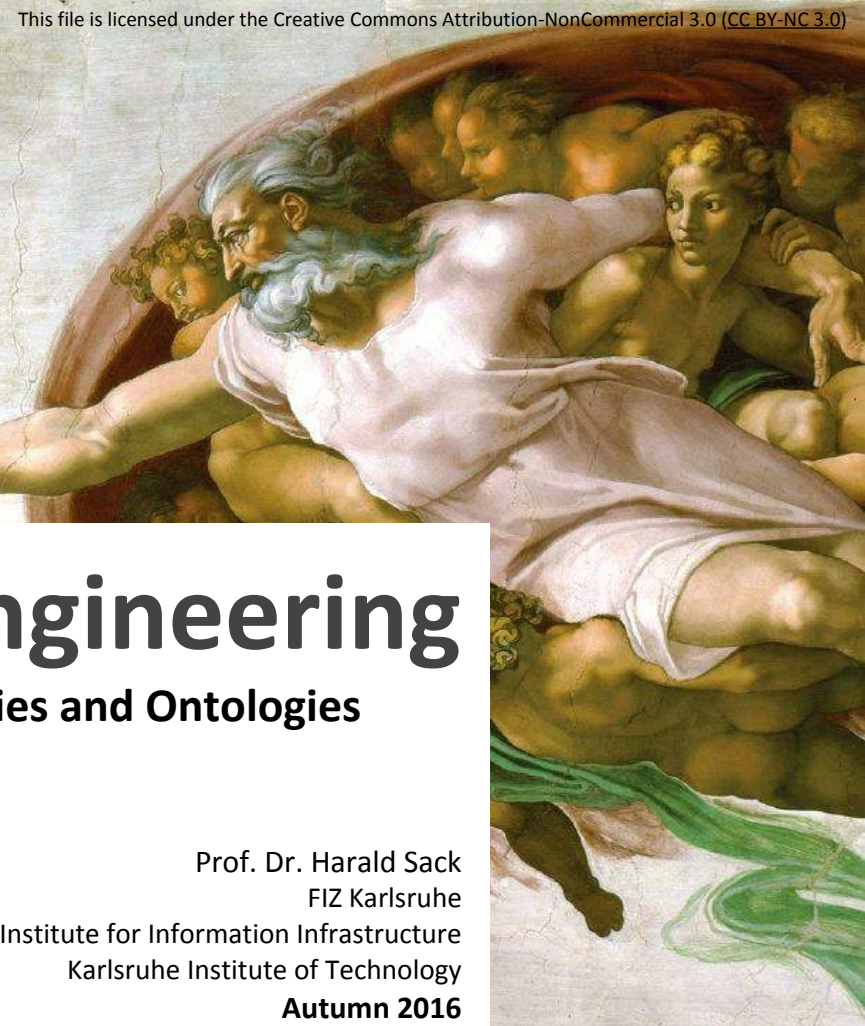




3.4 More OWL



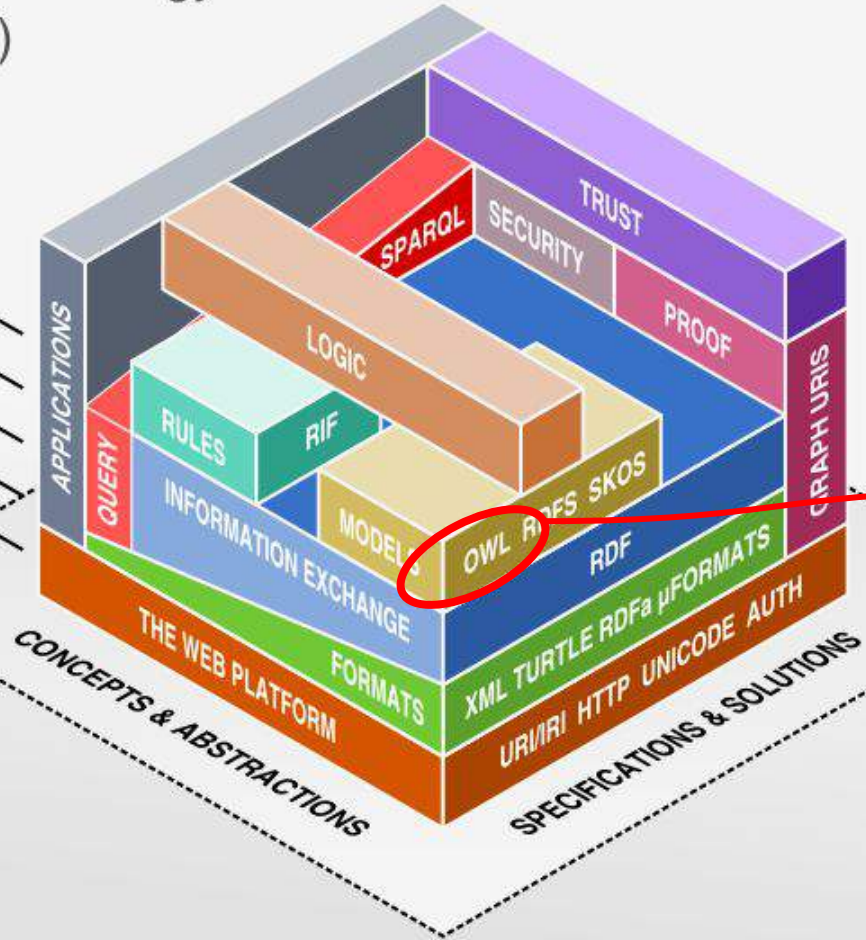
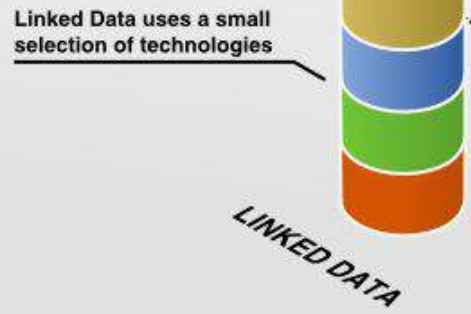
Leibniz Institute for Information Infrastructure

Leibniz Institute for Information Infrastructure
Karlsruhe Institute of Technology
Autumn 2016

Autumn 2016

The Semantic Web Technology Stack (not a piece of cake...)

- Most apps use only a subset of the stack
- Querying allows fine-grained data access
- Standardized information exchange is key
- Formats are necessary, but not too important
- The Semantic Web is based on the Web



Web Ontology Language (OWL)

NovelsInStore $\sqsubseteq$ {AnimalFarm, NineteenEightyfour}

-
- ```

graph TD
 OWL2[OWL2] --> OWLEL[OWL EL]
 OWL2 --> OWLRL[OWL RL]
 OWL2 --> OWLQL[OWL QL]
 OWL2 --> OWL_DL[OWL DL]
 OWLEL --> OWL_DL
 OWL_DL --> SWRLRIF[SWRL/RIF]
 SWRLRIF --> FOL[FOL]
 SWRLRIF --> OWL2
 OWL_DL --> OWL_Full[OWL Full]
 OWL2 --> OWL_Full
 OWL_Full --> RDFS[RDFS]
 OWL_Full --> CH[Concept Hierarchies]

```

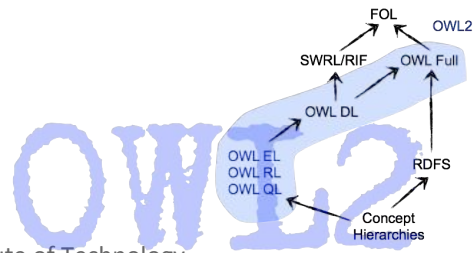


# OWL Logical Class Constructors

```
:Book a owl:Class .
:ThingsInStore a owl:Class .
:BooksInStore a owl:Class ;
 owl:intersectionOf (:ThingsInStore :Book) .
```

$\text{BooksInStore} \equiv \text{ThingsInStore} \sqcap \text{Books}$

- The class `:BooksInStore` results from the intersection of all individuals of the classes `:ThingsInStore` and `:Book`

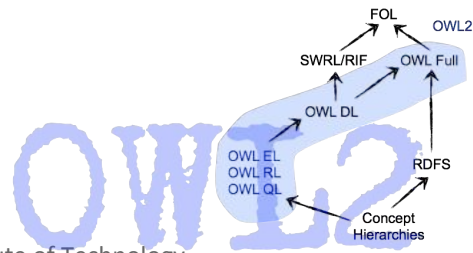


# OWL Logical Class Constructors

```
:Book a owl:Class ;
 owl:equivalentClass [
 owl:unionOf (:Novel
 :Poetry
 :NonFiction)
] .
```

Book  $\equiv$  Novel  $\sqcup$  Poetry  $\sqcup$  NonFiction

- Novels, poetry, and non-fiction are also books

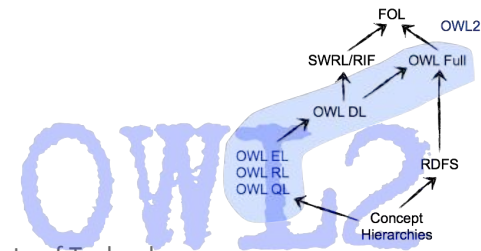




HPI Hasso Plattner Institut

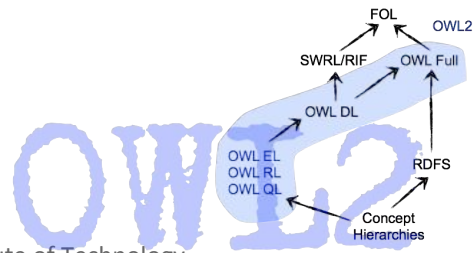
Book  $\sqsubseteq$   $\neg$ Writer

- ```
:Book a owl:Class ;
    owl:disjointWith :Writer .
```



OWL Property Restrictions

- OWL property restrictions are used to describe complex classes via properties
- restrictions on values:
 - `owl:hasValue`
 - `owl:allValuesFrom`
 - `owl:someValuesFrom`
- restrictions on cardinality:
 - `owl:cardinality`
 - `owl:minCardinality`
 - `owl:maxCardinality`

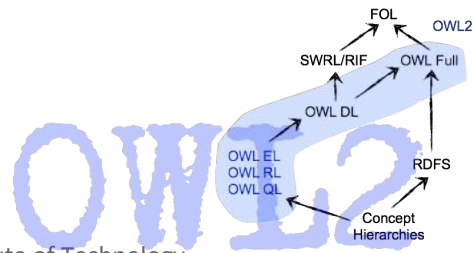


OWL Property Restrictions with Constants

```
:OrwellsBooks a owl:Class ;  
  rdfs:subClassOf  
    [ a owl:Restriction ;  
      owl:onProperty :author ;  
      owl:hasValue :GeorgeOrwell ] .
```

OrwellsBooks $\sqsubseteq$ author.(GeorgeOrwell)

- Class :OrwellsBooks is described via fixed value assignment (=constant) of the individual :GeorgeOrwell to the property :author

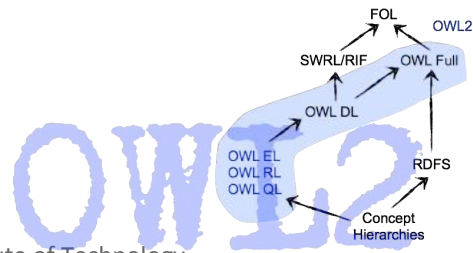


OWL Property Restriction with Strict Binding

```
:Poetry a owl:Class ;  
  rdfs:subClassOf  
    [ a owl:Restriction ;  
      owl:onProperty :author ;  
      owl:allValuesFrom :Poet ] .
```

Poetry $\sqsubseteq \forall \text{author.Poet}$

- **owl:allValuesFrom**
fixes all instances of a specific class C
as allowed range for a property p
(strict binding) $\forall p.C$



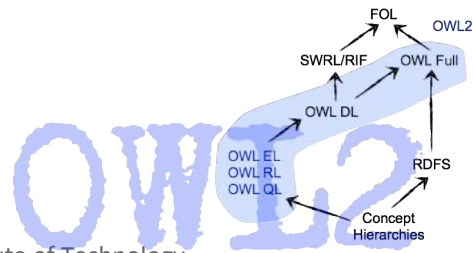
OWL Property Restriction with Loose Binding

```
:Reader a owl:Class ;  
  rdfs:subClassOf  
    [ a owl:Restriction ;  
      owl:onProperty :reads ;  
      owl:someValuesFrom :Book ] .
```

Reader $\sqsubseteq \exists \text{reads}.\text{Book}$

- **owl:someValuesFrom**

describes that there must exist an individual for p
and fixes its range to class C (loose binding) $\exists p.C$



Tetralogy $\sqsubseteq$ (=4)hasVolumes

-
- ```

graph TD
 OWL2[OWL2] --> OWL_EL[OWL EL]
 OWL2 --> OWL_RL[OWL RL]
 OWL2 --> OWL_QL[OWL QL]
 OWL2 --> OWL_DL[OWL DL]
 OWL2 --> OWL_Full[OWL Full]
 OWL2 --> RDFS[RDFS]
 OWL2 --> SWRL_RIF[SWRL/RIF]
 OWL2 --> FOL[FOL]
 OWL_DL --> OWL_Full
 OWL_Full --> RDFS
 OWL_Full --> SWRL_RIF
 OWL_Full --> FOL

```



**Next: 05 - OWL and beyond**

Lecture 3: Linked Data Vocabularies and Ontologies - OpenHPI - Course Linked Data Engineering