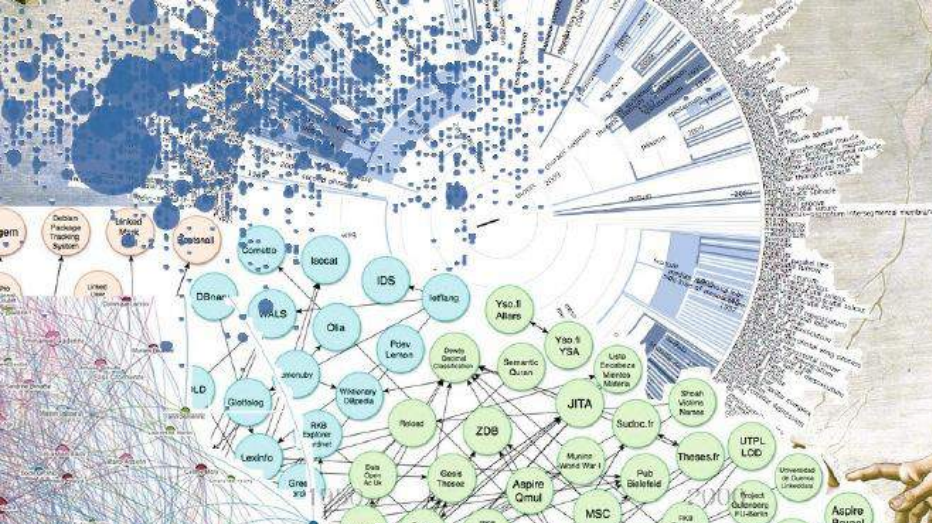




Linked Data Engineering

Lecture 3: Linked Data Vocabularies and Ontologies

3.1 Vocabularies and Ontologies



Leibniz Institute for Information Infrastructure

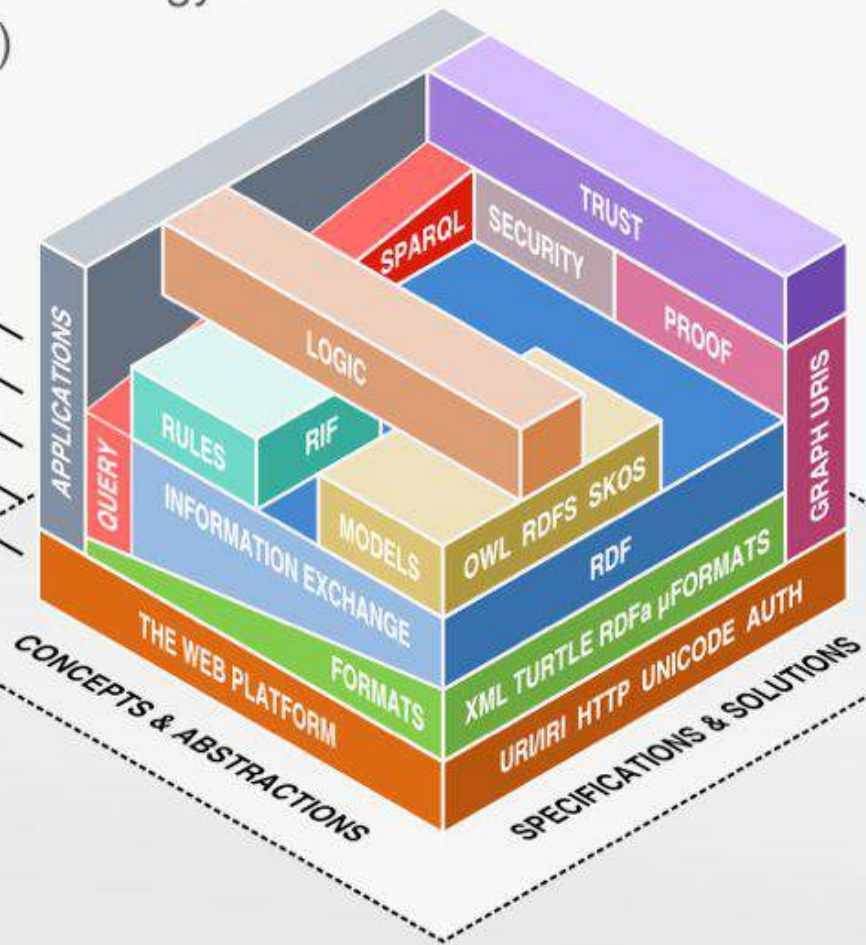
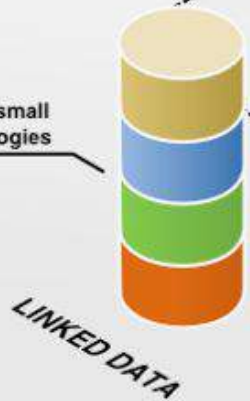
Prof. Dr. Harald Sack
FIZ Karlsruhe

Leibniz Institute for Information Infrastructure
Karlsruhe Institute of Technology

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
- Linked Data uses a small selection of technologies

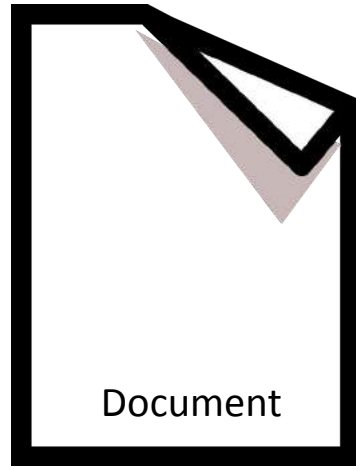


Vocabularies

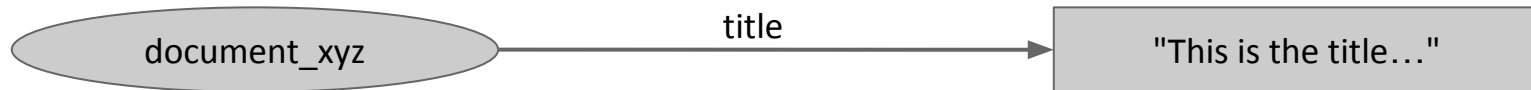
- A person's **vocabulary** is the set of words within a language that are familiar to that person. (*Wikipedia*)
- = “all the words known and used by a particular person”
(*Cambridge Advanced Learners Dictionary*)
- On the Semantic Web, **vocabularies** define the concepts and relationships used to describe and represent an area of concern. (W3C)
- **Vocabularies** are used to
 - **classify** the **terms** that can be used in a particular application,
 - **characterize** possible **relationships**, and
 - **define** possible **constraints** on using those terms.

Example

- The **Dublin Core Vocabulary** (Dublin Core Metadata Element Set)

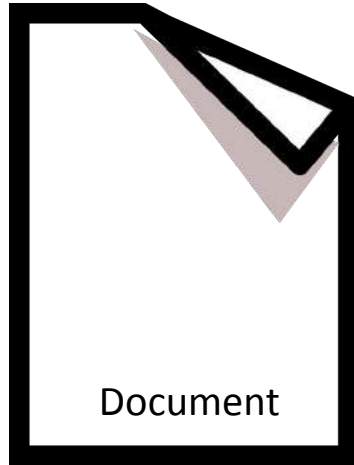


- | | |
|---------------|--------------|
| • Title | • Type |
| • Creator | • Format |
| • Subject | • Identifier |
| • Description | • Source |
| • Publisher | • Language |
| • Contributor | • Relation |
| • Date | • Coverage |
| | • Rights |



Example

- The **Dublin Core Vocabulary** (Dublin Core Metadata Element Set)



- Title
- Creator
- Subject
- Description
- Publisher
- Contributor
- Date
- Type
- Format
- Identifier
- Source
- Language
- Relation
- Coverage
- Rights

PREFIX dct: `<http://purl.org/dc/terms/>` .

`<http://example.org/document_xyz>` `dct:title` "This is the title..."@en .

Ontologies for Knowledge Representation

An ontology is an **explicit, formal specification of a shared conceptualization**. The term is borrowed from philosophy, where an Ontology is a systematic account of existence. For AI systems, what ‘exists’ is that which can be represented.

according to Thomas R. Gruber: A Translation Approach to Portable Ontology Specifications. Knowledge Acquisition, 5(2):199-220, 1993.

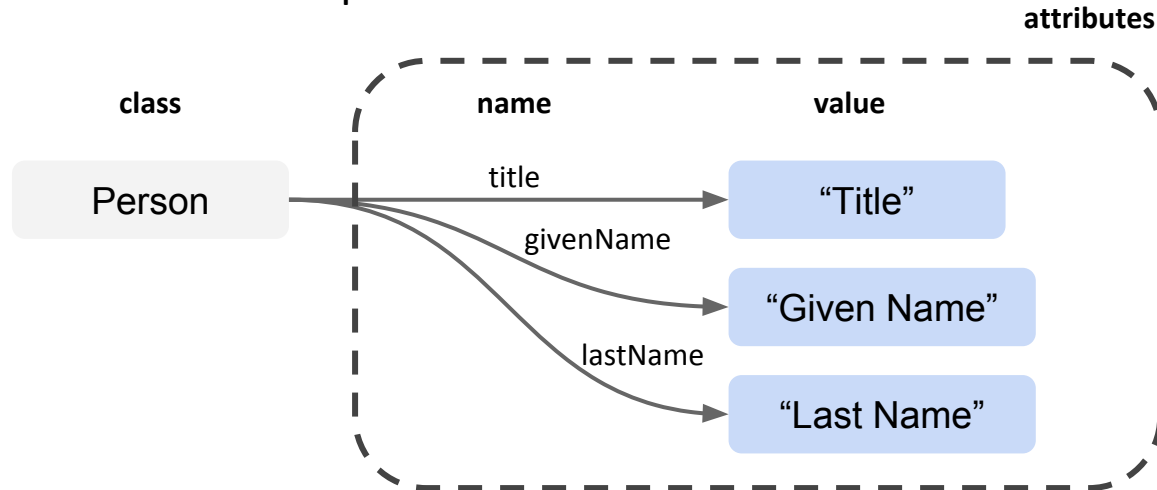
conceptualization:	abstract model (domain, identified relevant concepts, relations)
explicit:	meaning of all concepts must be defined
formal:	machine understandable
shared:	consensus about ontology

Ontology is the philosophical study of the nature of being, existence, or reality, as well as the basic categories of being and their relations...



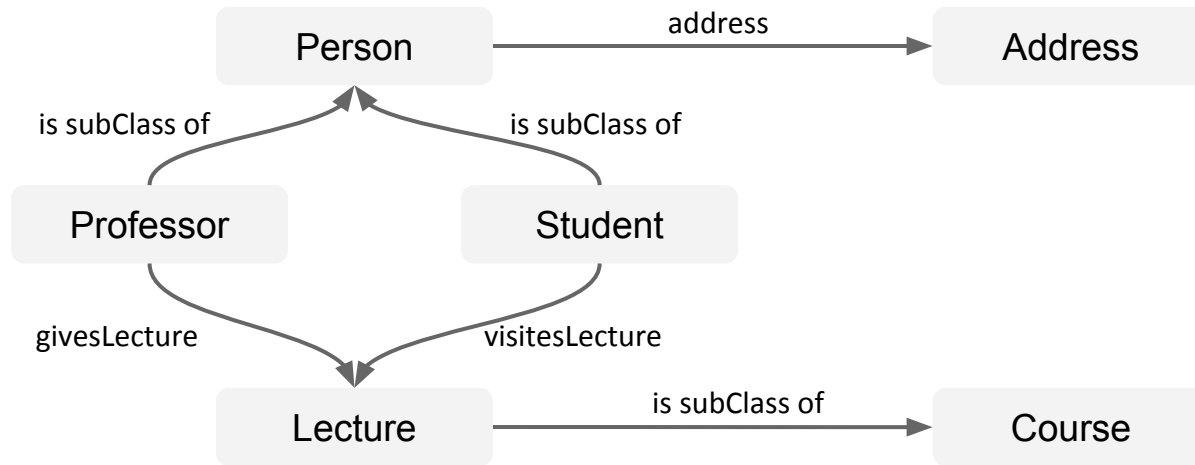
How to represent Ontologies

- Ontologies can be represented via **Classes**, **Relations** and **Instances**
- **Classes** are abstract groups, sets, or collections of objects and represent **ontology concepts**
- Classes are characterised via **attributes**
- **Attributes** are name-value pairs



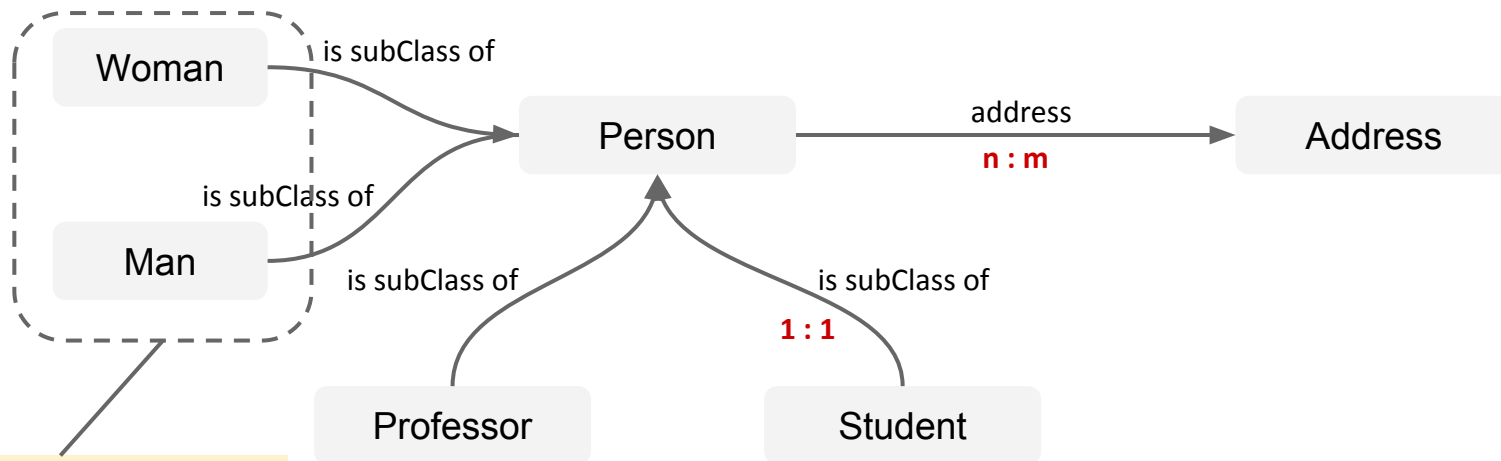
How to represent Ontologies

- Classes can be **related** to other classes
- **Relations** are special attributes, whose values are objects of (other) classes



How to represent Ontologies

- For Relations and Attributes **Rules (Constraints)** can be defined that determine allowed/valid values



$$\text{Woman} \cap \text{Man} = \emptyset$$

constraint

How to represent Ontologies

- Classes, relations, and constraints can be combined to form (complex) **Statements / Assertions**
- Special Case: **formal Axioms**

Example:

- *„it is not possible to attend two courses at the same time“*
-
- **Axioms** describe knowledge that cannot be expressed simply with the help of other existing components.

How to represent Ontologies

- **Instances** describe individuals of an ontology

