Semantic Web and Ontologies

Linked data and its semantics

doc. Ing. Radek Burget, Ph.D.
Brno University of Technology, CZ
burgetr@fit.vut.cz

Semantic Web

- A vision of *web of data* (as opposed to the *web of documents*)
  - Publishing machine-readable (or machine-understandable) data
- Basic concepts:
  - Knowledge representation – Resource description framework (RDF)
  - Shared conceptualization – ontologies
  - Agents – service producers and consumers

Berners-Lee, Tim, Hendler, James, et al. “The Semantic Web : a new form of Web content that is meaningful to computers will unleash a revolution of new possibilities” *Scientific American* 284 (2001) : 34-43

Technical solution

- Developing appropriate data representation technologies
  - Possibility of sharing data together with its semantics
  - Applicable technologies have long been available
- Integration with the existing web
  - Annotation in web pages
  - Somewhat slow, but improving

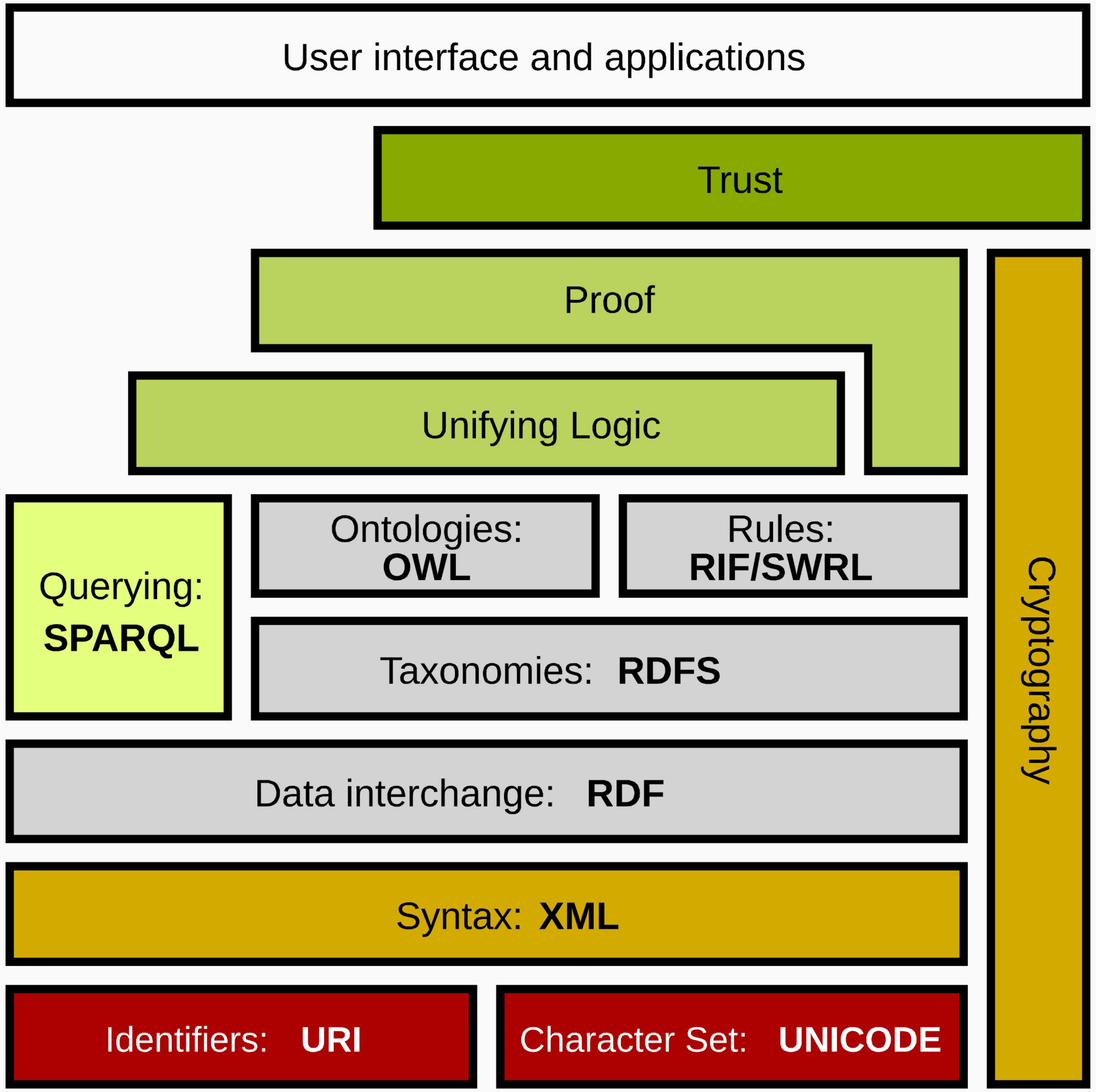
Web and the Semantic web

- World Wide Web (the Web)
  - Document as a basic unit
  - *Web of documents*
- Semantic Web
  - Data as a basic unit
  - *Web of Data*, *Linked data*

Semantic web technology

- Standard web technology
  - HTTP, URI
- Means for knowledge representation
  - Data representation (facts)
    - XML, RDF, ...
  - Sharing semantics
    - Ontologies
    - Languages for their explicit representation

Semantic Web Stack



Semantic web data model – RDF

Facts representation and exchange

Objectives and means

- Objectives
  - Representation of structured data and its meaning (semantics)
  - Ability to share data and its semantics across applications
- Common data representation in IS:
  - Relational/object/NoSQL databases - bound to particular application
  - Public API + serialization (JSON, XML) - no semantics formally defined

Data exchange - example

```xml
<flats>
    <item>
        <size>3 rooms + kitchen</size>
        <locality>Brno-střed</locality>
        <price currency="czk">8 200 000</price>
    </item>
    <item>
        <size>3 rooms</size>
        <locality>Valladolid</locality>
        <price currency="eur">342 500</price>
    </item>
</flats>
```

Problems

- The meaning of the elements is specific to the application
  - It is defined in the program code that generates or reads serialized data
  - Similar to columns in a relational database
- Other applications may assign a different meaning to the same elements
  - E.g. `<size>2 rooms<size>` vs. `<size>55 sqm</size>`
- Data is machine readable but not machine understandable
- We need a more formal way of representing **facts** and their **semantics** (meaning)

Representation of semantics

- Differentiating tags among applications
  - E.g. XML namespaces
  - Resolves tag collisions – a syntactic problem
- Separate definition of tag meaning
  - E.g. an accompanying document explaining meaning and use cases
- We need to define semantic relationships
  - E.g. a flat is a thing that has location, size and price
- **Ontology**
  - Formal definition of *concepts* and their *relationships*.

Representation of facts

- XML
  - Mapping elements to ontological concepts
  - Hierarchical structure only - restrictive
- RDF
  - A W3C standard
  - Abstract model
  - Graph structure
  - Can be written using XML or other languages

RDF triple

- A **RDF triple** is a basic unit

subject – **predicate** – **object**

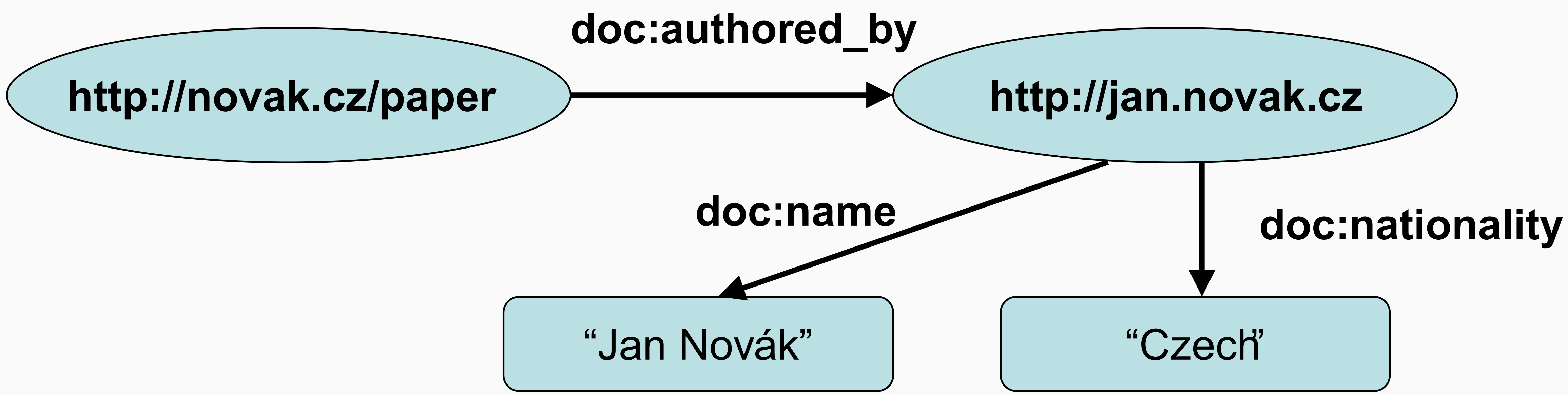
- a single *statement*.

RDF triple – a statement

- **document X** *is authored by* **Mr. Y**
  - Subject: **document X**
  - Predicate: *is authored by*
  - Object: **Mr. Y**
- The individual components *(resources)* are represented by **URIs** or **literals**.

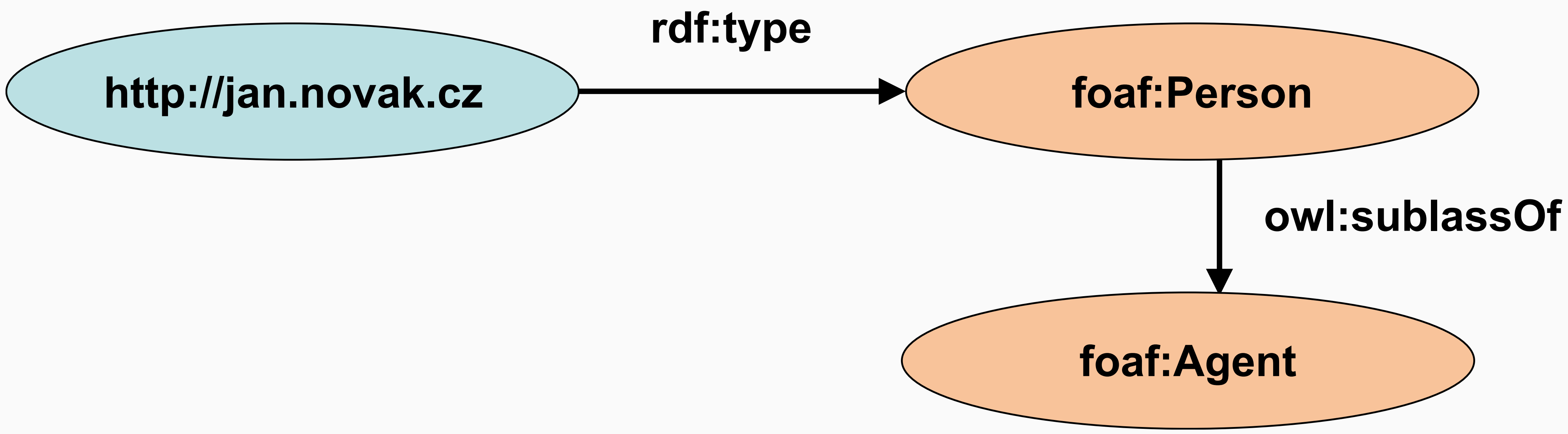
RDF statement (II)

RDF Graph



- The RDF graph can be decomposed into subject - predicate - object triples
- Subject and predicate are always **URIs**
  - `doc:` is a *URI prefix* that is expanded
  - E.g. `doc:name` => `http://my.docs.com/#name`
- The object is a **URI** or **literal** (of different data types)

Schema – an Ontology



- RDF data can be linked to metadata (ontology, schema)
  - Using the `rdf:type` predicate
- Metadata definition using RDF again
  - It is possible (but not necessary) to combine data and metadata into a single graph.

RDF data storage and transfer

- Saving to an RDF repository (e.g. RDF4J)
  - Decomposing into triples and storing in an internal structure
  - Possible querying (e.g. SPARQL language)
- Serialization and deserialization - several variants
  - RDF/XML (W3C standard)
  - N-triples (N3)
  - Turtle (subset of N3)

Turtle serialization

```
@prefix doc: <http://dokumenty.cz/def#> .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .

<http://novak.cz/paper>
    doc:authored-by <http://jan.novak.cz> .

<http://jan.novak.cz>
    doc:name "Jan Novák" ;
    doc:nationality "Czech" ;
    a foaf:Person .
```

XML serialization

```
<rdf:RDF
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:doc="http://dokumenty.cz/def#">

    <rdf:Description rdf:about="http://novak.cz/paper">
        <doc:authored-by
            rdf:resource="http://jan.novak.cz" />
    </rdf:Description>

    <rdf:Description rdf:about="http://jan.novak.cz">
        <doc:name>Jan Novák</doc:name>
        <doc:nationality>Czech</doc:nationality>
```

RDF as a database

- Repository - RDF triples repository
- Querying - SPARQL language
- Local repository:
  - Virtuoso http://virtuoso.openlinksw.com/
  - RDF4J (formerly Sesame) http://rdf4j.org/
  - Blazegraph https://www.blazegraph.com/product/
- Global *knowledge bases*
  - DBPedia http://dbpedia.org
  - http://dbpedia.org/resource/Berlin
  - http://dbpedia.org/sparql

Querying – SPARQL

- The query result may be
  - CSV (a table) – SELECT query
  - or a new RDF graph – CONSTRUCT query

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX yago: <http://dbpedia.org/class/yago/>
PREFIX dbpedia-owl: <http://dbpedia.org/ontology/>
PREFIX dbprop: <http://dbpedia.org/property/>
SELECT ?place ?name ?label WHERE {
    ?place rdf:type dbpedia-owl:Country .
    ?place dbprop:commonName ?name .
    ?place rdfs:label ?label .
   OPTIONAL {?place dbprop:yearEnd ?yearEnd}
   FILTER (!bound(?yearEnd))
}
```

Public knowledge bases

- DBPedia http://dbpedia.org
  - http://dbpedia.org/resource/Berlin
  - http://dbpedia.org/sparql
- Wikidata http://wikidata.org
  - http://wikidata.org/entity/Q42
- Many others
  - Can be linked together using URIs
  - *Linked open data*
  - http://lod-cloud.net/

Ontologies

Dictionaries for the semantic web

The notion of ontology

- Originally, more general meaning (philosophy)
- A means of sharing the meaning of concepts that occur in the target domain
- *"Formal, explicit specification of a shared conceptualization"*
- Defines the basic concepts of the modeled world and the relationships among them
- **Shared** and **reusable**

Purpose of ontologies

- Understanding between people (experts)
- **Understanding between computer applications**
  - **Giving meaning to individual URIs in the Semantic Web**
  - Ability to **integrate** data from different sources
- Design of knowledge-based applications

Types of ontologies

- Terminological (lexical)
  - Concepts and their relationships (taxonomy)
  - E.g. *WordNet*
- Generic ontologies
  - Regularities and relationships between generic concepts
  - „Upper ontology“, e.g. SUMO
- Domain ontologies
  - Domain specific (e.g. enterprise, medical, ...)
- Application ontologies
  - For a specific application

Ontology members

- **Classes (concepts)**
- **Individuals (objects, instances)**
- **Properties (roles, attributes)**
- Meta-slots (facets)
- Primitive data types
- Axioms (rules)

Concepts – classes

- Sets of specific objects
- No procedural methods
- Inheritance of classes (often multiple)

Individuals – objects – instances

- Concrete real world objects
- An individual is not necessarily an instance of a class
- Due to the purpose of ontologies, they are not used often
  - They represent specific data

Relations – attributes – slots – properties

- Property = relation
  - A separately defined element
  - Usually a binary relation
- Possible inheritance of relations (has father, has ancestor)
  - Parent relation contains all elements of the child relation
- A function – a specific relation
  - Value of argument *n* uniquely determined by the previous *n-1* arguments

Primitive values, data types

- The relation argument can be a *primitive value* (not an object)
  - Number, string, enumeration value, ...
  - *Datatype property* vs. *object property*
- We can consider datatype classes (data types) and datatype instances (values)
- We usually declare datatype slots as functional (they have only one value)

Ontological languages

RDF Schema, OWL

RDF Schema

- Provides basic elements for the description of ontologies
  - Actually a (meta) **ontology**
- Provides a vocabulary for defining
  - Classes
  - Binary relations (domain, range)
  - Class and property hierarchy
- RDF statements are used for defining the ontology
  - Containing the concepts and properties from RDFS
- Namespace (usually the **rdfs** prefix)

  `http://www.w3.org/2000/01/rdf-schema#`

Classes

- A class is assigned to a resource using `rdf:type`
  - `school:Person rdf:type rdfs:Class`
- Sub-classes
  - E.g. `school:Student rdfs:subClassOf school:Person`

Properties in RDFS

- Properties are instances of `rdfs:Property`
  - `school:hasRegistered rdf:type rdfs:Property`
- **rdfs:Range** – object range
  - `school:hasRegistered rdfs:range school:Course`
- **rdfs:Domain** – subject range (domain)
  - `school:hasRegistered rdfs:domain school:Student`
- `rdfs:subPropertyOf`
  - a sub-property

OWL

- Additional properties in addition to RDFS
- Complete ontology definitions
- Namespace `http://www.w3.org/2002/07/owl#`

Classes in OWL

- Possible combinations with RDFS
- A class can be defined using
  - Class identifier (no elements)
  - An enumeration of elements (instances)
  - By restricting properties
  - Unification or intersection of two or more classes
  - Complement

Class definition by identifier

Turtle

```
foaf:Person rdf:type owl:Class .
```

In Turtle (icluding prefixes)

```
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix owl: <http://www.w3.org/2002/07/owl#> .
@prefix foaf:<http://xmlns.com/foaf/0.1/>.

foaf:Person rdf:type owl:Class .
foaf:Person a owl:Class .
```

Definition by complement

```
<owl:Class>
  <owl:complementOf>
    <owl:Class rdf:about="#Student"/>
  </owl:complementOf>
</owl:Class>
```

Other class operators

- `owl:equivalentClass`
  - An equivalent class (e.g. from another ontology)
- `owl:disjointWith`
  - Disjoint class třída

Property definition

- RDFS constructors

```
<owl:ObjectProperty rdf:ID="studies">
  <rdfs:domain rdf:resource="#Student"/>
  <rdfs:range rdf:resource="#Branch"/>
</owl:ObjectProperty>
```

- Property relationships
  - `owl:equivalentProperty` – same values
  - `owl:inverseOf` – inverse property

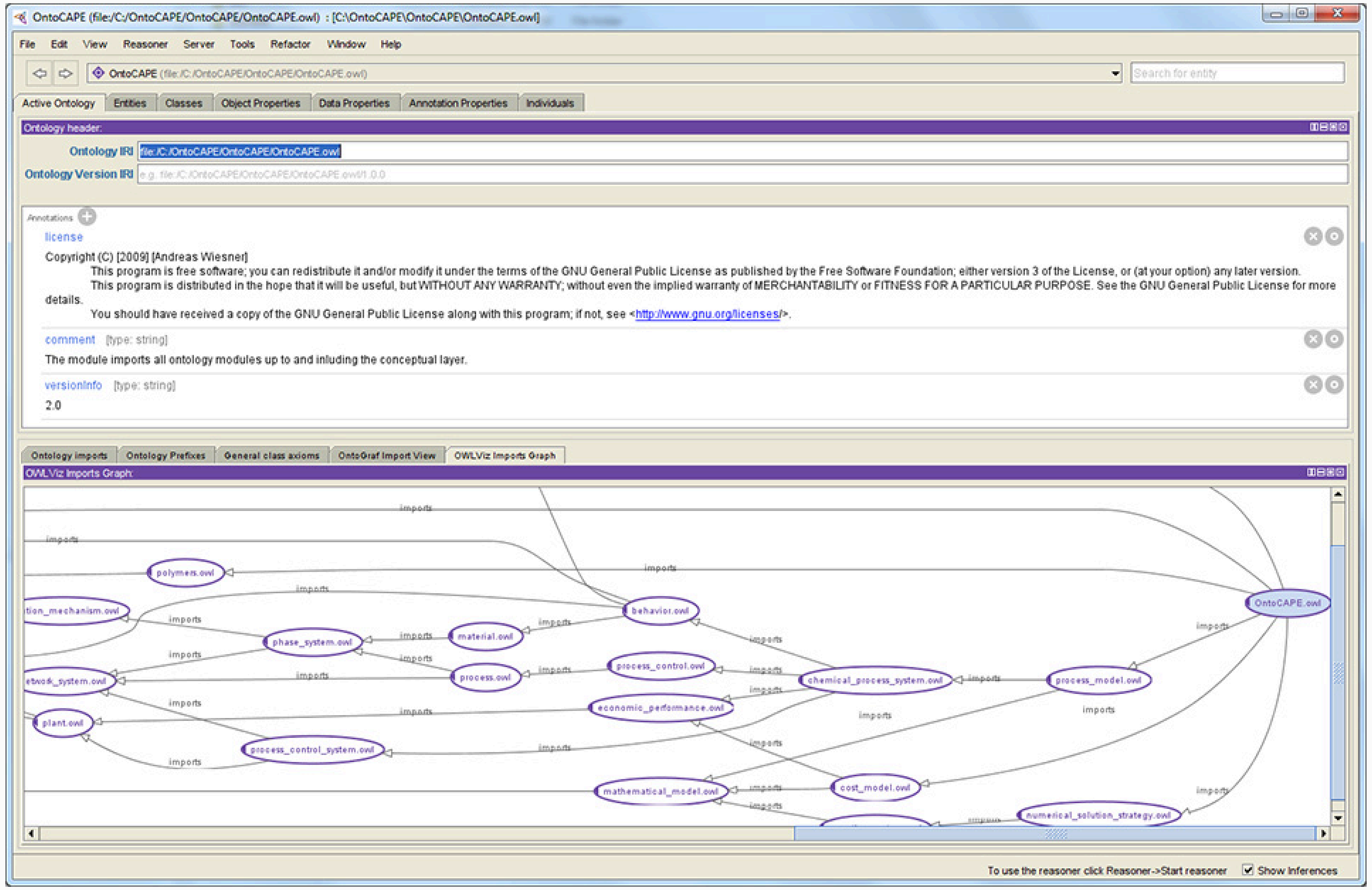
```
<owl:ObjectProperty rdf:ID="hasStudent">
    <owl:inverseOf rdf:resource="#studies"/>
</owl:ObjectProperty>
```

Data-type properties

- RDF Literals
- XSD (XML Schema) data types
  - Namespace `http://www.w3.org/2001/XMLSchema`
- `xsd:string`, `xsd:normalizedString`, `xsd:boolean`, `xsd:decimal`, `xsd:float`, `xsd:double`, `xsd:integer`, `xsd:nonNegativeInteger` `xsd:positiveInteger`, `xsd:nonPositiveInteger`, `xsd:negativeInteger`, `xsd:long`, `xsd:int`, `xsd:short`, `xsd:byte`, `xsd:unsignedLong`, `xsd:unsignedInt`, `xsd:unsignedShort`, `xsd:unsignedByte`, `xsd:hexBinary`, `xsd:base64Binary`, `xsd:dateTime`, `xsd:time`, `xsd:date`, `xsd:gYearMonth`, `xsd:gYear`, `xsd:gMonthDay`, `xsd:gDay`, `xsd:gMonth`, `xsd:anyURI`, `xsd:token`, `xsd:language`, `xsd:NMTOKEN`, `xsd:Name`, `xsd:NCName`

Protegé editor

http://protege.stanford.edu/



Existing ontologies

- Emphasis on maximum use of existing ontologies
  - It is possible to combine concepts and properties from different ontologies
- Overview
  - https://lov.linkeddata.es/dataset/lov/

Dublin core

- Document metadata
- Defines the properties of documents:

```
<rdf:Description rdf:about="http://www.w3schools.com">
   <dc:description>W3Schools</dc:description>
   <dc:publisher>Refsnes Data as</dc:publisher>
   <dc:date>2008-09-01</dc:date>
   <dc:type>Web Development</dc:type>
   <dc:format>text/html</dc:format>
   <dc:language>en</dc:language>
 </rdf:Description>
```

Friend-of-a-friend (FOAF)

- Ontology for describing persons and their relationships http://www.foaf-project.org/
- Classes for describing persons
  - `foaf:Agent`, `foaf:Person`, …
- Properties
  - `foaf:name`, `foaf:knows`, …

FOAF example

```
@prefix foaf:<http://xmlns.com/foaf/0.1/>.
@prefix dbr:<http://dbpedia.org/resource>.

dbr:Luke_Skywalker foaf:knows dbr:Han_Solo .
dbr:Luke_Skywalker foaf:name "Luke Skywalker" .
```

SKOS

- Simple Knowledge Organization System
- Allows the organization of concepts in a domain
  - Concept definition
  - Relationships between them: broader, narrower, related, ...
  - ...

Schema.org

- Primarily for web page annotation
  - https://schema.org
- Basic dictionaries for different domains
  - https://schema.org/docs/gs.html#schemaorg

Other ontologies

- Music ontology
  - http://musicontology.com/
- Event ontology
  - http://motools.sourceforge.net/event/event.html
- Time ontology
  - http://www.w3.org/TR/2006/WD-owl-time-20060927/
- Geo ontology
  - http://www.w3.org/2003/01/geo/

Ontologies and RDF knowledge bases

- DBPedia.org
  - Its own ontology + reusing the existing ones
  - http://dbpedia.org/resource/Berlin
  - http://dbpedia.org/page/Novak_Djokovic
- E.g.
  - Birth place property
  - Wikidata equivalent

RDF on the Web

Web of Documents vs. Web of Data

Semantic Annotations

- Links from HTML to semantic web concepts (URIs)
- Several standards exist
  - RDFa, HTML5 Microdata, JSON-LD
- Common crawl corpus
  - 2020: 50% (out of 3.4 billion) pages, 44.3% crawled domains
  - 2019: 37.9% (out of 2.45 billion) pages, 37.2% crawled domains

```
<div itemscope itemtype="https://schema.org/Person">
    <span itemprop="name">Jane Doe</span>
    <span itemprop="jobTitle">Professor</span>
    <div itemprop="address" itemscope itemtype="https://schema.org/Pos
        <span itemprop="streetAddress">
          20341 Whitworth Institute
          405 N. Whitworth
        </span>
        <span itemprop="addressLocality">Seattle</span>,
        <span itemprop="addressRegion">WA</span>
        <span itemprop="postalCode">98052</span>
    </div>
</div>
```

Bizer, C.; Meusel, R.; Primpeli, A.: Web Data Commons - RDFa, Microdata, and Microformat Data Sets - Extraction Results from the September 2021 Common Crawl Corpus.

The entire document annotation

```
<html xmlns="…">
<head>
<rdf:RDF
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:dc="http://purl.org/dc/elements/1.1/">

  <rdf:Description rdf:about="http://www.about.me/"
    dc:creator="John Smith"
    dc:title="Document title"
    dc:description="A description"
    dc:date="2021-09-10"/>
```

RDF and HTML integration

- HTML 5 – *Microdata*
- W3C standard – *RDFa*
- JSON notation – *JSON-LD*
- See e.g. https://schema.org/Person#examples

Another example – RDFa

```
<p xmlns:dc="http://purl.org/dc/elements/1.1/"
   about="http://www.example.com/books/wikinomics">

  In his latest book
  <cite property="dc:title">Wikinomics</cite>,
  <span property="dc:creator">Don Tapscott</span>
  explains deep changes in technology,
  demographics and business.
  The book is due to be published in
  <span property="dc:date" content="2006-10-01">October 2006</span>.

</p>
```

Event RDFa example

An event (a conference)

```
<div xmlns:event="http://www.w3.org/2002/12/cal#" typeof="event:Vevent">
     <h3 property="event:summary">WWW 2009</h3>
     <p property="event:description">18th International World Wide Web Confe
     <p>To be held from
        <span property="event:dtstart" content="2009-04-20">20th April 2009<
        until <span property="event:dtend" content="2009-04-24">24th April</
        in <span property="event:location">Madrid, Spain</span>.</p>
</div>
```

The `event:whatever` attribute values are shortened URIs `http://www.w3.org/2002/12/cal#whatever` (not necessarily a functional link to the WWW, it is just an identifier).

RDFa processing

1. RDFa parser
    - finding the elements and attributes in HTML
2. Representation of facts in the RDF way
    - Set of *subject – predicate – object* triples
3. Processing
    - Storage
        - RDF repository (*triple store*)
    - Serialization
        - Turtle, RDF/XML, JSON-LD, …

https://www.w3.org/2012/pyRdfa

Alternatively: JSON-LD

```
<html>
  <head>
    <title>WWW 2009</title>
    <script type="application/ld+json">
    {
        "@context": ""http://www.w3.org/2002/12/cal#",
        "@type": [ "Vevent" ],
        "description": "18th International World Wide Web Conference",
        "dtend": "2009-04-24",
        "dtstart": "2009-04-20",
        "location": "Madrid, Spain",
        "summary": "WWW 2009"
```

https://json-ld.org/

Google Structured Data

- Google is able to process structured data in HTML pages
- Both RDFa and JSON-LD
- Understands many schema.org vocabularies
- E.g. Produts, Movies, Recipes, ...

That's all!

Questions?