Ontologies and Semantic Web

Demián Tamás

Agenda

- Ontologies
- RDF and Semantic Web

ONTOLOGIES

Ontology overview

- Classic meaning: „the study of existence"
- For us: **Computer representation of domain knowledge**
- Created by
  - **domain experts**
  - **knowledge engineers**

  They identify the **concepts** to categorize **individuals** and the **relationships** that can hold between individuals (besides other kind of **axioms**)
- Ontology $\cong$ taxonomy + relationships

Ontologies overview 2

Main types:

- Domain ontologies (focuses on the given domain)
- Upper ontologies (most common concepts across vide range of domains)

In a different context, ontologies have other characteristics:

- Open world semantics (no default closure axiom)
- NO unique name assumption

(unlike in MetaModels, Domain Specific Languages)

Knowledge Representation Languages: **OWL**, RDF, CL, CASL, KIF, ... Typically based on **Description Logic** (DL) or First-order Logic (FOL)

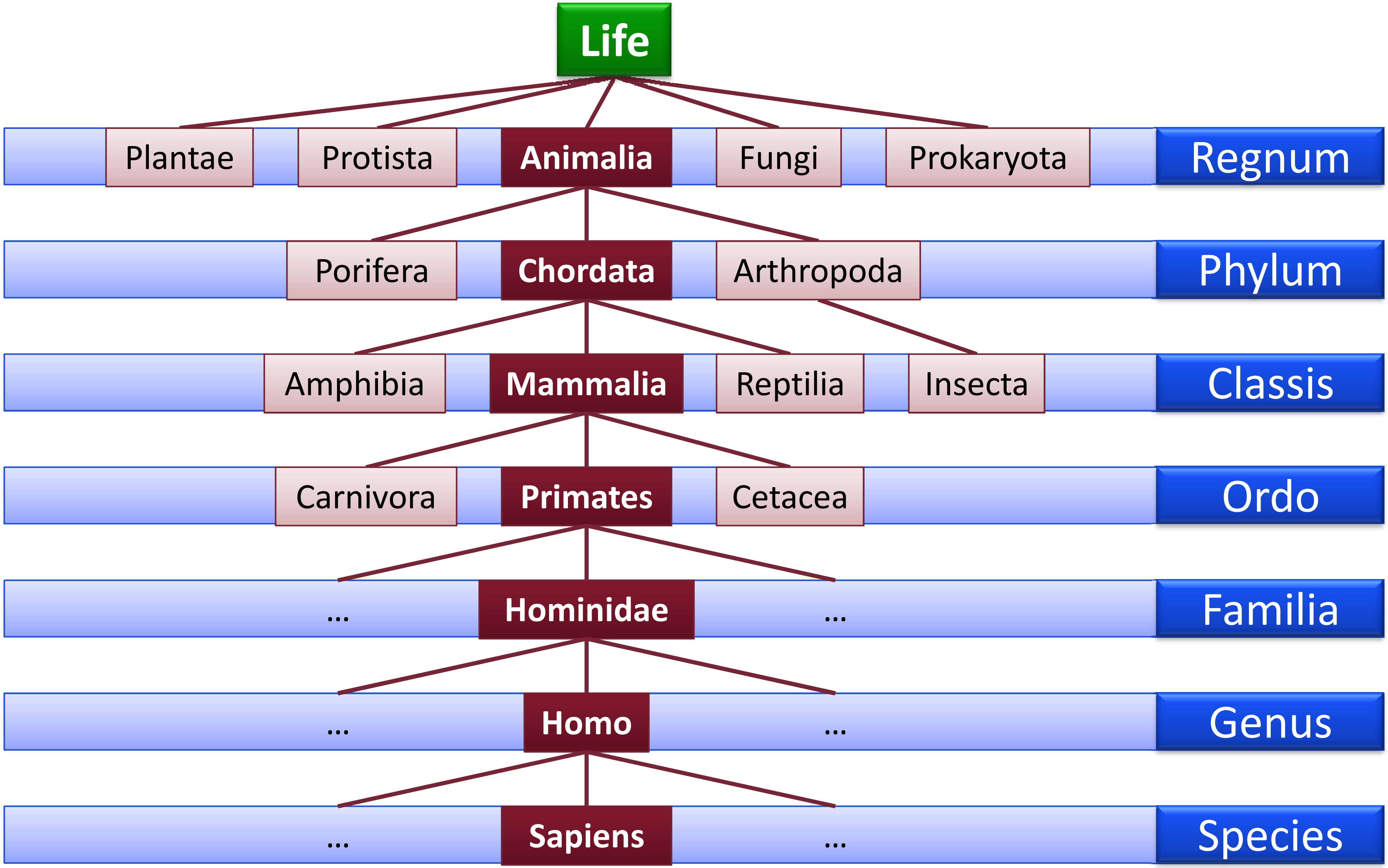
Comparison of terminologies

Mathematics	Description Logic (DL)	OWL-DL	RDF	UML	EMF
Set	Concept	Class	rdfs:Class	Class	Eclass
Relation	Role	Property	rdf:Property	Reference	Ereference
-	-	-	-	Attribute	Eattribute
Element Member	Individual	Object	rdf:Resource	Object	Eobject
Labeled, directed multi-graph	T-box (terminological axioms)	Only in DL, Lite dialects, Full uses MultiLevelModels	RDF scheme	Class diagram	Ecore
Labeled, directed multi-graph	A-box (assertional axioms)	Only in DL, Lite dialects, Full uses MultiLevelModels	RDF model	Object diagram	...

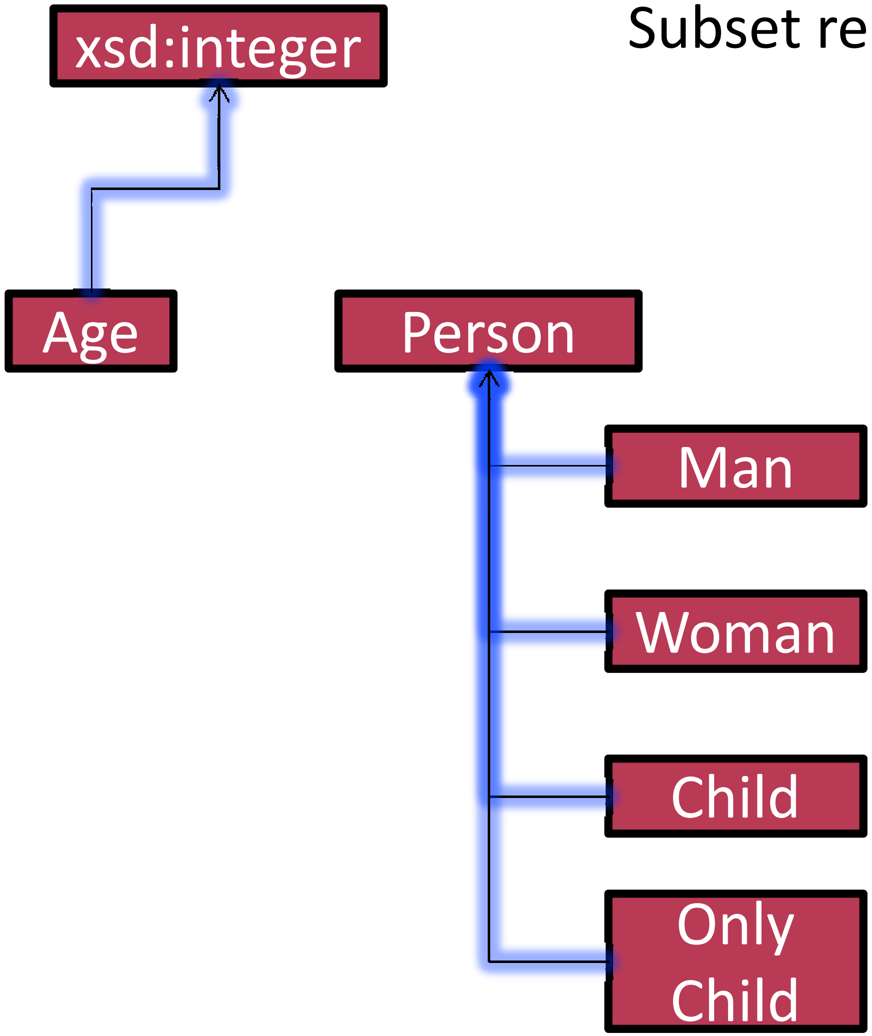
There is no strict distinction between attributes and relations in ontologies

In fact, it is a modeling freedom.

Example0 / A Classical Taxonomy



Example / Taxonomy



Subset relations only

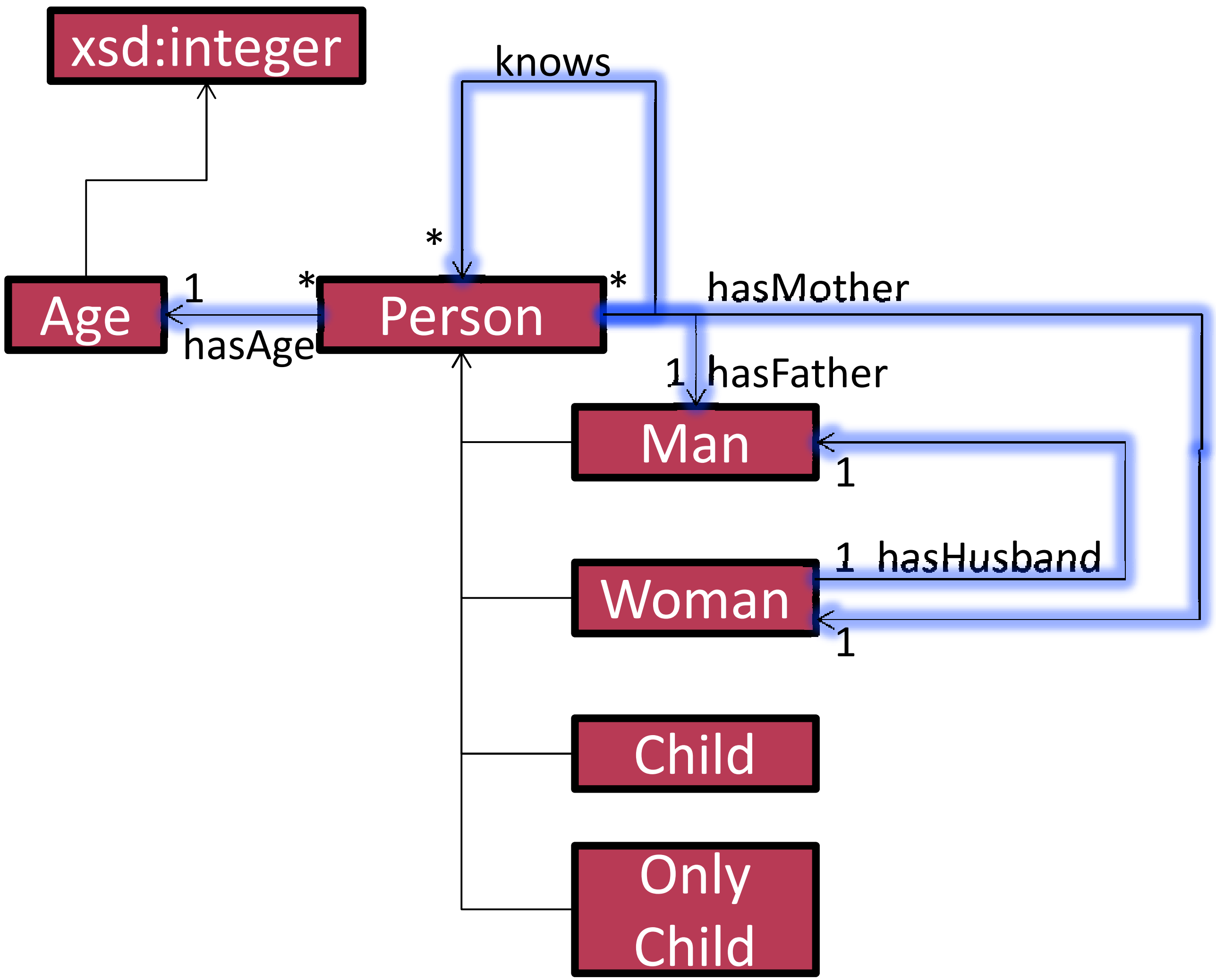
Concepts of a given level give not necessarily a partition

Corresponding axioms:

Subsumption:

Age $\subseteq$ xsd:integer
Man $\subseteq$ Person
Woman $\subseteq$ Person
Child $\subseteq$ Person
Only Child $\subseteq$ Person

Example / Relationships



Corresponding axioms:
(NOT in OWL or DL syntax)

Cardinality restrictions:
hasAge, hasMother, hasFather, hasHusband relations are **functions**
card(hasAge)=1, ..

Inverse statments:
$hasHusband^{-1}=hasWife$
$knows^{-1}=knows$

Role hierarchy:
hasMother $\subseteq$ hasParent
hasFather $\subseteq$ hasParent

Example / Other axioms

Concepts are often ambiguous in the natural languages. E.g. Is the (Child $\cap$ Woman) set empty?

The next two rows resolves this ambiguity:

- Man $\cap$ Woman = $\emptyset$, Man $\cup$ Woman = Person
  (the two genders form a partition)
- Child = hasAge<14
- OnlyChild = $\forall$parent.$\exists_{=1}$child
- Other Constraints: Parents are at least 18 years, Age<150, the graph of taxonomy is a DAG, ...

CLOSURE AXIOM

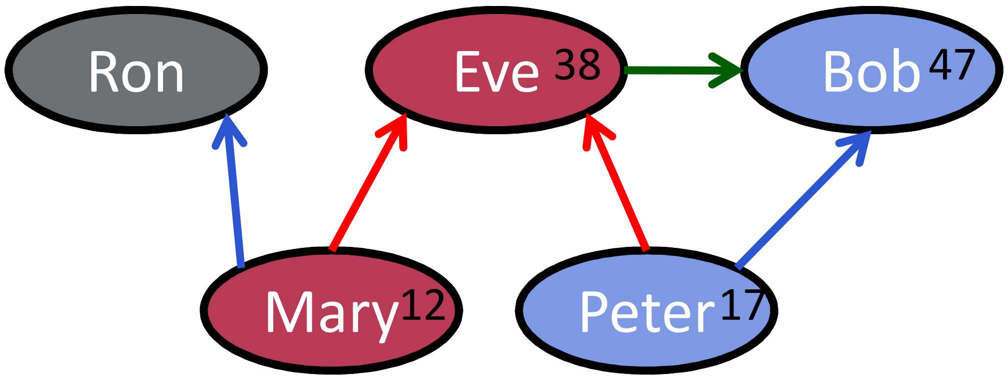
WARNING: In the world of ontologies, "axioms" also include the theories derived from axiomatic statements!

Example / Instances and Reasoning

Just the axioms:
Bob:Man
Eve:Woman
Peter:Man
Mary:Woman
Ron:Person

Bob hasAge 47
Eve hasAge 38
Peter hasAge 17
Mary hasAge 12

Peter hasFather Bob
Peter hasMother Eve
Mary hasMother Eve
Eve hasHusband Bob
Mary hasFather Ron



Inferred axioms(!):
T-box is consistent
Ron.hasAge>30
Ron:Man
...

We CAN'T deduce:
Bob knows Ron
MOREOVER:
Eve knows Bob!!!
...

Knowledge Representation Languages

- **Expressivity** vs. **Reasoning complexity**
- Languages typically based on Description Logics (**DL**)
  - o Family of logic languages with varying expressive power : it has many dialects like $\mathcal{SHOIQ(D)}$, $\mathcal{SHIQ}$, $\mathcal{SHIN}$, $\mathcal{ALCN}$, etc. where the letters encode the allowed operators
  - o Reasoning with tableau calculi
- DL-s are usually weaker than FOL (first-order logic), the complexities typically exceed the NP class (practically infeasible for bigger models)

Why we represent knowledge?

- To make domain assumption explicit
- To store and search data
- To share knowledge between and within domains
- To share data with others
- For deeper understanding of the domain

The main Reasoner services:

- Consistency checking (is the class empty?)
- Inferred class hierarchy

OWL dialects

- OWL (Web Ontology Language) is a **W3C standard**
- OWL is intended to be used over the Web, all its elements are defined as RDF resources, and identified by URIs. (see later)
- OWL tools: **Protégé**, ...
- Reasoners: **Pellet**, Racer, HermiT, ...
- OWL includes 3 dialects for scalability:

  OWL Lite: too week (restrictions on OWL-DL)

  OWL DL: Description Logic-compatible: $\mathcal{SHOIN(D)}$

    (complete and decidable)

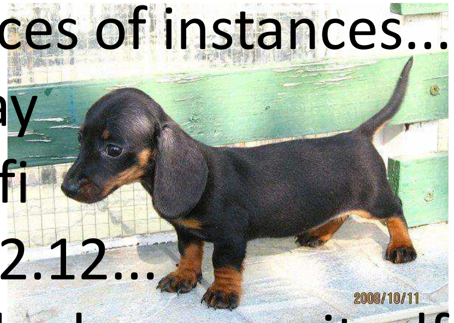
    disjointness of classes, properties, individuals and data values

OWL dialects 2

OWL Full allows free mixing of OWL with RDF Schema, so it does not enforce a strict separation of classes, properties, individuals and data values

Multilevel Modeling (instances of instances...) e.g.:

- Lexical Category/Verb/play
- Dog breed/Dachshund/Fifi
- Weekday/Sunday/2010.12.12...



OWL Full is able to extend the language itself (the pre-defined (RDF or OWL) vocabulary)

It is unlikely that any reasoning software will be able to support complete reasoning for OWL Full.

OWL 2 has 3 dialects:

OWL 2 EL: polynomial time reasoning complexity

OWL 2 QL: easier access and query to data stored in databases

OWL 2 RL: is a rule subset of OWL 2

Open World Semantics

- **because something hasn't been stated to be true, it cannot be assumed to be false**
- it is assumed that the knowledge just hasn't been added to the knowledge base
- Traditional databases have Closed World Semantics

  E.g. if somebody is not explicitly stated as child of Eve, then he/she is NOT her child!

  - Examples
    - Let us suppose, we have an asserted ontology seen above!
    - Eve hasChild Bella? Unknown!
    - Is Clara the mother of Mary? No! (if we state explicitly that Clara≠Eve)! See the next slide)

    (we know that Eve is her mother, and that everybody has only one mother)

NO Unique Name Assumption

- **Two things can be the same, unless contradicted**
  - Eve is the mother of Mary
  - Mary knows Evelyn.
  - Are they two different people? (Eve and Evelyn)
  - explicit control: owl:sameAs, owl:differentFrom
    - -equivalent/disjoint classes/properties
    - -same/different individuals

- Why? Distributed knowledge gathering: merging ontologies from heterogeneous sources (sources with different education, ideology, knowledge, experiences.) varying or expanding domain

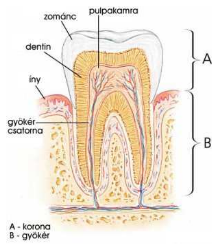
RDF AND SEMANTIC WEB

Metadata

- **Metadata**: description of data,
  - For people
  - For machines
- Example: image metadata
  - Generated partly automatically
  - „on this picture: John Doe, Jean-Baptiste Grenouille”
- Example: text document metadata
  - Author, literary category, year of publishing, etc.
- Metadata-based search

Syntactic Interpretation

- Can machines understand what we mean?
  - Textual / syntactic services can not
- Example: show me pictures depicting „fog”!



- Example: show me poems by female authors!
- Semantic solution
  - Machines should process the meaning, not the form
  - Use standardized concepts „fog”, „female”, „author”…
    - Refer to it in metadata and queries

Resource Description Framework

- W3C: Resource Description Framework (**RDF**)
  - **rdf:Resource** → something we talk about
    - a document (e.g. this photo)
    - a standardized meaning (e.g. tooth, Hungary)
    - identified by a URI
  - **rdf:Property** → relation type between resources
    - e.g depicts, taken_in etc.
    - also identified by a URI
  - Triplets → statements about properties of resources
- Open world, no unique names
- RDFS: RDF Schema

RDF Statements

- RDF statement = triplet
  - (resource, property, value)
  - resource, property are URIs
  - value: URI of other resource or raw data
- Example triplets
  - (this_photo, taken_in, Hungary)
  - (this_photo, file_name, „DSC00001.JPG”)
  - (this_photo, depicts, John Doe)
  - (this_photo, rdf:type, Photo)
  - (rdf:type, rdf:type, rdf:Property)

RDF Concrete Syntax

- Concrete syntaxes: **RDF+XML**, RDFa, N3, etc.

```
<rdf:RDF xmlns:… >
 <foaf:Person rdf:ID="#me">
    <foaf:name>Bergmann Gábor</f
    <foaf:gender>male</foaf:gender>
    <foaf:img rdf:resource="http://…/…jpg"/>
    <foaf:holdsAccount> <foaf:Onlin
      <foaf:accountServiceHomepage
"http://www.facebook.com/"/>
      <foaf:accountName>…</foaf:
    </foaf:OnlineAccount> </foa
    <foaf:knows>
      <foaf:Person rdf:about="#662…">
        <foaf:name>Zoltán Szatmári</foaf:name>
        <foaf:holdsAccount>…</foaf:holdsAccount>
        <foaf:based_near><geo:Feature>
          <geo:name>Budapest, Hungary</geo:name>
        </geo:Feature></foaf:based_near>
      </foaf:Person>
    …
</rdf:RDF>
```

(«base»:#me, rdf:type, foaf:Person)

(«base»:#me, foaf:gender, „male”)

(«base»:#me, foaf:img, «URL»)

(«base»:#me, foaf:knows, «Zee»)
(«Zee», rdf:type, foaf:Person)

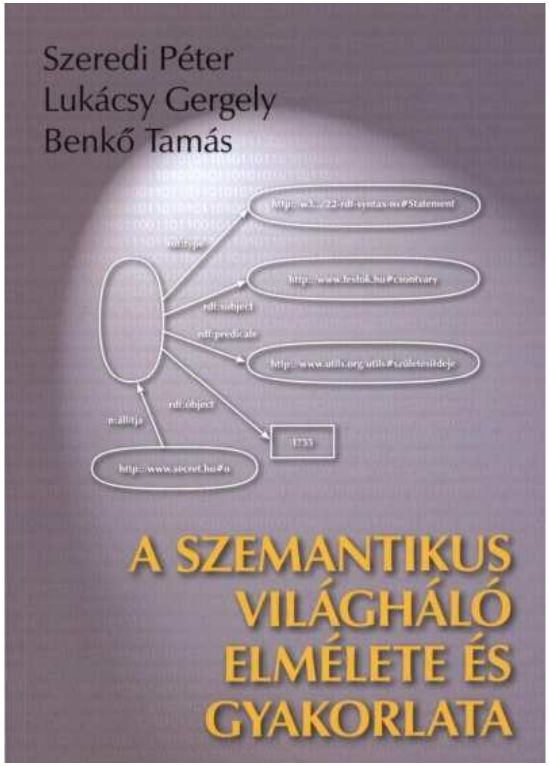
(«Zee», foaf:based_near, «nameless»)
(«nameless», rdf:type, geo:Feature)
(«nameless», geo:name, „Budapest…”)

RDF Application

- RDF Site Summary (**RSS**)
  - Items with title, description, link, creator, date, …
  - RSS 2.0 abandons RDF, backronym
- OWL itself is an RDF document
  - Classes, properties identifiable by URIs
- **Semantic Web**
  - Is a photo of my Porsche a photo of a car?
  - Need standard URIs for RDF resource/property types
  - Use OWL ontologies to provide type URIs
  - Local metadata + ontologies = semantic web

Recommended reading

Benkő-Szeredi-Lukácsy:
A szemantikus világháló elmélete és gyakorlata.
Typotex, 2005.



BMEVIMIM222
Információ- és tudásintegrálás
(MSc intelligens rendszerek szakirány)