AI INSTITUTE #AIISC
UNIVERSITY OF SOUTH CAROLINA

UNIVERSITY OF SOUTH CAROLINA

CSCE 771: Computer Processing of Natural Language
Lecture 14: Representation & Reasoning, Project Review

PROF. BIPLAV SRIVASTAVA, AI INSTITUTE

4TH OCTOBER, 2022

Carolinian Creed: "I will practice personal and academic integrity."

Acknowledgement: Used materials by NLTK, Russel & Norving, Khemani

Organization of Lecture 14

- Opening Segment
  - Quiz 2
- Main Lecture

Main Section
- Knowledge representation
- Project Review

- Concluding Segment
  - About Next Lecture – Lecture 15

Quiz 2

- Reduce and eliminate carelessness
- Examples
  - No one, apart from one, prefixed file name with their names
  - Some did not even write their names on their answers
  - Some wrote HW instead of Quiz
- Statistics
  - 14 received on time
  - Late submissions
    - 4 Within 1 hr of deadline - will get 50% credit only
    - 1 very late – will get 25% credit only
  - 3 did not submit
    - 1 given relief based on unique situation and submitted. No penalty

Recent Classes

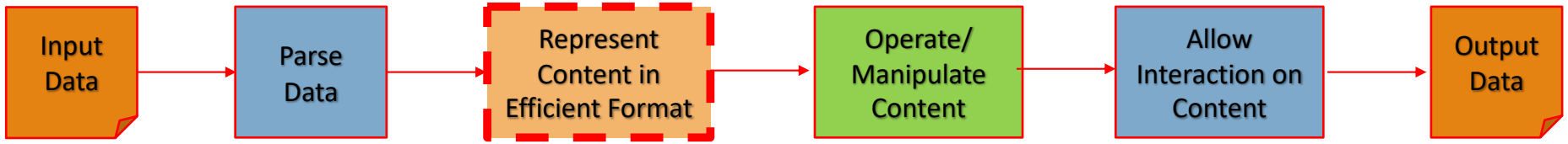
Sep 29 (Th)	Representation: Embeddings, Language Models, QUIZ
Oct 4 (Tu)	Review: Reasoning and Representation for NLP: Ontology, Knowledge Graph, PROJ REVIEW
Oct 6 (Th)	Entity extraction
Oct 11 (Tu)	Guest Lecture – Dr. Amitava Das: Using lang models to solve NLP tasks
Oct 13 (Th)	
Oct 18 (Tu)	Entity linking, Events extraction, spatio-temporal analysis
Oct 20 (Th)	Topic Analysis, QUIZ
Oct 25 (Tu)	NLP Task: Sentiment; Related papers presentation; PROJ REVIEW

Review of Lecture 13

- Word2Vec
- Glove
- Quiz 2

Main Lecture

Representation

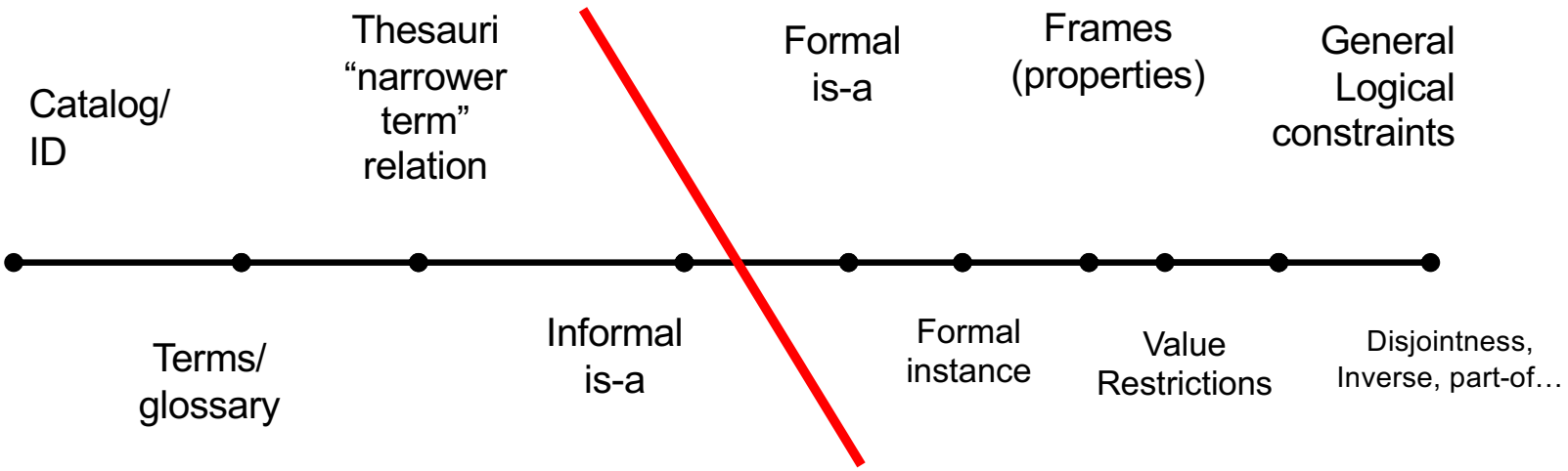


Formalizing Knowledge in an Ontology

Sources:
Achille Fokoue, Anastasios Kementsietsidis Tutorial
SCRIBE presentation by Rosario Usceda Sosa, Biplav Srivastava, Bob Schloss

- https://github.com/rschloss/ismp ,
- https://researcher.watson.ibm.com/researcher/view_group.php?id=2505

What is an Ontology?

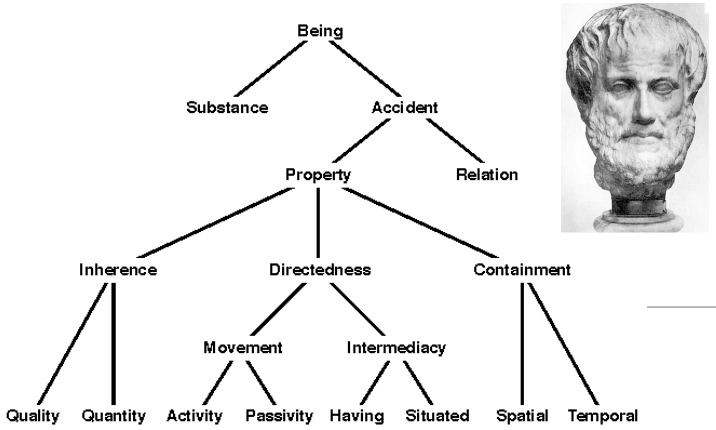


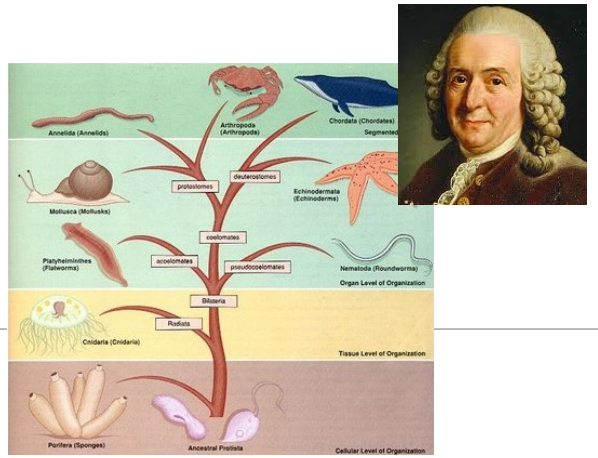
- Ontologies for information fusion, McGuinness, 2003
- Ontologies Come of Age – McGuinness, 2003, http://www.ksl.stanford.edu/people/dlm/papers/ontologies-come-of-age-mit-press-(with-citation).htm
- https://www.mkbergman.com/1842/conceptual-and-practical-distinctions-in-the-attributes-ontology/

What is an ontology?

In Computer Science, “An ontology is a formal explicit description of concepts in a domain of discourse (**classes** (sometimes called concepts)), **properties** of each concept describing various features and **attributes** of the concept (slots (sometimes called roles or properties)), and **restrictions** on slots (facets (sometimes called role restrictions)). An ontology together with a set of individual instances of classes constitutes a knowledge base. In reality, there is a fine line where the ontology ends and the knowledge base begins.” [Noy, 2000]

Not to be confused with ontologies (and/or taxonomies) in Philosophy or Life Sciences





Ontology = Class + Relations + Constraints

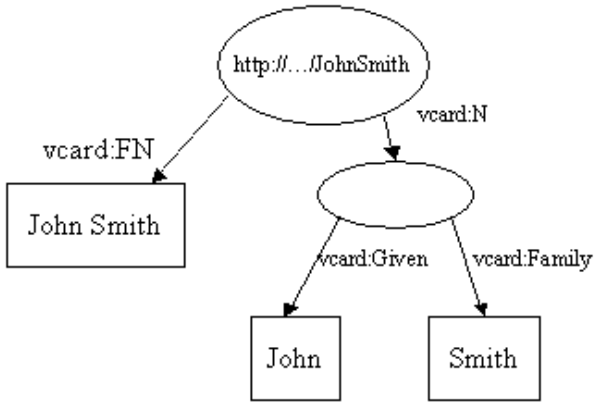
Knowledge Base = Ontology + instances + (Standard) Inference and rules

Slide courtesy: Rosario Usceda-Sosa

RDF / Turtle Example

vcard: https://www.w3.org/TR/vcard-rdf/

RFC Property	Note	Ontology Property	N-Ary Property
FN	The full name of the object (as a single string). This is the only mandatory property.	fn	hasFN
N	The name of the object represented in structured parts	hasName (range of class Name) given-name family-name additional-name honorific-prefix honorific-suffix	 hasGivenName hasFamilyName hasAdditionalName hasHonorificPrefix hasHonorifixSuffix
NICKNAME	A nickname for the object	nickname	hasNickname
PHOTO		hasPhoto	
BDAY	Birth date of the object. Should only apply to Individual.	bday	
ANNIVERSARY	Should only apply to Individual	anniversary	
GENDER	Should only apply to Individual. See Gender Codes in Section 2.11.	hasGender	




```
---- Turtle ----
<http://somewhere/JohnSmith>
    <http://www.w3.org/2001/vcard-rdf/3.0#FN>
          "John Smith" ;
    <http://www.w3.org/2001/vcard-rdf/3.0#N>
          [ <http://www.w3.org/2001/vcard-
     rdf/3.0#Family>
                "Smith" ;
           <http://www.w3.org/2001/vcard-
     rdf/3.0#Given>
                "John"
          ] .
```

```
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-
 syntax-ns#"
  xmlns:vcard="http://www.w3.org/2001/vcard-
 rdf/3.0#" >
 <rdf:Description rdf:nodeID="A0">
  <vcard:Given>John</vcard:Given>
  <vcard:Family>Smith</vcard:Family>
 </rdf:Description>
 <rdf:Description
 rdf:about="http://somewhere/JohnSmith">
  <vcard:FN>John Smith</vcard:FN>
  <vcard:N rdf:nodeID="A0"/>
 </rdf:Description>
</rdf:RDF>
```

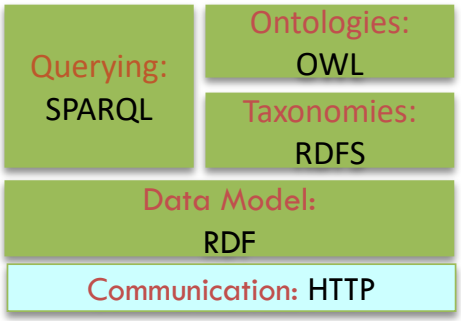
OWL extends RDF…

RDF-schema
- Class, subclass
- Property, subproperty

+ Restrictions
- Range, domain
- Local, global
- Existential
- Cardinality

+ Combinators
- Union, Intersection
- Complement
- Symmetric, transitive

+ Mapping
- Equivalence
- Inverse



Source: Achille Fokoue, Anastasios Kementsietsidis Tutorial

Not all ontologies are created equal

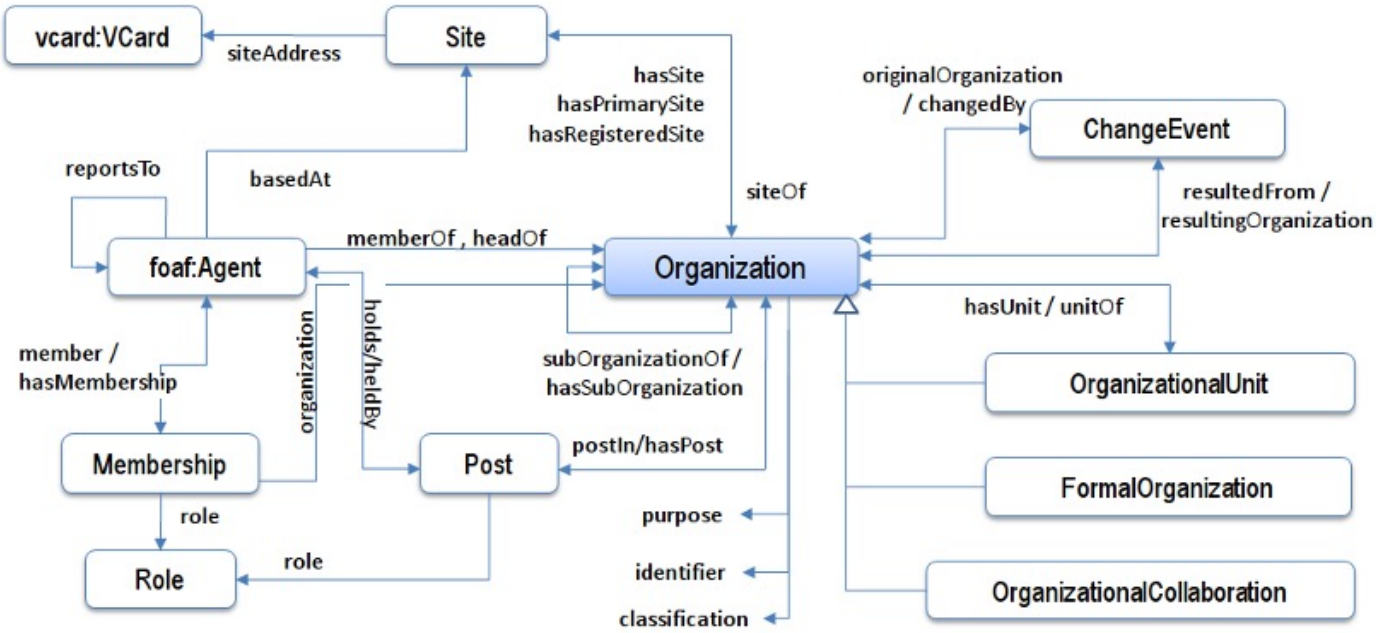
In practice, ontologies are used, together with inferencing engines and rules, for a variety of purposes. If we think of them as schemas, there are different ways

	Purpose	Instances	Inferencing	Examples
As a deductive system	Deductive System (axioms + deductive rules)	Part of the knowledge base	Defined by rules.	Expert systems, Planning, Optimization.
As a data blueprint	Constrain a domain	Must conform to the normative schema determined by the ontology	Subsumption, class inferencing	Biomedical and life sciences (FMA, Radlex)
As a data classifier	Classify open data	Unknown formats	Subsumption, class inferencing	Tag ontologies (MOAT, Echarte, SCOT, NAO, etc.)
As a data integrator	Integrating pre-defined model to existing data sources	Instances are mapped, no constraint enforcement.	Subsumption, class, entity inferencing	**SCRIBE**
As data mapping vocabulary	Mapping to/from existing data sources	Mined instances determine the ontology/schema.	Subsumption, class inferencing	D2RQ (a tool)

SCRIBE belongs to the **fourth** category: It has no constraints and was designed to support the programming of tools that allow domain experts to deal with entities natural to them (even if the recorded data is actually distributed).

https://github.com/rschloss/ismp

Larger Example: Organization Ontology



Ontology description: http://www.w3.org/TR/vocab-org/

Ontology: http://www.w3.org/ns/org.ttl

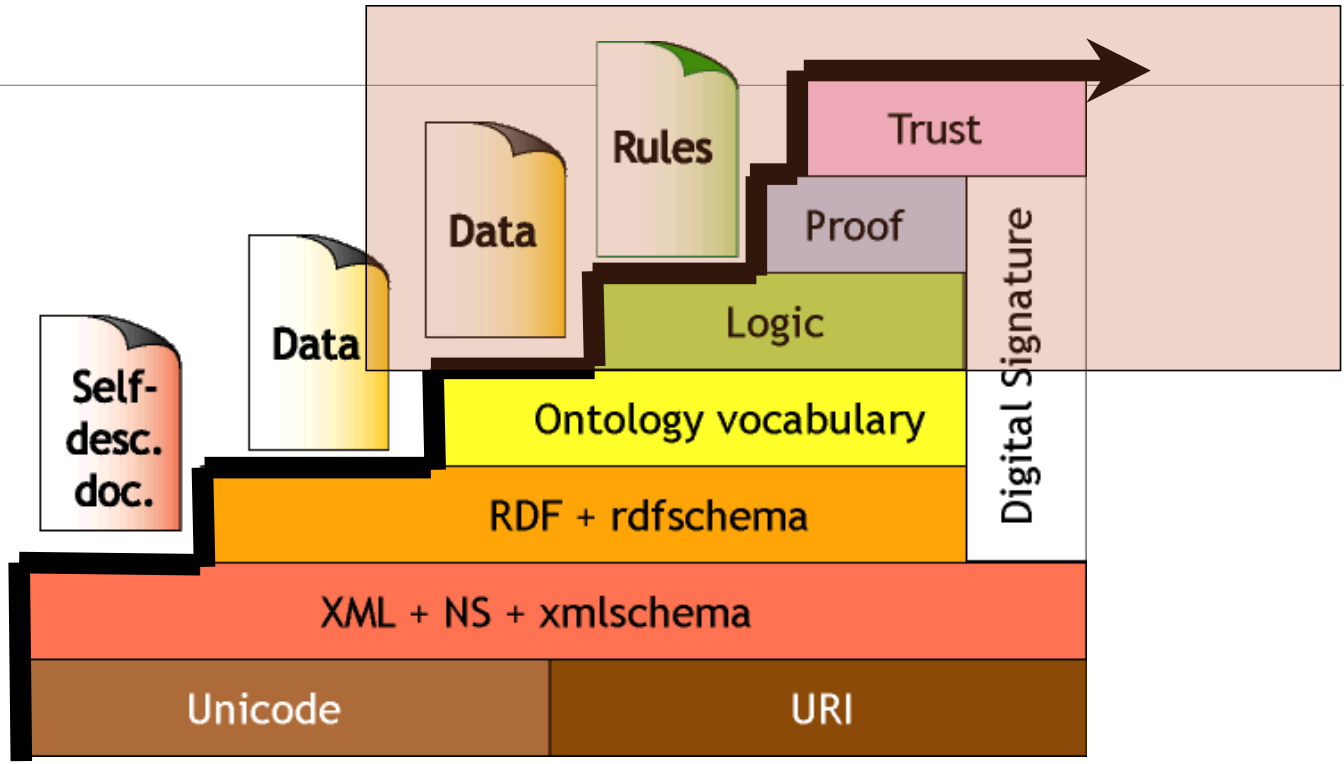
Larger Ontology

```
@prefix rdf:        <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs:       <http://www.w3.org/2000/01/rdf-schema#> .
@prefix owl:        <http://www.w3.org/2002/07/owl#> .
@prefix xsd:        <http://www.w3.org/2001/XMLSchema#> .
@prefix skos:       <http://www.w3.org/2004/02/skos/core#> .
@prefix foaf:       <http://xmlns.com/foaf/0.1/> .
…
@prefix :           <http://www.w3.org/ns/org#> .

# -- Meta data --------------------------------------------------------

<http://www.w3.org/ns/org#>
    a owl:Ontology;
    owl:versionInfo "0.7";
    rdfs:label "Core organization ontology"@en;
    rdfs:comment "Vocabulary for describing organizational structures,
specializable to a broad variety of types of organization."@en;
    dct:created "2010-05-28"^^xsd:date;
    dct:modified "2010-06-09"^^xsd:date;
    dct:modified "2010-10-08"^^xsd:date;
…
    rdfs:seeAlso <http://www.w3.org/TR/vocab-org/> ;
    .

# -- Organizational structure -----------------------------------------------
----------

org:Organization a owl:Class, rdfs:Class;
    rdfs:subClassOf  foaf:Agent;
    owl:equivalentClass foaf:Organization;
    rdfs:label "Organization"@en;
    rdfs:label "Organisation"@fr;
    owl:hasKey (org:identifier) ;
    rdfs:comment """Represents a collection of people organized together into a
community or other social, commercial or political structure. … Alternative
names: _Collective_ _Body_ _Org_ _Group_"""@en;
    rdfs:comment """Représente un groupe de personnes organisées en communauté
où tout autre forme de structure sociale, commerciale ou politique. … code
provenant d'une liste de code."""@fr;
    rdfs:isDefinedBy <http://www.w3.org/ns/org> ;
    .
```

http://www.w3.org/ns/org

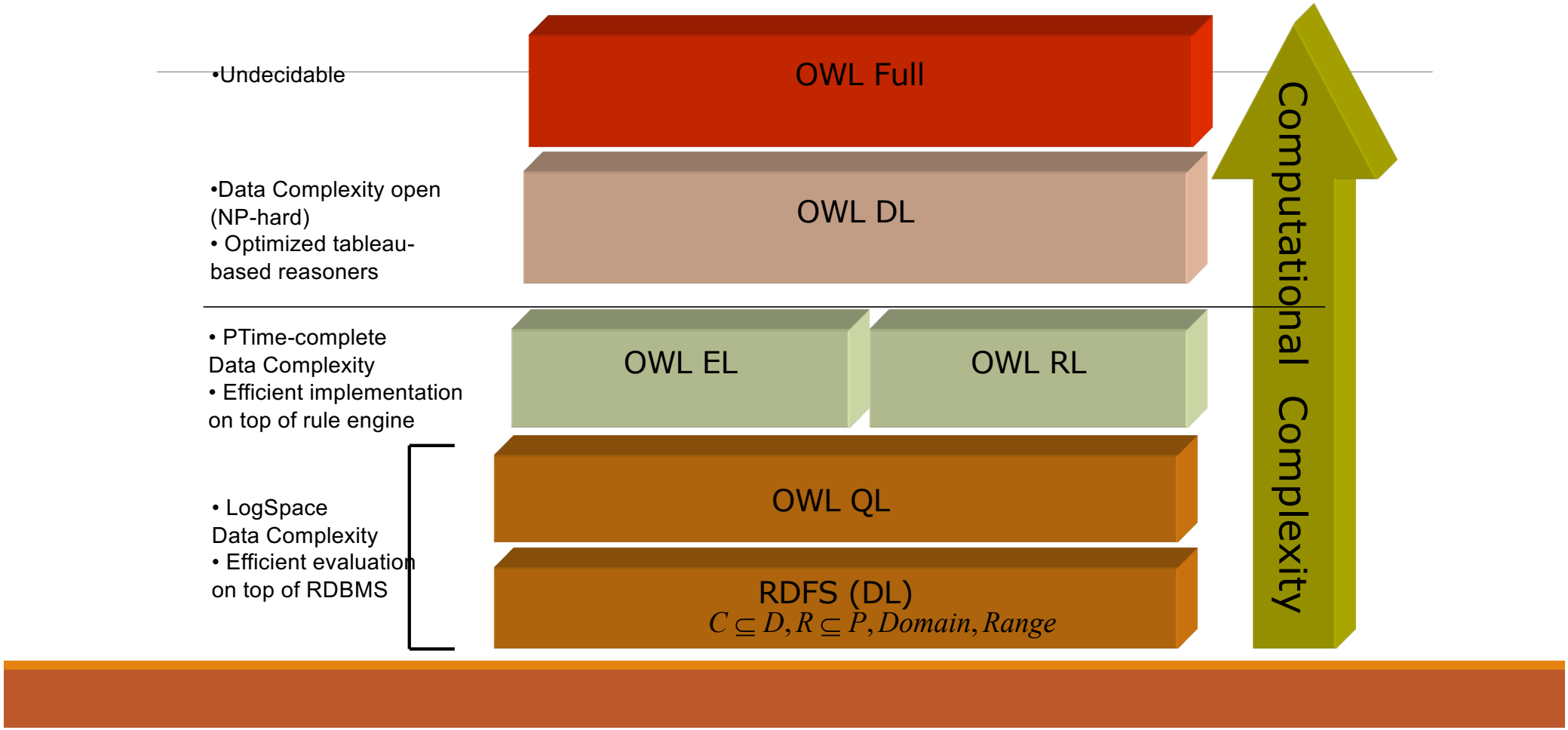
http://www.w3.org/ns/org.ttl

Moving to the futureof the web



Semantic Web LayerCake **(Berners-Lee, 99;Swartz-Hendler, 2001)**

Challenge of Reasoning on Ontologies



What makes a good ontology for data integration?

A *good* ontology is a *useful* ontology, an ontology that *both* humans and systems can process.

Human Usability	**Communicable**. Naming, natural language support, etc.
	Concise. A simple way to describe the key entities of the model and yet able to infer many facts
	Consistent. Naming conventions and modeling patterns
	Authoritative to domain experts
	Documented, not just descriptions, but also provenance
	Managed and maintained by people throughout the model lifecycle.
	Reusable in similar domains, for similar instances.
System Usability	**Scalable** so large amounts of data can be parsed, stored and retrieved.
	Efficient query and inferencing
	Programmable solutions, both in open and closed data paradigms.
	Open infrastructure and tools

- ***Formal** representation of **knowledge** in a particular domain*
- *Formally defines key **concepts** and **relations** in the domain*
- *Specifies relationships between those key concepts and relations*
- *Supports **automated reasoning** about entities in the domain*

Using Ontology

- Visually via tools like Protégé - https://protege.stanford.edu/
- Programmatically with APIs like
  - Jena (Java) - https://jena.apache.org/documentation/ontology/
  - OwlReady2 (Python) - https://bitbucket.org/jibalamy/owlready2/src/master/
  - Rdflib (Python) - https://github.com/RDFLib/OWL-RL
- A compendium of resources - https://github.com/totogo/awesome-knowledge-graph

Code Illustration

On Github:
https://github.com/biplav-s/course-nl/blob/master/l11-ontology/Exploring%20ontologies.ipynb

Project Review

Project Update Presentations

- W1 - Sep 26
- W2 – Oct 3
  - **Review presentation for class: 3 min each – Oct 4, 2022**
- W3 – Oct 10
- W4 – Oct 17
- W5 – Oct 24
  - **Review presentation for class: 3 min each – Oct 27, 2022**
- W6 – Oct 31
- W7 – Nov 7
- W8 – Nov 14
- W9 – Nov 21

Milestones

- Penalty: **not** ready by Sep 15, 2022 **[-20%]**
- Project report **not** ready by Nov 10, 2022 **[-20%]**
- Project presentations **not** ready by Nov 15, 2022 **[-10%]**

Format for Review Presentation Slide (2 mins)

- Project Name
- Problem
- Approach
- Status (based on your project plan)
- Comment:
  - Challenges faced
  - Need help

Test Case – *how will your program be run*

- Input
- Output
- Assumptions

Lecture 14: Concluding Comments

- We looked at ontology and knowledge graphs
- Connected it with reasoning capabilities provided
- Reviewed project status

Concluding Segment

About Next Lecture – Lecture 15

Lecture 15 Outline

- Knowledge graph/ Concept IO hands on
- Entity extraction