Intelligent Internet Technologies

Lecture 21.

Introduction to **OWL: Ontology Web Language**

Alexandra V. Vitko

KNURE, AI Department, alexandra_vitko@yahoo.com

OWL: what?

- Core of the World Wide Web Consortium's Semantic Web activity
- **W3C Recommendation 10 Feb 2004**
- In various senses a successor to previous work on "Web-friendly" knowledge modelling languages
  - RDF & RDF Schema
  - DAML-ONT
  - OIL / DAML+OIL

OWL: W3C definition

“language for defining **structured Web-based ontology**

which enable

richer **integration** and **interoperability of data** across application boundaries”

OWL General Goals and Requirements

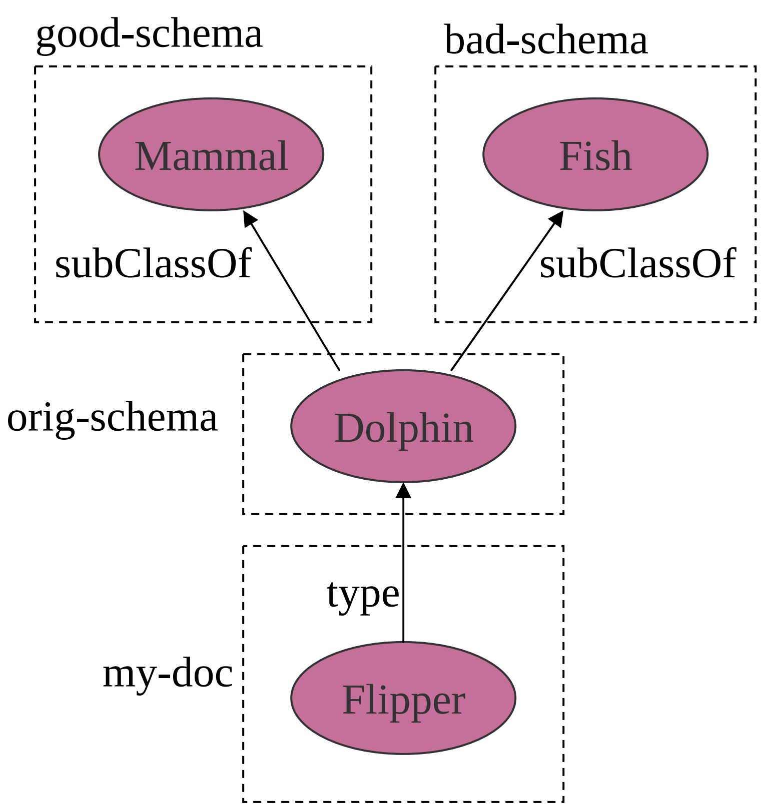
Goal 1. Shared Ontologies

- Ontologies are publicly available (at least read only (possibly read-write later)) and different data sources can commit to the same ontology for shared meaning.

Goal 2. Ontology Extension

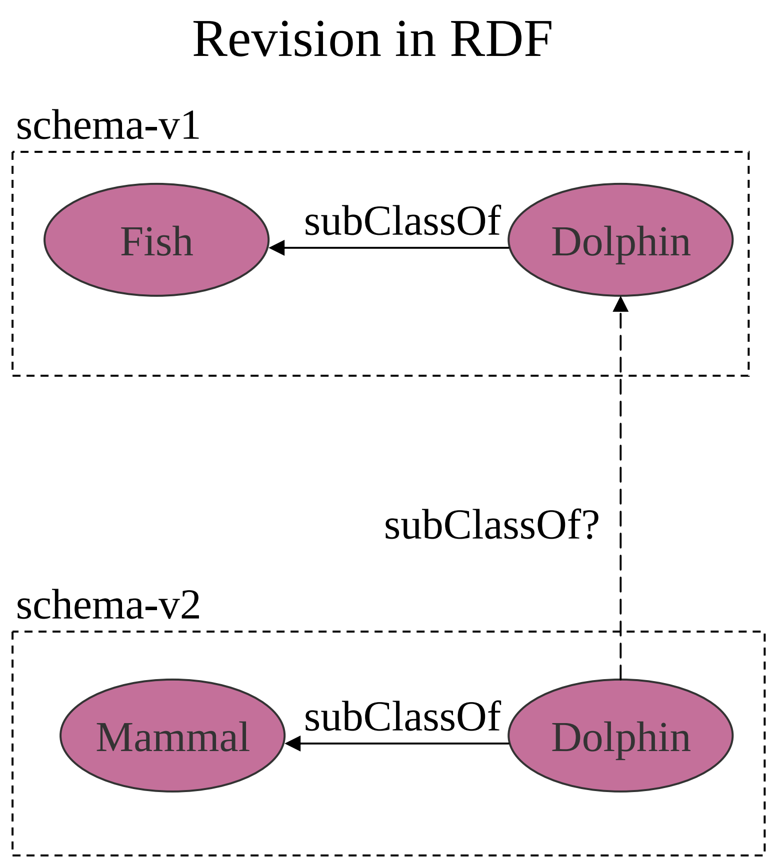
- Ontologies can be extended by other ontologies in order to provide additional definitions
- Possible Approach:
  - Explicit representation of extension

Multiple Schemas in RDF



Goal 3. Ontology Evolution

- Ontologies can be changed over time and data sources can specify which version of the ontology they commit to
- Possible Approach:
  - Revisions are separate documents
  - Explicit links to prior versions
  - Explicit backwards-compatibility



Goal 4. Ontology Interoperability

- Different ontologies may model the same concepts in different ways

- Possible Approach:
  - primitives for mapping

Goal 5. Detect Inconsistency

- Different ontologies or data sources may be contradictory

- Possible Approach:
  - allow language to express inconsistency
  - theory supports efficient detection of inconsistency
  - provide mechanism for reporting inconsistencies

Goal 6. Scalability

- Language can be used with large ontologies and large data sets

- Possible Approach:
  - restrict language for efficient reasoning

Goal 7. Ease of Use

- Language should provide a low learning barrier and have clear concepts and meaning

- Possible Approach:
  - When possible, use concepts and idioms familiar to average software engineers

Goal 8. Compatibility with other standards

- The language should have an XML serialization
- Should correspond to next Semantic Web standards (such as XML Schema and RDF, RDFS)

Goal 9. Ontology-based Search

- Search that exploits the meaning of terms instead of just the syntax

- Possible Approach:
  - use background ontologies for:
    - query expansion
    - understanding of term relationships
    - identify parameters and value restrictions

Goal 10. Expressiveness

- The language should be as expressive as possible, given a balance with

  Goal 6. Scalability

Goal 11. Internationalization

- Develop multi-language presentation of ontology in the language of each user
- Requirements
  - Character Model
    - Character set support (from XML unicode)
    - Uniqueness of unicode strings (unicode normal form c solution from w3c internationalization group c cedilla – jeremy provides details)
  - Localized display of an ontology (display ontology in foreign language to viewer)

OWL Use Cases

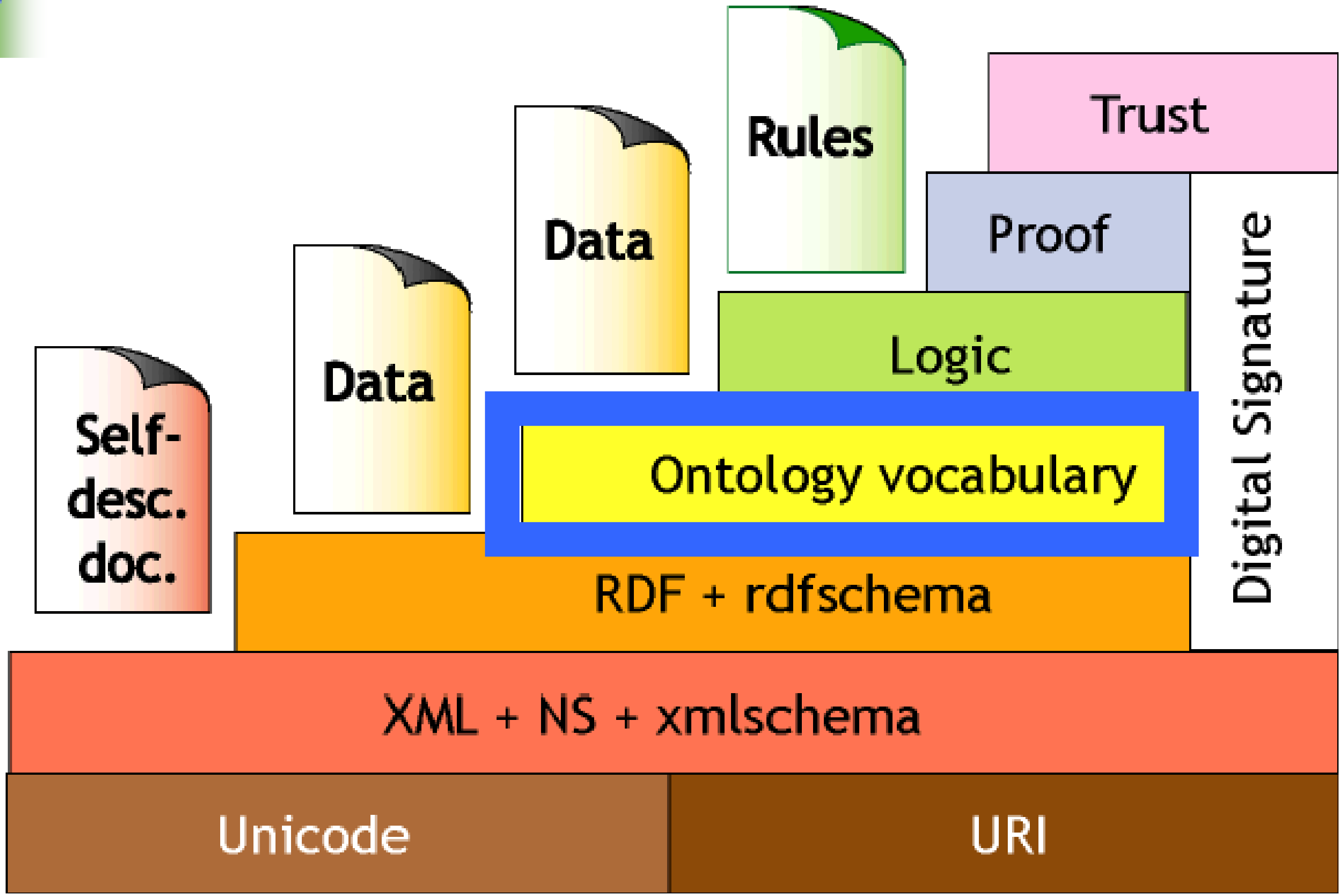
Semantic Web apps:

- portal Websites & intranets (information architecture)
- multimedia digital libraries (rich metadata)
- agents & Web services (interoperability, automation)
- etc.

OWL Capabilities

- ontology sharing, evolution, interoperability
- inconsistency detection
- expressivity
- scalability
- standards compliance

Architecture of Semantic Web



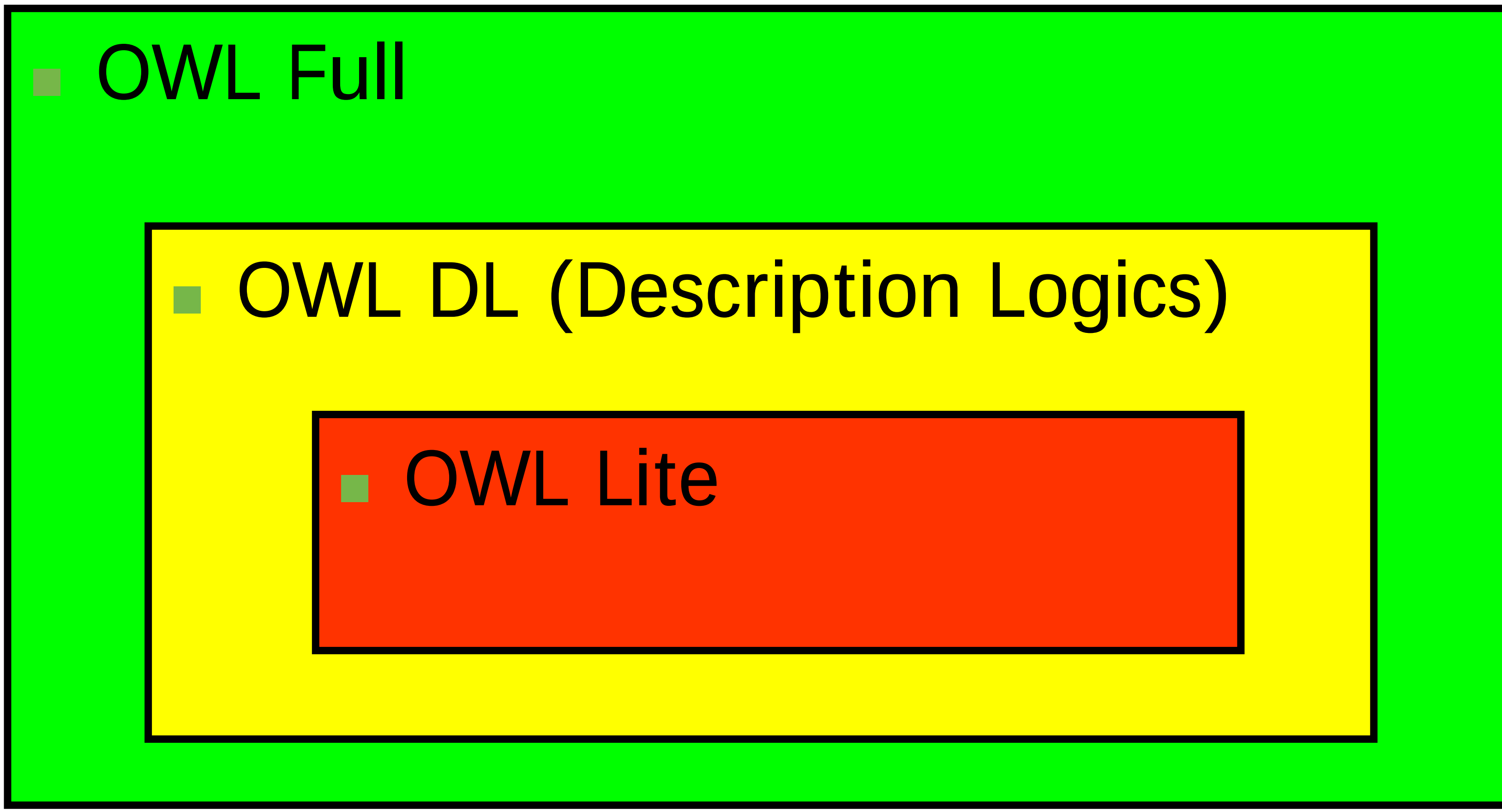
XML, RDF and OWL

- XML: universal syntax
- XML Schema: defines structure of XML docs
- RDF: datamodel for resource objects
- RDF Schema: basic vocabulary for defining RDF classes & properties, and hierarchies of each
- OWL: extended vocabulary for defining classes and properties, including
  - cardinality (e.g. minCardinality 1)
  - equality (e.g. equivalentClass)
  - relationships between classes (e.g. disjointWith)
  - characteristics of properties (e.g. FunctionalProperty)

OWL Sublanguages

- OWL Lite
  - “RDF-and-a-half”
  - Mainly intended for class hierarchies & simple constraints (cardinality 0 or 1, equality, ...)
- OWL DL
  - Description Logic properties
  - Intended where logical inference, completeness & decidability are an issue
- OWL Full
  - Max expressivity
  - No computational guarantees

OWL Sublanguages as Sets



OWL Sublanguages as Sets

- Every legal OWL Lite ontology is a legal OWL DL ontology
- Every legal OWL DL ontology is a legal OWL Full ontology
- Every valid OWL Lite conclusion is a valid OWL DL conclusion
- Every valid OWL DL conclusion is a valid OWL Full conclusion

The converse in each case does not hold!

OWL Lite: RDF Schema Constructs

By default owl:

- Class
- rdfs:subClassOf
- rdf:Property
- rdfs:subPropertyOf
- rdfs:domain
- rdfs:range
- Individual

OWL Lite: Property Characteristics

- ObjectProperty
- DatatypeProperty
- inverseOf
- TransitiveProperty
- SymmetricProperty
- FunctionalProperty
- InverseFunctionalProperty

OWL Lite: (In)Equality

- equivalentClass
- equivalentProperty
- sameAs
- differentFrom
- AllDifferent
- distinctMembers

OWL Lite: Property Restrictions

- Restriction
- onProperty
- allValuesFrom
- someValuesFrom

OWL Lite: Restricted Cardinality

- minCardinality (only 0 or 1)
- maxCardinality (only 0 or 1)
- cardinality (only 0 or 1)

OWL Lite: Class Intersection

- intersectionOf

OWL Lite: Datatypes

- XML Schema Datatypes

OWL Lite: Header Information

- Ontology
- imports

OWL Lite: Versioning

- versionInfo
- priorVersion
- backwardCompatibleWith
- incompatibleWith
- DeprecatedClass
- DeprecatedProperty

OWL Lite: Annotation Properties

- rdfs:label
- rdfs:comment
- rdfs:seeAlso
- rdfs:isDefinedBy
- AnnotationProperty
- OntologyProperty

OWL DL and OWL FULL: Class Axioms

- oneOf, dataRange
- disjointWith
- equivalentClass
  (applied to class expressions)
- rdfs:subClassOf
  (applied to class expressions)

OWL DL and OWL FULL: Boolean Combinations of Class Expressions

- unionOf
- complementOf
- intersectionOf

OWL DL and OWL FULL: Arbitrary Cardinality

- minCardinality
- maxCardinality
- cardinality

OWL DL and OWL FULL: Filler Information

- hasValue