Student Online Teaching Advice Notice

The materials and content presented within this session are intended solely for use in a context of teaching and learning at Trinity.

Any session recorded for subsequent review is made available solely for the purpose of enhancing student learning.

Students should not edit or modify the recording in any way, nor disseminate it for use outside of a context of teaching and learning at Trinity.

Please be mindful of your physical environment and conscious of what may be captured by the device camera and microphone during videoconferencing calls.

Recorded materials will be handled in compliance with Trinity's statutory duties under the Universities Act, 1997 and in accordance with the University's policies and procedures.

Further information on data protection and best practice when using videoconferencing software is available at https://www.tcd.ie/info_compliance/data-protection/.

University of Dublin
Trinity College

CSU22041: Information Management I

Resource Definition Framework Schema(RDFS)
Ontology Web Language(OWL)

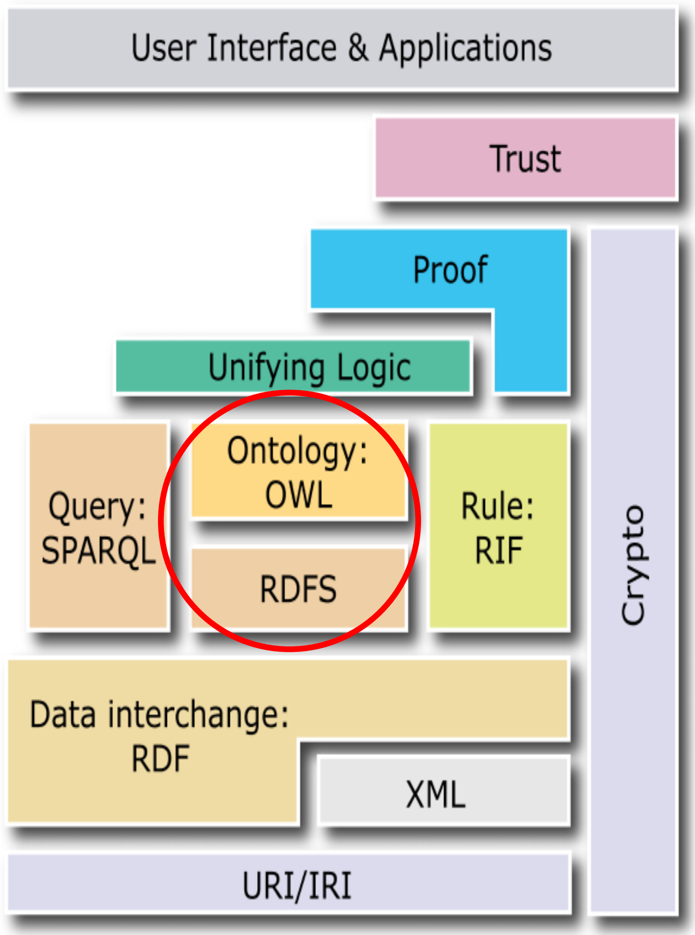
2020-2021

Gaye Stephens gaye.stephens@tcd.ie

The Semantic Web Stack

Traditional Web
URI/IRI
XML
Cryptography

Semantic Web
Resource Description Framework (RDF)
RDF Schema
Web Ontology Language
SPARQL
Rules: RIF (and SWRL)



(from https://www.w3.org/)

https://www.w3.org/TR/cooluris/

Structuring the knowledge

RDF provides a way of building graphs from triples, but doesn’t constrain the graph too much

- Nothing stops an application from giving a **place** a **surname**, for example, although this is probably nonsense

The problem is that RDF is an *untyped mechanism* for building graphs

- No knowledge of which triples are 'allowed', or what 'thing' must be the subject/object of an arc

This is a problem in two distinct ways

- In interpretation – different people may interpret the predicates subtly differently and use them between values you can’t handle
- In scaling – hard for an application to get it right

RDF and RDFS

RDF stands for Resource Description Framework

It is a W3C candidate recommendation (http://www.w3.org/RDF)

RDF is graphical formalism (+ XML syntax + semantics)

- for representing metadata
- for describing the semantics of information in a machine-accessible way

RDFS extends RDF with 'schema vocabulary', e.g.:

- Class, Property
- type, subClassOf, subPropertyOf
- range, domain

RDFS Examples

RDF Schema terms (just a few examples):

- Class
- Property
- type
- subClassOf
- range
- domain

These terms are the RDF Schema building blocks (constructors) used to create vocabularies:

```
<Person,type,Class>
<hasColleague,type,Property>
<Professor,subClassOf,Person>
<Carole,type,Professor>
<hasColleague,range,Person>
<hasColleague,domain,Person>
```

Problems with RDFS

RDFS too weak to describe resources in sufficient detail

- No localised range and domain constraints
  - *Can't say that the range of hasChild is person when applied to persons and elephant when applied to elephants*
- No existence/cardinality constraints
  - *Can't say that all* instances *of person have a mother that is also a person, or that persons have exactly 2 parents*
- No transitive, inverse or symmetrical properties
  - *Can't say that isPartOf is a transitive property, that hasPart is the inverse of isPartOf or that touches is symmetrical*
- …

Difficult to provide reasoning support

- No 'native' reasoners for non-standard semantics

Ontology Web Language (OWL)

Ontology = formalized vocabularies of terms

- often covering a specific domain and shared by a community of users.
- specify the definitions of terms by describing their relationships with other terms in the ontology

OWL standardised by W3C

- https://www.w3.org/TR/owl2-overview/

OWL has richer vocabulary and is more expressive than RDFS

- specify cardinalities of object relations and datatype properties (attributes)
- use logical operators in definitions (e.g. use union of classes as a range of relation)

Example: Defining terms and a subclass relationship

Define the term 'Room'

```
<owl:Class rdf:ID='Room'/>
```

Define term 'Restroom' and state that a Restroom is a type of Room

```
 <owl:Class rdf:ID='Restroom'>
    <rdfs:subClassOf rdf:resource='#Room'/>
</owl:Class>
```

Note: owl:Thing is a predefined OWL Class and is the root of all classes

Defining Classes

OWL provides several other mechanisms for defining classes

- equivalentClass allows you to state that two classes are synonymous
- disjointWith allows you to state that an instance of this class cannot be an instance of another
  - *E.g. Man and Woman could be stated as disjoint classes*
- unionOf allows you specify that a class contains things that are from more than one class
  - *E.g. Restroom could be defined as a union of MensRoom and LadiesRoom*
- intersectionOf allows you to specify that a class contains things that are both in one and the other
- complementOf allows you specify that a class contains things that are not other things
  - *E.g. Children are not SeniorCitizens*

Defining Properties

In RDF Schema the rdf:Property is used **both to**

- Relate one Resource to another Resource
  - *For example, a 'accessRestrictedToGender' property can relate a Restroom to a Gender*
- Relate a resource to a rdfs:Literal or datatype
  - *For example, a 'latitude' property relates a Room to a xsd:string type*

OWL provides **different statements for two cases**

- **owl:ObjectProperty** is used to relate a resource to another

```
<owl:ObjectProperty rdf:ID='accessRestrictedToGender'>
   <rdfs:range rdf:resource='#Gender'/>
   <rdfs:domain rdf:resource='#Restroom'/>
```

- **owl:DatatypeProperty** is used to relate a resource to a rdfs:Literal or XML schema data type

```
<owl:DatatypeProperty rdf:ID='latitude'>
   <rdfs:range rdf:resource='http://www.w3.org/2001/XMLSchema#string'/>
   <rdfs:domain rdf:resource='#Place'/>
```

In addition OWL allows you state that a property is…

Symmetric

- **<rdf:type rdf:resource='http://www.w3.org/2002/07/owl#SymmetricProperty'/>**
- If a property is symmetric, then if the pair (x,y) is an instance then it can be deduced that (y,x) is also an instance.
- Example**, property 'adjacentRoom' property of Room** or 'friend' property of person

Transitive

- **<rdf:type rdf:resource='http://www.w3.org/2002/07/owl#TransitiveProperty'/>**
- If a property is transitive, then if the pair (x,y) is an instance of P and (y,z) is an instance of P then it can be deduced that (x,z) is also an instance.
- Example, property 'isSpatiallySubsumedBy' property

```
<owl:ObjectProperty rdf:ID='isSpatiallySubsumedBy'
    rdf:type='http://www.w3.org/2002/07/owl#TransitiveProperty'>
   <rdfs:domain rdf:resource='http://www.w3.org/2002/07/owl#Thing'/>
   <rdfs:range rdf:resource='http://www.w3.org/2002/07/owl#Thing'/>
```

In addition OWL allows you state that a property is…

Functional

- If a property is functional, then it has no more than one value for each individual, that is equivalent to owl:maxCardinality restriction of 1
- Example, property 'Address' of a Room

```
<owl:DatatypeProperty rdf:ID='Address'
    rdf:type='http://www.w3.org/2002/07/owl#FunctionalProperty'>
   <rdfs:domain rdf:resource='#Room'/>
</owl:DatatypeProperty>
```

In addition OWL allows you state that a property is…

InverseOf

- One property may be stated to be the inverse of another property. It can be deduced that if the property P1 is stated to be the inverse of P2, then if X is related to Y via P2 then Y is related to X via P1
- Example, property 'spatiallySubsumes'

```
<owl:ObjectProperty rdf:ID='spatiallySubsumes'>
   <owl:inverseOf>
     <owl:ObjectProperty rdf:about='#isSpatiallySubsumedBy'/>
   </owl:inverseOf>
```

InverseFunctionalProperty

- The inverse of the property has most one value for each individual
- Example, property 'coordinates' property of a Room could be declared as inverse functional, as the inverse 'theRoomAt' property can only have one value
- **<rdf:type rdf:resource='http://www.w3.org/2002/07/owl#InverseFunctionalProperty'/>**

Summary Example

```
<?xml version '1.0'?>
<Room rdf:ID='LargeConferenceRoom'>
        <address rdf:resource='G.02'/>
        <spatiallySubsumedBy rdf:resource='O'Reilly Institute'/>
        <adjacentRoom rdf:resource='SmallConferenceRoom' />
        <coordinates rdf:resource=44,55 />
</Room>
```

Given the preceding definitions it can be inferred automatically/reasoned:

1. The O'Reilly Institute *spatially subsumes* the Large Conf Room (since spatiallySubsumedBy is an inverse property)
2. The Small Conference Room is *adjacent to* Large Conference Room (since adjacentRoom is symmetric)
3. Only the Large Conference Room can be found at coordinates 44,55 (since coordinates is inverse functional)
4. The Large Conference Room has only one address (since address is functional)

Semantic Model in Action (Example)

"Find me a camera with size X and aperture Y"

Web Agent

```xml
<SLR rdf:ID="Olympus-OM-10"
        xmlns="http://www.camera.org#">
    <lens>
        <Lens>
            <focal-length>75-300mm zoom</focal-length>
            <f-stop>4.5-5.6</f-stop>
        </Lens>
    </lens>
    <body>
        <Body>
            <shutter-speed rdf:parseType="Resource">
                <min>0.002</min>
                <max>1.0</max>
                <units>seconds</units>
            </shutter-speed>
        </Body>
    </body>
    <cost rdf:parseType="Resource">
        <rdf:value>325</rdf:value>
        <currency>USD</currency>
    </cost>
</SLR>
```

Hunts.xml

"Relationship between Camera and SLR?"

"SLR is a type of Camera."

"Relationship between size and focal-length?"

"focal-length is synonymous with size."

"Relationship between aperture and f-stop?"

"f-stop is synonymous with aperture."

Semantic Definitions

```xml
<owl:Class rdf:ID="SLR">
    <rdfs:subClassOf rdf:resource="#Camera"/>
</owl:Class>

<owl:DatatypeProperty rdf:ID="focal-length">
    <owl:equivalentProperty rdf:resource="#size"/>
    <rdfs:domain rdf:resource="#Lens"/>
    <rdfs:range rdf:resource="&xsd;#string"/>
</owl:DatatypeProperty>

<owl:DatatypeProperty rdf:ID="f-stop">
    <owl:equivalentProperty rdf:resource="#aperture"/>
    <rdfs:domain rdf:resource="#Lens"/>
    <rdfs:range rdf:resource="&xsd;#string"/>
</owl:DatatypeProperty>
```

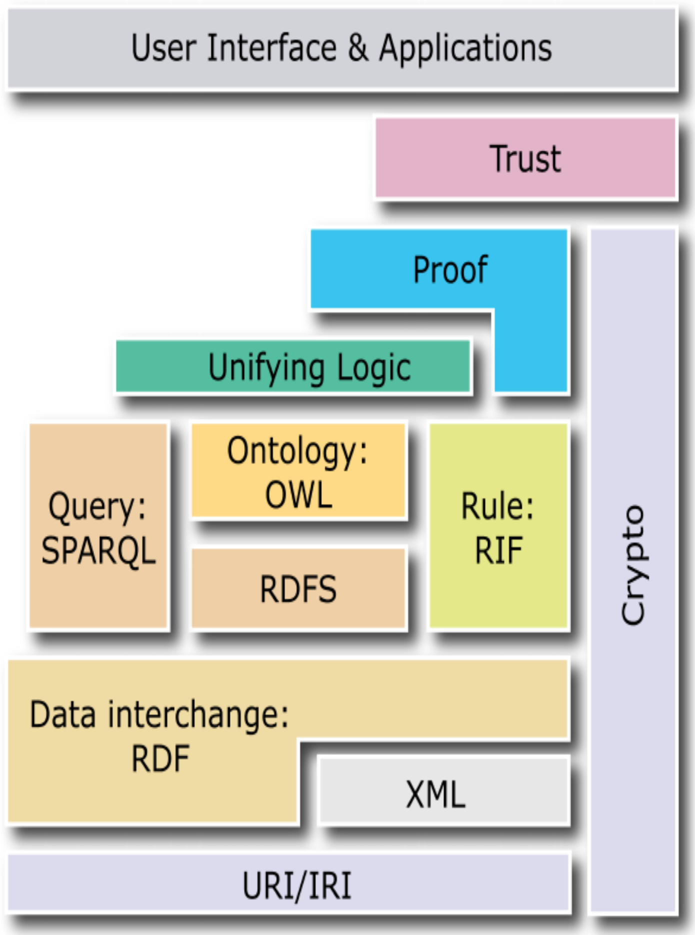
Camera.owl

[DARPA 2003]

The Semantic Web Stack

Traditional Web
URI/IRI
XML
Cryptography

Semantic Web
Resource Description Framework
RDF Schema
Web Ontology Language
SPARQL
Rules: RIF (and SWRL)



(from https://www.w3.org/)

https://www.w3.org/TR/cooluris/

The Semantic Web Stack

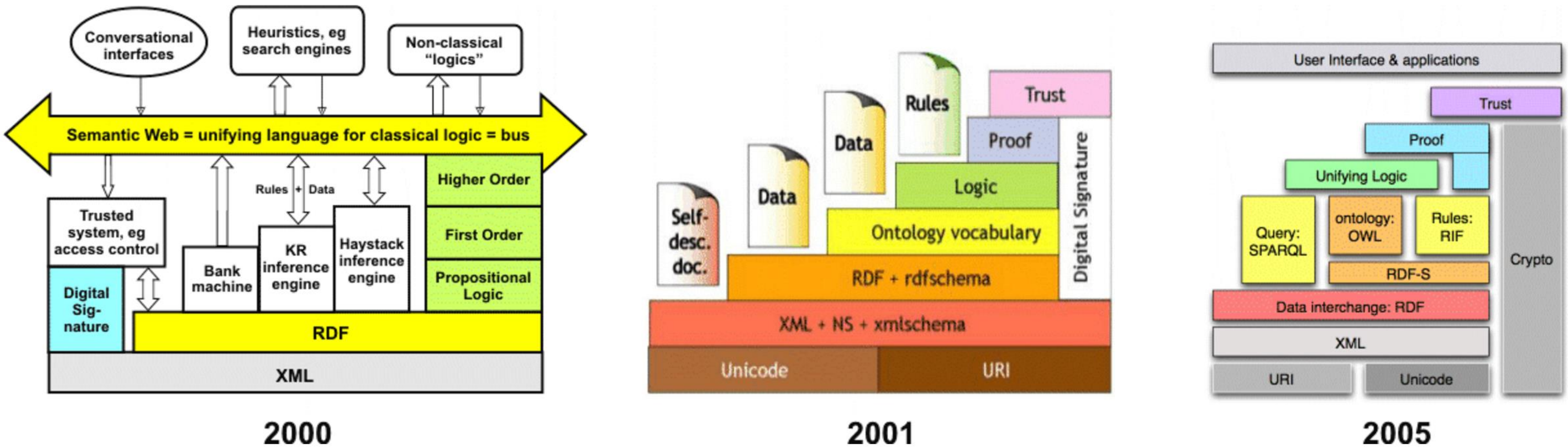


Figure 4. Evolution of the Semantic Web from 2000 to 2005

https://medium.com/openlink-software-blog/semantic-web-layer-cake-tweak-explained-6ba5c6ac3fab

Sample Linked Data Exam Question

1. The move to Linked Data (and eventually the Semantic Web) will bring benefits for application developers, compared to how data is currently available on the web.

   Discuss the statement above. Diagrams can be included to support or illustrate points made in your discussion.
   Include at least the following points in your answer.
   - Describe the benefits that Linked Data could bring;
   - Explain the concept of Linked Data;
   - Explain the concept of the Semantic Web;
   - Describe the Semantic Web Stack;
   - Explain in what way OWL builds on RDF and what benefits this brings.

[Total 50 Marks]

Essay based answer

That’s All Folks
Thank You for Listening and for your participation in the module

“Dear Andy: How have you been? Your mother and I are fine. We miss you. Please sign off your computer and come downstairs for something to eat. Love, Dad.”