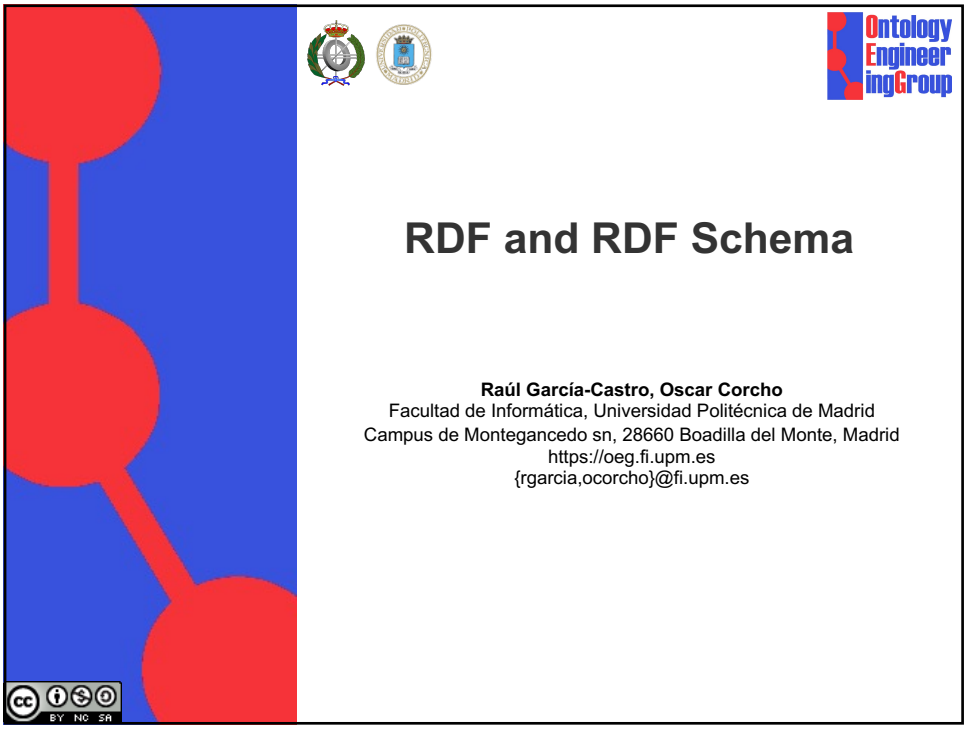


1

License

- This work is licensed under the license CC BY-NC-SA 4.0 International
  - http://purl.org/NET/rdflicense/cc-by-nc-sa4.0
- You are free:
  - to Share — to copy, distribute and transmit the work
  - to Remix — to adapt the work
- Under the following conditions
  - Non-commercial – You cannot use it for commercial purposes, nor for training inside a commercial company
  - Attribution — You must attribute the work by inserting
    - "[source http://www.oeg-upm.net/]" at the footer of each reused slide
    - a credits slide stating: "These slides are partially based on "RDF, RDF Schema and SPARQL" by R. García-Castro, O. Corcho"
  - Share-Alike

2

Índice

- **General introduction**
- RDF
  - RDF Components
  - RDF Serialisation
  - Inferences in RDF
- RDF Schema
  - RDFS Components
  - RDFS Serialisation
  - Inferences in RDFS

3

3

Let's start with an example…

- Let's use different strategies to formalise the following piece of text

  "Oscar Corcho and Raúl García are lecturers of the "Semantic Web" course for the academic year 2021-2022. This course is an optional course in the 7th semester of the Degree of Computer Science. Both of them belong to the Department of Artificial Intelligence. Oscar belongs to this department since 2007, and became a full professor on 2016"

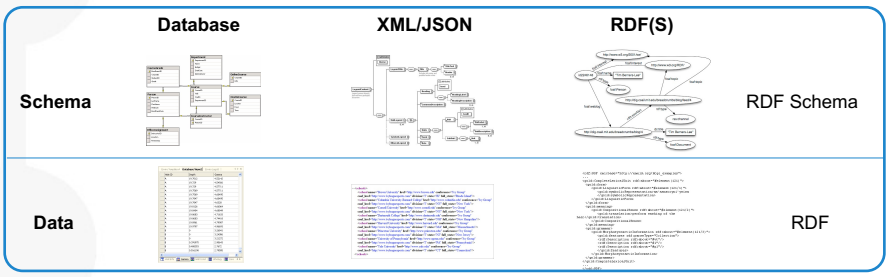
- Now let's use:
  - The relational model
  - JSON (or XML)
  - A graph-based representation

4

4

RDF and RDF Schema

- RDF: Resource Description Framework

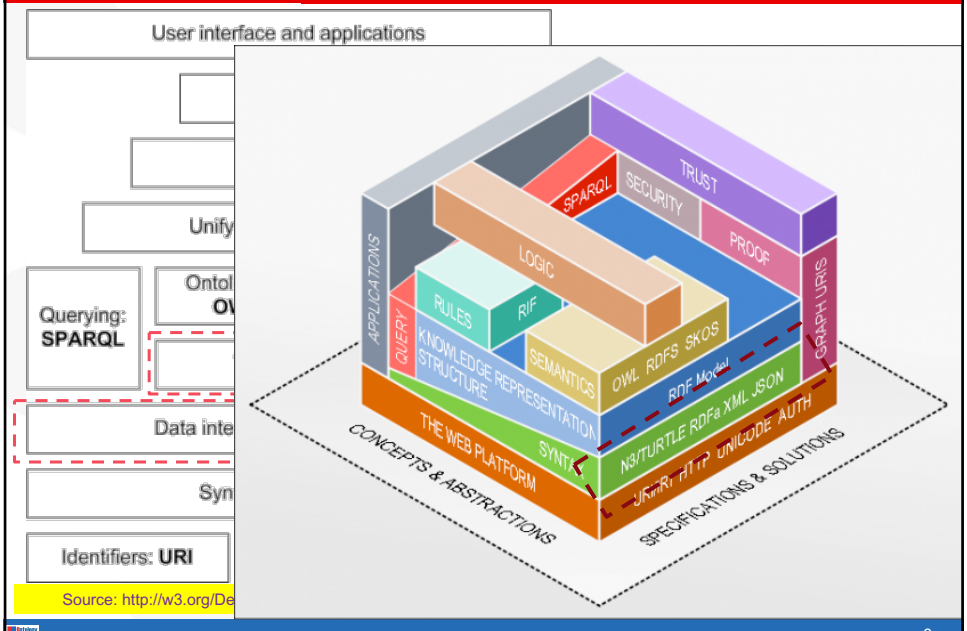


- W3C Recommendation
  - Model
  - Syntax
  - Semantics

5

5

RDF(S) and the Semantic Web



6

6

Índice

- **General introduction**
- **RDF**
  - **RDF Components**
  - **RDF Serialisation**
  - **Inferences in RDF**
- RDF Schema
  - RDFS Components
  - RDFS Serialisation
  - Inferences in RDFS

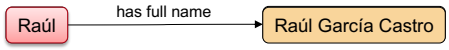
7

7

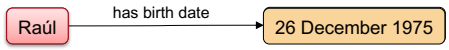
RDF Components

- Also known as triples
  - [Subject, Predicate, Object]
- "Raúl is a member of the Ontology Engineering Group"
  - [Raúl, is member of, Ontology Engineering Group]

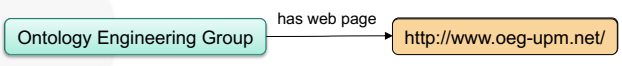
- "Raúl's full name is Raúl García Castro"
  - [Raúl, has full name, Raúl García Castro]



- "Raúl was born on December 26th 1975"
  - [Raúl, was born, 26 December 1975]



- "The homepage of the Ontology Engineering Group is http://www.oeg-upm.net/"
  - [Ontology Engineering Group, has web page, http://www.oeg-upm.net/]

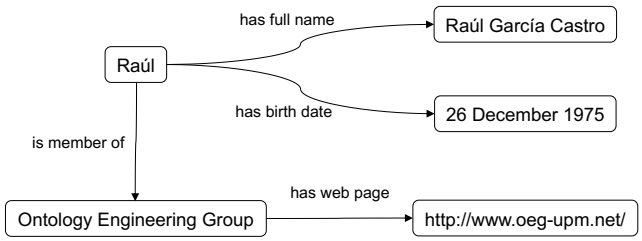


8

8

Grafos y Datasets RDF

- RDF graphs are sets of triples

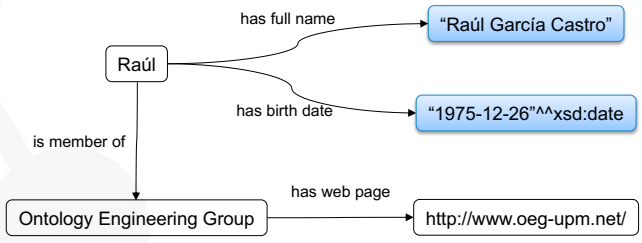


- A set of RDF graphs is an RDF Dataset (in RDF1.1)
  - There is a default graph
  - And zero or more named graphs

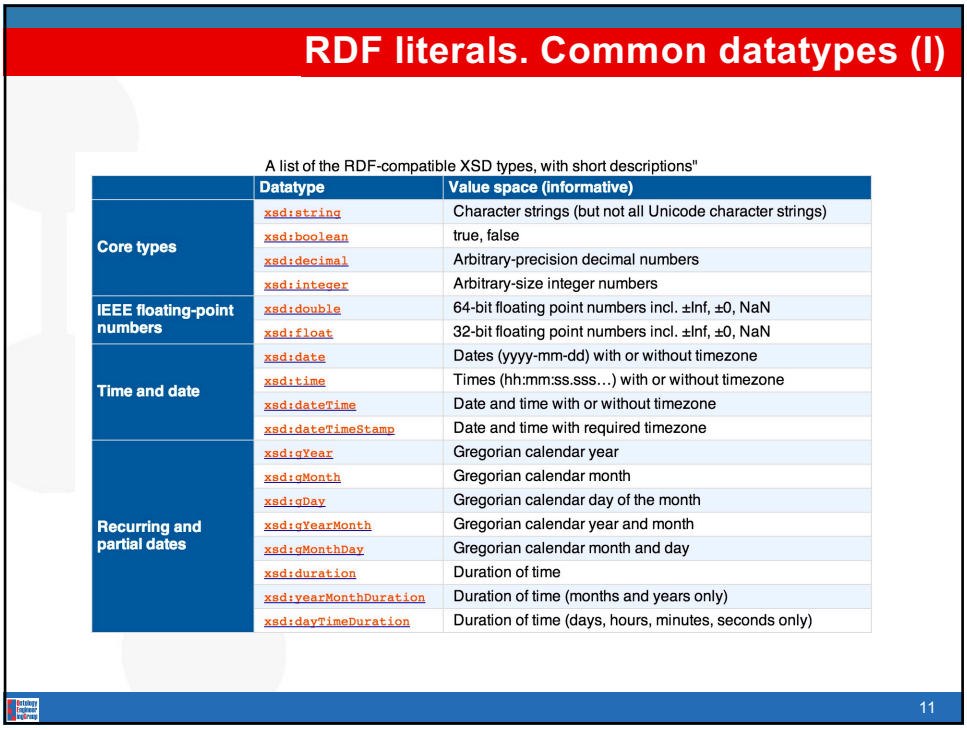
9

RDF literals

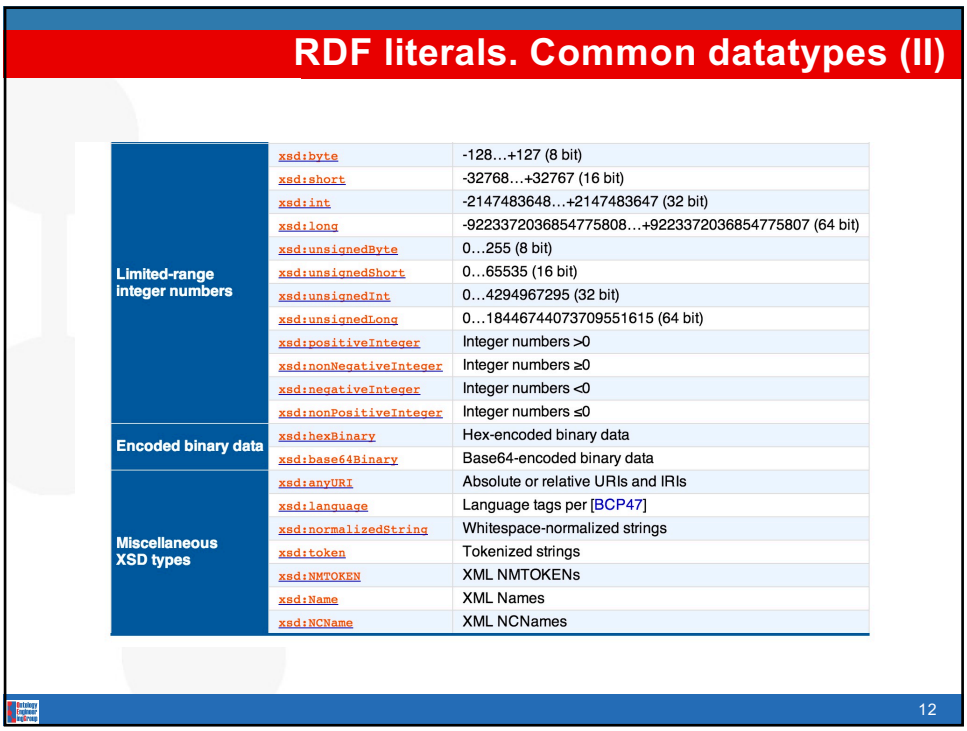
- Triple objects may be resources or literals
  - Subjects and predicates are always resources
- Literals may have a datatype
  - Commonly using XML Schema datatypes
  - And a few extra datatypes (RDF1.1)
    - rdf:langString (e.g., “Spain”@en)
    - rdf:HTML and rdf:XMLLiteral



10


RDF literals. Common datatypes (I)
A list of the RDF-compatible XSD types, with short descriptions"
Datatype
Value space (informative)
Core types
xsd:string
Character strings (but not all Unicode character strings)
xsd:boolean
true, false
xsd:decimal
Arbitrary-precision decimal numbers
xsd:integer
Arbitrary-size integer numbers
IEEE floating-point numbers
xsd:double
64-bit floating point numbers incl. ±Inf, ±0, NaN
xsd:float
32-bit floating point numbers incl. ±Inf, ±0, NaN
Time and date
xsd:date
Dates (yyyy-mm-dd) with or without timezone
xsd:time
Times (hh:mm:ss.sss…) with or without timezone
xsd:dateTime
Date and time with or without timezone
xsd:dateTimeStamp
Date and time with required timezone
Recurring and partial dates
xsd:gYear
Gregorian calendar year
xsd:gMonth
Gregorian calendar month
xsd:gDay
Gregorian calendar day of the month
xsd:gYearMonth
Gregorian calendar year and month
xsd:gMonthDay
Gregorian calendar month and day
xsd:duration
Duration of time
xsd:yearMonthDuration
Duration of time (months and years only)
xsd:dayTimeDuration
Duration of time (days, hours, minutes, seconds only)
11

11


RDF literals. Common datatypes (II)
Limited-range integer numbers
xsd:byte
-128…+127 (8 bit)
xsd:short
-32768…+32767 (16 bit)
xsd:int
-2147483648…+2147483647 (32 bit)
xsd:long
-9223372036854775808…+9223372036854775807 (64 bit)
xsd:unsignedByte
0…255 (8 bit)
xsd:unsignedShort
0…65535 (16 bit)
xsd:unsignedInt
0…4294967295 (32 bit)
xsd:unsignedLong
0…18446744073709551615 (64 bit)
xsd:positiveInteger
Integer numbers >0
xsd:nonNegativeInteger
Integer numbers ≥0
xsd:negativeInteger
Integer numbers <0
xsd:nonPositiveInteger
Integer numbers ≤0
Encoded binary data
xsd:hexBinary
Hex-encoded binary data
xsd:base64Binary
Base64-encoded binary data
Miscellaneous XSD types
xsd:anyURI
Absolute or relative URIs and IRIs
xsd:language
Language tags per [BCP47]
xsd:normalizedString
Whitespace-normalized strings
xsd:token
Tokenized strings
xsd:NMTOKEN
XML NMTOKENs
xsd:Name
XML Names
xsd:NCName
XML NCNames
12

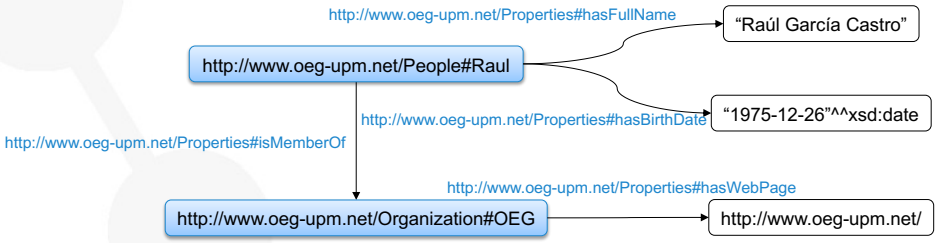
12

IRIs and URIs in RDF

- URI (RFC3986) or IRI (RFC3987) Components
  - http://www.oeg-upm.net:8080/Info/People?position=current#Raul

Scheme | Authority | Path | Query | Fragment

- -URIs vs IRIs in RDF:
  - Uniform vs Internationalized Resource Identifier
  - IRIs may contain Unicode characters



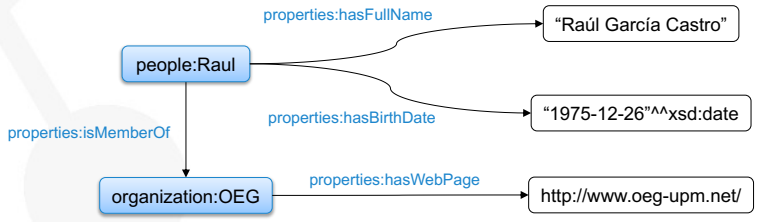
13

13

Namespaces in RDF

- URIs under a *namespace* are called vocabularies

Prefijo	URI
people	http://www.oeg-upm.net/People#
organization	http://www.oeg-upm.net/Organization#
properties	http://www.oeg-upm.net/Properties#
rdf	http://www.w3.org/1999/02/22-rdf-syntax-ns#
rdfs	http://www.w3.org/2000/01/rdf-schema#
xsd	http://www.w3.org/2001/XMLSchema#

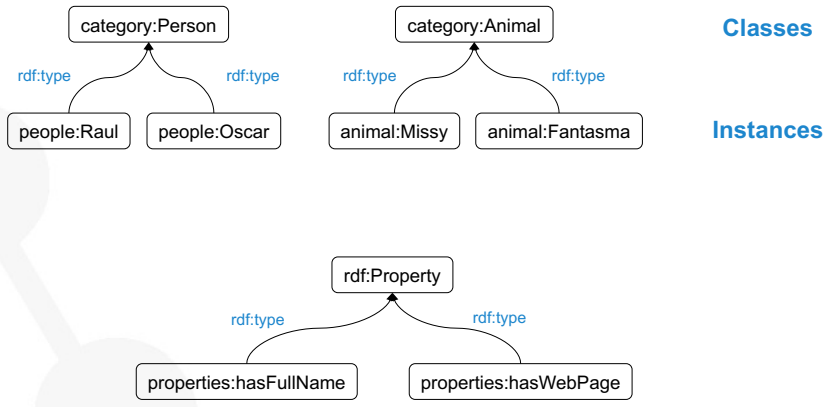


14

14

Class membership (rdf:type)

- ***rdf:type*** is used to associate resources to their corresponding categories/classes

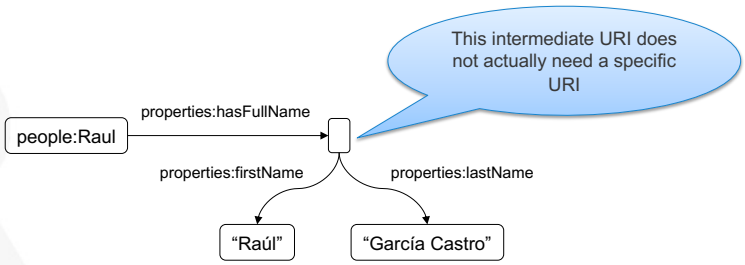


15

15

Blank nodes

- Sometimes we need complex data structures like the following

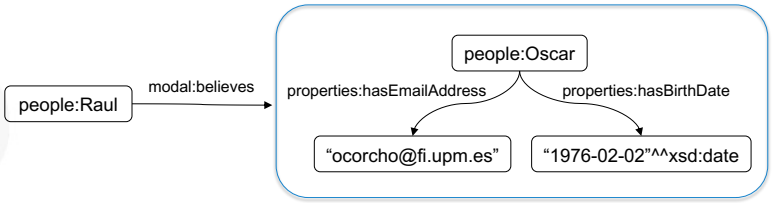


16

16

RDF Reification

- RDF triples about other RDF triples
  - "Raúl belives that Oscar was born on February 2nd, 1976, and that his –mail address is ocorcho@fi.upm.es"
- Expressed using ***rdf:Statement***, ***rdf:subject***, ***rdf:predicate***, and ***rdf:object***
  - *We will see other options when checking RDF-star and Wikidata*



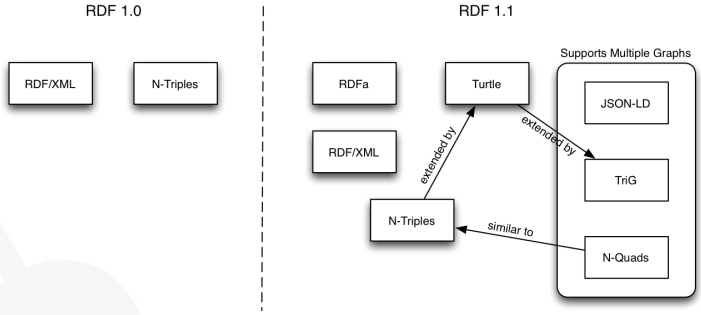
- RDF reification
  - It allows expressing beliefs (and other modalities)
  - It allows expressing confidence models.
  - It allows expressing metadata about dataå

20

20

RDF Serialisations

- Different syntaxes (some of them recently approved – February 2014)
  - **RDF/XML** (www.w3.org/TR/rdf-syntax-grammar/)
  - **Turtle** (http://www.w3.org/TR/turtle/)
  - **N-Triples** (http://www.w3.org/TR/n-triples/)
  - **TriG** (http://www.w3.org/TR/trig/)
  - **RDFa** (http://www.w3.org/TR/xhtml-rdfa-primer/)
  - **JSON-LD** (http://www.w3.org/TR/json-ld/)
  - **N-Quads** (http://www.w3.org/TR/n-quads/)

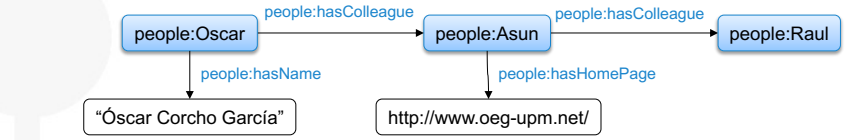


- **Important: the order of triples inside a document is not relevant**

23

23

RDF Serialisations. RDF/XML

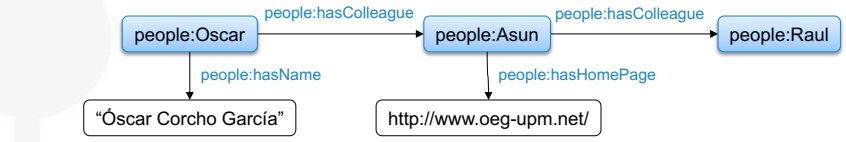



```
<?xml version="1.0"?>
    <rdf:RDF
        xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
        xmlns:person="http://www.ontologies.org/ontologies/people#"
        xmlns="http://www.oeg-upm.net/ontologies/people#"
        xml:base="http://www.oeg-upm.net/ontologies/people">

      <rdf:Property rdf:about="http://www.ontologies.org/ontologies/people#hasHomePage"/>
      <rdf:Property rdf:about="http://www.ontologies.org/ontologies/people#hasColleague"/>
      <rdf:Property rdf:about="http://www.ontologies.org/ontologies/people#hasName"/>

      <rdf:Description rdf:about="#Raul"/>
      <rdf:Description rdf:about="#Asun">
        <person:hasColleague rdf:resource="#Raul"/>
        <person:hasHomePage>http://www.oeg-upm.net/</person:hasHomePage>
      </rdf:Description>
      <rdf:Description rdf:about="#Oscar">
        <person:hasColleague rdf:resource="#Asun"/>
        <person:hasName>Oscar Corcho García</person:hasName>
      </rdf:Description>

    </rdf:RDF>
```

24

24

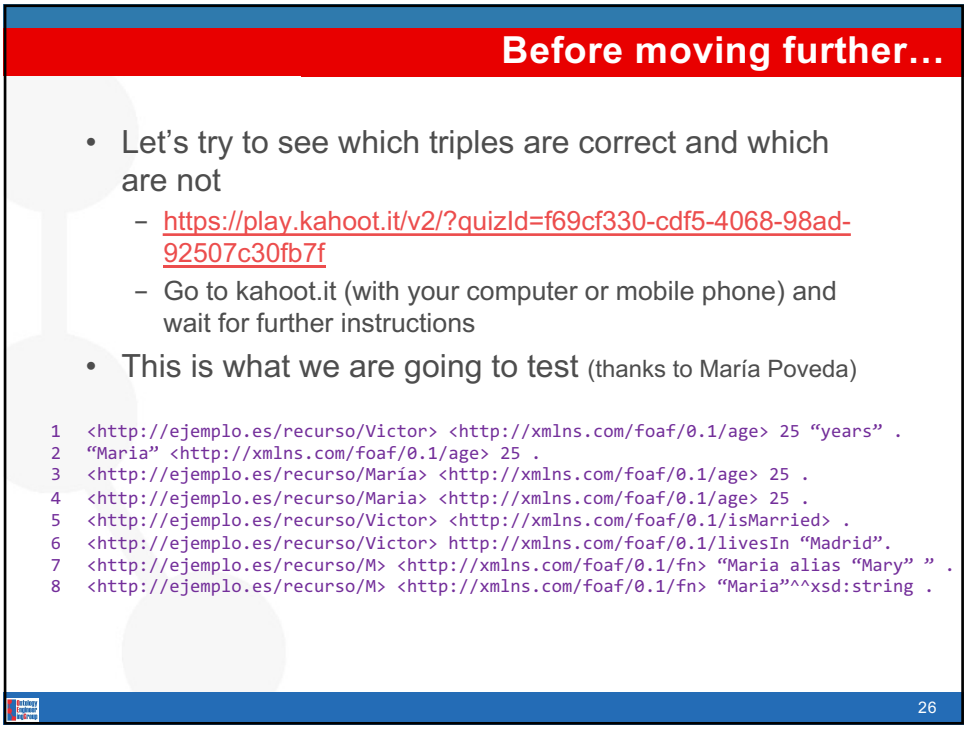
RDF Serialisations. Turtle



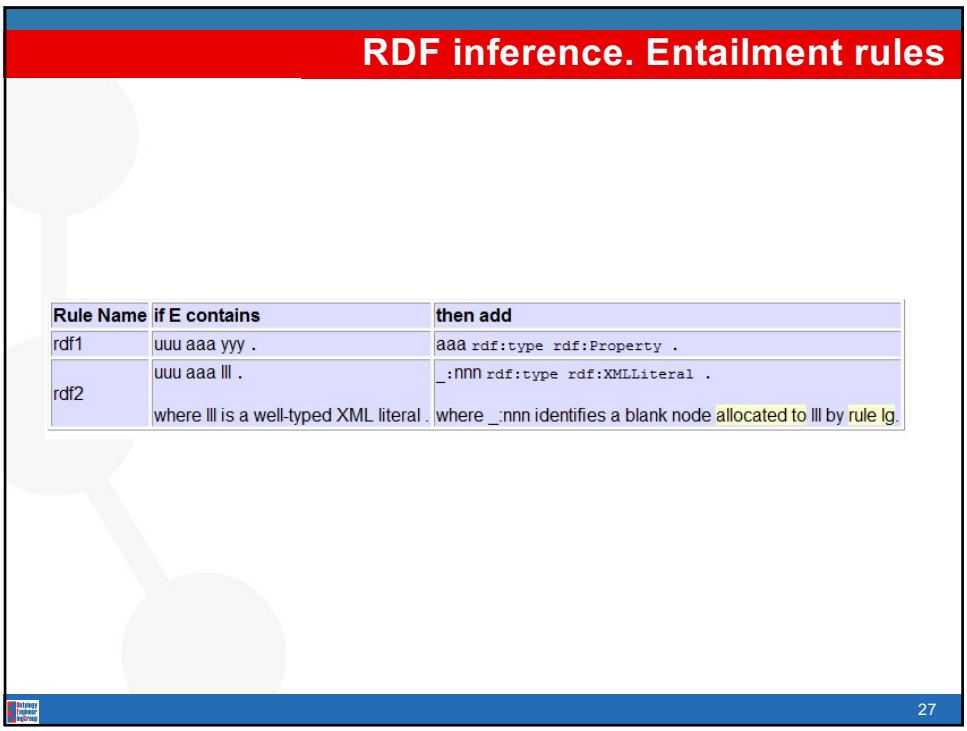

```
@base  <http://www.oeg-upm.net/ontologies/people/ >
@prefix vocab: <http://www.ontologies.org/ontologies/people#>
:Asun   vocab:hasColleague :Raul ;
        vocab:hasHomePage “http://www.oeg-upm.net/”.
:Oscar  vocab:hasColleague :Asun ;
        vocab:hasName “Óscar Corcho García”.
```

25

25


Before moving further…
• Let's try to see which triples are correct and which are not
– https://play.kahoot.it/v2/?quizId=f69cf330-cdf5-4068-98ad-92507c30fb7f
– Go to kahoot.it (with your computer or mobile phone) and wait for further instructions
• This is what we are going to test (thanks to María Poveda)
1 <http://ejemplo.es/recurso/Victor> <http://xmlns.com/foaf/0.1/age> 25 "years" .
2 "Maria" <http://xmlns.com/foaf/0.1/age> 25 .
3 <http://ejemplo.es/recurso/María> <http://xmlns.com/foaf/0.1/age> 25 .
4 <http://ejemplo.es/recurso/Maria> <http://xmlns.com/foaf/0.1/age> 25 .
5 <http://ejemplo.es/recurso/Victor> <http://xmlns.com/foaf/0.1/isMarried> .
6 <http://ejemplo.es/recurso/Victor> http://xmlns.com/foaf/0.1/livesIn "Madrid".
7 <http://ejemplo.es/recurso/M> <http://xmlns.com/foaf/0.1/fn> "Maria alias "Mary" " .
8 <http://ejemplo.es/recurso/M> <http://xmlns.com/foaf/0.1/fn> "Maria"^^xsd:string .
26

26


RDF inference. Entailment rules
Rule Name
if E contains
then add
rdf1
uuu aaa yyy .
aaa rdf:type rdf:Property .
rdf2
uuu aaa lll .
where lll is a well-typed XML literal .
_:nnn rdf:type rdf:XMLLiteral .
where _:nnn identifies a blank node allocated to lll by rule lg.
27

27

Exercise

- Objective
  - Get used to the different syntaxes of RDF
- Tasks
  - Take the text of an RDF file and create its corresponding graph
  - Take an RDF graph and create its corresponding Turtle files
    - Note: you can use http://rdf.greggkellogg.net/distiller to check them
    - And you can also use http://linkeddata3.dia.fi.upm.es/turtled-master/ to visualise the corresponding RDF

28

28

Exercise 1.a. Create a graph from a file

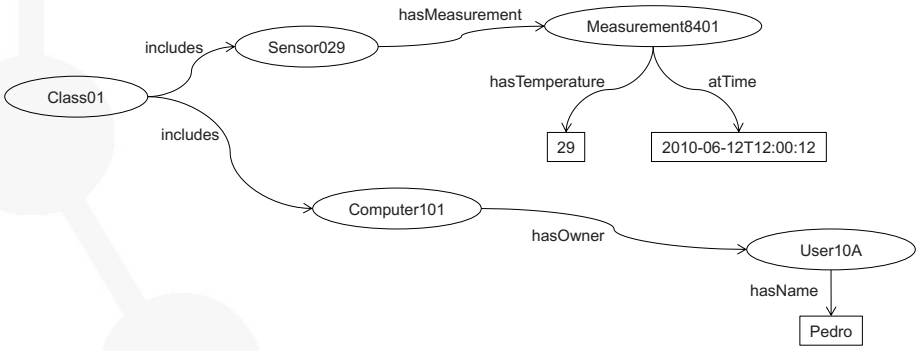
- Open the file StickyNote_PureRDF.rdf
  - Available in moodle
- Create the corresponding graph from it
- Compare your graph with those of your colleagues

29

29

Exercise 1.b. Create files from a graph

- Transform the following graph into Turtle



31

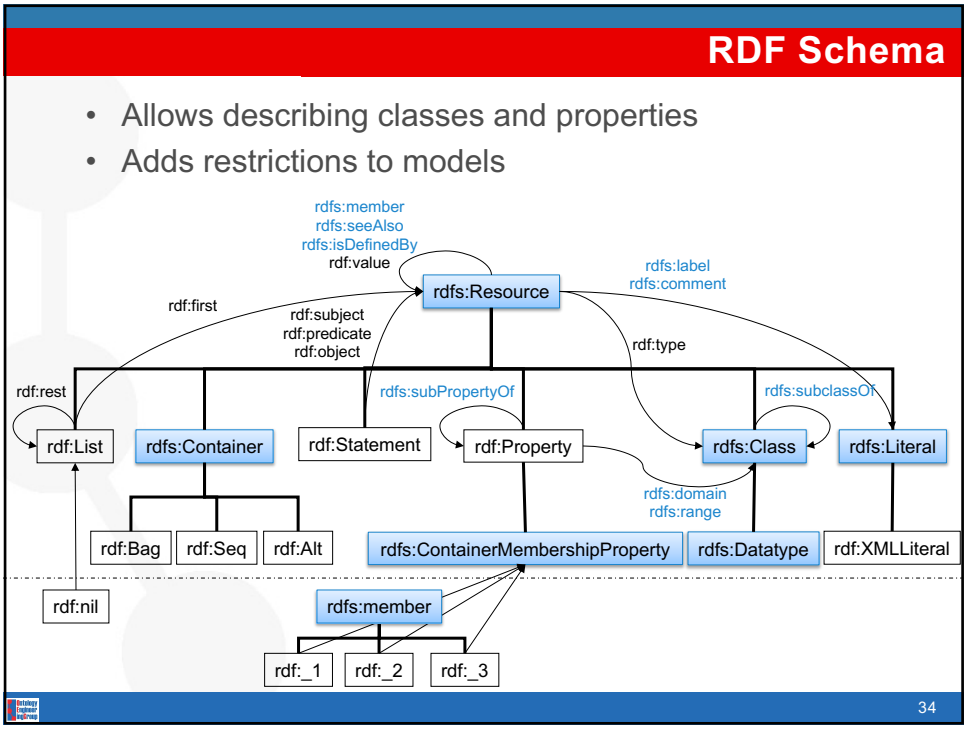
31

Índice

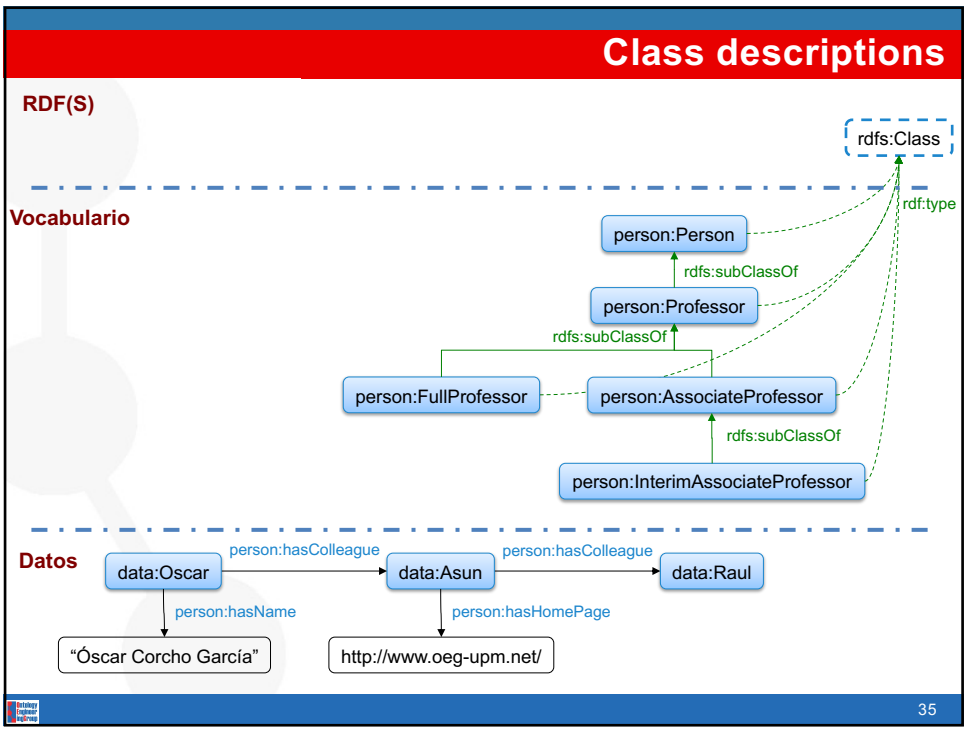
- General introduction
- RDF
  - RDF Components
  - RDF Serialisation
  - Inferences in RDF
- RDF Schema
  - RDFS Components
  - RDFS Serialisation
  - Inferences in RDFS

33

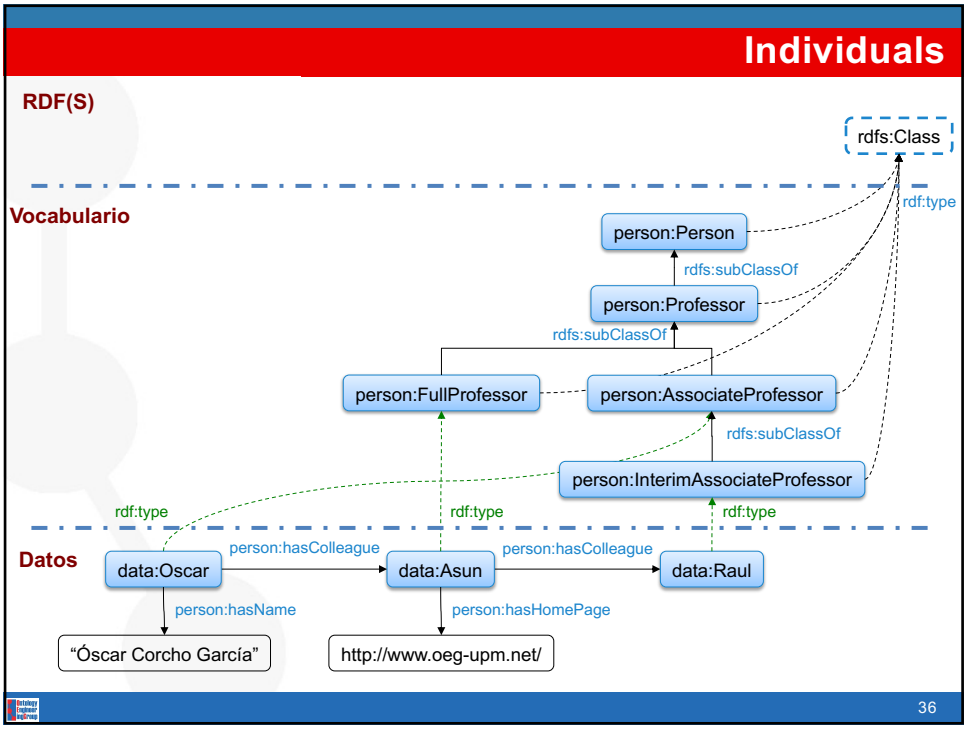
33


RDF Schema
Allows describing classes and properties
Adds restrictions to models
rdfs:member
rdfs:seeAlso
rdfs:isDefinedBy
rdf:value
rdfs:Resource
rdfs:label
rdfs:comment
rdf:first
rdf:subject
rdf:predicate
rdf:object
rdf:type
rdf:rest
rdfs:subPropertyOf
rdfs:subclassOf
rdf:List
rdfs:Container
rdf:Statement
rdf:Property
rdfs:Class
rdfs:Literal
rdfs:domain
rdfs:range
rdf:Bag
rdf:Seq
rdf:Alt
rdfs:ContainerMembershipProperty
rdfs:Datatype
rdf:XMLLiteral
rdf:nil
rdfs:member
rdf:_1
rdf:_2
rdf:_3
34

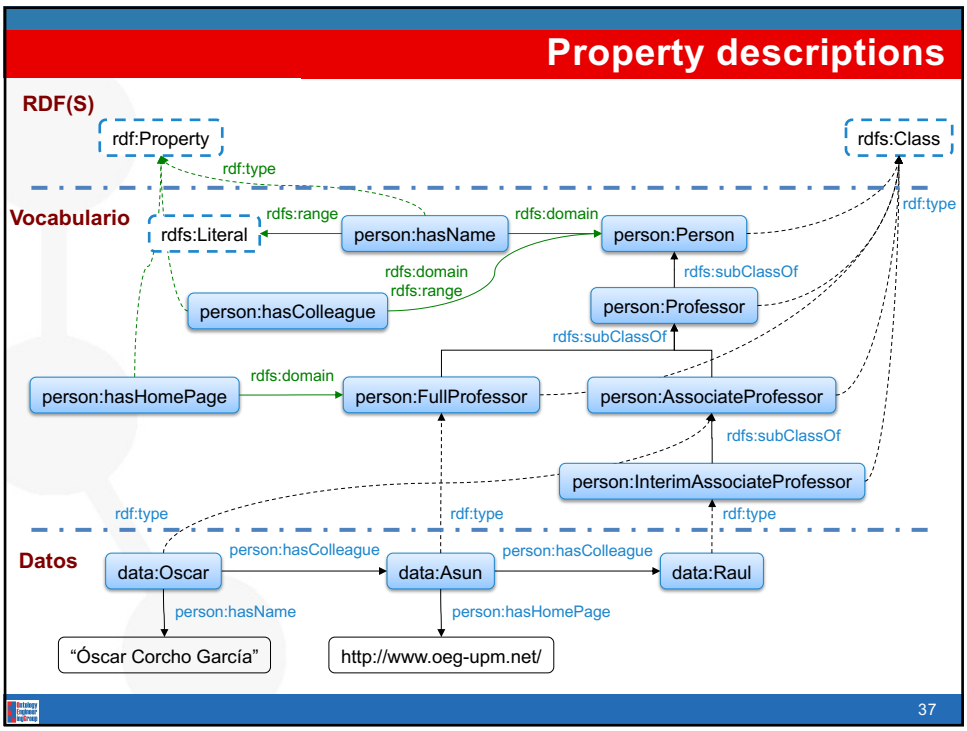
34


Class descriptions
RDF(S)
rdfs:Class
rdf:type
Vocabulario
person:Person
rdfs:subClassOf
person:Professor
rdfs:subClassOf
person:FullProfessor
person:AssociateProfessor
rdfs:subClassOf
person:InterimAssociateProfessor
Datos
person:hasColleague
person:hasColleague
data:Oscar
data:Asun
data:Raul
person:hasName
person:hasHomePage
“Óscar Corcho García”
http://www.oeg-upm.net/
35

35

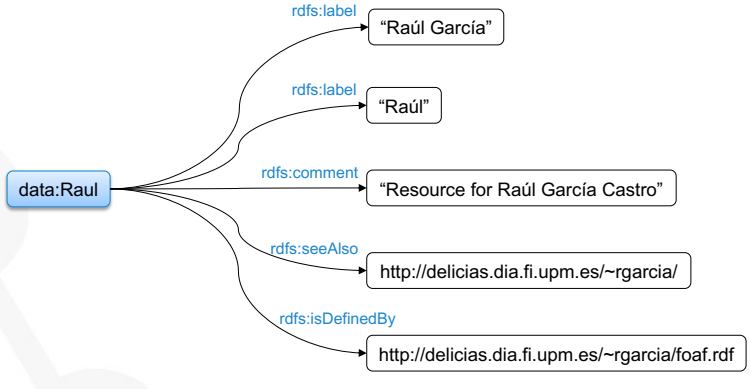

Individuals
RDF(S)
rdfs:Class
Vocabulario
rdf:type
person:Person
rdfs:subClassOf
person:Professor
rdfs:subClassOf
person:FullProfessor
person:AssociateProfessor
rdfs:subClassOf
person:InterimAssociateProfessor
rdf:type
rdf:type
rdf:type
Datos
person:hasColleague
person:hasColleague
data:Oscar
data:Asun
data:Raul
person:hasName
person:hasHomePage
"Óscar Corcho García"
http://www.oeg-upm.net/
36

36


Property descriptions
RDF(S)
rdf:Property
rdfs:Class
rdf:type
Vocabulario
rdf:type
rdfs:range
rdfs:Literal
person:hasName
rdfs:domain
person:Person
rdfs:domain
rdfs:range
rdfs:subClassOf
person:hasColleague
person:Professor
rdfs:subClassOf
rdfs:domain
person:hasHomePage
person:FullProfessor
person:AssociateProfessor
rdfs:subClassOf
person:InterimAssociateProfessor
rdf:type
rdf:type
rdf:type
Datos
person:hasColleague
person:hasColleague
data:Oscar
data:Asun
data:Raul
person:hasName
person:hasHomePage
"Óscar Corcho García"
http://www.oeg-upm.net/
37

37

Resource annotation



38

38

RDF Schema Serialisation. RDF/XML (1/2)

```
<?xml version="1.0"?>

   <!DOCTYPE rdf:RDF [
     <!ENTITY rdfs "http://www.w3.org/2000/01/rdf-schema#" >
     <!ENTITY rdf "http://www.w3.org/1999/02/22-rdf-syntax-ns#" >
     <!ENTITY person "http://www.oeg-upm.net/ontologies/person#" >
   ]>

   <rdf:RDF xmlns="http://www.oeg-upm.net/ontologies/person#"
        xml:base="http://www.oeg-upm.net/ontologies/person"
        xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
        xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
        xmlns:person="http://www.oeg-upm.net/ontologies/person#">

     <rdfs:Property rdf:about="&person;hasColleague">
       <rdfs:range rdf:resource="#Person"/>
       <rdfs:domain rdf:resource="#Person"/>
     </rdfs:Property>

     <rdfs:Property rdf:about="&person;hasHomePage">
       <rdfs:domain rdf:resource="#FullProfessor"/>
     </rdfs:Property>

     <rdfs:Property rdf:about="&person;hasName">
       <rdfs:domain rdf:resource="#Person"/>
       <rdfs:range rdf:resource="&rdfs;Literal"/>
     </rdfs:Property>
...
```

40

40

RDF Schema Serialisation. RDF/XML (2/2)

```
...
    <rdfs:Class rdf:about="#AssociateProfessor">
      <rdfs:subClassOf rdf:resource="#Professor"/>
    </rdfs:Class>

    <rdfs:Class rdf:about="#FullProfessor">
      <rdfs:subClassOf rdf:resource="#Professor"/>
    </rdfs:Class>

    <rdfs:Class rdf:about="#InterimAssociateProfessor">
      <rdfs:subClassOf rdf:resource="#AssociateProfessor"/>
    </rdfs:Class>

    <rdfs:Class rdf:about="#Person"/>

    <rdfs:Class rdf:about="#Professor">
      <rdfs:subClassOf rdf:resource="#Person"/>
    </rdfs:Class>

    <FullProfessor rdf:about="#Asun">
      <person:hasHomePage> http://www.oeg-upm.net/</person:hasHomePage>
      <person:hasColleague rdf:resource="#Raul"/>
    </FullProfessor>

    <AssociateProfessor rdf:about="#Oscar">
      <person:hasName>Oscar Corcho García</person:hasName>
      <person:hasColleague rdf:resource="#Asun"/>
    </AssociateProfessor>

    <InterimAssociateProfessor rdf:about="#Raul"/>
  </rdf:RDF>
```

41

41

RDF Schema Serialisation. Turtle (1/2)

```
@base <http://www.oeg-upm.net/ontologies/person> .
@prefix : <http://www.oeg-upm.net/ontologies/person#> .
@prefix person: <http://www.oeg-upm.net/ontologies/person#> .

:hasColleague a rdf:Property .
        rdfs:domain :Person .
        rdfs:range :Person ;

:hasHomePage a rdf:Property .
        rdfs:domain :FullProfessor .

:hasName a rdf:Property .
      rdfs:domain :Person ;
      rdfs:range rdfs:Literal .
...
```

a is equivalent to rdf:type

42

42

RDF Schema Serialisation. Turtle (2/2)

```
…
:Person a rdfs:Class .

:Professor a rdfs:Class ;
        rdfs:subClassOf :Person .

:FullProfessor a rdfs:Class ;
          rdfs:subClassOf :Professor .

:AssociateProfessor a rdfs:Class ;
              rdfs:subClassOf :Professor .

:InterimAssociateProfessor a rdfs:Class ;
                  rdfs:subClassOf :AssociateProfessor .

:Asun a :FullProfessor ;
    :hasHomePage "http://www.oeg-upm.net/" ;
    :hasColleague :Raul .

:Oscar a :AssociateProfessor ;
     :hasName "Oscar Corcho García" ;
     :hasColleague :Asun .

:Raul a :InterimAssociateProfessor .
```

a is equivalent to rdf:type

43

43

RDF(S) inference. Entailment rules

Rule Name	If E contains:	then add:
rdfs1	uuu aaa lll. where lll is a plain literal (with or without a language tag).	_:nnn rdf:type rdfs:Literal . where _:nnn identifies a blank node allocated to lll by rule rule lg.
rdfs2	aaa rdfs:domain XXX . uuu aaa yyy .	uuu rdf:type XXX .
rdfs3	aaa rdfs:range XXX . uuu aaa vvv .	vvv rdf:type XXX .
rdfs4a	uuu aaa xxx .	uuu rdf:type rdfs:Resource .
rdfs4b	uuu aaa vvv.	vvv rdf:type rdfs:Resource .
rdfs5	uuu rdfs:subPropertyOf vvv . vvv rdfs:subPropertyOf XXX .	uuu rdfs:subPropertyOf XXX .
rdfs6	uuu rdf:type rdf:Property .	uuu rdfs:subPropertyOf uuu .
rdfs7	aaa rdfs:subPropertyOf bbb . uuu aaa yyy .	uuu bbb yyy .
rdfs8	uuu rdf:type rdfs:Class .	uuu rdfs:subClassOf rdfs:Resource .
rdfs9	uuu rdfs:subClassOf XXX . vvv rdf:type uuu .	vvv rdf:type XXX .
rdfs10	uuu rdf:type rdfs:Class .	uuu rdfs:subClassOf uuu .
rdfs11	uuu rdfs:subClassOf vvv . vvv rdfs:subClassOf XXX .	uuu rdfs:subClassOf XXX .
rdfs12	uuu rdf:type rdfs:ContainerMembershipProperty .	uuu rdfs:subPropertyOf rdfs:member .
rdfs13	uuu rdf:type rdfs:Datatype .	uuu rdfs:subClassOf rdfs:Literal .

44

44

RDF(S) inference. Additional inferences

ext1	UUU rdfs:domain VV . VV rdfs:subClassOf ZZZ .	UUU rdfs:domain ZZZ .
ext2	UUU rdfs:range VV . VV rdfs:subClassOf ZZZ .	UUU rdfs:range ZZZ .
ext3	UUU rdfs:domain VV . WWW rdfs:subPropertyOf UUU .	WWW rdfs:domain VV .
ext4	UUU rdfs:range VV . WWW rdfs:subPropertyOf UUU .	WWW rdfs:range VV .
ext5	rdf:type rdfs:subPropertyOf WWW . WWW rdfs:domain VV .	rdfs:Resource rdfs:subClassOf VV .
ext6	rdfs:subClassOf rdfs:subPropertyOf WWW . WWW rdfs:domain VV .	rdfs:Class rdfs:subClassOf VV .
ext7	rdfs:subPropertyOf rdfs:subPropertyOf WWW . WWW rdfs:domain VV .	rdf:Property rdfs:subClassOf VV .
ext8	rdfs:subClassOf rdfs:subPropertyOf WWW . WWW rdfs:range VV .	rdfs:Class rdfs:subClassOf VV .
ext9	rdfs:subPropertyOf rdfs:subPropertyOf WWW . WWW rdfs:range VV .	rdf:Property rdfs:subClassOf VV .

45

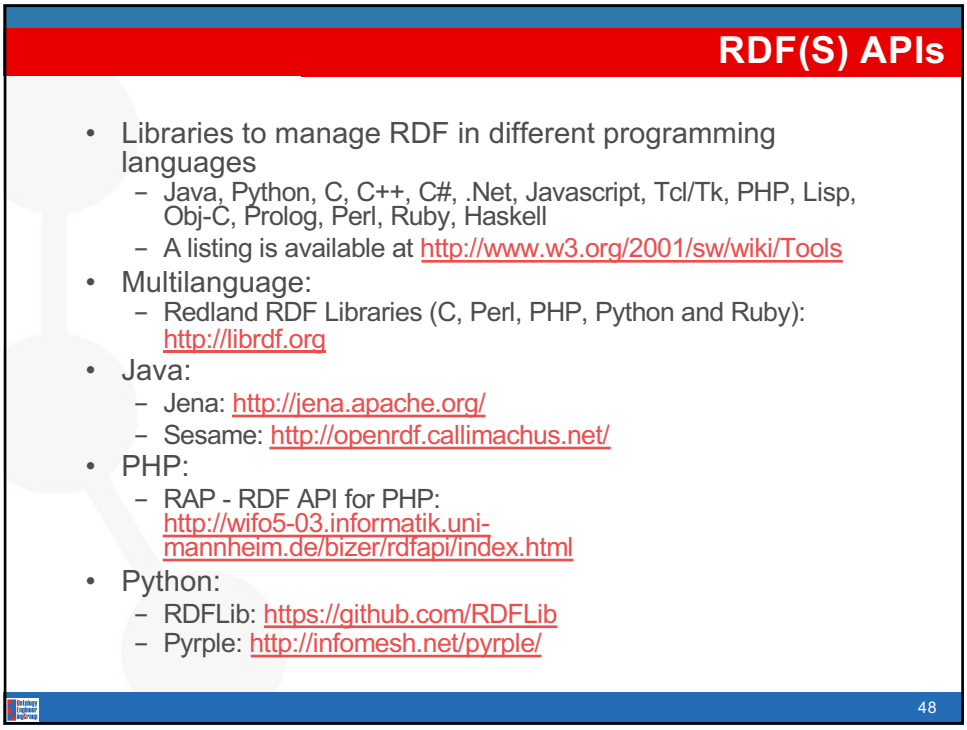
45

RDF(S) limitations

- RDFS is 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 boolean operators
    - Can't say or, not, etc.
  - 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
  - May be possible to reason via FOL axiomatisation

47

47


RDF(S) APIs
• Libraries to manage RDF in different programming languages
– Java, Python, C, C++, C#, .Net, Javascript, Tcl/Tk, PHP, Lisp, Obj-C, Prolog, Perl, Ruby, Haskell
– A listing is available at http://www.w3.org/2001/sw/wiki/Tools
• Multilanguage:
– Redland RDF Libraries (C, Perl, PHP, Python and Ruby): http://librdf.org
• Java:
– Jena: http://jena.apache.org/
– Sesame: http://openrdf.callimachus.net/
• PHP:
– RAP - RDF API for PHP: http://wifo5-03.informatik.uni-mannheim.de/bizer/rdfapi/index.html
• Python:
– RDFLib: https://github.com/RDFLib
– Pyrple: http://infomesh.net/pyrple/
48
48