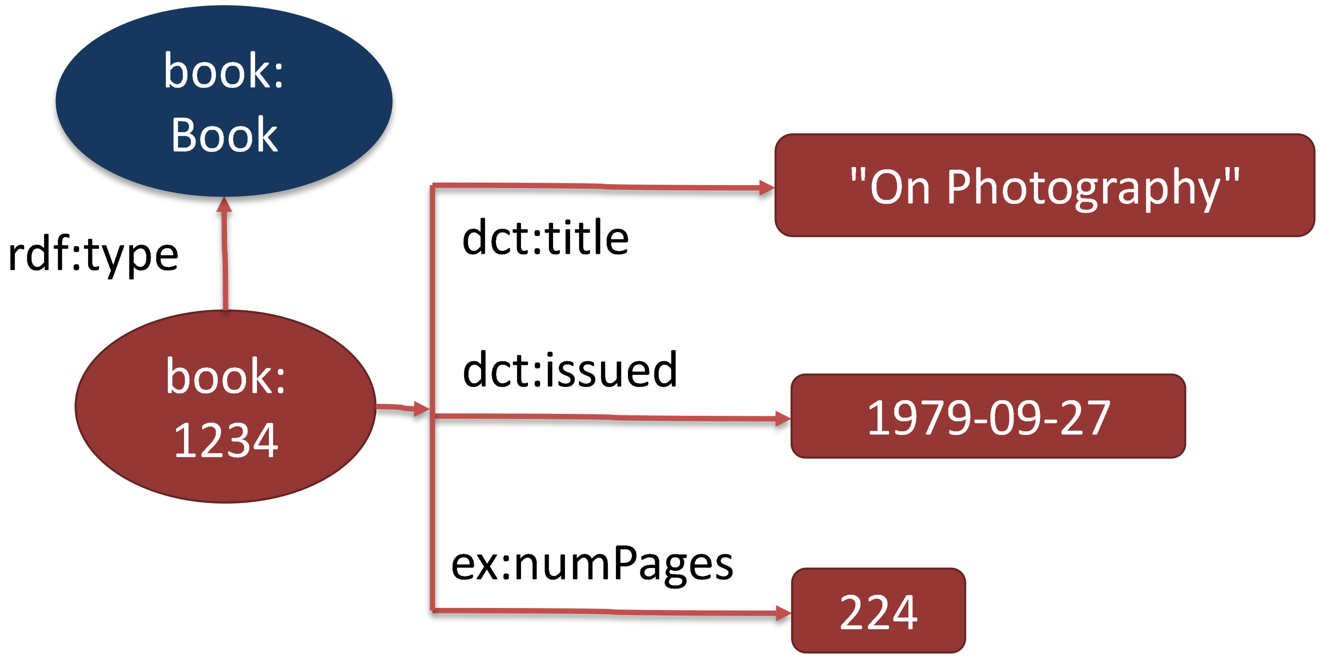
Introduction to linked data

RDF Standards Revisited

epimorphics

CONNECTING DATA

The RDF Conceptual Model



Datatypes

- Literal values have a datatype
  - e.g. string, date, integer, …
- So far we have been writing them in short form
- Really they have two parts
  - a string part, e.g. "1979-09-27"
  - a datatype, e.g. xsd:date
- Datatypes are identified by a URI
- Written in full a literal typically looks like this
  - "1979-09-27"^^xsd:date
- RDF implementation often store a typed value in 'canonical form'

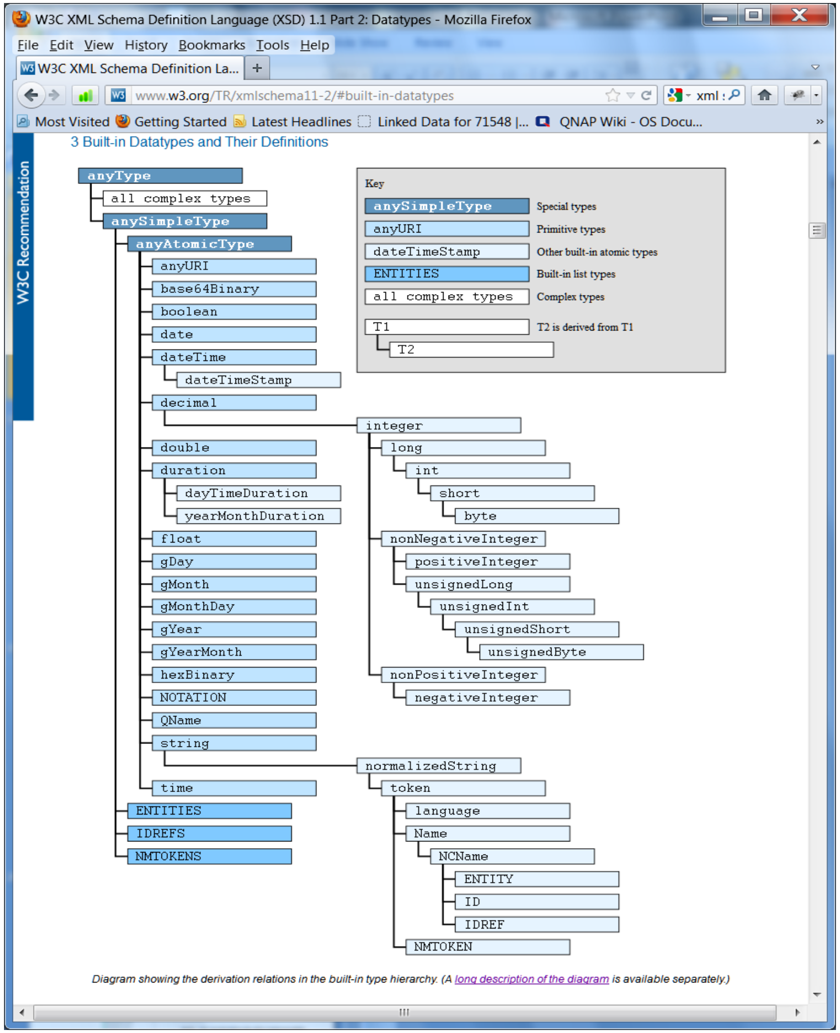
"On Photography"

1979-09-27

224

Standard Datatypes

- XML Schema defines a collection of datatypes
- Best practice is to use those datatypes
  - RDF software understands them
- You can also define your own datatypes



W3C XML Schema Definition Language (XSD) 1.1 Part 2: Datatypes - Mozilla Firefox

3 Built-in Datatypes and Their Definitions

Diagram showing the derivation relations in the built-in type hierarchy. (A long description of the diagram is available separately.)

Datatype characteristics

- A datatype defines a mapping from strings to values
  - "1"^^xsd:integer => the integer 1
  - "0001"^^xsd:integer => the integer 1
- the values define equality and ordering
- which may be different to string ordering of the lexical form

epimorphics
CONNECTING DATA

RDF Specific Datatypes

- rdf:XMLLiteral
  - a datatype for representing fragments of XML in literal values
  - value space is DOM fragment nodes
- rdf:HTML
  - a datatype for representing fragment of HTML in literal values
  - value space is lists of DOM fragments

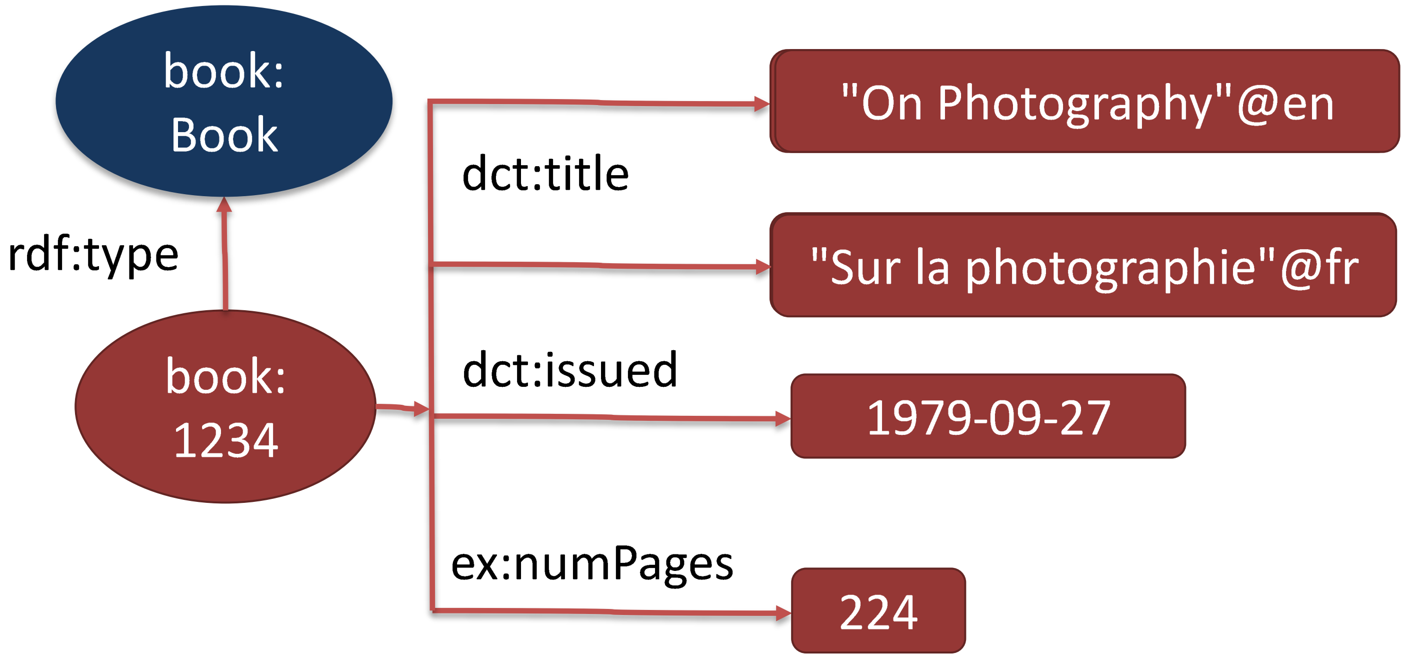
Exercise

- Represent the literals in full with datatypes

"On Photography"	"On Photography"^^xsd:string
1979-09-27	"1979-09-27"^^xsd:date
224	"224"^^xsd:integer

http://www.w3.org/TR/xmlschema-2/#built-in-datatypes

Language Tags



The datatype for lang strings is rdf:langString

Multiple Graphs

- Exercise
  - – An RDF graph is a thing – it's a collection of data
  - – When might you encounter or choose to have multiple graphs?
- Possible answers
  - separate graphs so can update them independently
  - different versions in different graphs
  - more generally different contexts
  - different publishers publishing inconsistent graphs
  - …

Datasets

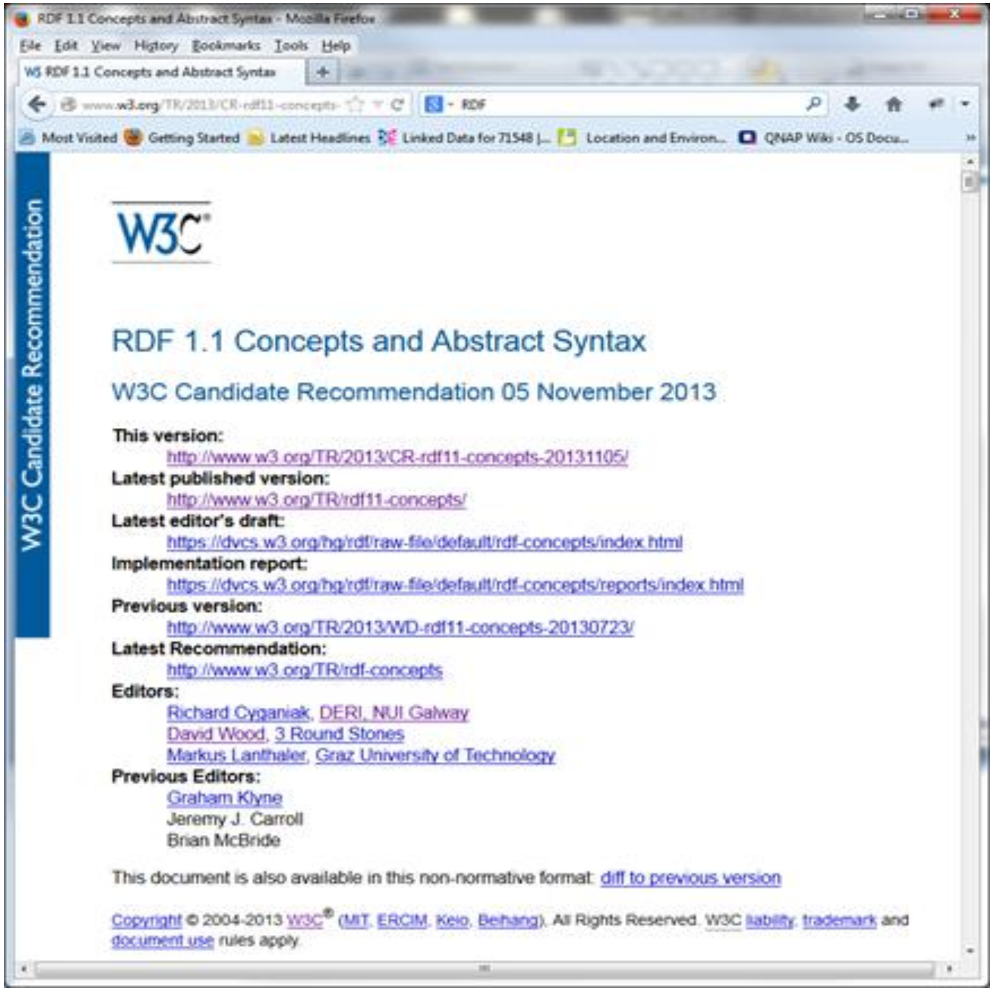
- Datasets are collections of graphs
  - RDF databases aka triple stores implement datasets
- A dataset consists of:
  - a default graph
  - a collection of named graphs
    - A named graph is a pair (URI or blank node, RDF graph)
- A named graph can be used to represent a context
  - versions
  - publisher
- Named graphs often used for data management
  - easy to add/delete/replace as a unit
  - default graph often the union of the named graphs

Blank Nodes

- Sometimes we just don't want to create a URI to identify something
  - we don't want to commit to maintaining the URI
  - we don't know what resource we are describing – but we want to record a partial description
  - we can't create a 'good' URI from the information we have
  - its just too burdensome to create a URI
- We can create a blank node in the graph to represent these
  - its just a resource without a URI
- Blank nodes have no identity outside the dataset they are in
- They can be problematic – e.g. round tripping

Questions?

RDF: a way to represent data


RDF 1.1 Concepts and Abstract Syntax - Mozilla Firefox
File Edit View History Bookmarks Tools Help
W3 RDF 1.1 Concepts and Abstract Syntax
www.w3.org/TR/2013/CR-rdf11-concepts-
RDF
Most Visited
Getting Started
Latest Headlines
Linked Data for 71548 |...
Location and Environ...
QNAP Wiki - OS Docu...
W3C Candidate Recommendation
W3C

RDF 1.1 Concepts and Abstract Syntax

W3C Candidate Recommendation 05 November 2013

This version:
http://www.w3.org/TR/2013/CR-rdf11-concepts-20131105/
Latest published version:
http://www.w3.org/TR/rdf11-concepts/
Latest editor's draft:
https://dvcs.w3.org/hg/rdf/raw-file/default/rdf-concepts/index.html
Implementation report:
https://dvcs.w3.org/hg/rdf/raw-file/default/rdf-concepts/reports/index.html
Previous version:
http://www.w3.org/TR/2013/WD-rdf11-concepts-20130723/
Latest Recommendation:
http://www.w3.org/TR/rdf-concepts
Editors:
Richard Cyganiak, DERI, NUI Galway
David Wood, 3 Round Stones
Markus Lanthaler, Graz University of Technology
Previous Editors:
Graham Klyne
Jeremy J. Carroll
Brian McBride

This document is also available in this non-normative format: diff to previous version

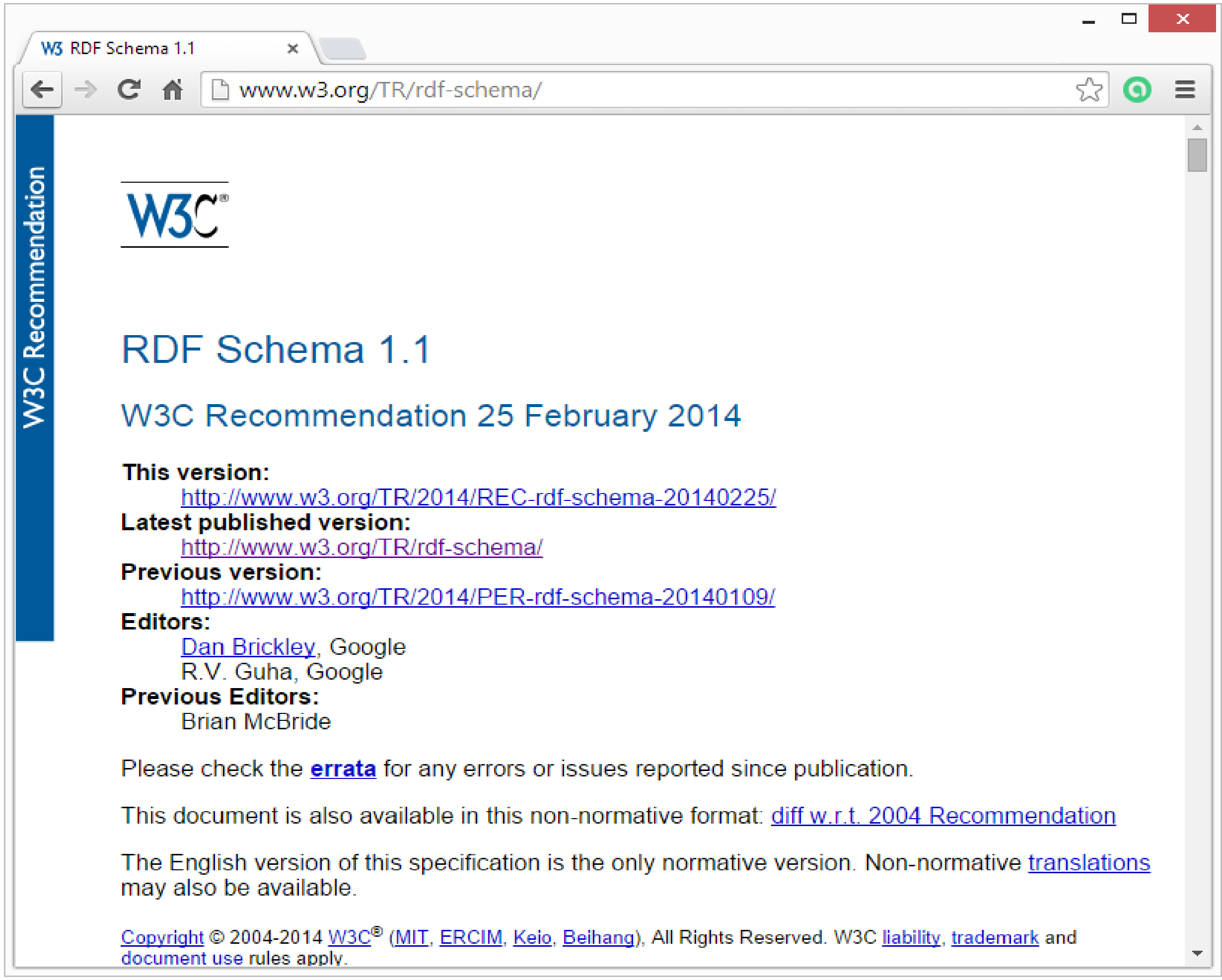
Copyright © 2004-2013 W3C® (MIT, ERCIM, Keio, Beihang), All Rights Reserved. W3C liability, trademark and document use rules apply.

- “Web Standard”
- Designed to support Linking
- Good for data integration

epimorphics
CONNECTING DATA

RDFS: The RDF Schema Language

- “Web Standard”
- AKA: RDF Vocabulary Description Language
- Supports the description of types and properties



RDF Schema 1.1

W3C Recommendation 25 February 2014

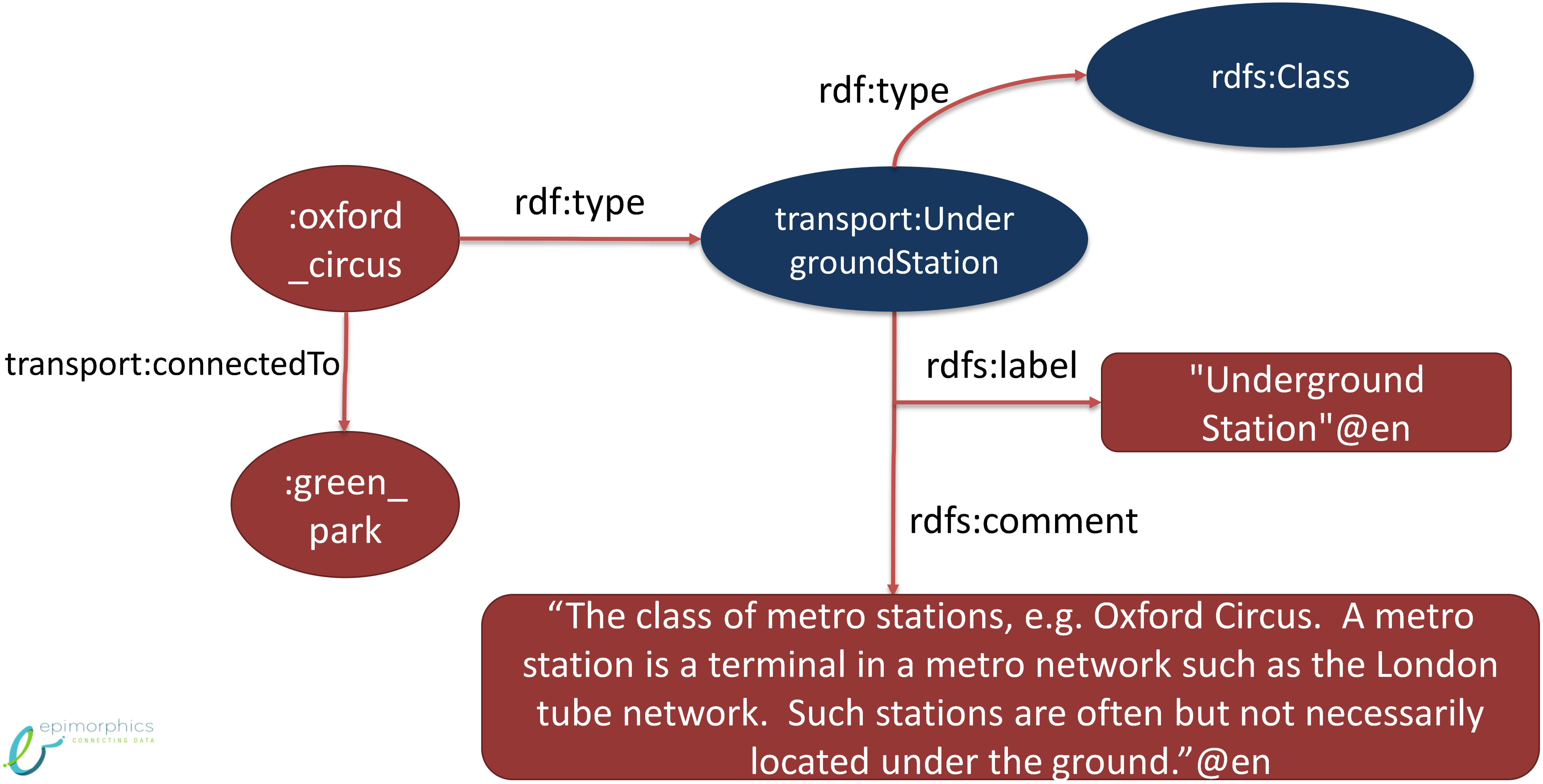
This version:
http://www.w3.org/TR/2014/REC-rdf-schema-20140225/
Latest published version:
http://www.w3.org/TR/rdf-schema/
Previous version:
http://www.w3.org/TR/2014/PER-rdf-schema-20140109/
Editors:
Dan Brickley, Google
R.V. Guha, Google
Previous Editors:
Brian McBride

Please check the **errata** for any errors or issues reported since publication.

This document is also available in this non-normative format: diff w.r.t. 2004 Recommendation

The English version of this specification is the only normative version. Non-normative translations may also be available.

Data Modelling
How do we describe a type?

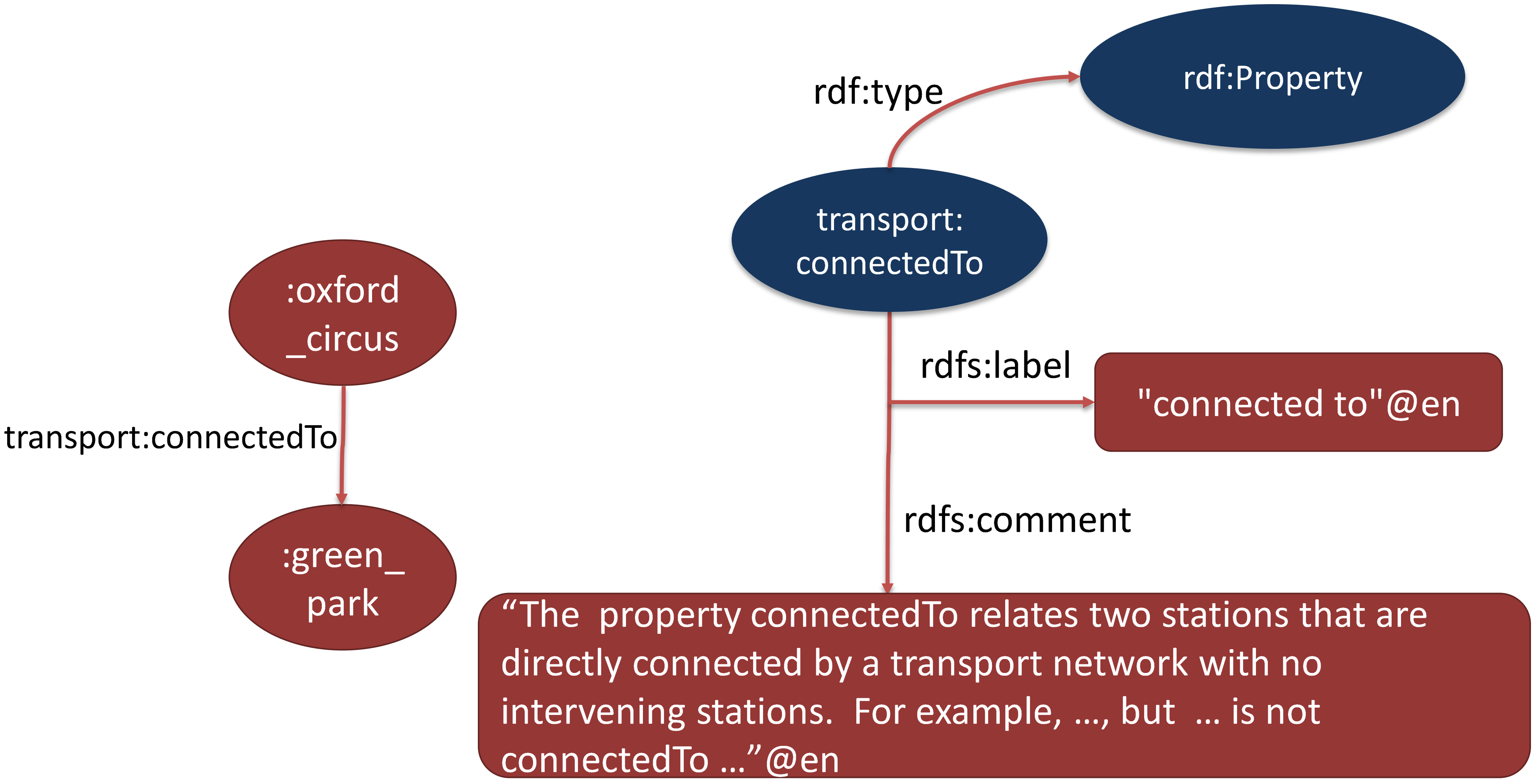


"The class of metro stations, e.g. Oxford Circus. A metro station is a terminal in a metro network such as the London tube network. Such stations are often but not necessarily located under the ground."@en

Defining a class

- Don't skimp on the documentation
- Intuition – what is the class about
- An example
- Necessary conditions for being an instance of the class
- Sufficient conditions for being an instance of the class
- Clarifying information
- Clarifying non-examples

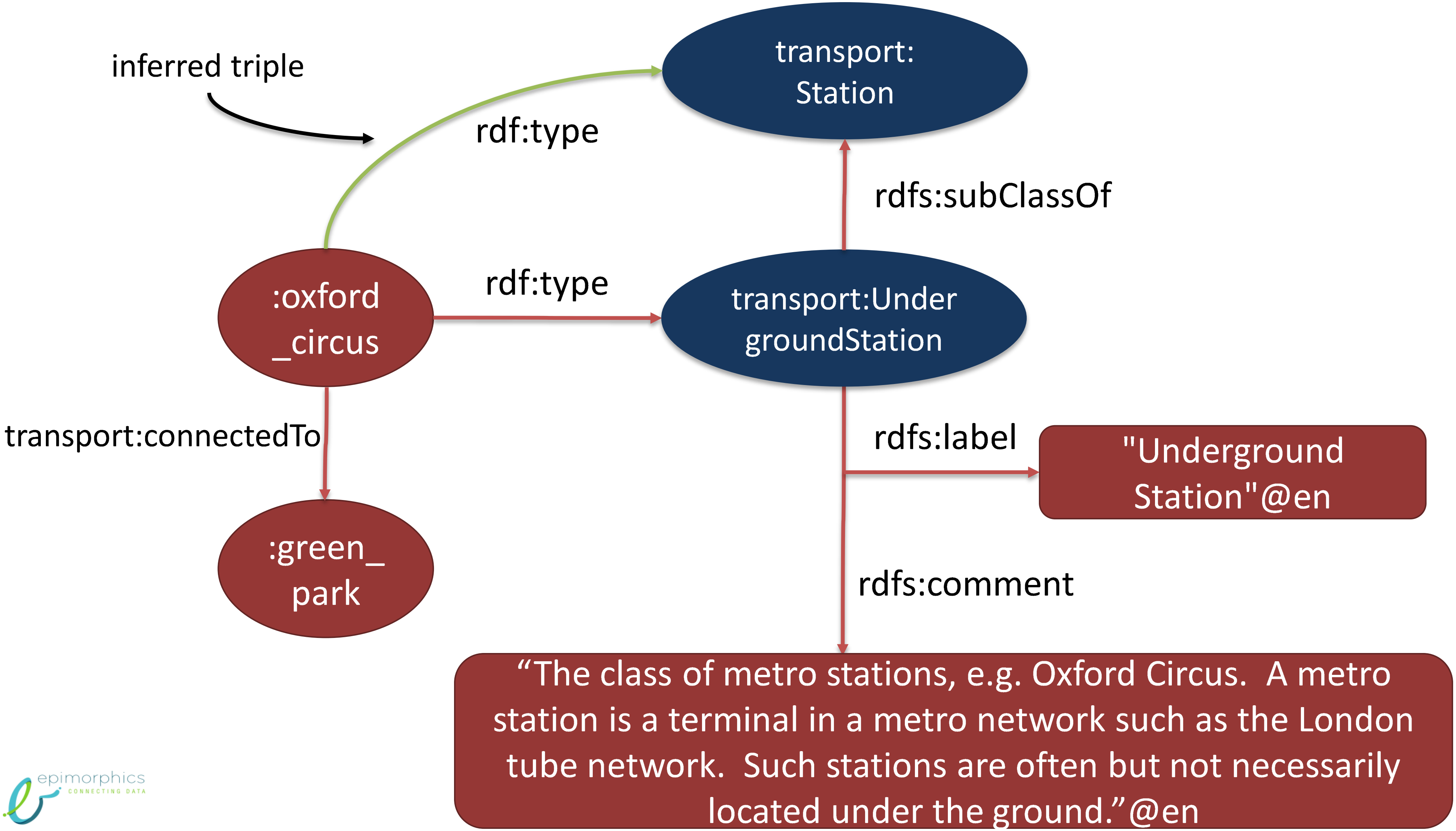
Properties are similar



rdf:type
rdf:Property
transport:
connectedTo
:oxford
_circus
rdfs:label
"connected to"@en
transport:connectedTo
rdfs:comment
:green_
park
“The property connectedTo relates two stations that are directly connected by a transport network with no intervening stations. For example, …, but … is not connectedTo …”@en

epimorphics
CONNECTING DATA

Subclass relationships



inferred triple

transport:
Station

rdf:type

rdfs:subClassOf

:oxford
_circus

rdf:type

transport:Under
groundStation

transport:connectedTo

rdfs:label

"Underground
Station"@en

:green_
park

rdfs:comment

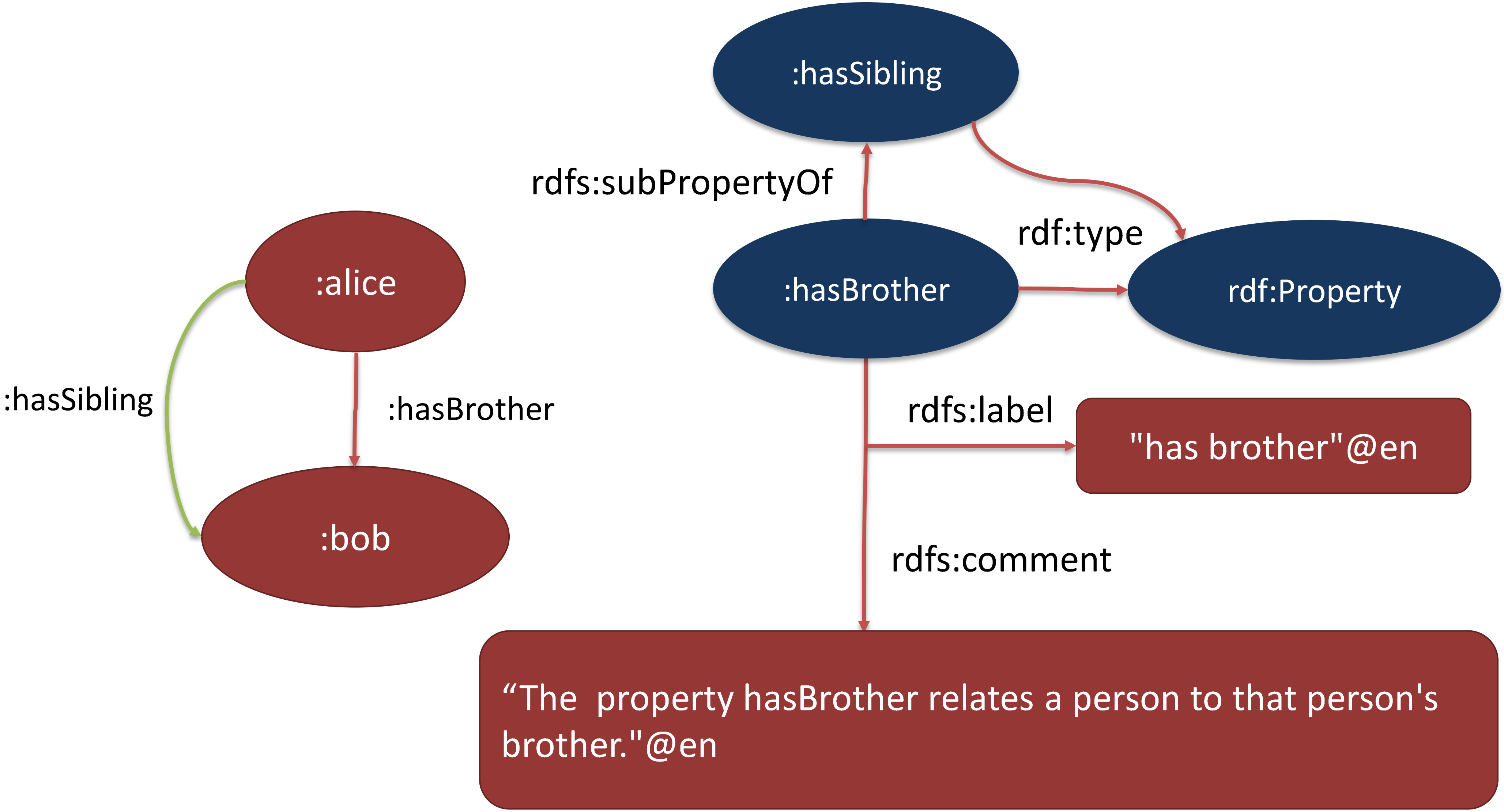
“The class of metro stations, e.g. Oxford Circus. A metro station is a terminal in a metro network such as the London tube network. Such stations are often but not necessarily located under the ground.”@en

epimorphics
CONNECTING DATA

A word about inference

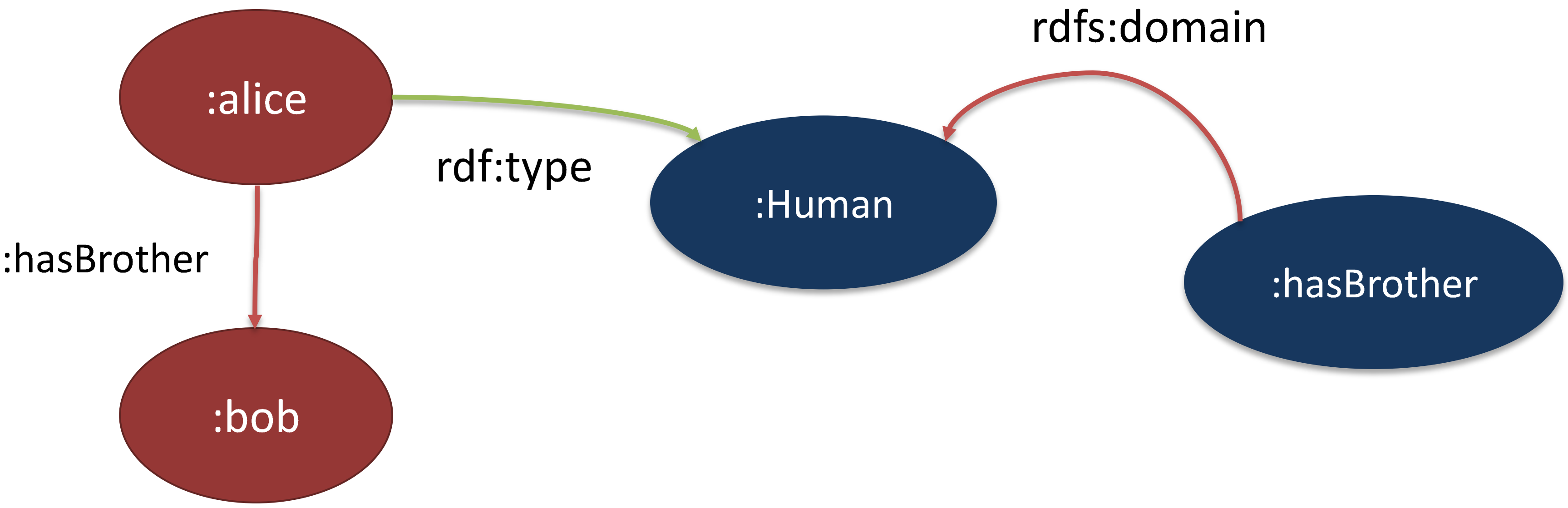
- Some RDF implementations will automatically 'fill in' the inferred triples
- Inference is computationally expensive
  - Especially on large datasets
- Check what your tool chain supports
  - some inference is more expensive than others
- Can do some in SPARQL directly
  - e.g find all the things of type transport:Station (including subclasses)
- Can create manifest triples on ingest
  - Take a small quantity of data, add the vocabs, run inference and load the resulting triples into the triple store
- Plan your use and test at scale

Subproperty relationships



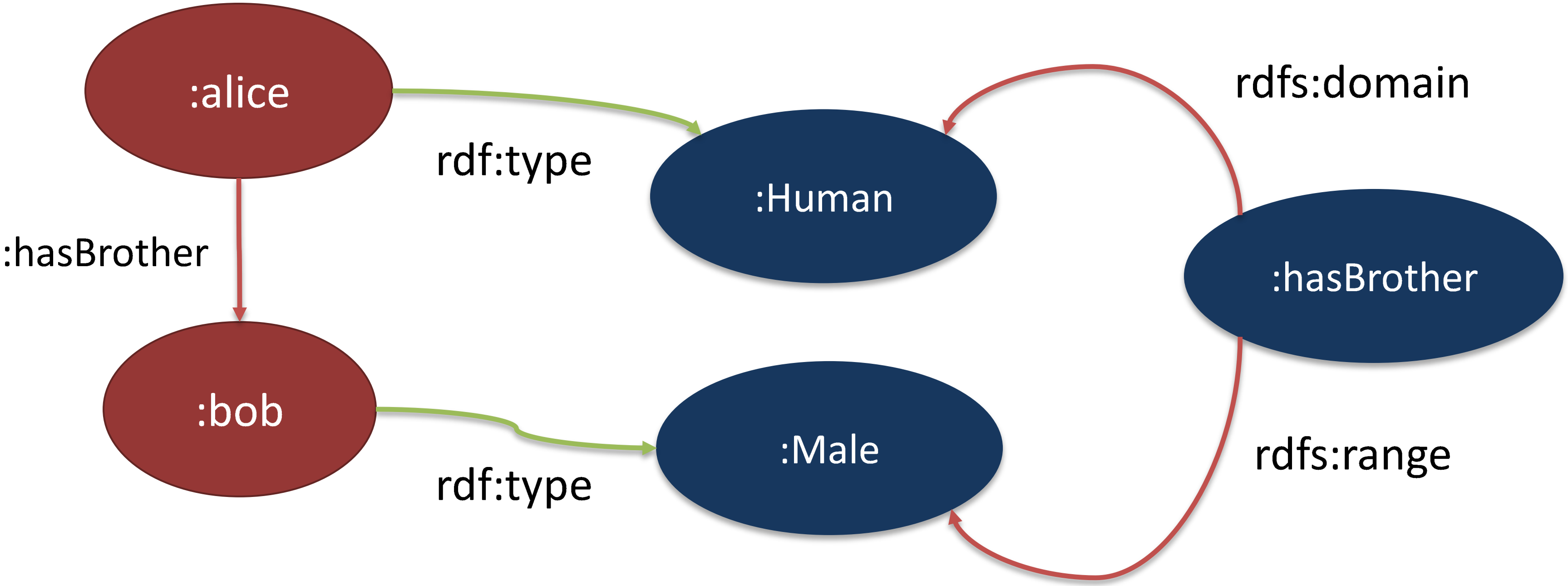
epimorphics
CONNECTING DATA

rdfs:domain



Anything at the **BLUNT** end of a :hasBrother property arc must be an instance of :Human

rdfs:range



Anything at the **SHARP** end of a :hasBrother property arc must be an instance of :Male

Exercises

- What is the domain of the :husbandOf?
  - :Human
- What is the range of the ex:dateOfBirth?
  - xsd:date
- What subclasses of :Dwelling might there be?
  - :Cave, :Wigwam, :Yurt, :Houseboat, …
- What subproperties of :hasSpouse might there be?
  - :hasHusband, :hasWife

Observations about linked data modelling

- Its not primarily about designing a data structure
- It is about building a conceptual model of the world
  - NOT: There is a table called stations with columns …
  - IS: There are things called stations and they have properties …
  - The computer figures out how to store the data
- It is based on logic and logical inference
- The result is often called an ONTOLOGY
- Usually ontologies are designed to be context free
- It is easy to underestimate the difference that makes

epimorphics
CONNECTING DATA

Don’t Reinvent the Wheel

- There are bunch of vocabularies out there already defined
- Before inventing your own – look at what’s out there already
- Evaluate what you find
  - Provenance – is the work of recognised standards body or is it a student project?
  - Does it do (some of) what you need
    - You can extend it; you don't need to use it all
  - Is it widely used and accepted?

epimorphics
CONNECTING DATA

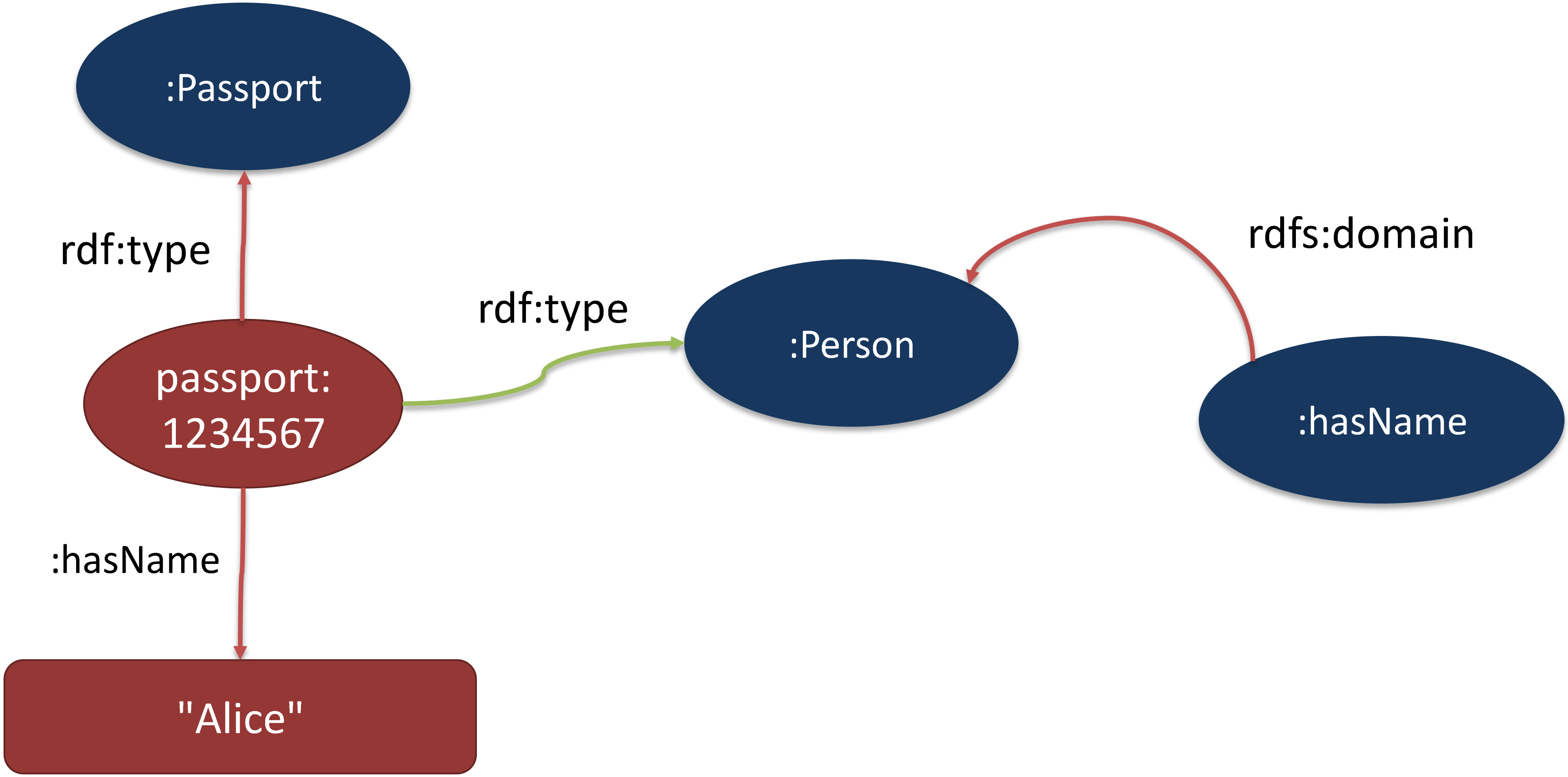
Examples of existing commonly used vocabularies

- Dublin core
  - Use for common document metadata, title, author etc
- FRBR
  - A model for relating abstract works (Bethoven’s 9th), peformances, recodings and disks you can play on your 78 player
- FIBO
  - An ontology for representing financial data
- schema.org
  - GOOGLE info boxes

Examples of existing commonly used vocabularies

- SKOS
  - Use for codelists, controlled vocabularies and taxonomies
- ORG
  - Use for org charts
- CUBE
  - Use for tables, spreadsheets, n-dimensional statistical data etc
- …

What is wrong with this?



The Web Ontology Language (OWL)

- RDF Schema (rdfs:)
  - Class, subclass, property, subproperty, domain, range
- Web Ontology Language (owl:)
  - extension of RDFS
  - Much more powerful and expressive
    - and computationally expensive
  - It has negation – RDFS does not
- For many applications
  - RDFS will suffice
  - But its common to use a bit of OWL to be able to say a bit more

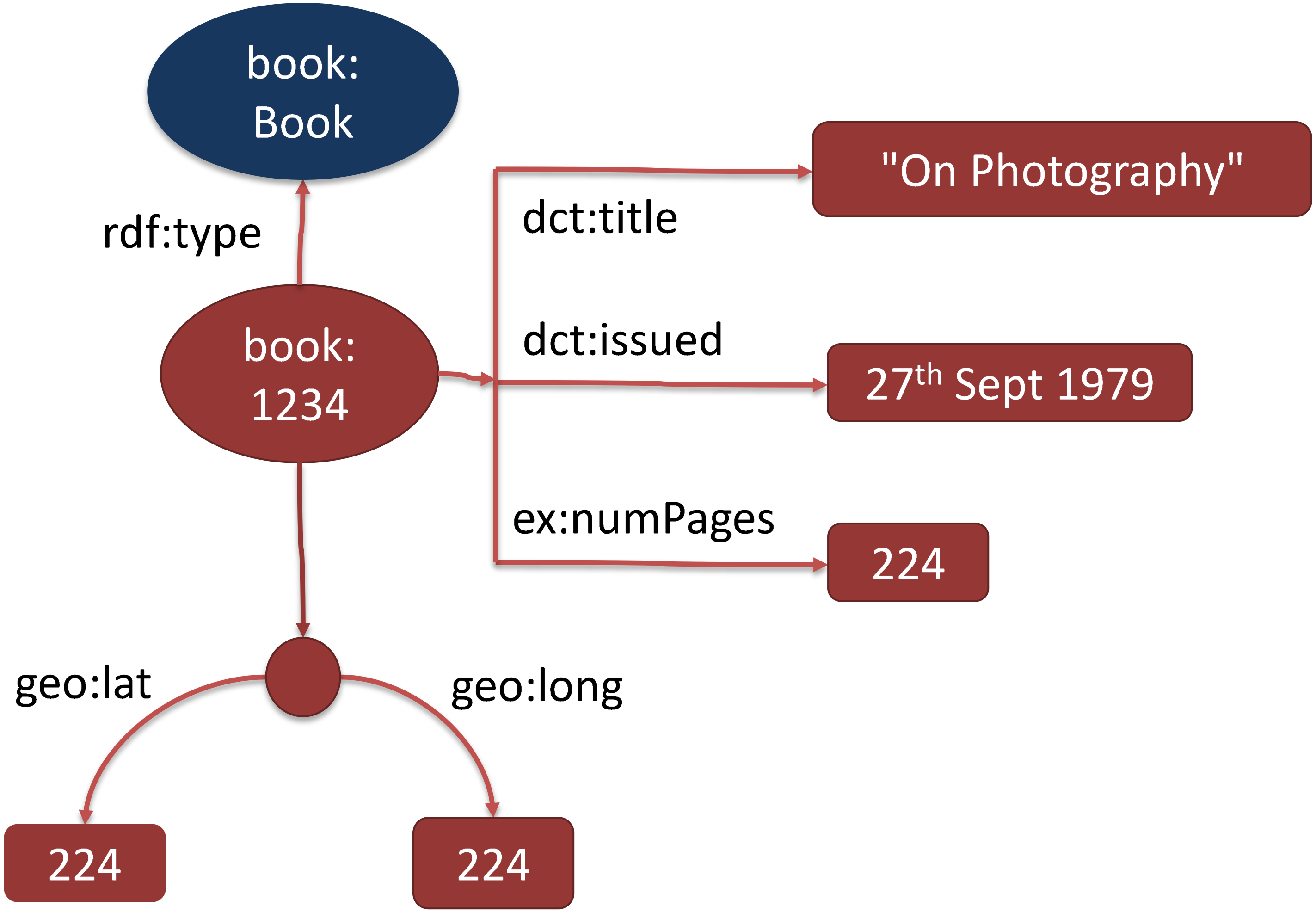
Questions/Observations

epimorphics

CONNECTING DATA

www.epimorphics.com

Bubble and Arc components


book:
Book
rdf:type
book:
1234
dct:title
"On Photography"
dct:issued
27th Sept 1979
ex:numPages
224
geo:lat
geo:long
224
224