Knowledge Representation & Reasoning
&
Introduction To Knowledge Graphs

Week 4 & 5 | Fall 2022

Dr. Amna Basharat

Knowledge Is Power?

Agree/Disagree?

Knowledge Is the Key to Ultimate Success?

Agree/Disagree

If yes, to what extent?

How Would You Say?

"My dear Watson, ...I suppose that it was the gardener who has killed the butler!"

RDF Reification

Making Statements about Statements

RDF Reification

- RDF also permits interleaving of statements, i.e. to make statements about statements
- Example:
  - Sherlock Holmes supposes that the gardener has killed the butler
-
  - **Part 1**: the gardener has killed the butler

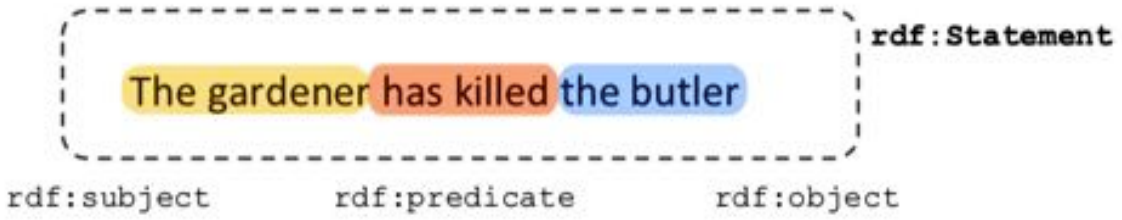
    ```
    ex:Gardener ex:hasKilled  ex:butler .
    ```

  - **Part 2**: Sherlock Holmes supposes...

    ```
    dbpedia:Sherlock_Holmes ex:supposes ????
    ```

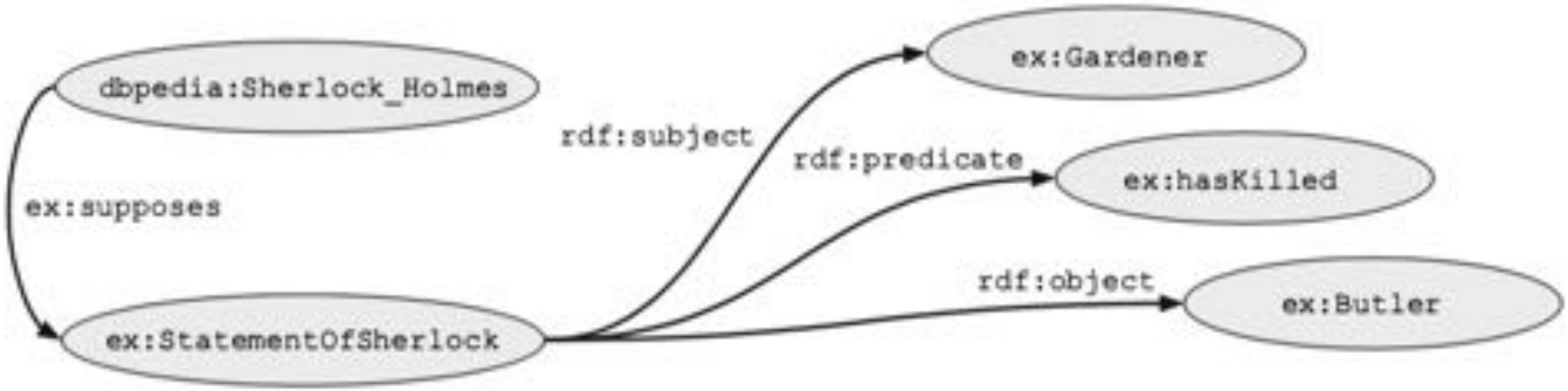
RDF Reification

- **rdf:Statement**
  defines an RDF statement consisting of subject, predicate, object
-
  - `rdf:subject` - the described resource
  - `rdf:predicate` - the original property
  - `rdf:object` - the value of the property



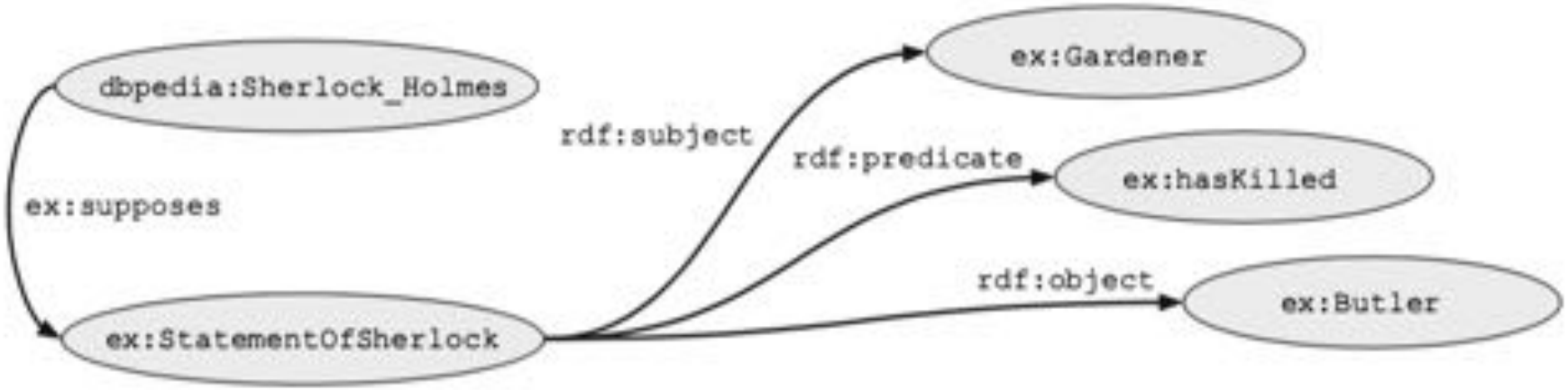
RDF Reification

- Sherlock Holmes supposes that the gardener has killed the butler



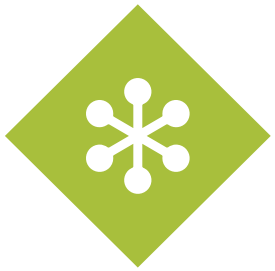
@prefix	**rdf: <http://www.w3.org/1999/02/22-rdf-syntax-** **.**
@prefix	dbpedia <http://dbpedia.org/resource/> .
@prefix	ex: <http://example.org/Crimestories#> .

RDF Reification




```
@prefix dbpedia: <http://dbpedia.org/resource/> .
@prefix rdf:     <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix ex:      <http://example.org/Crimestories#> .

dbpedia:SherlockHolmes ex:supposes ex:StatementOfSherlock .
ex:StatementOfSherlock a rdf:Statement ;
rdf:subject ex:Gardener ;
rdf:predicate ex:hasKilled ;
rdf:object ex:Butler .
```



RDF Reification

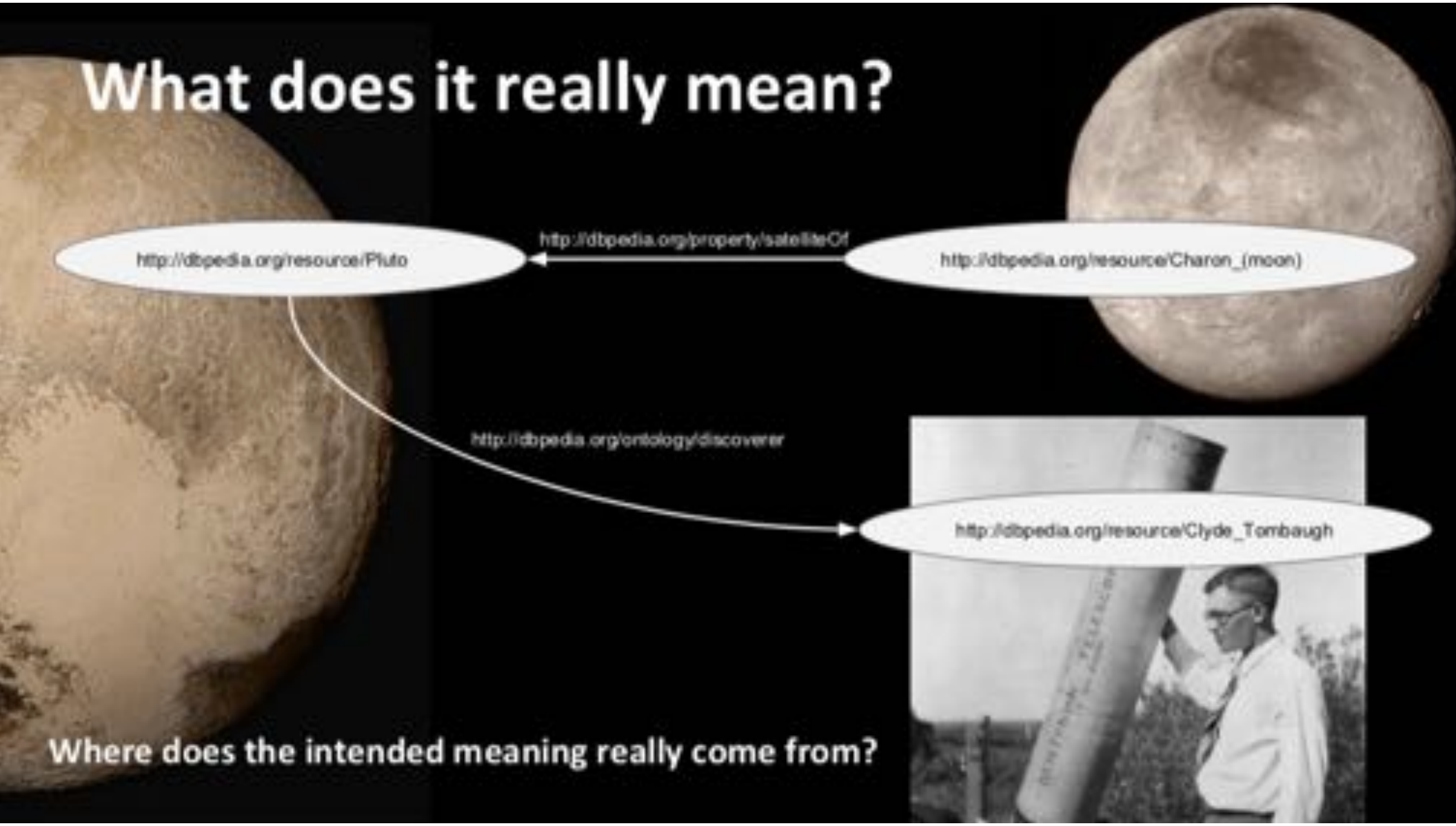
- What is the use of reification?
  - modeling data provenance
  - formalizing statements about reliability and trust
  - define metadata about statements

- But… you should be careful….
  - with reification relations can be transformed into classes/instances (type conflicts)
  - definition of infinite recursions and cycles

RDFS

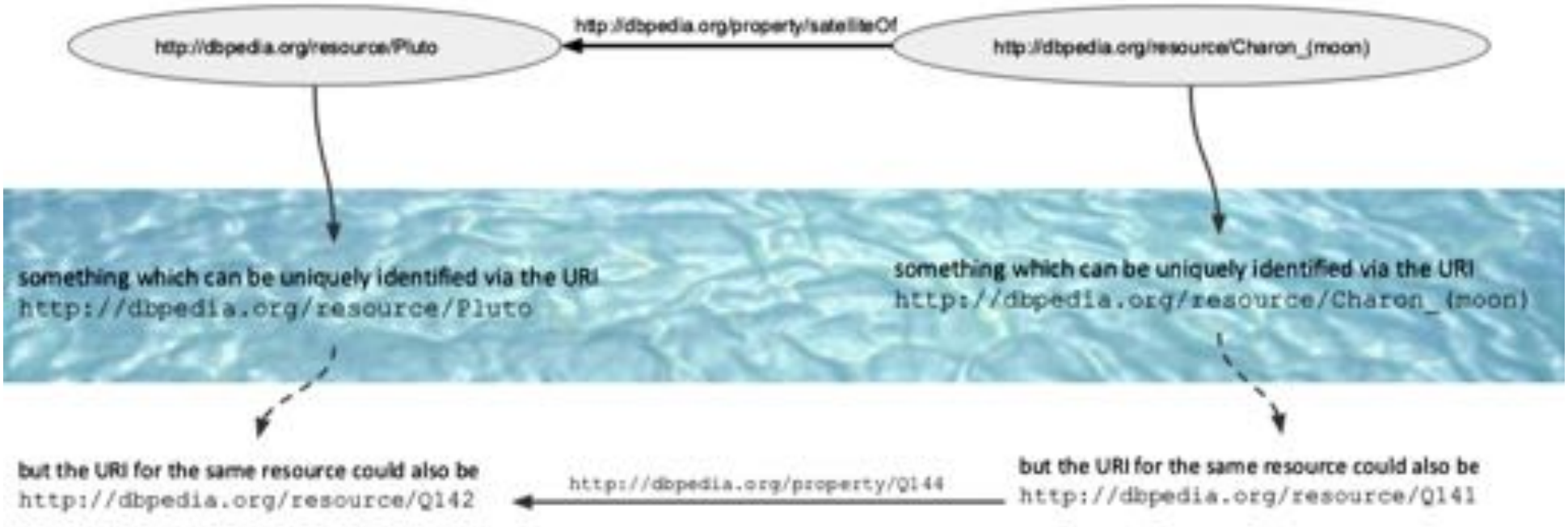
Enhancing the expressivity of RDF

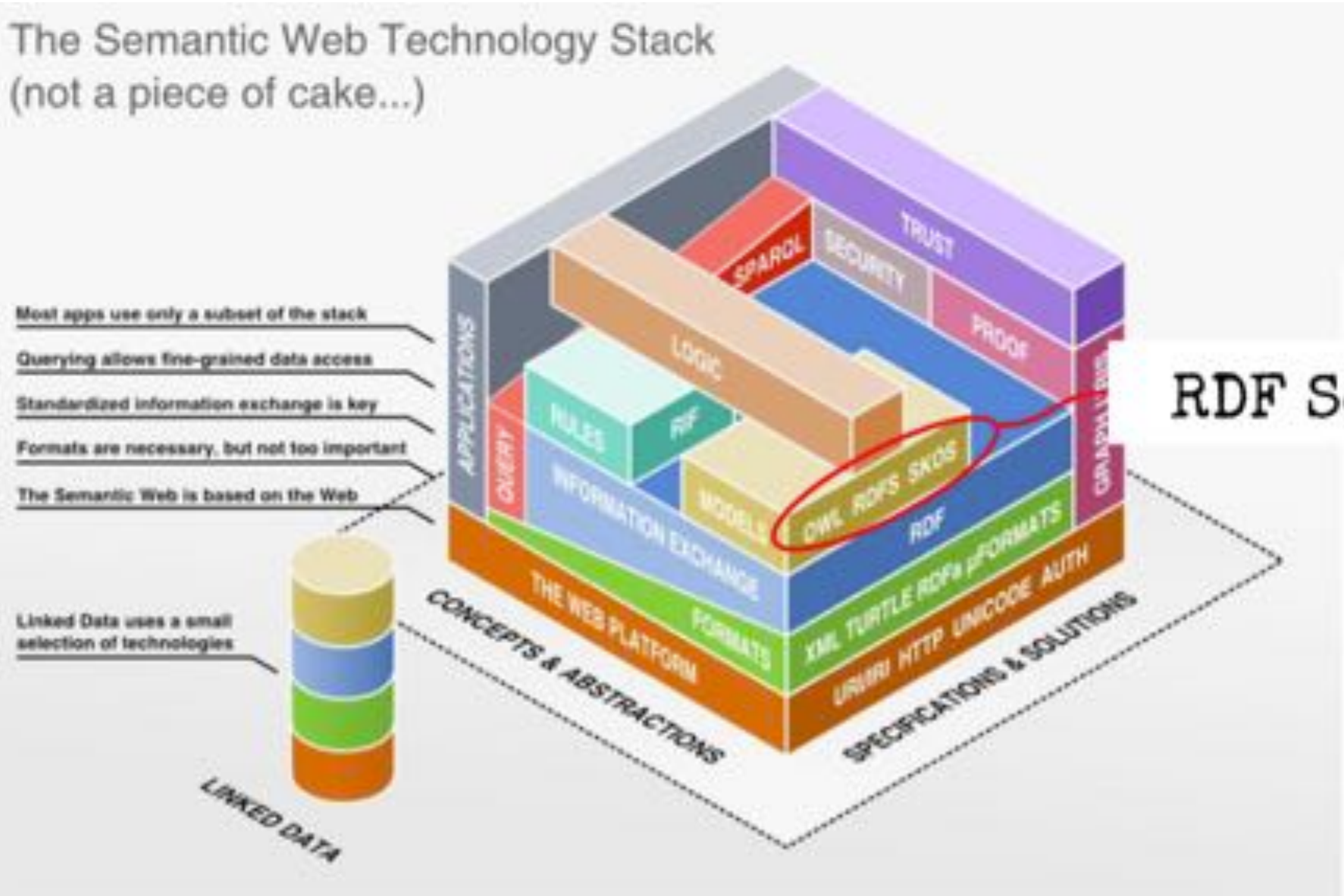
What does it really mean?



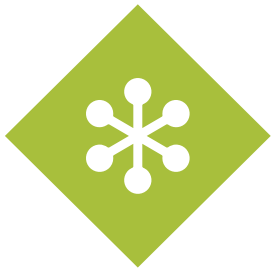
Where does the intended meaning really come from?

What Does It Really Mean?




The Semantic Web Technology Stack
(not a piece of cake...)
Most apps use only a subset of the stack
Querying allows fine-grained data access
Standardized information exchange is key
Formats are necessary, but not too important
The Semantic Web is based on the Web
Linked Data uses a small selection of technologies
LINKED DATA
APPLICATIONS
QUERY
SPARQL
TRUST
SECURITY
PROOF
LOGIC
RULES
RIF
INFORMATION EXCHANGE
MODELS
OWL RDFS SKOS
RDF
CRYPTOGRAPHY
THE WEB PLATFORM
FORMATS
XML TURTLE RDFa μFORMATS
URI/IRI HTTP UNICODE AUTH
CONCEPTS & ABSTRACTIONS
SPECIFICATIONS & SOLUTIONS

RDF Schema

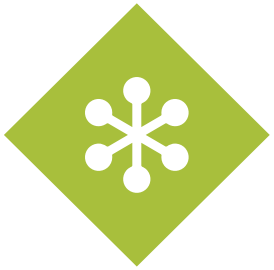


RDF Schema

- RDF Schema, officially called “RDF Vocabulary Description Language”

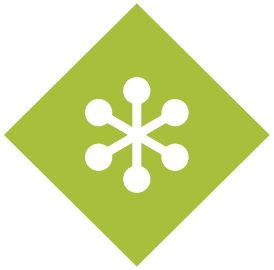
- RDF Schema allows:
  - Definition of classes via rdfs:Class
  - Class instantiation in RDF via rdf:type

- Example:
  - :Planet rdf:type rdfs:Class .
  - :Earth rdf:type :Planet .



RDF Schema

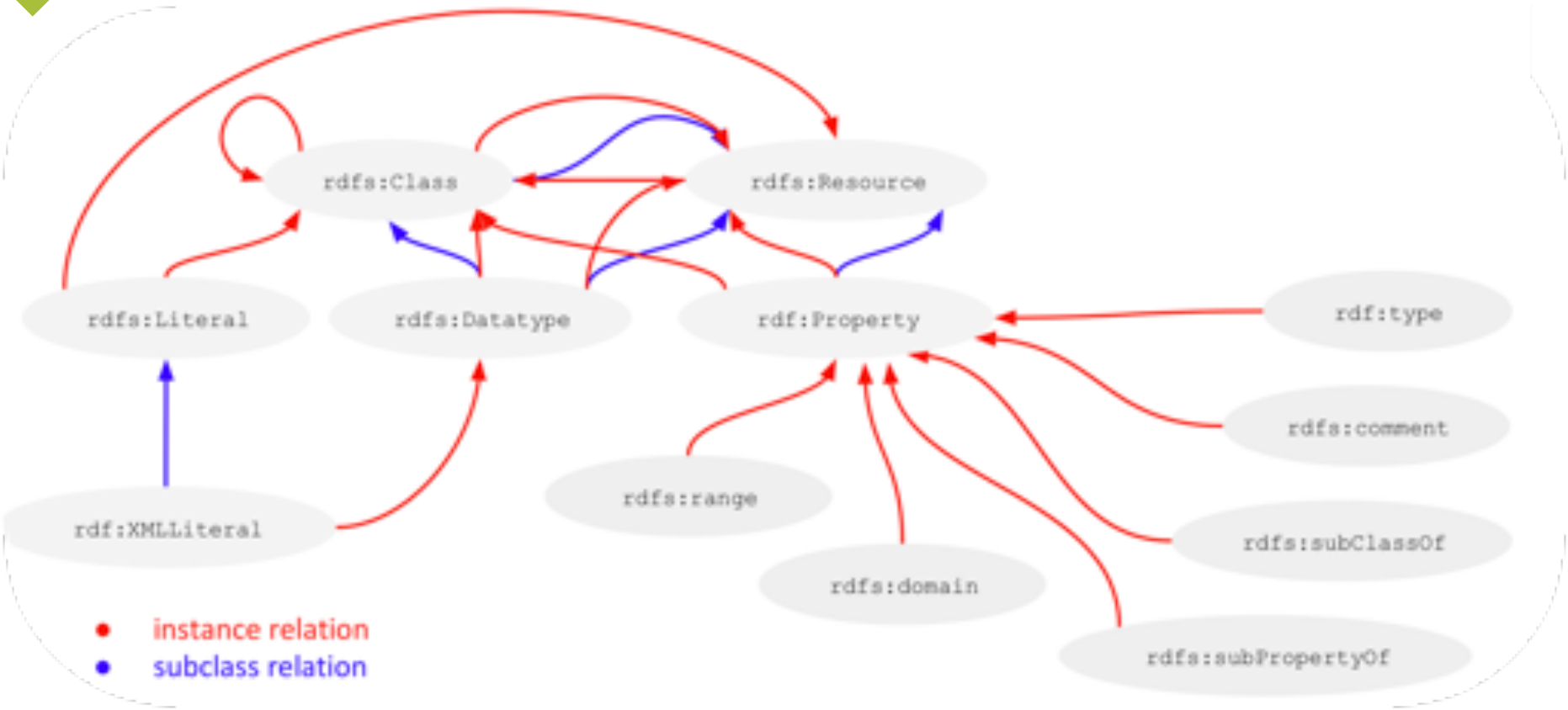
- Definition of properties via rdf:Property
- Definition of property restrictions on domain and range via rdfs:domain and rdfs:range
- Example
  - :CelestialBody rdf:type rdfs:Class .
  - :satelliteOf rdf:type rdf:Property .
  - :satelliteOf rdfs:domain :CelestialBody .
  - :satelliteOf rdfs:range :CelestialBody .



RDF Schema

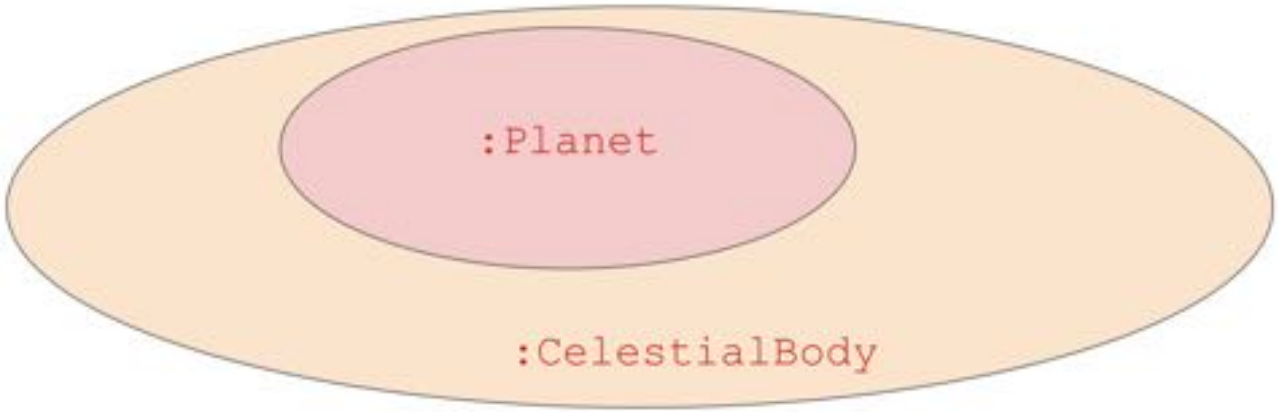
- Everything in the RDF model is a resource
  - rdfs:Class rdf:type rdfs:Resource .
  - rdf:Property rdf:type refs:resource .
  - rdfs:Literal rdf:type rdfs:Resource .
  - rdfs:XMLLiteral rdf:type rdfs:Resource .
  - rdfs:Datatype rdf:type rdfs:Resource .
  - rdfs:Container rdf:type rdfs:Resource .
  - rdfs:ContainerMembershipProperty rdf:type rdfs:Resource .

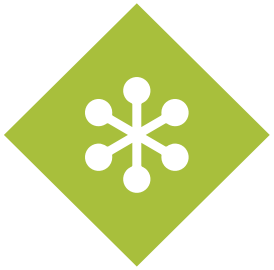
RDF Schema


rdfs:Class
rdfs:Resource
rdfs:Literal
rdfs:Datatype
rdf:Property
rdf:type
rdfs:comment
rdfs:range
rdf:XMLLiteral
rdfs:subClassOf
rdfs:domain
rdfs:subPropertyOf
instance relation
subclass relation

Definition of Hierarchical Relationships

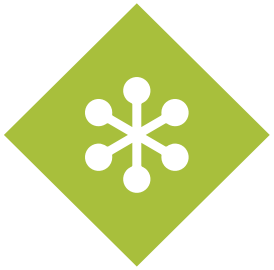
- Subclasses and superclasses via rdfs:subClassOf
  - Example:
    - *:Planet rdfs:subClassOf :CelestialBody .*





Definition of Hierarchical Relationships

- Definition of hierarchical relationships:
  - Subclasses and superclasses via rdfs:subClassOf
  - Example:
    - *:Planet rdfs:subClassOf :CelestialBody .*
  - Subproperties and superproperties via subPropertyOf
  - Example
    - *:artificialSatelliteOf rdfs:subPropertyOf :satelliteOf .*



RDF Schema

- Some more properties:
  - **rdfs:seeAlso**
    defines a relation of a resource to another, which explains it
  - **rdfs:isDefinedBy**
    subproperty of rdfs:seeAlso, defines the relation of a resource to its definition
  - **rdfs:comment** comment, usually as text
  - **rdfs:label**
    „readable“ name of a resource (contrary to ID)
  - **rdfs:member**
    super-property of all the container membership properties (e.g. rdf:_1, ...)

```
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix rdf:  <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix :     <http://example.org/Space#> .

:Planet rdf:type         rdfs:Class ;
        rdfs:subClassOf :CelestialBody .
:Satellite rdf:type         rdfs:Class ;
           rdfs:subClassOf :CelestialBody .
:ArtificialSatellite rdf:type         rdfs:Class ;
                     rdfs:subClassOf :Satellite .

:satelliteOf   rdf:type    rdf:Property ;
               rdfs:domain :CelestialBody .
               rdfs:range  :CelestialBody .

:Earth    rdf:type      :Planet .
:Moon     rdf:type      :Satellite ;
          :satelliteOf :Earth .
:Sputnik1 rdf:type      :ArtificialSatellite ;
          :satelliteOf :Earth ;
          rdfs:label    "Sputnik 1"@en ;
          rdfs:comment "the first artificial Earth satellite in 1957" .
```

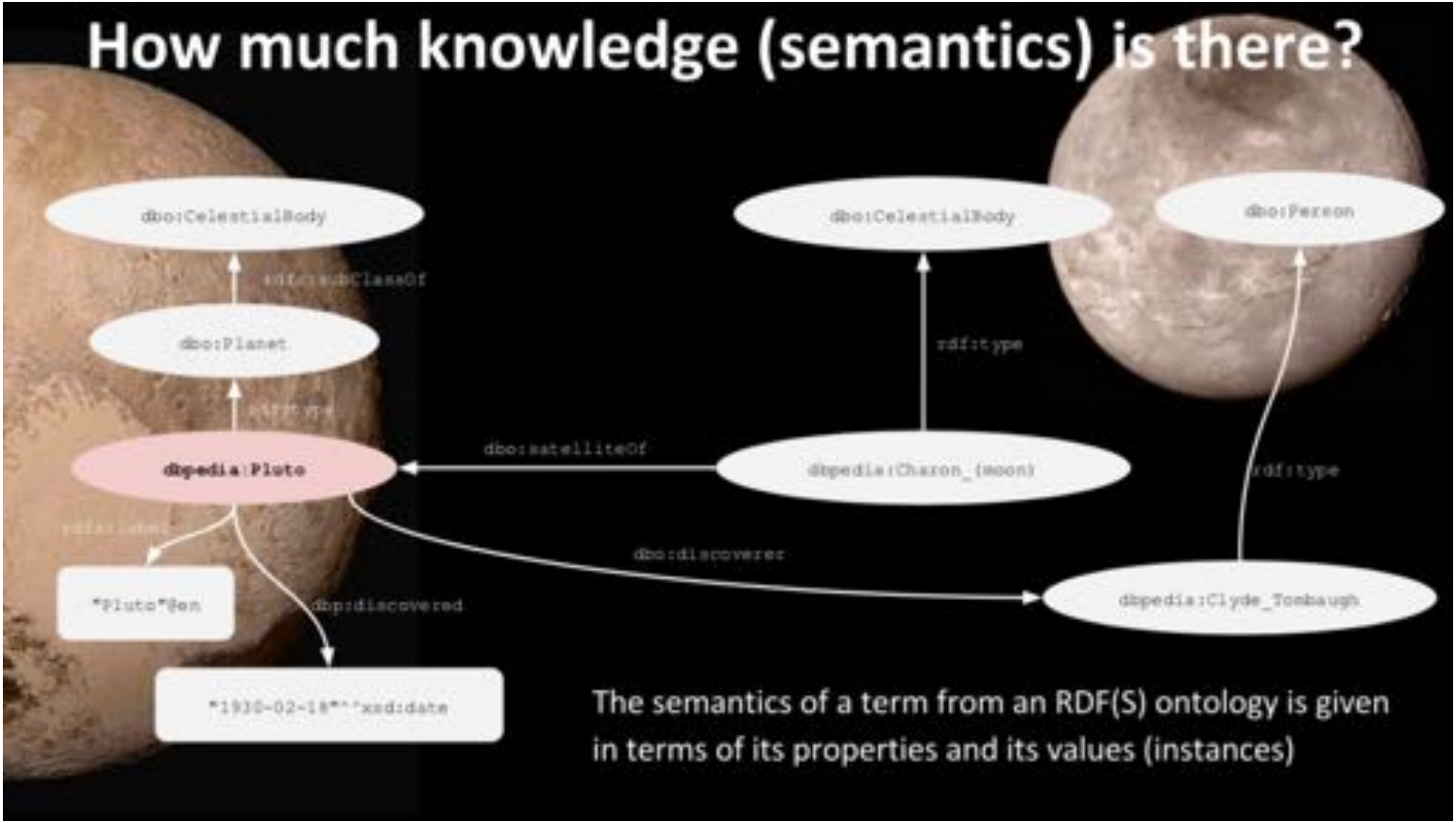
Class Definiti

Property Defi

Instance Defi

Logical Inference With RDFS

How much knowledge (semantics) is there?



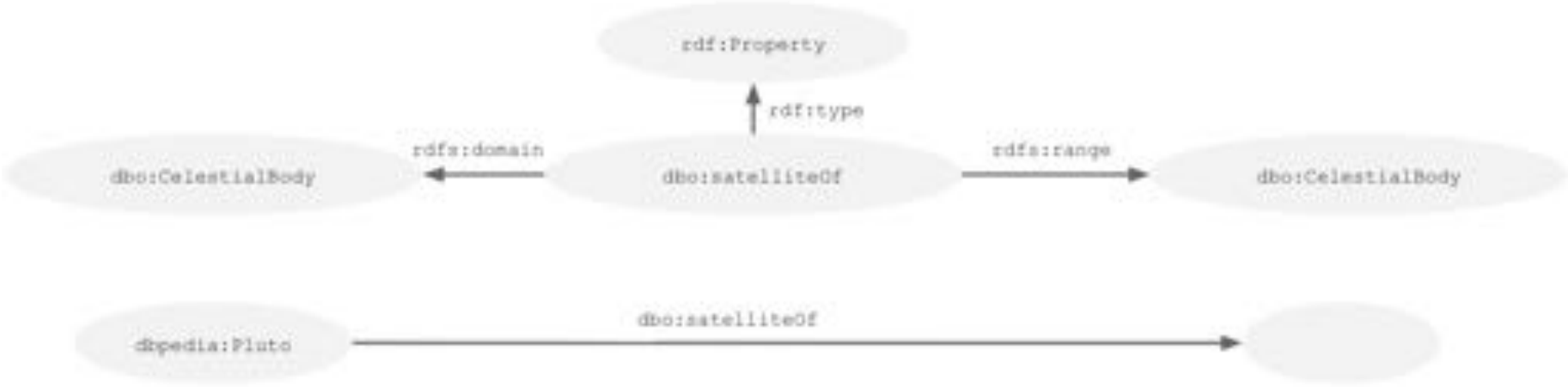
The semantics of a term from an RDF(S) ontology is given in terms of its properties and its values (instances)

RDF(S) Semantics

- In difference to other data definition languages, RDF(S) is based on **formal semantics**
- Formal semantics enables RDF(S) to draw **valid** and **sound logical inferences**

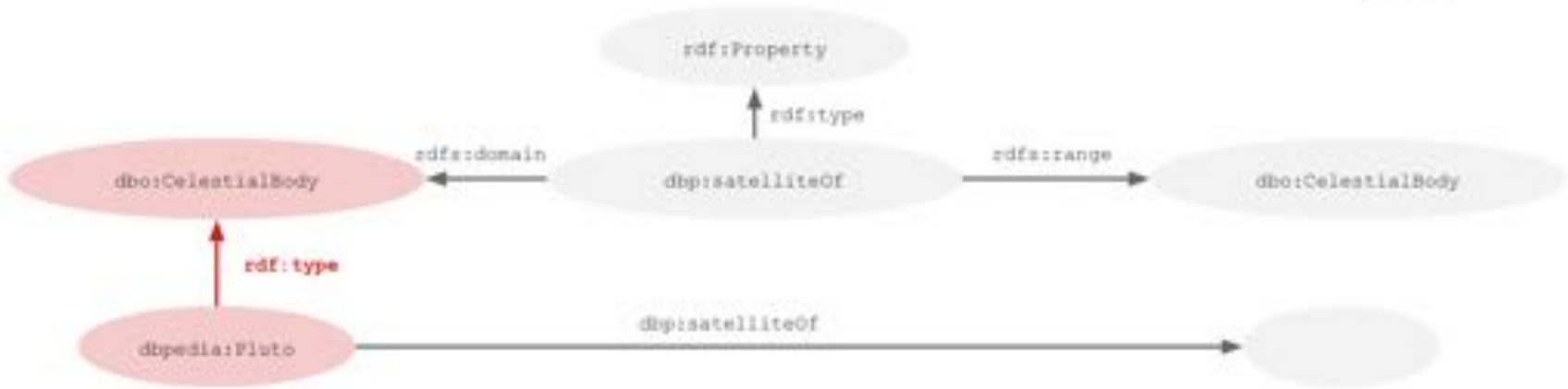
- Examples:
  - dbpedia:Pluto rdf:type dbo:Planet
  - dbo:Planet rdfs:subClassOf dbo:CelestialBody
  - dbo:artificialSatelliteOf rdfs:subPropertyOf dbo:satelliteOf

Which Conclusions Can We Deduce With RDF(S)?



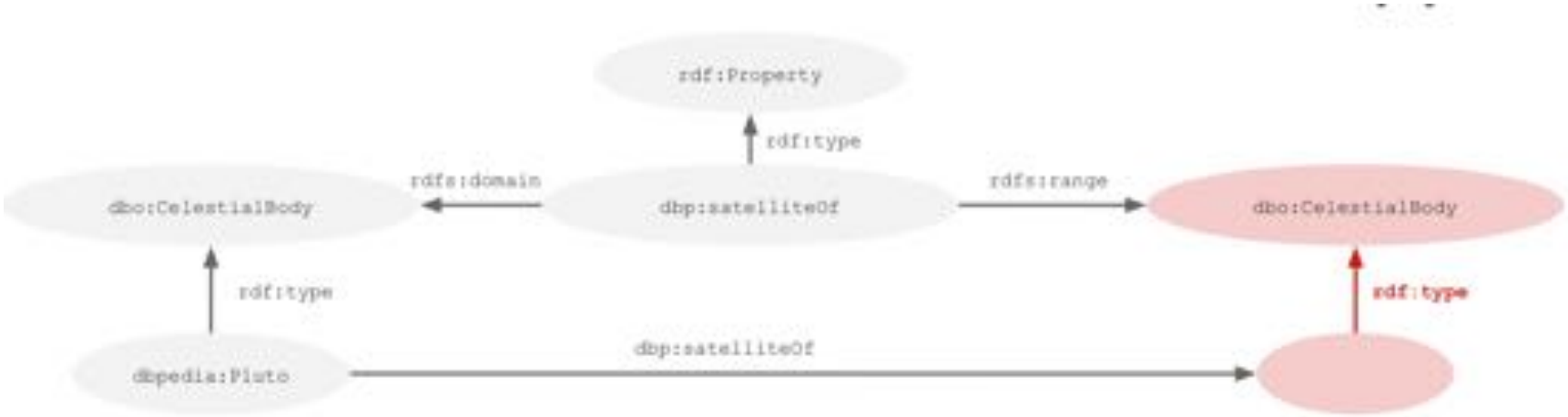
Which Conclusions Can We Deduce With RDF(S)?

Which Conclusions Can We Deduce With RDF(S)?



- Deduction of entity **class membership** from **domain** of one of its properties

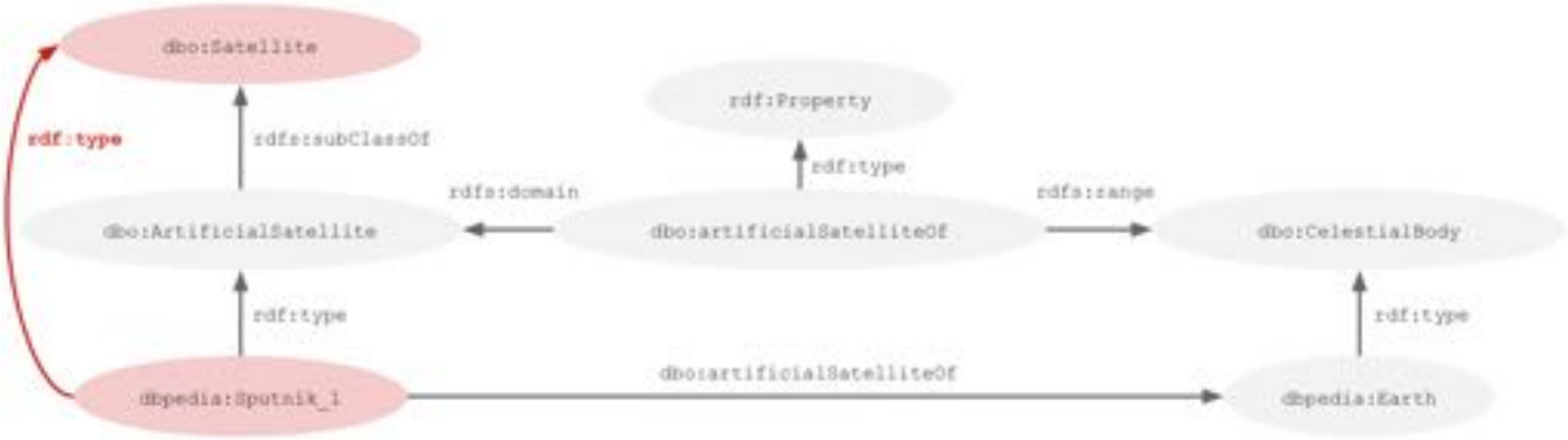
Which Conclusions Can We Deduce With RDF(S)?



- Deduction of entity **class membership** from the **range** of one of its properties

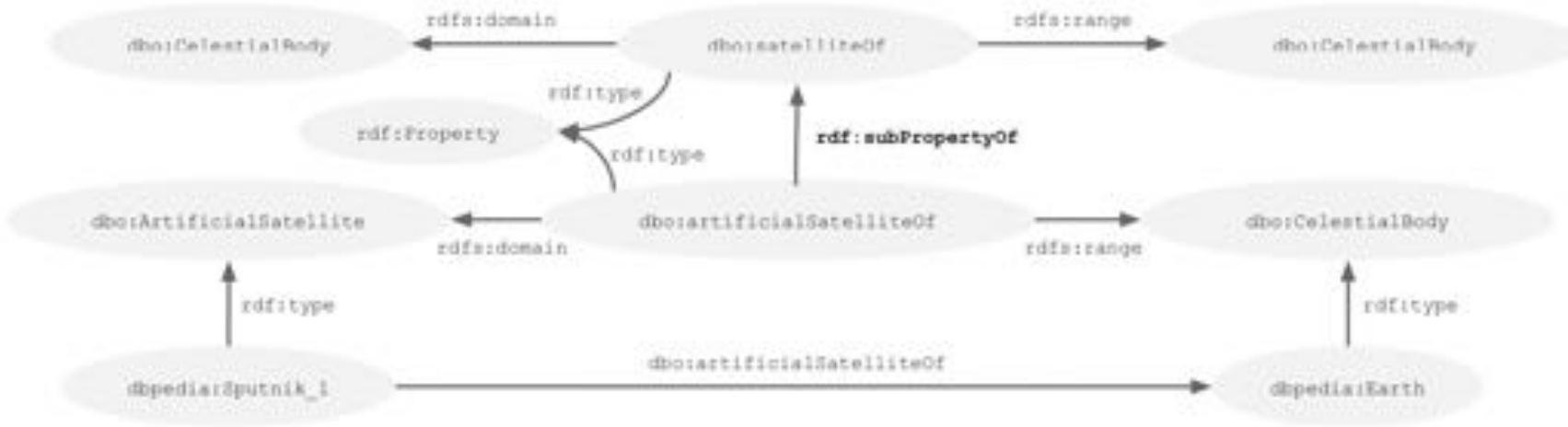
Which Conclusions Can We Deduce With RDF(S)?

Which Conclusions Can We Deduce With RDF(S)?

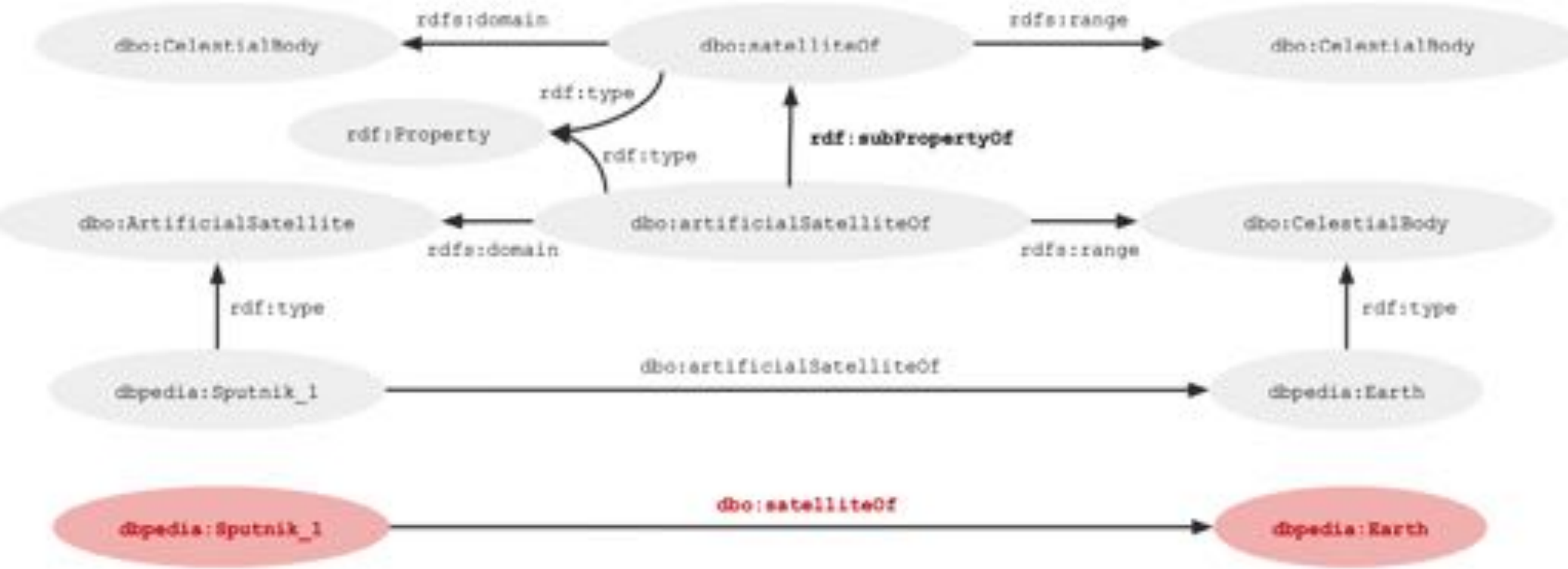


- Deduction of entity **superclass membership** from a **class hierarchy.**

Which Conclusions Can We Deduce With RDF(S)?



Which Conclusions Can We Deduce With RDF(S)?



- Deduction of entity **new facts** from **subproperty** relationships**.**