Knowledge Representation & Reasoning
&
Introduction To Knowledge Graphs

Week 3 & 4 | Fall 2022

Dr. Amna Basharat

What Is the Semantic Web?

What has it got to do with Knowledge Graphs?
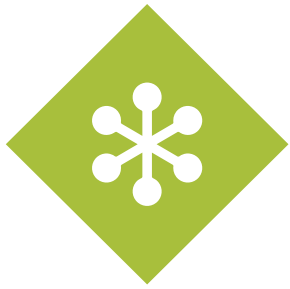
What Is Semantic Metadata?

What is the difference between explicit & implicit knowledge?
What are 3 main benefits of semantic web?

THE SEMANTIC WEB

A new form of Web content
that is meaningful to computers
will unleash a revolution of new possibilities

by
TIM BERNERS-LEE,
JAMES HENDLER and
ORA LASSILA

Simple Take Away

- If we use standard languages and semantic markup for our data on the web

  - we allow for easier discovery and reuse of data, even automatically by machines/agents

  - we can build "smarter applications"

- Semantic Web technologies are not ONLY for the open Web

In the (traditional) Web there is no explicit semantics
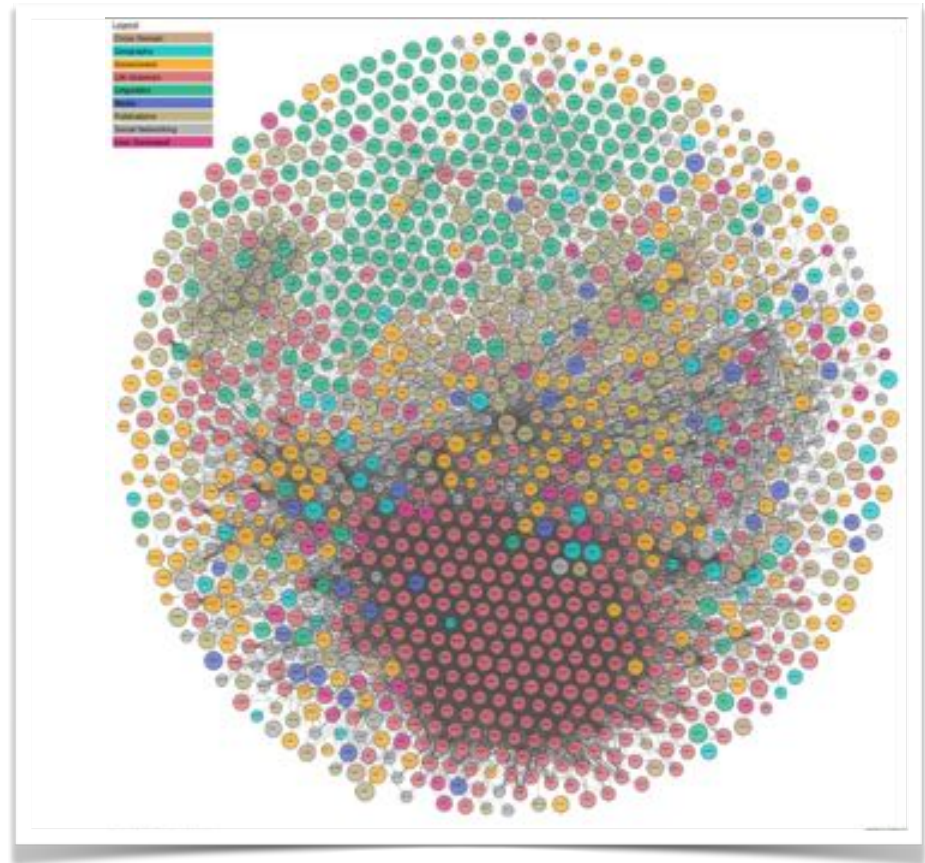
🍀 No (Explicit) Semantics!

From WWW To Web of Data

- Precondition:
  - Content can be **read and interpreted correctly** (= "understood") by machines

Natural Language Processing

- Technologies of **traditional Information Retrieval**

- Statistical models & machine learning

Semantic Web Technologies

- Natural language web content is **explicitly annotated with semantic metadata**

- Semantic metadata encode the **meaning** of the content and can be **read and interpreted correctly by machines**
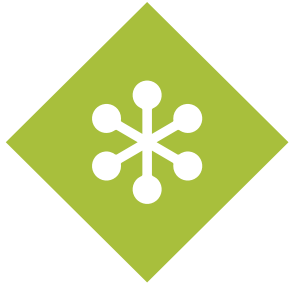
Understanding Content on the Web

Understanding Content on the Web

Text: **"Pluto"**

Entity Mapping
Disambiguation

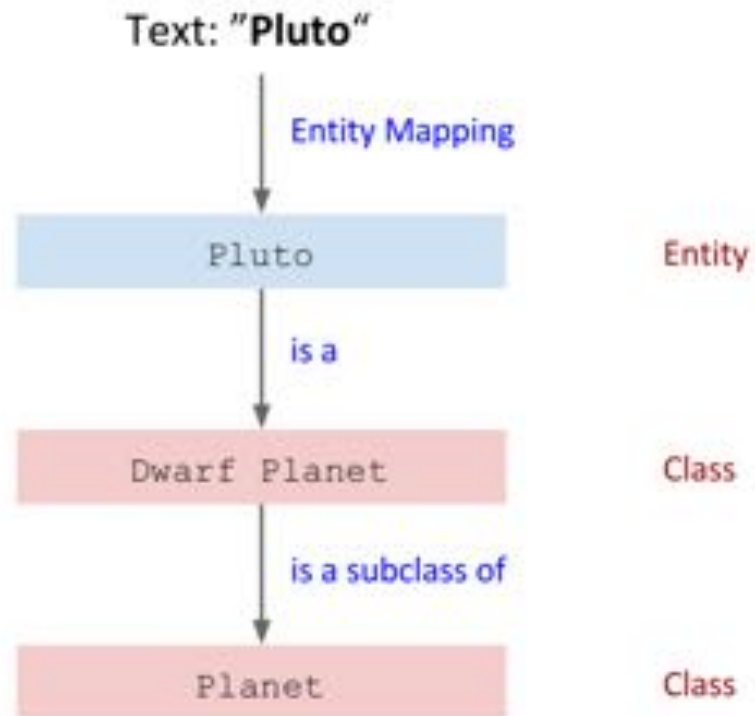|	
Pluto	a Disney cartoon character
Pluto	a Roman god
Pluto	a song by Björk
HMS Pluto	a ship
...	
Pluto	a dwarf planet

Disambiguation
- solution of linguistic ambiguities

Understanding Content on the Web

- The **Meaning (Semantics)** of entities and classes must be defined explicitly.

Text: "**Pluto**"

Entity Mapping

Pluto — Entity

is a

Dwarf Planet — Class

is a subclass of

Planet — Class

Understanding Content on the Web

The Meaning (Semantics) is expressed with the help of knowledge representations (**Ontologies**)
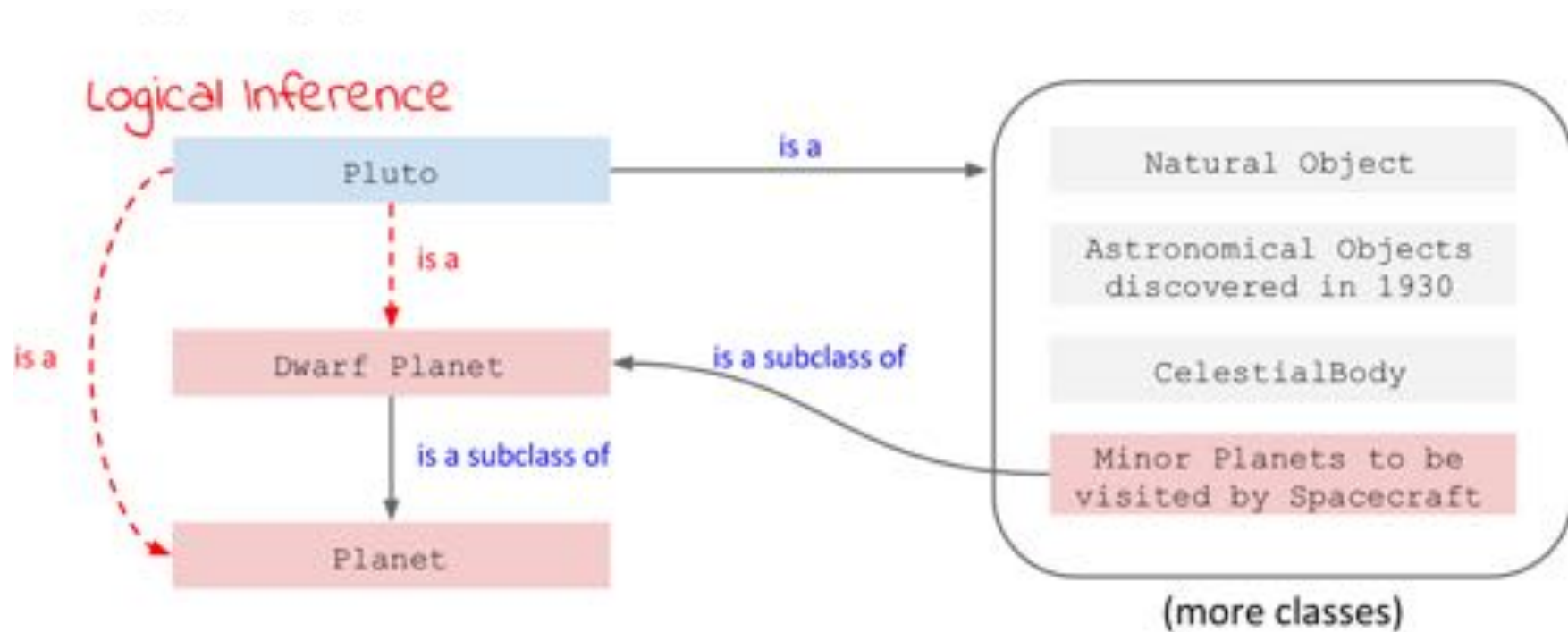
The Semantic Web - A Web of Data

- The meaning of information (Semantics) is made explicit by **formal (structured) and standardized knowledge representations (Ontologies).**

- Thus it will be possible,

  - To process the meaning of information automatically

  - To relate and integrate heterogeneous data

  - To deduce implicit (not evident) information from existing (evident) information in an automated way.

- The Semantic Web is kind of a **global database** that contains a **universal network of semantic propositions.**
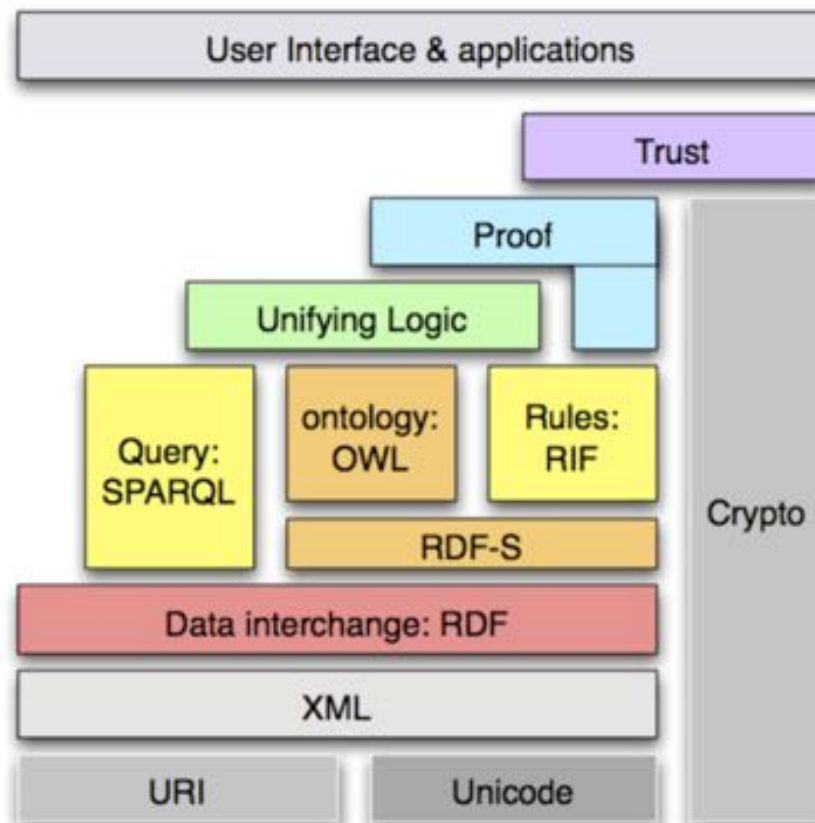
The Semantic Web - A Web of Data

„*The Semantic Web is an extension of the current web in which information is given well-defined meaning, better enabling computers and people to work in cooperation*"

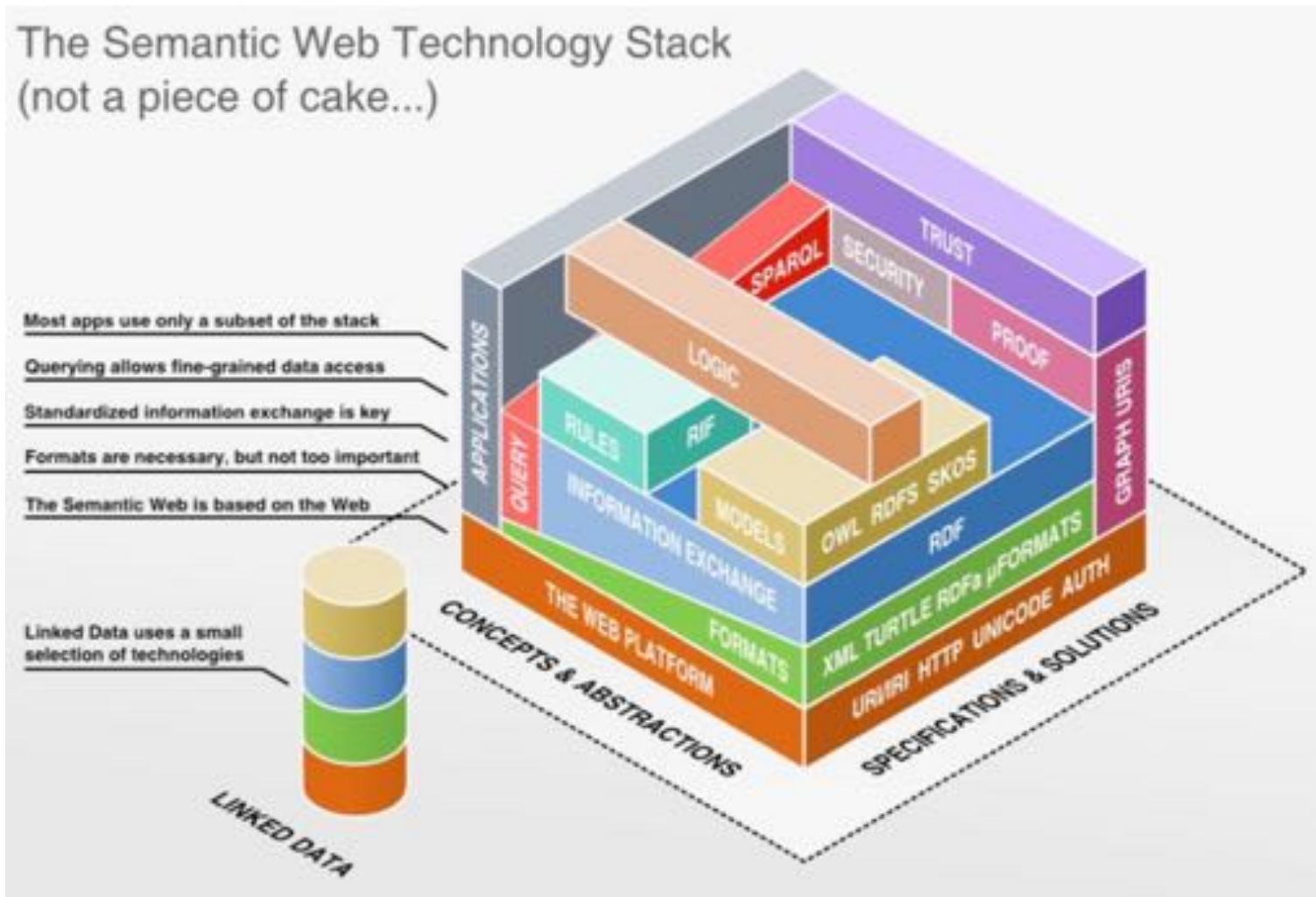Tim Berners-Lee, James Hendler, Ora Lassila: *The Semantic Web*, Scientific American, 284(5), pp. 34-43(2001)

The Semantic Web Layers

The Semantic Web Technology Stack
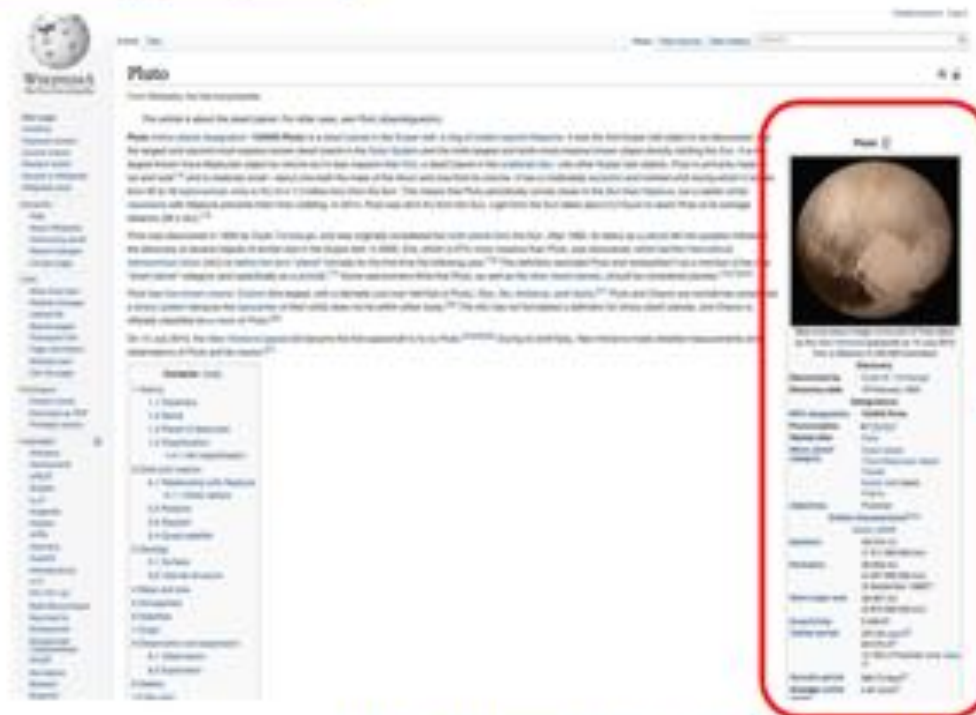

The Semantic Web Technology Stack
(not a piece of cake...)

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

APPLICATIONS
QUERY
RULES
RIF
LOGIC
SPARQL
SECURITY
TRUST
PROOF
GRAPH URIS
INFORMATION EXCHANGE
MODELS
OWL RDFS SKOS
RDF
THE WEB PLATFORM
FORMATS
XML TURTLE RDFa µFORMATS
URI/IRI HTTP UNICODE AUTH

CONCEPTS & ABSTRACTIONS
SPECIFICATIONS & SOLUTIONS
LINKED DATA

The Semantic Web Technology Stack

URI - Uniform Resource Identifier

Pluto ┈┈┈┈┈┈┈► http://dbpedia.org/resource/Pluto

From Wikipedia to DBpedia

http://en.wikipedia.org/wiki/Pluto

http://dbpedia.org/resource/Pluto

From Wikipedia to DBpedia

http://dbpedia.org/page/Pluto

http://dbpedia.org/resource/Pluto

```
:Pluto rdf:type dbo:Planet .
:Pluto foaf:name "Pluto"@en .
:Pluto dbo:discoverer :Clyde_Tombaugh .
:Pluto dbo:discovered "1930-02-18"^^xsd:date .
:Clyde_Tombaugh rdf:type dbo:Person .
:Clyde_Tombaugh dbo:birthdate "1906-02-04" ^^xsd:date
.
...
```

RDF Resource Description Framework

:Pluto	rdf:type	dbo:Planet	.
RDF Subject	*RDF Property*	*RDF Object*	

The Semantic Web Technology Stack

http://dbpedia.org/ontology/Planet
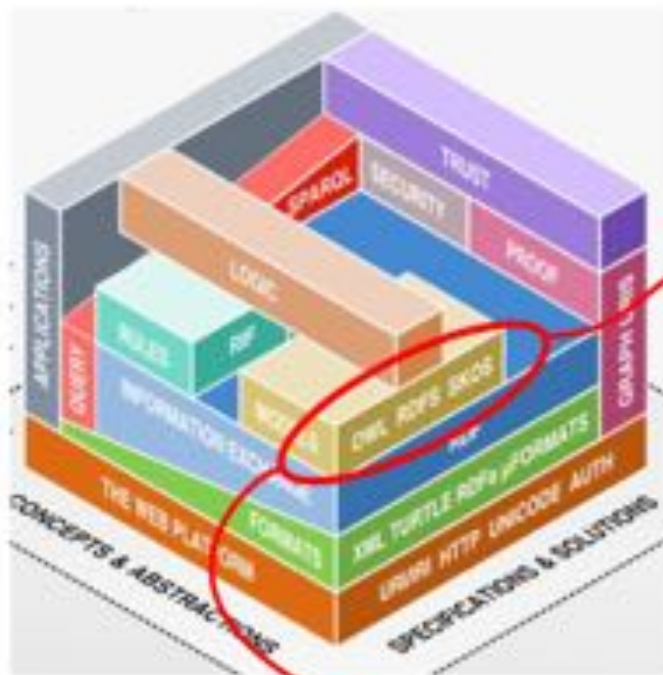
```
dbo:Planet rdf:type owl:class .
dbo:Planet rdfs:subClassOf dbo:CelestialBody .
dbo:discovered rdf:type rdf:Property .
dbo:discovered rdfs:domain owl:Thing .
dbo:discovered rdfs:range xsd:date .
dbo:discoverer rdf:type rdf: Property .
dbo:discoverer rdfs:domain owl:Thing .
dbo:discoverer rdfs:rang dbo:Person .
...
```
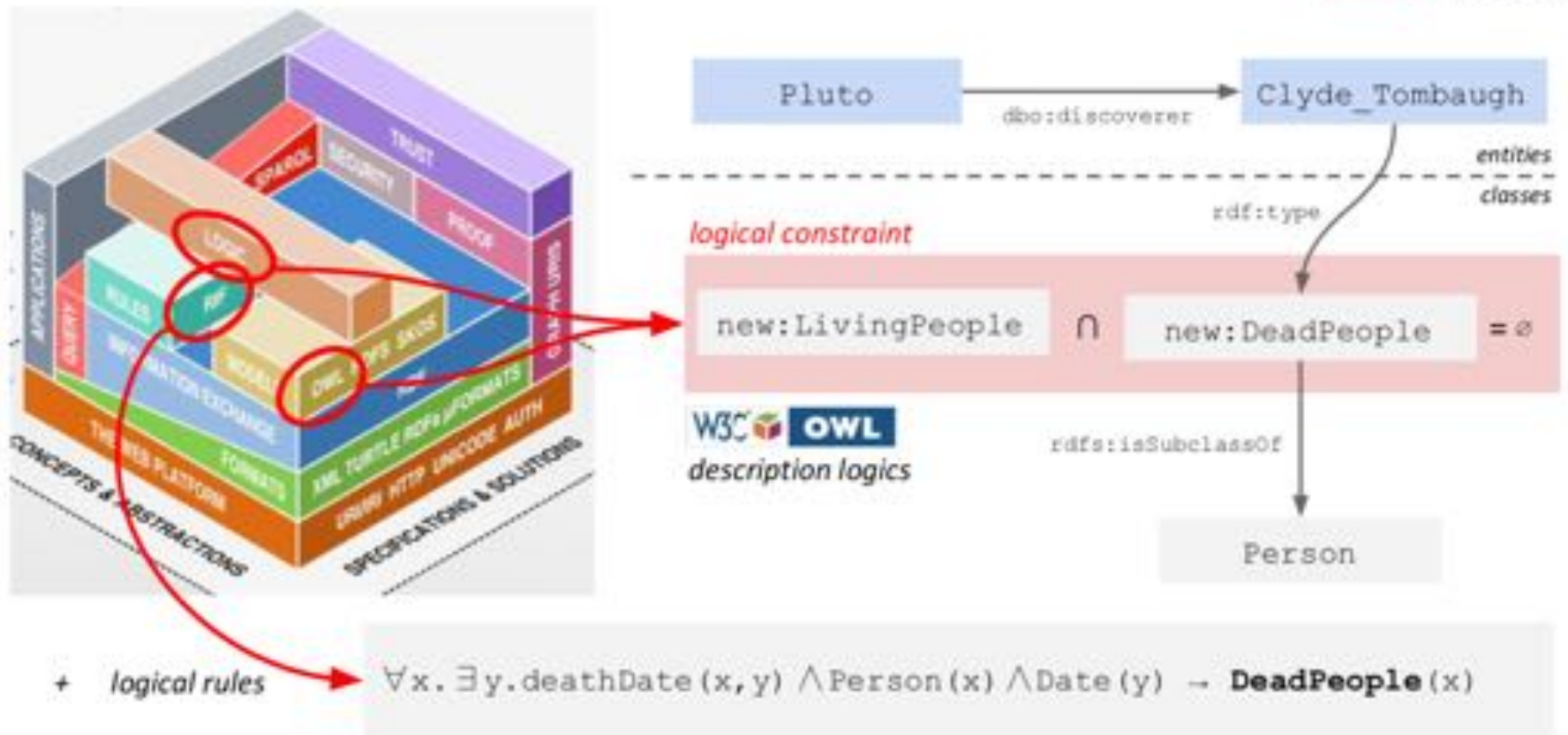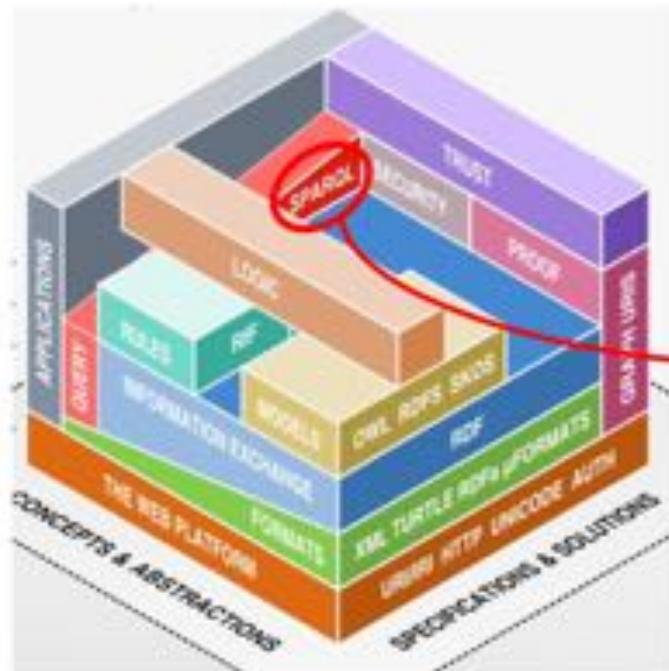
RDF Schema

The Semantic Web Technology Stack

The Semantic Web Technology Stack

Look for all **space missions in the Solar System** which have become a **satellite** of their target

```
PREFIX dcterms: <http://purl.org/dc/terms/>
PREFIX skos: <http://www.w3.org/2004/02/skos/core#>
PREFIX dbp: <http://dbpedia.org/property/>
PREFIX dbc: <http://dbpedia.org/resource/Category:>

SELECT distinct ?s ?o
FROM <http://dbpedia.org/>
WHERE{
?s dcterms:subject/skos:broader*
    dbc:Discovery_and_exploration_of_the_Solar_System ;
    dbp:satelliteOf ?o .
}
```

The Web of Data and How We Make Use of It

The Web of Data
- 9,960 datasets
- >85 billion facts
- >800 million links

(Aug. 2015)

http://stats.lod2.eu/
http://lod-cloud.net/

Traditional Web vs Web of Data

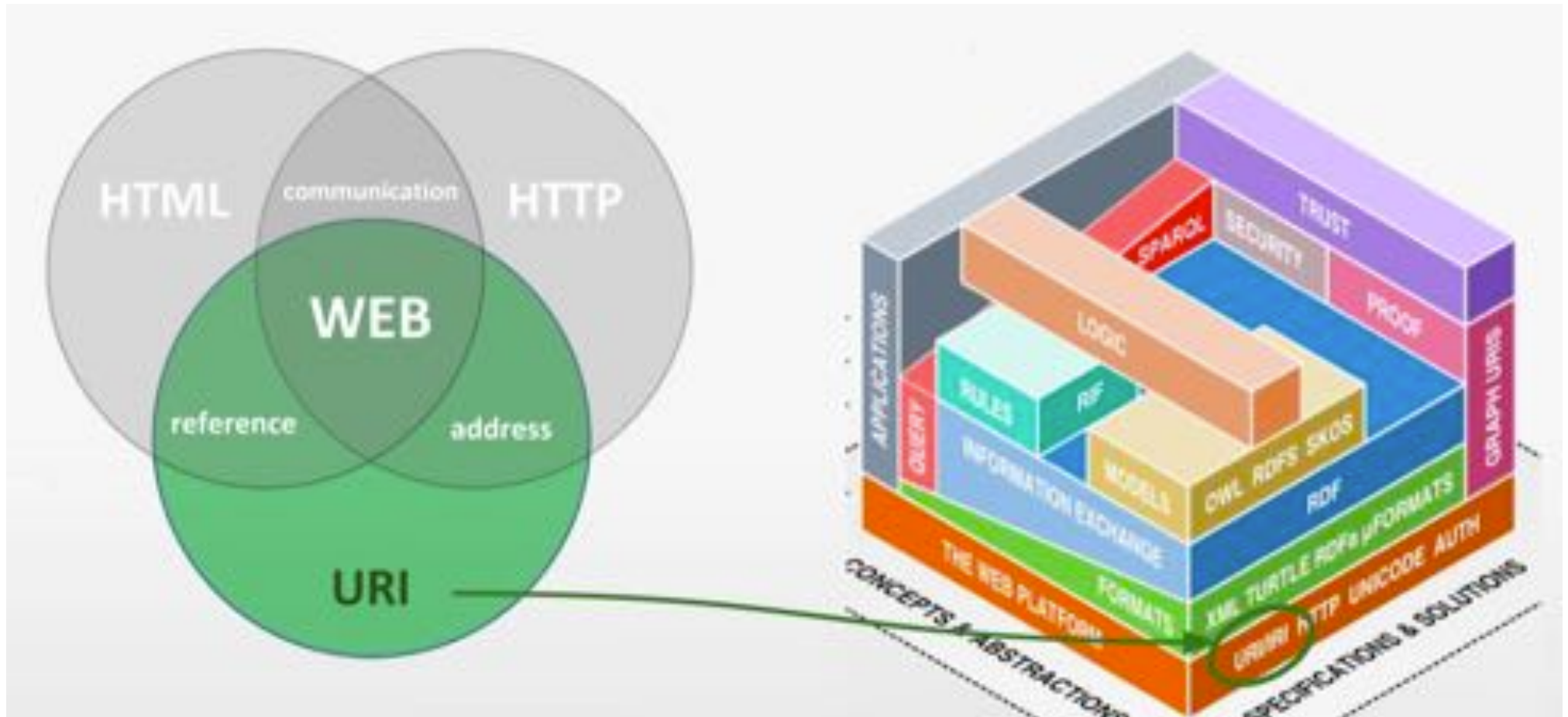# Basic Architecture of the Web

HTML · communication · HTTP · WEB · reference · address · URL
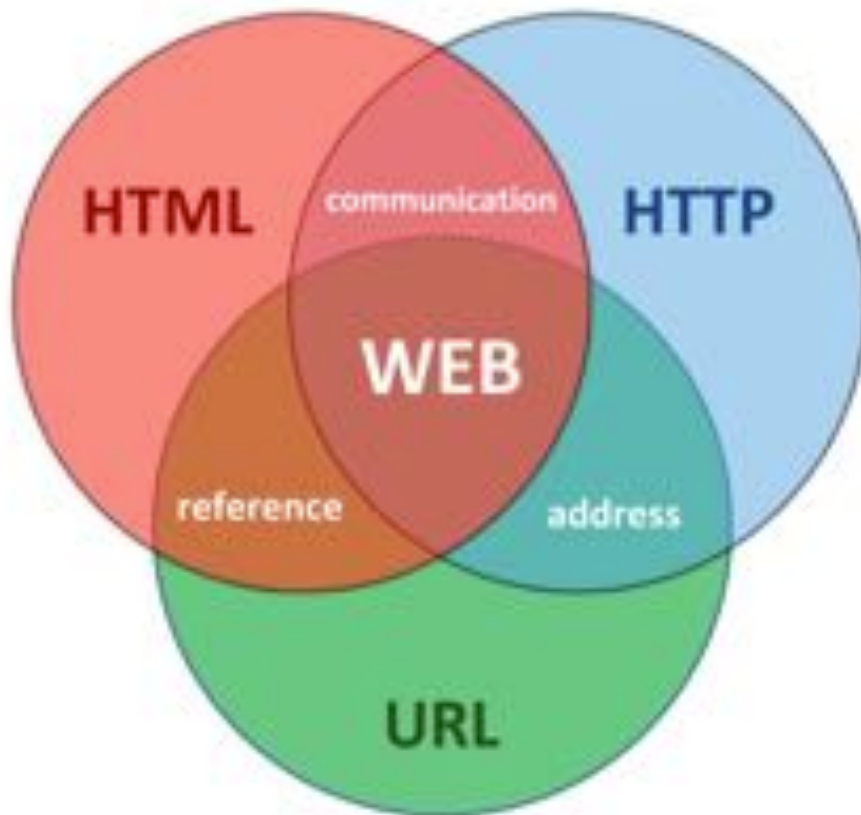
1. Identification (URI) & address (URL)
   e.g. http://fiz-karlsruhe.de

2. Communication / protocol (HTTP)
   GET /index HTTP/2
   Host: fiz-karlsruhe.de

3. Representation language (HTML)
   Mehwish works at
   `<a href="http://fiz-karlsruhe.de">`
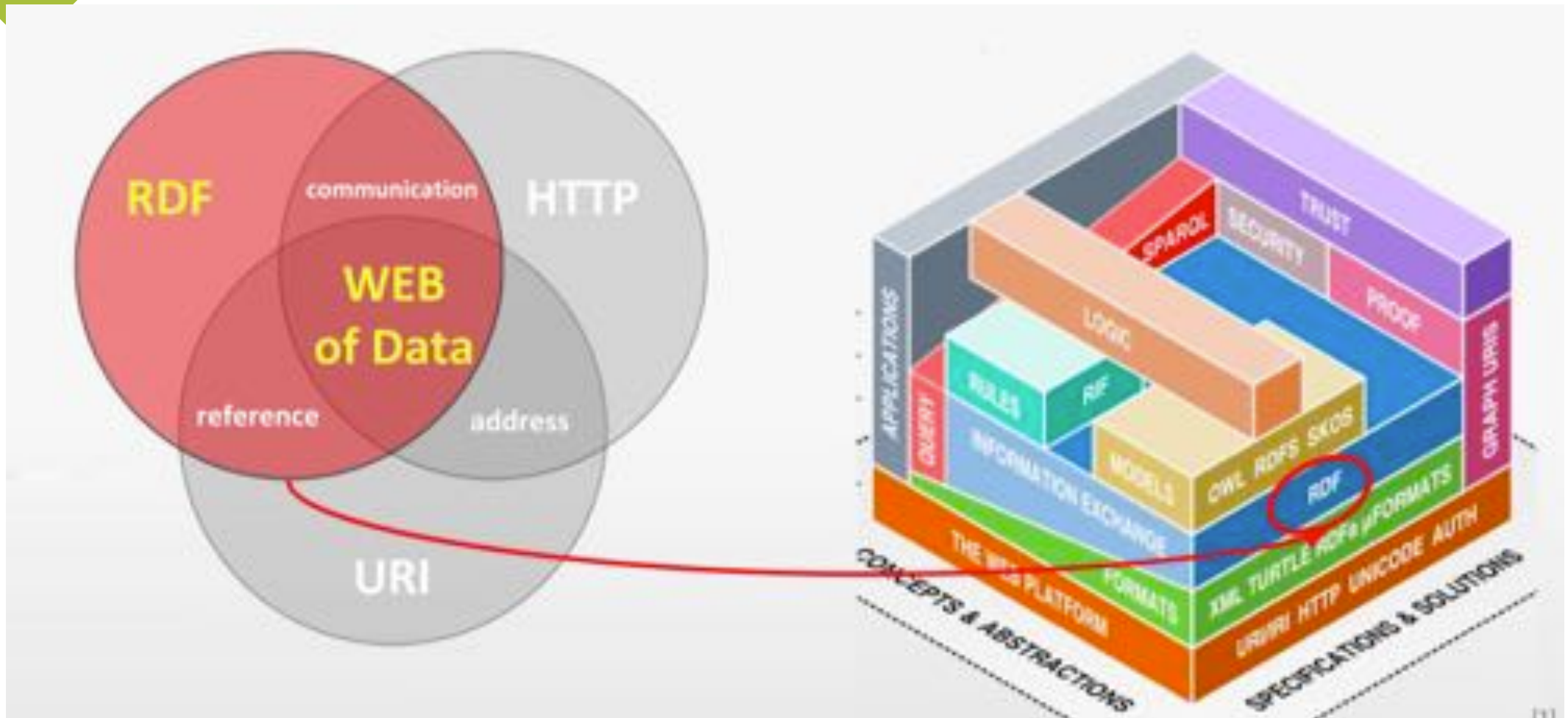   FIZ`</a>`.

Basic Architecture of the Web of Data

Basic Architecture of the Web of Data
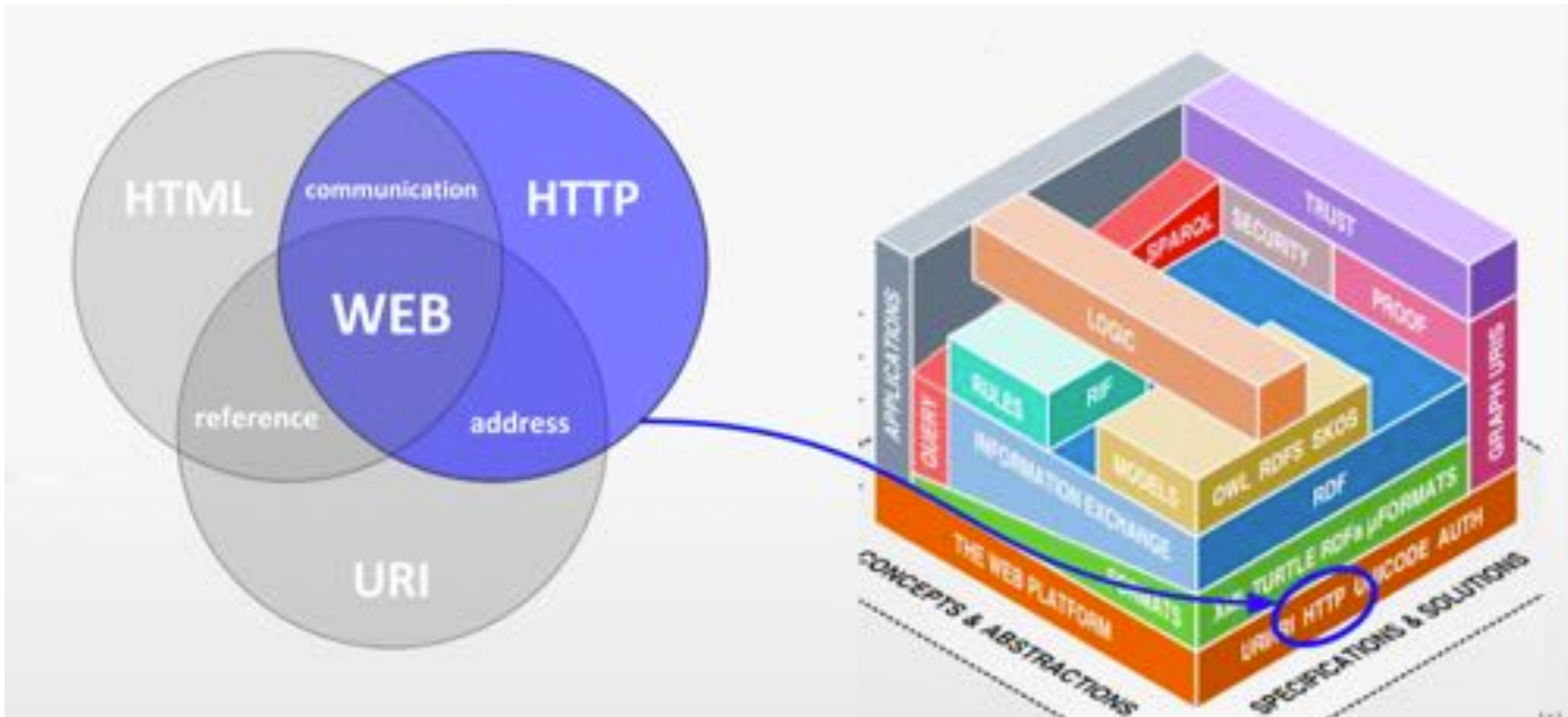
Basic Architecture of the Web of Data
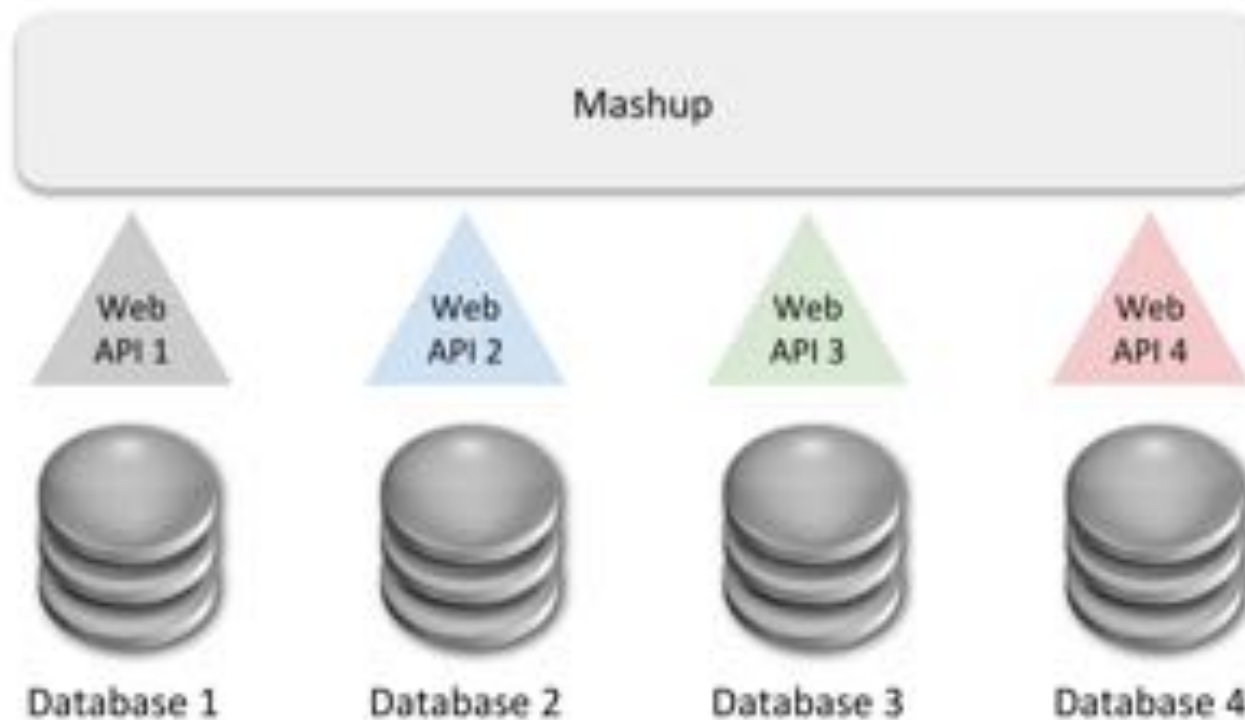
Data Access in Traditional Web

- Data can only be found on the Web, if it is available at some website.

Data Access in Traditional Web

- There is a number of different (proprietary) **Web APIs**, data exchange formats, and **Mashups** on top of that.
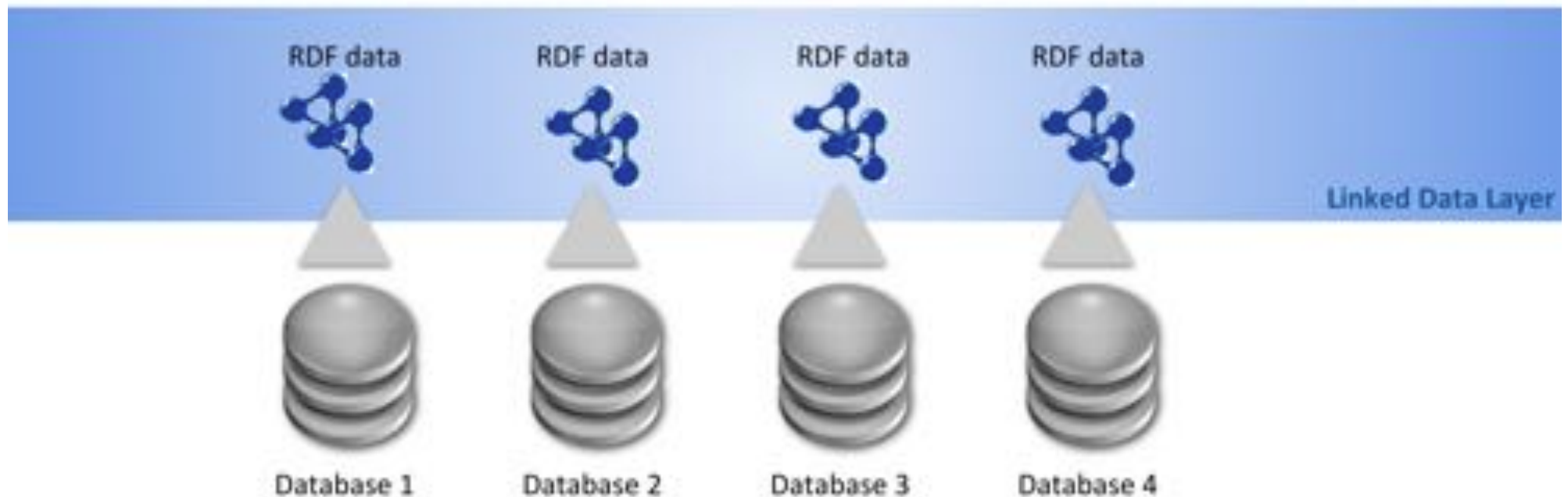
Data Silos in Traditional Web...

How To Get Rid of Closed Data Island?

- **Apply Linked Data technology**
    - to publish (structured) data on the Web
    - to draw connections from one data source to data from other data sources

The Semantic Web - A Web of Data

- The **Web of Data** is an upgrade of the traditional Web of Documents.
- It is the Web as a huge decentralised database (knowledge base) of machine-accessible data.

> „The web of **human-readable document** is being merged with a web of **machine understandable data**. The potential of the mixture of humans and machines working together and communication through the web could be immense."

Tim Berners-Lee, *The World Wide Web: A very short personal history*, May 1998
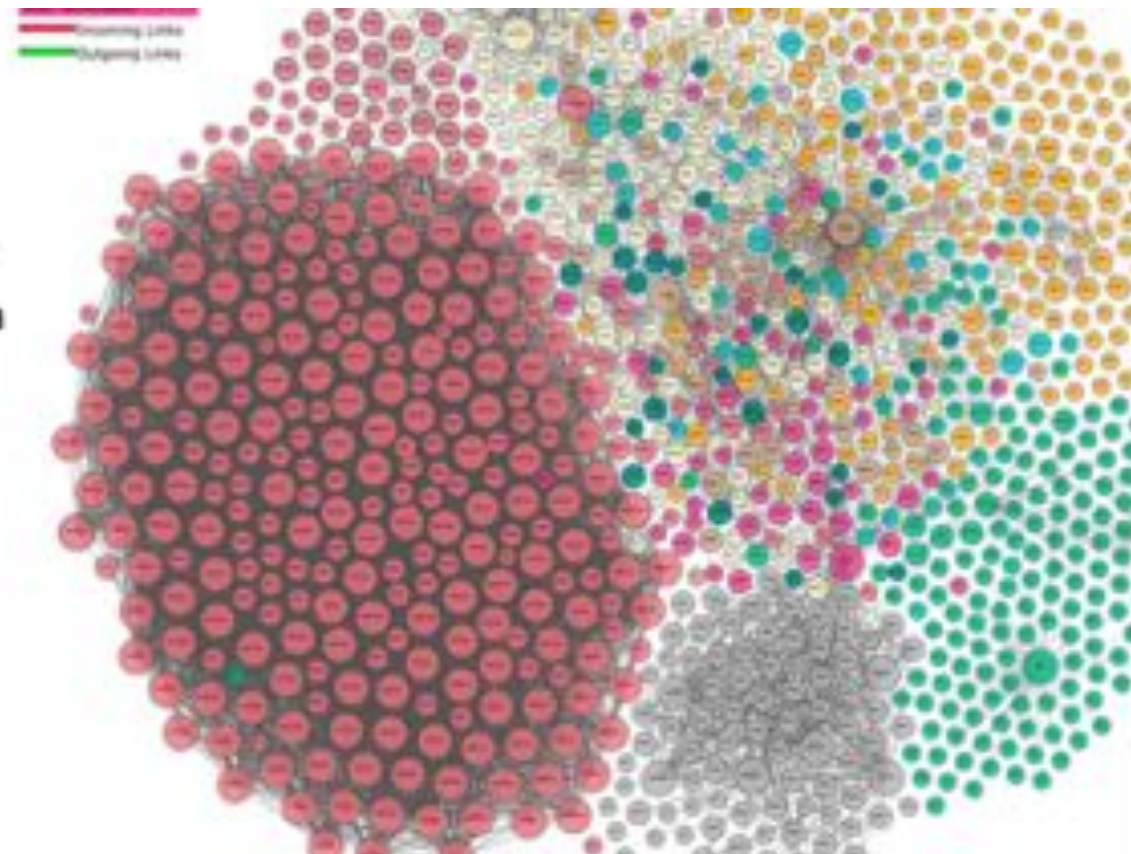
The Web of Data

Linked Data
- Linked Open Data (LOD) denote publicly available (RDF) Data in the Web, identified via URI and accessible via HTTP. Linked data link to other data via URI.
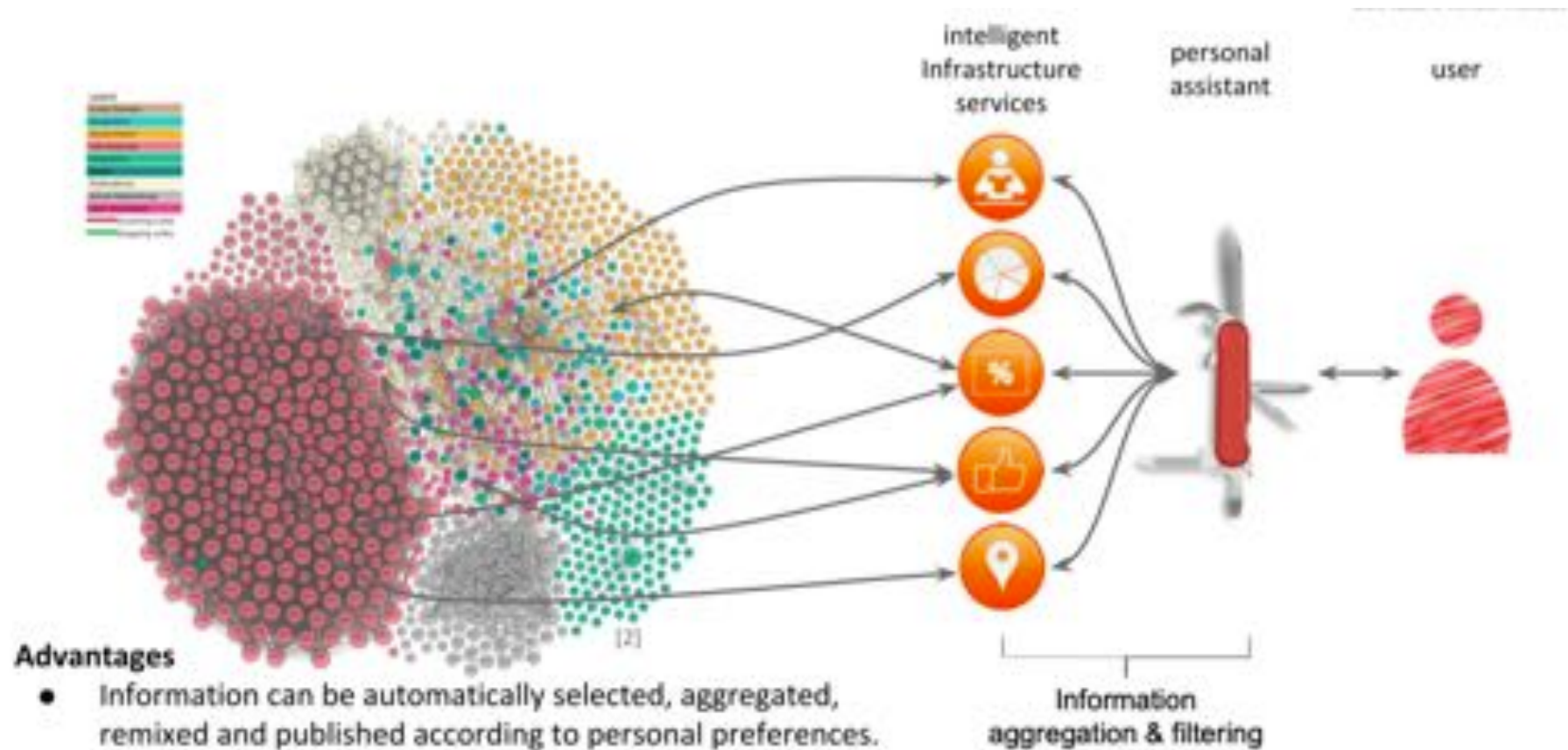
The Web of Data
- 9,960 datasets
- >149 billion facts
- >800 million links
  (April 2017)

http://lod-cloud.net/

How to Access Web of Data

intelligent Infrastructure services

personal assistant

user

Advantages

- Information can be automatically selected, aggregated, remixed and published according to personal preferences.

Information aggregation & filtering

[2]

Linked Data Principles

1. Use **URIs** as names for things.

2. Use **HTTP URIs**, so that people can **look up** those names.

3. When someone looks up a URI, provide **useful information**, using the **standards** (RDF, SPARQL).

4. Include **links to other URIs**, so that they can discover more things.

The Benefit of using Linked Data at BBC Website

- **Information is dynamically aggregated** from external, publicly available data (Wikipedia, MusicBrainz, Last.FM, Discogs,...)

- **no Screen Scraping**

- **no specialized API**

- all data **available as Linked Open Data**

- data access via **simple HTTP Request**

- data is **always up-to-date without manual interaction**

Linked Open Data

- Public Linked Data resources on the Web, licensed as Creative Common CC-BY
- Tim Berners-Lee's 5-Star Criteria for Linked Open Data

★ Available on the Web (whatever format) but with an **open licence**, to be Open Data

★★ Available as **machine-readable structured data** (e.g. excel instead of image scan of a table)

★★★ as (2) plus **non-proprietary format** (e.g. CSV instead of excel)

★★★★ All the above plus: use **open standards from W3C** (RDF and SPARQL) to identify things, so that people can point at your stuff

★★★★★ All the above, plus: **link your data to other people's data** to provide context

Knowledge
Representation with XML

Knowledge Representation - A Simple Example

- How do I represent the following fact:

  - "Pluto has been discovered in 1930" in XML?

```
<discovered>
  <discovery>Pluto</discovery>
  <year>1930</year>
</discovered>
```
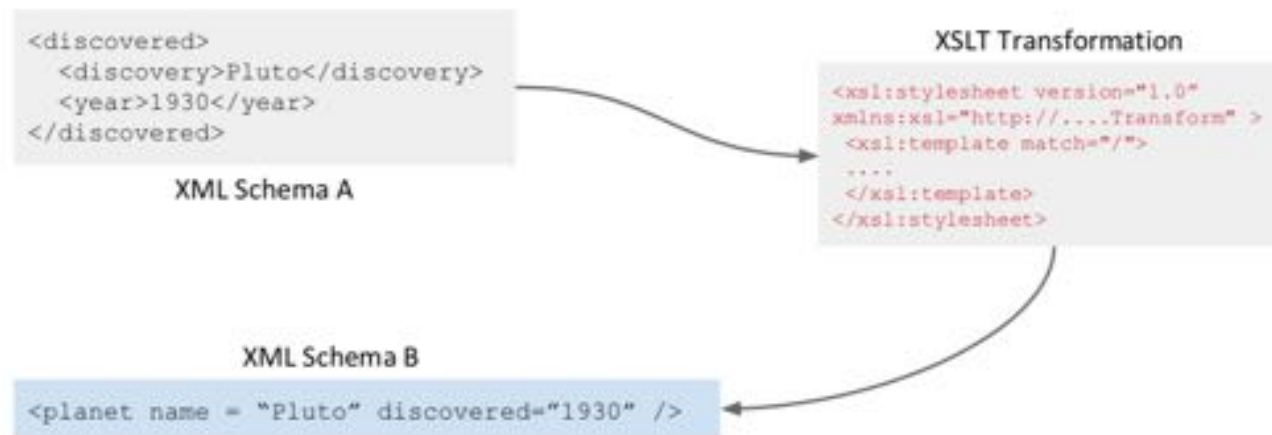
```
<planet name = "Pluto">
  <yearDiscovered>1930</yearDiscovered>
</planet>
```

  - is there a unique (intuitive) way to model knowledge (in XML)?

  -
```
<planet name = "Pluto" discovered="1930" />
```

Knowledge Representation - With XML

- In XML successful communication of information requires translation among different XML serializations

Knowledge Representation
- A Simple Example

- How do I represent the following fact:
  "Pluto has been discovered in 1930" in an intuitive way?

subject   Pluto

predicate   has been discovered in

object   1930

intuitive knowledge representation with a **directed graph**

Resource Description Framework

Basic concepts of RDF
Formats for serializing RDF data
Advanced features of RDF
What is Linked Data?
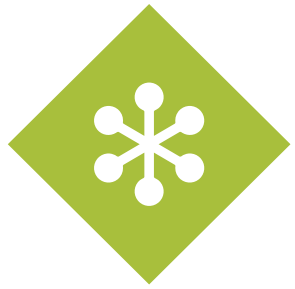
Resource Description Framework

- **Resource**
  - can be everything
  - must be uniquely identified and referencable via URI
- **Description**
  - = description of resources
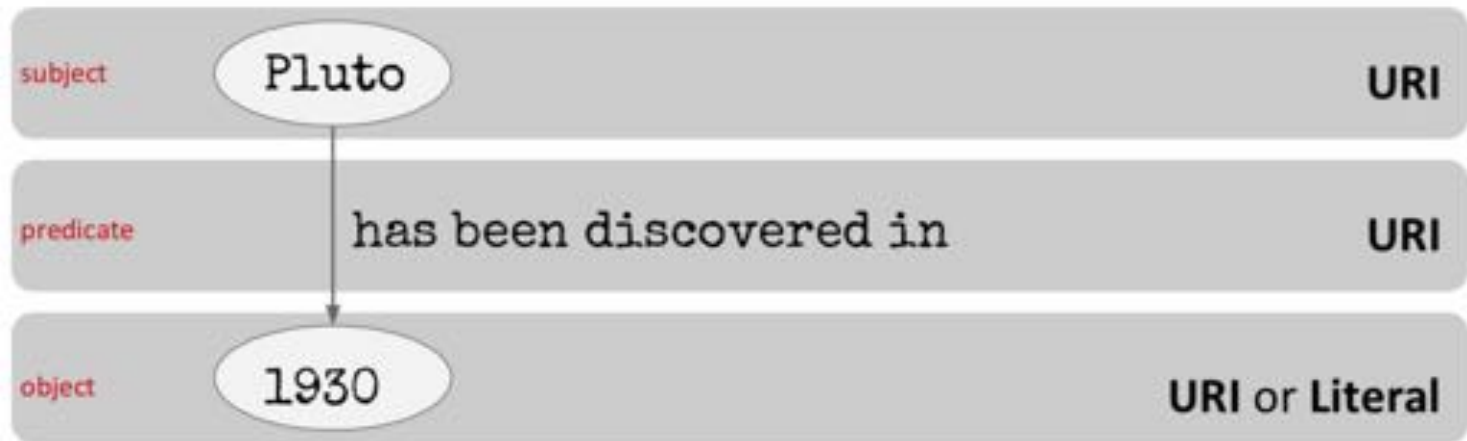 - ia representing properties and relationships among resources as graphs
- **Framework**
  - = combination of web based protocols (URI, HTTP, XML, Turtle, JSON, …)
  - based on formal model (semantics)

- Knowledge in RDF is expressed as a list of **statements**
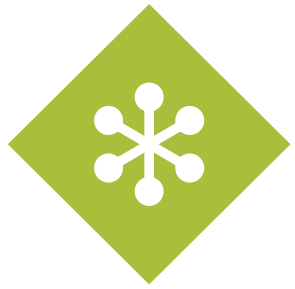- All RDF statements follow the same simple schema **(= RDF Triple)**

Resource Description Framework

URIs

Uniform Resource Identifier

- A Uniform Resource Identifier (URI) defines a simple and extensible schema for worldwide unique identification of abstract or physical resources (RFC 3986).

- A **Resource** can be every object with a clear identity (according to the context of the application)
  - as e.g., web pages, books, locations, persons, relations among objects, abstract concepts, etc.

- URI concept is already established in various domains, as e.g.,
  - the Web (URL),
  - Books and publications (ISBN, ISSN),
  - Digital Object Identifier (DOI)
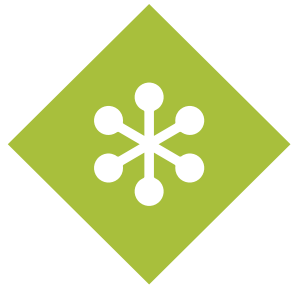
What if a URI Does Not Exist (Yet)?

- Define a URI by yourself:
  - avoid overlaps → use your own website
  - enable documentation at the same place → Content Negotiation

- Use separate URIs for the resource (Designatum) and its documentation (Designator) via
  - Content Negotiation and/or
  - URI references (e.g. via "#" fragment identifier)

Resource Description Framework

- **RDF Statements (RDF-Triple):**

Subject	+	Property	+	Object / Value	
URI		URI		URI / Literal	RDF Building Blocks

N-Triples Serialization

`<http://dbpedia.org/resource/Pluto>`   `<http://dbpedia.org/ontology/discovered>`   `"1930"` .

graph representation

Resource Description Framework

- **URIs and Literals**

  - URIs reference resources uniquely

  - Literals describe data values that don't have a separate existence.
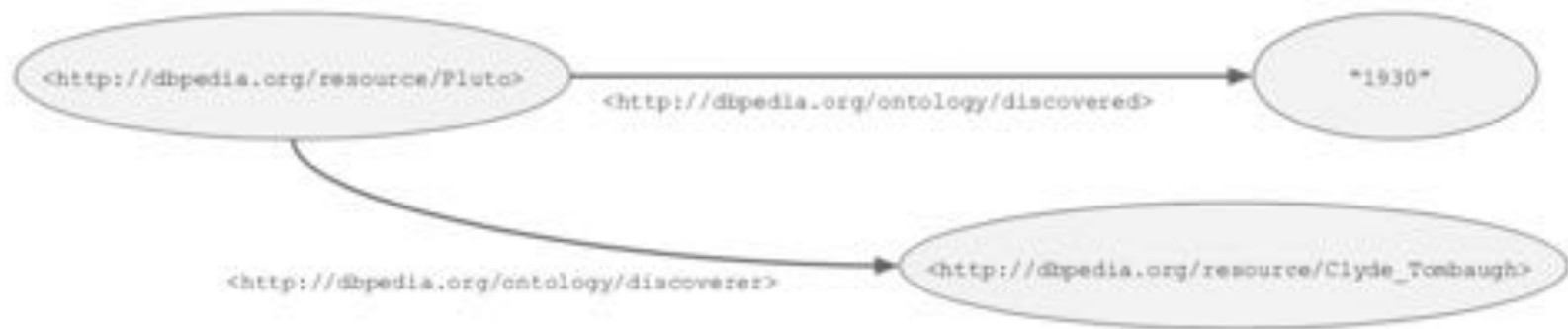
Literals and Data Types

- typed literals can be expressed via **XML Schema datatypes**
- Namespace for typed literals:
  http://www.w3.org/2001/XMLSchema#

- Examples:
  "Semantics"^^<http://www.w3.org/2001/XMLSchema#string>
  "1161.00"^^<http://www.w3.org/2001/XMLSchema#float>
  "2015-08-02"^^<http://www.w3.org/2001/XMLSchema#date>

- **Language Tags** denote the (natural) language of the text:
  - Example:
    "Semantik"@de , "Semantics"@en
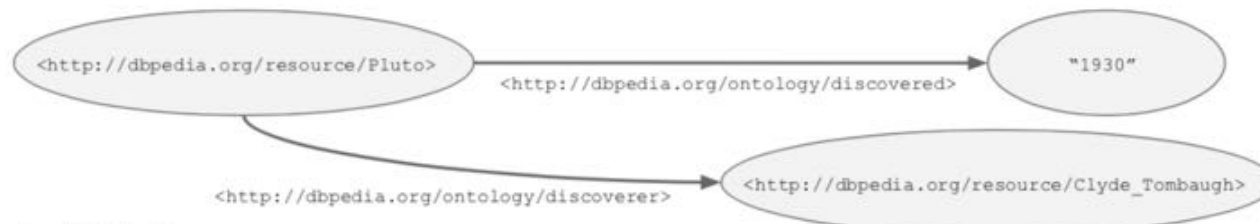
Blank Nodes

- **Blank Nodes**

  - denote existence of an individual with specific attributes, but without

  - providing an identification or reference
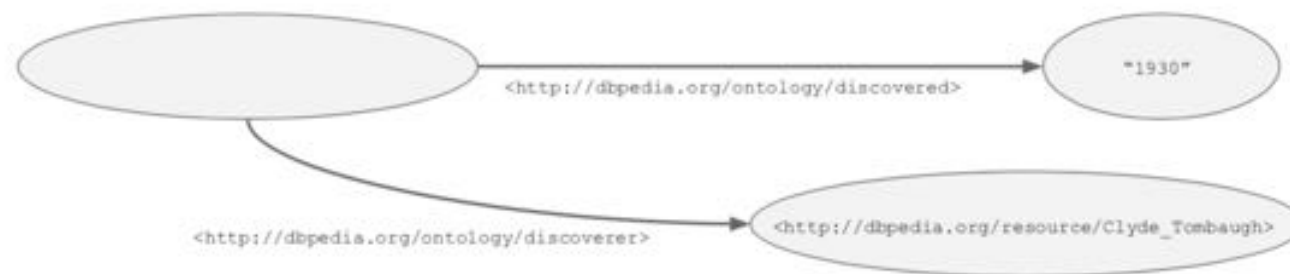
RDF Serialization

- **N-Triples Notation**
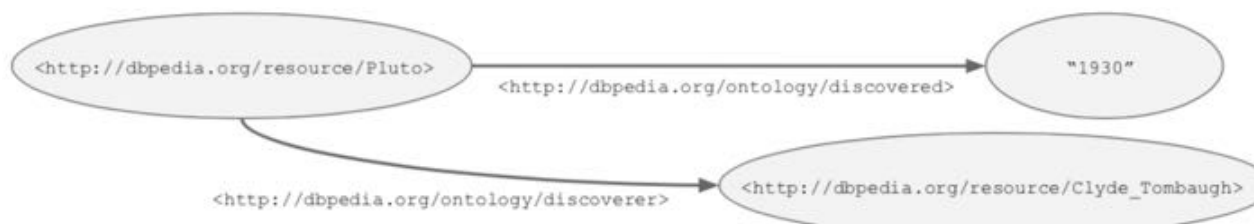 - URIs/IRIs in angle brackets
  - Literals in quotation marks
  - Triple ends with a period

```
<http://dbpedia.org/resource/Pluto> <http://dbpedia.org/ontology/discovered> "1930" .

<http://dbpedia.org/resource/Pluto> <http://dbpedia.org/ontology/discoverer>
<http://dbpedia.org/resource/Clyde_Tombaugh> .
```

RDF Serialization

- **RDF/XML Notation**

S	P	O

```xml
<?xml version="1.0" encoding="utf-8" ?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
         xmlns:ns0="http://dbpedia.org/ontology/">
  <rdf:Description rdf:about="http://dbpedia.org/resource/Pluto">
    <ns0:discovered>1930</ns0:discovered>
    <ns0:discoverer rdf:resource="http://dbpedia.org/resource/Clyde_Tombaugh"/>
  </rdf:Description>
</rdf:RDF>
```
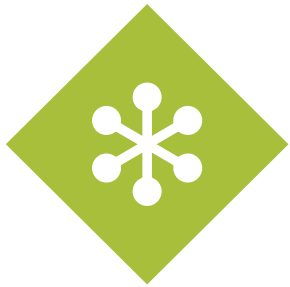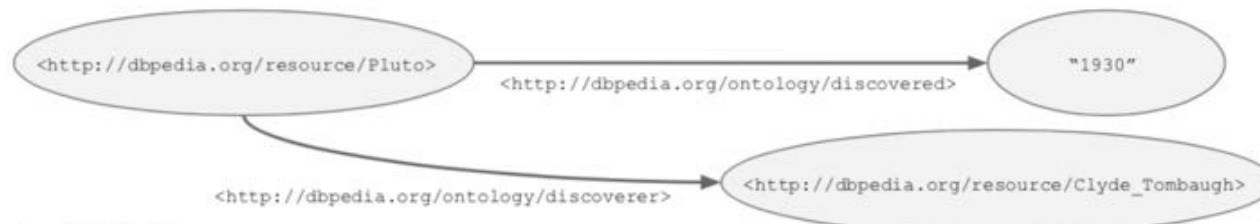
RDF Serialization

- **Turtle (Terse RDF Triple Language) Notation**
  - extension of N_Triples
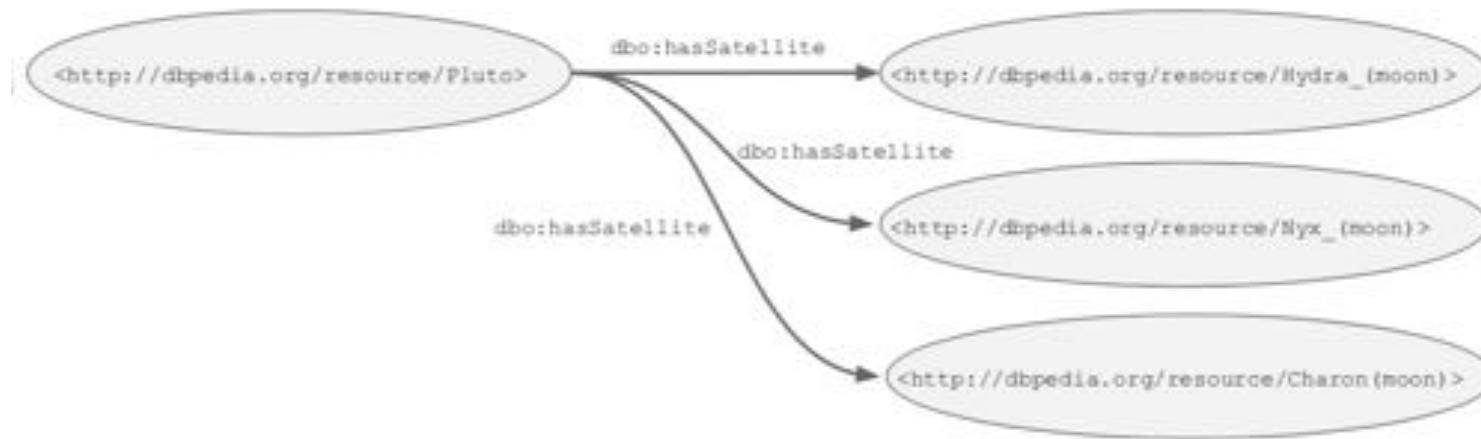
```
@prefix dbo: <http://dbpedia.org/ontology/> .
@base   <http://dbpedia.org/resource/> .

<Pluto> dbo:discovered "1930" .

<Pluto> dbo:discoverer <Clyde_Tombaugh> .
```

RDF Serialization

- **Abbreviating in Turtle**


```
@prefix dbo: <http://dbpedia.org/ontology/> .
@base    <http://dbpedia.org/resource/> .

<Pluto> dbo:hasSatellite <Hydra_(moon)> ,
                         <Nyx_(moon)> ,
                         <Charon_(moon)> .
```

comma indicates that subsequent triples have same subject and property **(object list)**

Turtle - Terse RDF

- **Typed Literals**

```
@prefix dbo: <http://dbpedia.org/ontology/> .
@prefix xsd: <http://www.w3c.org/2001/XMLSchema#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@base  <http://dbpedia.org/resource/> .

<Pluto> dbo:meanRadius "1161.00"^^xsd:double ;
        dbo:meanTemperature "44.00"^^xsd:double ;
        dbo:discovered "1930-02-18"^^xsd:date ;
        rdfs:label "Pluto"@en .
```
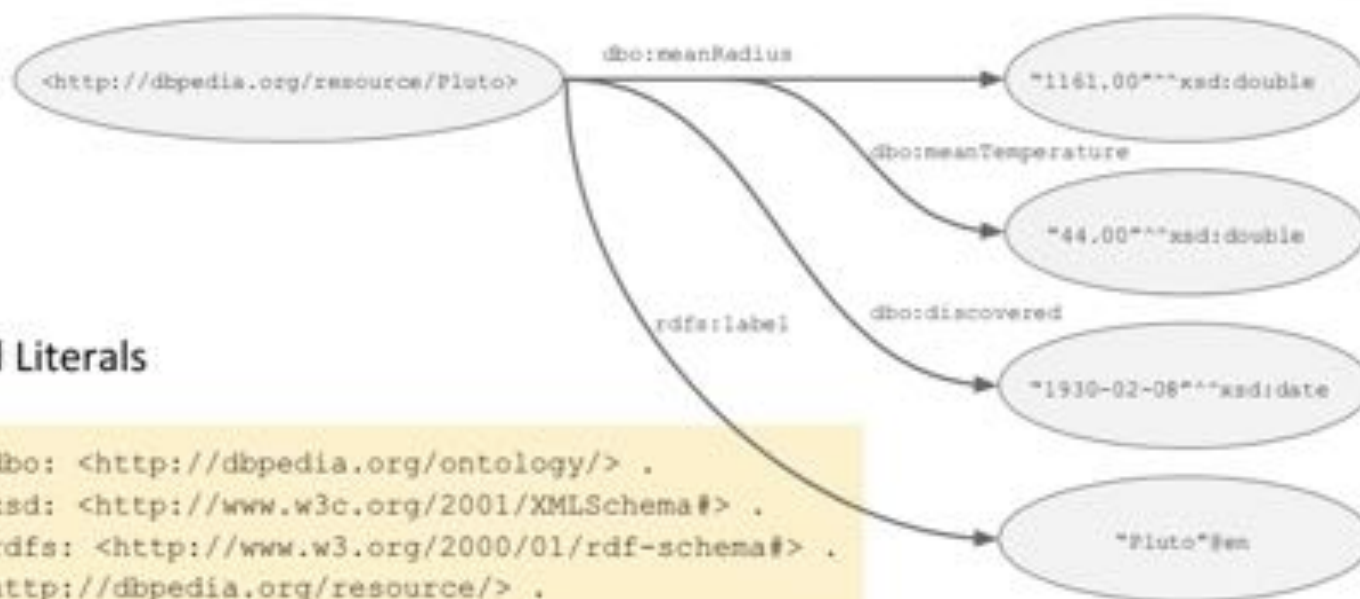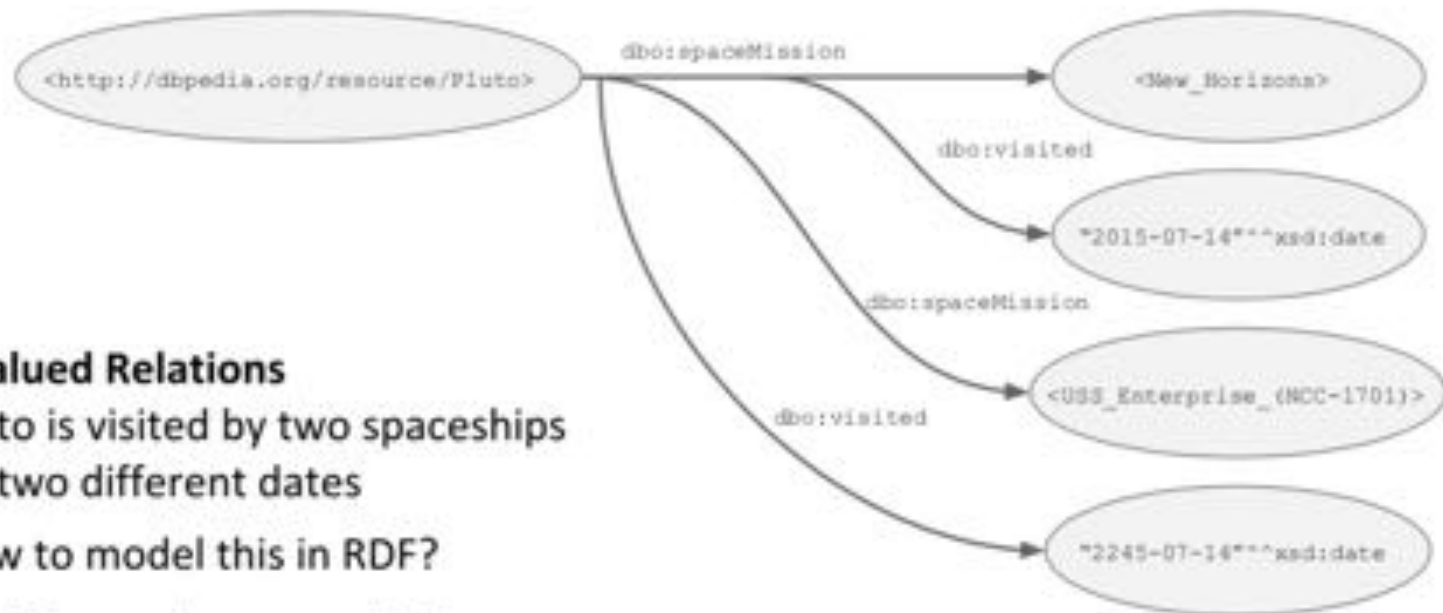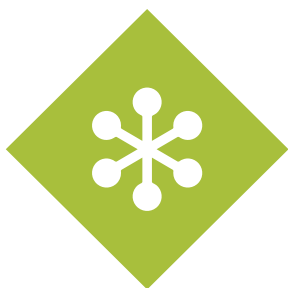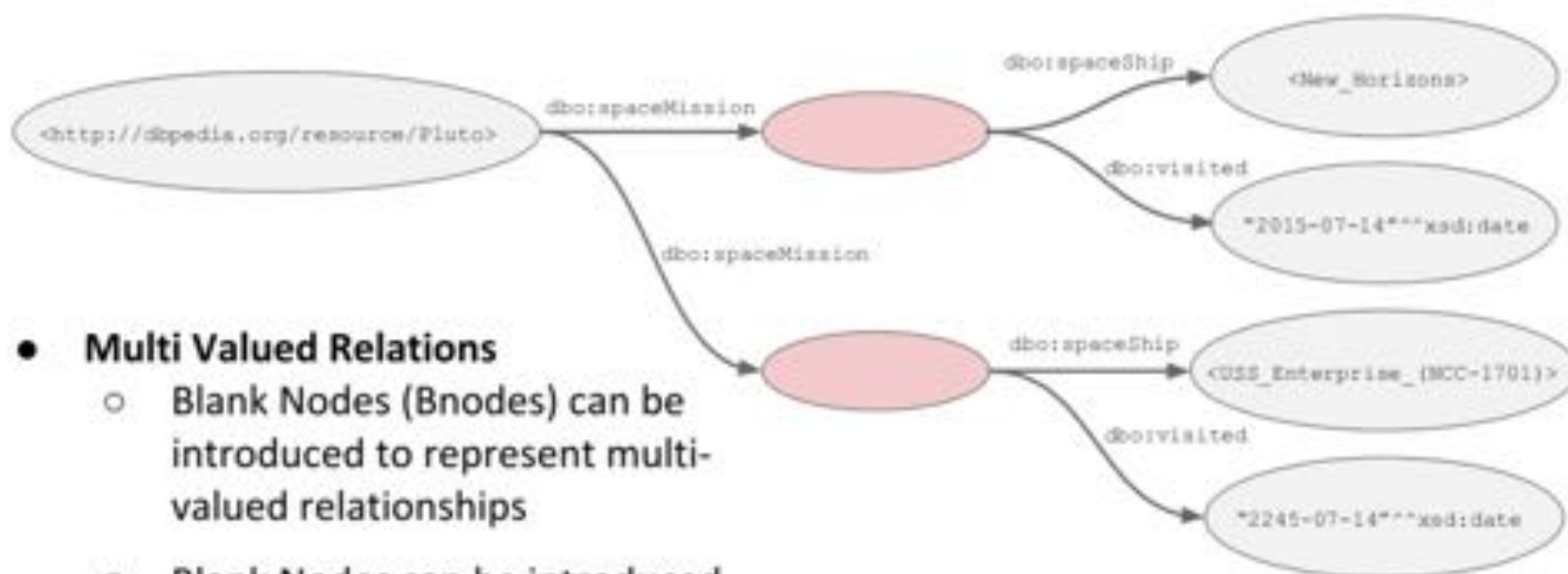
Multi-Valued Relations

- **Multi Valued Relations**
  - Pluto is visited by two spaceships on two different dates
  - How to model this in RDF?
  - **Problem:** unique association
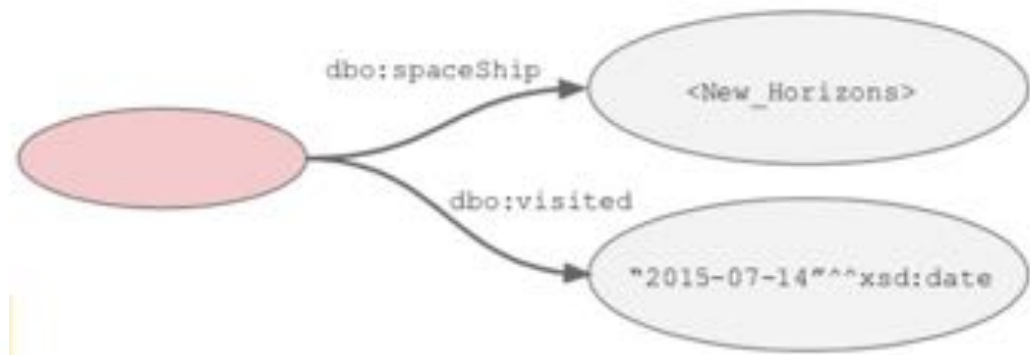
Blank Nodes

- **Multi Valued Relations**
  - Blank Nodes (Bnodes) can be introduced to represent multi-valued relationships
  - Blank Nodes can be introduced for resources that don't need a name (auxiliary nodes)
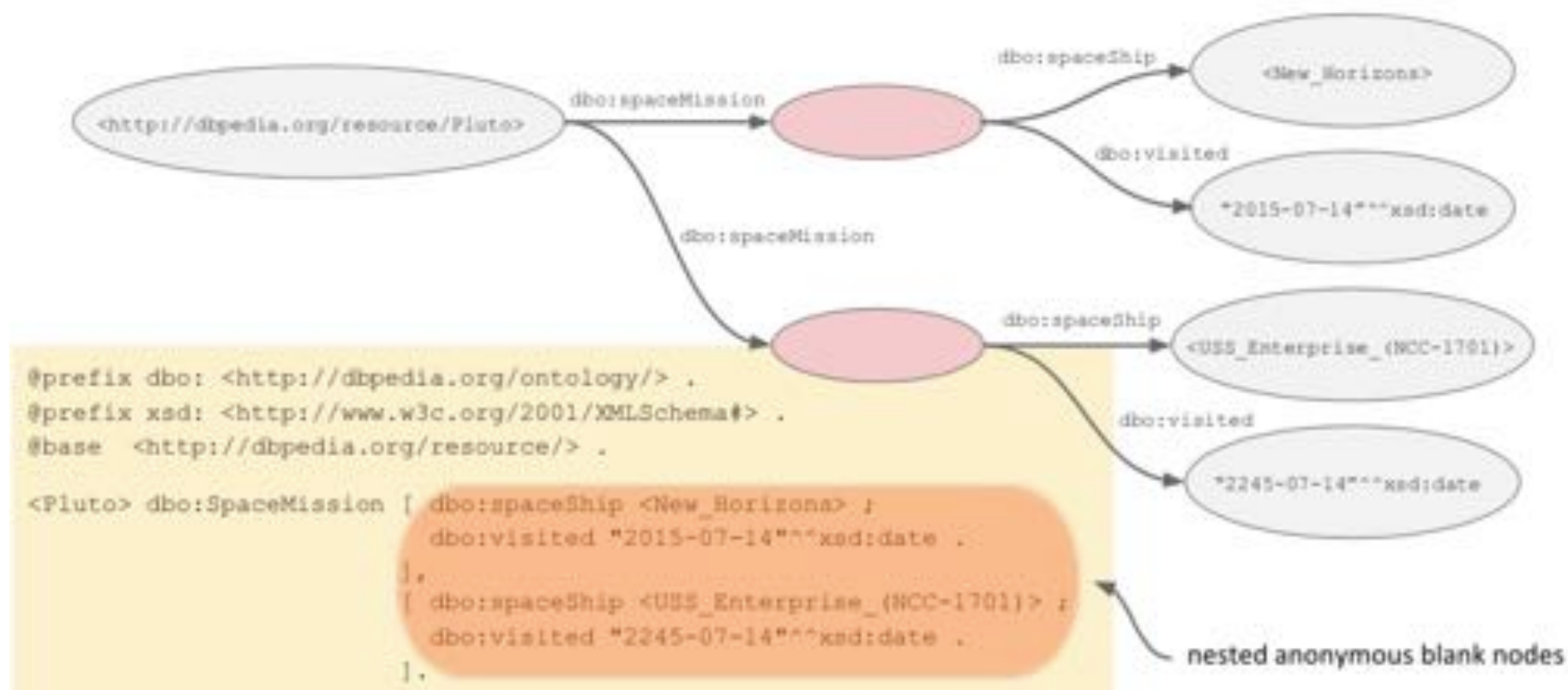
Turtle - Terse RDF


```
@prefix dbo: <http://dbpedia.org/ontology/> .
@prefix xsd: <http://www.w3c.org/2001/XMLSchema#> .
@base  <http://dbpedia.org/resource/> .

[] dbo:spaceShip <New_Horizons> ;
   dbo:visited "2015-07-14"^^xsd:date .
```
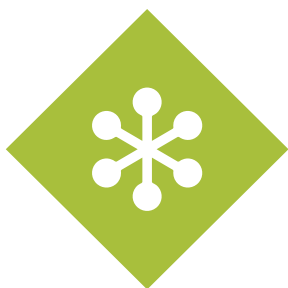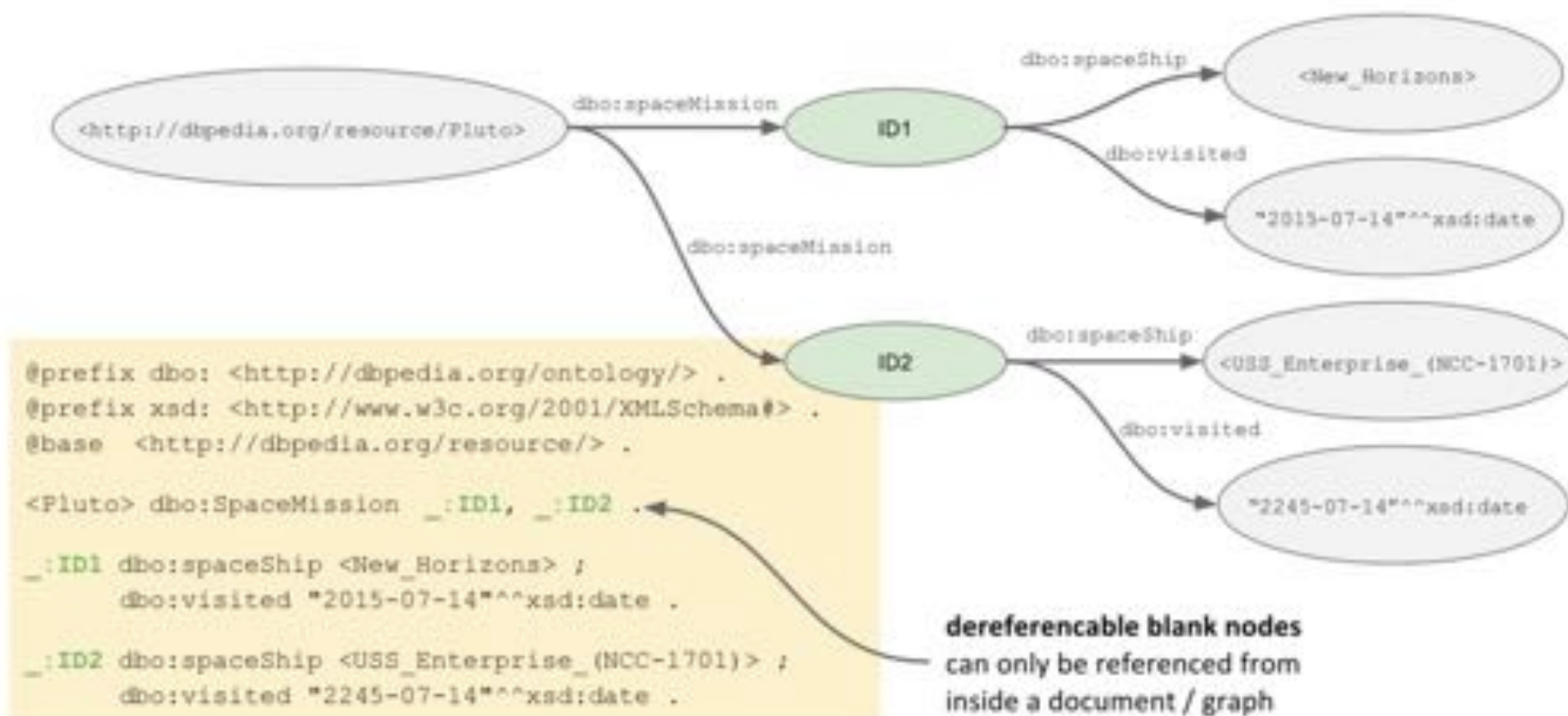
anonymous blank node as subject

Turtle - Terse RDF

Turtle - Terse RDF

dbo:spaceShip → <New_Horizons>
dbo:spaceMission → ID1
dbo:visited → "2015-07-14"^^xsd:date
dbo:spaceMission → ID2
dbo:spaceShip → <USS_Enterprise_(NCC-1701)>
dbo:visited → "2245-07-14"^^xsd:date

<http://dbpedia.org/resource/Pluto>

```
@prefix dbo: <http://dbpedia.org/ontology/> .
@prefix xsd: <http://www.w3c.org/2001/XMLSchema#> .
@base  <http://dbpedia.org/resource/> .

<Pluto> dbo:SpaceMission  _:ID1, _:ID2 .

_:ID1 dbo:spaceShip <New_Horizons> ;
      dbo:visited "2015-07-14"^^xsd:date .

_:ID2 dbo:spaceShip <USS_Enterprise_(NCC-1701)> ;
      dbo:visited "2245-07-14"^^xsd:date .
```

dereferencable blank nodes
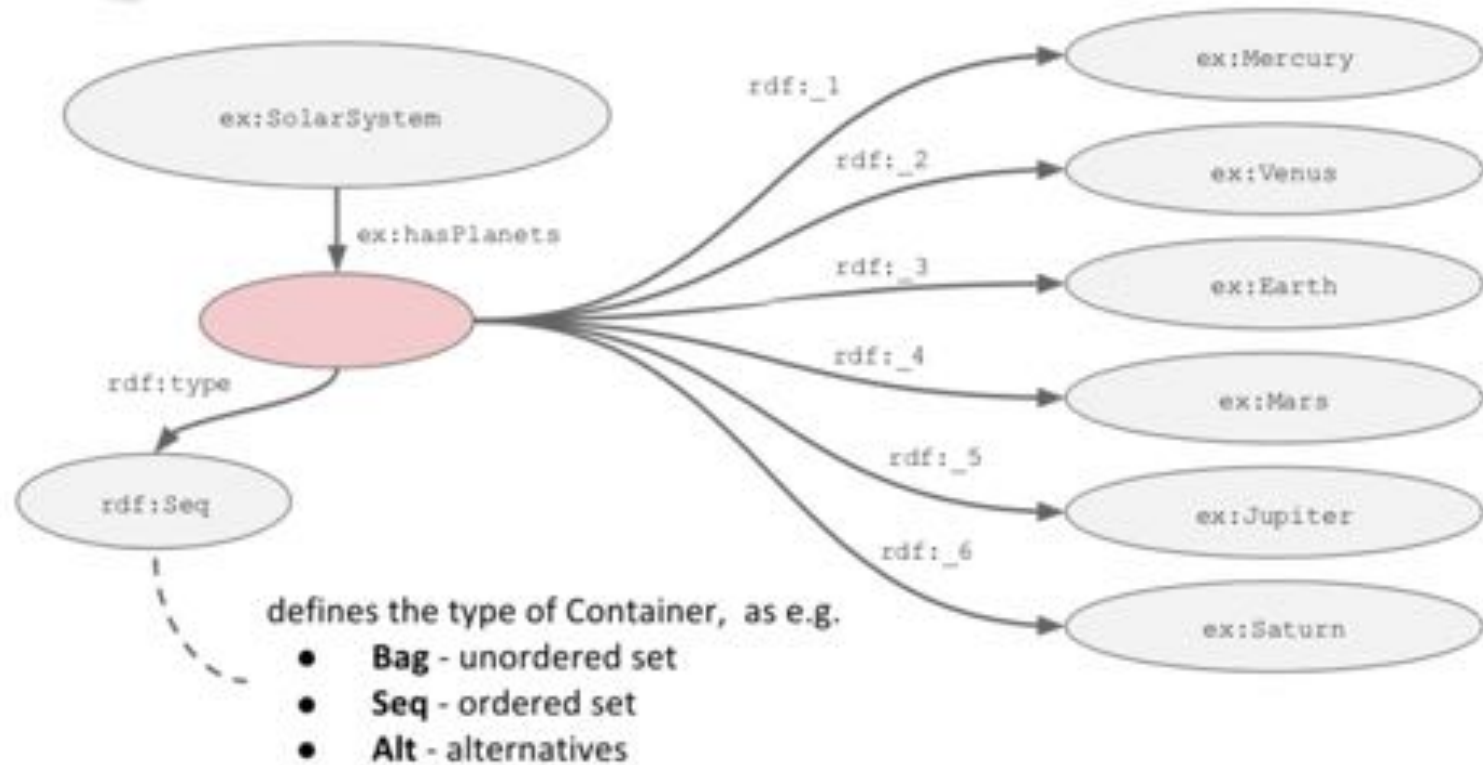can only be referenced from
inside a document / graph

RDF Lists

- General Data structure to enumerate any resources or literals
- Only shortcuts without additional semantic expressivity

- Distinguish between
  - **Container**: open list, i.e. extension (new entries) possible
  - **Collection**: closed list, i.e. no extension possible

RDF Container

RDF Container

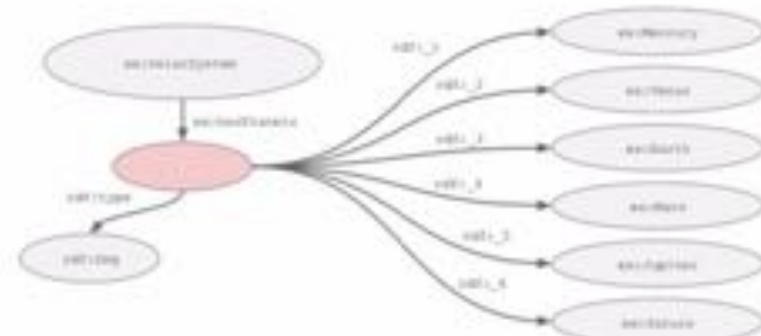```
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix ex: <http://example.org/test#> .

ex:SolarSystem ex:hasPlanets [
  a rdf:Seq ;
  rdf:_1 ex:Mercury ;
  rdf:_2 ex:Venus ;
  rdf:_3 ex:Earth ;
  rdf:_4 ex:Mars ;
  rdf:_5 ex:Jupiter ;
  rdf:_6 ex:Saturn
] .
```
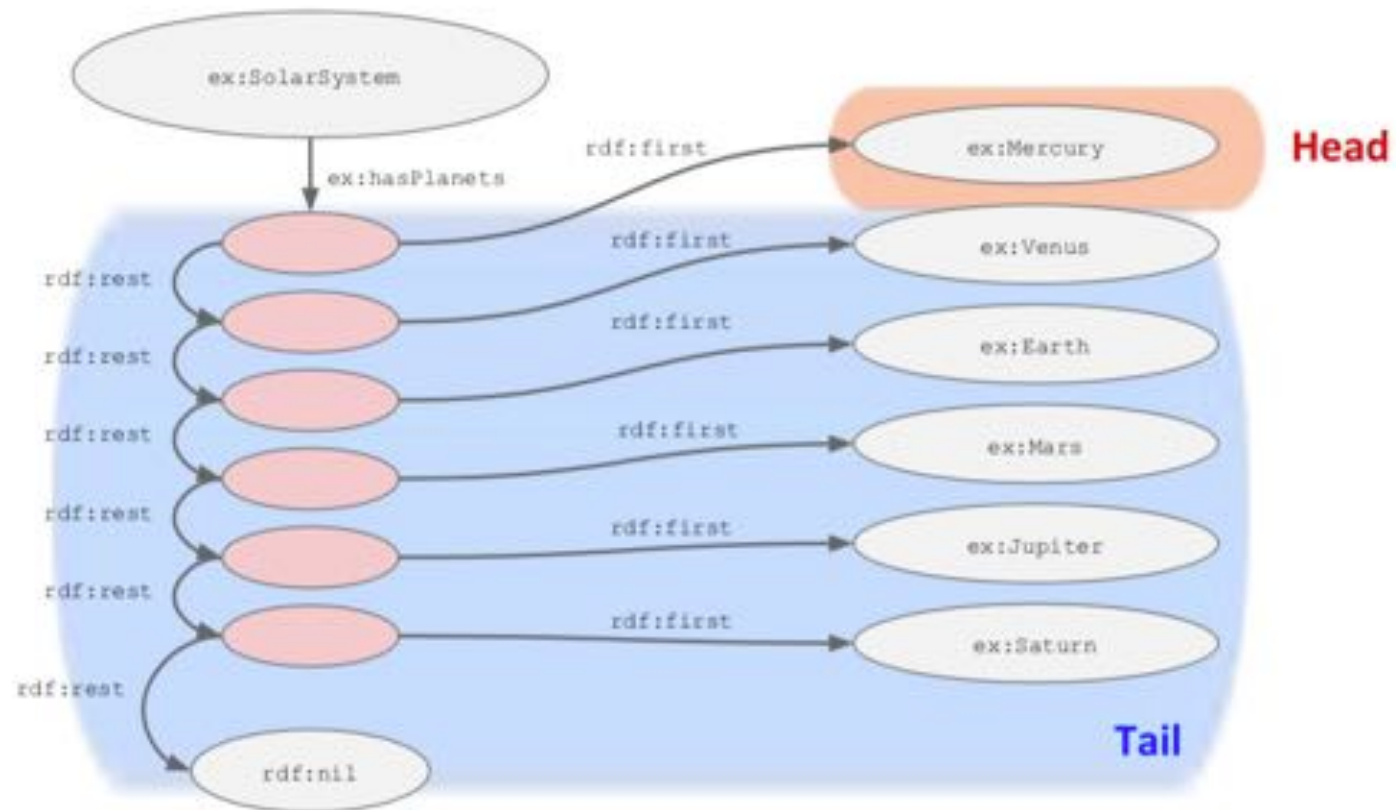
# RDF Collection

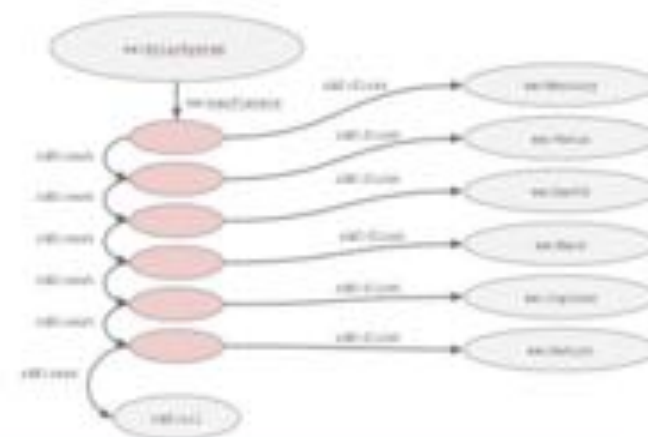# RDF Collection



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

ex:SolarSystem ex:hasPlanets [
  rdf:first ex:Mercury ; rdf:rest [
   rdf:first ex:Venus ; rdf:rest [
     rdf:first ex:Earth ; rdf:rest [
      rdf:first ex:Mars ; rdf:rest [
        rdf:first ex:Jupiter ; rdf:rest [
         rdf:first ex:Saturn ;
         rdf:rest rdf:nil
        ]
       ]
      ]
     ]
    ]
  ]
] .
```

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

ex:SolarSystem ex:hasPlanets (
  ex:Mercury ex:Venus ex:Earth ex:Mars ex:Jupiter ex:Saturn
) .
```

RDF Serialization

- **JSON-LD Notation (RDF 1.1)**