Web of Data -
Identifying standards and technologies

“RAW DATA NOW!” - Linked Open 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.**

https://www.w3.org/DesignIssues/LinkedData.html

“RAW DATA NOW!” - Linked Open Data Principles

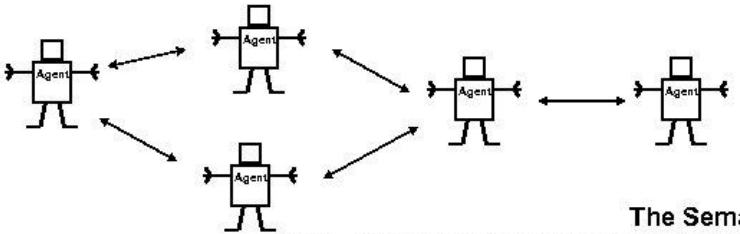
URI für Graz auf
http://dbpedia.org/resource/Graz
http://dbpedia.org/data/Graz.rdf

URI für Graz auf
https://www.wikidata.org/wiki/Q13298
https://www.wikidata.org/wiki/Special:EntityData/Q13298.rdf

The Semantic Web will enable machines to COMPREHEND semantic documents and data, not human speech and writings.

The Agent Vision

The Semantic Web will bring structure to the meaningful content of Web pages, creating an environment where software agents roaming from page to page can readily carry out sophisticated tasks for users.



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
Scientific American

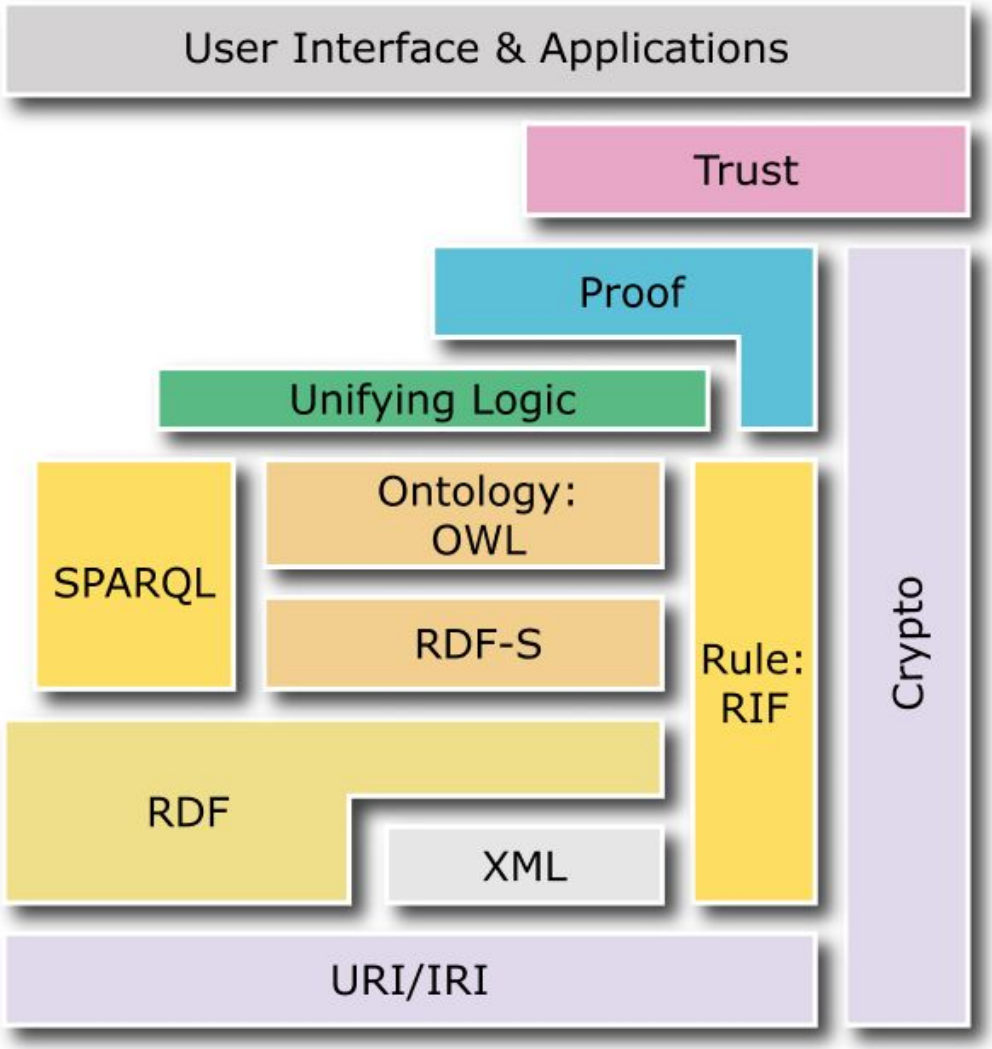
TIM BERNERS-LEE, JAMES HENDLER and ORA LASSILA: THE SEMANTIC WEB, https://www.jstor.org/stable/pdf/26059207.pdf?refreqid=excelsior%3A1d9c33aa1ea640d57940082b42df15e6 , 2001

Kritik am Semantic Web

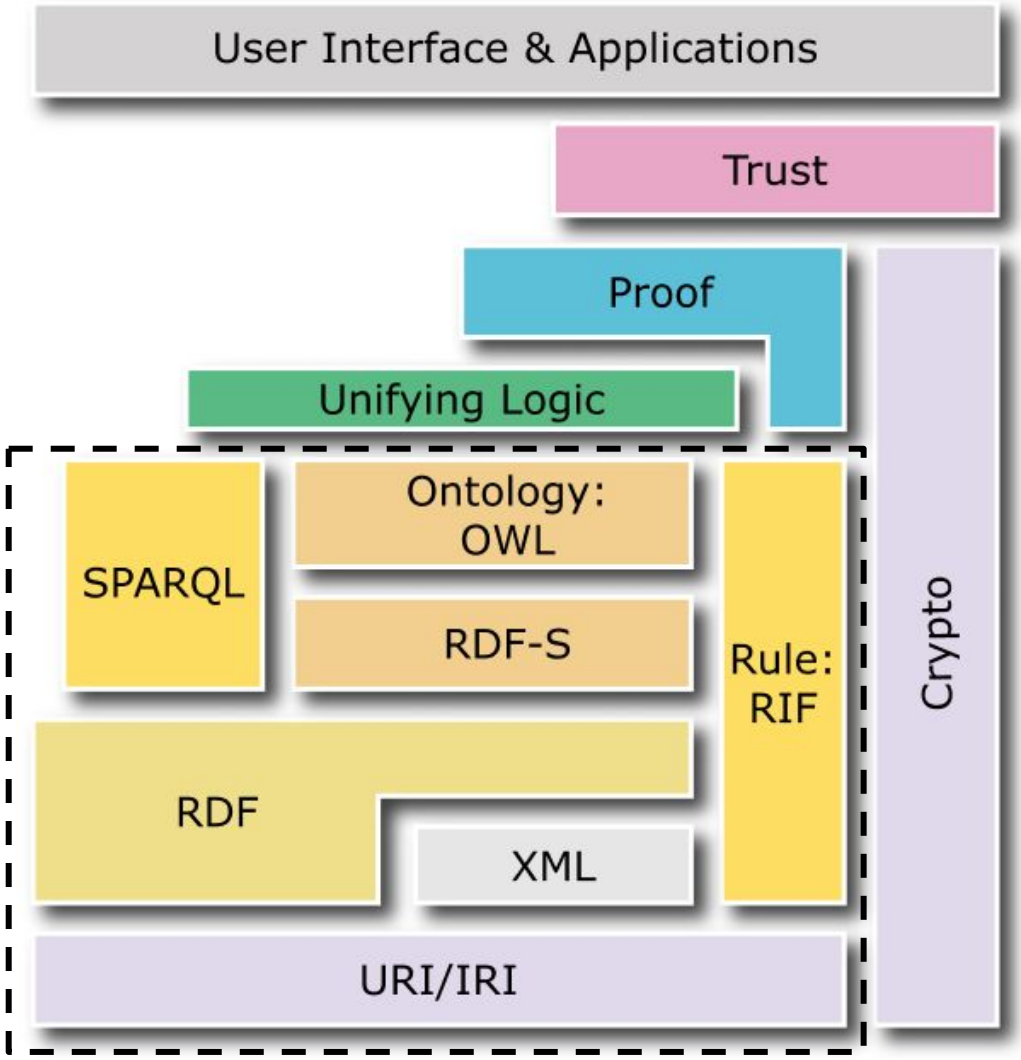
Zur Vertiefung

Aaron Swartz: A Programmable Web. An Unfinished Work, https://www.morganclaypool.com/doi/pdfplus/10.2200/S00481ED1V01Y201302WBE005 .

Web Of Data Stack



Web Of Data Stack

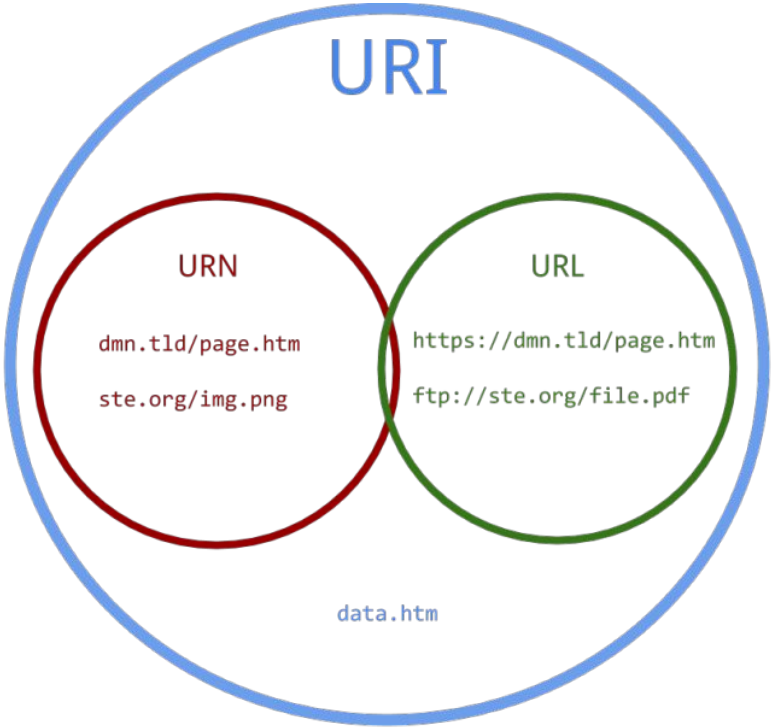


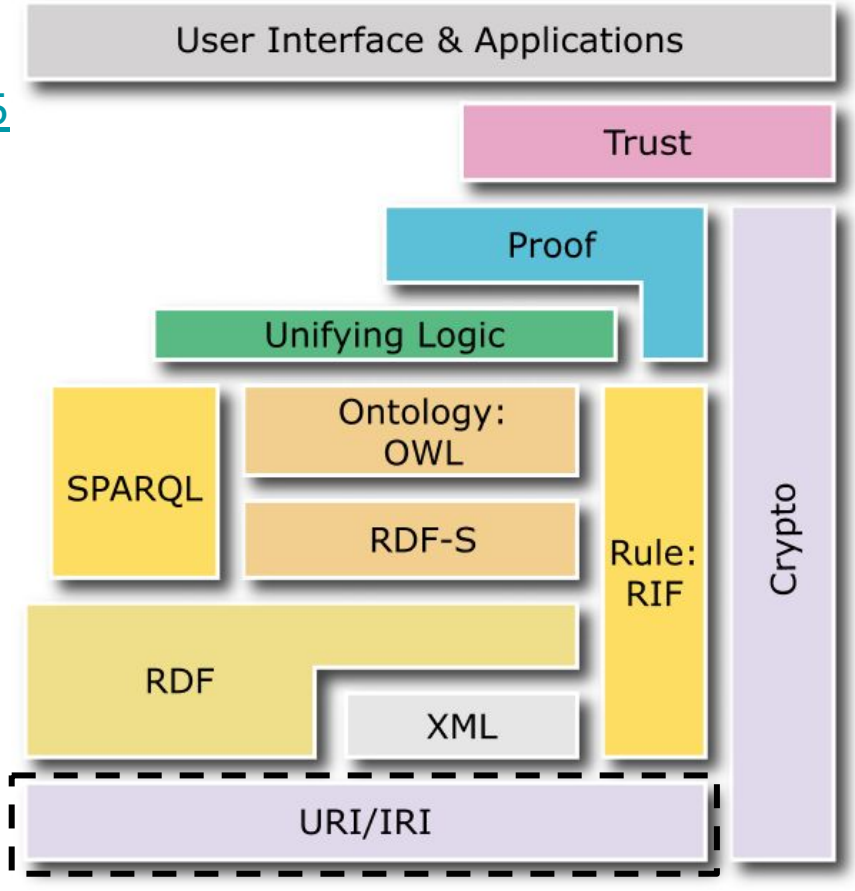
Semantic Web Layer Cake Tweak, Explained:

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

Uniform Resource Identifier / Internationalized Resource identifier

https://www.wikidata.org/wiki/Q2695156

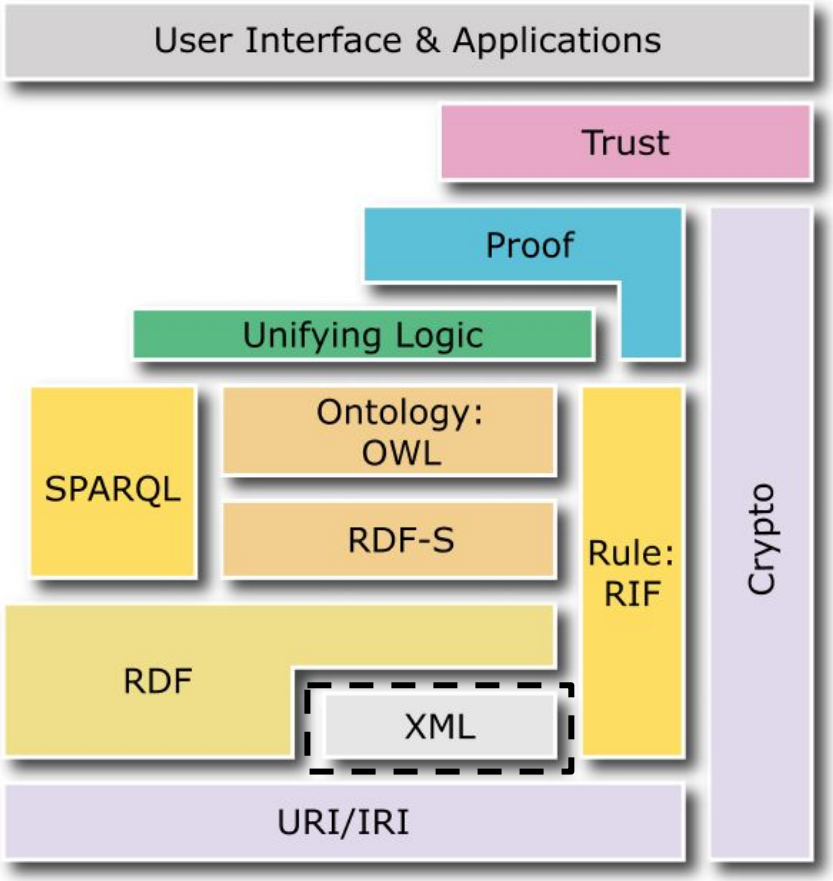




XML - Extensible Markup Language

[XML is no longer a fundamental standard.]

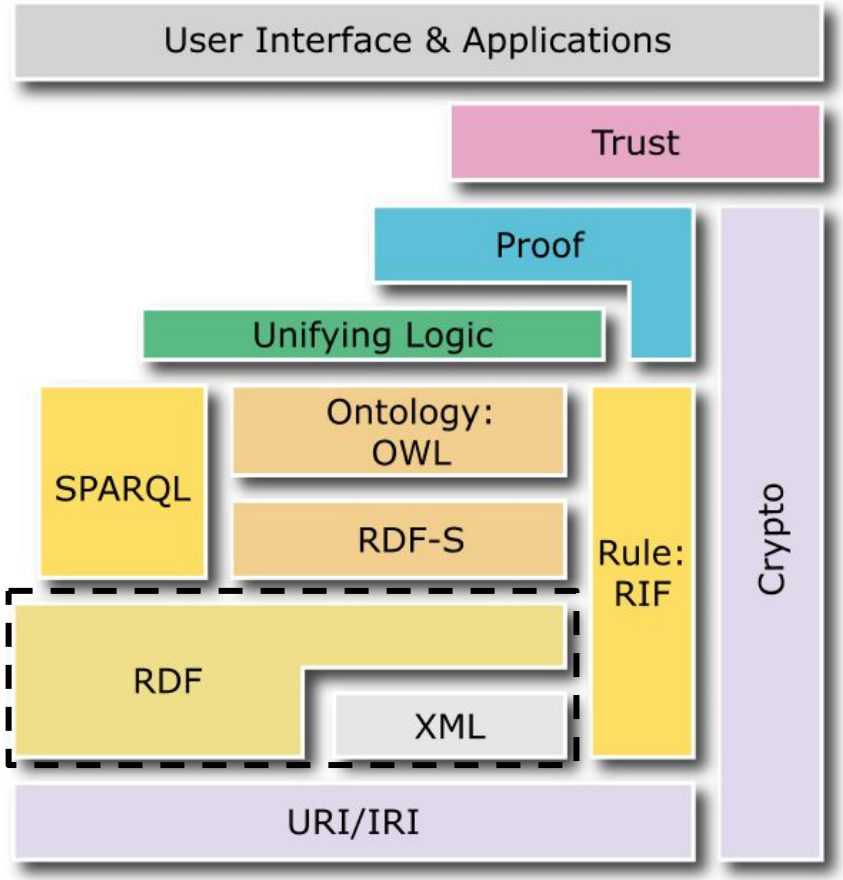
```
<note>
  <to>Gunter</to>
  <from>Christopher</from>
  <heading>Reminder</heading>
  <body>
    Modellieren ist cool!
  </body>
</note>
```

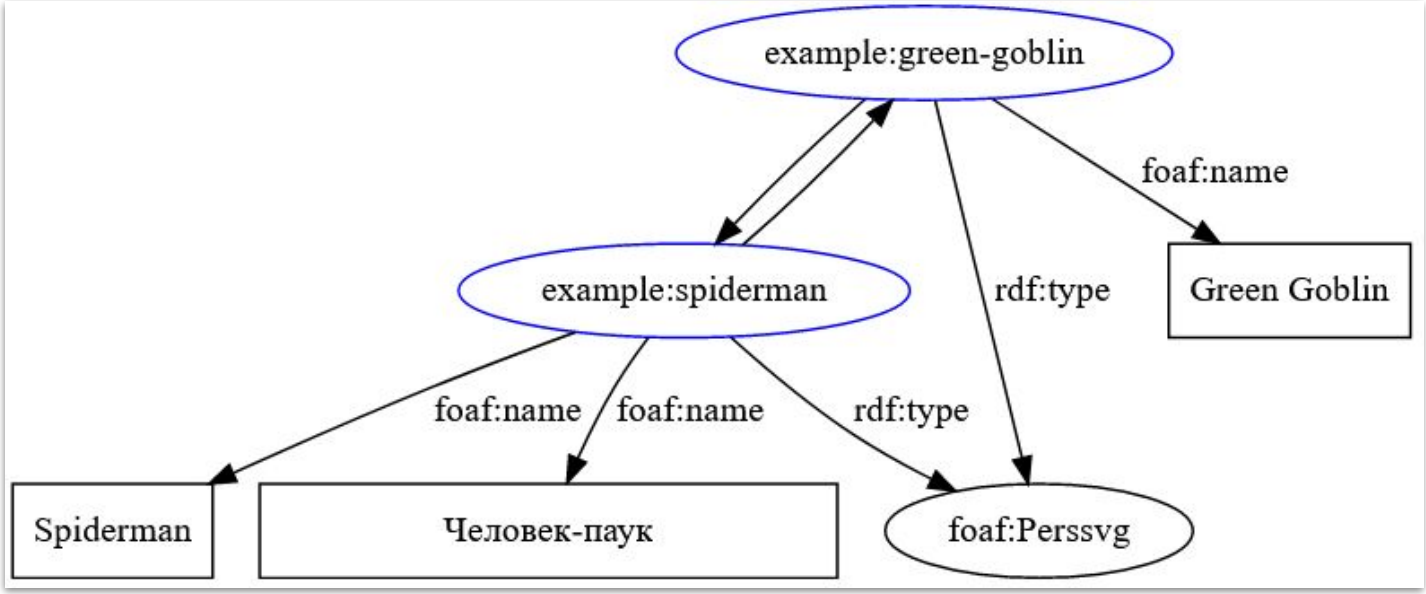


RDF - Resource Description Framework

```
@prefix example: <http://example.org/> .
@prefix rdf:     <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs:    <http://www.w3.org/2000/01/rdf-schema#> .
@prefix foaf:    <http://xmlns.com/foaf/0.1/> .
@prefix rel:     <http://www.perceive.net/schemas/relationship/> .
```

```
<example:green-goblin>
    rel:enemyOf <example:spiderman> ;
    a foaf:Person ;
    foaf:name "Green Goblin" .

<example:spiderman>
    rel:enemyOf <example:green-goblin> ;
    a foaf:Person ;
    foaf:name "Spiderman";
    foaf:name "Человек-паук"@ru .
```

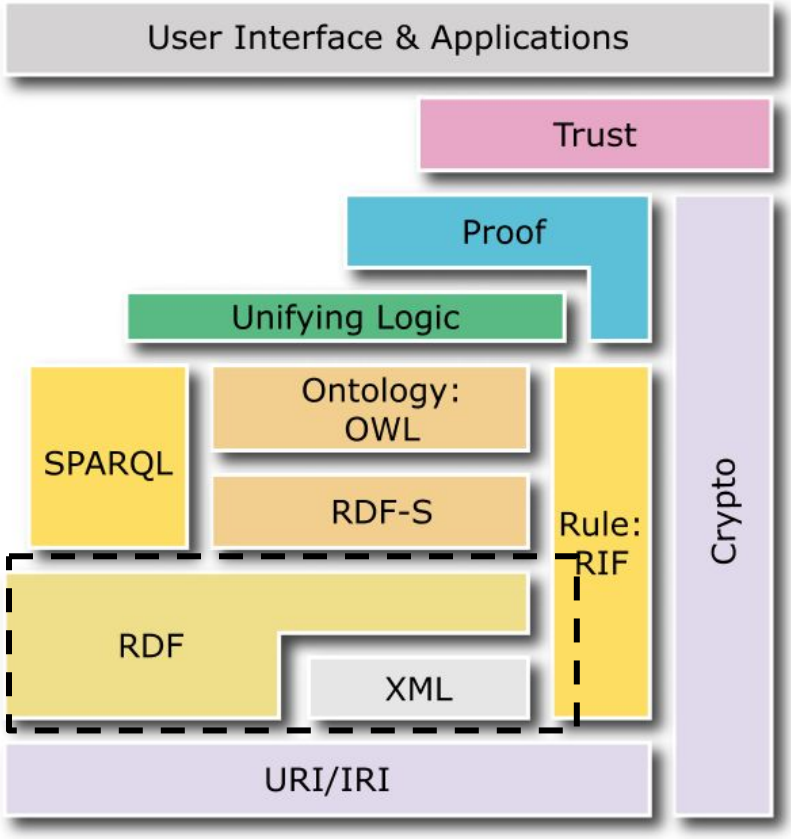





```
<example:green-goblin>
    rel:enemyOf <example:spiderman> ;
    a foaf:Person ;
    foaf:name "Green Goblin" .

<example:spiderman>
    rel:enemyOf <example:green-goblin> ;
    a foaf:Person ;
    foaf:name "Spiderman";
    foaf:name "Человек-паук"@ru .
```

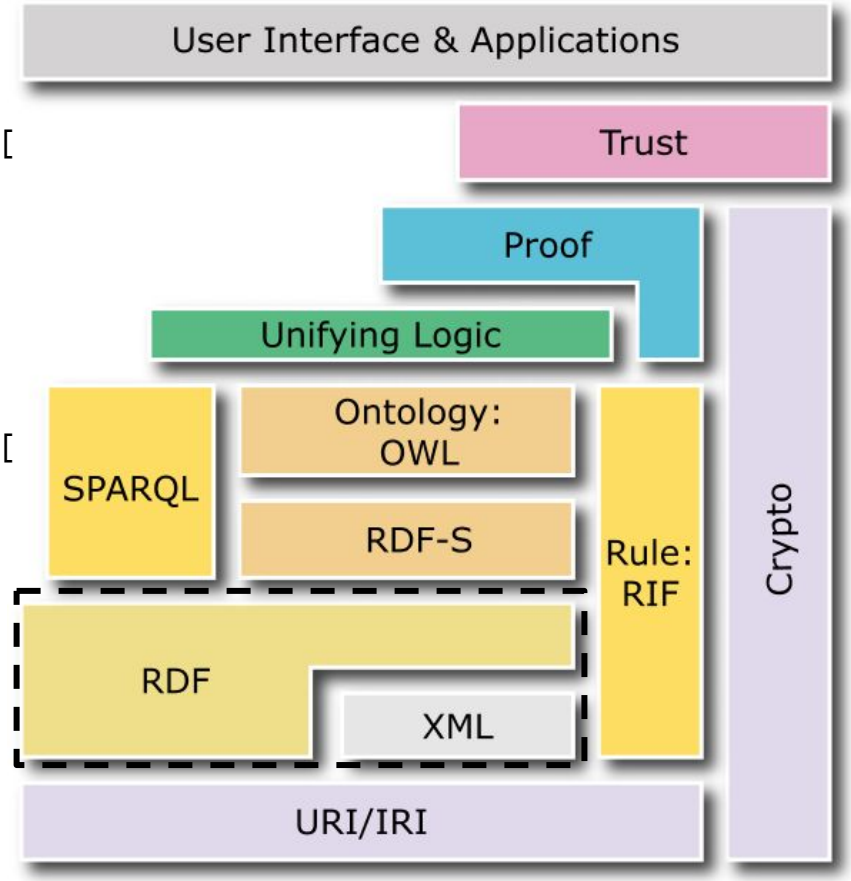
RDF/XML

```
<?xml version="1.0" encoding="utf-8" ?>
<rdf:RDF
xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:foaf="http://xmlns.com/foaf/0.1/"
xmlns:ns0="http://www.perceive.net/schemas/relationship/">

  <foaf:Person rdf:about="example:green-goblin">
    <ns0:enemyOf rdf:resource="example:spiderman"/>
    <foaf:name>Green Goblin</foaf:name>
  </foaf:Person>

  <foaf:Person rdf:about="example:spiderman">
        <ns0:enemyOf rdf:resource="example:green-goblin"/>
        <foaf:name>Spiderman</foaf:name>
        <foaf:name xml:lang="ru">Человек-паук</foaf:name>
  </foaf:Person>


</rdf:RDF>
```



JSON-LD

```
[{"@id":"example:green-goblin",
    "@type":["http://xmlns.com/foaf/0.1/Person"],
    "http://www.perceive.net/schemas/relationship/enemyOf":[
        {"@id":"example:spiderman"}],
      "http://xmlns.com/foaf/0.1/name":[
        {"@value":"Green Goblin"}
      ]
   },
   {"@id":"example:spiderman",
    "@type":["http://xmlns.com/foaf/0.1/Person"],
    "http://www.perceive.net/schemas/relationship/enemyOf":[
        { "@id":"example:green-goblin"}],
      "http://xmlns.com/foaf/0.1/name":[
        {"@value":"Spiderman"},
        { "@value":"Человек-паук","@language":"ru"}]
   },
   {
      "@id":"http://xmlns.com/foaf/0.1/Person"
   }
]
```



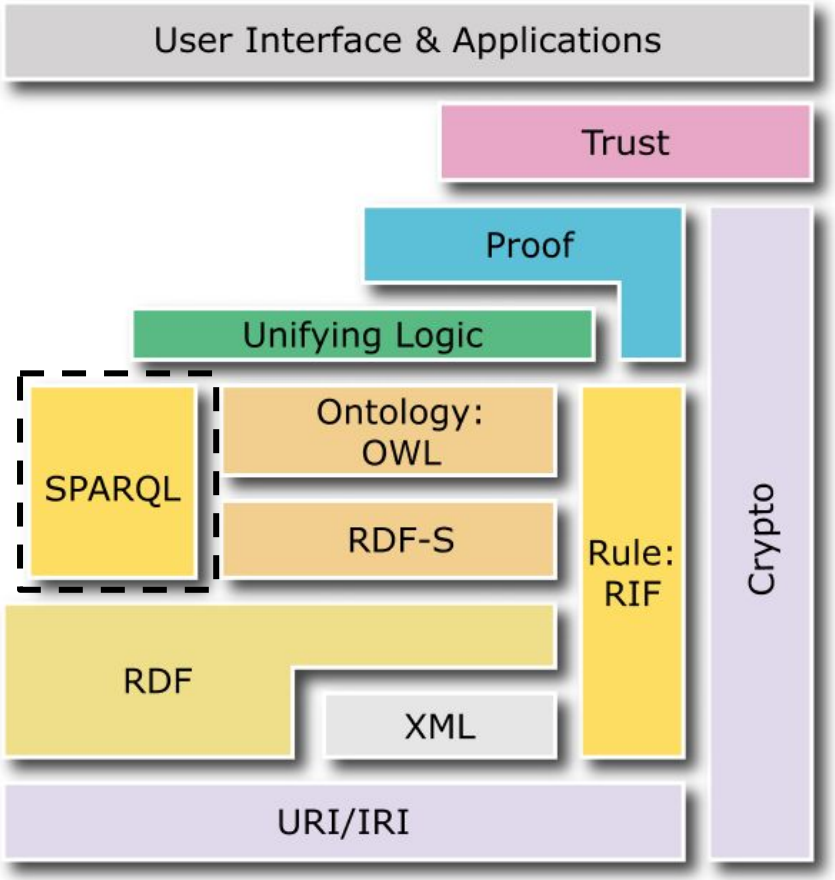
SPARQL Query Language for RDF

```
PREFIX foaf:<http://xmlns.com/foaf/0.1/>

SELECT ?name ?mbox
WHERE
  { ?x foaf:name ?name .
    ?x foaf:mbox ?mbox }
```

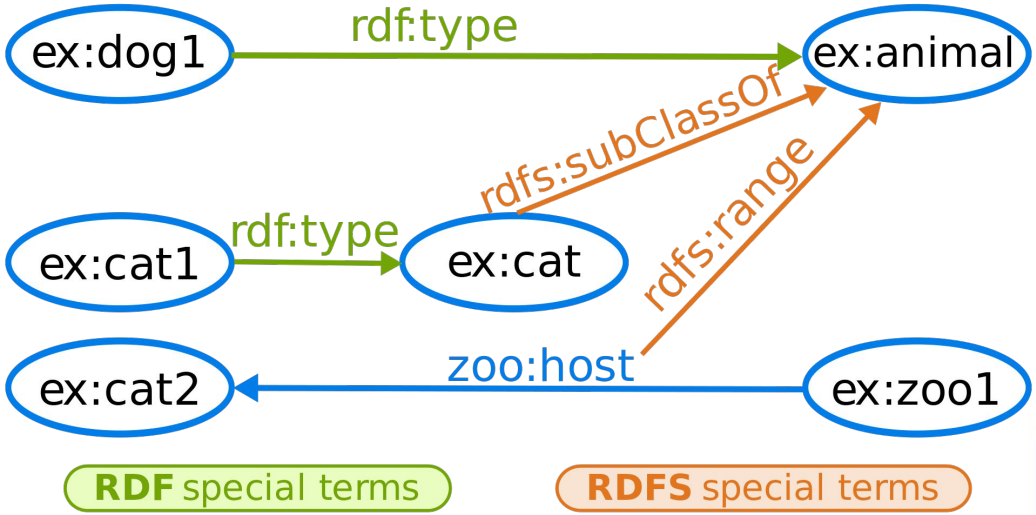
Query Result:

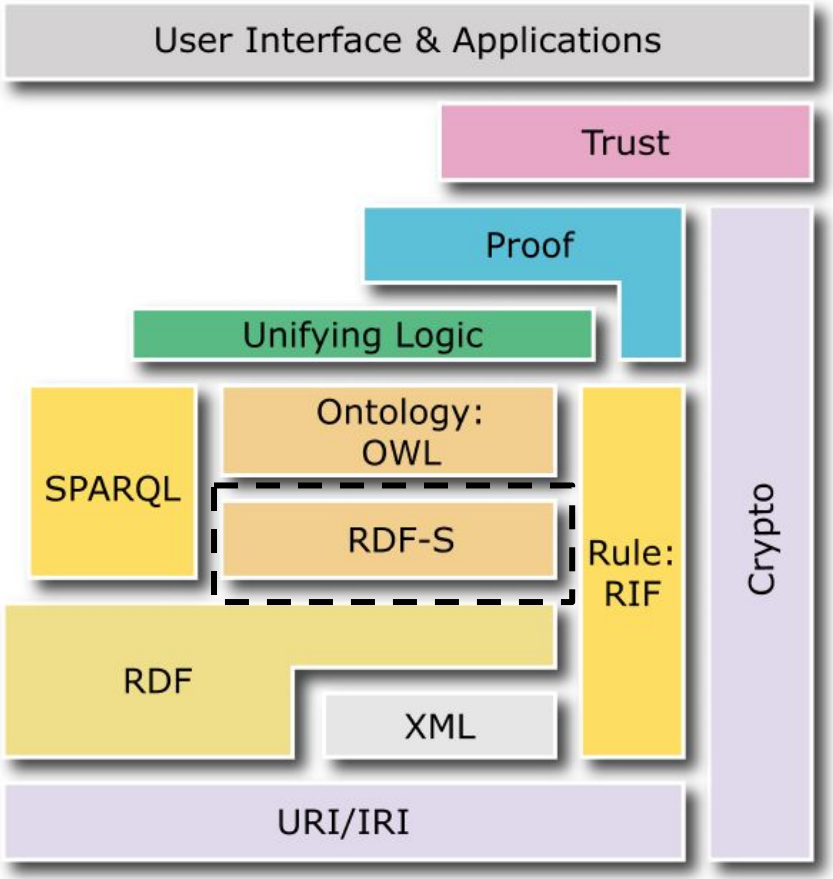
name	mbox
"Johnny Lee Outlaw"	<mailto:jlow@example.com>
"Peter Goodguy"	<mailto:peter@example.org>

https://query.wikidata.org

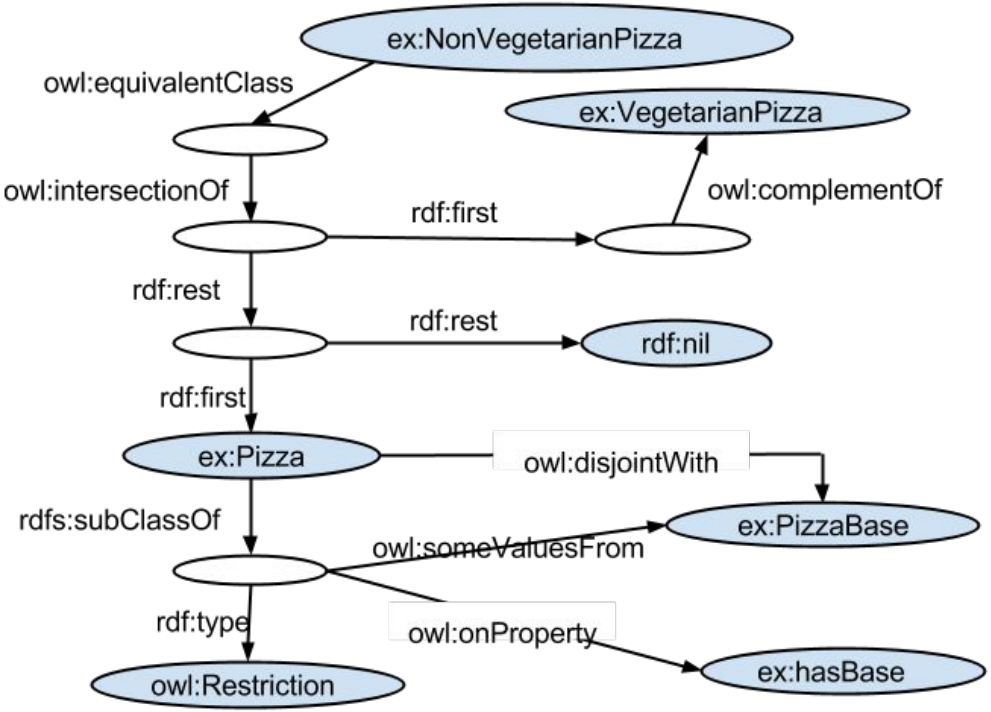


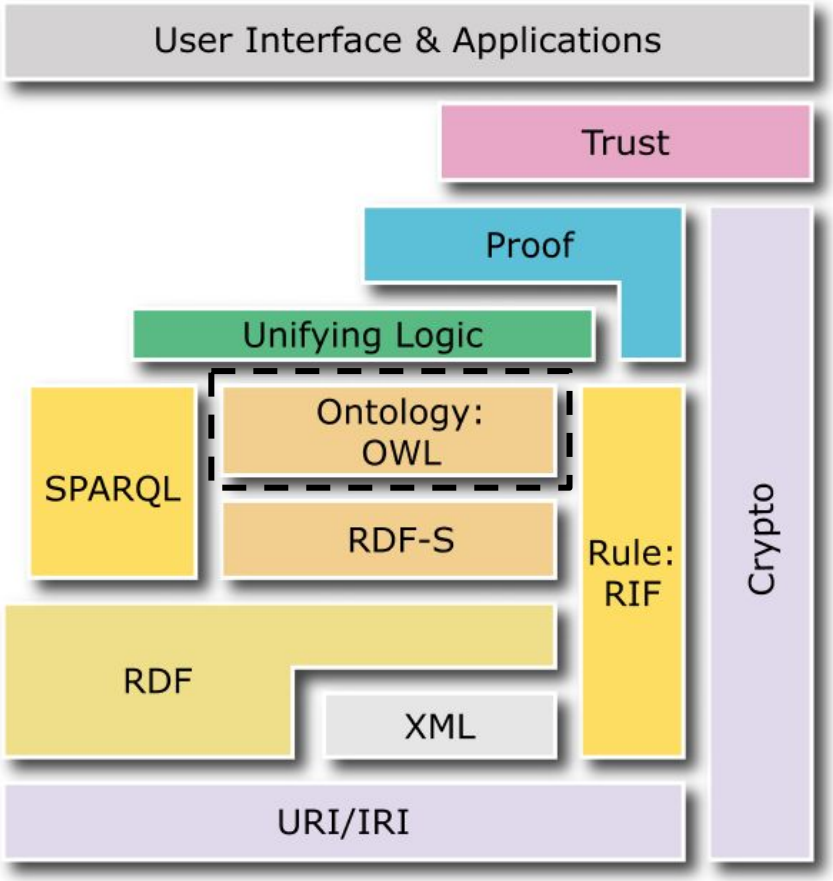
RDFs - Resource Description Framework Schema



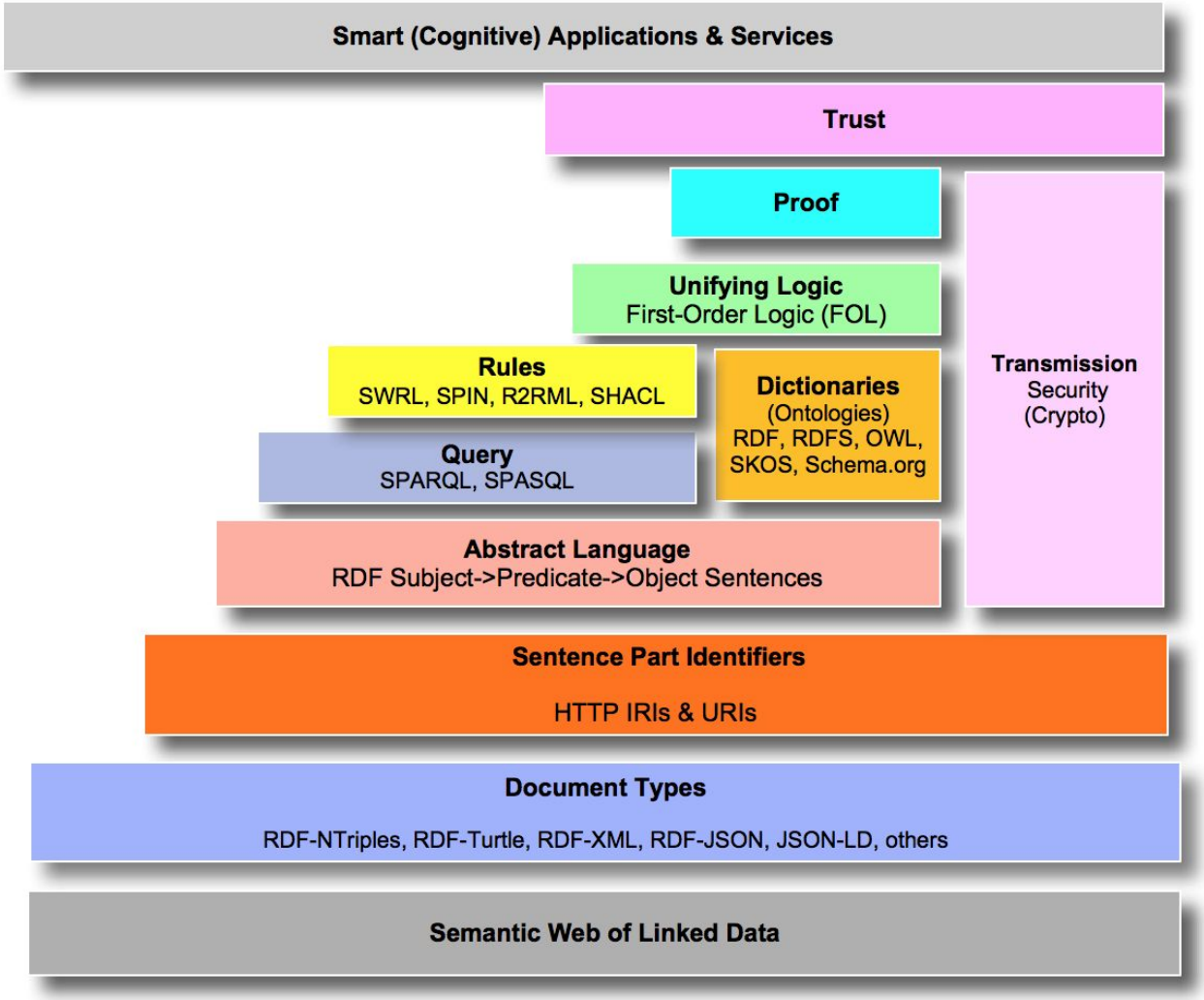


OWL - Web Ontology Language





Updated Web of Data Stack



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