

Οργάνωση Βάσεων Βιοϊατρικών Δεδομένων Εξόρυξη Γνώσης Βιοϊατρικών Δεδομένων

Σεμινάριο 9: Βάσεις NoSQL

**Ευάγγελος Καρκαλέτσης,
Αναστασία Κριθαρά, Γεώργιος Πετάσης**

Εργαστήριο Τεχνολογίας Γνώσεων & Λογισμικού,

Ινστιτούτο Πληροφορικής & Τηλεπικοινωνιών, Ε.Κ.Ε.Φ.Ε. “Δημόκριτος”

Τηλ.: 210-6503197, Fax: 210-6532175, {vangelis, akrihara, petasis}@iit.demokritos.gr

Ακαδημαϊκό Έτος: 2016 – 2017

Έχετε ακούσει αυτούς τους όρους;

- Semantic Web
- Linked Data
- NoSQL
- RDF
- Triple Stores
- SPARQL

Προκλήσεις στον σύγχρονο ιστό

Διασυνδεδεμένα δεδομένα

Big Data

Μειωμένη συνοχή
(relaxed consistency)

Ανοικτού Κώδικα

Τελικά συνεπείς
(eventually consistent)

Εύκολη αντιγραφή

Κατανεμημένες Βάσεις

Μη-σχεσιακές

Schema-free

Απλές Διεπαφές

Ημι-δομημένα δεδομένα

Παγκόσμιος Ιστός

- Τα αδόμητα δεδομένα (unstructured data) έχουν κατακλίσει τον παγκόσμιο ιστό
- Πώς μπορούμε να χειριστούμε καλύτερα αδόμητα δεδομένα;
 - Είναι οι σχεσιακές βάσεις λύση; Ίσως όχι...
 - Είναι τα μετα-δεδομένα λύση;
 - Τα μετα-δεδομένα (metadata) περιγράφουν συχνά το περιεχόμενο των αδόμητων δεδομένων
 - Χρειάζομαστε άλλες λύσεις;
 - Τεχνολογίες σαν τον «σημασιολογικό ιστό»;

Σημασιολογικός Ιστός

- Ο σημασιολογικός ιστός είναι μια Web 3.0 τεχνολογία
 - Ένας τρόπος διασύνδεσης δεδομένων μεταξύ συστημάτων
 - Οντότητες που επιτρέπουν πλούσιες, αυτό-περιγραφικές διασυνδέσεις δεδομένων παντού στον παγκόσμιο ιστό
- Αλλαγή τρόπου σκέψης:
 - Από έγγραφα που διαβάζονται από ανθρώπους, σε έγγραφα που διαβάζονται από μηχανές (machine readable documents)

Σημασιολογικός Ιστός

- Συνεργατικό κίνημα από έναν παγκόσμιο οργανισμό προτύπων, το World Wide Web Consortium



- Προωθεί πρότυπα δεδομένων στον παγκόσμιο ιστό
- Σκοπεύει να μετατρέψει τον παγκόσμιο ιστό (στον οποίο κυριαρχούν τα αδόμητα ή ημι-δομημένα δεδομένα) στον “web of data”

“The **Semantic Web** provides a common framework that allows **data** to be shared and reused across application, enterprise, and community boundaries.”

<http://www.w3.org/2001/sw/>

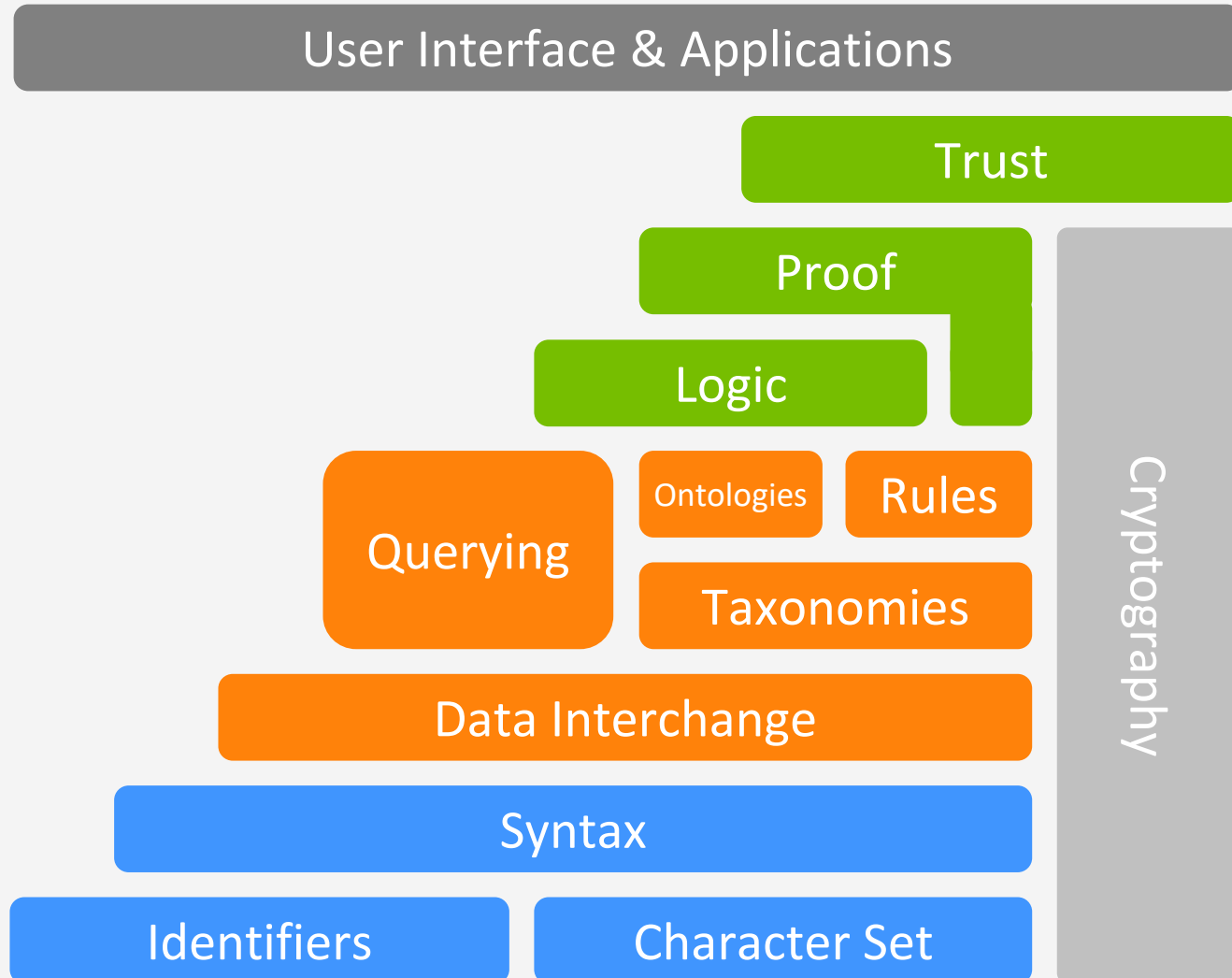
Ποιο πρόβλημα καλείται να λύσει;

- Τα δεδομένα στον παγκόσμιο ιστό είναι οργανωμένα σε **ιστοσελίδες**, έγγραφα σε HTML που συνδέονται μεταξύ τους με **υπερσυνδέσμους**.
 - Ένας άνθρωπος μπορεί να διαβάσει αυτές τις σελίδες.
 - Οι μηχανές όμως;
 - Μπορούν να εξάγουν λέξεις κλειδιά.
 - Πέρα από λέξεις κλειδιά;

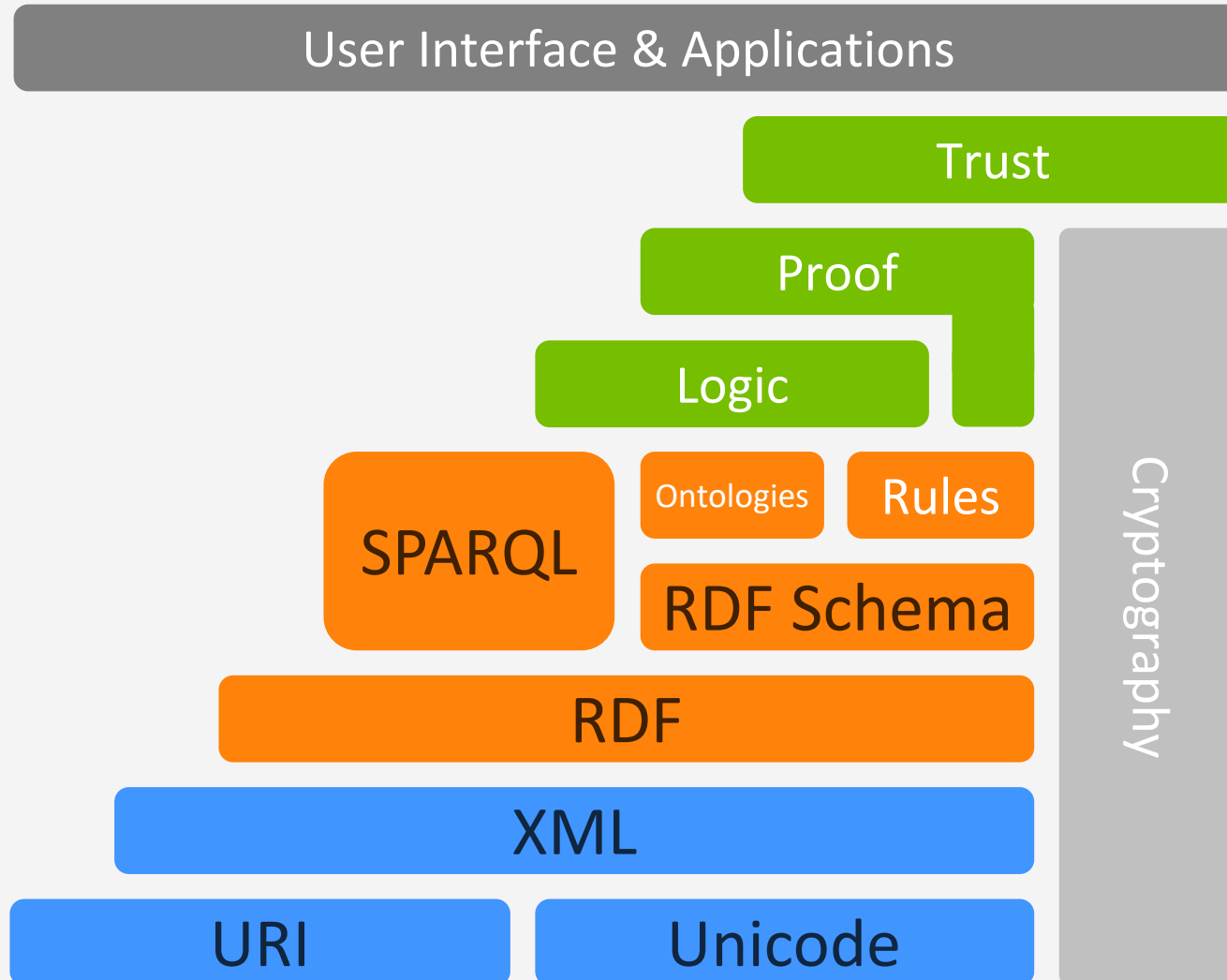
Ποιο πρόβλημα καλείται να λύσει;

- Η πλειοψηφία των ιστότοπων βασίζεται σε δεδομένα.
 - Αλλά τα δεδομένα *δεν είναι διαθέσιμα!*
 - Αυτό που είναι διαθέσιμο, είναι ιστοσελίδες που κατασκευάστηκαν από τα δεδομένα.
- Στόχοι σημασιολογικού ιστού:
 - Web of data: να κάνει τα δεδομένα διαθέσιμα σε μηχαναγνώσιμη μορφή
 - Να ενθαρρύνει οργανισμούς/εταιρίες να διαθέσουν τα δεδομένα τους σε κάποιο ανοικτό πρότυπο
 - Να ενθαρρύνει την επαναχρησιμοποίηση των δεδομένων

The semantic web stack!



The semantic web stack!



Resource Description Framework

- Σύνολο από πρότυπα από το W3C

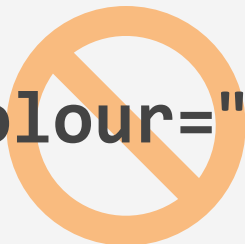
`http://www.w3c.org/RDF/`

- Κωδικοποιεί δηλώσεις (statements) για αντικείμενα (things)

Ο ουρανός έχει χρώμα γαλάζιο

- Η σύνταξη είναι σε XML

`<sky colour="blue"/>`



RDF: δηλώσεις για αντικείμενα

The sky has the colour blue

thing property value



triple

RDF: δηλώσεις για αντικείμενα

- Ένα απλό μοντέλο entity-attribute-value (EAV)
- Κάθε μοντέλο μπορεί να αναπαρασταθεί σαν triples

<entity>	<attribute-value>	<value>
E	A	V
<subject>	<predicate>	<object>

- Το RDF χρησιμοποιεί διαφορετική ορολογία

URIs στο RDF

- Παρέχονται χώροι ονομάτων (namespaces) για τον μονοσήμαντο προσδιορισμό των αντικειμένων στα οποία αναφερόμαστε
- Παρέχονται τρόποι να προσδιορίσουμε ιδιότητες και τύπους αντικειμένων
 - Με τρόπο κατανεμημένο και μοναδικό
- Καθένας μπορεί να πει οτιδήποτε για οποιοδήποτε αντικείμενο που προσδιορίζεται από ένα κατανεμημένο URI

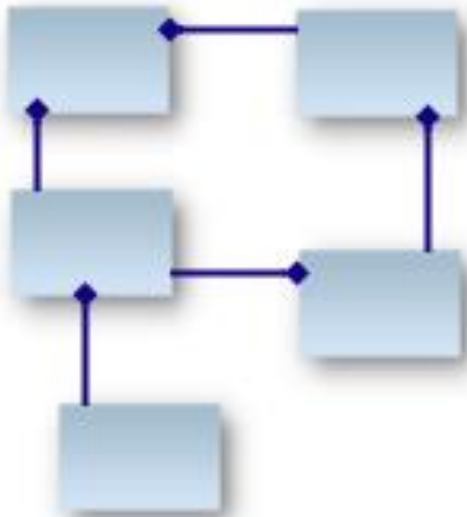
RDF: δηλώσεις για αντικείμενα

- Το υποκείμενο και το κατηγορημα είναι πάντα URIs
- Το αντικείμενο είναι είτε κυριολεκτική τιμή (literal) είτε URI

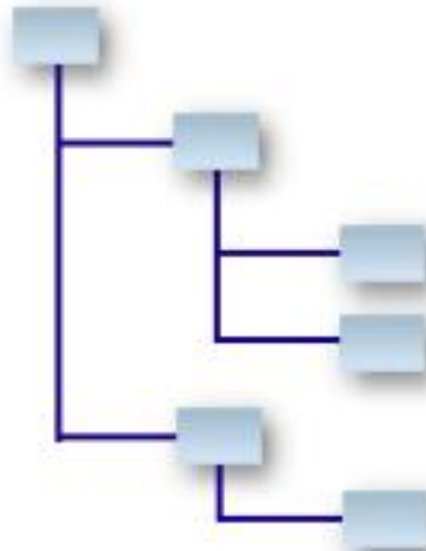
<code><http://iit.demokritos.gr/person/petasis_georgios></code>	subject
<code><http://iit.demokritos.gr/properties/name></code>	predicate
<code>"Georgios Petasis"</code>	object

<code><http://iit.demokritos.gr/person/petasis_georgios></code>	subject
<code><http://iit.demokritos.gr/properties/email></code>	predicate
<code>"petasis@iit.demokritos.gr"</code>	object

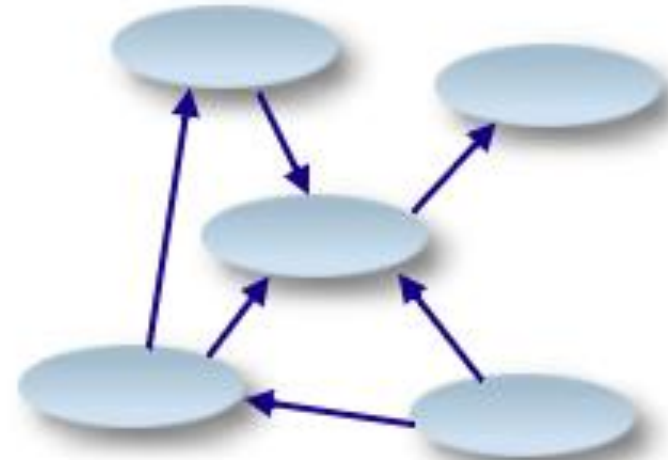
Graph Databases?



Relational DB
Tables Related By
Primary Key



Hierarchical DB
Parent Nodes Have More
Intrinsic Importance



Graph DB
Arbitrary Object Relations
No Intrinsic Importance

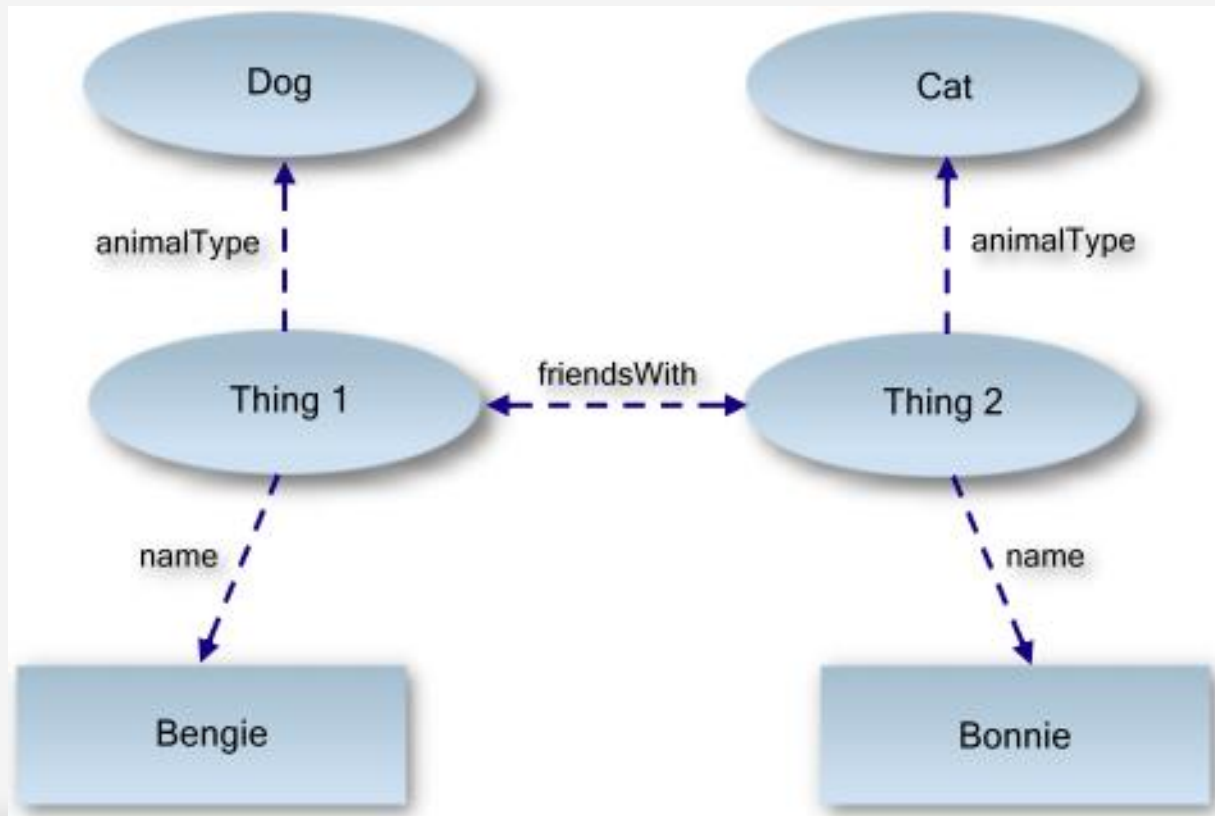
<http://www.linkeddatatools.com/introducing-rdf>

Graph Database: Παράδειγμα

Bengie is a dog.

Bonnie is a cat.

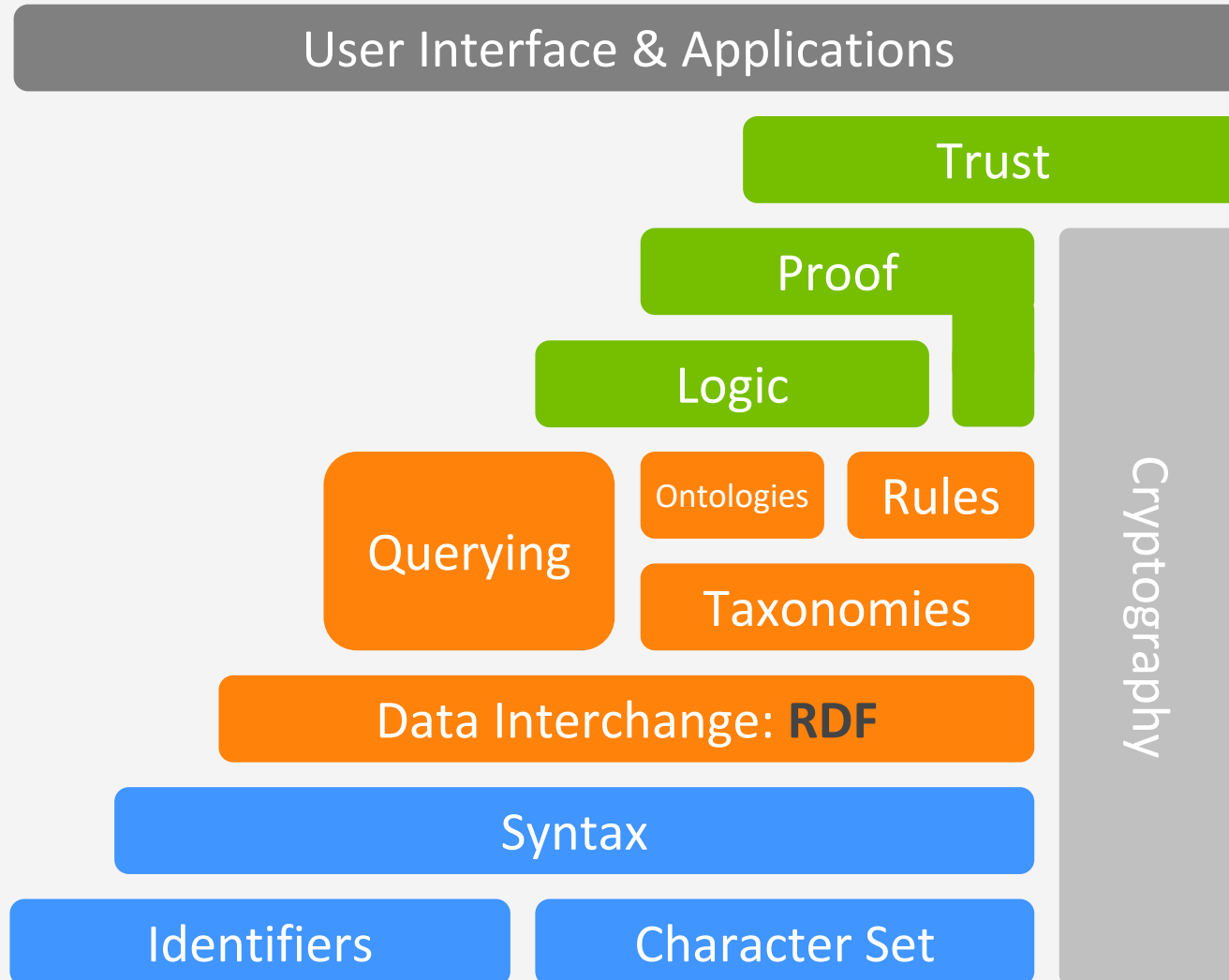
Bengie and Bonnie are friends.



Γράφοι RDF

- Ένας RDF γράφος είναι μια «λογική συλλογή» από triples
- Μια «αποθήκη» (store) μπορεί να περιέχει πολλούς γράφους
- Οι γράφοι μπορούν να έχουν όνομα (ένα URI)
 - Οι named RDF graphs ονομάζονται και context
- Λειτουργίες σε RDF γράφους
 - Εισαγωγή δεδομένων
 - Εξαγωγή του γράφου
 - Ερώτηση/ενημέρωση δεδομένων γράφου
 - Συγχώνευση δεδομένων από διαφορετικές πηγές
 - Έλεγχος πρόσβασης στα δεδομένα

The semantic web stack!



RDF Schema

- Το «core» RDF δεν έχει κάποιο «σχήμα»
 - Οποιοδήποτε «σχήμα» δεδομένων μπορεί να αναπαρασταθεί σε RDF
- Χρειαζόμαστε όμως ένα κοινό «λεξιλόγιο» ή «οντολογία»
 - Αν πρέπει να δημοσιεύουμε και να επαναχρησιμοποιούμε βάσης γνώσης

RDF Schema

- Σύνολο από ιδιότητες RDF για τον ορισμό τύπων και περιορισμών
- Τα σχήματα RDF αναπαριστώνται σε RDF

RDF Schema

- Ένα κοινό λεξιλόγιο για τύπους, και περιορισμούς σε τιμές τύπων

prefix:rdf <<http://www.w3c.org/1999/02/22-rdf-syntax-ns#>>

prefix:rdfs <<http://www.w3.org/2000/01/rdf-schema#>>

rdf:type

rdfs:domain

rdfs:range

rdfs:class

rdfs:property

xsd:integer

Ένα έγγραφο RDF σε XML

- Υπάρχει υποχρεωτικά ο κόμβος ρίζα:

```
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">
  <!-- Body Code Omitted -->
</rdf:RDF>
```

Προσθέτοντας μια δήλωση

- Μπορούμε να βάλουμε ένα ή περισσότερα κατηγορήματα (“statements”):

```
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">

  <rdf:Description
    rdf:about="http://www.test.com/clothes#t-shirt">
    <!-- Statement Code Omitted -->
  </rdf:Description>

</rdf:RDF>
```

Προσθέτοντας μια δήλωση

- Predicates:

```
<rdf:RDF
```

```
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">
```

```
    <rdf:Description
```

```
      rdf:about="http://www.test.com/clothes#t-shirt">
```

```
        <feature:size>12</feature:size>
```

```
        <feature:color
```

```
          rdf:resource="http://www.test.com/colors#white" />
```

```
    </rdf:Description>
```

```
</rdf:RDF>
```


Κατηγορία RDF

```
<rdf:Description rdf:about="subject">  
  <predicate rdf:resource="object" />  
  <predicate>literal value</predicate>  
</rdf:Description>
```

- **Subject** of the statement
- **Predicates** of the statement - including whether they are *resources* or *literals*
- **Objects** referenced by the resource predicates

Παράδειγμα

```
<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:dc="http://purl.org/dc/elements/1.1/"
  xmlns:region="http://www.country-regions.fake/">
  <rdf:Description rdf:about="http://en.wikipedia.org/wiki/Oxford">
    <dc:title>Oxford</dc:title>
    <dc:coverage>Oxfordshire</dc:coverage>
    <dc:publisher>Wikipedia</dc:publisher>
    <region:population>10000</region:population>
    <region:principaltown
      rdf:resource="http://www.country-regions.fake/oxford"/>
  </rdf:Description>
</rdf:RDF>
```

- Εκτενής χρήση των χώρων ονομάτων (namespaces)!

Turtle

```
@prefix :      <http://example.org/> .  
@prefix rdf:   <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .  
@prefix foaf:  <http://xmlns.com/foaf/0.1/> .
```

:Andy

```
  a foaf:Person ;  
  foaf:name "Andy Seaborne" ;  
  foaf:mbox <mailto:andy@apache.org> ;  
  foaf:knows :Paolo ;  
  foaf:knows :Steve .
```

:Paolo

```
  a foaf:Person ;  
  foaf:name "Paolo Castagna" ;  
  foaf:mbox <mailto:castagna@apache.org> .
```

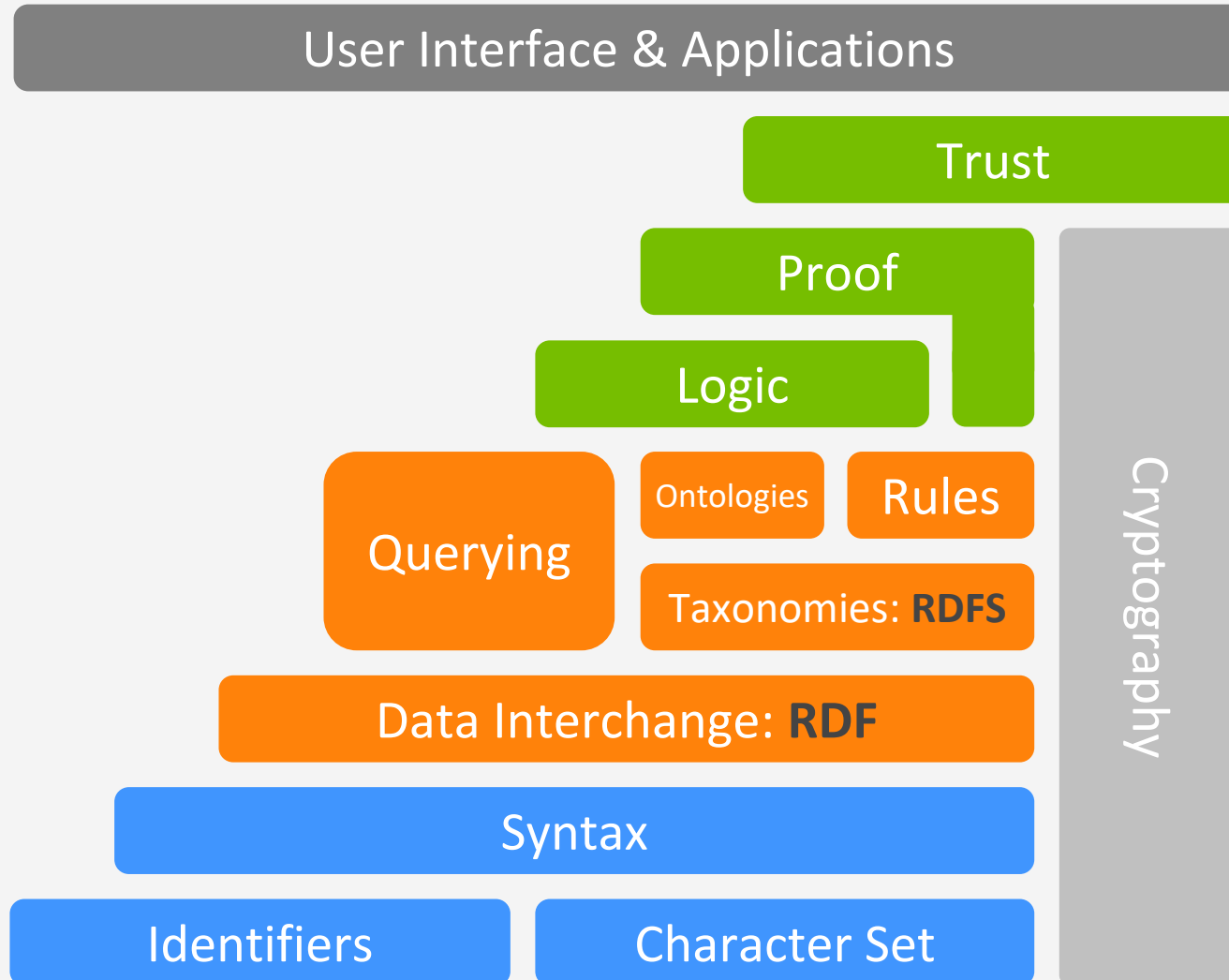
:Steve

```
  foaf:name "Steve Loughran" ;  
  foaf:mbox_sha1 "0678d36518d039a64ee4baba0a568afe535ce5f3" .
```

RDF Schema → RDFS

- Το RDFS μας δίνει ένα κοινό «λεξιλόγιο» για να φτιάχνουμε αντικείμενα και ταξινομίες
 - Όρους με καλά καθορισμένη σημασία, που είναι συνεπής (consistent) ανεξάρτητα από το περιβάλλον
 - Ένα πρώτο βήμα για να προσθέσουμε **σημασία** (semantics) στο μοντέλο γράφου που ορίζει το RDF
- Το επόμενο στάδιο: **οντολογίες**
 - Έχοντας ένα λεξιλόγιο, μια οντολογία επιτρέπει τον ορισμό **σχέσεων** που εξαρτώνται από τα **συμφραζόμενα/περιβάλλον** (*contextual relationships*)

The semantic web stack!



Τι είναι μια οντολογία;

- “An ontology is an **explicit specification** of a **conceptualization**” [Gruber, 1993]
 - Ένα μοντέλο μιας θεματικής περιοχής, π.χ. μια γενική αναπαράσταση γνώσης για μια θεματική περιοχή
 - Βασικά έννοιες/κλάσεις και (δυαδικές) σχέσεις
- Για **ποιο λόγο** έχει φτιαχτεί μια οντολογία;
 - Συνήθως για σκοπούς **διαμοιρασμού γνώσης** (knowledge sharing) και **επαναχρησιμοποίησης** (reuse)

Τι είναι μια οντολογία;

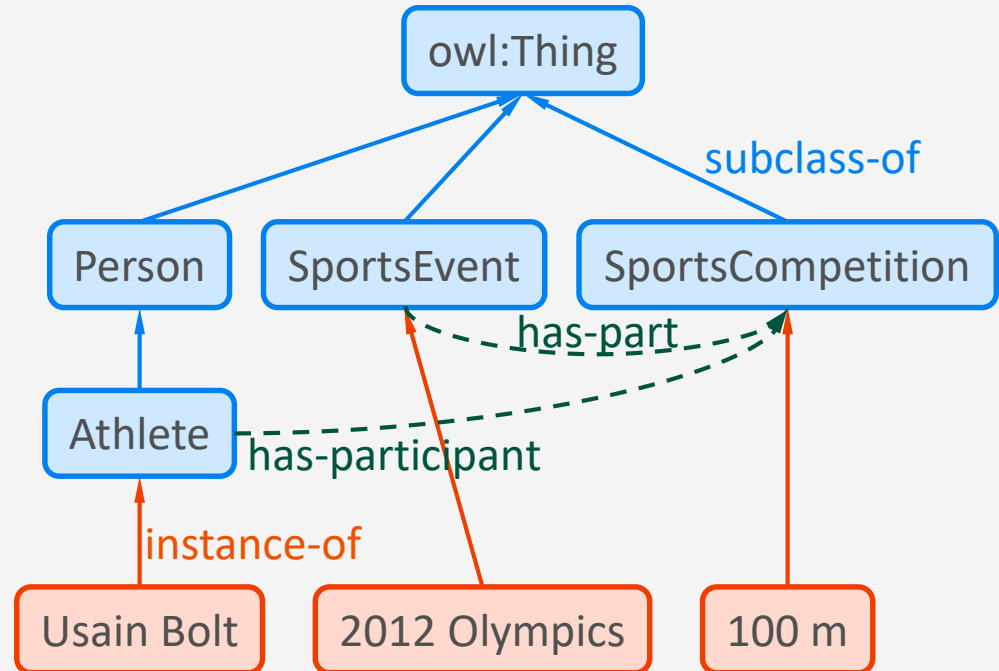
Αξιώματα (axioms)

$Athlete \sqsubseteq Person$

$Person \sqcap SportsEvent = \perp$

$Person \sqcap SportsCompetition = \perp$

$SportsEvent \sqcap SportsCompetition = \perp$



Κανόνες (rules)

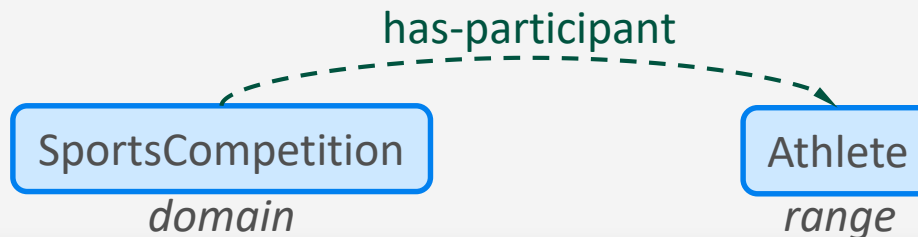
$has_participant(x, y) \wedge has_participant(z, y) \rightarrow compete(x, z)$

Τι είναι μια οντολογία;

- Κύρια συστατικά μιας οντολογίας
 - Ιεραρχία κλάσεων/εννοιών (classes/concepts)
 - Κλάσεις/έννοιες
 - Σχέσεις «is-a» (subclass-of)
 - Ιδιότητες τύπων (datatype properties - attributes)



- Ιδιότητες αντικειμένων (object properties - relations)



Web Ontology Language

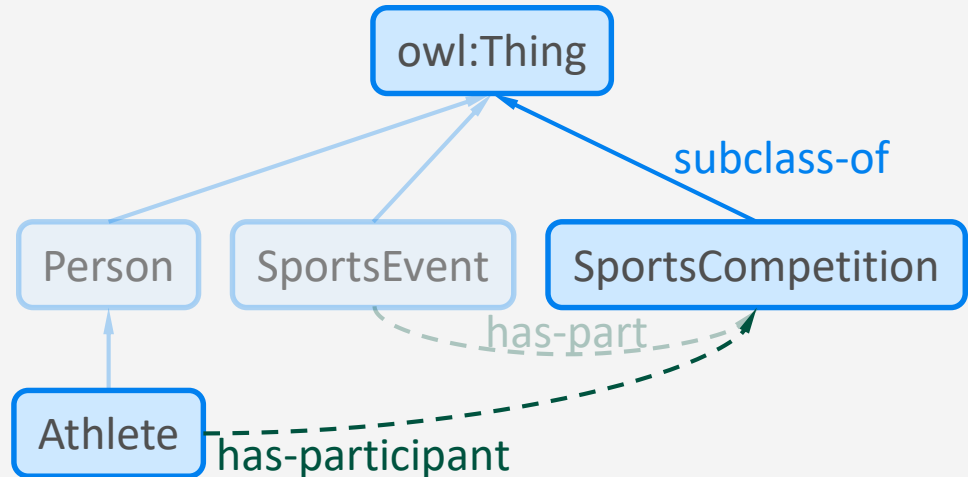
- Συστάσεις του W3C
 - Resource Description Framework (RDF)
 - URIs για την ονομασία σχέσεων μεταξύ 2 αντικειμένων (“triples”)
 - OWL 2 Web Ontology Language
 - Διάφορες υπο-γλώσσες, με διαφορετική εκφραστικότητα (και πολυπλοκότητα συλλογισμού - reasoning complexity)
 - OWL Lite (hierarchies and simple constraints)
 - OWL DL (maximum expressiveness retaining computational completeness & decidability – corresponds to description logic)
 - OWL Full (more compatible with RDF Schema)
 - Διάφορες μορφές
 - RDF/XML, OWL/XML, Functional syntax, Manchester syntax, ...

Web Ontology Language

OWL (RDF/XML syntax)

```
<owl:Class rdf:ID="SportsCompetition">  
  <rdfs:subClassOf rdf:resource="owl:Thing"/>  
</owl:Class>
```

```
<owl:ObjectProperty rdf:ID="has-participant">  
  <rdfs:range rdf:resource="Athlete" />  
</owl:ObjectProperty>
```



OWL (Manchester syntax)

Class: SportsCompetition
SubClassOf: owl:Thing
ObjectProperty: has-participant
Range: Athlete

Description Logic

$\text{SportsCompetition} \sqsubseteq \text{owl:Thing}$

$\text{SportsCompetition} \sqsubseteq \exists_{\leq 1} \text{has-participant. Athlete}$

Πρωτοβουλίες Μεταδεδομένων

- Υπάρχουν ήδη αρκετές έτοιμες οντολογίες, με τυποποιημένο λεξιλόγιο για αρκετές θεματικές περιοχές, π.χ.:
- [Dublin Core Metadata Initiative \(DCMI\)](#)
 - Creates ontologies for a range of subjects, particularly focusing on common, every day terms and terms important in media.
- [Friend Of A Friend \(FOAF\)](#)
 - Focuses on developing a standard vocabulary/ontology for social networking purposes.
- [OpenCyc](#)
 - An ontology of everyday, common sense terms.

Παράδειγμα OWL οντολογίας

```
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xmlns:owl="http://www.w3.org/2002/07/owl#"
  xmlns:dc="http://purl.org/dc/elements/1.1/">

  <!-- OWL Header Example -->
  <owl:Ontology rdf:about="http://www.test.com/plants">
    <dc:title>Example Plant Ontology</dc:title>
    <dc:description>An example ontology</dc:description>
  </owl:Ontology>

  <!-- OWL Class Definition Example -->
  <owl:Class rdf:about="http://www.test.com/plants#planttype">
    <rdfs:label>The plant type</rdfs:label>
    <rdfs:comment>The class of plant types.</rdfs:comment>
  </owl:Class>
</rdf:RDF>
```

OWL Classes, Subclasses & Individuals

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xmlns:owl="http://www.w3.org/2002/07/owl#"
  xmlns:dc="http://purl.org/dc/elements/1.1/"
  xmlns:plants="http://www.test.com/plants#">
  <!-- OWL Header Omitted For Brevity -->
  <!-- OWL Class Definition - Plant Type -->
  <owl:Class rdf:about="http://www.test.com/plants#planttype">
    <rdfs:label>The plant type</rdfs:label>
    <rdfs:comment>The class of all plant types.</rdfs:comment>
  </owl:Class>
  <!-- OWL Subclass Definition - Flower -->
  <owl:Class rdf:about="http://www.test.com/plants#flowers">
    <!-- Flowers is a subclassification of planttype -->
    <rdfs:subClassOf rdf:resource="http://www.test.com/plants#planttype"/>
    <rdfs:label>Flowering plants</rdfs:label>
    <rdfs:comment>Flowering plants, also known as angiosperms.</rdfs:comment>
  </owl:Class>
  <!-- OWL Subclass Definition - Shrub -->
  <owl:Class rdf:about="http://www.test.com/plants#shrubs">
    <!-- Shrubs is a subclassification of planttype -->
    <rdfs:subClassOf rdf:resource="http://www.test.com/plants#planttype"/>
    <rdfs:label>Shrubbery</rdfs:label>
    <rdfs:comment>Shrubs, a type of plant which branches from the base.</rdfs:comment>
  </owl:Class>
  <!-- Individual (Instance) Example RDF Statement -->
  <rdf:Description rdf:about="http://www.test.com/plants#magnolia">
    <!-- Magnolia is a type (instance) of the flowers classification -->
    <rdfs:type rdf:resource="http://www.test.com/plants#flowers"/>
  </rdf:Description>
</rdf:RDF>
```

Data type property

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xmlns:owl="http://www.w3.org/2002/07/owl#"
  xmlns:dc="http://purl.org/dc/elements/1.1/"
  xmlns:plants="http://www.test.com/plants#">
  <!-- OWL Header Omitted For Brevity -->
  <!-- OWL Classes Omitted For Brevity -->
  <!-- Define the family property -->
  <owl:DatatypeProperty rdf:about="http://www.test.com/plants#family"/>
    <rdf:Description rdf:about="http://www.test.com/plants#magnolia">
      <!-- Magnolia is a type (instance) of the flowers class -->
      <rdf:type rdf:resource="http://www.test.com/plants#flowers"/>
      <!-- The magnolia is part of the 'Magnoliaceae' family -->
      <plants:family>Magnoliaceae</plants:family>
    </rdf:Description>
  </rdf:RDF>
```

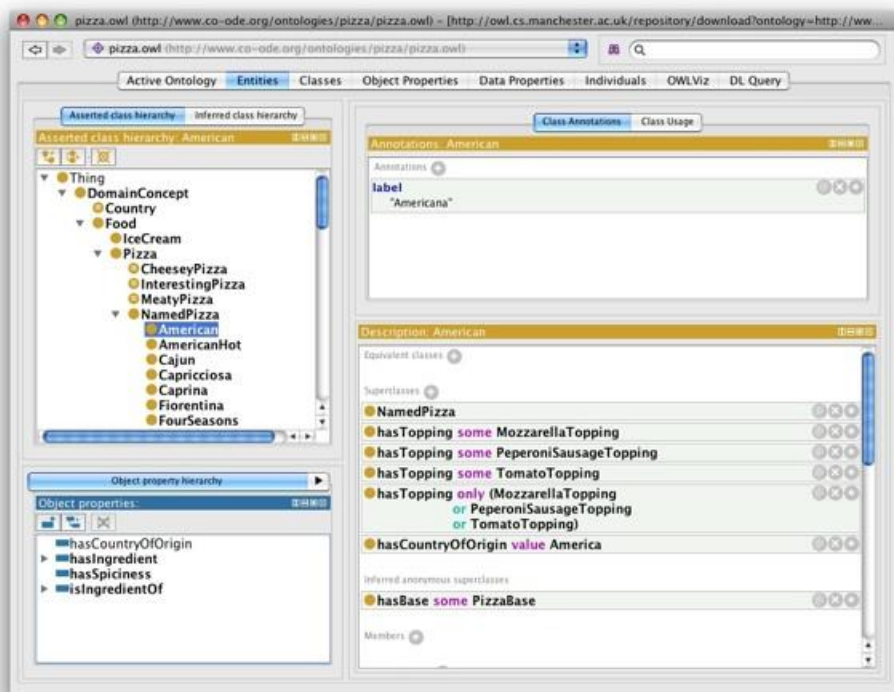
- Προσθέσαμε μια ιδιότητα σε ένα στιγμιότυπο (instance), και όχι σε μια κλάση!

Object Property

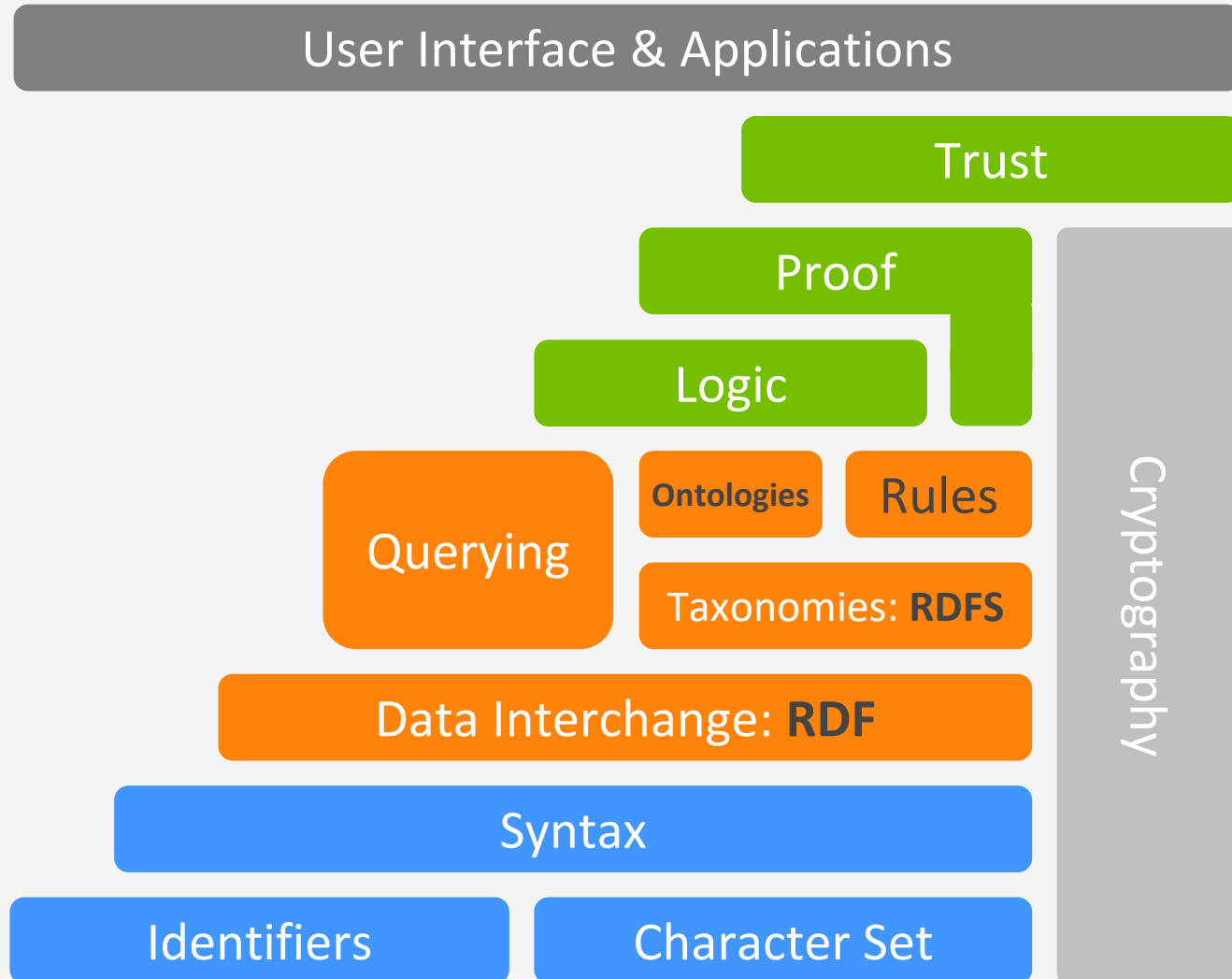
```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:owl="http://www.w3.org/2002/07/owl#"
  xmlns:dc="http://purl.org/dc/elements/1.1/"
  xmlns:plants="http://www.test.com/plants#">
  <!-- OWL Header Omitted For Brevity -->
  <!-- OWL Classes Omitted For Brevity -->
  <!-- Define the family property -->
  <owl:DatatypeProperty rdf:about="http://www.test.com/plants#family"/>
  <!-- Define the similarlyPopularTo property -->
  <owl:ObjectProperty rdf:about="http://www.test.com/plants#similarlyPopularTo"/>
  <!-- Define the Orchid class instance -->
  <rdf:Description rdf:about="http://www.test.com/plants#orchid">
    <!-- Orchid is an individual (instance) of the flowers class -->
    <rdf:type rdf:resource="http://www.test.com/plants#flowers"/>
    <!-- The orchid is part of the 'Orchidaceae' family -->
    <plants:family>Orchidaceae</plants:family>
    <!-- The orchid is similarly popular to the magnolia -->
    <plants:similarlyPopularTo rdf:resource="http://www.test.com/plants#magnolia"/>
  </rdf:Description>
  <!-- Define the Magnolia class instance -->
  <rdf:Description rdf:about="http://www.test.com/plants#magnolia">
    <!-- Magnolia is an individual (instance) of the flowers class -->
    <rdf:type rdf:resource="http://www.test.com/plants#flowers"/>
    <!-- The magnolia is part of the 'Magnoliaceae' family -->
    <plants:family>Magnoliaceae</plants:family>
    <!-- The magnolia is similarly popular to the orchid -->
    <plants:similarlyPopularTo rdf:resource="http://www.test.com/plants#orchid"/>
  </rdf:Description>
</rdf:RDF>
```

Διαχείριση Οντολογιών

- Υπάρχουν αρκετά εργαλεία για την επεξεργασία οντολογιών
- Το protégé είναι ένα από αυτά: <http://protege.stanford.edu/>



The semantic web stack!



Ανάκτηση: SPARQL

SPARQL Protocol and RDF Query Language

- Προφέρεται “sparkle”
- Επίσημο W3C recommendation από το 2008
- Επιτρέπει ερωτήματα που απαρτίζονται από:
 - triple patterns,
 - conjunctions,
 - disjunctions, and
 - optional patterns

Απλά παραδείγματα

- Επιστρέφει όλα τα triples από μια βάση:

```
select * where {  
  ?a ?b ?c  
}
```

- Επιστρέφει όλα τα λουλούδια:

```
PREFIX plant:    <http://www.test.com/plants>  
FROM <http://www.test.com/plantsdata/plants.rdf>  
SELECT ?name WHERE {  
  ?name rdf:type plant:flowers  
}  
ORDER BY ?name
```

Γενική δομή ερωτημάτων ανάκτησης

- PREFIX (Namespace Prefixes)
 - π.χ. "PREFIX plant: <http://www.test.com/plants>"
- SELECT (Result Set)
 - π.χ. "SELECT ?name"
- FROM (Data Set)
 - π.χ. "FROM <http://www.test.com/plantsdata/plants.rdf>"
- WHERE (Query Triple Pattern)
 - π.χ. "WHERE ?name rdf:type plant:flowers"
- ORDER BY, DISTINCT, κλπ. (Modifiers)
 - π.χ. "ORDER BY ?name"

Παραδείγματα

- Επιλέγει όλα τα URIs φυτών (subjects) και τα ονόματα οικογενειών:

```
PREFIX plants:    <http://www.test.com/plants>
SELECT * WHERE
{
    ?name plants:family ?family
}
```

Έξοδος σε XML

```
<?xml version="1.0" ?>
<sparql xmlns="http://www.w3.org/2005/sparql-results#"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <head>
    <variable name="name"/>
    <variable name="family"/>
  </head>
  <results>
    <result>
      <binding name="name">
        <uri>http://www.test.com/plants#magnolia</uri>
      </binding>
      <binding name="family">
        <literal>Magnoliaceae</literal>
      </binding>
    </result>
    <result>
      <binding name="name">
        <uri>http://www.test.com/plants#african_lilly</uri>
      </binding>
      <binding name="family">
        <literal>Liliaceae</literal>
      </binding>
    </result>
  </results>
</sparql>
```

Παραδείγματα

- Αν ήθελα τα φυτά μιας συγκεκριμένης οικογένειας;
PREFIX plants: <http://www.test.com/plants>
SELECT * WHERE
{
 ?name plants:family "Magnoliaceae"
}

Έξοδος σε XML

```
<?xml version="1.0" ?>
<sparql xmlns="http://www.w3.org/2005/sparql-results#"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <head>
    <variable name="name"/>
  </head>
  <results>
    <result>
      <binding name="name">
        <uri>http://www.test.com/plants#magnolia</uri>
      </binding>
    </result>
    <result>
      <binding name="name">
        <uri>http://www.test.com/plants#manglietia</uri>
      </binding>
    </result>
  </results>
</sparql>
```


Άλλα παραδείγματα

- Επιστρέφει όλα τα πρόσωπα που έχουν ηλικία, σορταρισμένα ως προς την ηλικία:

```
select ?a ?age where {  
  ?a rdf:type ont:person .  
  ?a ont:age ?age  
}  
order by ?age
```

- Επιστρέφει όλα τα πρόσωπα πάνω από 18:

```
select ?a where {  
  ?a rdf:type ont:person .  
  ?a ont:age ?age .  
  filter (?age > 18)  
}
```

CREATE, INSERT, UPDATE?

- Εδώ περιπλέκονται τα πράγματα
 - Δεν περιλαμβάνονται στο αρχικό πρότυπο
 - Υπάρχουν επεκτάσεις όπως το [SPARUL \(SPARQL/Update\)](#)
 - Το επίπεδο υλοποίησης διαφέρει ανάμεσα στις υλοποιήσεις

CREATE, INSERT, UPDATE

- Αλλαγή όλων των “Bill” σε “William”:

FOAF Vocabulary Specification 0.98

prefix: **foaf** <http://xmlns.com/foaf/0.1/>

```
delete {  
    ?person foaf:givenName “Bill”  
}  
insert {  
    ?person foaf:givenName “William”  
}  
where {  
    ?person foaf:givenName “Bill”  
}
```

CREATE, INSERT, UPDATE

- Διαγραφή παλιών εγγραφών:

Dublin Core Metadata Initiative

prefix:dcmi <http://purl.org/dc/elements/1.1/>

prefix:xsd <http://www.w3.org/2001/XMLSchema#>

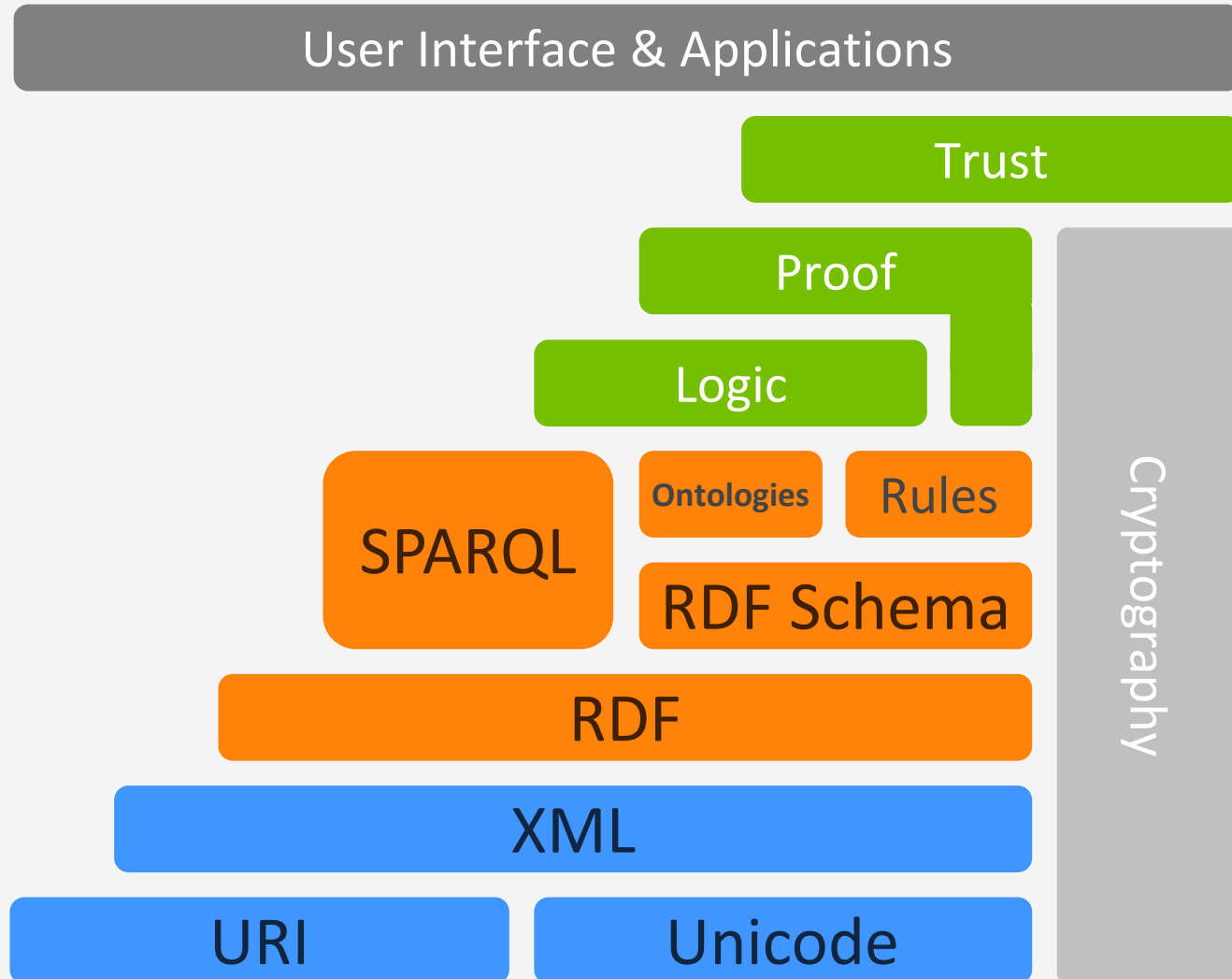
```
delete {  
    ?book ?p ?v  
}
```

```
where {  
    ?book dcmi:date ?date .  
    filter ( ?date < "1970-01-01T00:00-02:00"^^xsd:dateTime )  
    ?book ?p ?v  
}
```

Περισσότερα για το SPARQL

<http://www.w3.org/TR/rdf-sparql-query/>

The semantic web stack!



Ανακεφαλαίωση

- Οι βασικοί όροι του σημασιολογικού ιστού
 - Είναι ένας τεράστιος γράφος
 - Ορίζεται μέσω του RDF
- Το RDF περιγράφει γράφους μέσω τριπλετών
 - Κάθε τριπλέτα έχει υποκείμενο, κατηγορημα, και αντικείμενο
 - Το υποκείμενο είναι πάντα ένα URI
 - Το αντικείμενο είτε μια τιμή (literal) είτε ένας πόρος (resource) – πάλι URI

Ανακεφαλαίωση

- Ορίσαμε τους όρους λεξιλόγιο και οντολογία, που προσδίδουν σημασιολογία στο μοντέλο δεδομένων
- Σημασία των πρωτοβουλιών μεταδεδομένων για την προτυποποίηση και την ανταλλαγή δεδομένων
- Ανάκτηση δεδομένων με SPARQL

Triple Stores

- Triple Stores: αποθήκες τριπλετών
 - Μόνιμη αποθήκευση δεδομένων σε RDF
 - Η δομή δεδομένων είναι τριπλέτες και γράφοι
 - Είναι συνήθως transactional
 - Δεν αποθηκεύουν τριπλέτες
 - Αποθηκεύουν ονοματισμένα σύνολα από τριπλέτες!
 - Δεν έχουν σχήμα (schema-less)
 - Υποστηρίζουν ερωτήματα SPARQL, και μέσω HTTP

Τρεις κύριες κατηγορίες

In-Memory Stores

Transient storage of triples in memory

Native Stores

Persistent storage of triples in native stores with their own storage implementation

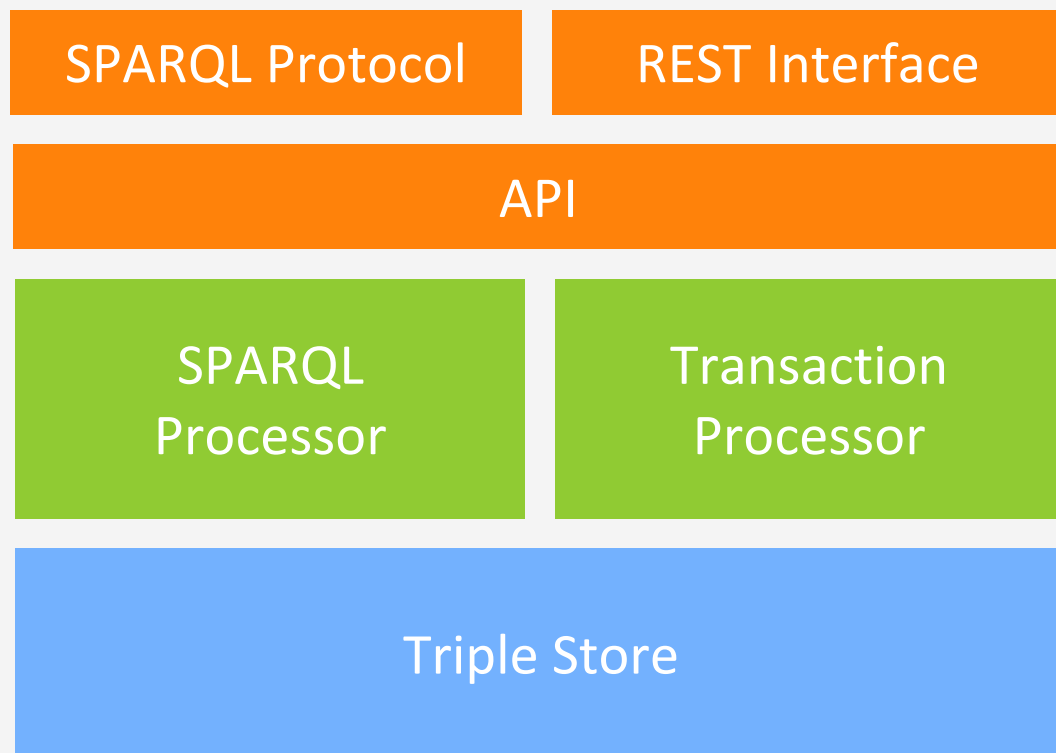
Non-Native Stores

Persistent storage of triples on top of third-party databases

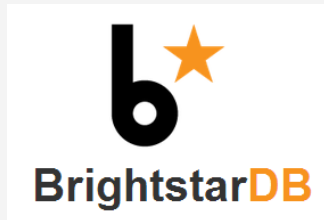
Τυπική Αρχιτεκτονική

Triple Store Operations

- Insert/delete
- Batch insert/delete
- Export graph
- SPARQL query
- SPARQL update



Triple Stores



Sesame

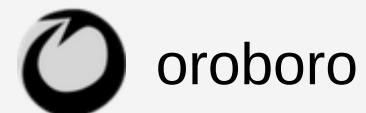
4store



Capsenta



AllegroGraph



Διαφορές με Σχεσιακές Βάσεις

- Δεν χρειάζεται η δημιουργία σχημάτων
- Δεν χρειάζεται η σύνδεση πινάκων, αφού παρέχονται σχέσεις από ένα σε πολλά εγγενώς
- Μπορούν να δημιουργηθούν νέα κατηγορήματα (predicates), που αντιστοιχούν σε στήλες, ανά πάσα στιγμή
- Υποστηρίζουν ανάκτηση δεδομένων
- Είναι πιο ευέλικτα και υποστηρίζουν πιο σύνθετα ερωτήματα

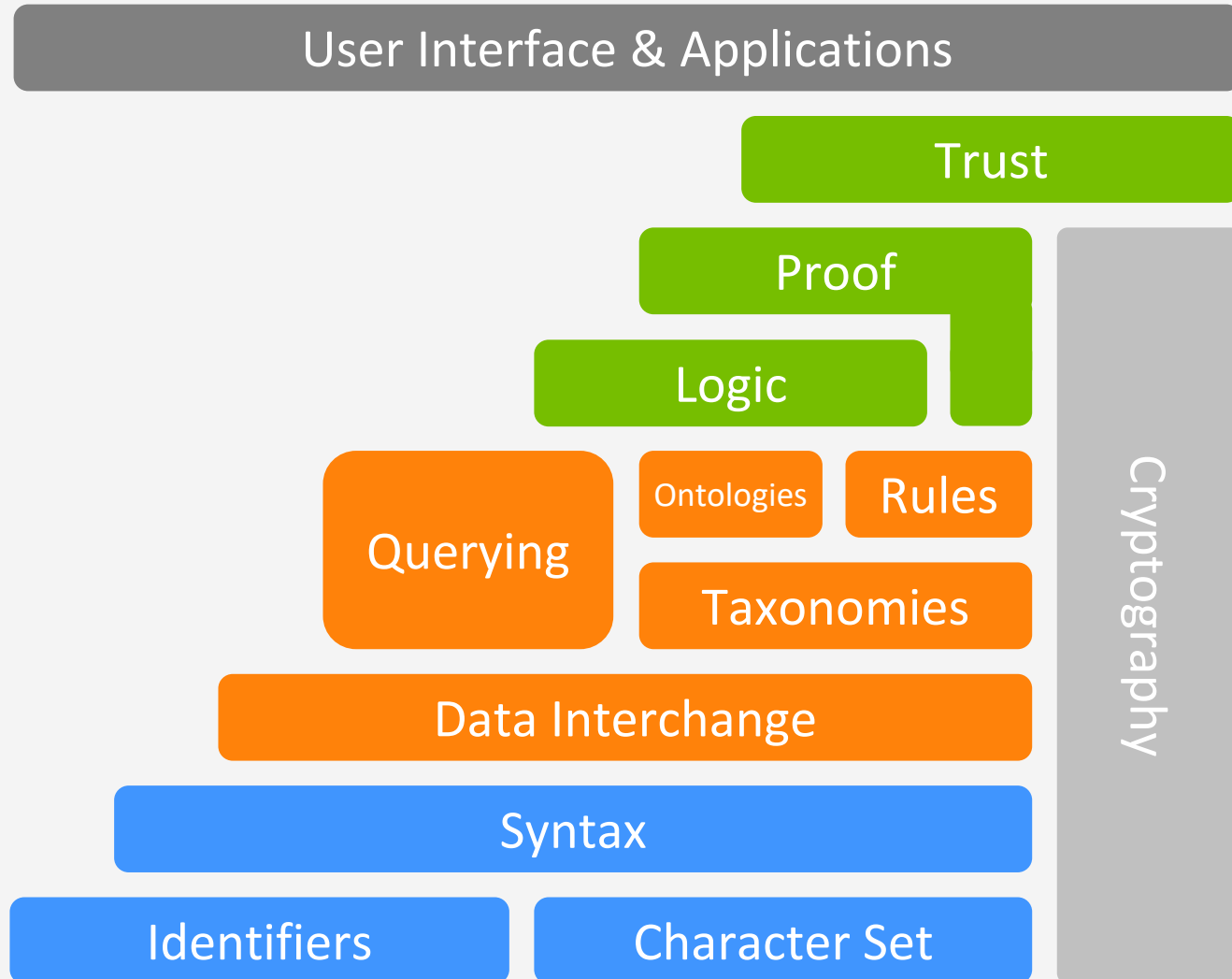
NoSQL

- Key-Value stores
- Column Stores
- Document Stores
- Graph Databases
- Triples Stores

NoSQL

- Non-relational
- Distributed
- Open-source
- Horizontally scalable
- Schema-free
- Easy replication support
- Simple API
- Eventually consistent / BASE (not ACID)

The semantic web stack!



References

- Dr. Stephan Volmer: Triple Stores
<http://www.slideshare.net/svolmer/triple-stores>
- [Introducing Linked Data And The Semantic Web](#)