



# Σημασιολογικός και Κοινωνικός Ιστός

## Διάλεξη 06 - RDF

Γεώργιος Δημητρακόπουλος  
dimitrakopoulos@ionio.gr

# RDF

- ▶ Resource Description Framework
- ▶ Το RDF είναι ένα πρότυπο του World Wide Web Consortium (W3C) που σχεδιάστηκε αρχικά ως μοντέλο δεδομένων για μεταδεδομένα. Έχει καταλήξει να χρησιμοποιείται ως μια γενική μέθοδος για την περιγραφή και την ανταλλαγή δεδομένων γραφήματος.

# RDF

- ▶ Είναι η επίσημη γλώσσα για την περιγραφή δομημένων πληροφοριών.
- ▶ Ο πρωταρχικός στόχος του RDF είναι να ανταλλάσσει δεδομένα στον Ιστό, διατηρώντας παράλληλα το αρχικό νόημα των δεδομένων. Είναι ένα μοντέλο δεδομένων που χρησιμοποιείται για την περιγραφή πόρων. Για παράδειγμα φυσικά πράγματα, αφηρημένες έννοιες, αριθμοί και κείμενο. Το RDF επιτρέπει την επεξεργασία πληροφοριών.
- ▶ Η αναπαράσταση RDF μπορεί να είναι με τη μορφή **τριπλέτων** ή **γραφημάτων**.

# RDF Graph

- ▶ Το γράφημα RDF είναι ένα κατευθυνόμενο γράφημα που χρησιμοποιείται για να χρησιμεύσει ως γλώσσα περιγραφής για δεδομένα στον παγκόσμιο ιστό και σε άλλα ηλεκτρονικά δίκτυα.
- ▶ Μια δήλωση γραφήματος RDF αντιπροσωπεύεται από:
  - 1) έναν κόμβο για το υποκείμενο,
  - 2) μια ακμή που μεταβαίνει από ένα υποκείμενο σε ένα αντικείμενο για το κατηγορημα
  - 3) έναν κόμβο για το αντικείμενο.



# RDF - τριπλέτες

- ▶ Οι πόροι περιγράφονται χρησιμοποιώντας τριπλέτες.
- ▶ Οι τριπλέτες αποτυπώνουν τη σχέση μεταξύ υποκειμένου και αντικειμένου: έχουν υποκείμενο, κατηγορημα και αντικείμενο. Οι τριπλέτες περικλείονται μέσα σε αγκύλες.

# RDF

Παράδειγμα:

- ▶ Η Αθήνα είναι πρωτεύουσα της Ελλάδας:
- ▶ <Αθήνα> <είναι πρωτεύουσα><Ελλάδα>

Οι τριάδες μπορούν επίσης να αναπαρασταθούν με τη μορφή URI (Uniform Resource Identifier). Παράδειγμα τριπλέτας URI:

- ▶ <<http://www.abc.org/subject/Athens>>
- ▶ <<http://www.abc.org/predicate/capitalOf>>
- ▶ <<http://www.abc.org/object/Greece>>.

Κάθε τριπλέτα τερματίζεται με τελεία.

# RDF Schema

- ▶ Η RDF παρέχει μέθοδο για να εκφράζουμε απλές δηλώσεις που αφορούν πόρους, χρησιμοποιώντας ιδιότητες (με ονόματα) και τιμές.
- ▶ Οι κοινότητες χρηστών της RDF χρειάζονται ακόμη τη δυνατότητα να ορίζουν **λεξιλόγια όρων** (*vocabularies* or *terms*) που μπορούν να χρησιμοποιηθούν σε δηλώσεις. Δηλαδή να δηλώσουν **κλάσεις** (*classes*) πόρων και **ιδιότητες** (*properties*) που θα χρησιμοποιήσουν για να περιγράψουν αυτούς τους πόρους.
- ▶ Παράδειγμα: όσοι ενδιαφέρονται για τη περιγραφή βιβλιογραφικών πόρων χρειάζονται κλάσεις όπως *Book* ή *MagazineArticle*, και ιδιότητες όπως *author*, *title*, κ.α., για να τους περιγράψουν.
- ▶ Η RDF δεν παρέχει τα μέσα για ορισμό τέτοιων κλάσεων και ιδιοτήτων που αφορούν συγκεκριμένες εφαρμογές.
- ▶ Αντίθετα, τέτοιες κλάσεις και ιδιότητες περιγράφονται σαν ένα **λεξιλόγιο RDF**, χρησιμοποιώντας επεκτάσεις της RDF που παρέχονται από την γλώσσα περιγραφής λεξιλογίων RDF που ονομάζεται **RDF Schema**.

# RDF Schema

- ▶ Η RDF Schema δεν παρέχει λεξιλόγιο για κλάσεις και ιδιότητες που αφορούν εφαρμογές. Παρέχει όμως τα μέσα για να **περιγράψουμε** τέτοιες κλάσεις και ιδιότητες. Με άλλα λόγια, η RDF Schema παρέχει ένα **σύστημα τύπων (type system)** για την RDF.
- ▶ Το σύστημα τύπων της RDF Schema μοιάζει αρκετά με το σύστημα τύπων που διαθέτουν αντικειμενοστραφείς γλώσσες προγραμματισμού όπως η Java.
- ▶ Η RDF Schema επιτρέπει να δηλώνεται ότι κάποιοι πόροι αποτελούν **στιγμιότυπα (instances)** μιας ή περισσότερων κλάσεων. Επιτρέπει επίσης την ιεραρχική οργάνωση των κλάσεων.
- ▶ Παράδειγμα: μπορούμε να δηλώσουμε ότι η κλάση *Dog* είναι υποκλάση της κλάσης *Mammal* η οποία με τη σειρά της είναι υποκλάση της κλάσης *Animal*. Κάθε πόρος επομένως που ανήκει στην κλάση *Dog* υπονοείται ότι ανήκει επίσης και στις κλάσεις *Mammal* και *Animal*.



# RDF Schema

- ▶ Οι δυνατότητες αυτές παρέχονται με τη μορφή λεξιλογίου της RDF, δηλαδή σαν εξειδικευμένο σύνολο προκαθορισμένων πόρων της RDF, καθένας με δική του ξεχωριστή σημασία.
- ▶ Οι πόροι στο λεξιλόγιο της RDF Schema διαθέτουν τα δικά τους URIs με πρόθεμα το <http://www.w3.org/2000/01/rdf-schema#> (που συνήθως συσχετίζεται με το πρόθεμα `rdfs:`).
- ▶ Οι περιγραφές λεξιλογίων (schemas) που γράφονται σε RDF Schema είναι έγκυροι RDF γράφοι. Έτσι, το λογισμικό της RDF (που δεν έχει γραφτεί για να επεξεργάζεται λεξιλόγια της RDF Schema) αναγνωρίζει τέτοιους γράφους ως έγκυρους RDF γράφους που περιλαμβάνουν πόρους και ιδιότητες όμως δεν θα «κατανοεί» την ενσωματωμένη σημασία αυτών των όρων της RDF Schema (για την κατανόηση των οποίων χρειαζόμαστε επιπλέον λογισμικό).

# Λεξιόλογο Όρων της RDF

- ▶ Όροι για κλάσεις
  - `rdfs:Class`
  - `rdfs:subClassOf`
- ▶ Όροι για ιδιότητες
  - `rdfs:domain`
  - `rdfs:range`
  - `rdfs:subPropertyOf`
- ▶ Όροι για συλλογές
  - `rdfs:member`
  - `rdfs:Container`
  - `rdfs:ContainerMembershipProperty`
- ▶ Ειδικές κλάσεις
  - `rdfs:Resource`
  - `rdfs:Literal`
  - `rdfs:Datatype`
- ▶ Ειδικές ιδιότητες
  - `rdfs:comment`
  - `rdfs:seeAlso`
  - `rdfs:isDefinedBy`
  - `rdfs:label`

Πλήρης Λίστα

[https://en.wikipedia.org/wiki/Resource\\_Description\\_Framework](https://en.wikipedia.org/wiki/Resource_Description_Framework)

# Κλάσεις και πόροι

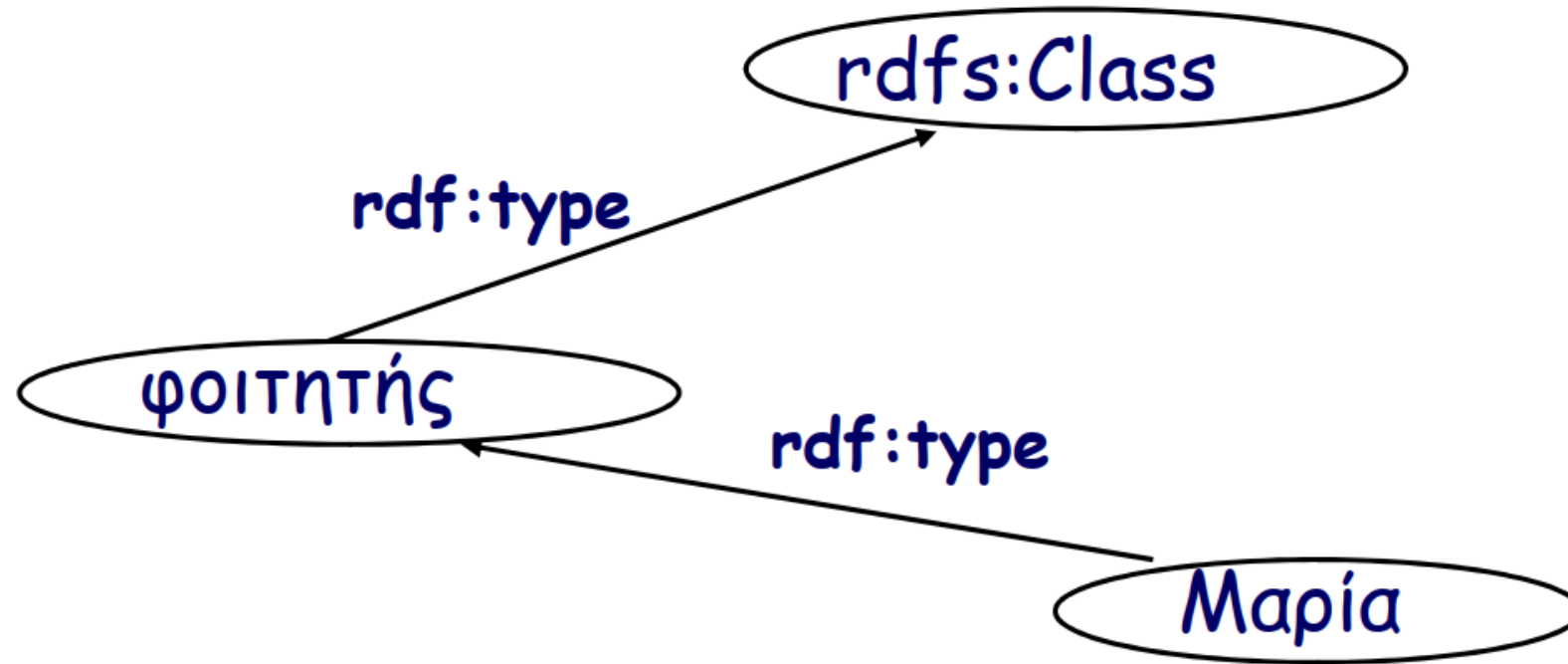
- ▶ Η RDFS ορίζει λεξιλόγιο για **πόρους** και **κλάσεις**:
- ▶ οτιδήποτε στην RDF είναι “πόρος”
- ▶ οι “κλάσεις” είναι επίσης πόροι, αλλά ...
- ▶ είναι ταυτόχρονα συλλογές πιθανών πόρων
  
- ▶ Ανάμεσα στους πόρους ορίζονται σχέσεις:
- ▶ “typing”: ένας πόρος ανήκει σε μια συγκεκριμένη κλάση
- ▶ “subclassing”: τα στιγμιότυπα της μίας είναι και στιγμιότυπα της άλλης
- ▶ Σημειώστε ότι ο ίδιος πόρος μπορεί να ανήκει σε περισσότερες από μια κλάσεις
  
- ▶ “Type”, “subclass” είναι απλές δηλώσεις πάνω σε πόρους
- ▶ Οι πόροι μπορούν να προσδιορίζονται μέσω των URIs
- ▶ επομένως, αυτές οι δηλώσεις μπορούν να εκφραστούν σε RDF.

# Κλάσεις

- ▶ Οι πόροι είναι δυνατό να χωριστούν σε ομάδες που ονομάζονται **κλάσεις (classes)**. Τα μέλη μια κλάσης ονομάζονται **στιγμιότυπα (instances)** της κλάσης.
- ▶ Οι κλάσεις είναι και οι ίδιες πόροι.
- ▶ Στην RDF υπάρχει διάκριση μεταξύ μιας κλάσης και των στιγμιότυπων της. Με κάθε κλάση συνδέεται ένα σύνολο το οποίο ονομάζεται **έκταση της κλάσης (class extension)**, το οποίο είναι το σύνολο των στιγμιότυπων της κλάσης. Δύο κλάσεις μπορεί να έχουν το ίδιο σύνολο στιγμιότυπων αλλά να είναι διαφορετικές κλάσεις.
- ▶ Μια κλάση μπορεί να είναι μέλος της επέκτασης της καθώς και στιγμιότυπο του εαυτού της.
- ▶ Οι κλάσεις περιγράφονται με τη βοήθεια των πόρων `rdfs:Class` and `rdfs:Resource` και των ιδιοτήτων `rdf:type` and `rdfs:subClassOf` της RDF Schema.
- ▶ Στην RDF Schema, κλάση είναι κάθε πόρος που διαθέτει την ιδιότητα `rdf:type` με τιμή τον πόρο `rdfs:Class`. Σημειώστε ότι η ιδιότητα `rdf:type` χρησιμοποιείται για να δηλώσει ότι ένας πόρος είναι στιγμιότυπο κάποιας κλάσης.

# Κλάσεις

- ▶ Παράδειγμα δήλωσης κλάσης:



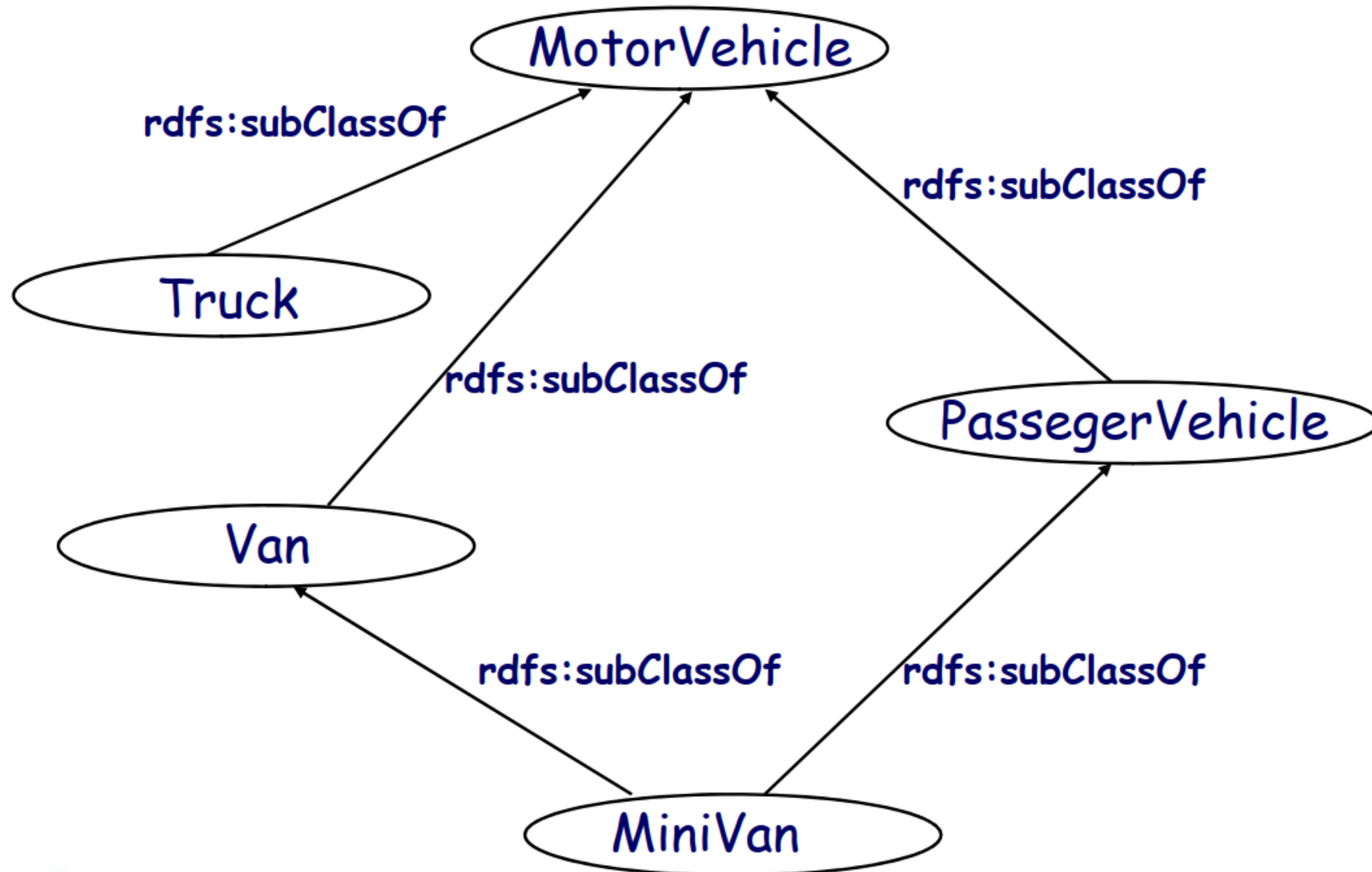
# Κλάσεις - Ιεραρχία

- ▶ Η ιδιότητα `rdfs:subClassOf` μπορεί να χρησιμοποιηθεί για να δηλώσει ότι μια κλάση είναι υποκλάση μιας άλλης κλάσης.
- ▶ Αν μια κλάση C είναι υποκλάση μιας κλάσης C', τότε όλα τα στιγμιότυπα της C είναι επίσης και στιγμιότυπα της C'.
- ▶ Ο όρος **υπερκλάση** (super-class) χρησιμοποιείται σαν ο ανάστροφος του όρου **υποκλάση**. Αν μια κλάση C' είναι *υπερκλάση* μιας κλάσης C, τότε όλα τα στιγμιότυπα της C είναι επίσης και στιγμιότυπα της C'.
- ▶ Όλα τα αντικείμενα τα οποία περιγράφονται με RDF ονομάζονται **πόροι**, και είναι στιγμιότυπα της κλάσης `rdfs:Resource`. Αυτή η κλάση περιλαμβάνει τα πάντα. Όλες οι άλλες κλάσεις είναι **υποκλάσεις** της κλάσης αυτής.
- ▶ Η `rdfs:Resource` είναι ένα στιγμιότυπο της `rdfs:Class`

# Κλάσεις

- ▶ Η `rdfs:Class` είναι η κλάση των πόρων που είναι κλάσεις της RDF.
- ▶ Η `rdfs:Class` είναι στιγμιότυπο της `rdfs:Class`.
- ▶ Η `rdfs:Literal` είναι η κλάση που περιλαμβάνει όλες τις τιμές με χαρακτήρες (string).
- ▶ Η `rdfs:Literal` είναι ένα στιγμιότυπο της `rdfs:Class`. Η `rdfs:Literal` είναι υποκλάση της `rdfs:Resource`.
- ▶ Η `rdf:Property` είναι η κλάση των ιδιοτήτων της RDF.
- ▶ Η `rdf:Property` είναι στιγμιότυπο της `rdfs:Class`.
- ▶ Η `rdfs:Datatype` είναι η κλάση όλων των τύπων δεδομένων.
- ▶ Η `rdfs:Datatype` είναι ταυτόχρονα και στιγμιότυπο και υποκλάση της `rdfs:Class`.
- ▶ Κάθε στιγμιότυπο της `rdfs:Datatype` είναι υποκλάση της `rdfs:Literal`.

# Παράδειγμα





# Ιδιότητες

- ▶ Η **Property** είναι μια ειδική κλάση (**rdf:Property**)
- ▶ επομένως οι ιδιότητες είναι επίσης πόροι
- ▶ Οι ιδιότητες περιορίζονται από το **πεδίο ορισμού** (**domain**) και το **πεδίο τιμών** (**range**) τους
- ▶ δηλ., από το τι βρίσκεται “αριστερά” ή “δεξιά” τους
  
- ▶ Π.χ., η **parentOf** είναι μια ιδιότητα με πεδίο ορισμού το **μperson** και πεδίο τιμών το **person**.
- ▶ Μια ιδιότητα μπορεί να είναι **υπό-ιδιότητα** (**subproperty**) μιας άλλης ιδιότητας
- ▶ Π.χ., η **fatherOf** είναι **subProperty** της **ParentOf**

# Ιδιότητες

- ▶ Η ιδιότητα `rdfs:subPropertyOf` μπορεί να χρησιμοποιηθεί για να δηλώσει ότι μια ιδιότητα είναι **υπο-ιδιότητα** (**sub-property**) μιας άλλης.
- ▶ Αν μια ιδιότητα  $P$  είναι υπο-ιδιότητα μιας ιδιότητας  $P'$ , τότε όλα τα ζεύγη πόρων τα οποία συσχετίζονται με την  $P$  συσχετίζονται επίσης με την  $P'$ .
- ▶ Π.χ. Αν ο **Νίκος** είναι πατέρας (**fatherOf**) του **Πέτρου**, τότε ο **Νίκος** είναι γονιός (**parentOf**) του **Πέτρου**.
- ▶ Ο όρος **υπερ-ιδιότητα** (**super-property**) χρησιμοποιείται συχνά σαν ο αντίστροφος του όρου υπο-ιδιότητα.
- ▶ Αν μια ιδιότητα  $P'$  είναι υπερ-ιδιότητα μιας ιδιότητας  $P$ , τότε όλα τα ζεύγη πόρων τα οποία συσχετίζονται με την  $P$  συσχετίζονται επίσης με την  $P'$ .
- ▶ Δεν έχει οριστεί μια ιδιότητα η οποία είναι υπερ-ιδιότητα όλων των ιδιοτήτων.

# Ιδιότητες

- ▶ Οι ιδιότητες είναι επίσης πόροι
- ▶ Έτσι, ιδιότητες ιδιοτήτων μπορούν να εκφραστούν σαν ιδιότητες της RDF
- ▶ Π.χ.: το πεδίο τιμών της ιδιότητας `parentOf` είναι το `Person`
- ▶ Η ιδιότητα `rdfs:range` είναι στιγμιότυπο της `rdf:Property` και χρησιμοποιείται για να δηλώσει ότι οι τιμές μιας ιδιότητας είναι στιγμιότυπα μιας ή περισσότερων κλάσεων.
- ▶ Η RDF δήλωση `P rdfs:range C` έχει υποκείμενο=`P`, κατηγορημα=`rdfs:range`, και αντικείμενο=`C`
- ▶ Σημαίνει: το `P` είναι στιγμιότυπο της κλάσης `rdf:Property`, η `C` είναι στιγμιότυπο της `rdfs:class`, και οι πόροι που αντιστοιχούν στα αντικείμενα των τριάδων με κατηγορημα `P`, πρέπει να είναι στιγμιότυπα της κλάσης `C`.

# Ιδιότητες

- ▶ Η ιδιότητα `rdfs:domain` είναι στιγμιότυπο της `rdf:Property` και χρησιμοποιείται για να δηλώσει ότι κάθε πόρος που έχει μια συγκεκριμένη ιδιότητα είναι στιγμιότυπο μιας ή περισσοτέρων κλάσεων.
- ▶ Η δήλωση: `P rdfs:domain C` υποδηλώνει ότι το `P` είναι ένα στιγμιότυπο της κλάσης `rdf:Property`, η `C` είναι ένα στιγμιότυπο της `rdfs:class` και οι πόροι που αντιστοιχούν στα υποκείμενα των τριάδων των οποίων το κατηγορήμα είναι το `P`, είναι στιγμιότυπα της κλάσης `C`.

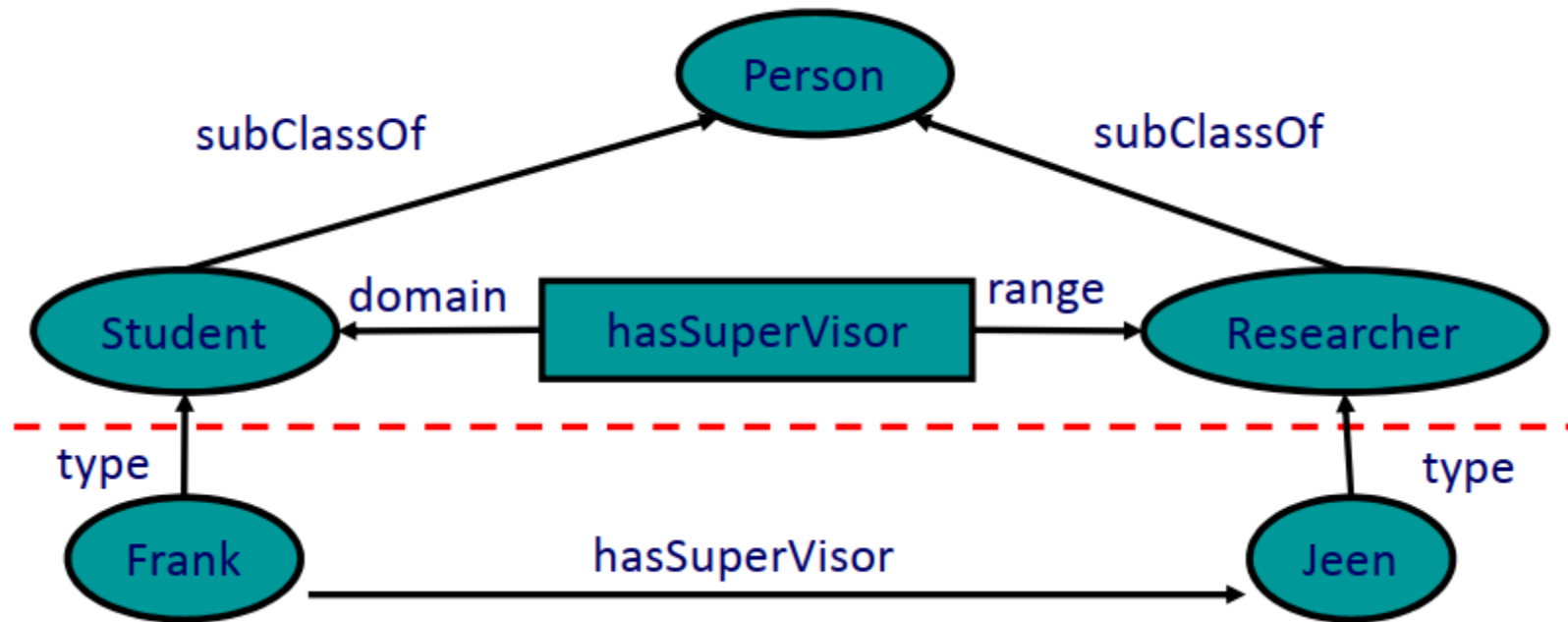
# Ιδιότητες

- ▶ Η ιδιότητα `rdf:type` είναι στιγμιότυπο της `rdf:Property` και χρησιμοποιείται για να δηλώσει ότι ένας πόρος είναι στιγμιότυπο μιας κλάσης.
- ▶ Η δήλωση: `R rdfs:type C` υποδηλώνει ότι το `C` είναι ένα στιγμιότυπο της `rdfs:class` και το `R` είναι στιγμιότυπο της `C`.
- ▶ Η ιδιότητα `rdfs:subClassOf` είναι στιγμιότυπο της `rdf:Property` και χρησιμοποιείται για να δηλώσει ότι τα στιγμιότυπα μιας κλάσης είναι και στιγμιότυπα της άλλης.
- ▶ Η ιδιότητα `rdfs:subClassOf` είναι μεταβατική.
- ▶ Η ιδιότητα `rdfs:subPropertyOf` είναι στιγμιότυπο της `rdf:Property` και χρησιμοποιείται για να δηλώσει ότι όλοι οι πόροι που σχετίζονται με μια ιδιότητα σχετίζονται επίσης και με μια άλλη.
- ▶ Η δήλωση: `P1 rdfs:subPropertyOf P2` υποδηλώνει ότι τα `P1` και `P2` είναι στιγμιότυπα της `rdf:Property`, καθώς και ότι το `P1` είναι υπο-ιδιότητα της `P2`.
- ▶ Η ιδιότητα `rdfs:subPropertyOf` είναι μεταβατική

# Ιδιότητες

- ▶ Η ιδιότητα `rdfs:label` παρέχει μια εύκολα αναγνώσιμη από τον άνθρωπο παραλλαγή του ονόματος ενός πόρου.
- ▶ Η δήλωση: `R rdfs:label L` υποδηλώνει ότι το `L` μια εύκολα αναγνώσιμη από τον άνθρωπο ετικέτα για τον πόρο `R`.
- ▶ Η ιδιότητα `rdfs:comment` χρησιμοποιείται για να παρέχει μια αναγνώσιμη από τον άνθρωπο περιγραφή ενός πόρου.
- ▶ Ένα σχόλιο σε μορφή κειμένου διευκολύνει στο ξεκαθάρισμα της σημασίας των κλάσεων και των ιδιοτήτων της RDF.

# Παράδειγμα



# Σύνταξη της RDF Schema σε XML

```
<rdf:Description ID="MotorVehicle">
  <rdf:type resource="http://www.w3.org/...#Class"/>
  <rdfs:subClassOf rdf:resource="http://www.w3.org/...#Resource"/>
</rdf:Description>
<rdf:Description ID="Truck">
  <rdf:type resource="http://www.w3.org/...#Class"/>
  <rdfs:subClassOf rdf:resource="#MotorVehicle"/>
</rdf:Description>
<rdf:Description ID="registeredTo">
  <rdf:type resource="http://www.w3.org/...#Property"/>
  <rdfs:domain rdf:resource="#MotorVehicle"/>
  <rdfs:range rdf:resource="#Person"/>
</rdf:Description>
  <rdf:Description ID="ownedBy">
  <rdf:type resource="http://www.w3.org/...#Property"/>
  <rdfs:subPropertyOf rdf:resource="#registeredTo"/>
</rdf:Description>
```



# Σύνταξη της RDF Schema σε XML

```
<?xml version="1.0"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#" xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#">
  <rdfs:Class rdf:ID="Person">
    <rdfs:comment>Person Class</rdfs:comment>
    <rdfs:subClassOf rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#Resource"/>
  </rdfs:Class>
  <rdfs:Class rdf:ID="Student">
    <rdfs:comment>Student Class</rdfs:comment>
    <rdfs:subClassOf rdf:resource="#Person"/>
  </rdfs:Class>
  <rdfs:Class rdf:ID="Teacher">
    <rdfs:comment>Teacher Class</rdfs:comment>
    <rdfs:subClassOf rdf:resource="#Person"/>
  </rdfs:Class>
  <rdfs:Class rdf:ID="Course">
    <rdfs:comment>Course Class</rdfs:comment>
    <rdfs:subClassOf rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#Resource"/>
  </rdfs:Class>
```

# Σύνταξη της RDF Schema σε XML

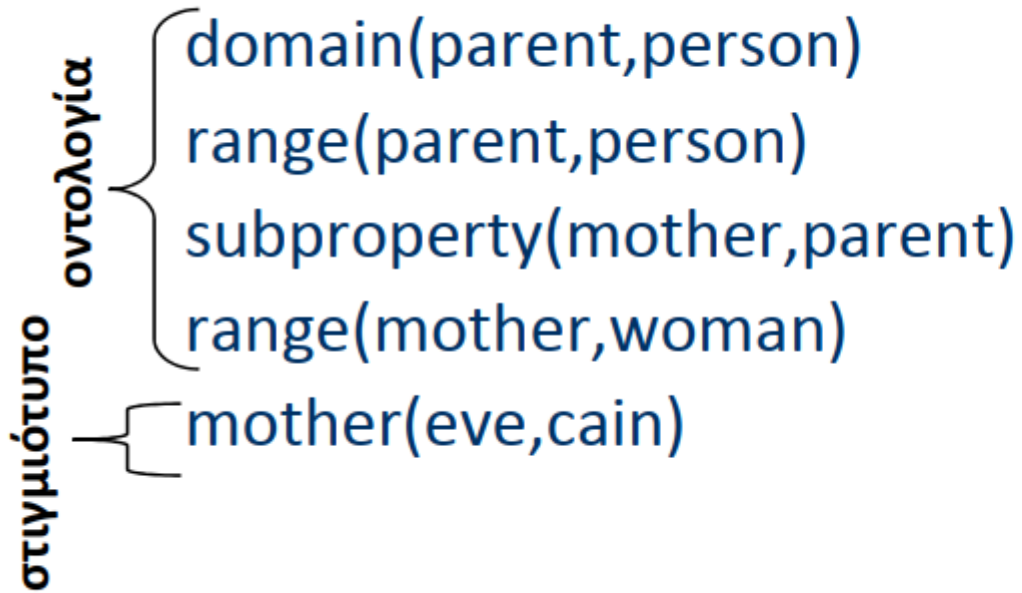
- ▶ `<rdf:Property rdf:ID="teacherOf">`
  - `<rdfs:comment>Teacher of a course</rdfs:comment>`
  - `<rdfs:domain rdf:resource="#Course"/>`
  - `<rdfs:range rdf:resource="#Teacher"/>`
- ▶ `</rdf:Property>`
- ▶ `<rdf:Property rdf:ID="studentsOf">`
  - `<rdfs:comment>List of Students of a course in alphabetical order</rdfs:comment>`
  - `<rdfs:domain rdf:resource="#Course"/>`
  - `<rdfs:range rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#Seq"/>`
- ▶ `</rdf:Property>`
- ▶ `<rdf:Property rdf:ID="nameOf">`
- ▶ `<rdfs:comment>Name of a Person or Course</rdfs:comment>`
  - `<rdfs:domain rdf:resource="#Person"/>`
  - `<rdfs:domain rdf:resource="#Course"/>`
  - `<rdfs:range rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#Literal"/>`
- ▶ `</rdf:Property>`
- ▶ `</rdf:RDF>`

# RDF σε XML

- ▶ Περισσότερες λεπτομέρειες και παραδείγματα:
- ▶ [https://www.w3schools.com/xml/xml\\_rdf.asp](https://www.w3schools.com/xml/xml_rdf.asp)
- ▶ [https://en.wikipedia.org/wiki/Resource\\_Description\\_Framework](https://en.wikipedia.org/wiki/Resource_Description_Framework)
- ▶

# Η RDFS υποστηρίζει απλές συνεπαγωγές

- ▶ Μια οντολογία RDFS μαζί με μερικές δηλώσεις RDFS μπορεί να παράγουν ως συμπεράσματα επιπλέον δηλώσεις RDFS.
- ▶ Αυτό δεν συμβαίνει στην XML.
- ▶ Παράδειγμα:



## **συνεπάγεται:**

```
subclass(woman,person)
parent(eve,cain)
person(eve)
person(cain)
woman(eve)
```

# Μειονεκτήματα RDFS

- ▶ RDFS είναι **αρκετά αδύναμη** να περιγράψει πόρους σε ικανοποιητική λεπτομέρεια, π.χ.:
- ▶ Δεν υποστηρίζει τοπικούς περιορισμούς πεδίου ορισμού και πεδίου τιμών
  - Δεν μπορούμε να πούμε ότι το πεδίο τιμών της **hasChild** είναι το **person** όταν η ιδιότητα εφαρμόζεται σε ανθρώπους, ενώ είναι το **elephant** όταν εφαρμόζεται σε ελέφαντες
- ▶ Δεν υποστηρίζει περιορισμούς *ύπαρξης* και *πλήθους*
  - Δεν μπορεί να δηλώσει ότι όλα τα *στιγμιότυπα* της **person** έχουν (ακριβώς) μια **mother** η οποία είναι επίσης **person**, καθώς και ότι τα μέλη της κλάσης **persons** έχουν ακριβώς 2 **parents**
- ▶ Δεν υποστηρίζει *μεταβατικές*, *ανάστροφες* και *συμμετρικές* ιδιότητες
  - Δεν μπορεί να δηλώσει ότι η **isPartOf** είναι μεταβατική, ότι η **hasPart** είναι η ανάστροφη της **isPartOf** ή ότι η **touches** είναι συμμετρική
- ▶ Η OWL είναι ένα επίπεδο πάνω στην RDFS για την παροχή των χαρακτηριστικών αυτών.

# Serialization formats

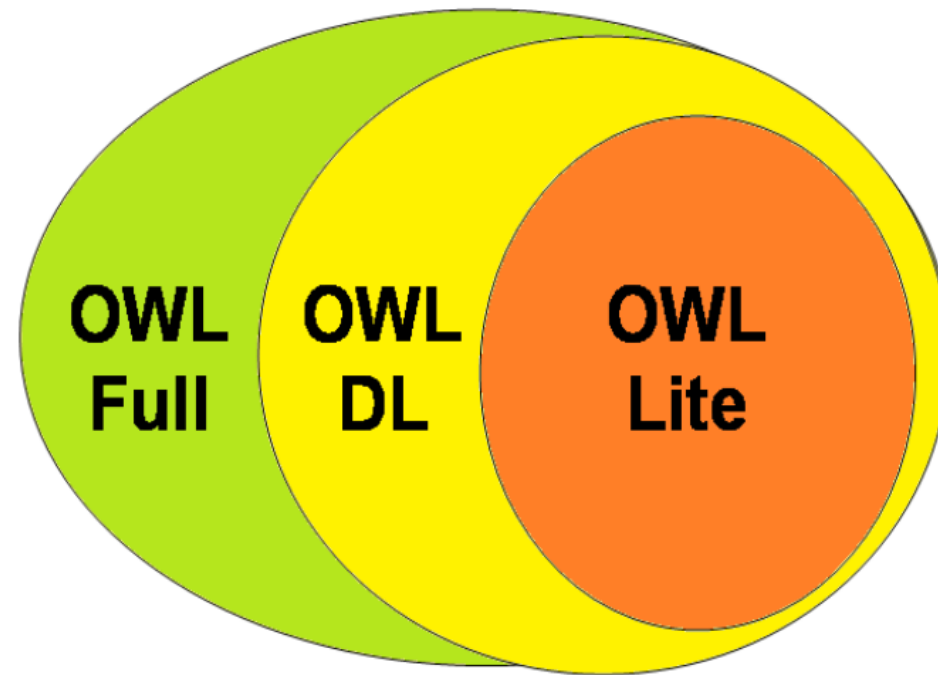
- ▶ [https://en.wikipedia.org/wiki/Resource\\_Description\\_Framework#Serialization formats](https://en.wikipedia.org/wiki/Resource_Description_Framework#Serialization_formats)

- ▶ **Web Ontology Language**

- ▶ Η OWL είναι μια οικογένεια γλωσσών αναπαράστασης γνώσης για τη σύνταξη οντολογιών. Οι οντολογίες είναι ένας επίσημος τρόπος περιγραφής ταξινομιών και δικτύων ταξινόμησης, ο οποίος ουσιαστικά ορίζει τη δομή της γνώσης για διάφορους τομείς: τα ουσιαστικά που αντιπροσωπεύουν κατηγορίες αντικειμένων και τα ρήματα που αντιπροσωπεύουν τις σχέσεις μεταξύ των αντικειμένων.

# OWL

- ▶ OWL Lite
- ▶ OWL DL
- ▶ OWL Full





# Σύνταξη

- ▶ Κλάσεις και ιδιότητες: ίδιες όπως στην RDF
- ▶ `<owl:Class rdf:ID="Person" />`
- ▶ `<owl:Class rdf:ID="PostGraduateStudent" />`
- ▶ Χρησιμοποιείται ένα στοιχείο `rdf:ID` ως αναγνωριστικό της κλάσης. Τα πλήρη URI που αντιστοιχούν στα ονόματα των πόρων καθορίζονται με βάση το `base URI` της οντολογίας.
- ▶ Δύο είδη ιδιοτήτων:
- ▶ **Ιδιότητα αντικειμένου (object property)**: περιγράφουν συσχετίσεις μεταξύ ενός ζεύγους πόρων
- ▶ **Ιδιότητα τύπου δεδομένων (datatype property)**: περιγράφουν συσχετίσεις μεταξύ ενός πόρου και μίας τιμής κάποιου τύπου δεδομένων.
- ▶ Η ιδιότητα **`rdfs:domain`** υποδηλώνει ότι κάθε υποκείμενο, οποιασδήποτε ολοκληρωμένης πρότασης που χρησιμοποιεί μία ιδιότητα, είναι και μέλος της κλάσης που καθορίζει η ιδιότητα αυτή.
- ▶ Η ιδιότητα **`rdfs:range`** υποδηλώνει ότι κάθε αντικείμενο, οποιασδήποτε ολοκληρωμένης πρότασης που περιγράφεται από μία ιδιότητα, είναι μέλος της κλάσης ή του τύπου δεδομένων που περιγράφει η ιδιότητα αυτή.

# Σύνταξη

- ▶ Περισσότερες λεπτομέρειες και παραδείγματα σε ξεχωριστό PDF

# Examples

- ▶ [https://en.wikipedia.org/wiki/Web\\_Ontology\\_Language#Examples](https://en.wikipedia.org/wiki/Web_Ontology_Language#Examples)
- ▶ <https://cambridgesemantics.com/blog/semantic-university/learn-owl-rdfs/owl-references-humans/>

# Εργαλεία συγγραφής οντοτήτων

- ▶ Protégé
- ▶ <https://protege.stanford.edu/>

# Σχέσεις μεταξύ τεχνολογιών σημασιολογικού ιστού

