

ΑΝΑΠΑΡΑΣΤΑΣΗ ΓΝΩΣΗΣ ΣΤΟΝ ΠΑΓΚΟΣΜΙΟ ΙΣΤΟ

RDF Schema
(RDFS)

Ι. Χατζηλυγερούδης

RDF vs RDFS

■ Η RDF

1. Περιγράφει συγκεκριμένους πόρους και (προτασιακές) σχέσεις μεταξύ τους

Αλλά δεν

2. περιγράφει τύπους πόρων (δηλ. γενικές έννοιες) και τις σχέσεις τους
3. περιγράφει τύπους σχέσεων και τις σχέσεις τους
4. καθορίζει σημασιολογία σχέσεων

■ Η RDFS

Παρέχει τα 2, 3 και 4.

RDFS-Κλάσεις

- Η RDFS διαχειρίζεται **κλάσεις** (classes) όχι στιγμιότυπα (instances)
 - Π.χ. περιγράφει τις έννοιες «μάθημα», «λέκτορας», «καθηγητής» κλπ ως κλάσεις, δηλ. ως γενικές έννοιες, που αντιπροσωπεύουν ένα σύνολο στιγμιότυπων (δηλ. συγκεκριμένων οντοτήτων)
 - Μια χρήση των κλάσεων είναι για τον **καθορισμό περιορισμών** στα υποκείμενα/πόρους ή στα αντικείμενα/τιμές των κατηγορημάτων/ιδιοτήτων
 - Π.χ. για να αποτραπεί η χρήση των παρακάτω προτάσεων (που δεν είναι δόκιμες)
 - «Το μάθημα Java διδάσκεται από το μάθημα C»
 - «Η αίθουσα B4 διδάσκεται από τον John Hatzis»
- πρέπει να τεθεί ως περιορισμός η κλάση «διδάσκων» (tutor) για την τιμή (λόγω της πρώτης πρότασης) και η κλάση «μάθημα» (course) για το υποκείμενο/πόρο (λόγω της δεύτερης πρότασης) της ιδιότητας «διδάσκεται από» (is taught by). Στην πρώτη περίπτωση έχουμε περιορισμό στο **πεδίο τιμών** (range), ενώ στη δεύτερη στο **πεδίο ορισμού** (domain).

RDFS-Κλάσεις

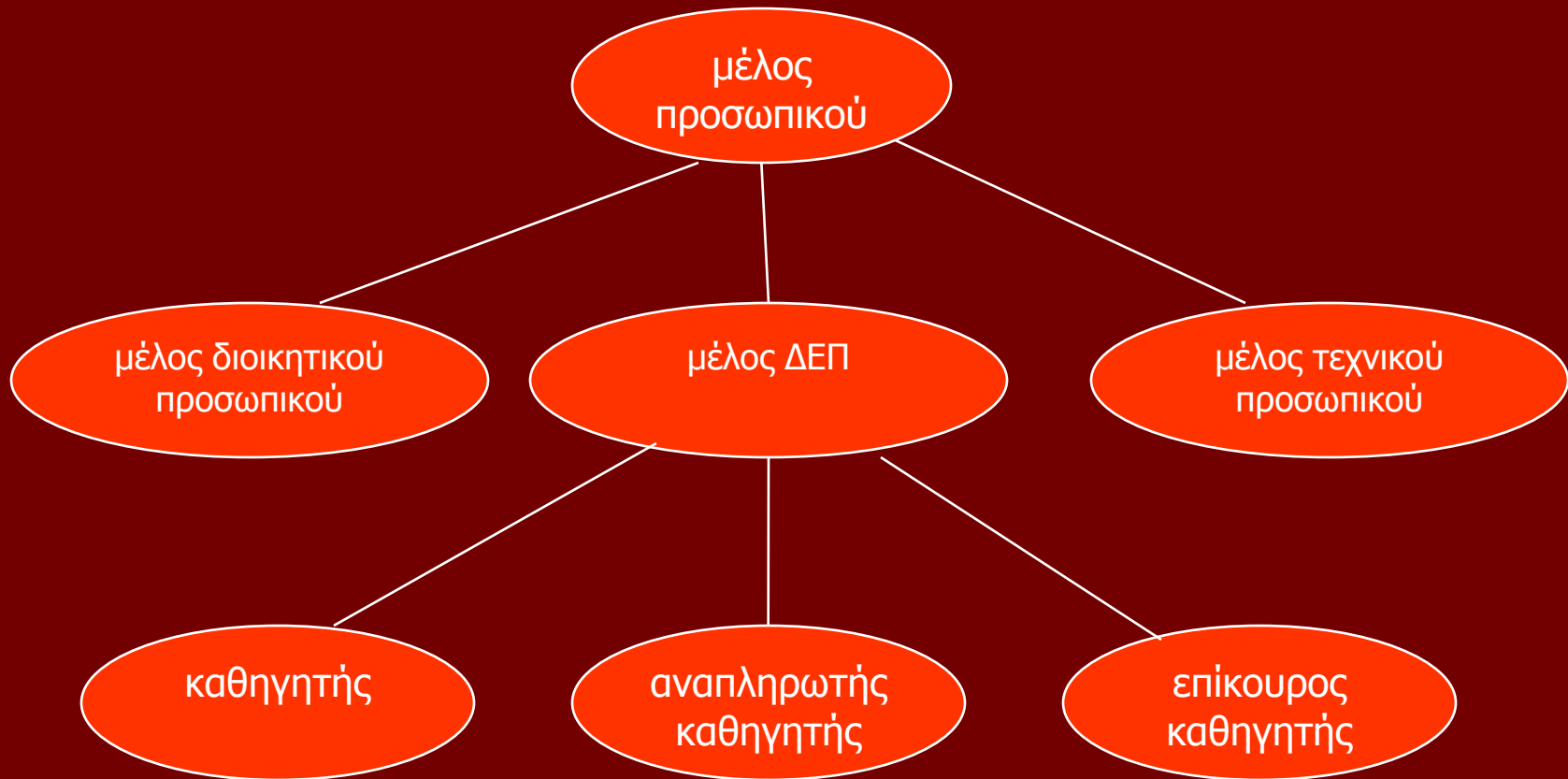
- Οι κλάσεις είναι οργανωμένες σε ιεραρχίες (hierarchies)
- Υπάρχει μια βασική σχέση μεταξύ κλάσεων σε μια ιεραρχία: «είναι υποκλάση της» (is subclass of) και η αντίστροφή της: «είναι υπερκλάση της» (is superclass of)
 - Π.χ. η κλάση «καθηγητής» είναι υποκλάση της κλάσης «μέλος ΔΕΠ» και αντίστροφα η κλάση «μέλος ΔΕΠ» είναι υπερκλάση της κλάσης «καθηγητής»
- Μια κλάση A είναι υποκλάση της B αν κάθε στιγμιότυπο της A είναι και στιγμιότυπο της B
 - Π.χ. τα στιγμιότυπα της κλάσης «καθηγητής» είναι στιγμιότυπα της κλάσης, δηλ. όλοι οι καθηγητές είναι μέλη ΔΕΠ
- Μια κλάση μπορεί να έχει περισσότερες από μια υπερκλάσεις
 - Π.χ. αν η κλάση A είναι υποκλάση της B1 και της B2, αυτό σημαίνει ότι κάθε στιγμιότυπο της A είναι στιγμιότυπο της B1 και της B2

RDFS-Κλάσεις

- Οι κλάσεις κληρονομούν ιδιότητες των υπερκλάσεων τους-κληρονομικότητα (inheritance): καθορίζει τη σημασιολογία της σχέσης «είναι υποκλάση της»
 - Π.χ. έστω ο περιορισμός «τα μαθήματα πρέπει να διδάσκονται μόνο από μέλη ΔΕΠ».
 - Αν ο John Hatzis έχει οριστεί καθηγητής (δηλ. ως στιγμιότυπο της κλάσης «καθηγητής») τότε δεν επιτρέπεται να διδάξει (δηλ. να είναι τιμή της ιδιότητας «διδάσκεται από»), ενώ θάπρεπε.
 - Λόγω κληρονόμησης από την κλάση «καθηγητής» των ιδιοτήτων της υπερκλάσης της «μέλος ΔΕΠ» επιτρέπεται (αλλιώς θάπρεπε να προστεθεί αντίστοιχη πρόταση).

RDFS-Κλάσεις

Ιεραρχία Κλάσεων



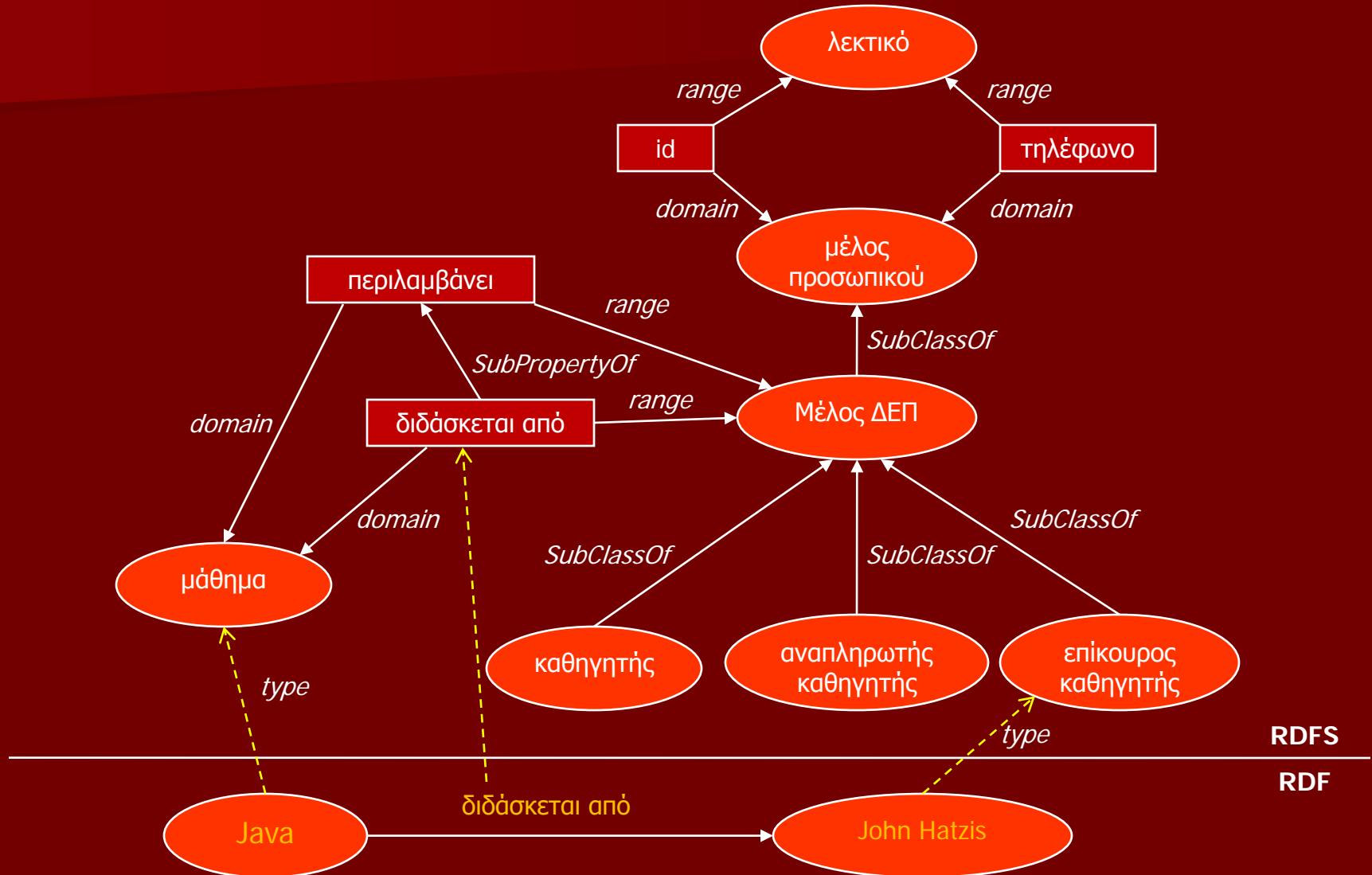
RDFS-Ιδιότητες

- Αντίστοιχα με τις ιεραρχίες κλάσεων υπάρχουν και ιεραρχίες ιδιοτήτων
 - Π.χ. η ιδιότητα «διδάσκεται από» (is taught by) είναι υποϊδιότητα της «περιλαμβάνει» (involves)
 - Αν ένα μάθημα c διδάσκεται από ένα μέλος ΔΕΠ, τότε το c περιλαμβάνει επίσης και το διδάσκοντα a . Το αντίστροφο δεν είναι απαραίτητα αληθές (π.χ. μπορεί ο a να είναι απλώς υπεύθυνος ή βαθμολογητής αλλά όχι διδάσκων).

RDF vs RDFS

- Η RDF ασχολείται με περιγραφές πόρων ως στιγμιοτύπων με τη μορφή προτάσεων.
- Η RDFS ασχολείται με περιγραφές κλάσεων, ιδιοτήτων σχέσεων μεταξύ τους (δηλ. ιεραρχίες) που δρουν ως περιορισμοί στα στοιχεία προτάσεων της RDF
- Δρουν σε διαφορετικά επίπεδα

Επίπεδα RDF-RDFS



RDFS-Σύνταξη

- Η RDFS χρησιμοποιεί την RDF για να εκφράσει την απαιτούμενη μοντελοποίηση
- Αυτό γίνεται με τη χρήση πόρων και ιδιοτήτων
- Π.χ. Για την «lecturer is a subClassOf academicStaff” :
 1. Ορίζουμε τους απαιτούμενους πόρους για τα lecture, academicStaff, subClassOf
 2. Ορίζουμε το subClassOf ως ιδιότητα
 3. Γράφουμε την τριάδα (lecture, subClassOf, academicStaff)

RDFS-Σύνταξη

Βασικές Κλάσεις

- `rdfs:Resource` , η κλάση όλων των πόρων
- `rdfs:Class` , η κλάση όλων των κλάσεων
- `rdfs:Literal` , η κλάση όλων των λεκτικών (αλφαριθμητικών)
- `rdf:Property` , η κλάση όλων των ιδιοτήτων
- `rdf:Statement` , η κλάση όλων των υποστασιοποιημένων προτάσεων

Π.χ. ορισμός της κλάσης `lecturer`

```
<rdfs:Class rdf:ID="lecturer" >  
  ...  
</rdfs:Class >
```

RDFS-Σύνταξη

Βασικές ιδιότητες για ορισμό σχέσεων

- **rdf:type** : Συνδέει ένα πόρο με την κλάση του, ο οποίος δηλώνεται ως στιγμιότυπο της συγκεκριμένης κλάσης.
- **rdfs:subClassOf** : Συσχετίζει μια κλάση με μια από τις υπερκλάσεις της. Όλα τα στιγμιότυπα της κλάσης είναι και στιγμιότυπα της υπερκλάσης. Μια κλάση ενδέχεται να είναι υποκλάση περισσότερων της μιας κλάσεων (π.χ. η `femaleProf` υποκλάση των `female` και `professor`).
- **rdfs:subPropertyOf** : Συσχετίζει μια ιδιότητα με μια από τις υπεριδιότητες της.

Π.χ. «Όλοι οι λέκτορες είναι μέλη του προσωπικού».

```
<rdfs:Class rdf:about="lecturer" >  
  <rdfs:subClassOf rdf:resource = "staffMember"/>  
</rdfs:Class>
```

RDFS-Σύνταξη

Βασικές ιδιότητες για περιορισμό ιδιοτήτων

- **rdfs:domain** : Καθορίζει το πεδίο ορισμού μιας ιδιότητας P και δηλώνει ότι όλοι οι πόροι που έχουν την ιδιότητα P είναι στιγμιότυπα των κλάσεων του πεδίου ορισμού.
- **rdfs:range** : Καθορίζει το σύνολο τιμών μιας ιδιότητας P και δηλώνει ότι οι τιμές της ιδιότητας P είναι στιγμιότυπα των κλάσεων του συνόλου τιμών.

```
<rdf:Property rdf:ID="phone" >  
  <rdfs:domain rdf:resource = "#staffMember"/>  
  <rdfs:range rdf:resource = "&rdf;Literal"/>  
</rdfs:Property>
```

[Όταν ένα πόρος έχει (ως ιδιότητα) τηλεφωνικό αριθμό (phone) τότε είναι (μέσω συμπερασμού) μέλος του προσωπικού και η τιμή του είναι ένα λεκτικό].

RDFS-Σύνταξη

Ιδιότητες για υποστασιοποίηση

- **rdf:subject** : Συνδέει μια υποστασιοποιημένη πρόταση με το υποκείμενό της.
- **rdf:predicate** : Συνδέει μια υποστασιοποιημένη πρόταση με το κατηγορημά της.
- **rdf:object** : Συνδέει μια υποστασιοποιημένη πρόταση με το αντικείμενό της

Κλάσεις-υποδοχείς

- **rdf:Bag** : Η κλάση των πολυσυνόλων
- **rdf:Seq** : Η κλάση των ακολουθιών
- **rdf:Alt** : Η κλάση των εναλλακτικών
- **rdf:Container** : Μια υπερκλάση όλων των κλάσεων-υποδοχέων (συμπεριλαμβανομένων των παραπάνω)

RDFS-Σύνταξη

Βοηθητικές ιδιότητες

- **rdfs:seeAlso** : Συνδέει ένα πόρο με ένα άλλο πόρο που τον περιγράφει.
- **rdfs:isDefinedBy** : Υποϊδιότητα της **rdfs:seeAlso** και συνδέει ένα πόρο με το σημείο όπου μπορεί να βρεθεί ο ορισμός του, συνήθως ένα σχήμα RDF.
- **rdfs:comment** : Σχόλια, συνήθως μακροσκελές κείμενο, που μπορεί να συσχετιστεί με κάποιο πόρο.
- **rdfs:label** : Μια φιλική ετικέτα σχετική με ένα πόρο. Μπορεί να χρησιμεύσει και ως όνομα κόμβου σε μια γραφική αναπαράσταση του εγγράφου RDF.

Παράδειγμα: Πανεπιστήμιο

```
<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="lecturer" >
    <rdfs:comment>
      Η κλάση των λεκτόρων
      Οι λέκτορες (διδάσκοντες) είναι μέλη του ακαδημαϊκού προσωπικού
    </rdfs:comment>
    <rdfs:subClassOf rdf:resource="#academicStaffMember"/>
  </rdfs:Class>

  <rdfs:Class rdf:ID="academicStaffMember" >
    <rdfs:comment>
      Η κλάση των μελών του ακαδημαϊκού προσωπικού
    </rdfs:comment>
    <rdfs:subClassOf rdf:resource="#staffMember"/>
  </rdfs:Class>
```


Παράδειγμα: Πανεπιστήμιο

```
<rdfs:Class rdf:ID="staffMember">  
  <rdfs:comment>Η κλάση των μελών του προσωπικού</rdfs:comment>  
</rdfs:Class>
```

```
<rdfs:Class rdf:ID="course">  
  <rdfs:comment>Η κλάση των μαθημάτων</rdfs:comment>  
</rdfs:Class>
```

```
<rdf:Property rdf:ID="involves">  
  <rdfs:comment>  
    Συσχετίζει μόνο μαθήματα με λέκτορες (διδάσκοντες)  
  </rdfs:comment>  
  <rdfs:domain rdf:resource="#course"/>  
  <rdfs:range rdf:resource="#lecturer"/>  
</rdf:Property>
```

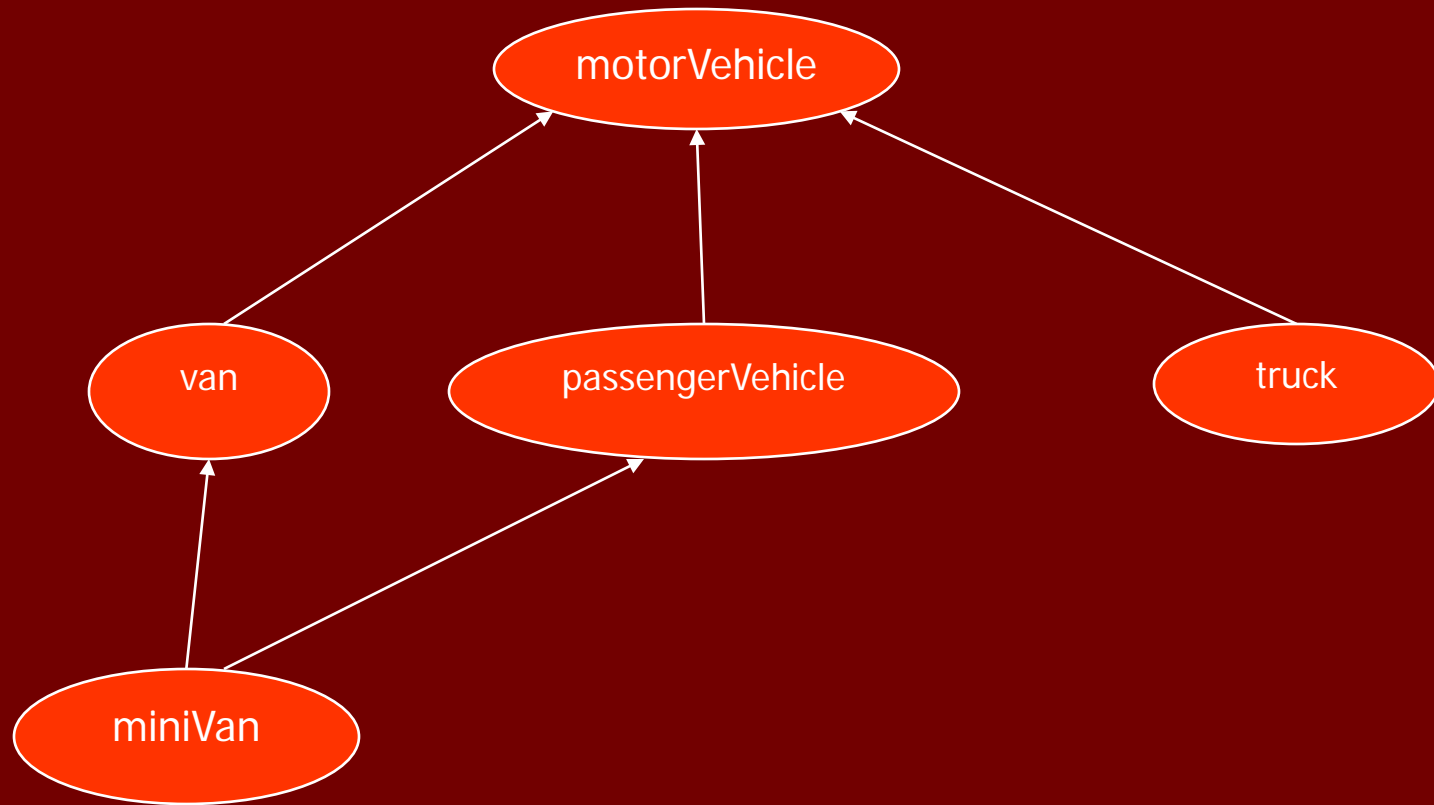
Παράδειγμα: Πανεπιστήμιο

```
<rdf:Property rdf:ID="isTaughtBy" >
  <rdfs:comment>
    Κληρονομεί το πεδίο ορισμού ("course") και το σύνολο τιμών ("lecturer")
    από την υπεριδιότητα "involves".
  </rdfs:comment>
  <rdfs:subPropertyOf rdf:resource="#involves"/>
</rdf:Property>
```

```
<rdf:Property rdf:ID="phone" >
  <rdfs:comment>
    Είναι ιδιότητα των μελών του προσωπικού και παίρνει λεκτικά ω τιμές.
  </rdfs:comment>
  <rdfs:domain rdf:resource="#staffMember"/>
  <rdfs:range rdf:resource="&rdf;Literal"/>
</rdf:Property>
```

Παράδειγμα: Οχήματα

Ιεραρχία Κλάσεων



Παράδειγμα: Οχήματα

```
<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="motorVehicle"/>
```

```
<rdfs:Class rdf:ID="van" >
```

```
  <rdfs:subClassOf rdf:resource="#motorVehicle"/>
```

```
</rdfs:Class>
```

```
<rdfs:Class rdf:ID="truck" >
```

```
  <rdfs:subClassOf rdf:resource="#motorVehicle"/>
```

```
</rdfs:Class>
```

```
<rdfs:Class rdf:ID="passengerVehicle" >
```

```
  <rdfs:subClassOf rdf:resource="#motorVehicle"/>
```

```
</rdfs:Class>
```

```
<rdfs:Class rdf:ID="miniVan" >
```

```
  <rdfs:subClassOf rdf:resource="#passengerVehicle"/>
```

```
  <rdfs:subClassOf rdf:resource="#van"/>
```

```
</rdfs:Class>
```

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

Ορισμός RDF μέσω RDFS

```
<?xml version="1.0" encoding="UTF-16" >
```

```
<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="Statement"/>
```

```
  rdfs:comment="Η κλάση των τριάδων που αποτελούνται από κατηγορημα,  
                υποκείμενο και αντικείμενο (οι υποστασιοποιημένες  
                προτάσεις)"/>
```

```
<rdfs:Class rdf:ID="Property">
```

```
  rdfs:comment="Η κλάση των ιδιοτήτων»/>
```

```
<rdfs:Class rdf:ID="Bag">
```

```
  rdfs:comment="Η κλάση των μη διατεταγμένων συλλογών»/>
```

```
<rdfs:Class rdf:ID="Seq">
```

```
  rdfs:comment="Η κλάση των διατεταγμένων συλλογών»/>
```

```
<rdfs:Class rdf:ID="Alt">
```

```
  rdfs:comment="Η κλάση των συλλογών των εναλλακτικών»/>
```

Ορισμός RDF μέσω RDFS

```
<rdfs:Property rdf:ID="predicate">  
  rdfs:comment="Προσδιορίζει την ιδιότητα που χρησιμοποιείται σε μια πρόταση, όταν η  
    τελευταία αναπαρίσταται με υποστασιοποιημένη μορφή">  
  <rdfs:domain rdf:resource="#Statement"/>  
  <rdfs:range rdf:resource="#Property"/>  
</rdf:Property>  
  
<rdfs:Property rdf:ID="subject">  
  rdfs:comment="Προσδιορίζει τον πόρο που περιγράφεται από μια πρόταση, όταν η  
    τελευταία αναπαρίσταται με υποστασιοποιημένη μορφή">  
  <rdfs:domain rdf:resource="#Statement"/>  
</rdf:Property>  
  
<rdfs:Property rdf:ID="object">  
  rdfs:comment="Προσδιορίζει το αντικείμενο μιας πρότασης, όταν η  
    τελευταία αναπαρίσταται με υποστασιοποιημένη μορφή"/>  
  
<rdfs:Property rdf:ID="type">  
  rdfs:comment="Προσδιορίζει την κλάση ενός πόρου. Ο πόρος είναι  
    στιγμιότυπο της συγκεκριμένης κλάσης"/>  
  
</rdf:RDF>
```

Ορισμός RDFS μέσω RDFS

```
<?xml version="1.0" encoding="UTF-16" >
<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="Resource"/>
    rdfs:comment="Η πιο γενική κλάση"/>

  <rdfs:Class rdf:ID="comment">
    rdfs:comment="Χρήση για περιγραφές"/>
    <rdfs:domain rdf:resource="#Resource"/>
    <rdfs:range rdf:resource="#Literal"/>
  </rdfs:Class>

  <rdfs:Class rdf:ID="Class">
    rdfs:comment="Η έννοια των κλάσεων. Οι κλάσεις είναι πόροι"/>
    <rdfs:subClassOf rdf:resource="#Resource"/>
  </rdfs:Class>
```

Ορισμός RDFS μέσω RDFS

```
<rdf:Property rdf:ID="subClassOf">  
  <rdfs:domain rdf:resource="#Class"/>  
  <rdfs:range rdf:resource="#Class"/>  
</rdf:Property>
```

```
<rdf:Property rdf:ID="subPropertyOf">  
  <rdfs:domain rdf:resource="&rdf;Property"/>  
  <rdfs:range rdf:resource="&rdf;Property"/>  
</rdf:Property>
```


Σημασιολογία RDF και RDFS

- Μέσω κατηγορηματικής λογικής με ισότητα
- Προσέγγιση:
 - Θεμελιώδη στοιχεία → σταθερές
Resource, Class, Property, subClassOf κλπ
 - Χρήση θεωρίας λιστών
nil, cons(x,l), first(l), rest(l), item(x,l), list(l)
 - Συνήθης μορφή αξιωμάτων σημασιολογίας:
Type(subClassOf, Property) (Η *subClassOf* είναι ιδιότητα)
 - Κλ με ισότητα, οι μεταβλητές ξεκινούν με ?

Σημασιολογία RDF και RDFS

- Βασικά κατηγορήματα:

PropVal(P, R, V) (αναπαράσταση πρότασης RDF)

Type(R, T) (αντί του *PropVal(type, R, T)*: ο πόρος *R* είναι τύπου *T*)

Σημασιολογία RDF

■ Κλάσεις

- Όλες οι κλάσεις είναι στιγμιότυπα της Class:

Type(Class, Class), Type(Resource, Class)

Type(Property, Class), Type(Literal, Class)

- Η Resource είναι η γενικότερη κλάση. Όλες οι κλάσεις και οι ιδιότητες είναι πόροι:

Type(?p, Property) → Type(?p, Resource)

Type(?c, Class) → Type(?c, Resource)

- Το κατηγορημα σε μια πρόταση είναι ιδιότητα

PropVal(?p, ?r, ?c) → Type(?p, Property)

Σημασιολογία RDF

■ Ιδιότητα *type*

– $Type(type, Property) \equiv PropVal(type, type, Property)$

(Το *type* είναι ιδιότητα)

– $Type(?r, ?c) \rightarrow (Type(?r, Resource) \wedge Type(?c, Class))$

■ Βοηθητική ιδιότητα *FuncProp*

– Συναρτησιακή ιδιότητα=ιδιότητα που αποτελεί συνάρτηση, δηλ. συσχετίζει ένα πόρο με μια τιμή το πολύ

– Δεν αποτελούν έννοια της RDF, αλλά χρησιμοποιούνται για αξιωματοποίηση άλλων θεμελιωδών στοιχείων

– *FuncProp*: αναπαριστά την κλάση όλων των συναρτησιακών ιδιοτήτων.

$Type(?p, FuncProp) \leftrightarrow (Type(?p, Property) \wedge$

$\forall ?r \forall ?v1 \forall ?v2$

$(PropVal(?p, ?r, ?v1) \wedge$

$PropVal(?p, ?r, ?v2) \rightarrow ?v1 \equiv ?v2))$

Σημασιολογία RDF

- Υποστασιοποιημένες προτάσεις
 - Όλες είναι πόροι, η Statement είναι στιγμιότυπο της Class:
 $Type(?s, Statement) \rightarrow Type(?s, Resource)$
 $Type(Statement, Class)$
 - Μια πρόταση μπορεί να αναλυθεί στα μέρη της τριάδας RDF:
 $Type(?st, Statement) \rightarrow$
 $\exists ?p \exists ?r \exists ?v (PropVal(Predicate, ?st, ?p) \wedge$
 $PropVal(Subject, ?st, ?r) \wedge PropVal(Object, ?st, ?v))$
 - Οι ιδιότητες Subject, Predicate και Object είναι συναρτησιακές ιδιότητες (: κάθε πρόταση έχει ακριβώς ένα υποκείμενο, ένα κατηγορημα, ένα αντικείμενο):
 $Type(Subject, FuncProp)$
 $Type(Predicate, FuncProp)$
 $Type(Object, FuncProp)$

Σημασιολογία RDF

- Υποστασιοποιημένες προτάσεις (συνέχεια)

- Πληροφορίες τυποποίησης:

- $PropVal(Subject, ?st, ?r) \rightarrow$

- $(Type(?st, Statement) \wedge Type(?r, Resource))$

- $PropVal(Predicate, ?st, ?p) \rightarrow$

- $(Type(?st, Statement) \wedge Type(?p, Property))$

- $PropVal(Object, ?st, ?v) \rightarrow$

- $(Type(?st, Statement) \wedge (Type(?v, Resource) \vee$
 $Type(?v, Literal)))$

Σημασιολογία RDF

■ Υποδοχείς

- Όλοι είναι πόροι:

$Type(?c, Container) \rightarrow Type(?c, Resource)$

- Είναι λίστες:

$Type(?c, Container) \rightarrow list(?c)$

- Οι υποδοχείς είναι πολυσύνολα (bags), ακολουθίες (sequences) ή εναλλακτικά (alternatives):

$Type(?c, Container) \leftrightarrow$

$(Type(?c, Bag) \vee Type(?c, Seq) \vee Type(?c, Alt))$

- Τα πολυσύνολα και οι ακολουθίες (sequences) είναι ξένα μεταξύ τους:

$\neg (Type(?c, Bag) \wedge Type(?c, Seq))$

Σημασιολογία RDFS

- Υποκλάσεις και υποϊδιότητες

- Η *subClassOf* είναι ιδιότητα:

Type(subClassOf, Property)

- Αν μια κλάση C είναι υποκλάση της C' τότε όλα τα στιγμιότυπα της C είναι και στιγμιότυπα της C':

PropVal(subClassOf, ?c, ?c') ↔

(Type(?c, Class) ∧ Type(?c', Class) ∧

∀?x(Type(?x, ?c) → Type(?x, ?c')))

- Τα ίδια και για την *subPropertyOf*:

Type(subPropertyOf, Property)

PropVal(subPropertyOf, ?p, ?p') ↔

(Type(?p, Property) ∧ Type(?p', Property) ∧

∀?r ∀?v(PropVal(?p, ?r, ?v) → PropVal(?p', ?r, ?v)))

Σημασιολογία RDFS

■ Περιορισμοί

- Κάθε πόρος περιορισμού (constraint resource) είναι πόρος:

PropVal(subClassOf, ConstraintResource, Resource)

- Οι ιδιότητες περιορισμού (constraint properties) είναι ιδιότητες, που είναι και πόροι περιορισμού:

Type(?cp, ConstraintProperty) ↔

(Type(?cp, ConstraintResource) ∧ (Type(?cp, Property)))

- Τα domain και range είναι ιδιότητες περιορισμού:

Type(domain, ConstraintProperty)

Type(range, ConstraintProperty)

Σημασιολογία RDFS

■ Περιορισμοί (συνέχεια)

- Τα domain και range ορίζουν το πεδίο ορισμού και το σύνολο τιμών μιας ιδιότητας αντίστοιχα.
- *Πεδίο ορισμού* μιας ιδιότητας P: το σύνολο αντικειμένων για τα οποία ισχύει η P. Αν D είναι το πεδίο ορισμού, τότε για κάθε $P(x, y)$, $x \in D$.

$PropVal(domain, ?p, ?d) \rightarrow$

$\forall ?x \forall ?y (PropVal(?p, ?x, ?y) \rightarrow Type(?x, ?d))$

- *Πεδίο τιμών* μιας ιδιότητας P: το σύνολο όλων των τιμών που μπορεί να πάρει η P. Αν R είναι το σύνολο τιμών, τότε για κάθε $P(x, y)$, $y \in R$.

$PropVal(range, ?p, ?r) \rightarrow$

$\forall ?x \forall ?y (PropVal(?p, ?x, ?y) \rightarrow Type(?x, ?r))$

Σημασιολογία RDFS

- Περιορισμοί (συνέχεια)

- Οι παρακάτω τύποι εξάγονται μέσω συμπερασμού από τους προηγούμενους.

PropVal(domain, range, Property)

PropVal(range, range, Property)

PropVal(domain, domain, Property)

PropVal(range, range, Property)

Συλλογισμός/Συμπερασμός σε RDF και RDFS

- Με βάση το προηγούμενο αξιωματικό σύστημα ορισμού σημασιολογίας για RDF και RDFS, θα μπορούσε να οριστεί ένα σύστημα αυτόματου συλλογισμού από προτάσεις RDF και RDFS. Μειονέκτημα: υψηλή πολυπλοκότητα, χαμηλή αποδοτικότητα (λόγω ΚΛΠΤ).
- Υπάρχει ένα σύστημα συλλογισμού που αντιστοιχεί σε μια σημασιολογία RDF, και είναι ακριβές και πλήρες γι' αυτήν, και που λειτουργεί απ' ευθείας σε προτάσεις/τριάδες RDF, χωρίς διαμεσολάβηση ενδιάμεσης μετατροπής σε ΚΛΠΤ.
- Το σύστημα αυτό διαθέτει κανόνες της μορφής:
ΑΝ το E περιέχει συγκεκριμένες τριάδες
ΤΟΤΕ πρόσθεσε στο E συγκεκριμένες επί πλέον τριάδες
(το E είναι ένα σύνολο τριάδων RDF)

Συλλογισμός/Συμπερασμός σε RDF και RDFS

- Βασικά παραδείγματα κανόνων συμπερασμού:

ΑΝ το Ε περιέχει την τριάδα ($?x, ?p, ?y$)

ΤΟΤΕ το Ε επίσης περιέχει την τριάδα ($?p, \text{rdf:type}, \text{rdf:property}$)

(οποιοσδήποτε πόρος $?p$ που χρησιμοποιείται στη θέση της ιδιότητας μιας τριάδας μπορεί να θεωρηθεί μέλος της κλάσης `rdf:Property` μέσω συμπερασμού)

ΑΝ το Ε περιέχει τις τριάδες ($?u, \text{rdfs:subClassOf}, ?v$)

και ($?v, \text{rdfs:subClassOf}, ?w$)

ΤΟΤΕ το Ε επίσης περιέχει την τριάδα ($?u, \text{rdfs:subClassOf}, ?w$)

(μεταβατικότητα της σχέσης της υποκλάσης)

Συλλογισμός/Συμπερασμός σε RDF και RDFS

- Βασικά παραδείγματα κανόνων συμπερασμού:

ΑΝ το E περιέχει τις τριάδες ($?x, \text{rdf:type}, ?u$)
και ($?u, \text{rdfs:subClassOf}, ?v$)

ΤΟΤΕ το E επίσης περιέχει την τριάδα ($?x, \text{rdf:type}, ?u$)
(νόημα της ιδιότητας rdfs:subClassOf)

ΑΝ το E περιέχει τις τριάδες ($?x, ?p, ?y$)
και ($?p, \text{rdfs:range}, ?u$)

ΤΟΤΕ το E επίσης περιέχει την τριάδα ($?y, \text{rdf:type}, ?u$)
(οποιοσδήποτε πόρος $?y$ που εμφανίζεται ως τιμή της ιδιότητας $?p$
μπορεί να θεωρηθεί μέλος του συνόλου τιμών της $?p$ μέσω
συμπερασμού-χρήση ορισμών συνόλου τιμών όχι για *περιορισμό*,
αλλά *συμπερασμό* των μελών του συνόλου)

Ερωτήματα SPARQL

- Η υποβολή ερωτημάτων σε έγγραφο RDF με χρήση γλωσσών ερωτημάτων βασισμένων σε XML (π.χ. XPath) είναι προβληματική, λόγω πολλών παραλλαγών αναπαράστασης περιγραφών.
- Χρησιμοποιείται η SPARQL ως το υπό υιοθέτηση πρότυπο από το W3C.

Ερωτήματα SPARQL

■ Βασικά ερωτήματα

- Βασίζονται στην ταύτιση υποδειγμάτων γράφων
- Απλούστερο υπόδειγμα γράφου: υπόδειγμα τριάδας
- Μοιάζει με τριάδα RDF, αλλά επιτρέπονται μεταβλητές στις θέσεις υποκειμένου, κατηγορήματος ή αντικειμένου
- Ο συνδυασμός υποδειγμάτων τριάδων παράγει ένα βασικό υπόδειγμα γράφου
- Απαιτείται ακριβής ταύτιση με κάποιο γράφο

Ερωτήματα SPARQL

- Βασικά ερωτήματα (συν.)

- Απλό παράδειγμα:

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
```

```
PREFIX rdfs: <http://www.w3.org/2000/01/22-rdf-schema#>
```

```
SELECT ?c
```

```
WHERE
```

```
{
```

```
  ?c rdf:type rdfs:Class .
```

```
}
```

(τα υποδείγματα τριάδων, όπου `rdf:type` είναι η ιδιότητα και `rdfs:Class` το αντικείμενο: ανάκτηση όλων των κλάσεων)

Ερωτήματα SPARQL

- Βασικά ερωτήματα (συν.)

- άλλο παράδειγμα:

```
PREFIX uni: <http://www.mydomain.org/uni-ns# >
```

```
SELECT ?i
```

```
WHERE
```

```
{
```

```
    ?i rdf:type rdfs:course .
```

```
}
```

(ανάκτηση όλων των στιγμιοτύπων της κλάσης course)

Ερωτήματα SPARQL

- Δομή select-from-where

- **SELECT**: καθορίζει τον αριθμό και τη σειρά των προς ανάκτηση δεδομένων
- **FROM**: προσδιορίζει την πηγή των δεδομένων (προαιρετικό)
- **WHERE**: επιβάλλει περιορισμούς στις δυνατές απαντήσεις

Π.χ.

```
SELECT ?x ?y
```

```
WHERE
```

```
{
```

```
  ?x uni:phone ?y .
```

```
}
```

(ανάκτηση όλων των τηλεφώνων των μελών του προσωπικού)

Ερωτήματα SPARQL

```
SELECT ?x ?y
WHERE
{
  ?x rdf:type uni:Lecturer ;
    uni:phone ?y .
}
```



```
SELECT ?x ?y
WHERE
{
  ?x rdf:type uni:Lecturer .
  ?x uni:phone ?y .
}
```

(ανάκτηση όλων των διδασκόντων και των τηλεφώνων τους)

1. Ο όρος `?x rdf:type uni:Lecturer` συγκεντρώνει όλα τα στιγμιότυπα της κλάσης Lecturer και συνδέει το αποτέλεσμα με την μεταβλητή ?x
2. Το `uni:phone ?y` συγκεντρώνει όλες τις τριάδες με κατηγορημα phone
3. Η *έμμεση συνένωση* (implicit join) (λόγω του «;») περιορίζει τις τριάδες αυτές σε κείνες με κοινό υποκείμενο με τις πρώτες (?x)

Ερωτήματα SPARQL

```
SELECT ?n
WHERE
{
  ?x rdf:type uni:Course ;
      uni:isTaughtBy :949352 .
  ?c uni:name ?n .
  FILTER (?c = ?x) .
}
```

(ανάκτηση όλων των μαθημάτων που διδάσκονται από τον διδάσκοντα με κωδικό (ID) 949352)

Το FILTER χρησιμοποιείται για υπόδειξη ενός λογικού περιορισμού. Εδώ ο περιορισμός είναι η *άμεση συνένωση* (explicit join) των μεταβλητών ?c και ?x (χρήση τελεστή ισότητας «=»).

Ερωτήματα SPARQL

■ Προαιρετικά υποδείγματα

- Μέχρι τώρα επιστρέφεται απάντηση αν υπάρχει πλήρης ταύτιση υποδείγματος στη βάση γνώσης
- Συχνά όμως απαιτείται μεγαλύτερη ευελιξία

Π.χ.

```
<uni:lecturer rdf:about="949352">  
  <uni:name>Grigoris Antoniou</uni:name>  
</uni:lecturer>
```

```
<uni:professor rdf:about="949318">  
  <uni:name>John Hatzis</uni:name>  
  <uni:email>ihatz@cti.gr</uni:email>  
</uni:lecturer>
```

```
SELECT ?name ?email  
WHERE  
{  
  ?x rdf:type uni:Lecturer ;  
  uni:name ?name ;  
  uni:email ?email .  
}
```



?name	?email
John Hatzis	ihatz@cti.gr

Ερωτήματα SPARQL

```
<uni:lecturer rdf:about="949352">  
  <uni:name>Grigoris Antoniou</uni:name>  
</uni:lecturer>
```

```
<uni:professor rdf:about="949318">  
  <uni:name>John Hatzis</uni:name>  
  <uni:email>ihatz@cti.gr</uni:email>  
</uni:lecturer>
```

```
SELECT ?name ?email  
WHERE  
{  
  ?x rdf:type uni:Lecturer ;  
    uni:name ?name ;  
  OPTIONAL {?x uni:email ?email }  
}
```



?name	?email
Grigoris Antoniou	
John Hatzis	ihatz@cti.gr