



Περιγραφή Πόρων Ιστού: RDF

Εισαγωγή στο Σημασιολογικό Ιστό

Εισαγωγή (1/5)

- Η XML είναι μία καθολική μετα-γλώσσα για τον ορισμό σήμανσης
- Παρέχει ένα ενιαίο πλαίσιο, καθώς και ένα σύνολο από εργαλεία όπως οι συντακτικοί αναλυτές για την ανταλλαγή δεδομένων και μεταδεδομένων μεταξύ εφαρμογών
- Παρόλα αυτά, η XML δεν παρέχει κάποιο τρόπο καθορισμού της σημασιολογίας (σημασίας) των δεδομένων
 - Π.χ. δεν υπάρχει κάποιο επιδιωκόμενο νόημα που συνδέεται με την ένθεση των ετικετών
- Παράδειγμα: έστω ότι θέλουμε να εκφράσουμε το ακόλουθο γεγονός:
 - *Ο David Billington είναι ο διδάσκων του μαθήματος Διακριτά Μαθηματικά.*
- Υπάρχουν διάφοροι τρόποι αναπαράστασης αυτής της πρότασης σε XML
- Τρεις από αυτούς είναι:

```
<course name="Discrete Mathematics"> <lecturer>David Billington</lecturer> </course>  
  
<lecturer name="David Billington"> <teaches>Discrete Mathematics</teaches> </lecturer>  
  
<teachingOffering>  
<lecturer>David Billington</lecturer> <course>Discrete Mathematics</course>  
</teachingOffering>
```

- Δεν υπάρχει κάποιος τυπικός τρόπος αντιστοίχισης νοήματος στην ένθεση των ετικετών

Εισαγωγή (2/5)

- Η RDF (Resource Description Framework) είναι στην ουσία ένα μοντέλο δεδομένων
 - Παρόλο που αποκαλείται συχνά «γλώσσα»
- Το βασικό δομικό στοιχείο του είναι μία τριάδα αντικειμένου-χαρακτηριστικού-τιμής, η οποία ονομάζεται πρόταση (*statement*)
- Ένα αφηρημένο μοντέλο δεδομένων χρειάζεται μία καθορισμένη σύνταξη ώστε να αναπαρίσταται και να μεταδίδεται, και η RDF διαθέτει μία τέτοια σύνταξη σε XML
 - Το αποτέλεσμα είναι ότι κληρονομεί τα πλεονεκτήματα που σχετίζονται με την XML
 - Ωστόσο είναι δυνατό να υπάρχουν και άλλες συντακτικές αναπαραστάσεις της RDF που δεν βασίζονται στην XML

Εισαγωγή (3/5)

- Η RDF είναι ανεξάρτητη πεδίου, με την έννοια ότι δεν έχει γίνει καμία υπόθεση αναφορικά με κάποιο συγκεκριμένο πεδίο χρήσης της
- Οι χρήστες είναι υπεύθυνοι για τον ορισμό της δικής τους ορολογίας μέσω μίας γλώσσας σχήματος, που αποκαλείται *RDF Schema (RDFS)*
- RDF Schema έναντι XML Schema
 - Η XML Schema περιορίζει τη *δομή* των εγγράφων XML, ενώ η RDF Schema καθορίζει το *λεξιλόγιο* που χρησιμοποιείται στα μοντέλα δεδομένων RDF
 - Στην RDFS μπορούμε να ορίσουμε το λεξιλόγιο, να καθορίσουμε ποιες ιδιότητες ισχύουν, για ποια είδη αντικειμένων και τι τιμές μπορούν να πάρουν, καθώς επίσης και να περιγράψουμε τις σχέσεις ανάμεσα στα αντικείμενα

Εισαγωγή (4/5)

- Π.χ. μπορούμε να γράψουμε: *Ο λέκτορας (lecturer) είναι υποκλάση του μέλους ακαδημαϊκού προσωπικού (academic staff member)*
 - Αυτή η πρόταση σημαίνει ότι όλοι οι λέκτορες είναι και μέλη του ακαδημαϊκού προσωπικού
 - Επιδιώκουμε να συσχετίσουμε τη φράση «είναι υποκλάση του» (is a subclass of) με κάποιο νόημα
 - Η ερμηνεία του συγκεκριμένου όρου δεν εξαρτάται από την εφαρμογή
 - Το επιδιωκόμενο νόημα πρέπει να ακολουθείται από όλα τα προγράμματα επεξεργασίας RDF
 - Οι γλώσσες **RDF/RDFS** μας επιτρέπουν να μοντελοποιούμε συγκεκριμένα πεδία, μέσω του καθορισμού της σημασιολογίας συγκεκριμένων συστατικών

Η σπουδαιότητα της RDF Schema μπορεί να φανεί με ένα παράδειγμα:

```
<academicStaffMember>Grigoris Antoniou</academicStaffMember>  
<professor>Michael Maher</professor>  
<course name="Discrete Mathematics">  
<isTaughtBy>David Billington</isTaughtBy>  
</course>
```

Εισαγωγή (5/5)

- Έστω ότι θέλουμε να συγκεντρώσουμε όλα τα μέλη του ακαδημαϊκού προσωπικού
- Μία παράσταση διαδρομής στη γλώσσα Xpath θα μπορούσε να είναι: `//academicStaffMember`
 - Το αποτέλεσμα είναι μόνο ο Grigoris Antoniou
- Παρόλο που αυτό είναι σωστό από την οπτική γωνία της XML, η απάντηση είναι *σημασιολογικά ανεπαρκής*
 - Οι άνθρωποι-αναγνώστες θα είχαν συμπεριλάβει και τους Michael Maher και David Billington στην απάντηση, καθώς
 - Όλοι οι καθηγητές (professors) είναι μέλη του ακαδημαϊκού προσωπικού (δηλαδή η κλάση professor είναι υποκλάση της academicStaffMember)
 - Τα μαθήματα διδάσκονται μόνο από μέλη του ακαδημαϊκού προσωπικού
- Τέτοιες πληροφορίες χρησιμοποιούν το *σημασιολογικό μοντέλο* του συγκεκριμένου πεδίου και δεν μπορούν να αναπαρασταθούν σε XML ή RDF αλλά αποτελούν αντιπροσωπευτικό δείγμα γνώσης που είναι γραμμένη σε RDF Schema
- Επομένως, η γλώσσα RDFS καθιστά τη σημασιολογική πληροφορία προσπελάσιμη από υπολογιστές, σύμφωνα με το όραμα του ΣΙ



RDF: ΒΑΣΙΚΕΣ ΙΔΕΕΣ

Οι θεμελιώδεις έννοιες της RDF είναι οι πόροι (resources), οι ιδιότητες (properties), και οι προτάσεις (statements)

Πόροι

- Μπορούμε να θεωρήσουμε ότι ένας πόρος είναι ένα αντικείμενο, ένα «πράγμα» για το οποίο θέλουμε να μιλήσουμε
 - Οι πόροι μπορεί να είναι συγγραφείς, βιβλία, εκδότες, τοποθεσίες, άτομα, ξενοδοχεία, κ.ο.κ.
- Κάθε πόρος έχει μια διεύθυνση **URI** (**Uniform Resource Identifier**, **Ενιαίο Αναγνωριστικό Πόρων**)
 - Η διεύθυνση **URI** μπορεί να είναι μία διεύθυνση **URL** (**Uniform Resource Locator**, ή διεύθυνση Ιστού) ή κάποιο άλλο είδος μοναδικού αναγνωριστικού
 - Το αναγνωριστικό δεν επιτρέπει απαραίτητα την πρόσβαση σε έναν πόρο
 - Οι συνδυασμοί **URI** δεν έχουν οριστεί μόνο για τοποθεσίες στον Ιστό, αλλά και για αντικείμενα διαφόρων ειδών
 - Όπως τηλεφωνικούς αριθμούς, αριθμούς **ISBN** και γεωγραφικές τοποθεσίες
 - Γενικά θεωρούμε ότι μία διεύθυνση **URI** είναι το αναγνωριστικό ενός διαδικτυακού πόρου

Ιδιότητες και Προτάσεις

- Οι **ιδιότητες** είναι μια ειδική περίπτωση πόρων
 - Περιγράφουν σχέσεις μεταξύ πόρων, για παράδειγμα «γραμμένο από», «ηλικία», «τίτλος», κ.ο.κ.
- Οι ιδιότητες στην **RDF** καθορίζονται επίσης από διευθύνσεις **URI**
 - Και στην πράξη από **URL**
- Η ιδέα της χρήσης των **URI** για τον προσδιορισμό των «πραγμάτων» και των μεταξύ τους σχέσεων είναι πολύ σημαντική
 - Αυτή η επιλογή μας παρέχει με μία κίνηση μία μέθοδο καθολικής, παγκόσμιας, μοναδικής ονομασίας
 - Η χρήση μιας τέτοιας μεθόδου μειώνει δραστικά το πρόβλημα της ομωνυμίας, που «μαστιρίζει» την αναπαράσταση των κατανεμημένων δεδομένων μέχρι τώρα
- Οι **προτάσεις** βεβαιώνουν τις ιδιότητες των πόρων
- Η πρόταση είναι μία τριάδα (**triple**) αντικειμένου-χαρακτηριστικού-τιμής, η οποία αποτελείται από έναν πόρο, μία ιδιότητα, και μία τιμή
 - Οι τιμές μπορούν να είναι είτε πόροι ή **λεκτικά** (*literals*)
 - Τα λεκτικά είναι ατομικές τιμές (αλφαριθμητικά, *strings*)

Τρεις Θεωρήσεις μιας Πρότασης (1/4)

- Ένα παράδειγμα πρότασης είναι

*David Billington is the owner of the Web page
<http://www.cit.gu.edu.au/~db>.*

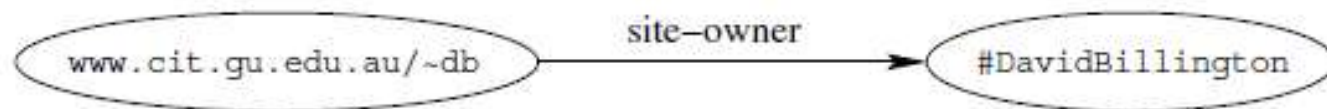
- Ο απλούστερος τρόπος ερμηνείας αυτής της πρότασης είναι να χρησιμοποιήσουμε τον ορισμό και να θεωρήσουμε την τριάδα

(<http://www.cit.gu.edu.au/~db>, <http://www.mydomain.org/site-owner>, #DavidBillington)

- Μπορούμε να θεωρήσουμε την τριάδα (x, P, y) ως ένα λογικό τύπο $P(x, y)$
 - Όπου το δυαδικό κατηγορημα P συσχετίζει το αντικείμενο x με το αντικείμενο y
- Στην πραγματικότητα, η *RDF* παρέχει μόνο δυαδικά κατηγορήματα (ιδιότητες)
 - Παρατηρήστε ότι η ιδιότητα “site-owner”, καθώς και τα δύο αντικείμενα, προσδιορίζονται από διευθύνσεις **URL**

Τρεις Θεωρήσεις μιας Πρότασης (2/4)

- Η δεύτερη θεώρηση βασίζεται σε γράφους
- Ο γράφος που αντιστοιχεί στην προηγούμενη πρόταση είναι ο παρακάτω
 - Είναι ένας κατευθυνόμενος γράφος με κόμβους και ακμές που έχουν ετικέτες
 - Οι ακμές κατευθύνονται από τον πόρο (το *υποκείμενο* της πρότασης) προς την τιμή (το *αντικείμενο* της πρότασης)



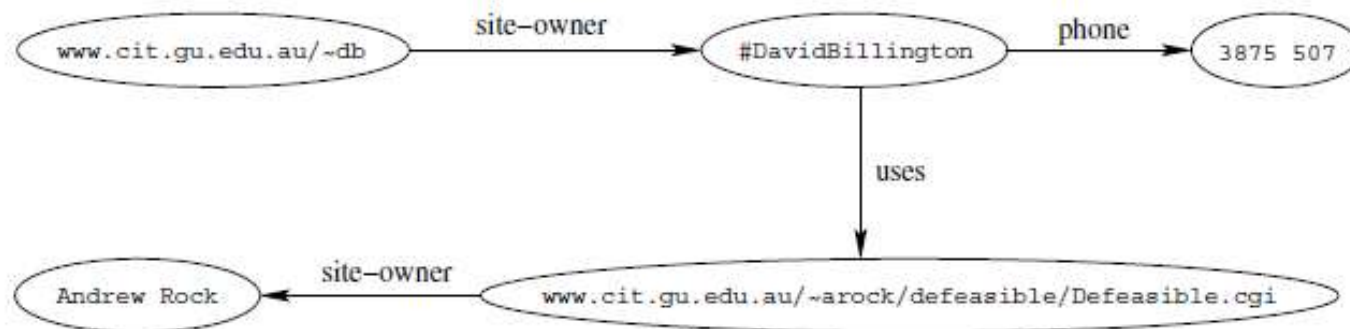
- Η τιμή μιας πρότασης μπορεί να είναι πόρος
 - Επομένως, ενδέχεται να συνδέεται και με άλλους πόρους

Τρεις Θεωρήσεις μιας Πρότασης (3/4)

- Θεωρείστε τις παρακάτω τριάδες:

(<http://www.cit.gu.edu.au/~db>, <http://www.mydomain.org/site-owner>, #DavidBillington)
(#DavidBillington, <http://www.mydomain.org/phone>, "3875507")
(#DavidBillington, <http://www.mydomain.org/uses>,
<http://www.cit.gu.edu.au/~arock/defeasible/Defeasible.cgi>)
(<http://www.cit.gu.edu.au/~arock/defeasible/Defeasible.cgi>,
<http://www.mydomain.org/site-owner>, "Andrew Rock")

- Η γραφική αναπαράστασή τους:



Τρεις Θεωρήσεις μιας Πρότασης (4/4)

- Η τρίτη αναπαράσταση βασίζεται στην XML
 - Ένα έγγραφο RDF αναπαρίσταται από ένα στοιχείο XML με την ετικέτα *rdf:RDF*
 - Το περιεχόμενο του στοιχείου αυτού είναι ένας αριθμός από περιγραφές, που χρησιμοποιούν ετικέτες *rdf:Description*
 - Κάθε περιγραφή διατυπώνει μία πρόταση σχετικά με κάποιον πόρο, ο οποίος προσδιορίζεται με τρεις διαφορετικούς τρόπους:
 - Ένα χαρακτηριστικό *about*, που αναφέρεται σε υπάρχοντα πόρο
 - Ένα χαρακτηριστικό *ID*, το οποίο δημιουργεί ένα νέο πόρο
 - Χωρίς ονομασία, γεγονός που δημιουργεί έναν ανώνυμο πόρο

```
<?xml version="1.0" encoding="UTF-16"?>
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:mydomain="http://www.mydomain.org/my-rdf-ns">
  <rdf:Description rdf:about="http://www.cit.gu.edu.au/~db">
  <mydomain:site-owner rdf:resource="#DavidBillington"/>
  </rdf:Description>
</rdf:RDF>
```

Υποστασιοποίηση

- Στην RDF είναι δυνατό να διατυπώσουμε προτάσεις σχετικά με προτάσεις:

Ο Grigoris πιστεύει ότι ο David Billington είναι ο δημιουργός (creator) της ιστοσελίδας <http://www.cit.gu.edu.au/~db>

- Μία τέτοια πρόταση μπορεί να χρησιμοποιηθεί για να περιγράψει πίστη ή εμπιστοσύνη σε άλλες προτάσεις
- Η λύση είναι να αντιστοιχίσουμε ένα μοναδικό αναγνωριστικό σε κάθε πρόταση, το οποίο θα χρησιμοποιήσουμε για να αναφερόμαστε στην πρόταση
- Η RDF επιτρέπει κάτι τέτοιο με τη χρήση ενός μηχανισμού υποστασιοποίησης (*reification mechanism*)
- Η κύρια ιδέα είναι η εισαγωγή ενός βοηθητικού αντικειμένου, π.χ. του *belief1*, και η συσχέτισή του με καθένα από τα τρία τμήματα της αρχικής πρότασης μέσω των ιδιοτήτων *subject* (υποκείμενο), *predicate* (κατηγορημα), και *object* (αντικείμενο)
 - Στο προηγούμενο παράδειγμα, το υποκείμενο του *belief1* θα ήταν *David Billington*, το κατηγορημα θα ήταν *δημιουργός* (*creator*), και το αντικείμενο θα ήταν <http://www.cit.gu.edu.au/~db>

Τύποι Δεδομένων (1/2)

- Θεωρείστε τον τηλεφωνικό αριθμό “3875507”
- Ένα πρόγραμμα που διαβάζει αυτό το μοντέλο δεδομένων RDF δεν μπορεί να γνωρίζει αν το λεκτικό “3875507” πρέπει να ερμηνευτεί ως ακέραιος, ή ως αλφαριθμητικό, ή αν είναι όντως ακέραιος, καθώς και αν η αναπαράστασή του είναι δεκαδική ή δεκαεξαδική
- Η εφαρμογή μπορεί να ξέρει πώς να ερμηνεύσει αυτόν τον πόρο, μόνο αν δοθεί ρητά σε αυτήν η πληροφορία ότι το λεκτικό αυτό αναπαριστά έναν αριθμό, καθώς επίσης και ο συγκεκριμένος αριθμός που υποτίθεται ότι αναπαριστά

Τύποι Δεδομένων (2/2)

- Η κοινή πρακτική είναι να παρέχεται τέτοια πληροφόρηση μέσω του συσχετισμού του λεκτικού με κάποιον τύπο δεδομένων, στην περίπτωση αυτή, με τον τύπο δεδομένων των δεκαδικών ή των ακεραίων αριθμών
- Τα τυποποιημένα λεκτικά (*typed literals*) χρησιμοποιούνται στην RDF για να παρέχουν αυτές τις πληροφορίες
 - Αν χρησιμοποιήσουμε ένα τέτοιο λεκτικό, μπορούμε να περιγράψουμε ότι η ηλικία του David Billington είναι ο ακέραιος αριθμός 27 με τη χρήση της τριάδας:

```
(#DavidBillington, http://www.mydomain.org/age, "27"^^http://www.w3.org/2001/XMLSchema#integer)
```

- Το παράδειγμα αυτό δείχνει δύο πράγματα:
 - Τη χρήση της σημειογραφίας ^^ για την υπόδειξη του τύπου ενός λεκτικού
 - Τη χρήση των προκαθορισμένων τύπων δεδομένων της γλώσσας XML Schema

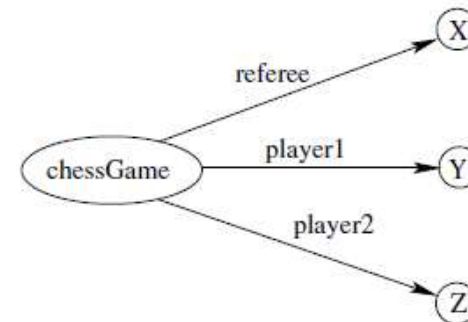
Κριτική της RDF (1/3)

- Η RDF χρησιμοποιεί μόνο δυαδικές ιδιότητες
- Συχνά χρησιμοποιούμε κατηγορήματα με περισσότερα από δύο ορίσματα
 - Τέτοια κατηγορήματα μπορούν να προσομοιωθούν με έναν αριθμό δυαδικών κατηγορημάτων
- Θα παρουσιάσουμε αυτή την τεχνική για ένα κατηγορημα *referee* με τρία ορίσματα
- Το προφανές νόημα του *referee(X,Y,Z)* είναι

Ο *X* είναι διαιτητής (*referee*) σε μία παρτίδα σκάκι ανάμεσα στους παίκτες (*players*) *Y* και *Z*

- Θα εισάγουμε τώρα ένα βοηθητικό πόρο *chessGame* και τα δυαδικά κατηγορήματα *ref*, *player1*, και *player2*
- Το κατηγορημα *referee(X,Y,Z)* μπορεί να αναπαρασταθεί ως εξής:

ref(chessGame, X)
player1(chessGame, Y)
player2(chessGame, Z)



Κριτική της RDF (2/3)

- Ένα άλλο πρόβλημα με τη χρήση της RDF αφορά το χειρισμό των ιδιοτήτων
 - Όπως αναφέραμε, οι ιδιότητες αποτελούν ειδικό τύπο πόρων
 - Επομένως, οι ίδιες οι ιδιότητες μπορούν να χρησιμοποιηθούν ως το αντικείμενο σε μία τριάδα αντικειμένου-χαρακτηριστικού-τιμής (μία πρόταση)
 - Παρόλο που η δυνατότητα αυτή προσφέρει ευελιξία, είναι μάλλον ασυνήθιστη για τις γλώσσες μοντελοποίησης και μπορεί να προκαλέσει σύγχυση στους σχεδιαστές μοντέλων
- Επίσης, ο μηχανισμός υποστασιοποίησης είναι αρκετά ισχυρός και η χρήση του σε μία απλή γλώσσα όπως η RDF φαίνεται άστοχη
 - Η διατύπωση προτάσεων σχετικά με προτάσεις εισάγει ένα επίπεδο πολυπλοκότητας που δεν είναι απαραίτητο για το βασικό επίπεδο του ΣΙ
 - Αντιθέτως, θα φαινόταν πιο φυσικό να περιλαμβάνεται στα ισχυρότερα επίπεδα, τα οποία παρέχουν πιο πλούσιες δυνατότητες αναπαράστασης
- Τέλος, η σύνταξη της RDF που βασίζεται στην XML είναι κατάλληλη για επεξεργασία από υπολογιστές, αλλά δεν είναι ιδιαίτερα φιλική προς τους ανθρώπους

Κριτική της RDF (3/3)

- Συνοπτικά, η RDF έχει τις ιδιαιτερότητές της και δεν είναι η βέλτιστη γλώσσα μοντελοποίησης
 - Είναι όμως γεγονός ότι αποτελεί ήδη ένα ντε φάκτο πρότυπο
 - Στην ιστορία της τεχνολογίας, δεν υιοθετήθηκε πάντα η καλύτερη τεχνολογία
- Η θετική πλευρά είναι ότι η RDF έχει επαρκή εκφραστική ισχύ
 - Τουλάχιστον ως βάση για τη δημιουργία περισσότερων επιπέδων
- Τελικά ο ΣΙ δε θα προγραμματιστεί σε RDF, αλλά με φιλικά προς το χρήστη εργαλεία, τα οποία θα μεταφράζουν αυτόματα τις αναπαραστάσεις υψηλότερων επιπέδων σε RDF
- Η χρήση της RDF παρέχει το πλεονέκτημα ότι οι πληροφορίες αντιστοιχίζονται ξεκάθαρα με ένα μοντέλο



RDF: ΣΥΝΤΑΞΗ ΒΑΣΙΣΜΕΝΗ ΣΕ XML

RDF – Εισαγωγή (1/5)

```
<!DOCTYPE rdf:RDF [ <!ENTITY xsd "http://www.w3.org/2001/XMLSchema#"> ]>
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
  xmlns:uni="http://www.mydomain.org/uni-ns#">
  <rdf:Description rdf:about="949352">
    <uni:name>Grigoris Antoniou</uni:name> <uni:title>Professor</uni:title>
  </rdf:Description>
  <rdf:Description rdf:about="949318">
    <uni:name>David Billington</uni:name> <uni:title>Associate Professor</uni:title>
    <uni:age rdf:datatype="&xsd;integer">27</uni:age>
  </rdf:Description>
  <rdf:Description rdf:about="CIT1111">
    <uni:courseName>Discrete Mathematics</uni:courseName>
    <uni:isTaughtBy>David Billington</uni:isTaughtBy>
  </rdf:Description>
  <rdf:Description rdf:about="CIT1112">
    <uni:courseName>Concrete Mathematics</uni:courseName>
    <uni:isTaughtBy>Grigoris Antoniou</uni:isTaughtBy>
  </rdf:Description>
</rdf:RDF>
```

An RDF document consists of an *rdf:RDF* element, the content of which is a number of descriptions

RDF – Εισαγωγή (2/5)

- Μερικά σχόλια στο προηγούμενο παράδειγμα
 - Πρώτον, χρησιμοποιείται ο μηχανισμός χώρου ονομάτων της XML, αλλά με επεκτεταμένο τρόπο
 - Στην XML, οι χώροι ονομάτων χρησιμεύουν μόνο για την απαλοιφή της αμφισημίας
 - Στην RDF, οι εξωτερικοί χώροι ονομάτων υποτίθεται ότι είναι έγγραφα RDF που ορίζουν πόρους, οι οποίοι με τη σειρά τους χρησιμοποιούνται στο έγγραφο RDF που τους εισάγει
 - Αυτός ο μηχανισμός επιτρέπει την επαναχρησιμοποίηση των πόρων από άλλα άτομα, που μπορεί να αποφασίσουν να εισάγουν πρόσθετες δυνατότητες στους πόρους αυτούς
 - Το αποτέλεσμα είναι η εμφάνιση μεγάλων, κατανεμημένων συλλογών γνώσης

RDF – Εισαγωγή (3/5)

- Δεύτερον, το χαρακτηριστικό *rdf:about* του στοιχείου *rdf:Description* είναι ισοδύναμο ως προς το νόημα με το χαρακτηριστικό ID
 - Αλλά χρησιμοποιείται συχνά για να υποδηλώσει ότι το αντικείμενο για το οποίο διατυπώνεται η πρόταση έχει ήδη «οριστεί» αλλού
 - Ένα σύνολο προτάσεων RDF σχηματίζει απλώς ένα μεγάλο γράφο, συσχετίζοντας κάποια πράγματα με άλλα μέσω των ιδιοτήτων
 - Δεν υπάρχει η έννοια του «ορισμού» ενός αντικειμένου σε ένα σημείο και η αναφορά σε αυτό σε κάποιο άλλο σημείο
 - Παρόλα αυτά, είναι μερικές φορές χρήσιμο να υπονοείται στη σειριακά διατεταγμένη σύνταξη της XML ότι μία τοποθεσία στη σειριακή διάταξη είναι η τοποθεσία «ορισμού»
 - Ενώ οι άλλες τοποθεσίες δηλώνουν «πρόσθετες» ιδιότητες για ένα αντικείμενο που έχει «οριστεί αλλού»

RDF – Εισαγωγή (4/5)

- Στην πραγματικότητα, το προηγούμενο παράδειγμα είναι ελαφρώς παραπλανητικό
 - Αν θέλαμε να είμαστε απολύτως σωστοί, θα έπρεπε να αντικαταστήσουμε όλες τις εμφανίσεις των ID για τα μαθήματα (courses) και το προσωπικό (staff), όπως τα 949352 και CIT1112, με αναφορές στον εξωτερικό χώρο ονομάτων:

```
<rdf:Description rdf:about="http://www.mydomain.org/uni-ns/# CIT1112">
```

- Τα περιεχόμενα των στοιχείων *rdf:Description* ονομάζονται *στοιχεία ιδιοτήτων (property elements)*

```
<rdf:Description rdf:about=" CIT1112 ">  
<uni:courseName>Concrete Mathematics</uni:courseName>  
<uni:isTaughtBy>Grigoris Antoniou</uni:isTaughtBy>  
</rdf:Description>
```

- Τα δύο στοιχεία *uni:courseName* και *uni:isTaughtBy* ορίζουν ζεύγη ιδιότητας-τιμής για το CIT1112
 - Η προηγούμενη περιγραφή αντιστοιχεί σε δύο προτάσεις RDF

RDF – Εισαγωγή (5/5)

- Τρίτον, το χαρακτηριστικό `rdf:datatype="&xsd;integer"` χρησιμοποιείται για να δηλώσει τον τύπο δεδομένων της τιμής της ιδιότητας της ηλικίας (age)
 - Παρόλο που έχουμε ορίσει το `"&xsd;integer"` ως το σύνολο τιμών της ιδιότητας αυτής, είναι απαραίτητο να υποδεικνύουμε τον τύπο της τιμής της ιδιότητας κάθε φορά που αυτή χρησιμοποιείται
 - Έτσι, εξασφαλίζουμε ότι ο επεξεργαστής της RDF θα αναθέσει το σωστό τύπο της τιμής της ιδιότητας, ακόμα κι αν δεν έχει δει ποτέ τον αντίστοιχο ορισμό της RDF Schema
- Τα στοιχεία ιδιοτήτων μιας περιγραφής πρέπει να διαβάζονται συζευκτικά
 - Στο προηγούμενο παράδειγμα το υποκείμενο ονομάζεται *"Concrete Mathematics"* (Διακριτά Μαθηματικά) και διδάσκεται από τον *Grigoris Antoniou*

Το χαρακτηριστικό `rdf:resource` (1/2)

- Το προηγούμενο παράδειγμα δεν ήταν ικανοποιητικό ως προς την εξής άποψη:
 - Οι σχέσεις ανάμεσα στα μαθήματα (courses) και στους διδάσκοντες (lecturers) δεν είχαν οριστεί τυπικά, αλλά υπήρχαν εμμέσως μέσω της χρήσης του ίδιου ονόματος
 - Για έναν υπολογιστή, η χρήση του ίδιου ονόματος μπορεί να είναι απλά σύμπτωση
 - Π.χ. ο David Billington που διδάσκει το μάθημα CIT3112 μπορεί να μην είναι το ίδιο άτομο με αυτό που έχει ID 949318 που τυχαίνει να ονομάζεται David Billington

```
<rdf:Description rdf:about="CIT1111">
<uni:courseName>Discrete Mathematics</uni:courseName>
<uni:isTaughtBy rdf:resource="949318"/>
</rdf:Description>
<rdf:Description rdf:about="949318">
<uni:name>David Billington</uni:name>
<uni:title>Associate Professor</uni:title>
</rdf:Description>
```

- Αυτό που χρειαζόμαστε είναι ένας τυπικός ορισμός του γεγονότος ότι, π.χ., ο διδάσκων του μαθήματος CIT1111 είναι το μέλος του προσωπικού με αριθμό 949318, το όνομα του οποίου είναι David Billington
- Αυτό μπορεί να επιτευχθεί με χρήση του χαρακτηριστικού `rdf:resource`

Το χαρακτηριστικό `rdf:resource`

(2/2)

- Αν είχαμε ορίσει τον πόρο του υπαλλήλου με αριθμό *ID* 939318 στο έγγραφο RDF, χρησιμοποιώντας το χαρακτηριστικό ID αντί του `about`, θα έπρεπε να χρησιμοποιήσουμε το σύμβολο `#` μπροστά από τον αριθμό 949318 στην τιμή του `rdf:resource`:

```
<rdf:Description rdf:about="CIT1111">
  <uni:courseName>Discrete Mathematics</uni:courseName>
  <uni:isTaughtBy rdf:resource="#949318"/>
</rdf:Description>
<rdf:Description rdf:ID="949318">
  <uni:name>David Billington</uni:name>
  <uni:title>Associate Professor</uni:title>
</rdf:Description>
```

- Το ίδιο ισχύει και για τους πόρους που ορίζονται εξωτερικά: π.χ. αναφερόμαστε στον εξωτερικά ορισμένο πόρο CIT1111 χρησιμοποιώντας το

```
http://www.mydomain.org/uni-ns/#CIT1111
```

- Ως τιμή του `rdf:about`, όπου `www.mydomain.org/uni-ns/` είναι η διεύθυνση URI στην οποία μπορεί να βρεθεί ο ορισμός του μαθήματος CIT1111

Ένθετες Περιγραφές

- Οι περιγραφές είναι δυνατό να ορίζονται μέσα σε άλλες περιγραφές
- Π.χ. οι περιγραφές του προηγούμενου παραδείγματος μπορούν να αντικατασταθούν με την ακόλουθη, ένθετη περιγραφή:

```
<rdf:Description rdf:about="CIT1111">  
<uni:courseName>Discrete Mathematics</uni:courseName>  
<uni:isTaughtBy>  
<rdf:Description rdf:about="949318">  
<uni:name>David Billington</uni:name>  
<uni:title>Associate Professor</uni:title>  
</rdf:Description>  
</uni:isTaughtBy>  
</rdf:Description>
```

- Άλλα μαθήματα, όπως το CIT3112, μπορούν ωστόσο να αναφέρονται στον καινούριο πόρο 949318
- Με άλλα λόγια, παρόλο που μία περιγραφή μπορεί να ορίζεται μέσα σε μία άλλη περιγραφή, η εμβέλειά της είναι καθολική

Το στοιχείο `rdf:type`

- Στα παραδείγματα που είδαμε, οι περιγραφές εμπίπτουν σε δύο κατηγορίες: μαθήματα και διδάσκοντες
- Το γεγονός αυτό είναι ξεκάθαρο στους ανθρώπους-αναγνώστες, αλλά δεν έχει δηλωθεί πουθενά τυπικά, άρα δεν είναι προσπελάσιμο από υπολογιστές
- Είναι δυνατό να διατυπώσουμε τέτοιες προτάσεις στην RDF με τη χρήση του στοιχείου `rdf:type`
- Ακολουθούν δύο περιγραφές που περιλαμβάνουν πληροφορίες τυποποίησης:

```
<rdf:Description rdf:about="CIT111">
  <rdf:type rdf:resource="&uni;course"/>
  <uni:courseName>Discrete Mathematics</uni:courseName>
  <uni:isTaughtBy rdf:resource="949318"/>
</rdf:Description>
<rdf:Description rdf:about="949318">
  <rdf:type rdf:resource="&uni;lecturer"/>
  <uni:name>David Billington</uni:name>
  <uni:title>Associate Professor</uni:title>
</rdf:Description>
```

- Το `rdf:type` επιτρέπει την εισαγωγή δομής στο έγγραφο RDF

Σύντομη Σύνταξη (1/2)

- Είναι δυνατή η σύντμηση της σύνταξης των εγγράφων RDF
- Οι κανόνες απλοποίησης είναι οι εξής:
 1. Στοιχεία ιδιοτήτων, που βρίσκονται μέσα σε στοιχεία περιγραφής και δεν έχουν παιδιά, μπορούν να αντικατασταθούν από χαρακτηριστικά XML, όπως συμβαίνει και στην XML
 2. Για στοιχεία περιγραφής με ένα στοιχείο τυποποίησης, μπορούμε να χρησιμοποιήσουμε το όνομα που ορίζεται εντός του στοιχείου `rdf:type` αντί του `rdf:Description`
- Π.χ. η περιγραφή:

```
<rdf:Description rdf:about="CIT1111">  
<uni:courseName>Discrete Mathematics</uni:courseName>  
<uni:isTaughtBy>  
<rdf:Description rdf:about="949318">  
<uni:name>David Billington</uni:name>  
<uni:title>Associate Professor</uni:title>  
</rdf:Description>  
</uni:isTaughtBy>  
</rdf:Description>
```

Σύντομη Σύνταξη(2/2)

- Σύμφωνα με τον κανόνα 1 στο *uni:courseName* είναι ισοδύναμη με την:

```
<rdf:Description rdf:ID="CIT1111"  
uni:courseName="Discrete Mathematics">  
<rdf:type rdf:resource="#uni;course"/>  
<uni:isTaughtBy rdf:resource="#949318"/>  
</rdf:Description>
```

- Και επίσης (λόγω του κανόνα 2) με την:

```
<uni:course rdf:ID="CIT1111"  
uni:courseName="Discrete Mathematics">  
<uni:isTaughtBy rdf:resource="#949318"/>  
</uni:course>
```

- Οι τρεις αυτές αναπαραστάσεις είναι απλά συντακτικές παραλλαγές της ίδιας πρότασης RDF
 - Αυτό σημαίνει ότι είναι ισοδύναμες σύμφωνα με το μοντέλο δεδομένων της RDF, παρόλο που έχουν διαφορετική σύνταξη XML

Στοιχεία-Υποδοχείς (1/6)

- Τα στοιχεία-υποδοχής (container elements) χρησιμεύουν στη συλλογή ενός αριθμού πόρων ή χαρακτηριστικών, για το σύνολο των οποίων επιθυμούμε να διατυπώσουμε μία πρόταση
 - Π.χ. μπορεί να θέλουμε να αναφερθούμε στα μαθήματα που διδάσκει ένας συγκεκριμένος διδάσκων
- Υπάρχουν τρεις τύποι υποδοχέων στην RDF:
 - **rdf:Bag**, ένας μη διατεταγμένος υποδοχέας, ο οποίος μπορεί να περιέχει πολλές εμφανίσεις (δεν ισχύει για σύνολο)
 - Τυπικά παραδείγματα: τα μέλη του συλλόγου διδασκόντων μιας σχολής και τα έγγραφα ενός φακέλου
 - **rdf:Seq**, ένας διατεταγμένος υποδοχέας, ο οποίος μπορεί να περιέχει πολλές εμφανίσεις
 - Τυπικά παραδείγματα: τα διάφορα μαθήματα ενός πτυχίου, τα θέματα μιας ημερήσιας διάταξης, μία αλφαβητική λίστα των μελών του προσωπικού
 - **rdf:Alt**, ένα σύνολο από εναλλακτικές (τιμές)
 - Τυπικά παραδείγματα: η αρχική θέση και οι θέσεις-είδωλα ενός εγγράφου, καθώς και οι μεταφράσεις ενός εγγράφου σε διάφορες γλώσσες
- Τα περιεχόμενα των στοιχείων-υποδοχέων είναι στοιχεία με ονόματα `rdf:_1`, `rdf:_2`, κ.ο.κ.

Στοιχεία-Υποδοχείς (2/6)

```
<rdf:RDF
xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:uni="http://www.mydomain.org/uni-ns#">
<uni:lecturer rdf:about="949352" uni:name="Grigoris Antoniou" uni:title="Professor">
<uni:coursesTaught>
<rdf:Bag> <rdf:_1 rdf:resource="CIT1112"/> <rdf:_2 rdf:resource="CIT3116"/> </rdf:Bag>
</uni:coursesTaught>
</uni:lecturer>
<uni:lecturer rdf:about="949318"
uni:name="David Billington"
uni:title="Associate Professor">
<uni:coursesTaught>
<rdf:Bag> <rdf:_1 rdf:resource="CIT1111"/> <rdf:_2 rdf:resource="CIT3112"/> </rdf:Bag>
</uni:coursesTaught>
</uni:lecturer>
<uni:course rdf:about="CIT1111" uni:courseName="Discrete Mathematics">
<uni:isTaughtBy rdf:resource="949318"/>
</uni:course>
<uni:course rdf:about="CIT1112" uni:courseName="Concrete Mathematics">
<uni:isTaughtBy rdf:resource="949352"/>
</uni:course>
</rdf:RDF>
```

Στοιχεία-Υποδοχείς (3/6)

- Τα στοιχεία-υποδοχείς έχουν ένα προαιρετικό χαρακτηριστικό ID, με το οποίο μπορεί να αναγνωρίζεται ο υποδοχέας και να γίνεται αναφορά σε αυτόν:
- Μία τυπική εφαρμογή των στοιχείων-υποδοχέων είναι η αναπαράσταση των κατηγορημάτων με περισσότερα από δύο ορίσματα
- Θεωρούμε ξανά το παράδειγμα *referee(X,Y,Z)*
 - όπου *X* είναι ο διαιτητής μιας παρτίδας σκακιού μεταξύ των παικτών *Y* και *Z*
 - Μία λύση είναι να διαχωριστεί ο διαιτητής *X* από τους παίκτες *Y* και *Z*

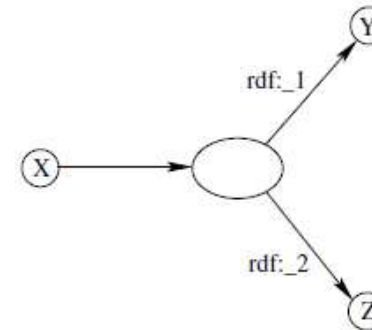
```
<uni:lecturer rdf:about="949318"
uni:name="David Billington"
uni:title="Associate Professor">
<uni:coursesTaught>
<rdf:Bag rdf:ID="DBcourses">
<rdf:_1 rdf:resource="CIT1111"/>
<rdf:_2 rdf:resource="CIT3112"/>
</rdf:Bag>
</uni:coursesTaught>
</uni:lecturer>
```

Στοιχεία-Υποδοχείς (4/6)

- Η λύση σε σύνταξη βασισμένη στην XML:

```
<referee rdf:about="..#X">
<players>
<rdf:Bag>
<rdf:li rdf:resource="..#Y"/>
<rdf:li rdf:resource="..#Z"/>
</rdf:Bag>
</players>
</referee>
```

- Η γραφική αναπαράσταση:



- Ο τύπος *rdf:Bag* ορίζει έναν ανώνυμο βοηθητικό πόρο
 - Επιλέξαμε το συγκεκριμένο τύπο, επειδή υποθέσαμε ότι δε γίνεται καμία διάκριση ανάμεσα στους παίκτες
 - Αν η σειρά ήταν όντως σημαντική (π.χ. ο 1^{ος} παίκτης είχε τα λευκά και ο 2^{ος} τα μαύρα) τότε θα χρησιμοποιούσαμε μια ακολουθία

Στοιχεία-Υποδοχείς (5/6)

- Ένας περιορισμός των υποδοχέων είναι ότι δεν υπάρχει τρόπος να κλείσουν
 - Δηλ. να δηλώσουμε ότι «αυτά είναι τα μέλη του υποδοχέα»
- Ενώ ένας γράφος μπορεί να περιγράψει ορισμένα από τα μέλη
 - Δεν υπάρχει τρόπος να αποκλειστεί η πιθανότητα ότι υπάρχει κάπου αλλού ένας άλλος γράφος που περιγράφει πρόσθετα μέλη
- Η RDF υποστηρίζει την περιγραφή ομάδων που περιέχουν μόνο τα καθορισμένα μέλη, με τη μορφή των συλλογών RDF
 - Μία συλλογή RDF (RDF collection) είναι μία ομάδα αντικειμένων που αναπαρίσταται ως δομή λίστας στο γράφο RDF
 - Αυτή η δομή λίστας κατασκευάζεται με τη χρήση ενός προκαθορισμένου λεξιλογίου συλλογής, το οποίο αποτελείται από
 - Τον προκαθορισμένο τύπο *rdf:List*
 - Τις προκαθορισμένες ιδιότητες *rdf:first* και *rdf:rest*
 - Τον προκαθορισμένο πόρο *rdf:nil*

Στοιχεία-Υποδοχείς (6/6)

```
<rdf:Description rdf:about="CIT2112">
  <uni:isTaughtBy>
    <rdf:List>
      <rdf:first> <rdf:Description rdf:about="949111"/> </rdf:first>
      <rdf:rest>
        <rdf:List>
          <rdf:first> <rdf:Description rdf:about="949352"/> </rdf:first>
          <rdf:rest>
            <rdf:List>
              <rdf:first> <rdf:Description rdf:about="949318"/> </rdf:first>
              <rdf:rest>
                <rdf:Description rdf:about="&rdf:nil"/>
              </rdf:rest>
            </rdf:List>
          </rdf:rest>
        </rdf:List>
      </rdf:rest>
    </rdf:List>
  </rdf:rest>
</rdf:Description>
```

- Το μάθημα CIT2112 διδάσκεται από τους καθηγητές που προσδιορίζονται από τους πόρους 949111, 949352, και 949318, και όχι από άλλα άτομα (γεγονός που υποδηλώνεται από το σύμβολο τερματισμού nil)
- Έχει οριστεί μια συντομευμένη σύνταξη, η οποία χρησιμοποιεί την τιμή "Collection" για το χαρακτηριστικό rdf:parseType:

```
<rdf:Description rdf:about="CIT2112">
  <uni:isTaughtBy rdf:parseType="Collection">
    <rdf:Description rdf:about="949111"/>
    <rdf:Description rdf:about="949352"/>
    <rdf:Description rdf:about="949318"/>
  </uni:isTaughtBy>
</rdf:Description>
```

Υποστασιοποίηση (1/2)

- Συχνά θέλουμε να διατυπώσουμε προτάσεις σχετικά με άλλες προτάσεις
 - Για να γίνει αυτό, πρέπει να είμαστε σε θέση να αναφερόμαστε σε μία πρόταση με χρήση κάποιου αναγνωριστικού
 - Η RDF επιτρέπει τέτοιου είδους αναφορές μέσω ενός μηχανισμού υποστασιοποίησης, ο οποίος μετατρέπει μία πρόταση σε πόρο
 - Π.χ. η περιγραφή:

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

Υποστασιοποιείται
ως εξής:

```
<rdf:Statement  
rdf:about="StatementAbout949352">  
<rdf:subject rdf:resource="949352"/>  
<rdf:predicate rdf:resource="&uni;name"/>  
<rdf:object>Grigoris Antoniou</rdf:object>  
</rdf:Statement>
```


Υποστυσιοποίηση (2/2)

- Note that *rdf:subject*, *rdf:predicate*, and *rdf:object* allow us to access the parts of a statement
- The ID of the statement can be used to refer to it
 - We can either write an *rdf:Description* if we don't want to talk about it further, or an *rdf:Statement* if we wish to refer to it
- If more than one property element is contained in a description element, the elements correspond to more than one statement
 - These statements can either be placed in a bag and referred to as an entity, or they can reify separately