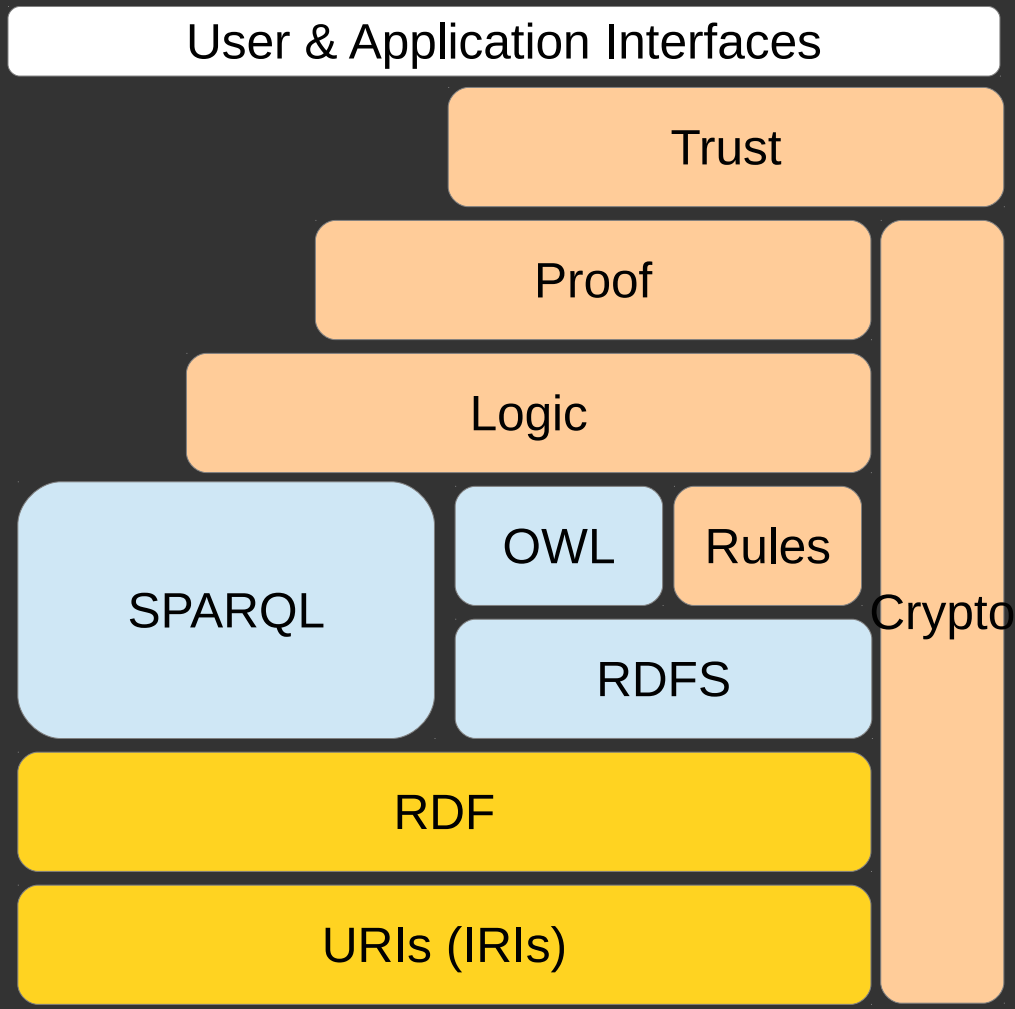
Προγραμματισμός Σημασιολογικού Ιστού

Ενότητα 6: Resource Description Framework (RDF)

Μ.Στεφανιδάκης

20-3-2018

Τα επίπεδα του Σημασιολογικού Ιστού



RDF: Το κύριο πρότυπο του Σημασιολογικού Ιστού, χρησιμοποιεί αναγνωριστικά URIs

Resource Description Framework (RDF)

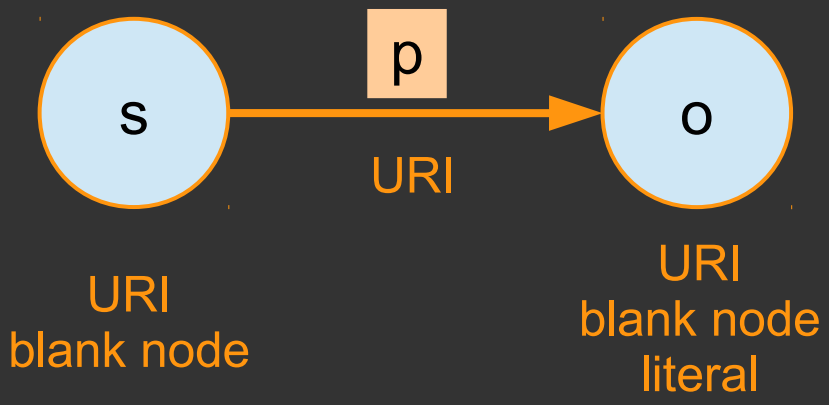
- Βασικό πρότυπο του Σημασιολογικού Ιστού
- Αν και λέμε συχνά “η RDF” (υπονοώντας “η γλώσσα RDF”)
- Στην πραγματικότητα είναι ένα μοντέλο οργάνωσης δεδομένων (ή γνώσης)
  - Που επιτρέπει να κάνουμε δηλώσεις (statements)
  - σε μορφή τριάδων (triples) (και του αντίστοιχου γράφου (graph))
  - σχετικά με οντότητες (entities)
  - οι οποίες συμβολίζονται με URIs
- Από τα πρώτα πρότυπα του Σημασιολογικού Ιστού (2004)
  - Με μια πρόσφατη επανέκδοση (RDF 1.1, Φεβρουάριος 2014)

Μοντέλο δεδομένων κατά το πρότυπο RDF

- Η RDF προσδιορίζει ένα μοντέλο (αφηρημένη σύνταξη – abstract syntax) βασισμένο στις τριάδες, ακριβώς όπως τις έχουμε δει ως τώρα
- υποκείμενο s – κατηγόρημα p – αντικείμενο o
  - ως μέρος γράφου με δύο κόμβους (s,o) και μία κατευθυνόμενη ακμή (από το s προς το o)
- οι κόμβοι μπορούν να είναι URIs (IRIs), ανώνυμοι (blank nodes) ή απλές τιμές (literals)

Το κατηγόρημα ως διμελής σχέση

- Το κατηγόρημα p δηλώνει μια ιδιότητα (property), μια διμελή σχέση (binary relation) μεταξύ υποκειμένου s και αντικειμένου o



URIs, blank nodes και literals

- Τα URIs δρουν ως σφαιρικά αναγνωριστικά οντοτήτων
  - Ένα URI δεν πρέπει ποτέ να αναφέρεται σε περισσότερες από μία οντότητα
  - Ένα URI, άπαξ και δημιουργηθεί, δεν πρέπει ποτέ να αλλάξει οντότητα, στην οποία αναφέρεται
  - Αν και δεν είναι υποχρεωτικό, ένα URI καλό θα ήταν να οδηγεί σε κάποιο έγγραφο στο web, με πληροφορία σχετική με την οντότητα του URI
- Οι ανώνυμοι κόμβοι (blank nodes) δεν αναγνωρίζουν οντότητες με ρητό όνομα
  - απλά λένε ότι κάτι (ανώνυμο) έχει τις περιγραφόμενες σχέσεις
- Οι σταθερές literal έχουν εξ'ορισμού τιμές που δεν αλλάζουν
  - Η RDF όμως τους προσδίδει τύπο δεδομένων (datatype)!

Τύποι δεδομένων: γιατί χρειάζονται;

- Τι θα απαντήσετε στο ερώτημα: `"1" + "2" = ?`
  - Σίγουρα 3!
- Η “μηχανή” όμως;
  - Κατά πάσα πιθανότητα: `"1" + "2" = "12"`
- Οι τύποι δεδομένων προσθέτουν ρητά τη σημασιολογία των `"1"` και `"2"`
  - που, ως άνθρωποι, δεν χρειαζόμαστε

RDF Datatypes

- Η RDF προσδίδει τύπο δεδομένων (datatype) στις απλές τιμές (literals)
  - που εμφανίζονται σε θέση αντικειμένου (o) στις τριάδες
- Ο τύπος δεδομένων συμβολίζεται επίσης με ένα URI
  - συνήθως της μορφής: http://www.w3.org/2001/XMLSchema#xxx
  - xxx είναι ο εκάστοτε τύπος δεδομένων
  - βασίζεται στο πρότυπο XML Schema
  - συντομογραφικά: xsd:xxx

RDF Datatypes (2)

- Η RDF περιγράφει μια σειρά συμβατών τύπων δεδομένων
  - xsd:string, xsd:boolean, xsd:integer, xsd:double, xsd:float,..
  - xsd:date, xsd:time, xsd:dateTime,..
  - κ.ο.κ..
- Η RDF χρησιμοποιεί επίσης το URI
  - http://www.w3.org/1999/02/22-rdf-syntax-ns#langString
  - για κείμενο με ένδειξη γλώσσας (π.χ. en, el, el-GR ..)

Literals και Datatypes

- Τι προσδίδει η σύνδεση ενός literal με έναν τύπο δεδομένων;
  - Προσδιορίζει τη μέθοδο χειρισμού της τιμής του literal
    - Πώς το κείμενο του literal (lexical form) θα μετατραπεί στην κατάλληλη τιμή
    - Η μετατροπή προσδιορίζεται από τον τύπο δεδομένων!
- Παράδειγμα: ο τύπος xsd:boolean
  - Διαθέτει δύο τιμές (value space): `{true,false}`
  - Δέχεται τα εξής strings (lexical space): {“true”, “false”, “1”, “0”}
  - Μετατρέπει ως εξής (Lexical-to-value mapping): $< \text{“}true\text{”} \rightarrow true >, < \text{“}false\text{”} \rightarrow false >, < \text{“}1\text{”} \rightarrow true >, < \text{“}0\text{”} \rightarrow false >$

Τύποι δεδομένων: πρακτική αντιμετώπιση

- Η RDF δεν απαιτεί από τις εφαρμογές να είναι σε θέση να χειριστούν τύπους δεδομένων
  - Αρκεί να μπορούν να χειριστούν απλά strings!
  - Αν συναντήσετε άγνωστο τύπο, δεν πρέπει να απορρίψετε τα δεδομένα αυτά
    - Φυσικά χάνετε σε σημασιολογική ισχύ
- Μπορείτε να χρησιμοποιήσετε και άλλους τύπους δεδομένων εκτός του XSD
  - Η εφαρμογή σας βέβαια θα πρέπει να τους αναγνωρίζει..

Πηγές RDF και συλλογές γράφων RDF

- RDF Source
  - Πηγή πληροφορίας RDF, περιέχει συλλογές γράφων RDF σε δεδομένη χρονική στιγμή
  - Οι γράφοι (και οι τριάδες) που περιέχει μπορούν να αλλάξουν με την πάροδο του χρόνου
- RDF Dataset
  - Μια συλλογή γράφων RDF, όπου
  - όλοι οι γράφοι εκτός από έναν αναγνωρίζονται με ένα URI (ή blank node) και ονομάζονται επώνυμοι γράφοι (named graphs)
  - Ο μοναδικός γράφος χωρίς σύνδεση με κάποιο URI είναι ο γράφος default

Επώνυμοι Γράφοι (Named Graphs)

- Εισαγωγή στο πρότυπο της RDF 1.1
  - Ένας μηχανισμός για τη διαίρεση των τριάδων RDF σε υποσύνολα
  - Χωρίς καθορισμένη σημασιολογία
  - Η χρήση τους προσδιορίζεται από την εκάστοτε εφαρμογή
- Στην τριάδα RDF προστίθεται ένα τέταρτο μέλος g (URI ή blank node)
  - Έχουμε πλέον μια τετράδα (quad)
- Για τον χωρισμό των τριάδων ανά προέλευση, χρονική στιγμή, προνόμια πρόσβασης κ.ο.κ.
- Και για δηλώσεις επί των τριάδων (reification)

Χώροι ονομάτων RDF

- Η RDF (και το συνοδευτικό RFDS που θα δούμε σε επόμενα) χρησιμοποιούν δικά τους (built-in) λεξιλόγια (vocabularies)
  - Για την “οντολογική” περιγραφή των διαφόρων οντοτήτων
  - Και για μια σειρά πρόσθετων βοηθητικών εννοιών (utilities)
- Οι χώροι ονομάτων για τα λεξιλόγια αυτά είναι
  - http://www.w3.org/1999/02/22-rdf-syntax-ns# (συντομογραφικό πρόθεμα rdf)
  - http://www.w3.org/2000/01/rdf-schema# (συντομογραφικό πρόθεμα rdfs)
- Παράδειγμα: `rdfs:label`
  - Χρησιμοποιείται για να συνδέσει μια ετικέτα αναγνώσιμη από τον άνθρωπο σε μια οντότητα
  - (http://ex.com/A, rdfs:label, ”Semantic Web”@en)

Αναπαράσταση δεδομένων RDF

- Η RDF εκτός από το αφηρημένο μοντέλο οργάνωσης, περιγράφει και διάφορες συγκεκριμένες συντάξεις (concrete syntaxes, μορφότυπα αποθήκευσης) των τριάδων σε αρχεία κειμένου
- Το απλούστερο από τα μορφότυπα αυτά ονομάζεται N-Triples
  - Ξεκίνησε ως “η γλώσσα των παραδειγμάτων” της RDF
  - Αλλά πολύ γρήγορα χρησιμοποιήθηκε για μαζική ανταλλαγή δεδομένων RDF
  - Πολύ απλή επεξεργασία, δεν χρειάζονται εξειδικευμένες βιβλιοθήκες
    - Κάθε γραμμή του αρχείου είναι ακριβώς μια τριάδα
  - Σήμερα υποστηρίζει Unicode χαρακτήρες (κωδικοποίηση utf-8)
    - Αρχικά, μόνο ASCII χαρακτήρες: όλοι οι άλλοι χρειάζονταν ειδική κωδικοποίηση

N-Triples: βασική σύνταξη

- Κάθε γραμμή του αρχείου περιέχει ακριβώς μία τριάδα
  - Στη μορφή `s p o .` (κενά/tab μετά από κάθε ένα s,p,o, στη συνέχεια ακολουθεί τελεία και newline)
    - Στη συνιστώμενη κανονική μορφή: ακριβώς ένα κενό
- Τα URIs γράφονται μεταξύ < και >
  - `<http://ex.com/A>`
  - σε πλήρη μορφή, χωρίς συντομογραφικά προθέματα
- Τα literals γράφονται μεταξύ ” και ”
  - `"Semantic Web"`
  - Προαιρετικά ακολουθεί ο τύπος δεδομένων ή η γλώσσα

```
"Semantic Web"@en
"1.663E-4"^^<http://www.w3.org/2001/XMLSchema#double>
```

N-Triples: βασική σύνταξη (2)

- Οι ανώνυμοι κόμβοι (blank nodes) έχουν πρόθεμα _:
  - _:b1234
  - Μετά το _: ακολουθεί η ετικέτα του ανώνυμου κόμβου
    - Η τελεία δεν μπορεί να είναι στην αρχή ή το τέλος της ετικέτας
    - Το - δεν μπορεί να είναι στην αρχή της ετικέτας

Η απλότητα της μορφής N-Triples

- Κάθε γραμμή περιέχει ακριβώς μια τριάδα και είναι αυτοδύναμη
  - Δεν χρειάζεται να αναλύσετε άλλες γραμμές, ούτε να περιμένετε να ολοκληρωθεί η σάρωση του αρχείου για να μάθετε την τριάδα της τρέχουσας γραμμής
  - Με άλλα λόγια: δεν χρειάζεται να κρατάτε μεγάλο μέρος του αρχείου στη μνήμη κατά την ανάλυσή του
    - Βασικό πλεονέκτημα όταν ένα αρχείο περιέχει εκατομμύρια τριάδες!
- Πολύ εύκολη σύνταξη
  - Η ανάλυση των τριάδων μπορεί να γίνει ακόμα και “στο χέρι”
  - Χωρίς πρόσθετες βιβλιοθήκες κώδικα