Μοντέλα Δεδομένων

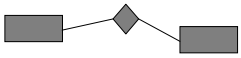
1. ΜΟΝΤΕΛΑ ΔΕΔΟΜΕΝΩΝ
   - **Μοντέλα**
     - **παραδοσιακά (Οντοτήτων/Συσχετίσεων, σχεσιακό)**
     - **ημι-δομημένα**
     - **XML**
   - Ερωτήσεις
2. ΑΡΧΙΤΕΚΤΟΝΙΚΕΣ
3. ΤΡΟΠΟΙ ΜΕΤΑΔΟΣΗΣ

Σχεδιασμός μιας ΒΔ

- ανάλυση ποιας πληροφορίας και της σχέσης ανάμεσα στα στοιχεία της
- περιγραφή της δομής - σχήμα σε διάφορους συμβολισμούς ή μοντέλα
- Μοντέλο Οντοτήτων - Συσχετίσεων

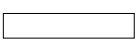
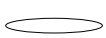
  γραφικό μοντέλο (εννοιολογικό)
- Μετατροπή σε σχεσιακό -> είσοδο σε ένα ΣΔΒΔ

Μοντέλο Οντοτήτων-Συσχετίσεων

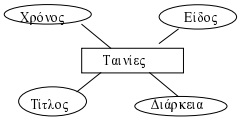


- **Οντότητα** ← **Σύνολο οντοτήτων - ανάπτυξη**
- Κάθε οντότητα έχει συγκεκριμένες ιδιότητες - **γνωρίσματα**
- Μια συγκεκριμένη οντότητα θα έχει μια τιμή για καθένα από τα γνωρίσματα

  Περιγράφει το σχήμα ή πρόθεση →
- **Τύπος οντοτήτων** ορίζει ένα σύνολο από οντότητες που έχουν τα ίδια γνωρίσματα
- Περιγράφεται από ένα όνομα και μια λίστα γνωρισμάτων

- **Τύπος Οντοτήτων** 
- **Γνώρισμα** 

Παράδειγμα



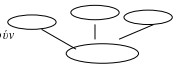
Gone with the Wind, 1939, 231, color

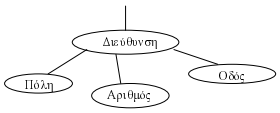
Τύποι Γνωρισμάτων

- απλά ή ατομικά

σύνθετα

τιμή: συνένωση των τιμών των απλών γνωρισμάτων που το αποτελούν
χρήσιμο όταν γίνεται αναφορά στα επιμέρους γνωρίσματα





Τύποι Γνωρισμάτων

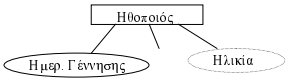
- μονότιμα

πλειότιμα *σύνολο από τιμές (κάτω-πάνω όριο)*

Τύποι Γνωρισμάτων

- παραγόμενα *μπορεί να υπολογιστεί από σχετιζόμενες οντότητες ή γνωρίσματα*

αποθηκευμένα



π.χ., αριθμός εργαζομένων σε ένα Τμήμα

Η τιμή **null**

Όταν μια οντότητα δεν έχει τιμή για ένα γνώρισμα

- Δεν υπάρχει δυνατή τιμή (not applicable)
- Υπάρχει δυνατή τιμή αλλά δεν είναι γνωστή
  - -- ξέρουμε ότι υπάρχει (missing)
  - -- δεν ξέρουμε αν υπάρχει (not known)

Η έννοια του **κλειδιού**

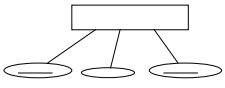
- **Υπερκλειδί**: σύνολο από ένα η περισσότερα γνωρίσματα που προσδιορίζουν μοναδικά μια οντότητα (superkey)

- **Υποψήφιο κλειδί**: ελάχιστο (μικρότερο αριθμό γνωρισμάτων) υπερκλειδί (candidate key)

- **Πρωτεύον κλειδί:** το υποψήφιο κλειδί που επιλέγουμε (primary key)

Η έννοια του **κλειδιού**

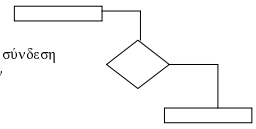
Ισχύει: υπερκλειδί $\supseteq$ κάθε υποψήφιο κλειδί



Παράδειγμα

Προσοχή: ο περιορισμός κλειδιού είναι μέρος του σχήματος

Συσχετίσεις



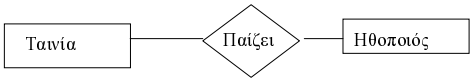
Τύποι συσχετίσεων R ορίζει μια σύνδεση (σχέση) μεταξύ n τύπων οντοτήτων

Μαθηματικά το R είναι ένα σύνολο από στιγμιότυπα συσχετίσεων r_i όπου κάθε r_i συνδέει n οντότητες

R υποσύνολο καρτεσιανού γινομένου

Συσχετίσεις

Παράδειγμα



Συχνά ως ένα σύνολο/πίνακα

Basic Instinct	Sharon Stone
Total Recall	Arnold Schwarzenegger
Total Recall	Sharon Stone

Βαθμός ενός τύπου συσχέτισης (degree): πλήθος των τύπων οντοτήτων που συμμετέχουν

Οι τύποι συσχετίσεων μπορεί να έχουν και γνωρίσματα

Λόγος πληθικότητας

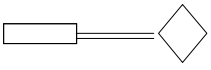
Για ένα τύπο συσχετίσεων

σε πόσες συσχετίσεις (στιγμιότυπα συσχετίσεων) μια οντότητα μπορεί να συμμετέχει

Για δυαδικές συσχετίσεις

- ένα-προς-ένα
- ένα-προς-πολλά
- πολλά-προς-ένα
- πολλά-προς-πολλά

Η συμμετοχή ενός συνόλου οντοτήτων E σε ένα σύνολο συσχετίσεων R είναι **ολική** αν κάθε οντότητα του E συμμετέχει τουλάχιστον σε μια συσχέτιση στο R



Αν κάποιες οντότητες του E δεν συμμετέχουν στο R τότε **μερική**

Μη **ισχυροί ή ασθενείς τύποι** οντοτήτων

Όταν μια οντότητα δεν έχει αρκετά γνωρίσματα για να σχηματίσει πρωτεύον κλειδί

Ανακεφαλαίωση

- Μοντελοποίηση του προβλήματος χρησιμοποιώντας το μοντέλο Οντοτήτων-Συσχετίσεων
- Δυο βασικά στοιχεία: Τύποι **Οντοτήτων** και Τύποι **Συσχετίσεων** ανάμεσα σε τύπους οντοτήτων
- Περιγράφουν το **σχήμα** - στιγμιότυπο
- Είδη γνωρισμάτων

Ανακεφαλαίωση

- Η έννοια του **κλειδιού**
- **Πληθικότητα** μιας συσχέτισης (για δυαδικές: 1-1, 1-M, M-N)
- **Ολική** συμμετοχή
- Ασθενής τύπος οντοτήτων: απαιτεί γνωρίσματα από έναν (ή περισσότερους) σχετιζόμενους τύπους οντοτήτων για τη διάκριση των οντοτήτων του

Προσδιορίζουσα συσχέτιση - προσδιορίζον τύπος οντοτήτων

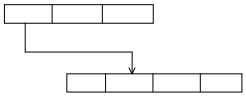
Κριτήρια Σχεδιασμού

Πρέπει να ακολουθεί πιστά τους περιορισμούς (specifications)

Αποφυγή Πλεονασμού

Απλότητα

Σχεσιακό Μοντέλο



Ένας απλός τρόπος αναπαράστασης δεδομένων: ένας διδιάστατος πίνακας που λέγεται σχέση

Γνωρίσματα

τίτλος	χρόνος	διάρκεια	είδος
Star Wars	1997	124	έγχρωμη
Mighty Ducks	1991	104	έγχρωμη
Wayne's World	1992	95	έγχρωμη

Σχήματα — **πρόθεση**

Σχήμα σχέσης R που δηλώνεται $R(A_1, A_2, \ldots, A_n)$ αποτελείται από ένα όνομα σχέσης και μια λίστα από γνωρίσματα.

Παράδειγμα - Ταινία(τίτλος, χρόνος, διάρκεια, είδος)

Βαθμός: το πλήθος των γνωρισμάτων

Σχήμα μιας σχεσιακής βάσης δεδομένων είναι ένα σύνολο από σχήματα σχέσεων

Πλειάδες

Οι γραμμές της σχέσης (εκτός της επικεφαλίδας) ονομάζονται **πλειάδες**.

Παράδειγμα: (Star Wars, 1997, 124, έγχρωμη)

(Wayne's World, 1992, 95, έγχρωμη)

έκταση ή κατάσταση

Μία **σχέση** r ή r(R) (ή ένα στιγμιότυπο r του σχήματος σχέσης R) είναι ένα *σύνολο* από πλειάδες.

Πεδίο Ορισμού

Κάθε στοιχείο μιας πλειάδας ατομικό.

Πεδίο ορισμού D: ένα σύνολο από ατομικές τιμές

(παράδειγμα: ακέραιοι, συμβολοσειρές - όχι εγγραφές, πίνακες, λίστες)

Κάθε γνώρισμα A_i είναι το όνομα ενός ρόλου που παίζει κάποιο πεδίο ορισμού D στο σχήμα σχέσης R. Το D λέγεται **πεδίο ορισμού του A_i** και συμβολίζεται με $dom(A_i)$.

Κάθε πλειάδα είναι μια διατεταγμένη λίστα από τιμές $<v_1, v_2, \ldots, v_n>$ όπου κάθε τιμή v_i είναι ένα στοιχείο του $dom(A_i)$ ή η ειδική τιμή null

Κάθε σχέση είναι ένα υποσύνολο του καρτεσιανού γινομένου:

$$r(R) \subseteq (dom(A_1) \times dom(A_2) \times \ldots \times dom(A_n))$$

Παρατηρήσεις

- Διάταξη των πλειάδων σε μια σχέση
- Διάταξη των γνωρισμάτων στο σχήμα σχέσης

Ταινία

Τίτλος	Έτος	Διάρκεια	Είδος

Ηθοποιός

Όνομα	Διεύθυνση	Έτος-Γέννησης

Παίζει

Όνομα-Ηθοποιού	Τίτλος	Έτος

Περιορισμός Πεδίου Ορισμού

Η τιμή κάθε γνωρίσματος A πρέπει να είναι μία *ατομική* τιμή από το πεδίο ορισμού αυτού του γνωρίσματος dom(A)

Περιορισμός Κλειδιού

Μια σχέση ορίζεται ως ένα σύνολο πλειάδων, άρα όλες οι πλειάδες πρέπει να είναι διαφορετικές.

Υποσύνολο γνωρισμάτων SK του σχήματος σχέσης R τέτοια ώστε σε κάθε στιγμιότυπο r(R) κανένα ζευγάρι πλειάδων δε μπορεί να έχει τον ίδιο συνδυασμό τιμών για τα γνωρίσματα αυτά, δηλαδή

για δυο διαφορετικές πλειάδες t_1 και t_2, $t_1[SK] \neq t_2[SK]$

Περιορισμός Κλειδιού

SK **υπερκλειδί - υποψήφιο κλειδί - (πρωτεύον) κλειδί**

υποψήφιο κλειδί Κ: υπερκλειδί με την ιδιότητα ότι αν αφαιρεθεί ένα οποιοδήποτε γνώρισμα Α από το Κ, το Κ' που προκύπτει δεν είναι υπερκλειδί

Κάθε σχέση τουλάχιστον ένα υπερκλειδί, ποιο;

Συμβολισμός: υπογραμμίζουμε τα γνωρίσματα του πρωτεύοντος κλειδιού

Ταινία

Τίτλος	Έτος	Διάρκεια	Είδος

Ηθοποιός

Όνομα	Διεύθυνση	Έτος-Γέννησης

Παίζει

Όνομα-Ηθοποιού	Τίτλος	Έτος

Περιορισμός Ακεραιότητας Οντοτήτων

Δε μπορεί η τιμή του πρωτεύοντος κλειδιού να είναι null.

Περιορισμός Αναφορικής Ακεραιότητας

Ορίζεται μεταξύ δύο σχημάτων σχέσεων

όταν μια πλειάδα μιας σχέσης **αναφέρεται** σε μια άλλη αυτή η άλλη πρέπει να υπάρχει

Ταινία

Τίτλος	Έτος	Διάρκεια	Είδος

Παίζει

Όνομα-Ηθοποιού	Τίτλος	Έτος

Περιορισμός Αναφορικής Ακεραιότητας

Ταινία

Τίτλος	Έτος	Διάρκεια	Είδος

Παίζει

Όνομα-Ηθοποιού	Τίτλος	Έτος

Ένα σύνολο από γνωρίσματα FK ενός σχήματος σχέσης R_1 είναι ένα **ξένο κλειδί** του R_1 αν

- τα γνωρίσματα του FK έχουν το ίδιο πεδίο με το πρωτεύον κλειδί PK ενός άλλου σχήματος R_2
- μια τιμή του FK σε μια πλειάδα t_1 της R_1 είτε εμφανίζεται ως τιμή του PK σε μια πλειάδα t_2 της R_2, δηλαδή $t_1[FK] = t_2[PK]$ είτε είναι null

Περιορισμός Αναφορικής Ακεραιότητας

Συνήθως προκύπτουν από συσχετίσεις μεταξύ οντοτήτων

Το ξένο κλειδί μπορεί να αναφέρεται στη δική του σχέση

Ηθοποιός

Όνομα	Διεύθυνση	Έτος-Γέννησης	Σύζυγος-Ηθοποιού

Περιορισμός Σημασιολογικής Ακεραιότητας

Παραδείγματα:

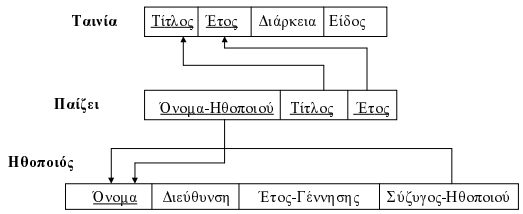
• ο μισθός ενός εργαζομένου δεν μπορεί να υπερβαίνει το μισθό του προϊσταμένου του

• ο μέγιστος αριθμός ωρών που ένας εργαζόμενος μπορεί να απασχοληθεί σε όλα τα έργα ανά εβδομάδα είναι 56.

Ένα **σχεσιακό σχήμα βάσης δεδομένων** είναι ένα σύνολο από σχήματα σχέσεων $\Sigma = \{R_1, R_2, \ldots, R_n\}$ και ένα σύνολο από περιορισμούς ακεραιότητας.

Ένα **στιγμιότυπο** μιας σχεσιακής βάσης δεδομένων ΒΔ του Σ είναι ένα σύνολο από στιγμιότυπα σχέσεων (σχέσεις) $ΒΔ = \{r_1, r_2, \ldots, r_n\}$ τέτοια ώστε κάθε r_i είναι ένα στιγμιότυπο του R_i που ικανοποιούν τους περιορισμούς ορθότητας (πεδίου ορισμού, κλειδιού, ακεραιότητας οντοτήτων, και αναφορικής ακεραιότητας)

Προσοχή: οι περιορισμοί ακεραιότητας πρέπει να ισχύουν *σε κάθε* στιγμιότυπο.



Ημι-δομημένα Δεδομένα (Semistructured Data)

• Δεν υπάρχει ξεχωριστή περιγραφή της δομής των δεδομένων - δεν υπάρχει σχήμα

• Αυτοπεριγραφικά - περιγράφουμε απευθείας τα δεδομένα χρησιμοποιώντας μιαν απλή σύνταξη

Παράδειγμα

{name: "Alan", tel: 2157786, email: "agb@abc.com"}

• Απλά ζεύγη: label : τιμή

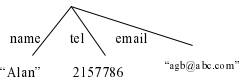
• Οι τιμές μπορεί να είναι δομές

```
{name: {first: "Alan", last: "Black"},
 tel: 2157786,
 email: "agb@abc.com"
}
```

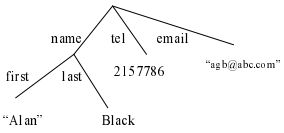
Ημι-δομημένα Δεδομένα

Γραφική αναπαράσταση

{name: 'Alan', tel: 2157786, email: 'agb@abc.com'}



{name: {first: 'Alan', last: 'Black'},
tel: 2157786,
email: 'agb@abc.com'
}



Ημι-δομημένα Δεδομένα

- Δεν απαιτούμε μοναδικά labels

{name: "Alan", tel: 2157786, tel: 3498765}

- Σύνολα πλειάδων

{ person:
{name: "Alan", tel: 2157786, email: "agb@abc.com"},
person:
{name: "Sara", tel: 06517870, email: "sara@cs.uoi.gr"},
person:
{name: "Fred", tel: 3357786, email: "fdr@nbc.com"},
}

Ημι-δομημένα Δεδομένα

- Δεν είναι απαραίτητο όλες οι πλειάδες να έχουν την ίδια δομή - συνήθως όμως κοντά σε κάποιο τύπο (παραλλαγές όπως επαναλαμβανόμενα πεδία, τιμές που λείπουν, κλπ)

{ person:
{name: {first: "Alan", last: "Black"},
tel: 2157786,
email: "agb@abc.com"
},
person:
{name: "Sara", tel: 06517870, email: "sara@cs.uoi.gr"},
person:
{name: "Fred", tel: 3357786, height: 183},
}

Ημι-δομημένα Δεδομένα

- Με κάθε δεδομένο - την περιγραφή του (π.χ., name, email) - αυτό - περιγραφικά

- Σειριοποίηση (serialization) μετατροπή δεδομένων σε byte streams - εύκολη μετάδοση και επανα-κατασκευή

- Σπατάλη χώρου (επανάληψη τύπου με κάθε δεδομένο)

- Interoperability

Σχεσιακά → Ημι-δομημένα Δεδομένα

Σχήμα: R1(A, B, C) και R2(C, D)

Στιγμιότυπο

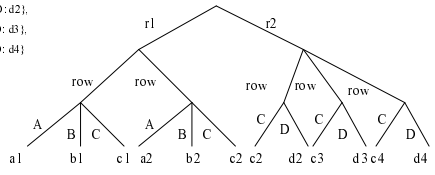
A	B	C
a1	b1	c1
a2	b2	c2

C	D
c2	d2
c3	d3
c4	d4

{r1: {row {A: a1, B: b1, C:c1},
row:{A: a2, B: b2, C: c2}
},
r2: {row: {C: c2, D: d2},
row: {C: c3, D: d3},
row: {C: c4, D: d4}
}
}

Σχεσιακά → Ημι-δομημένα Δεδομένα

{r1: {row {A: a1, B: b1, C:c1},
row:{A: a2, B: b2, C: c2}
},
r2: {row: {C: c2, D: d2},
row: {C: c3, D: d3},
row: {C: c4, D: d4}
}
}



Σχεσιακά → Ημι-δομημένα Δεδομένα

r1

A	B	C
a1	b1	c1
a2	b2	c2

r2

C	D
c2	d2
c3	d3
c4	d4

Εναλλακτικά

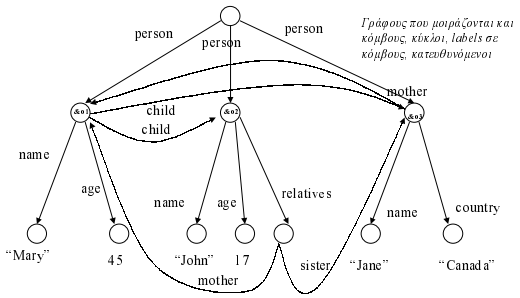
{r1: {row {A: a1, B: b1, C:c1}},
r1: {row:{A: a2, B: b2, C: c2}},
r2: {row: {C: c2, D: d2}},
r2: {row: {C: c3, D: d3}},
r2: {row: {C: c4, D: d4}}
}

Ημι-δομημένα Δεδομένα

```
{ person: &o1{name: "Mary",
              age: 45,
              child: &o2,
              child: &03
             },
person:  &o2{name: "John",
              age: 17,
              relatives: {mother: &o1,
                          sister: &o3}
             },
person:  &o3{name: "Jane",
              country: "Canada",
              mother: &o1
                          },
}
```

Χρήση object identifiers όταν θέλουμε να αναφερθούμε σε άλλα αντικείμενα

Ημι-δομημένα Δεδομένα



Ημι-δομημένα Δεδομένα

Θεωρούμε ότι το σύστημα αναθέτει oids

{a: &o1 {b:&o2 5}}
{a: {b:5}}
ισομορφικοί γράφοι

Αντικειμενοστραφή Μοντέλα

ODMG ODL (Object Definition Language)

```
class State
    (extent states)
{       attribute string scode;
        attribute string sname;
        attribute City capital;
        relationship set<City> cites-in
            inverse City:state-of;
}
```


Σύνταξη Ημι-δομημένων Δεδομένων

<ssd-expr> ::= <value> | oid<value | oid
<value> ::= atomicvalue | < complexvalue>
<complexvalue> ::= {lable : <ssd-expr>, …, lable: <ssd-expr>}

Το Μοντέλο Ανταλλαγής Δεδομένων (OEM)

Object **E**xchange **M**odel: Μοντέλο που έχει οριστεί για την ανταλλαγή δεδομένων μεταξύ ετερογενών συστημάτων

Ένα OEM αντικείμενο είναι μια πεντάδα (label, oid, type, value)

label (character string), *oid* (object's identifier), *type* (complex or atomic), *value* (if complex, a set (or list) of oids else, an atomic value of the atomic type)

Γράφοι με labels στους κόμβους

XML

XML

• Νέο standard του W3C ως συμπλήρωμα της HTML για ανταλλαγή δεδομένων στο Web

• Η HTML περιγράφει την αναπαράσταση και όχι το περιεχόμενο

• Η XML **περιγράφει το περιεχόμενο**, π.χ., tag: <people> </people>

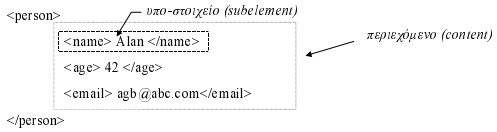
(για τον ορισμό της αναπαράστασης μπορεί να χρησιμοποιηθεί ένα ξεχωριστό stylesheet σε μια ειδική γλώσσα την XSL)

• Επιτρέπεται ο ορισμός νέων tags

• Ένα XML δεδομένο μπορεί να περιέχει (προαιρετικά) μια περιγραφή της γραμματικής του

Η Βασική Σύνταξη της XML

Element (στοιχείο) : ένα τμήμα κειμένου ανάμεσα σε δύο matching tags - μέσα σε ένα κείμενο μπορεί να έχουμε απλό κείμενο, άλλα στοιχεία ή και τα δύο



Η Βασική Σύνταξη της XML

```
<table>
  <description> People on the fourth floor </description>
  <people>
    <person>
        <name> Alan </name>
        <age> 42 </age>
        <email> agb@abc.com</email>
    </person>
    <person>
        <name> Patsy </name>
        <age> 36 </age>
        <email> ptn@abc.com</email>
    </person>
  </people>
</table>
```


Η Βασική Σύνταξη της XML

• Σε HTML πληροφορίες μόνο για την αναπαράσταση, π.χ.,

<h1> People on the fourth floor </h1>

  Alan , 42 years, <i> agb@abc.com </il>

  Patsy , 36 years, <i> ptn@abc.com </il>

Η Βασική Σύνταξη της XML

Γνωρίσματα (attributes)

Ορίζονται ως ζεύγη (όνομα, τιμή)

<product>
<name language="French">trompette six trous</name>
<price currency="Euro"> 420.12 </price>
</product>

Η τιμή ενός γνωρίσματος είναι πάντα ένα string και πρέπει να περικλείεται σε εισαγωγικά

Η Βασική Σύνταξη της XML

Γνωρίσματα (attributes)

Το όνομα ενός γνωρίσματος είναι μοναδικό σε κάθε tag ενώ τα ονόματα υπο-στοιχείων δεν είναι

<person>
<name> Alan </name>
<age> 42 </age>
<email> agb@abc.com</email>
</person>

<person> name="Alan" age="42" email= "agb@abc.com" />

Η Βασική Σύνταξη της XML

<book>
<title> The spy who came in from the cold </title>
<author> John <lastname> Le Carre </lastname><author>
<price currency="USD">5.59</price>
<review><author>Ben<author> Perhaps one of the finest .. <review>
<review><author>Jerry</author>An intriguing tale of .. </review>
<bestseller autority ="NY Times"/>
</book>

κενό στοιχείο

Η Βασική Σύνταξη της XML

Well-formed ακολουθεί τη σωστή σύνταξη

Η Βασική Σύνταξη της XML

Αναφορές (references)

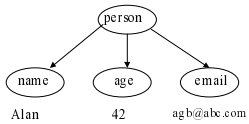
<state id="s2">
<scode NE </scode>
<sname> Nevada </sname>
</state>

<city id="c2">
<ccode> CCN <ccode>
<cname> Carson City </name>
<state-of idref="s2"/>
</city>

XMLκαι Ημι-δομημένα Δεδομένα

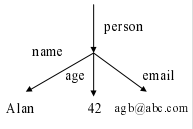
XML

<person>
<name> Alan </name>
<age> 42 </age>
<email> agb@abc.com</email>
</person>



Ημι-δομημένα Δεδομένα

{person: {name: "Alan", age: 42, email: "agb@abc.com"}}



XML και Ημι-δομημένα Δεδομένα

Η XML επιτρέπει να έχουμε και κείμενο και υπο-στοιχεία σε ένα στοιχείο

Η διάταξη έχει σημασία

DTDs

Document **T**ype **D**efinition **(DTD):** Περιγράφει μια κλάση από XML documents χρησιμοποιώντας μια γλώσσα που είναι ουσιαστικά μια γραμματική χωρίς συμφραζόμενα με αρκετούς περιορισμούς

DTDs

Ορισμός τύπου στοιχείου

<!ELEMENT book (title, author+, price, review*, bestseller?)>
<!ELEMENT title (#PCDATA)>
<!ELEMENT author (#PCDATA | lastname | firstname | fullename)*>
<!ELEMENT price (#PCDATAT>)
<!ATTLIST price currency CDATA "USD"
source (list|regular|sale) list
taxed CDATA #FIXED "yes">
<!ELEMENT bestseller EMPTY>
<!ATTLIST bestseller authority CDATA #required>

Parsed Character Data

* για ορισμένα ονόματα στοιχείων δεν υπάρχει ορισμός

DTDs

<!ELEMENT book (title, author+, price, review*, bestseller?)>
<!ELEMENT title (#PCDATA)>
<!ELEMENT author (#PCDATA | lastname | firstname | fullename)*>
<!ELEMENT price (#PCDATAT>)
<!ATTLIST price currency CDATA "USD"
source (list|regular|sale) list
taxed CDATA #FIXED "yes">
<!ELEMENT bestseller EMPTY>
<!ATTLIST bestseller authority CDATA #required>

τύπος

Default τιμή

DTDs

Ένα XML document που ικανοποιεί τους περιορισμούς ενός DTD ονομάζεται νόμιμο (valid) ως αναφορά το DTD

Document type declaration

<!DOCTYPE BOOKCATALOG SYSTEM "http://tt.com/bookcatalog.dtd">

DTDs ως Γραμματικές

- Μια γραμματική χωρίς συμφραζόμενα
- Επιτρέπεται η αναδρομή

<!ELEMENT node (leaf | (node, node))>
<!ELEMENT leaf (#PCDATA)>

DTDs ως Σχήματα

```
DOCTYPE db [
        <!ELEMENT db (r1*, r2*)>
        <!ELEMENT r1 (a, b, c)>
        <!ELEMENT r2 (c, d)>
        <!ELEMENT a (#PCDATA)>
        <!ELEMENT b (#PCDATA)>
        <!ELEMENT c (#PCDATA)>
        <!ELEMENT d (#PCDATA)>
]>
```

Order??

Προαιρετικά πεδία!

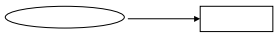
RDF

Resource **D**escription **F**ramework παρέχει μια γενική μέθοδο για την περιγραφή μετα-δεδομένων (metadata) για XML document

Παρέχει ένα μοντέλο και μια σύνταξη

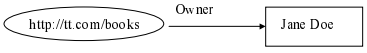
Απλό μοντέλο: τριάδες που αποτελούνται από ένα υποκείμενο (πόρος), ένα κατηγόρημα (ιδιότητα) και ένα αντικείμενο (τιμή της ιδιότητας)

Γραφική αναπαράσταση

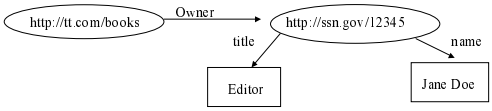


RDF

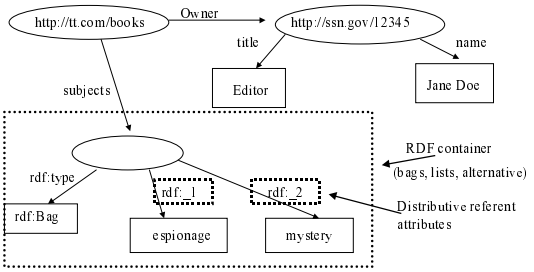
(Subject: http://tt.com/book; Predicate: Owner; Object: "Jane Doe")



•*Συγκεκριμένα περιγράφει πόρους (resources) που αναγνωρίζονται μέσω Uniform Resource Identifiers (URIs)*



RDF



RDF

```
<?xml version="1.0"?>
<rdf: RDF xmlns: rdf=' http://w3.org/TR/1999/PR-rdf-syntax#'
          xmlns: bs=' http://myschemas.org/books-schema#'>
  <rdf:Description about=' http://tt.co/books'>
          <bs:Owner rdf:resource=' http://ssn/gov/12345'/>
  </rdf:Description>
  <rdf:Description about=' http://ssn/gov/12345'>
          <bs:Name>Jane Doe </bs:Name>
          <bs:Title>Editor </bs:Title>
  </rdf:Description>
  <rdf:Description about=' http://tt.co/books'>
          <bs:Subjects>
            <rdf:Bag><rdf:li>espionage</rdf:li><rdf:li>mystery<rdf:li><rdf:Bag>
          </bs:Subects>
  </rdf:Description>
  <rdf: RDF>
```

Recources → ή ID="&o2"

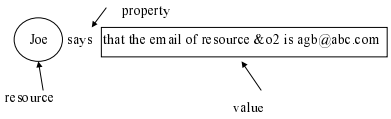
Statements (edges, properties) →

RDF

```
<?xml version="1.0"?>                          → XML declaration
<rdf: RDF xmlns:rdf=' http://w3.org/TR/1999/PR-rdf-syntax#'   → XML namespaces
          xmlns:bs=' http://myschemas.org/books-schema#'>
  <rdf:Description about='http://tt.co/books'>
          <bs:Owner rdf:resource='http://ssn/gov/12345'/>
  </rdf:Description>
  <rdf:Description about=' http://ssn/gov/12345'>
          …
  <rdf:Description about=' http://tt.co/books'>
          <bs:Subjects>
            <rdf:Bag><rdf:li>espionage</rdf:li><rdf:li>mystery <rdf:li><rdf:Bag>
          </bs:Subects>
  </rdf:Description>
<rdf: RDF>
```

RDF

Επίσης, δυνατότητα για "higher-order statements"



Stylesheets

Γραμματικές ειδικού σκοπού για την μετατροπή XML documents σε HTML documents ώστε να είναι δυνατή η αναπαράστασή τους

Δύο προτάσεις

- Cascading Style Sheets
- Extensible Stylesheet Language XSL

Η βασική ιδέα είναι να αντιστοιχήσουμε με κάθε τύπο στοιχείου μια αναπαράσταση

XSL

Extensible **S**tylesheet **L**anguage XSL: μια γλώσσα για μετασχηματισμό και μορφοποίηση XML documents

XSLTransformation Language (XSLT)

XSLTransformation Language (XSLT) Stylesheet: μια σειρά από κανόνες μετασχηματισμού (**transformation rules**) που παίρνουν ως είσοδο ένα πηγαίο XML document (source tree) και παράγουν ένα νέο XML document (result tree)

XSL

- Κάθε κανόνας αποτελείται από ένα πρότυπο (**pattern**) και ένα **template**
- Τα patterns ταιριάζουν με τους κόμβους του πηγαίου δέντρου, ενεργοποιούνται οι αντίστοιχες templates
- Η επεξεργασία αρχίζει από το template του κανόνα που ταιριάζει τη ρίζα

XSL

```
<xsl:stylesheet xmlns:xsl="http://w3.org/XSL?transform/1.0"
                xmlns="http://w3.org/TR/xhtml1"
                indent-result="yes">
<!-- Rule 1 ---> <xsl:template match = "/">
                  <html><head><title>Our New Catalog</title><head>
                        <body>
                            <xsl:apply-templates/>
                        </body>
                  </html>
                </xsl:template>
```

πρότυπο (pattern)

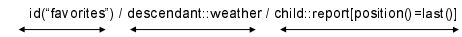
XPath

XML Path Language: μια γλώσσα για τον προσδιορισμό μονοπατιών - ορίζει τη σύνταξη και τη σημασιολογία εκφράσεων μονοπατιών (path expressions)

```
id("favorites")/descendant::weather/child::report[position()=last()]
```

- **Βήματα** (steps) που χωρίζονται με /
- Κάθε βήμα αποτελείται από μία **βάση** (basis) και ακολουθείται από μια προαιρετική λίστα από κατηγορήματα
- Η βάση καθορίζει μια επιλογή από κόμβους navigational - τα υπόλοιπα βήματα περιορίζουν την αρχική επιλογή
- Η βάση έχει τη μορφή AxisName::NodeTest

XPath



Βήμα 1

predefined function που επιλέγει το μοναδικό κόμβο που το γνώρισμα ID έχει τιμή favorites

Βήμα 2

με βάση descendants και node test weather που επιστρέφει όλους του απογόνους

Βήμα 3

με βάση child