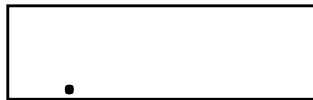


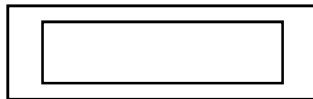
# Ασκήσεις - Επανάληψη

Βάσεις Δεδομένων  
2016-2017

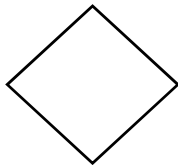
# E-R μοντέλα - Flashback



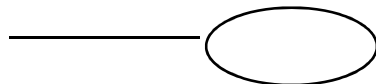
ENTITY SET



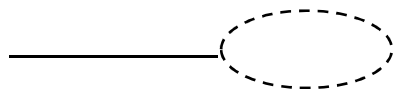
WEAK  
ENTITY SET



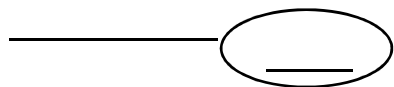
RELATIONSHIP  
SET



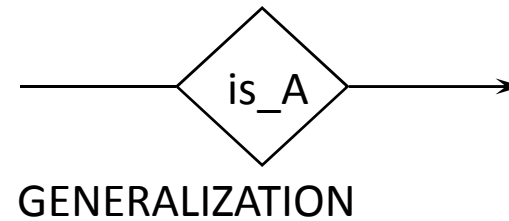
ATTRIBUTE



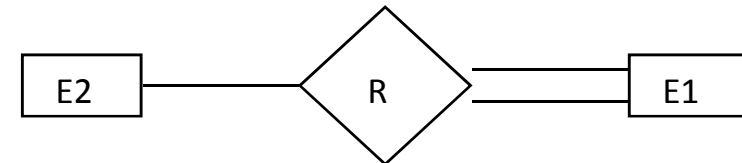
DERIVED  
Attribute



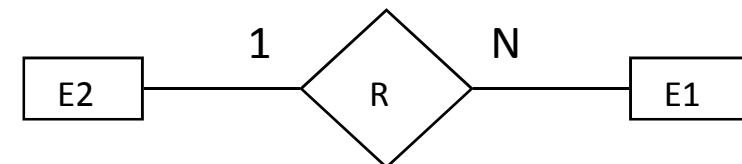
PRIMARY KEY



GENERALIZATION



Total PARTICIPATION of E1 in R1



Cardinality ratio 1:N for E2:E1 in R1

# E-R Μοντέλα

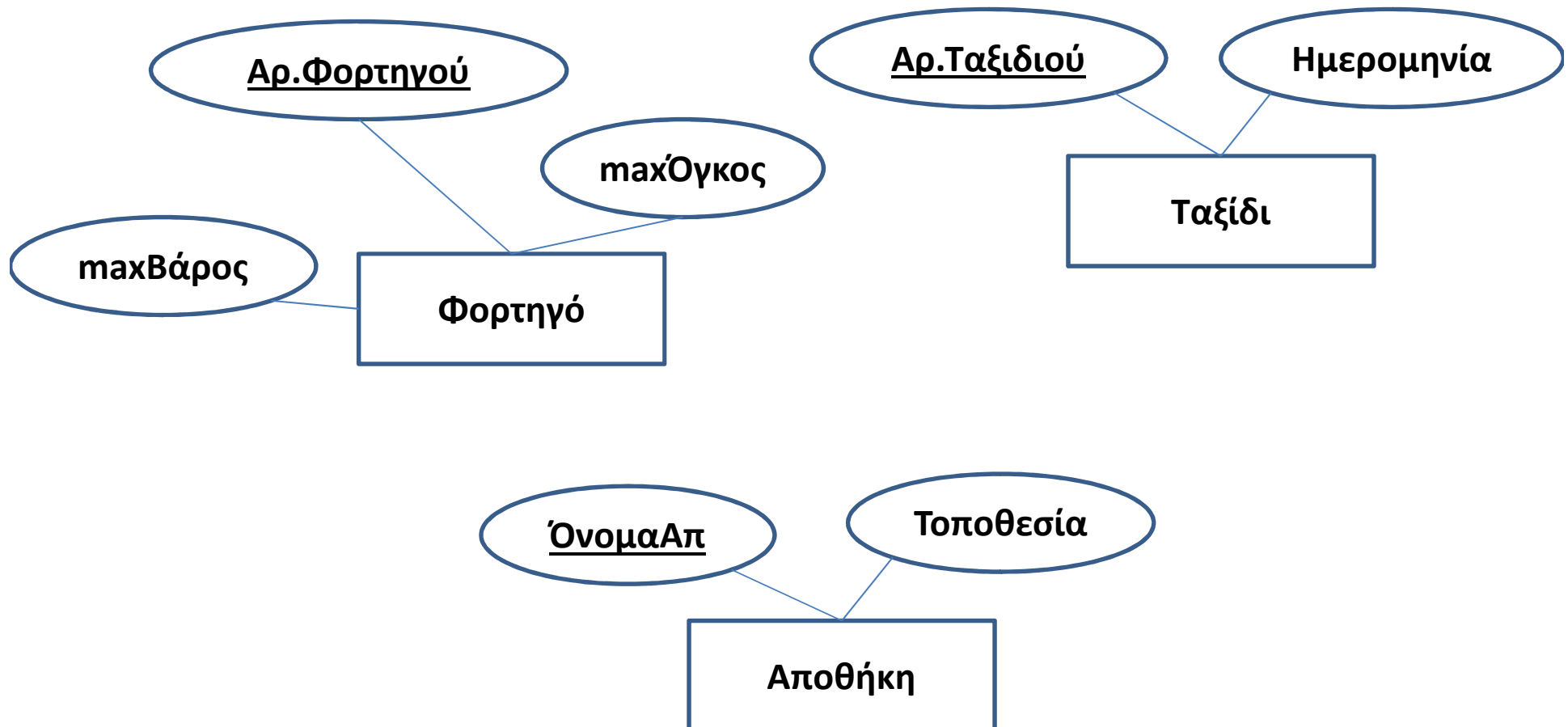
---

Μια εταιρεία μεταφορών, ονομαζόμενη ΚΟΥΒΑΛΑ, μεταφέρει αγαθά για μια μεγάλη αλυσίδα supermarkets, που ονομάζεται ΠΑΡΕΚΟΣΜΕ. Τα **φορτηγά** της ΚΟΥΒΑΛΑ παραλαμβάνουν αγαθά από **αποθήκες** και τα παραδίνουν σε συγκεκριμένα **καταστήματα** (supermarkets). Ένα φορτηγό είναι δυνατόν να μεταφέρει περισσότερες από μια “**αποστολές**” (σύνολο αγαθών) σε ένα απλό **ταξίδι** (ο αριθμός ταξιδιού προσδιορίζει μονοσήμαντα το ταξίδι) και παραδίδει αυτές τις αποστολές σε μια σειρά από καταστήματα.

Κάθε αποστολή αναγνωρίζεται από ένα ΑριθμοΑποστολής και έχει άλλες πληροφορίες, όπως όγκο αποστολής, βάρος, προορισμός, κλπ. Τα φορτηγά έχουν διαφορετικές δυναμικότητες και όσον αφορά τον όγκο αγαθών που χωρούν, αλλά και το βάρος που μπορούν να μεταφέρουν. Μια βάση δεδομένων είναι υπό σχεδιασμό για να κρατάει στοιχεία σχετικά με την χρήση των φορτηγών καθώς και τις αποστολές που γίνονται.

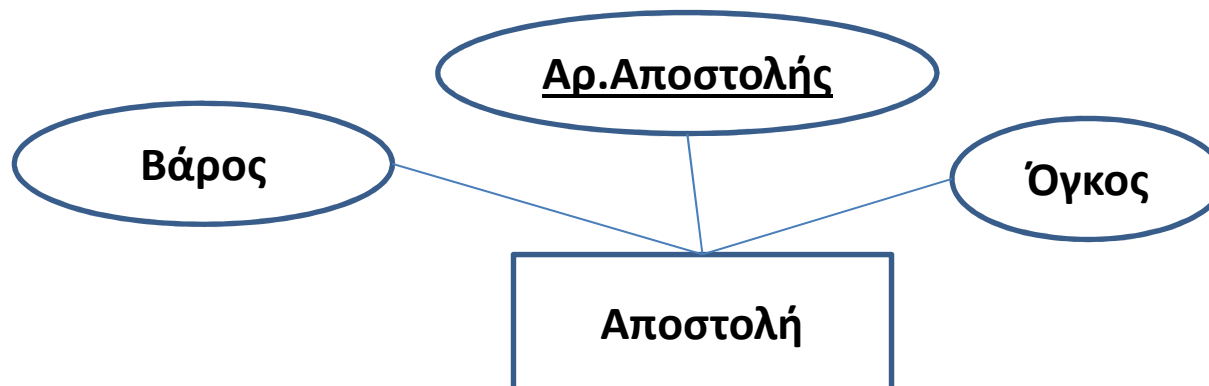
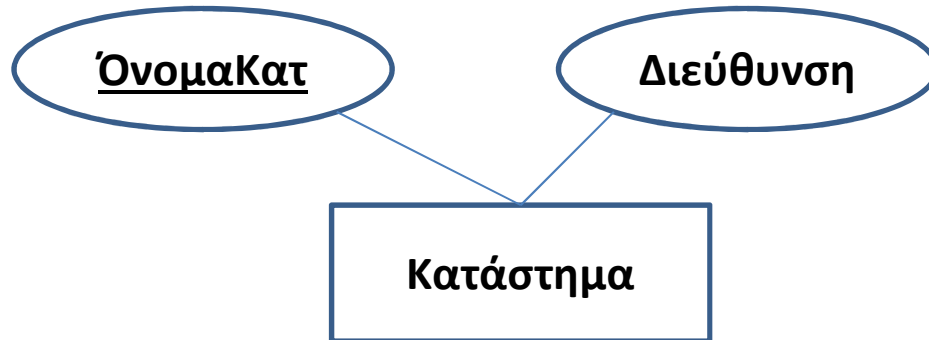
# Οντότητες (1)

---

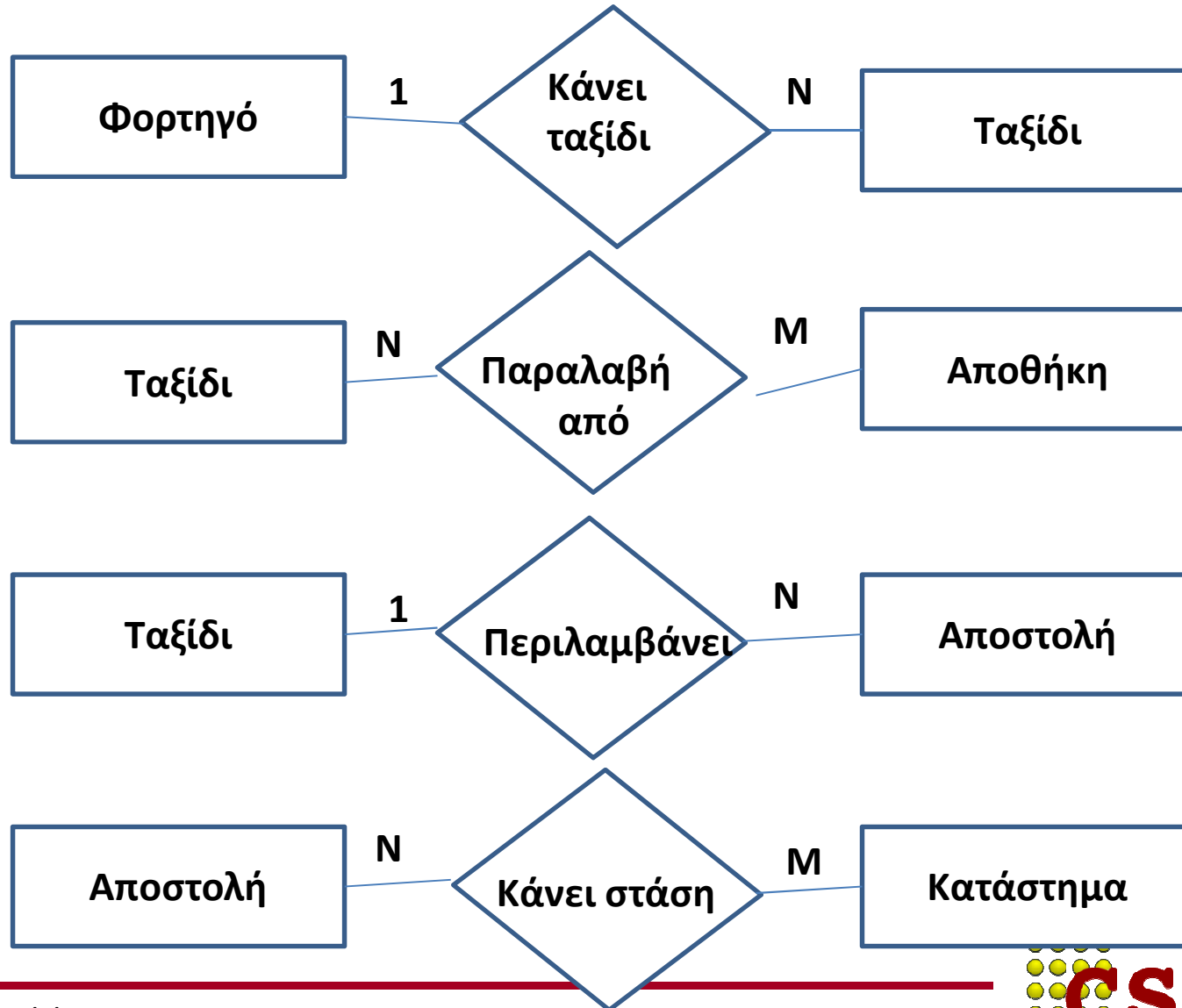


# Οντότητες (1)

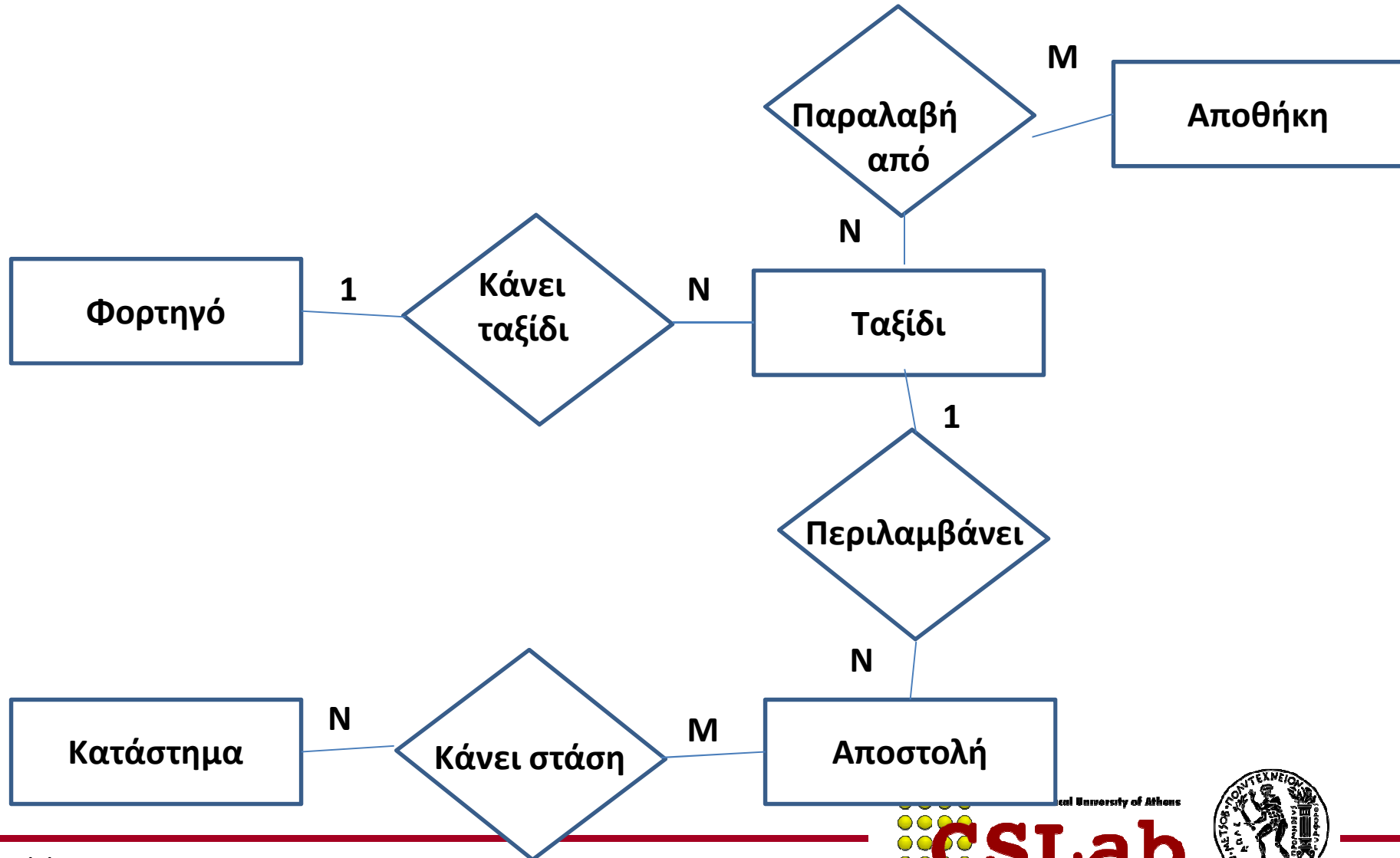
---



# Συσχετίσεις



# Τελικά



# Σχεσιακό μοντέλο

---

- ΦΟΡΤΗΓΟ (Αρ.Φορτηγού, ΜαχΒάρος, ΜαχΌγκος,...)
- ΑΠΟΘΗΚΗ ( ΌνομαΑπ, Τοποθεσία, ...)
- ΤΑΞΙΔΙ (Αρ.Ταξιδιού, Ημερομηνία, ..., **ΑριθμόςΦορτηγού**)
- ΑΠΟΣΤΟΛΗ (Αρ.Αποστολής, Βάρος, Όγκος, ..., **Αρ.Ταξιδιού**)
- ΚΑΤΑΣΤΗΜΑ (ΌνομαΚατ, Διεύθυνση, ...)
- ΠΑΡΑΛΑΒΗ\_ΑΠΟ(Αρ.Ταξιδιού, ΌνομαΑπ)
- ΚΑΝΕΙ\_ΣΤΑΣΗ(Αρ.Αποστολής,ΌνομαΚατ)



# Σχεσιακή Άλγεβρα - Flashback

---

- Select  $\sigma_c(R)$
- Project  $\rho_L(R)$
- Union  $R \cup S$
- set difference  $R - S$
- Cartesian product  $R \times S$
- Rename  $\rho_x(E)$
- Intersection  $R \cap S$
- natural join  $R \bowtie S$       theta join  $R \bowtie_c S$
- Division  $R / S$

# Σχεσιακή Άλγεβρα

---

**EMPLOYEE** ( SSN, Name, BirthDate, Address, Sex, Salary, SupSSN, DNumber)

**DEPARTMENT** ( DNumber, DName, MgrSSN, MgrStartDate)

**PROJECT** ( PNumber, PName, Location, DNumber)

**DEPT\_LOCATION** ( DNumber, DLocation )

**WORKS\_ON** ( SSN, Pnumber, HoursPW)

*Βρες τα ονόματα των υπαλλήλων που δουλεύουν σε όλα τα έργα*

*Υπάλληλοι σε Εργα =  $\pi_{PNumber,SSN}(WORKS\_ON)$*

*Όλα τα Εργα =  $\pi_{PNumber}(PROJECT)$*

*Υπάλλ. σε όλα τα εργα =  $\pi_{PNumber,SSN}(WORKS\_ON) \div \pi_{PNumber}(PROJECT)$*

Τελικό query =

$\pi_{Name}(EMPLOYEE) \bowtie_{SSN=SSN} (\pi_{Pnumber,SSN}(WORKS\_ON) \div \pi_{Pnumber}(PROJECT))$

# Σχεσιακή Άλγεβρα

---

**EMPLOYEE** ( SSN, Name, BirthDate, Address, Sex, Salary, SupSSN, DNumber)

**DEPARTMENT** ( DNumber, DName, MgrSSN, MgrStartDate)

**PROJECT** ( PNumber, PName, Location, DNumber)

**DEPT\_LOCATION** ( DNumber, DLocation )

**WORKS\_ON** ( SSN, Pnumber, HoursPW)

*Βρες τα ονόματα των υπαλλήλων που δε δουλεύουν σε κανένα έργο*

Όλοι οι υπάλληλοι =  $\pi_{SSN}(\text{EMPLOYEES})$

Όλοι οι υπάλληλοι που εργάζονται =  $\pi_{SSN}(\text{WORKS\_ON})$

Τελικό query =

$\pi_{\text{Name}}(\text{EMPLOYEE}) \bowtie_{SSN=SSN} (\pi_{SSN}(\text{EMPLOYEES}) - \pi_{SSN}(\text{WORKS\_ON}))$

# SQL

---

**EMPLOYEE** ( SSN, Name, BirthDate, Address, Sex, Salary, SupSSN, DNumber)

**DEPARTMENT** ( DNumber, DName, MgrSSN, MgrStartDate)

**PROJECT** ( PNumber, PName, Location, DNumber)

**DEPT\_LOCATION** ( DNumber, DLocation )

**WORKS\_ON** ( SSN, Pnumber, HoursPW)

*Για κάθε Τμήμα, βρες το όνομα του Τμήματος και το συνολικό κόστος σε μισθούς για τους εργαζομένους στο Τμήμα*

```
select Dname, sum(Salary)
```

```
from EMPLOYEE, DEPARTMENT
```

```
where EMPLOYEE.DNumber = DEPARTMENT.DNumber
```

```
group by DName
```

**EMPLOYEE** ( SSN, Name, BirthDate, Address, Sex, Salary, SupSSN, DNumber)

**DEPARTMENT** ( DNumber, DName, MgrSSN, MgrStartDate)

**PROJECT** ( PNumber, PName, PLocation, DNumber)

---

**DEPT\_LOCATION** ( DNumber, DLocation )

**WORKS\_ON** ( SSN, Pnumber, HoursPW)

*Βρες ονόματα υπαλλήλων που δουλεύουν τουλάχιστον σε ένα έργο στα Πατήσια, για το οποίο το τμήμα που το ελέγχει δε βρίσκεται στα Πατήσια*

select Name, Address

from EMPLOYEE

where exists (select \* from WORKS\_ON, PROJECT

where WORKS\_ON.Pnumber = PROJECT.PNumber and

EMPLOYEE.SSN = WORKS\_ON.SSN and

PLocation = 'patisia' ) and

not exists (select \* from DEPT\_LOCATION, WORKS\_ON, PROJECT

where WORKS\_ON.Pnumber = PROJECT.PNumber and

EMPLOYEE.SSN = WORKS\_ON.SSN and

DEPT\_LOCATION.Dnumber=PROJECT.DNumber and

DLocation = 'patisia' )

# Κανονικοποίηση

---

$R(A, B, C, D, E, G, H)$

$F = \{ AB \rightarrow C, AC \rightarrow B, AD \rightarrow E, B \rightarrow D, BC \rightarrow A, E \rightarrow G \}$

- Σύνολο εξαρτήσεων και ελάχιστη κανονική κάλυψη
- Έλεγχος BCNF
- Αν όχι, αποσύνθεση

$\{A, B, C\}$

$\{A, B, C, E, G\}$

$\{A, C, E, H\}$

# Ελάχιστη κανονική κάλυψη

---

- Διαισθητικά μια κανονική κάλυψη του  $F$  είναι ένα ελάχιστο σύνολο από συναρτησιακές εξαρτήσεις ισοδύναμο με το  $F$ , που δεν έχει πλεονάζουσες εξαρτήσεις ή πλεονάζοντα μέρη εξαρτήσεων.
- Ένα σύνολο  $F$  από  $FD$  είναι **ελάχιστη κάλυψη** αν ικανοποιεί τα παρακάτω:
  - (1) Κάθε εξάρτηση στο  $F$  είναι της μορφής:  $X \rightarrow A$ , όπου  $A$  είναι ένα απλό γνώρισμα
  - (2) Δεν μπορούμε να αποσύρουμε μια  $FD$  από το  $F$  και να εξακολουθούμε να έχουμε ένα σύνολο ισοδύναμο του  $F$
  - (3) Δεν μπορούμε να αντικαταστήσουμε μια  $FD$   $X \rightarrow A$  από το  $F$  με μια  $FD$   $Y \rightarrow A$ , όπου  $Y \subset X$  και να εξακολουθούμε να έχουμε ένα σύνολο ισοδύναμο του  $F$

# Λύση

---

$AB \rightarrow C$        $\{A, B, C\}$

$AC \rightarrow B$

$AD \rightarrow E$

$B \rightarrow D$

$BC \rightarrow A$

$E \rightarrow G$

$AB \rightarrow C$

$AC \rightarrow B$

$BC \rightarrow A$

*Κανένα γνώρισμα δεν είναι εξωτερικό  
(extraneous)*

*Άρα ελάχιστη κάλυψη*



# Λύση

---

$AB \rightarrow C$	$\{A, B, C, E, G\}$
$AC \rightarrow B$	$AB \rightarrow C$
$AD \rightarrow E$	$AC \rightarrow B$
$B \rightarrow D$	$BC \rightarrow A$
$BC \rightarrow A$	$E \rightarrow G$
$E \rightarrow G$	

*Κανένα γνώρισμα δεν είναι εξωτερικό  
(extraneous)  
Άρα ελάχιστη κάλυψη*

# Λύση

---

$AB \rightarrow C$        $\{A, C, E, H\}$

$AC \rightarrow B$

$AD \rightarrow E$

$B \rightarrow D$

$BC \rightarrow A$

$E \rightarrow G$

*Δεν υπάρχουν εξαρτήσεις – ελάχιστη κάλυψη*

# BCNF - Flashback

---

*Εάν και μόνο εάν για κάθε  $X \rightarrow Y$  που ισχύει τότε ένα εκ των δύο συμβαίνει:*

- *$X \rightarrow Y$  είναι τετριμμένη λειτουργική εξάρτηση*
- *Το  $X$  είναι υπερκλειδί*

# 2NF Flashback

---

Η R είναι σε 2NF αν είναι σε 1NF και κανένα μη-πρωτεύον Γνώρισμα δεν είναι μερικώς εξαρτώμενο από ένα υποψήφιο κλειδί

# Αποσύνθεση - Flashback

---

Το  $R$  σε  $R_1$  και  $R_2$

- $R_1$  και  $R_2$  είναι σε BCNF
- Lossless-join decomposition
- Dependency preserving

# Λύση

---

$AB \rightarrow C$        $\{A, B, C\}$

$AC \rightarrow B$

$AD \rightarrow E$

$B \rightarrow D$

$BC \rightarrow A$

$E \rightarrow G$

$AB \rightarrow C$

$AC \rightarrow B$

$BC \rightarrow A$

*Είναι σε BCNF διότι τα  $AB$ ,  $AC$  και  $BC$  είναι όλα υποψήφια κλειδιά στη σχέση*

# Λύση

---

$AB \rightarrow C$        $\{A, B, C, E, G\}$   
 $AC \rightarrow B$        $AB \rightarrow C$   
 $AD \rightarrow E$        $AC \rightarrow B$   
 $B \rightarrow D$        $BC \rightarrow A$   
 $BC \rightarrow A$        $E \rightarrow G$   
 $E \rightarrow G$

Υποψήφια κλειδιά:  $ABE, ACE, BCE$

Δεν είναι σε  $BCNF$ . Δεν είναι ούτε σε  $2NF$  γιατί υπάρχει μη πρωτεύον γνώρισμα που εξαρτάται από υποσύνολο των κλειδιών ( $E \rightarrow G$ )

$ABC$  και  $ABEG$  όμως  $ABEG$  δεν είναι  $BCNF$  γιατί το υποψήφιο κλειδί είναι  $ABE$  και υπάρχει  $E \rightarrow G$   
 $EG$  και  $ABE$

# Λύση

---

$AB \rightarrow C$        $\{A, C, E, H\}$

$AC \rightarrow B$

$AD \rightarrow E$

$B \rightarrow D$

$BC \rightarrow A$

$E \rightarrow G$

*Το κλειδί είναι η ίδια η ACEH  
Είναι σε μορφή BCNF*



# Ισοδυναμία συνόλου εξαρτήσεων

---

$$F1 = \{ A \rightarrow C, AB \rightarrow C, D \rightarrow AC, D \rightarrow E \}$$

$$F2 = \{ A \rightarrow BC, D \rightarrow AE \}$$

Δείξτε ότι είναι ισοδύναμα

1.  $F1^+ = F2^+$
2. Μετασχηματισμοί με κανόνες Armstrong

# Flashback

---

- Αν  $Y$  υποσύνολο του  $X$ , τότε  $X \rightarrow Y$  (αντανεκλαστικότητα)
- Αν  $X \rightarrow Y$  ισχύει, τότε  $XZ \rightarrow YZ$  (προσαύξηση)
- Αν  $X \rightarrow Y$  και  $Y \rightarrow Z$ , τότε  $X \rightarrow Z$  (μεταβατικότητα)
- Αν  $X \rightarrow Y$  και  $X \rightarrow Z$ , τότε  $X \rightarrow YZ$  (ένωση)
- Αν  $X \rightarrow YZ$ , τότε  $X \rightarrow Y$  και  $X \rightarrow Z$  (αποσύνθεση)
- Αν  $X \rightarrow Y$  και  $ZY \rightarrow P$ , τότε  $XZ \rightarrow P$  (ψευδομεταβατικότητα)

# Λύση

---

$$F1 = \{ A \rightarrow B, AB \rightarrow C, D \rightarrow AC, D \rightarrow E \}$$

$$F2 = \{ A \rightarrow BC, D \rightarrow AE \}$$

1.  $A \rightarrow B$

2.  $AB \rightarrow C$

3.  $D \rightarrow AC$

4.  $D \rightarrow E$

3α.  $D \rightarrow A$  και  $D \rightarrow C$  (αποσύνθεση)

2α.  $A \rightarrow C$  λόγω της 1, και τώρα το  $D \rightarrow C$  μπορεί να φύγει

$$\begin{array}{l} \bullet A \rightarrow B, \\ \bullet A \rightarrow C \\ \bullet D \rightarrow A \\ \bullet D \rightarrow E \end{array} \left. \vphantom{\begin{array}{l} \bullet A \rightarrow B, \\ \bullet A \rightarrow C \\ \bullet D \rightarrow A \\ \bullet D \rightarrow E \end{array}} \right\} A \rightarrow BC$$
$$\left. \vphantom{\begin{array}{l} \bullet D \rightarrow A \\ \bullet D \rightarrow E \end{array}} \right\} D \rightarrow AE$$

# Ευρετήρια - Βελτιστοποίηση

---

Suppliers(sid: integer, sname: char(20), city: char(20))

Supply(sid: integer, pid: integer)

Parts(pid: integer, pname: char(20), price: real)

```
SELECT S.sname, P.pname
```

```
FROM Suppliers S, Parts P, Supply Y
```

```
WHERE S.sid = Y.sid AND Y.pid = P.pid AND S.city = "Madison"  
      AND P.price < 1000
```

# Βελτιστοποίηση

---

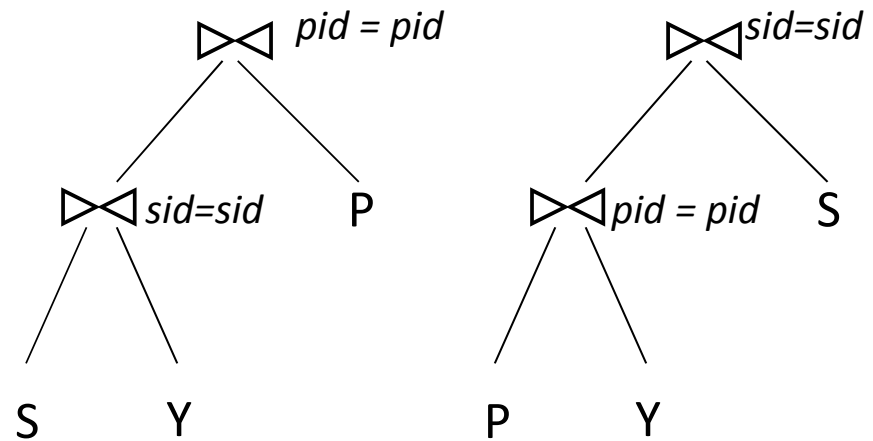
Τι πληροφορίες για τις παραπάνω Σχέσεις χρειάζεται ένας βελτιστοποιητής για να επιλέξει ένα καλό πλάνο εκτέλεσης για το συγκεκριμένο SQL Ερώτημα?

- *Ποια και τι τύπου ευρετήρια υπάρχουν στα S.sid, Y.sid, Y.pid, P.pid, S.city, P.price .*
- *Στατιστικά για την κατανομή τιμών*

# Βελτιστοποίηση

Πόσες διαφορετικές διατάξεις συνδέσμων (joins) ένας τέτοιος βελτιστοποιητής θα εξετάσει όταν αποφασίζει πως θα εκτελεστεί το ερώτημα και ποιες?

- ΜΟΝΟ αριστεροβαθείς  
 $((S \text{ join } Y) \text{ join } P)$  και  
 $((Y \text{ join } P) \text{ join } S)$



# Βελτιστοποίηση

---

Ποια ευρετήρια είναι ενδεχομένως χρήσιμα στον υπολογισμό του ερωτήματος.

- Ένα ταξινομημένο ευρετήριο στο *P.price* θα ήταν χρήσιμο (*range retrieval*)
- Επίσης, *B+* - δενδρικά ευρετήρια στα *Y.sid*, *Y.pid*, *P.pid* για τη χρήση του αλγορίθμου *sort-merge*

# Βελτιστοποίηση

---

Πώς επηρεάζονται τα πλάνα εκτέλεσης από την πρόσθεση ενός “ORDER BY *name*” στο ερώτημα?

- *Ο βελτιστοποιητής θα πρέπει να εξετάσει πλάνα εκτέλεσης που αφήνουν το *name* ταξινομημένο (παράπλευρη επιτυχία), αλλιώς θα πρέπει να έχει πλάνα εκτέλεσης που στο τέλος ταξινομούν απευθείας στο *name*.*