



EPL342 –Databases

Lecture 2: Introduction II

Database Users and Database System Concepts and Architecture

(Chapter 1.4-1.9, 2.1, Elmasri-Navathe 5ED)

Διδάσκων: Παναγιώτης Ανδρέου

<http://www.cs.ucy.ac.cy/courses/EPL342>

Περιεχόμενο Διάλεξης 2



Κεφάλαιο 1

- 1.4) Χρήστες Βάσεων
- 1.5) Άλλα Πλεονεκτήματα Χρήσης Βάσεων
- 1.6) Πότε να μην χρησιμοποιούμε Βάσεις
- 1.7) Ιστορική Εξέλιξη των Βάσεων & Νέες Κατευθύνσεις

Κεφάλαιο 2

- 2.1) Μοντέλα Δεδομένων, Σχήματα και Στιγμιότυπα

Χρήστες Βάσεων

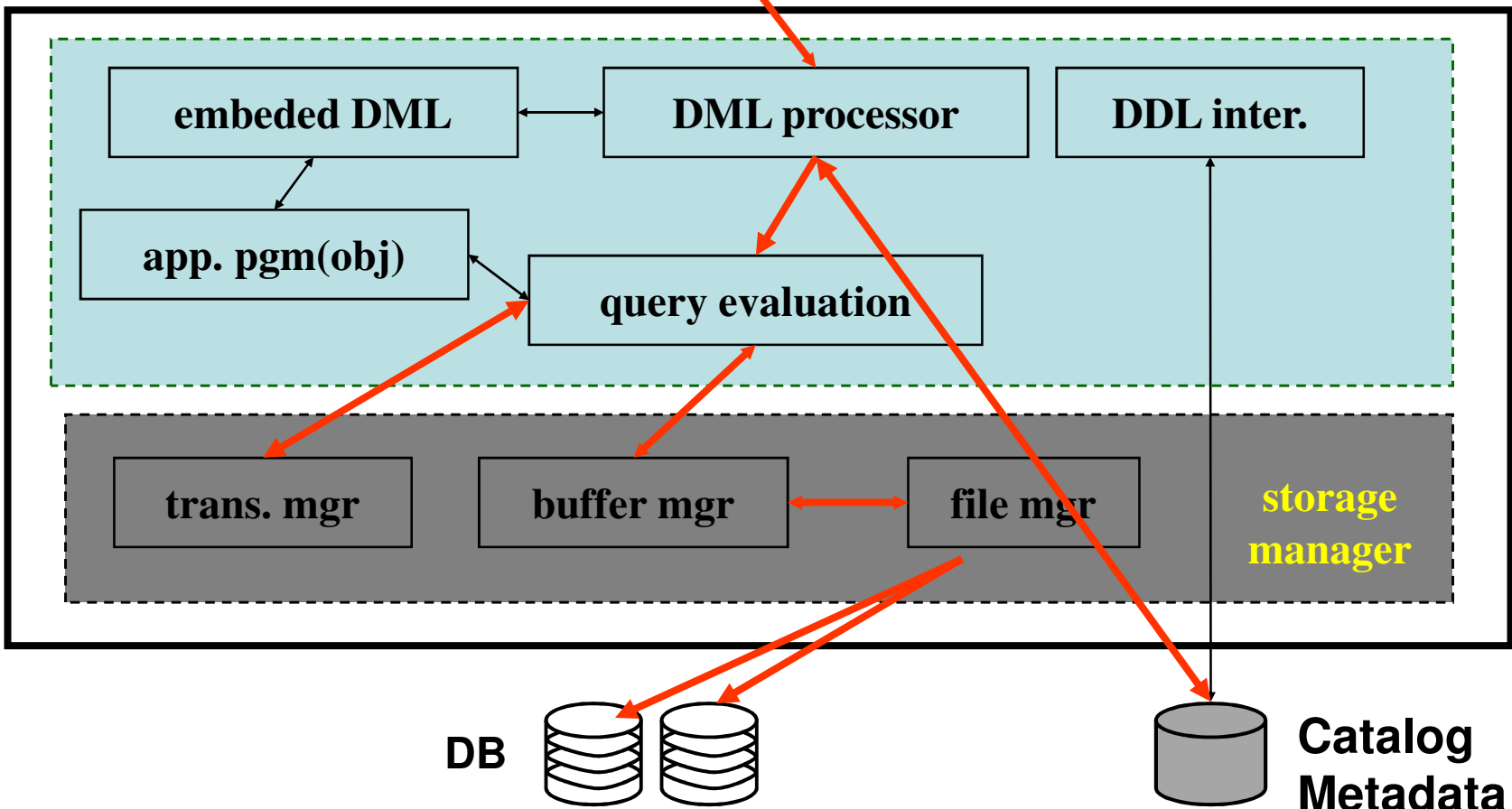


- Οι Χρήστες Βάσεων μπορούν να χωριστούν σε δυο κατηγορίες:
 - A) Αυτοί που Εργαζόμενοι Στο Προσκήνιο**
 - Οι χρήστες εκείνοι οι οποίοι έρχονται σε επαφή με τα **δεδομένα μιας βάσης**.
 - Κύριος στόχος του **ΕΠΛ342**
 - B) Αυτοί που Εργάζονται στο Παρασκήνιο**
 - Οι χρήστες εκείνοι οι οποίοι **δημιουργούν την τεχνολογία** πάνω στην οποία κτίζουν οι άλλοι εφαρμογές βάσεων δεδομένων.
 - Κύριος στόχος του **ΕΠΛ446**

Α. Χρήστες στο Προσκήνιο



- **Διαχειριστές Βάσεων (Database Administrators - DBA)**
 - Έλεγχος πρόσβασης, επίδοσης, ορθότητας, backups, κτλ.
 - Συνήθως κατέχουν επαγγελματικά πιστοποιητικά: π.χ., Microsoft MCDBA, Oracle Certified Master (OCM), κτλ.,
 - Είναι οι υπεύθυνοι για την πληροφορία σε ένα οργανισμό.
- **Σχεδιαστές / Προγραμματιστές Βάσεων (DB Designers / Developers)**
 - **Εννοιολογική Αναπαράσταση** των Απαιτήσεων του Χρήστη
 - Παραγωγή **Διαγραμμάτων** και **Δομών** της Βάσης για αναπαράσταση της πληροφορίας (Συνήθως από Αναλυτές)
- **Τελικοί Χρήστες (End Users)**
 - **Έμπειροι:** Αναλυτές Συστημάτων, Επιστήμονες, Μηχανικοί, κ.α., οι οποίοι κατέχουν ικανότητες διαχείρισης εφαρμογών & SQL.
 - **Άπειροι:** Ταμίες, Πωλητές, κ.α.. άτομα τα οποία χρησιμοποιούν προκατασκευασμένα λογισμικά που αλληλεπιδρούν με την βάση.²⁻⁴



Χρήστες

Αφελής

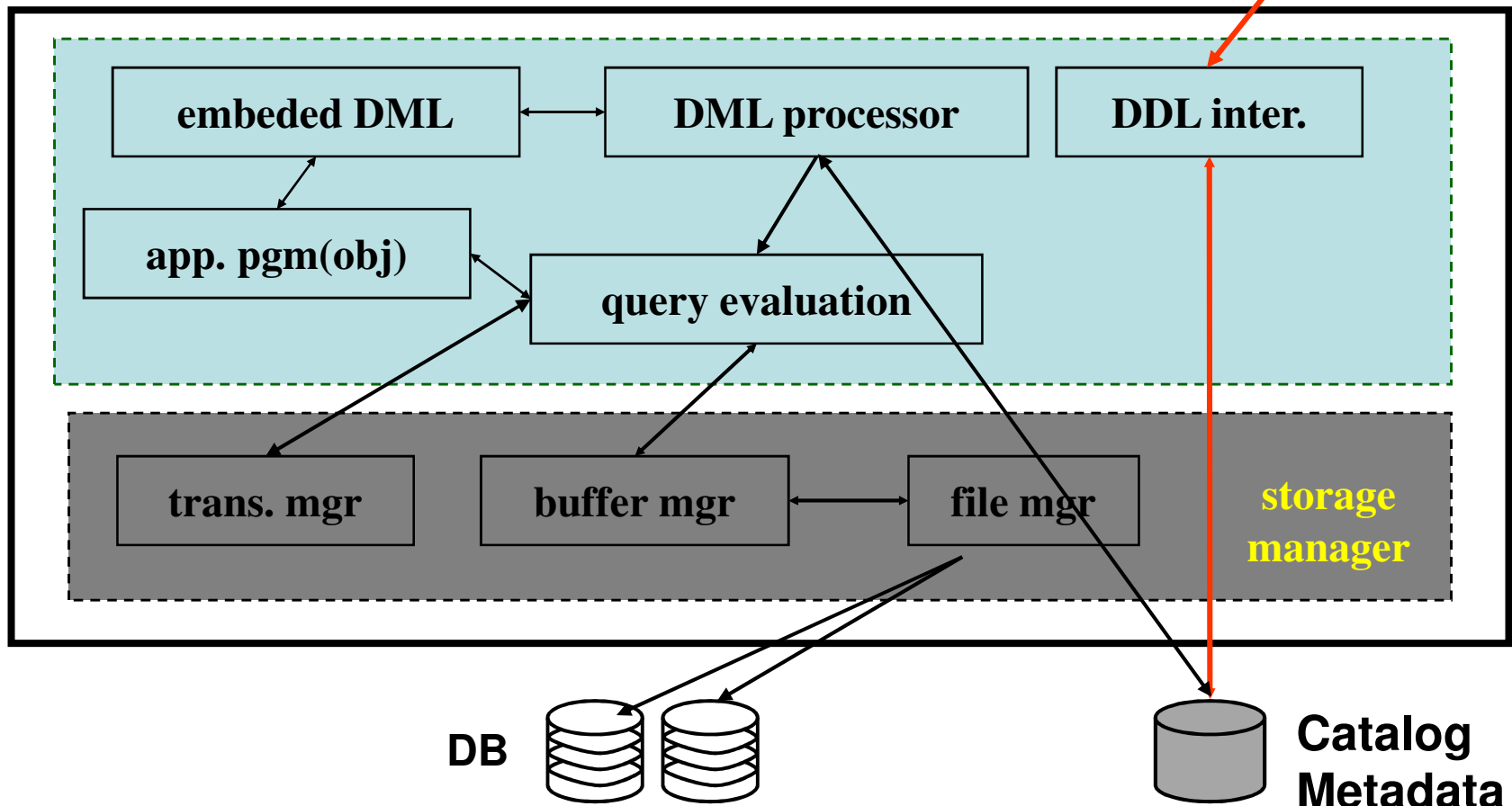
Απλός

Έμπειρος

db designer

appl. progr.

DBA



Β. Χρήστες στο Παρασκήνιο

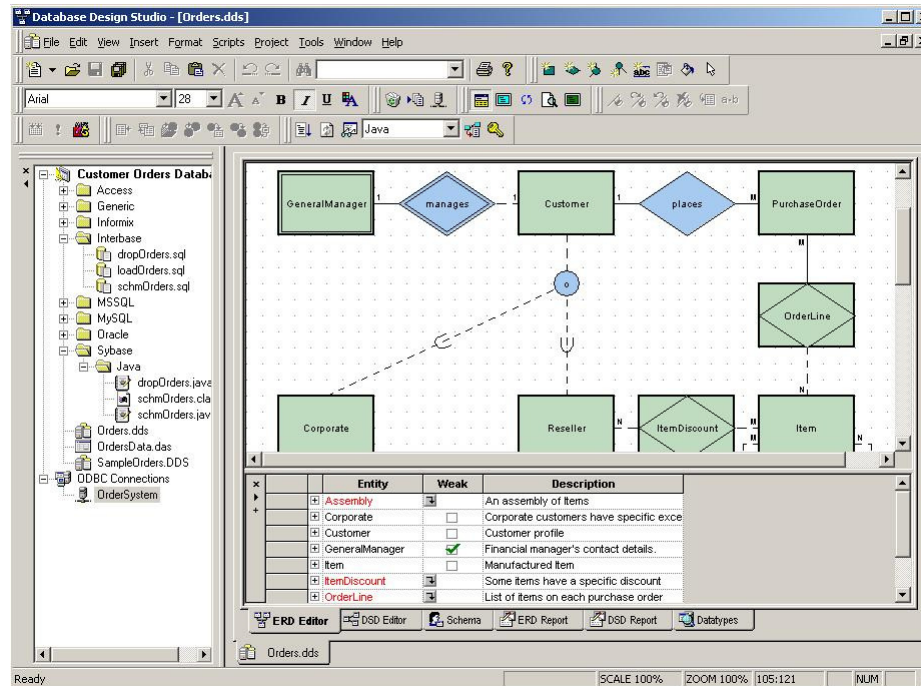


- **Σχεδιαστές και Προγραμματιστές Συστημάτων Βάσεων (Database System Designers and Developers)**
 - Υλοποίηση των Συστατικών (Modules) που απαρτίζουν μια βάση.
 - Προϋποθέτει καλή γνώση χαμηλού επιπέδου γλωσσών (π.χ., C, C++) και εννοιών συστημάτων (π.χ., Λειτουργικών, κτλ.)
 - Εργάζονται σε εταιρείες κατασκευής βάσεων δεδομένων: π.χ., Oracle, Microsoft, κτλ., ή οργανισμούς με εξειδικευμένες ανάγκες (CERN, NASA, κτλ.)
- **Προγραμματιστές Εργαλείων (Tool Developers)**
 - Παράγουν εργαλεία που χρησιμοποιούνται για την εννοιολογική αναπαράσταση, σχεδίαση και αύξηση της επίδοσης βάσεων.
 - Π.χ., Κατασκευαστές MS Visio, Oracle Designer, DDS Studio.
- **Διαχειριστές Συστημάτων (Συμβατικοί Administrators)**
 - Γενική Διαχείριση περιβάλλοντος λειτουργίας μιας βάσης

Β. Χρήστες στο Παρασκήνιο



Παράδειγμα Εργαλείου για την Ανάπτυξη Βάσεων Δεδομένων



Άλλα Πλεονεκτήματα Χρήσης Βάσεων



- Αναφέραμε ήδη **4 πλεονεκτήματα** των Βάσεων σε σχέση με **συμβατικά αρχεία**.
 1. Αυτό-περιγραφική Φύση της ΒΔ (Κατάλογος)
 2. Ανεξαρτησία Προγράμματος-Δεδομένων
 3. Παροχή Πολλαπλών Όψεων
 4. Διαμοιρασμός Φόρτου και Επεξεργασία Δοσοληψιών.
- Τώρα θα δούμε τι άλλα πλεονεκτήματα μπορεί να αξιοποιήσει ένας DBA.

Άλλα Πλεονεκτήματα Χρήσης Βάσεων



5) Έλεγχος των πλεονασμών (redundancy)

- Π.χ., Στο παράδειγμα του University μπορεί να υπάρχουν πολλές ανεξάρτητες βάσεις: π.χ., LIBRARY-DB, CS-DB, UCY-DB, etc.
- Αυτό δημιουργεί προβλήματα:
 - Σπατάλη Χώρου (Storage Space Waste)
 - Περιπτώσεις Ασυνέπειας (Inconsistencies)
 - Σπατάλη Χρόνου (Duplication of effort)
- Ιδανικά θα θέλαμε η πληροφορία να **αποθηκεύεται σε μια μόνο τοποθεσία.**
- **Πως; Κατά τη φάση της σχεδίασης και κανονικοποίησης.**

Άλλα Πλεονεκτήματα Χρήσης Βάσεων



6) Περιορισμός της Μη-Εξουσιοδοτημένης Προσπέλασης

- Δυνατότητα δημιουργίας ρόλων, ομάδων, κτλ. όπως σε συμβατικά Λειτουργικά Συστήματα.

7) Παροχή Εφεδρικών Αντιγράφων (Backup) και Μηχανισμών Ανάκαμψης (Recovery)

- Π.χ., Εάν έχουμε διακοπή ρεύματος (χωρίς UPS) δεν θα βρεθεί σε ασυνεπή κατάσταση η βάση δεδομένων.

8) Υποστήριξη Δομών Αποθήκευσης για Αποτελεσματική Επεξεργασία Επερωτήσεων

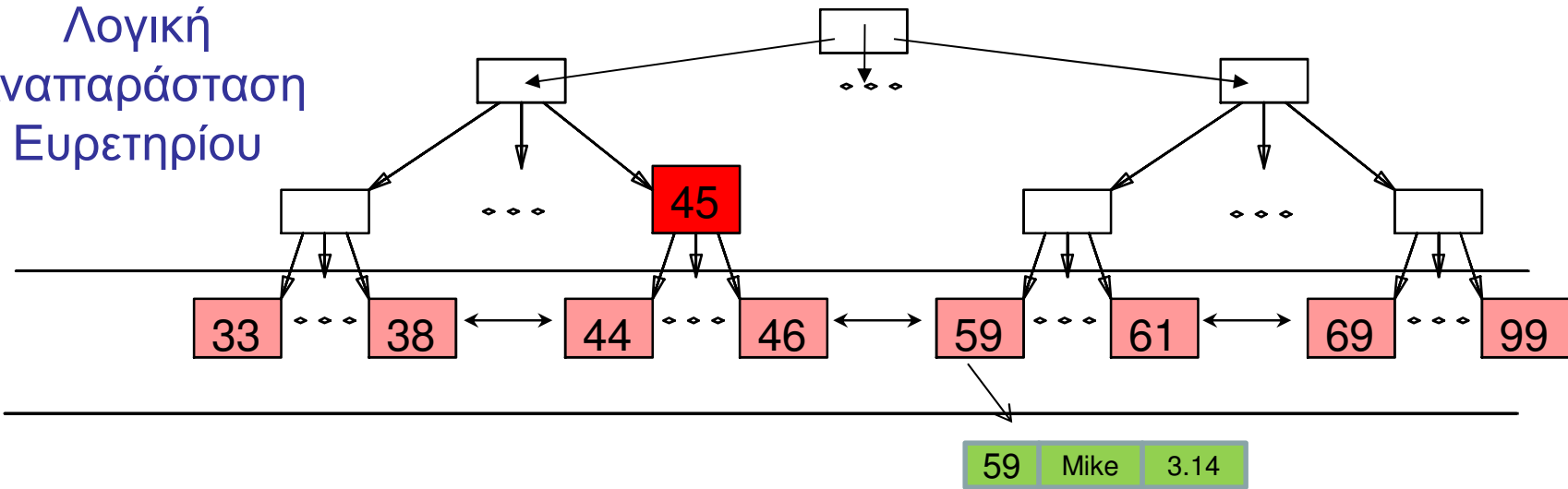
- Π.χ., Ευρετήρια Κατακερματισμού, Δενδρικά Ευρετήρια, Χωρικά Ευρετήρια και πολλά άλλα.

Άλλα Πλεονεκτήματα Χρήσης Βάσεων

(Παράδειγμα Δενδρικού Ευρετηρίου)

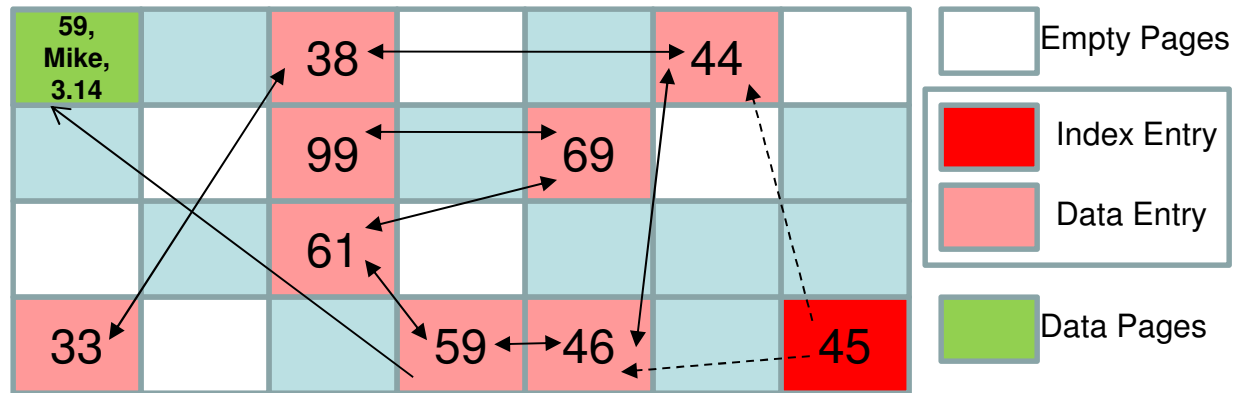


Λογική
Αναπαράσταση
Ευρετηρίου



Φυσική
Αναπαράσταση
Ευρετηρίου στη
Δευτερεύουσα
Μνήμη

Physical Layout (on Disk)



Άλλα Πλεονεκτήματα Χρήσης Βάσεων



9) Δυνατότητα Επιβολής Κανόνων Ακεραιότητας

- Π.χ., Επιβολή κανόνα: «Σε κανένα μάθημα δεν μπορούν να εγγραφούν πάνω από 99 φοιτητές»
- Παράδειγμα σε SQL-DDL*:

```
CREATE TABLE COURSE
```

```
( CourseID INT NOT NULL DEFAULT 1
```

```
Positions INT NOT NULL CHECK (Pos<100).....)
```

** Η δηλωτική αυτή γλώσσα θα μελετηθεί αργότερα στο μάθημα*

10) Δυνατότητα Εκτέλεσης Σκανδαλών (Triggers)

Π.χ., Όποτε αλλάξει κάτι στον λογαριασμό του χρήστη στείλε του SMS!

```
CREATE TRIGGER change AFTER INSERT OR DELETE OR  
UPDATE ON CustomerAccount
```

```
FOR EACH ROW EXECUTE PROCEDURE sendSMS ();
```

Κάποια Μέτα-Πλεονεκτήματα

Χρήσης Βάσεων



- **Δυνατότητα Επιβολής Κανόνων στην Αναπαράσταση Δεδομένων σε ένα οργανισμό:**
 - Π.χ., Σχεδιάζοντας και Υλοποιώντας τη βάση UNIVERSITY προτυποποιείται ότι το STUDENT.Name είναι μέχρι 30 χαρακτήρες.
- **Μειωμένος χρόνος ανάπτυξης εφαρμογών**
 - Υπολογίζεται ότι ο χρόνος ανάπτυξης εφαρμογών με βάσεις μειώνεται από **75-84%** του χρόνου που απαιτείται.
- **Ευελιξία**
 - Εύκολη μετάβαση σε άλλη κωδικοποίηση βάσης (με εργαλεία EXPORT).
 - Όχι τόσο απλό εάν είναι σε ειδική κωδικοποίηση

Πότε να μην χρησιμοποιούμε Βάσεις;



A) Υπάρχει συνήθως ψηλό κόστος αγοράς (9/9/09), π.χ.,

- Oracle Standard Edition (per CPU): **5,900\$**
- Oracle Enterprise Edition (per CPU): **47,500\$**
- IBM DB2 v9.7 Enterprise: **25,000\$**
- SQL Server 2008 Enterprise: **25,000\$**
- Τα πιο πάνω ΔΕΝ περιλαμβάνουν κόστος αγοράς υλικού (server), λειτουργικού συστήματος, training, κτλ.!
- Όταν ο προϋπολογισμός ενός project είναι περιορισμός τότε λαμβάνεται σημαντικά υπόψη μια τέτοια παράμετρος.
- Ακόμη και εάν χρησιμοποιηθεί DBMS **ανοικτού πηγαίου κώδικα** (π.χ., MySQL, PostgreSQL, κτλ.), οι οποίες είναι **δωρεάν**, τότε υπάρχουν και πάλι λόγοι για τους οποίους δεν θέλουμε να χρησιμοποιήσουμε τέτοιες βάσεις
 - Δες επόμενη διαφάνεια...

Πότε να μην χρησιμοποιούμε Βάσεις



- Β)** Εάν η βάση δεδομένων είναι **απλή**, καλά ορισμένη και δεν πρόκειται να αλλάξει εύκολα
- Π.χ., ένα απλό κατάλογο τηλεφώνων ενός γραφείου.
- Γ)** Εάν η πληροφορία είναι **πάρα πολύ μεγάλη**
- π.χ., το Google ΔΕΝ αποθηκεύει το ευρετήριο σε μια DBMS. **Γιατί;**
 - Αυτό βέβαια δεν είναι κανόνας. Λέγεται ότι το Facebook χρησιμοποιεί την MySQL:
 - Το Facebook έχει πάνω από 1.5 petabytes (10^{15}) σε φωτογραφίες (περίπου 10 δισεκατομμύρια φωτογραφίες!)
- Δ)** Εάν δεν υπάρχει το **προσωπικό («DBA»)** για παροχή ασφάλειας, ανάκαμψης, αντιγράφων, κτλ.
- Ε)** Εάν δεν υπάρχει πολλή **ταυτόχρονη πρόσβαση** στα δεδομένα, εάν δεν μπορεί εύκολα να αναπαρασταθεί κάποια πληροφορία σε μια βάση.

Ιστορική Εξέλιξη των Βάσεων



- **1960: Αρχικές Εφαρμογές «Βάσεων»:**
 - Ξεκίνησαν με την δημιουργία των Η/Υ τη δεκαετία του '60 και έχουν από τότε αλλάξει σε όλα τα επίπεδα.
 - Το πρώτο λογισμικό ΒΔ το **Integrated Data Store (IDS) / Codasyl** το οποίο προσωποποιήθηκε το 1971.
- **1970: Σχισιακές Βάσεις Δεδομένων**
 - Το **Σχισιακό Μοντέλο** (όπου η πληροφορία αναπαριστάται με με σχέσεις) **προτάθηκε** το **1969** από τον **E.F. Codd**.
 - Στη συνέχεια **ερευνήθηκε** και **δοκιμάστηκε** εκτενώς στο IBM Research και πολλά πανεπιστήμια.
 - Προϊόντα Σχισ. DBMS δημιουργούνται στις αρχές του 1980
 - Η **Oracle** δημιουργείται το **1977** από ένα project το οποίο επιχορηγείται από το **CIA**. Η Oracle είναι σήμερα ένας από τους κολοσσούς στο χώρο των βάσεων με γραφεία σε **126 χώρες** του κόσμου!

Ιστορική Εξέλιξη των Βάσεων



- **1980: Αντικειμενοστρεφείς Βάσεις:**
 - **Object-Oriented Database Management Systems (OODBMSs)** εισήχθηκαν περί τα τέλη του **1980s** σε μια ανάγκη να αναπαριστούν περίπλοκες ανάγκες διαχείρισης δεδομένων τα οποία προέκυπταν σε **CAD (Computer-Aided design) εφαρμογές**.
 - Σε αυτές, τα δεδομένα αναπαριστώνται από **Αντικείμενα (Objects)** κατά αντίστοιχο τρόπο με ΟΟ Προγραμματισμό.
 - Σήμερα **δεν** χρησιμοποιούνται **πολύ εκτενώς**
 - Ωστόσο πολλές έννοιες εισήχθηκαν σε Σχεσιακές DBMS δημιουργώντας την έννοια **Σχεσιό-Αντικειμενοστρεφών Βάσεων Δεδομένων [Object-Relational DBMSs (ORDBMSs)]**
 - Π.χ., η PostgreSQL είναι μια τέτοια βάση δεδομένων η οποία επιτρέπει τη δημιουργία οντοτήτων με κληρονομικότητα και πολλά άλλα.

Ιστορική Εξέλιξη των Βάσεων



- **1990: Δεδομένα στο Web και εφαρμογές Ηλεκτρονικού Εμπορίου:**
 - Το Web περιέχει περιεχόμενο στη Γλώσσα Μορφοποίησης HTML (Hypertext markup language) με συνδέσμους ανάμεσα σε ιστοσελίδες.
 - Αυτό έχει δώσει χώρο σε ένα **νέο σύνολο εφαρμογών στο χώρο των βάσεων δεδομένων**. Συγκεκριμένα:
 - Διαδικτυακές Βάσεις Δεδομένων.
 - Παραγωγή Περιεχομένου HTML από τα δομημένα δεδομένα ΒΔ με χρήση γλωσσών όπως PHP, Java, ASP, κτλ.
 - Επιτρέπει επίσης αλλαγές στα δεδομένα της βάσης
 - Εφαρμογές Business-to-Business (Ανταλλαγή Εγγράφων με χρήση της XML (eXtended Markup Language))
 - Παράδειγμα XML ακολουθεί στην επόμενη διαφάνεια.

Ιστορική Εξέλιξη των Βάσεων (HTML vs. XML με Παράδειγμα)



Μορφοποιώντας Δεδομένα με HTML

```
<html>
  <head><title>Bread Recipe Page (prepare in 5 minutes!)</title></head>
  <body>
    <p><font color=red><b>Flour:</b></font> 8 dL</p>
    <p><font color=red><b>Yeast:</b></font> 10 grams</p>
    <p><font color=red><b>Water:</b></font> 8dL (warm)</p>
    <p><font color=red><b>Salt:</b></font> 1 teaspoon</p>
    <p><font size=5><b>Cook for 3 hours!</b></font></p>
  </body>
</html>
```

Οντότητες

Γνωρίσματα

Δεδομένα

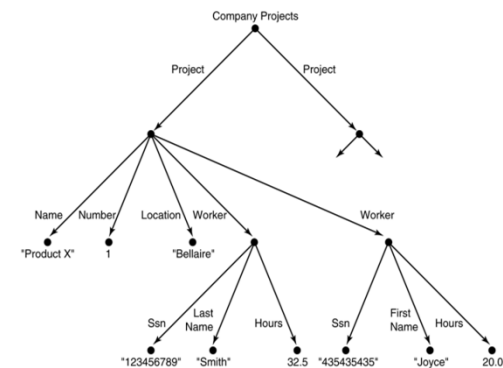
Περιγράφοντας Δεδομένα με XML

```
<recipe name="bread" prep_time="5 mins" cook_time="3 hours">
  <title>Basic bread</title>
  <ingredient amount="8" unit="dL">Flour</ingredient>
  <ingredient amount="10" unit="grams">Yeast</ingredient>
  <ingredient amount="4" unit="dL" state="warm">Water</ingredient>
  <ingredient amount="1" unit="teaspoon">Salt</ingredient>
</recipe>
```

Ιστορική Εξέλιξη των Βάσεων (HTML vs. XML με Παράδειγμα)



- Εφόσον η πληροφορία σε XML είναι **ιεραρχική**, αυτό δημιουργεί διάφορα προβλήματα αναπαράστασης της σε τυπικά μοντέλα **Βάσεων Δεδομένων**.
- Αυτό οδήγησε σε **Νέες Εξελίξεις**
 - Νέοι Τύποι Βάσεων Δεδομένων
 - MonetDB, existDB, κτλ.
 - Επέκταση Υφιστάμενων Βάσεων
 - Oracle XML DB, κτλ.
 - Νέες Γλώσσες (XPath, XQuery, κτλ.)

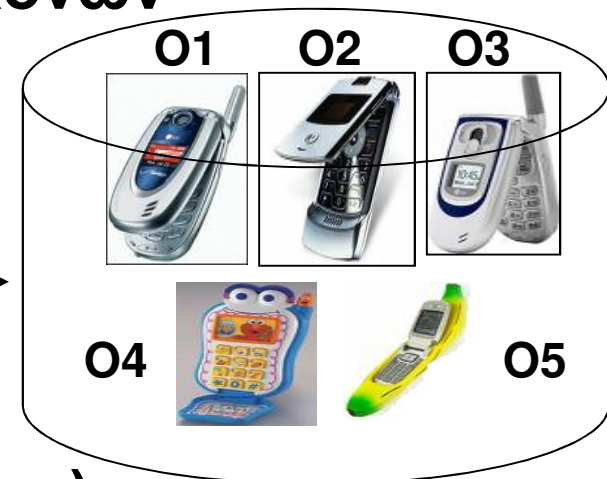


ΕΠΕΚΤΕΙΝΟΝΤΑΣ ΤΙΣ ΔΥΝΑΤΟΤΗΤΕΣ ΒΔ

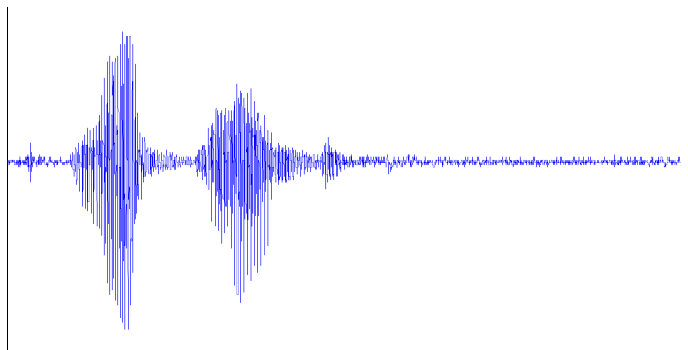
- Υπάρχουν προσθήκες δυνατοτήτων στα ακόλουθα πεδία:
 - Αποθήκευση και Διαχείριση Εικόνων**

$Q=(\text{χρώμα, texture, κτλ.})$

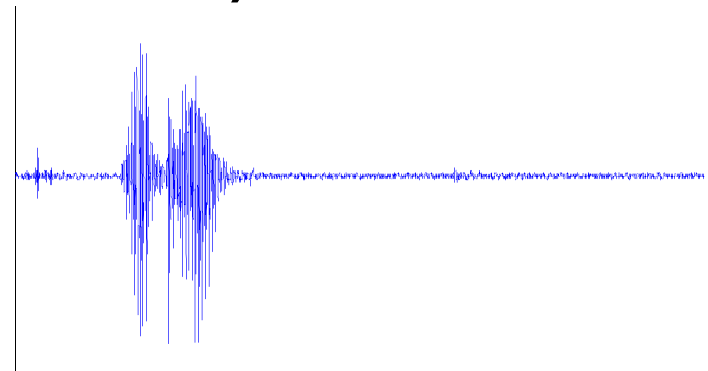
Q



- Διαχείριση Ήχου (αλλά και Βίντεο)**



----Mat-lab-----

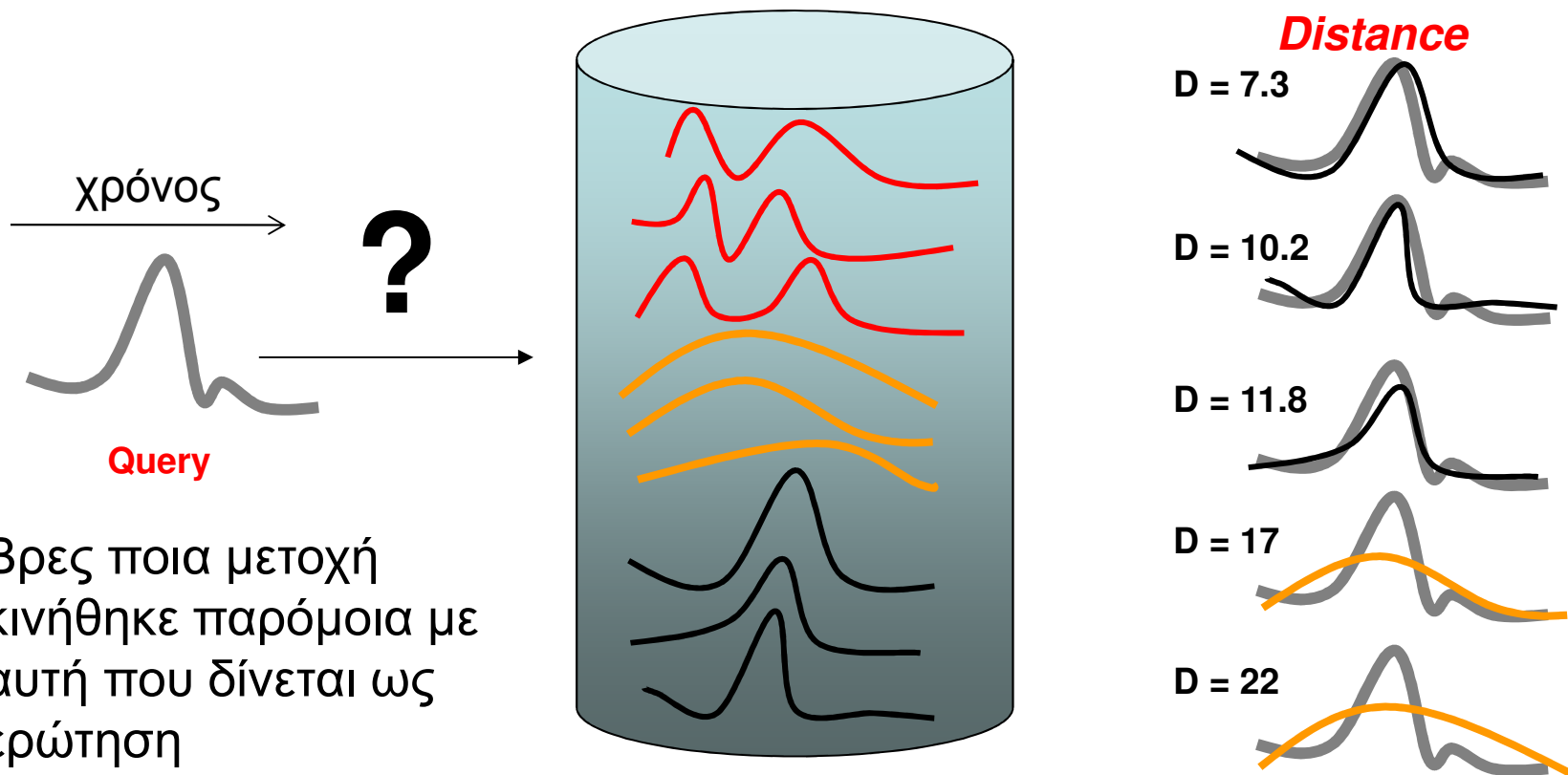


---Maat--Ilaabb-----

Επ. 242: Dataspecs Demetris Zinilour © (University of Cyprus)
Σύγκριση Spectrographs για εντοπισμό ομοίων λέξεων

ΕΠΕΚΤΕΙΝΟΝΤΑΣ ΤΙΣ ΔΥΝΑΤΟΤΗΤΕΣ ΒΔ

- Υπάρχουν προσθήκες δυνατοτήτων στα ακόλουθα πεδία:
 - Διαχείριση Χρονικών (Temporal) Δεδομένων



Βρες ποια μετοχή
κινήθηκε παρόμοια με
αυτή που δίνεται ως
ερώτηση

Μοντέλα δεδομένων



- Ένα **Μοντέλο Δεδομένων (Data Model)** είναι ένα **σύνολο εννοιών** που μπορούν να χρησιμοποιηθούν για την περιγραφή της **δομής ενός database**
- Τέτοιες έννοιες περιλαμβάνουν **έννοιες Δομής**
 - **Οντότητες (Elements)** και τους *τύπους δεδομένων τους*
 - *Ομάδες Οντοτήτων (Εγγραφές, Πίνακες, κτλ)*
 - **Συσχετίσεις (Relationships)** μεταξύ των Οντοτήτων.
 - **Περιορισμούς (Constraints)** τα οποία περιλαμβάνουν κάποιους κανόνες οι οποίοι πρέπει να τηρούνται πάντα για να είναι η ΒΔ σε μια συνεπή κατάσταση.
- Τέτοιες έννοιες περιλαμβάνουν **έννοιες Τελεστών**
 - Τελεστές **Ανάκτησης** και **Ενημέρωσης** Δεδομένων
 - Χωρίζεται σε **βασικούς τελεστές** (insert, delete, update) και τελεστών χρηστών (π.χ., compute_student_gpa, update_inventory)

Κατηγορίες Μοντέλων



- **Υψηλού Επιπέδου Μοντέλα, εννοιολογικό μοντέλο (high-level or conceptual)**

- Παρέχει έννοιες κοντά στον τρόπο που **πολλοί χρήστες καταλαβαίνουν** τα διάφορα δεδομένα
- Π.χ., **Entity-Relationship Model**

- **Χαμηλού Επιπέδου Μοντέλα (low-level or physical)**

- Παρέχει έννοιες που περιγράφουν τις λεπτομέρειες του πως τα δεδομένα είναι αποθηκευμένα στη **δευτερεύουσα μνήμη**
- Π.χ., Specific Storage Model

- **Ενδιάμεσου Επίπεδου Μοντέλα (Αναπαραστατικό) (Representational or implementational)**

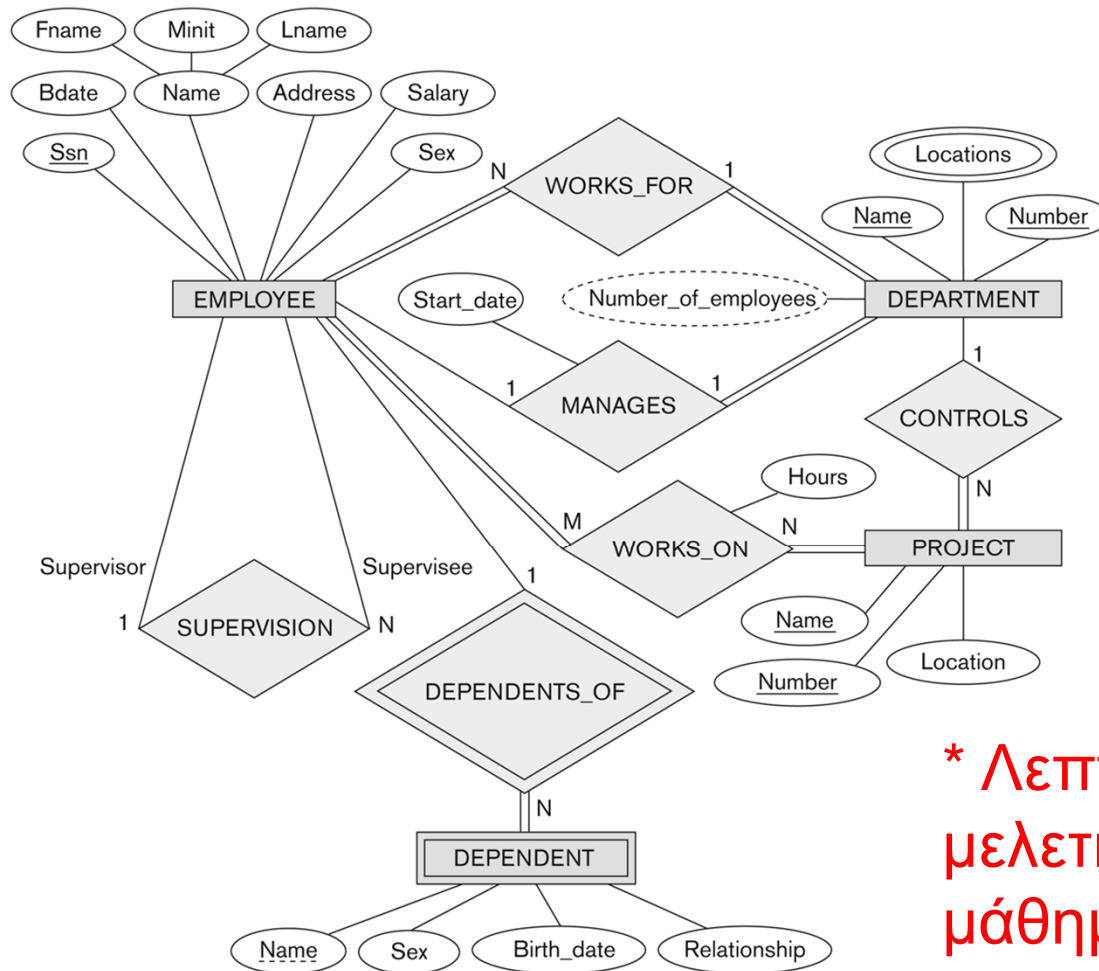
- Παρέχει έννοιες που είναι **μεν κατανοητές** από τους **χρήστες** αλλά **όχι πολύ απομακρυσμένες** από το τρόπο αποθήκευσης
- Π.χ., **Relational Model** and DB Schemas

Κατηγορίες Μοντέλων

(Παράδειγμα **Εννοιολογικού Μοντέλου**)



Υψηλού Επίπεδου



* Λεπτομέρειες θα μελετηθούν αργότερα στο μάθημα.

Figure 3.2

An ER schema diagram for the COMPANY database. The diagrammatic notation is introduced gradually throughout this chapter.

Κατηγορίες Μοντέλων

(Παράδειγμα Αναπαραστατικού Μοντέλου)



Ενδιάμεσου Επίπεδου

Figure 2.1

Schema diagram for the database in Figure 1.2.

STUDENT

Name	Student_number	Class	Major
------	----------------	-------	-------

COURSE

Course_name	Course_number	Credit_hours	Department
-------------	---------------	--------------	------------

PREREQUISITE

Course_number	Prerequisite_number
---------------	---------------------

* Λεπτομέρειες θα μελετηθούν αργότερα στο μάθημα.

SECTION

Section_identifier	Course_number	Semester	Year	Instructor
--------------------	---------------	----------	------	------------

GRADE_REPORT

Student_number	Section_identifier	Grade
----------------	--------------------	-------

Σχήματα και Στιγμιότυπα



- **Σχήμα Βάσης (Database Schema)**
 - Η περιγραφή μιας βάσης.
 - Περιλαμβάνει περιγραφές της δομής, τύπων δεδομένων, και περιορισμούς.

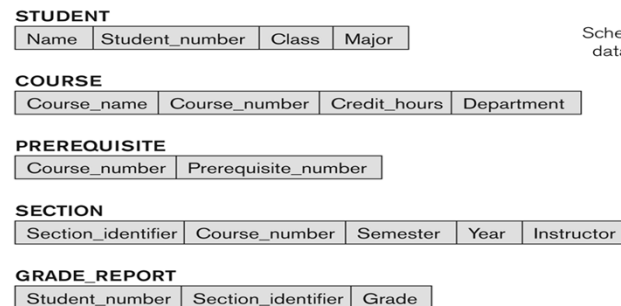


Figure 2.1
Schema diagram for the database in Figure 1.2.

- **Διάγραμμα Σχήματος:**
 - Ένας γραφικός τρόπος αναπαράστασης των πληροφοριών ενός σχήματος.
 - Στο πιο πάνω διάγραμμα δεν φαίνονται οι τύποι δεδομένων και οι περιορισμοί.

Σχήματα και Στιγμιότυπα



- **Κατάσταση Βάσης (Database State):**
 - Η **πραγματική πληροφορία** που αποθηκεύεται σε μια βάση μια **δεδομένη στιγμή**
 - Αυτό περιλαμβάνει τη συλλογή **όλων** των δεδομένων της βάσης δεδομένων.
 - Ονομάζεται επίσης **Στιγμιότυπο ΒΔ (DB instance, occurrence or snapshot)**.
 - Με την ίδια λογική, ο όρος **στιγμιότυπο** μπορεί να εφαρμοστεί πάνω σε επί μέρους συστατικά μιας βάσης (π.χ., στιγμιότυπο **εγγραφής**, στιγμιότυπο **πίνακα**, στιγμιότυπο **οντότητας**, κτλ.)
 - **Παρατήρηση:**
 - Το **Σχήμα** μιας βάσης **ΔΕΝ** αλλάζει συχνά
 - Η **Κατάσταση** μια βάσης **αλλάζει συχνά**.

Κατηγορίες Μοντέλων

(Παράδειγμα Αναπαραστατικού Μοντέλου)



STUDENT

Name	Student_number	Class	Major
------	----------------	-------	-------

COURSE

Course_name	Course_number	Credit_hours	Department
-------------	---------------	--------------	------------

PREREQUISITE

Course_number	Prerequisite_number
---------------	---------------------

SECTION

Section_identifier	Course_number	Semester	Year	Instructor
--------------------	---------------	----------	------	------------

GRADE_REPORT

Student_number	Section_identifier	Grade
----------------	--------------------	-------

Διάγραμμα
Σχήματος

COURSE

Course_name	Course_number	Credit_hours	Department
Intro to Computer Science	CS1310	4	CS
Data Structures	CS3320	4	CS
Discrete Mathematics	MATH2410	3	MATH
Database	CS3380	3	CS

SECTION

Section_identifier	Course_number	Semester	Year	Instructor
85	MATH2410	Fall	04	King
92	CS1310	Fall	04	Anderson
102	CS3320	Spring	05	Knuth
112	MATH2410	Fall	05	Chang
119	CS1310	Fall	05	Anderson
135	CS3380	Fall	05	Stone

GRADE_REPORT

Student_number	Section_identifier	Grade
17	112	B
17	119	C
8	85	A
8	92	A
8	102	B
8	135	A

PREREQUISITE

Course_number	Prerequisite_number
CS3380	CS3320
CS3380	MATH2410
CS3320	CS1310

Figure 1.2
A database that stores student and course information.

Κατάσταση Σχήματος τη
Στιγμή X