



ΕΛΛΗΝΙΚΗ ΔΗΜΟΚΡΑΤΙΑ  
ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΡΗΤΗΣ

# Συστήματα Διαχείρισης Βάσεων Δεδομένων

## Εισαγωγή

Δημήτρης Πλεξουσάκης  
Τμήμα Επιστήμης Υπολογιστών

HY 460:

Υλοποίηση Συστημάτων  
Διαχείρισης Βάσεων Δεδομένων

<http://www.csd.uoc.gr/~hy460>

Δ. Πλεξουσάκης [dp@ics.forth.gr](mailto:dp@ics.forth.gr)

Χ. Κονδυλάκης [kondylak@ics.forth.gr](mailto:kondylak@ics.forth.gr)

Τμήμα Επιστήμης Υπολογιστών, Πανεπιστήμιο Κρήτης  
Ηράκλειο Κρήτης

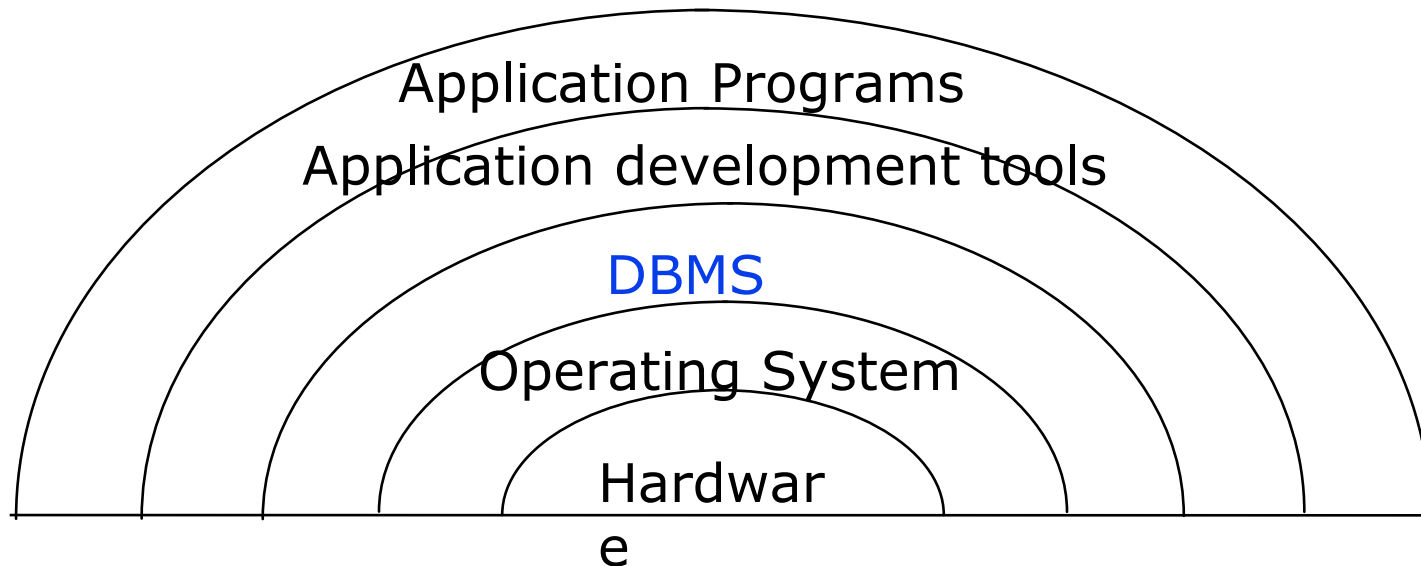
# Υποστήριξη Μαθήματος

## λ Βιβλία

- ① Συστήματα Βάσεων Δεδομένων Τόμος I & II H. Garcia-Molina, J. D. Ullman and J. D. Widom Πανεπιστημιακές Εκδόσεις Κρήτης, 2011 (Μετάφραση Γ. Γεωργακόπουλος, Θ. Δρίτσας)
- ② Συστήματα Διαχείρισης Βάσεων Δεδομένων 2η Έκδοση, Τόμος A & B, R.Ramakrishnan and J. Gehrke, Εκδόσεις Τζιόλα 2001 (Μετάφραση Δ. Δέρβος, Α.Ευαγγελίδης)
- ③ Database Tuning: Principles Experiments and Troubleshooting Techniques D. Shasha and P. Bonnet, Morgan Kaufmann Publishers 2002

# Τι είναι ένα Σύστημα Διαχείρισης Βάσεων Δεδομένων (ΣΔΒΔ)?

- λ Βάση Δεδομένων (ΒΔ): «Λογικά» Συνεκτική Συλλογή Πληροφοριών
  - υ Αποθηκευμένη: Μόνιμα Δεδομένα με κάποια εγγενή σημασία
  - υ Προσπελάσιμη: Ανακτήσιμα ή εξαγόμενα δεδομένα για κάποιο σκοπό
- λ Σύστημα Διαχείρισης Βάσεων Δεδομένων (ΣΔΒΔ): Λογισμικό γενικού σκοπού που υποστηρίζει την διαδικασία ορισμού, κατασκευής και χειρισμού βάσεων δεδομένων για συγκεκριμένες εφαρμογές



# Ποιος είναι ο Όγκος των Διαθέσιμων Δεδομένων ?

λ TeraBytes και Gigabytes είναι μεγάλοι όγκοι!

υ Mega – μεγάλη κατοικία στην Αθήνα

υ Giga – ένα πολύ πλούσιο πρόσωπο (ή το ανθρώπινο γονιδίωμα)

υ Tera – ~ το δημόσιο χρέος της Αμερικής (ή μια στοίβα από βιβλία μήκους 150 μιλίων)

υ Peta – περισσότερα και από το διαθέσιμο χρήμα σε όλο τον κόσμο

λ Η ιστορική τάση δείχνει ότι σε 20 χρόνια:

υ Όλη η διαθέσιμη πληροφορία θα έχει καταχωρηθεί και ευρετηριαστεί

(see <http://www.lesk.com/mlesk/ksg97/ksg.html>,

<http://www.sims.berkeley.edu/research/projects/how-much-info/> )

στην μνήμη του Jim Gray

Everything!  
Recorded

All Books  
MultiMedia

All  
books  
(words)

Movie

A Photo

A Book

Yotta( $10^{24}$ )

Zetta( $10^{21}$ )

Exa ( $10^{18}$ )

Peta( $10^{15}$ )

Tera( $10^{12}$ )

Giga( $10^9$ )

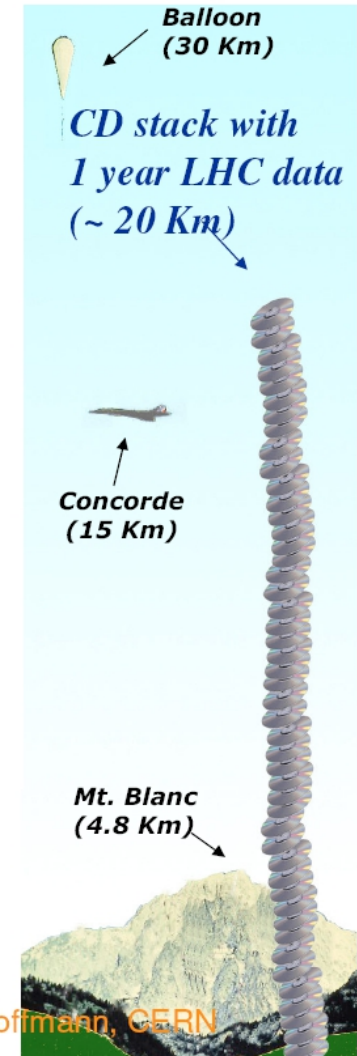
Mega( $10^6$ )

Kilo( $10^3$ )<sub>4</sub>

# Η Χιονοστιβάδα των Δεδομένων

## Παραδείγματα:

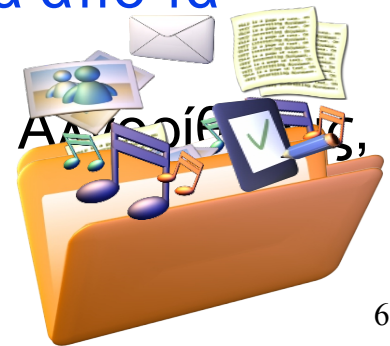
- λ **Τηλεπικοινωνίες:** AT&T 20TB βάση τηλεφωνικών κλήσεων, ασύρματων δικτύων
- λ **Εμπορικές:** WalMart 70TB βάση με καταναλωτικά πρότυπα αγοραστών
- λ **WWW:** ευρετήριο 200M σελίδων και 2000M υπερσυνδέσμων στον Παγκόσμιο Ιστό, η Akamai αποθηκεύει  $7 \cdot 10^9$  επισκέψεις (clicks) ημερησίως, το Facebook διαθέτει  $1.8 \cdot 10^9$  φωτογραφίες και έχει  $10^5$  καινούργιους χρήστες ημερησίως
- λ **Αστρονομία:** οι δορυφόροι της NASA παράγουν 1.2TB εικόνες ημερησίως
- λ **Φυσική:** ο Μεγάλος Επιταχυντή Αδρονίων LHC του CERN παράγει 12-14 PB δεδομένων ημερησίως



Courtesy: Dr Hans F Hoffmann, CERN

# Γιατί Μελετούμε τα Συστήματα Βάσεων Δεδομένων?

- λ Η χρήση των ΣΔΒΔ έχει γενικευτεί τα τελευταία χρόνια
  - υ Εταιρικές εφαρμογές: διαχείριση αποθεμάτων, προσωπικού, πελατών, προμηθευτών, κλπ.
  - υ Επιστημονικές εφαρμογές: ψηφιακές βιβλιοθήκες, βάσεις γονιδίων, διαστημικών δεδομένων, φυσικών αισθητήρων, κλπ.
- λ Αλλαγή εστίασης από τους υπολογισμούς στην πληροφορία
  - υ Διαρκής ανάγκη στο πλαίσιο των εταιρικών εφαρμογών
  - υ Το Διαδίκτυο ανέδειξε αυτήν την ανάγκη επίσης στο πλαίσιο των προσωπικών εφαρμογών
  - υ Όλο και μεγαλύτερη ανάγκη σήμερα στις επιστημονικές εφαρμογές
- λ Τα ΣΔΒΔ ενσωματώνουν από μία πρακτική σκοπιά αρκετά από τα επιτεύγματα της Επιστήμης των Υπολογιστών
  - υ Λειτουργικά Συστήματα, Τυπικές Γλώσσες, Θεωρία και Αλγόριθμοι, Τεχνητή Νοημοσύνη και Λογική
  - υ Εστίαση σε πραγματικές εφαρμογές και ανάγκες



# Ποιο είναι το Ενδιαφέρον Γνωστικό Αντικείμενο?

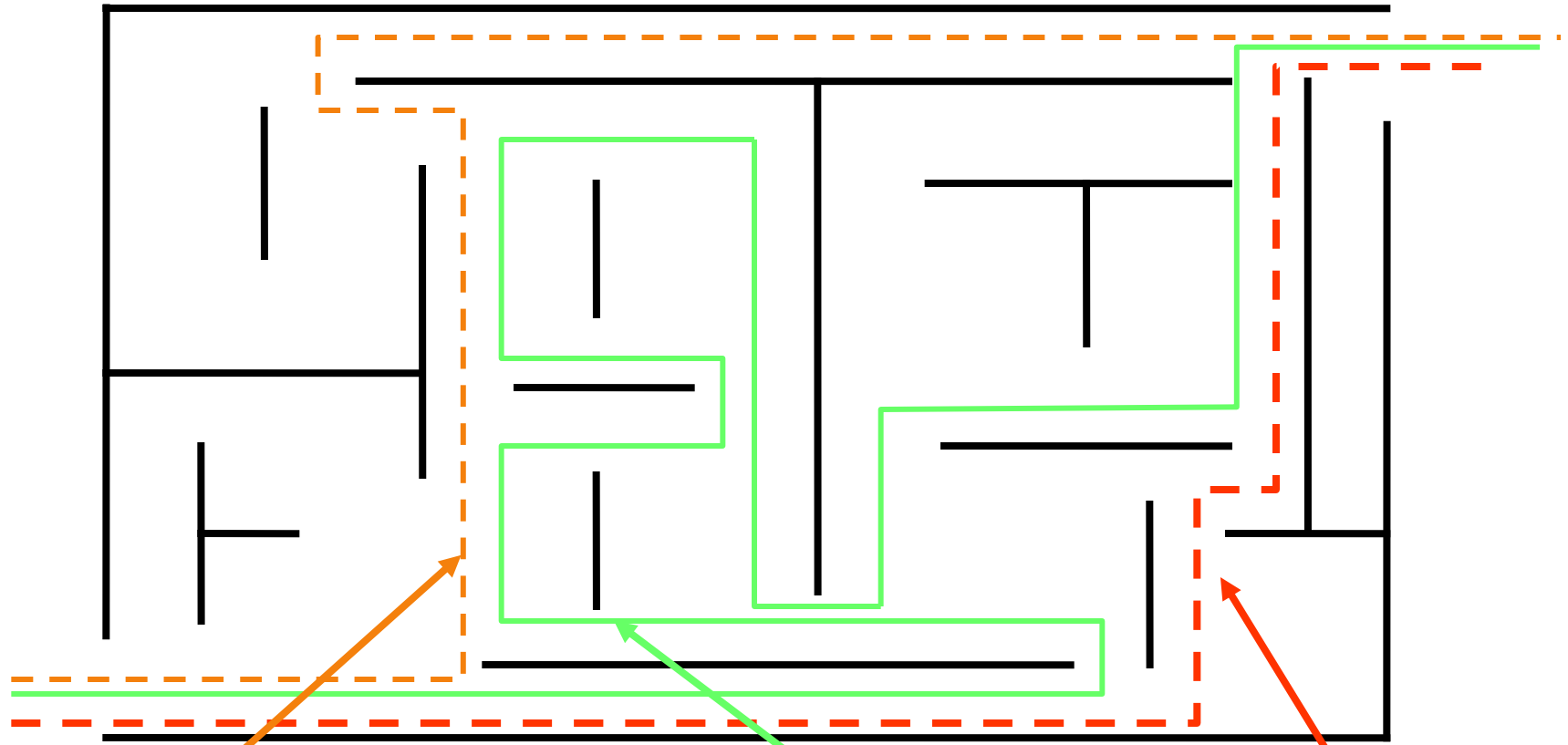
- λ Αναπαράσταση Πληροφοριών Ανεξαρτήτως των Εφαρμογών
  - υ Μοντελοποίηση δεδομένων
- λ Δηλωτικές Γλώσσες Επερώτησης Δεδομένων
  - υ Σύνθετες επερωτήσεις με πρακτική σημασιολογία\*
- λ Ταυτόχρονη Πρόσβαση στον Χειρισμό Δεδομένων
  - υ Έλεγχος ταυτόχρονων προσπελάσεων
  - υ Εξασφαλίζοντας την σημασιολογία των δοσοληψιών
- λ Αξιόπιστος Αποθηκευτικός Χώρος
  - υ Διατήρηση της σημασιολογίας των δεδομένων ακόμα και όταν δεν έχουμε πρόσβαση
- λ Αποδοτική Πρόσβαση σε Δεδομένα
  - υ Μεγάλου όγκου



\* σημασιολογία: η σημασία ενός συμβόλου η ενός συνόλου συμβόλων



# Τι Σημαίνει Βελτιστοποίηση Επερωτήσεων?



**Good**

(Normally, but still be improved)

**Very Bad**

(most of the times for complex SQL)

**Very Good**

(sometimes for simple SQL)

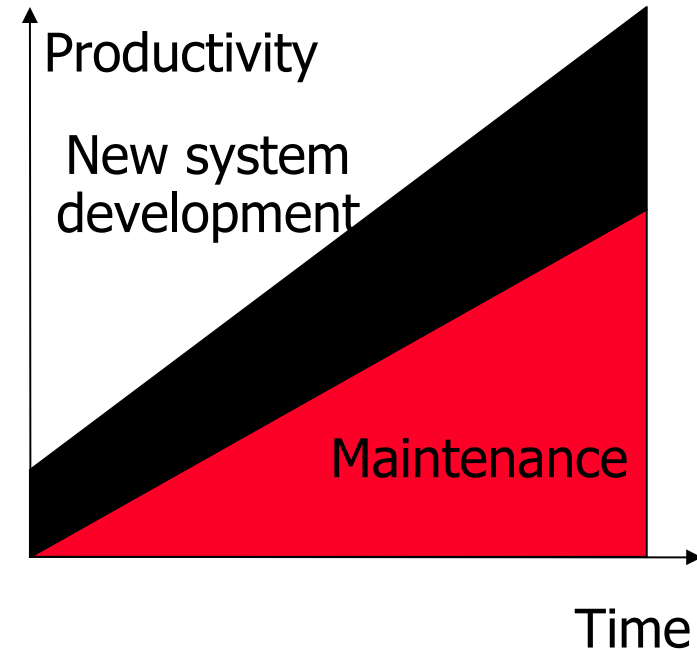
# Συγκρίνοντας τα ΣΔΒΔ με Άλλου Τύπου Λογισμικό

## λ Αποτελεσματική διαχείριση της μονιμότητας των δεδομένων

- υ Ανάνηψη από αποτυχίες (failures)
- υ Ταυτόχρονη πρόσβαση
- υ Έλεγχος ασφάλειας και ακεραιότητας
- υ Ανεξαρτησία των δεδομένων

## λ Αποτελεσματική πρόσβαση σε μεγάλες ποσότητες δεδομένων

- υ Διαχείριση αποταμιευτών μνήμης (buffer)
- υ Ευρετήρια/Συστάδες (clustering) δεδομένων
- υ Βελτιστοποίηση επερωτήσεων
- υ Κατανεμημένα δεδομένα/εφαρμογές



# Γιατί όχι ΣΔΒΔ;

- λ Η πολυπλοκότητα και το κόστος εγκατάστασης και συντήρησης ενός ΣΔΒΔ πρέπει να δικαιολογείται από τις ανάγκες των εφαρμογών
- λ Επένδυση σε λογισμικό και υλικό, καθώς και για εκπαίδευση
- λ Η γενικότητα που παρέχει προκαλεί χρονική επιβάρυνση (overhead)
- λ Δε χρειάζονται όσα προσφέρει
- λ Λογισμικό γενικής χρήσης το οποίο δεν είναι κατάλληλο σε εξειδικευμένες εφαρμογές (π.χ. αναζήτηση πλήρους κειμένου!)

# Σκοπός Μαθήματος (I)

- λ Αυτό το μάθημα παρέχει μια εισαγωγή στις **αρχές** που διέπουν την **υλοποίηση Συστημάτων Διαχείρισης Βάσεων Δεδομένων (ΣΔΒΔ)**
- λ Ο σκοπός του είναι
  - υ να διδάξει τις **βασικές έννοιες** και **αλγόριθμους** που χρησιμοποιούνται σήμερα ευρέως για την υλοποίηση των **κύριων συνιστωσών** ενός **Συστήματος Διαχείρισης Βάσεων Δεδομένων**
  - υ Διαχείριση
    - Αποθήκευσης,
    - Επερωτήσεων
    - Δοσοληψιών

## Σκοπός Μαθήματος (II)

- λ Φοιτητές που ολοκληρώνουν το μάθημα θα είναι σε θέση:
- υ να κατανοούν καλύτερα τις **συνέπειες** που έχουν **στις επιδόσεις του συστήματος** οι διαφορετικές **επιλογές εσωτερικής υλοποίησης**
    - (διαχειριστής σελίδων, αποταμιευτών, και αρχείων, ευρετήρια βασισμένα σε B+-δέντρα και κατακερματισμό, αλγόριθμοι εξωτερικής ταξινόμησης και σύνδεσης, κλπ.)
  - υ να ακολουθούν μια δομημένη προσέγγιση στην **επίλυση προβλημάτων ρύθμισης** (database tuning) ενός **συστήματος** μέσω κατάλληλων παραμέτρων (πλατφορμών Λ/Υ, δομών δεδομένων, πλάνων εκτέλεσης επερωτήσεων) που το καθιστούν **πιο αποδοτικό** στις απαιτήσεις συγκεκριμένων εφαρμογών
  - υ να αξιολογούν τα πλεονεκτήματα και τα μειονεκτήματα εμπορικών (ORACLE, DB2, Sybase, SQLServer) και μη (MySQL, Postgres) **Συστημάτων Διαχείρισης Βάσεων Δεδομένων**

# Επισκόπηση Μαθήματος

## λ Αποθήκευση Δεδομένων και Ευρετήρια

- υ Δίσκοι και Αρχεία
- υ Δενδρικά Ευρετήρια και Ευρετήρια Κατακερματισμού
- υ Εξωτερική Ταξινόμηση

## λ Επεξεργασία και Βελτιστοποίηση Επερωτήσεων

- υ Υλοποίηση Σχεσιακών Αλγεβρικών Τελεστών
- υ Λογική και Φυσική Βελτιστοποίηση Σχεσιακών Επερωτήσεων
- υ Αρχιτεκτονική ενός Σχεσιακού Βελτιστοποιητή Επερωτήσεων

## λ Διαχείριση Δοσοληψιών

- υ Ανάνηψη από αποτυχίες
- υ Έλεγχος συγχρονισμού
  - Σενάρια εκτέλεσης και σειριακοποιησιμότητα
  - Κλείδωμα
  - Δεντρικά Πρωτόκολλα

# Η εξέλιξη των Βάσεων Δεδομένων (πριν το 1960)

## Γεγονότα

- 1945 Οι μαγνητικές ταινίες εμφανίζονται (είναι το πρώτο μέσο αποθήκευσης το οποίο επιτρέπει αναζήτηση)
- 1957 Κατασκευάζεται ο πρώτος εμπορικός υπολογιστής
- 1959 McGee προτείνει την έννοια της γενικευμένης πρόσβασης σε ηλεκτρονικά αποθηκευμένα δεδομένα
- 1959 IBM εισάγει το σύστημα Ramac

## Συνέπειες

- Αντικατάσταση των διατρητικών καρτών (punch cards) και των ταινιών χαρτιού
- Το διάβασμα δεδομένων με μη σειριακό τρόπο και η πρόσβαση στα αρχεία γίνεται πιο ευέλικτη

# Η εξέλιξη των Βάσεων Δεδομένων (Δεκαετία του 60)

## Γεγονότα

- **1961** Το πρώτο γενικευμένο σύστημα DBMS-GE'S για την αποθήκευση δεδομένων το οποίο σχεδιάστηκε από τον Bachman (ευρεία αποδοχή μέχρι το 1964). Τα διαγράμματα δεδομένων έγιναν πολύ δημοφιλή
- **1965-1970**
  - Γενίκευση των συστημάτων διαχείρισης αρχείων. Αναπτύχθηκαν πολλά συστήματα από πολλές εταιρίες λογισμικού
  - Information Management System (IMS) από την IBM
  - IMS DB/DC. Το πρώτο σύστημα για μεγάλες ΒΔ
  - SABRE, αναπτύχθηκε από την IBM και American Airlines

## Συνέπειες

- Τυποποιήθηκε το Δικτυακό μοντέλο δεδομένων το οποίο αναπτύχθηκε από το Conference on Data Systems Languages Database Task Group (CODASYL DBTG)
- Παρέχεται η δυνατότητα για εννοιολογική και από άποψη χρήστη οργάνωση των δεδομένων
- Τυποποιήθηκε το Ιεραρχικό μοντέλο δεδομένων
- Υποστηρίζονται όψεις δεδομένων
- Επιτρέπεται η πολύ χρηστική πρόσβαση των δεδομένων μέσω δικτύου



# Η εξέλιξη των Βάσεων Δεδομένων (Δεκαετία του 70)

## Γεγονότα

- Η τεχνολογία των Βάσεων Δεδομένων αναπτύσσεται ραγδαία
  - 1970** Το σχεσιακό μοντέλο αναπτύσσεται από τον Ted Codd, στην IBM
  - 1971** CODASYL Database Task Group Report
  - 1975** ACM Special Interest Group on Management of Data organized first (SIGMOD) international conference
  - 1975** Ιδρύεται το Very Large Data Base Foundation και οργανώνει το πρώτο διεθνές συνέδριο VLDB
  - 1976** Εισαγωγή του μοντέλου Οντοτήτων-Συσχετίσεων (ER) από τον Chen
- Αναπτύσσονται οι πρώτες δηλωτικές γλώσσες επερωτήσεων: SQUARE, SEQUEL (SQL), QBE, QUEL

## Συνέπειες

- Εμπορικά συστήματα ακολουθούν το CODASYL DBTG Proposal, αλλά κανένα από αυτά δεν είναι πλήρως υλοποιημένο. IDMS system by B. F. Goodrich, Honeywell's IDS II, UNIVACs DMS 1100, Burroughs's DMS-II, CDCs DMS-170, Phillips's PHOLAS, and Digital's DBMS-11. Several integrated DB/DC systems: Cincom's TOTAL plus ENVIRON/I
- ΣΔΒΔ αναπτύσσονται ως ακαδημαϊκά εργαλεία έρευνας
- Αρχίζει έρευνα στην θεωρία των Βάσεων Δεδομένων
- Δημιουργείται ένα forum για την διάδοση της έρευνας στις ΒΔ

# Η εξέλιξη των Βάσεων Δεδομένων (Δεκαετία του 80)

- Αναπτύσσονται ΣΔΒΔ για προσωπικούς υπολογιστές (DBASE, PARADOX, etc)

**1983** ANSI/SPARC survey revealed >100 relational systems had been implemented by the beginning of the '80s

**1984** Δημοσιεύονται τα βασικά πρότυπα για την SQL. Business world influenced by "Fourth Generation Languages." Proposal for Network Definition Language (NDL) made by ANSI

**-Κύριες τάσεις του 80:** Expert Database Systems, Object-oriented DBMSS, client-server architecture for distributed databases

- Επιτρέπεται σε χρήστες PC να επεξεργάζονται δεδομένα
- Επιτρέπεται η ταυτόχρονη πρόσβαση στα δεδομένα μέσω διαφορετικών προγραμμάτων
- Εμφανίζονται πολλά εμπορικά σχεσιακά ΣΔΒΔ (DB2, ORACLE, SYBASE, INFORMIX, etc)
- Δημιουργούνται πολλές και πλήρεις εφαρμογές οι οποίες δεν αφορούν προγραμματιστές αλλά χρήστες
- Επιτρέπονται πολλές νέες εφαρμογές με χρήση δικτύου και κατανομής των δεδομένων

# Η εξέλιξη των Βάσεων Δεδομένων (Δεκαετία του 90)

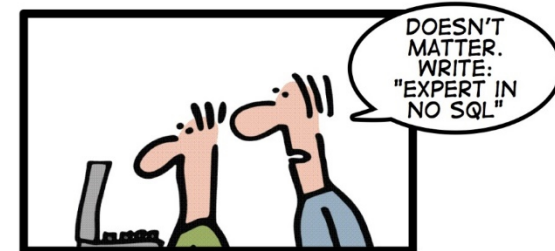
- λ Απαίτηση για επέκταση των ΣΔΒΔ για να υλοποιηθούν νέες εφαρμογές
- λ Ανάγκη για εμπορικά αντικειμενοστραφή ΣΔΒΔ
- λ Απαίτηση για ανάπτυξη εφαρμογών ολοκλήρωσης δεδομένων από πολλές διαφορετικές μεταξύ τους πηγές
- λ Απαίτηση για υποστήριξη παράλληλων εφαρμογών και επεξεργαστών (MPPs)
- λ Προστέθηκαν στα ΣΔΒΔ χαρακτηριστικά για την υποστήριξη χρονικών, χωρικών και πολυμεσικών δεδομένων καθώς και της δυνατότητας εξαγωγής συμπερασμάτων από τα αποθηκευμένα δεδομένα
- λ Εμφανίστηκαν νέα πρότυπα για την μοντελοποίηση δεδομένων και την εκτέλεση ερωτήσεων (SQL2, PDES, STEP)
- λ Επέκταση των ΣΔΒΔ με δυνατότητες διαχείρισης ετερογενών και διαδικτυακών δεδομένων

# Σήμερα

- λ Μεγάλος όγκος δεδομένων
  - υ (BIG DATA)
- λ Αλλαγές σε υλικό
  - υ Επεξεργαστές με πολλούς πύρηνες
  - υ Ιεραρχία αποθήκευσης (SSDs κτλ.)
- λ Cloud computing, software as service

- λ Κίνηση NoSQL

## HOW TO WRITE A CV

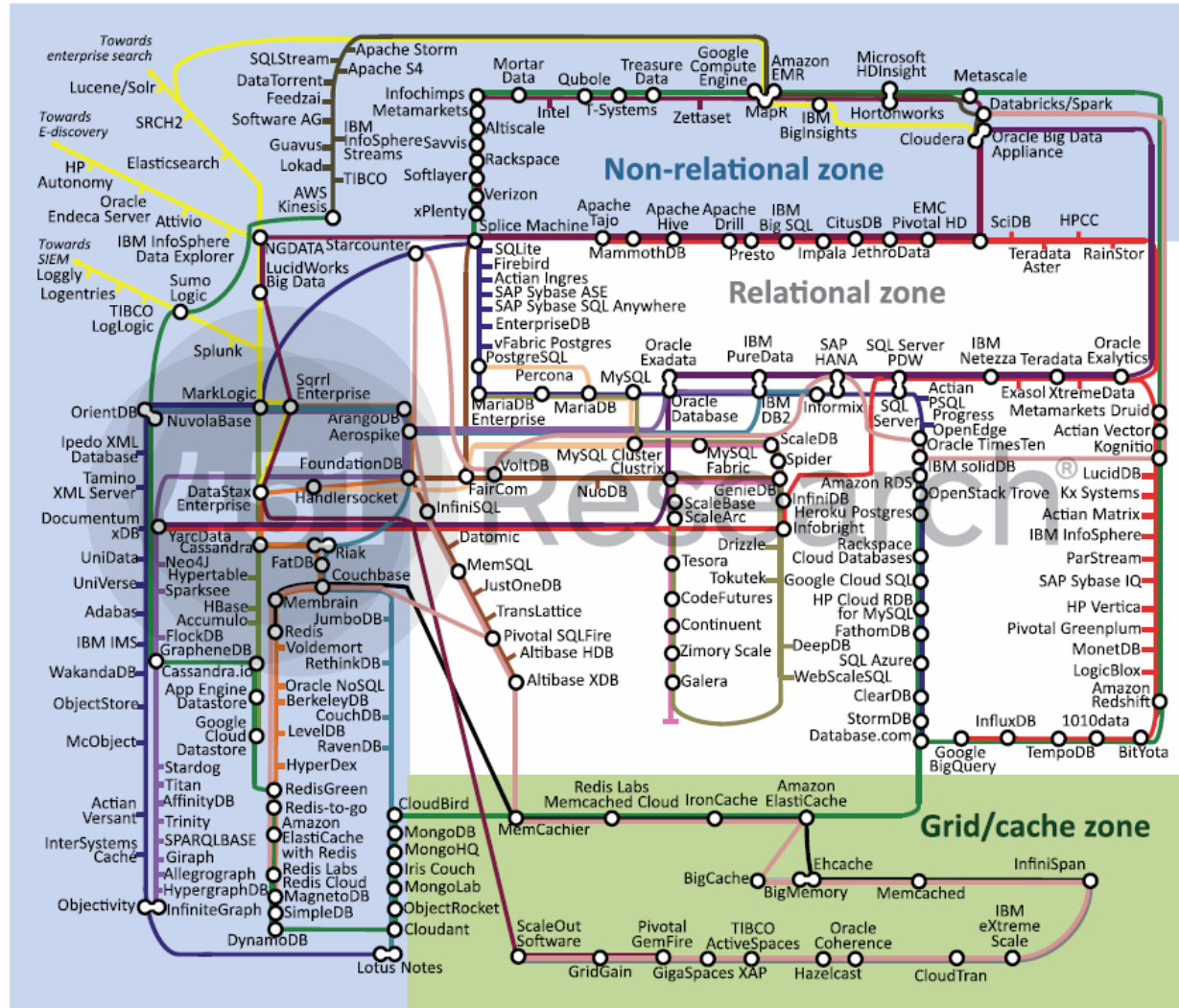


Leverage the NoSQL boom



# Data Platforms Landscape Map

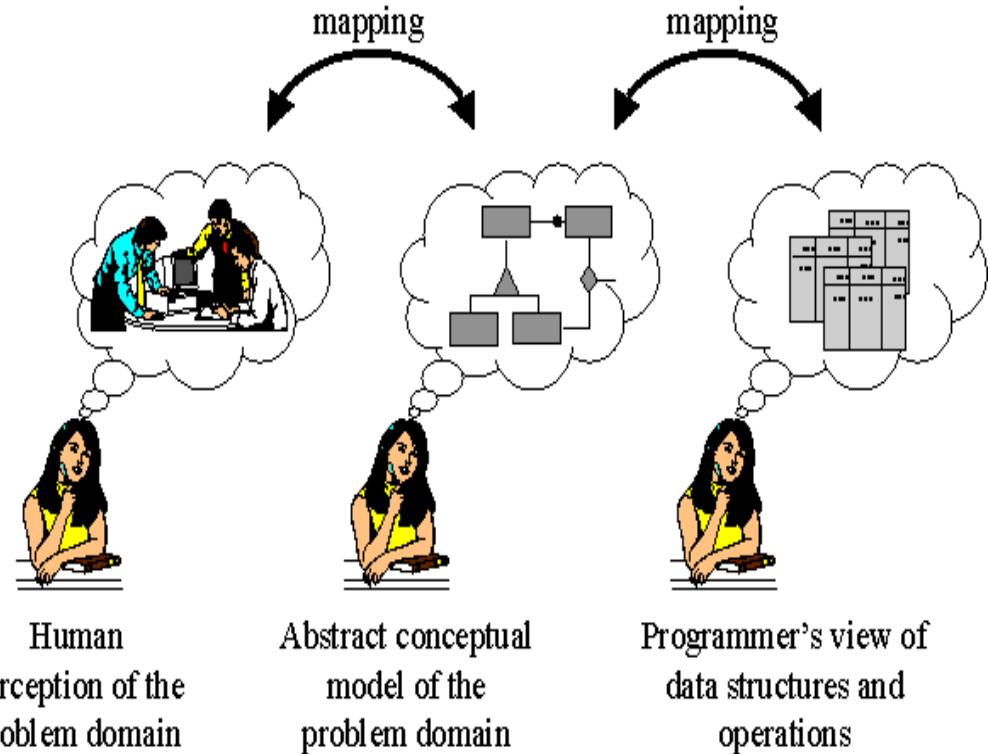
FEBRUARY 2014



- Key:**
- General purpose
  - Specialist analytic
  - -as-a-Service
  - - - NoSQL extension
  - Big Tables
  - Graph
  - Document
  - Key value stores
  - Key value direct access
  - Hadoop
  - - - NewSQL extension
  - Advanced clustering/sharding
  - New SQL databases
  - Data caching
  - Data grid
  - Search
  - Appliances
  - Off-heap memory
  - In-memory
  - Stream processing

# Περιγραφή των Δεδομένων: Τα μοντέλα ενός ΣΔΒΔ

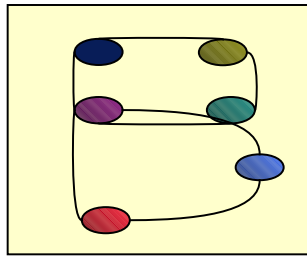
- λ Ένα **μοντέλο δεδομένων** είναι μια συλλογή από έννοιες που χρησιμοποιούνται για την περιγραφή δεδομένων του πραγματικού κόσμου
- λ Ένα **σχήμα** είναι μια περιγραφή μιας συγκεκριμένης συλλογής δεδομένων. Για αυτήν την περιγραφή έχει χρησιμοποιηθεί ένα συγκεκριμένο μοντέλο δεδομένων
- λ Το **σχεσιακό μοντέλο δεδομένων** είναι το μοντέλο που χρησιμοποιείται ευρέως σήμερα
  - υ Κύριο χαρακτηριστικό: **σχέση**, βασικά ένας πίνακας με γραμμές και στήλες
  - υ Κάθε σχέση έχει ένα **σχήμα**, το οποίο περιγράφει τις στήλες (ή αλλιώς τα πεδία)



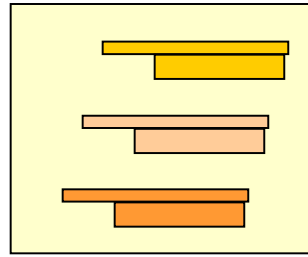
# Επίπεδα Αφαίρεσης ενός ΣΔΒΔ

- λ Σε ένα ΣΔΒΔ, τα δεδομένα αναπαριστώνται σε 3 επίπεδα αφαίρεσης (abstraction levels):
- υ **Φυσικό επίπεδο** (physical level): το χαμηλότερο επίπεδο αφαίρεσης
    - περιλαμβάνει σύνθετες δομές και λεπτομέρειες αναπαράστασης και αποθήκευσης
  - υ **Λογικό επίπεδο** (logical level): περιγράφει τα δεδομένα και τις μεταξύ τους σχέσεις χρησιμοποιώντας το μοντέλο δεδομένων
    - οι χρήστες οι οποίοι βλέπουν το σύστημα στο λογικό επίπεδο δεν χρειάζεται να γνωρίζουν τις λεπτομέρειες του φυσικού επιπέδου
  - υ **Εξωτερικό Επίπεδο** (external level): το υψηλότερο επίπεδο αφαίρεσης
    - περιγράφει μέρος των αποθηκευμένων δεδομένων, καθώς οι χρήστες χρειάζονται πρόσβαση συνήθως μόνο σε υποσύνολο της ΒΔ
    - αποτελεί συγχρόνως την βάση για **μηχανισμούς ασφάλειας**
- λ Η δυνατότητα να τροποποιούμε το σχήμα περιγραφής των δεδομένων μας σε κάποιο επίπεδο χωρίς να επηρεάζονται τα ανώτερα επίπεδα ονομάζεται **ανεξαρτησία των δεδομένων**

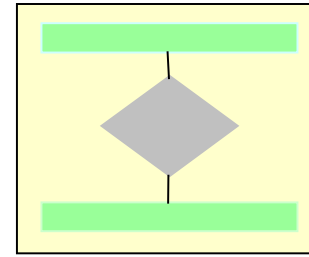
# Αρχιτεκτονική ANSI-SPARC



VIEW 1



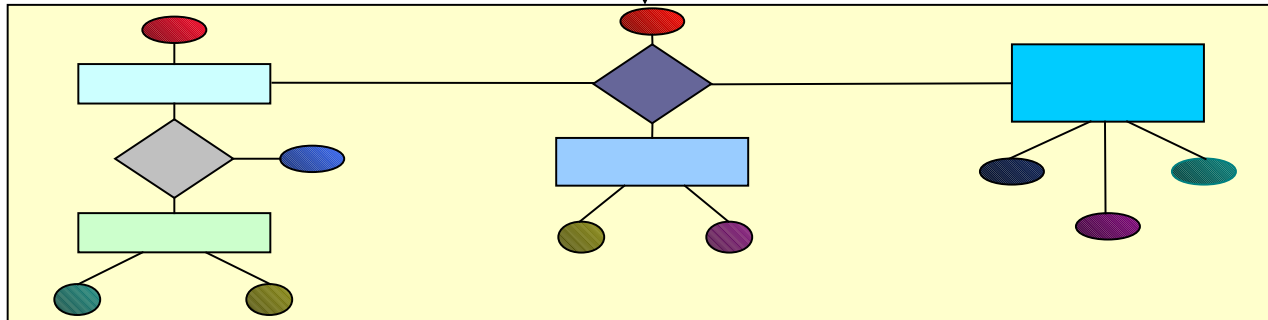
VIEW 2



VIEW 3

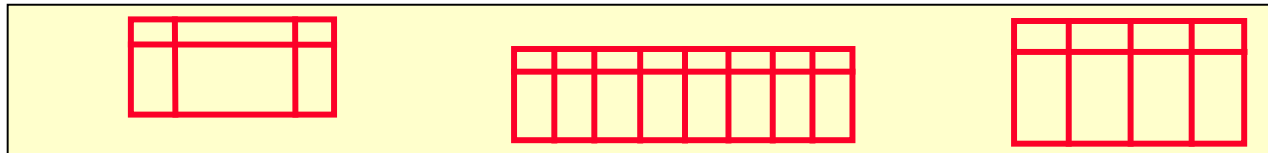
Εξωτερικό  
Επίπεδο:

INTEGRATION

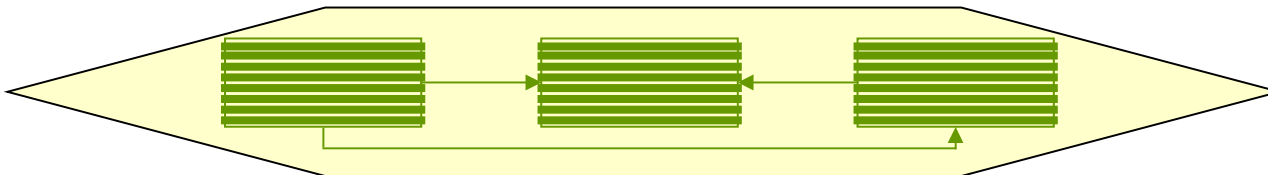


Λογικό  
Επίπεδο

Λογικό  
σχήμα



Φυσικό  
σχήμα



Εσωτερικό  
Επίπεδο



# Ανεξαρτησία Δεδομένων

- λ **Φυσική** (για βελτιστοποίηση και συντήρηση)
  - υ Αλλαγές στις φυσικές δομές δεν επηρεάζουν την λογική οργάνωση των δεδομένων
  - υ Παράδειγμα: Προσθήκες/αφαιρέσεις ευρετηρίων σε φυσικό επίπεδο
- λ **Λογική** (για ευκολότερη διαχείριση)
  - υ Η τροποποίηση της λογικής δομής των δεδομένων δεν πρέπει να επηρεάζει τα προγράμματα εφαρμογών του εξωτερικού επιπέδου
  - υ Παράδειγμα: Προσθήκη νέων σχέσεων και πεδίων στο λογικό σχήμα ενός ΣΔΒΔ
- λ **Χρήστες** (για ευελιξία)
  - υ Πολλαπλές όψεις στο ίδιο λογικό σχήμα
  - υ Παράδειγμα: Οι γραμματείς και οι καθηγητές ενός πανεπιστημίου δεν χρειάζεται να χρησιμοποιούν τις ίδιες όψεις μιας βάσης
- ➔ **Ερώτηση: Γιατί η ανεξαρτησία είναι τόσο σημαντική?**
  - υ Γιατί ο ρυθμός των αλλαγών που λαμβάνουν χώρα σε εφαρμογές **ΣΔΒΔ** είναι πολύ μικρός συγκρινόμενος με τον ρυθμό αλλαγών στο υπολογιστικό περιβάλλον (συσκευές, αποθήκευση δεδομένων, κλπ.)

# Αποθήκευση Δεδομένων

- λ Ένα ΣΔΒΔ μπορεί να υλοποιηθεί χρησιμοποιώντας απ'ευθείας ένα «τυπικό» σύστημα διαχείρισης και επεξεργασίας αρχείων (file processing system). Όμως η προσέγγιση αυτή πάσχει από διάφορα μειονεκτήματα:
- υ Πλεονασμός (redundancy) και ασυνέπεια (inconsistency) δεδομένων
    - η ίδια πληροφορία πιθανόν να επαναλαμβάνεται σε διαφορετικά αρχεία με αποτέλεσμα να υπάρχει μεγαλύτερο κόστος αποθήκευσης και ανάκτησης
    - επιπλέον, η ενημέρωση ενός αρχείου πρέπει να συνοδεύεται από την ενημέρωση όλων των αρχείων στα οποία η πληροφορία επαναλαμβάνεται
  - υ Δυσκολία πρόσβασης (access) στα δεδομένα
    - τα συστήματα διαχείρισης αρχείων δεν παρέχουν αποδοτικούς μηχανισμούς για αναζήτηση και ανάκτηση πληροφορίας με διαφορετικούς τρόπους πρόσβασης
    - επιπλέον, τα δεδομένα διασκορπίζονται σε αρχεία που χρησιμοποιούν διαφορετικές μορφές

# Αποθήκευση Δεδομένων

- υ Προβλήματα **ακεραιότητας** (integrity) δεδομένων
  - τα δεδομένα υπόκεινται σε **περιορισμούς** (constraints) **συνέπειας** (consistency) ή **ακεραιότητας** (integrity)
  - Οι περιορισμοί αυτοί ελέγχονται μέσω των προγραμμάτων εφαρμογών τα οποία φυσικά θα πρέπει να τροποποιηθούν για την προσθήκη νέων ή τη μεταβολή των υπαρχόντων περιορισμών
- υ Προβλήματα **ατομικότητας** (atomicity) δοσοληψιών
  - Π.χ. μεταφορά ενός ποσού  $X$  από λογαριασμό  $A$  σε λογαριασμό  $B$ . Αν συμβεί κάποιο σφάλμα κατά τη διάρκεια της μεταφοράς, ο λογαριασμός  $A$  μπορεί να χρεωθεί χωρίς να πιστωθεί ο λογαριασμός  $B$
  - Οι **δοσοληψίες** πρέπει να χαρακτηρίζονται από ατομικότητα, δηλαδή, ή όλη η δοσοληψία ολοκληρώνεται ή κανένα μέρος αυτής
  - το σύστημα πρέπει να εγγυάται ότι σε περίπτωση **σφάλματος** η πληροφορία μπορεί να αποκατασταθεί σε **προηγούμενη συνεπή κατάσταση**

# Αποθήκευση Δεδομένων

- υ Ανωμαλίες ταυτόχρονης πρόσβασης (concurrent access)
  - Η ταυτόχρονη ανάγνωση και ενημέρωση της πληροφορίας από πολλούς χρήστες μπορεί να δημιουργήσει προβλήματα συνέπειας
  - Π.χ. έστω ότι ο λογαριασμός A έχει υπόλοιπο X. Αν δύο χρήστες αφαιρέσουν ποσά  $\Delta 1$  και  $\Delta 2$  "ταυτόχρονα" από τον A, το αποτέλεσμα μπορεί να μην είναι ορθό
- υ Προβλήματα ασφάλειας (security)
  - διαφορετικοί χρήστες θα πρέπει να έχουν διαφορετικά δικαιώματα πρόσβασης σε αρχεία ή σε συγκεκριμένα δεδομένα

# Έλεγχος Ταυτόχρονης Πρόσβασης

- λ Ταυτόχρονη εκτέλεση πολλών προγραμμάτων: Αυτή είναι το κλειδί για την καλή απόδοση ενός ΣΔΒΔ
  - υ Η συχνή πρόσβαση στον δίσκο προκαλεί καθυστερήσεις
  - υ Στόχος είναι να υπάρχουν πολλά προγράμματα τα οποία θα χρησιμοποιούν τον κεντρικό επεξεργαστή όταν τα υπόλοιπα περιμένουν δεδομένα από τον δίσκο
- λ Οι ενέργειες διαφορετικών προγραμμάτων μπορούν να αλληλοεπιδρούν: αυτό αποτελεί Πρόβλημα!
  - υ Π.χ μεταφορές από τραπεζικούς λογαριασμούς, εκτυπώσεις την ίδια στιγμή στον εκτυπωτή
- λ Ένα ΣΔΒΔ εξασφαλίζει ότι δεν θα εμφανίζονται τέτοια προβλήματα
  - υ Χρήστες/προγραμματιστές μπορούν να θεωρούν ότι μόνο αυτοί χρησιμοποιούν το σύστημα ( “απομόνωση”)
  - υ Ευτυχώς! Δεν χρειάζεται να προγραμματίσουμε «πολύ, πολύ προσεκτικά»

# Δοσοληψία: Μια εκτέλεση ενός προγράμματος ΒΔ

- λ Η έννοια της δοσοληψίας είναι πολύ σημαντική
  - υ μια ακολουθία από ατομικές ενέργειες σε μια ΒΔ (διάβασμα/γράψιμο)
- λ Κάθε δοσοληψία η οποία εκτελείται πλήρως έχει ως αποτέλεσμα μια συνεπή κατάσταση της ΒΔ
- λ Οι χρήστες μπορούν να καθορίσουν απλούς περιορισμούς συνέπειας ή ακεραιότητας των δεδομένων
  - υ Το ΣΔΒΔ εφαρμόζει αυτούς τους κανόνες
  - υ Παρόλα αυτά, το ΣΔΒΔ δεν καταλαβαίνει την σημασιολογία των δεδομένων
  - υ Η εξασφάλιση ότι μια δοσοληψία η οποία εκτελείται μόνη της έχει συνεπή αποτελέσματα είναι αποκλειστική ευθύνη του χρήστη!

# Εξασφάλιση Καλών Ιδιοτήτων των Δοσοληψιών

- λ Το ΣΔΒΔ εξασφαλίζει την **ατομικότητα** (είτε εκτελείται εξολοκλήρου ή καθόλου) ακόμα και αν το σύστημα διακόψει απρόσμενα την λειτουργία του (crashes) στην μέση της εκτέλεσης μιας δοσοληψίας
- λ Το ΣΔΒΔ εξασφαλίζει την **μονιμότητα** των αποτελεσμάτων των δοσοληψιών που έχουν ολοκληρωθεί (committed transactions) ακόμα και αν το σύστημα έχει διακοπή λειτουργίας
- λ **Ιδέα**: Να κρατείται ένα **ιστορικό (log)** όλων των ενεργειών οι οποίες λαμβάνουν χώρα στο ΣΔΒΔ όταν εκτελούνται δοσοληψίες:
  - υ Πριν γίνει οποιαδήποτε αλλαγή στην βάση, δημιουργείται η αντίστοιχη εγγραφή ιστορικού η οποία και αποθηκεύεται σε ασφαλές μέρος
  - υ Μετά από μια απρόσμενη διακοπή, γίνονται τα εξής:
    - Όλα τα αποτελέσματα που έχουν δημιουργηθεί από **δοσοληψίες οι οποίες δεν τελείωσαν την εκτέλεση τους, αναιρούνται (aborted)**. Αυτό γίνεται με χρήση του αρχείου ιστορικού
    - Όλα τα αποτελέσματα που έχουν δημιουργηθεί από **δοσοληψίες οι οποίες τελείωσαν την εκτέλεση τους, μονιμοποιούνται (committed)**. Αυτό γίνεται με χρήση του αρχείου ιστορικού
  - υ Φυσικά η υπηρεσία αυτή είναι πιο πολύπλοκη από ότι ακούγεται!

# Το Ιστορικό (Log)

- λ Οι ακόλουθες ενέργειες καταγράφονται σε ένα ιστορικό:
  - υ Δι γράφει νέα τιμή σε ένα αντικείμενο: η παλιά και νέα τιμή πρέπει να αποθηκευτούν στο ιστορικό πριν γίνει η αλλαγή στην ΒΔ!
  - υ Δι μονιμοποιείται/αναιρείται (commits/aborts): μια εγγραφή ιστορικού υποδεικνύει την αντίστοιχη ενέργεια
  
- λ Οι εγγραφές ιστορικού συσχετίζονται με την ταυτότητα (id) μιας δοσοληψίας, έτσι είναι εύκολο να ακυρωθούν (undo) όταν π.χ. πέφτουμε σε αδιέξοδο (deadlock)
  - υ Το ιστορικό συνήθως αποθηκεύεται διπλά σε ένα ασφαλές χώρο
  
- λ Όλες οι σχετικές με το ιστορικό δραστηριότητες (και κατά συνέπεια, όλες οι δραστηριότητες σχετικά με το έλεγχο της ταυτόχρονης πρόσβασης όπως κλείδωμα/ξεκλείδωμα, αντιμετώπιση αδιεξόδων κτλ.) διαχειρίζονται με διαφανή τρόπο από το ΣΔΒΔ



# Οι Συνιστώσες ενός ΣΔΒΔ

## λ Συλλογή Δεδομένων:

- υ Ελαχιστοποίηση της πλεονάζουσας πληροφορίας
- υ Δυνατότητα ταυτόχρονης πρόσβασης

## λ Άνθρωποι:

- υ Ο σχεδιαστής της ΒΔ
- υ Ο διαχειριστής της ΒΔ
- υ Ο προγραμματιστής εφαρμογών
- υ Ο τελικός χρήστης

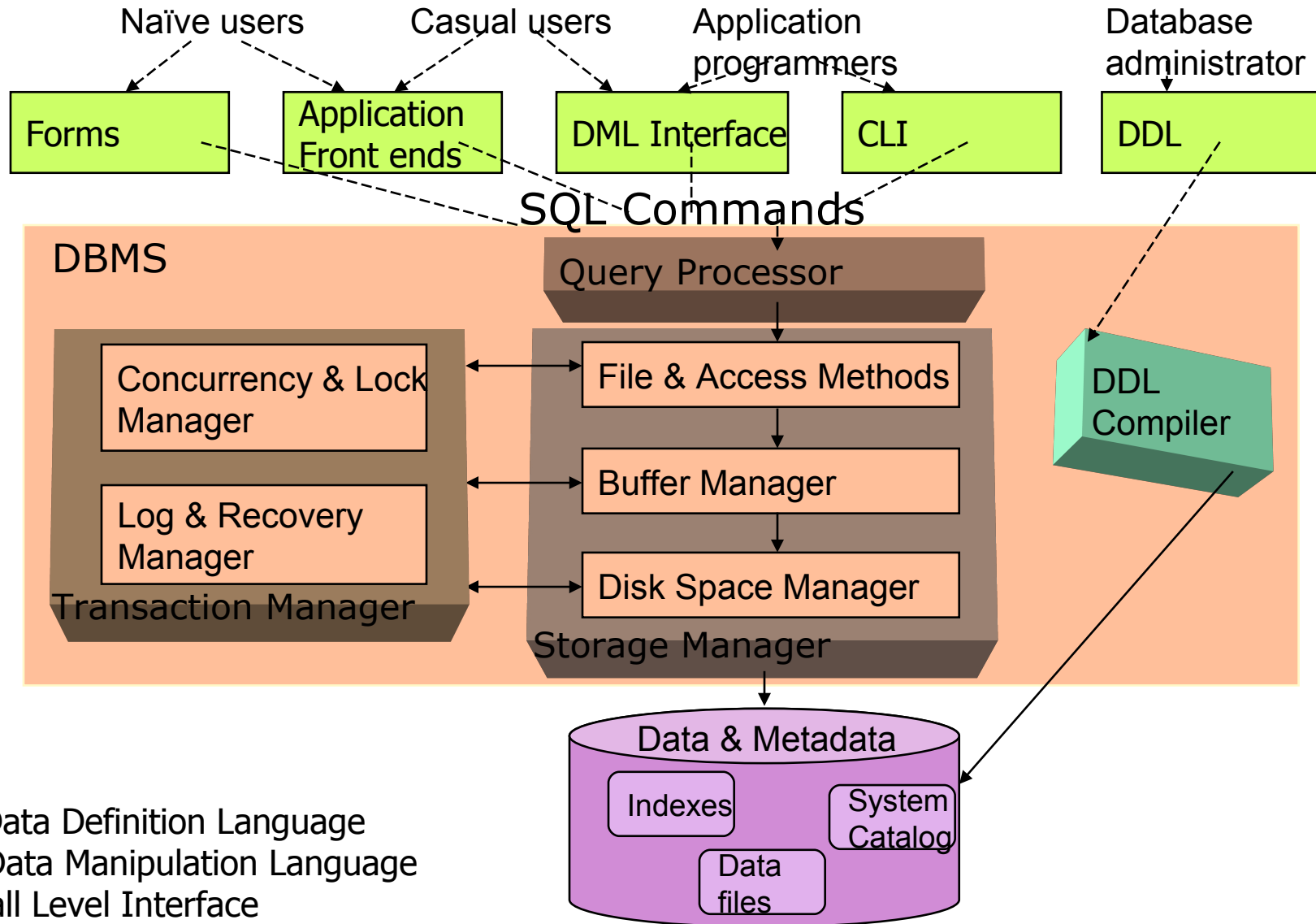
## λ Υλικό:

- υ Επεξεργαστής(ες)
- υ Κύρια/Ενδιάμεση Μνήμη
- υ Δευτερεύοντες αποθηκευτικοί χώροι(π.χ., δίσκοι, CD-ROMs, κτλ.)

## Λογισμικό

- υ Ο διαχειριστής αποθήκευσης (**Storage Manager**): διαχειρίζεται τα αρχεία, την ενδιάμεση μνήμη και την πρόσβαση στον δίσκο
- υ Ο διαχειριστής των δοσοληψιών (**Transaction Manager**): κλείδωμα, ιστορικό, and καλές ιδιότητες δοσοληψιών (ACID)
- υ Ο επεξεργαστής των ερωτήσεων (**Query Processor**): συντακτική ανάλυση, βελτιστοποίηση, εκτέλεση
- υ Διάφορα **Εργαλεία Αυτοματοποίησης εργασιών**: Βοήθειες Ανάπτυξης και Σχεδίασης, Γεννήτριες Αναφορών, κλπ.

# Τυπική Αρχιτεκτονική ενός ΣΔΒΔ



# Τυπική Αρχιτεκτονική ενός ΣΔΒΔ

## Διαχειριστής Αποθήκευσης

- λ Παρέχει μια διεπαφή μεταξύ των δεδομένων τα οποία είναι αποθηκευμένα στην δευτερεύουσα μνήμη και των προγραμμάτων εφαρμογών καθώς και των επερωτήσεων που υποβάλλονται στο σύστημα.
- λ Είναι υπεύθυνος για την αποδοτική αποθήκευση, ανάκτηση και ενημέρωση των δεδομένων
  - υ Ο **διαχειριστής ενδιάμεσης μνήμης** (buffer manager)
    - Είναι υπεύθυνος για τον διαμερισμό της διαθέσιμης μνήμης σε **αποταμιευτές**, π.χ., περιοχές μνήμης μεγέθους μιας **σελίδας** στις οποίες μεταφέρονται τα μπλοκ των δεδομένων από τον δίσκο
    - **Επιτρέπει** στα ψηλότερα επίπεδα του ΣΔΒΔ να **υποθέσουν ότι όλα τα δεδομένα που χρειάζονται βρίσκονται στην κύρια μνήμη**
  - υ Ο **διαχειριστής του χώρου των δίσκων** (disc space manager)
    - Βασίζεται στην έννοια της **σελίδας** ως μονάδας δεδομένων και είναι υπεύθυνος για την δέσμευση/αποδέσμευση, όπως επίσης και για το διάβασμα/ γράψιμο αυτών των σελίδων στον δίσκο
    - **Επιτρέπει** στα ψηλότερα επίπεδα του ΣΔΒΔ να **θεωρούν ότι γίνεται αποτελεσματική χρήση του διαθέσιμου χώρου στον δίσκο** και ότι **υπάρχει γρήγορη πρόσβαση στα αρχεία της βάσης δεδομένων**

# Τυπική Αρχιτεκτονική ενός ΣΔΒΔ

## Διαχειριστής των δοσοληψιών

- λ Εξασφαλίζει ότι η ΒΔ παραμένει σε συνεπή κατάσταση ανεξαρτήτως των αποτυχιών του συστήματος (π.χ., απώλεια ενέργειας ή αποτυχιών του λειτουργικού συστήματος) καθώς και αποτυχιών των δοσοληψιών.
- λ Είναι υπεύθυνος για την ενημέρωση του ιστορικού (logging), το κατάλληλο κλείδωμα των δεδομένων (locking) και την αντιμετώπιση των αδιεξόδων (deadlock)
  - υ Ο **διαχειριστής ταυτόχρονης πρόσβασης** (concurrency control manager) ή διαφορετικά ο **δρομολογητής** (scheduler)
    - Ελέγχει την ταυτόχρονη εκτέλεση πολλών δοσοληψιών, εξασφαλίζοντας ότι το αποτέλεσμα τους θα ήταν το ίδιο εάν αυτές εκτελούνταν σειριακά
    - **Εγγυάται την συνέπεια (consistency) και την απομόνωση (isolation)** των δοσοληψιών
  - υ Ο **διαχειριστής ιστορικού και ανάνηψης από αποτυχίες** (log and recovery manager)
    - Είναι υπεύθυνος για την μη απώλεια δοσοληψιών καθώς επίσης και για την εκτέλεση των δοσοληψιών, των ενημερώσεων και εγγραφών στον δίσκο μία και μόνο μία φορά
    - **Εγγυάται την ατομικότητα (atomicity) και την διάρκεια (durability)** των δοσοληψιών

# Τι είναι η Ρύθμιση ενός ΣΔΒΔ?

- λ Η ρύθμιση (tuning) είναι μια δραστηριότητα η οποία επιτρέπει σε μια εφαρμογή ΒΔ να εκτελείται πιο γρήγορα
  - υ “Πιο γρήγορα” σημαίνει **μεγαλύτερη απόδοση** (throughput), η οποία σε ορισμένες περιπτώσεις συνεπάγεται **μικρότερο μέσο χρόνο απόκρισης** (response time) για εφαρμογές κρίσιμες στον χρόνο εκτέλεσης τους
  - υ Μια βελτίωση της τάξης 5% είναι πολύ σημαντική!
  
- λ Ρύθμιση ορισμένων παραμέτρων και σχεδιαστικών επιλογών με στόχο την βελτίωση της απόδοσης του ΣΔΒΔ για μια συγκεκριμένη εφαρμογή. Η ρύθμιση ενός ΣΔΒΔ γίνεται σε 3 επίπεδα:
  - υ **Υλικού** – π.χ. , πρόσθεση δίσκων που έχουν μεγαλύτερη ταχύτητα σε E/E, πρόσθεση μνήμης για καλύτερη χρήση των αποταμιευτών (buffer hits), ή τοποθέτηση γρηγορότερου επεξεργαστή
  - υ **Παραμέτρων του ΣΔΒΔ** – π.χ. Καθορισμός του μεγέθους των αποταμιευτών για την ελαχιστοποίηση της άσκοπης μεταφοράς σελίδων (page fault), καθορισμός σημείων ελέγχου (checkpointing) για τον περιορισμό του μεγέθους του αρχείου ιστορικού
  - υ **Σχεδιαστικών επιλογών υψηλότερου επιπέδου**, – π.χ. λογικό (κανονικοποίηση) και φυσικό σχήμα (ευρετήρια) μιας ΒΔ, καθώς επίσης και την **αναγνώριση** και τον **περιορισμό** των **σημείων συμφόρησης** (bottlenecks) του συστήματος π.χ. δοσοληψιών ή επερωτήσεων οι οποίες **τρέχουν για ώρες άσκοπα**

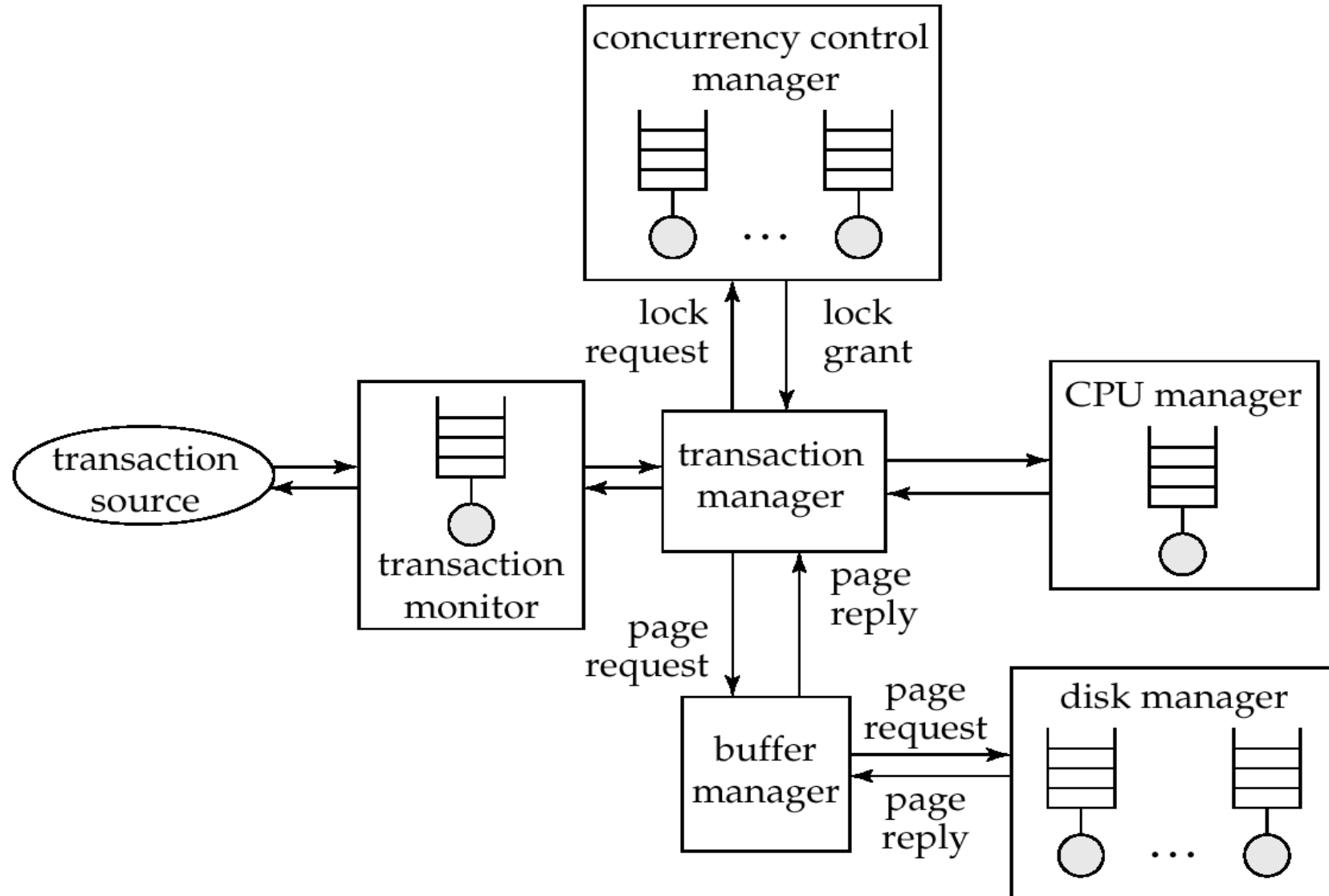
# Τι είναι η Ρύθμιση ενός ΣΔΒΔ?

- λ Η απόδοση του συστήματος (τουλάχιστον πριν από την ρύθμιση) περιορίζεται συνήθως από ένα ή περισσότερα σημεία **συμφόρησης**
  - υ **Κανόνας 80/20**: 80% του φορτίου (επερωτήσεις/δοσοληψίες) εκτελείται στο 20% του χρόνου ενώ το υπόλοιπο 20% στο 80% του χρόνου
- λ Τα σημεία συμφόρησης μπορεί να **οφείλονται σε προβλήματα υλικού** (π.χ. οι δίσκοι είναι πολύ απασχολημένοι, η CPU είναι ανενεργή), ή ακόμα και **σε προβλήματα λογισμικού**
  - υ Αναιρώντας ένα σημείο συμφόρησης συχνά δημιουργούμε ένα άλλο
  - υ Επαναληπτική διαδικασία αναγνώρισης και αποφυγής όλων των σημείων συμφόρησης (ευρηστική μέθοδος)
- λ Για παράδειγμα έστω ότι δοσοληψίες ζητούν με σειρά από υπηρεσίες π.χ. CPU, Disk I/O, locks
  - υ Με την ταυτόχρονη εκτέλεση των δοσοληψιών κάποιες δοσοληψίες μπορεί να περιμένουν πόρους οι οποίοι χρησιμοποιούνται από άλλες

# Αναγνώριση των Σημείων Συμφόρησης

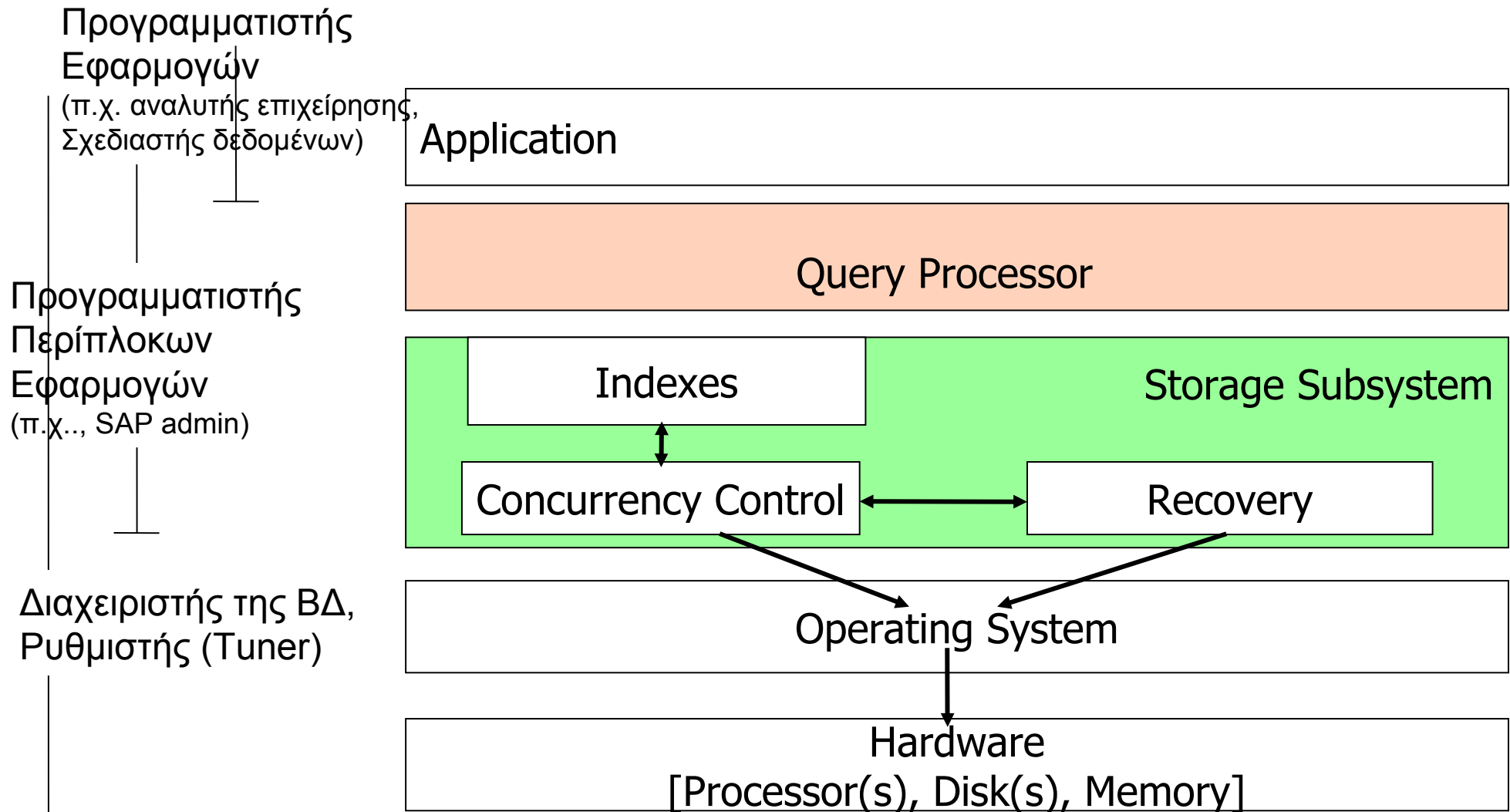
- λ Μπορούμε να μοντελοποιήσουμε ένα ΣΔΒΔ ως ένα **σύστημα ουρών** με μια ουρά για κάθε υπηρεσία
  - υ Οι δοσοληψίες επαναληπτικά εκτελούν τα ακόλουθα:
    - Ζητούν μια υπηρεσία, περιμένουν στην αντίστοιχη ουρά και εξυπηρετούνται όταν έρθει η σειρά τους
  
- λ Τα σημεία συμφόρησης σε ένα ΣΔΒΔ **εμφανίζονται εκεί που υπάρχουν μεγάλες ουρές αναμονής** και συνεπώς μεγάλη χρήση της αντίστοιχης υπηρεσίας
  - υ Π.χ. Χρήση του δίσκου ή της CPU
  - υ 100% χρήση οδηγεί σε μεγάλο χρόνο αναμονής:
    - **Χρυσός Κανόνας**: το σύστημα σχεδιάζεται για μέγιστη χρήση στο 70%
    - Η χρήση πάνω από 90% πρέπει να αποφεύγεται

# Ουρές σε ένα ΣΔΒΔ





# Ρόλοι σε ένα ΣΔΒΔ



# Τέλος Ενότητας



Ευρωπαϊκή Ένωση  
Ευρωπαϊκό Κοινωνικό Ταμείο



Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



# Χρηματοδότηση

- Το παρόν εκπαιδευτικό υλικό έχει αναπτυχθεί στα πλαίσια του εκπαιδευτικού έργου του διδάσκοντα.
- Το έργο «**Ανοικτά Ακαδημαϊκά Μαθήματα στο Πανεπιστήμιο Κρήτης**» έχει χρηματοδοτήσει μόνο τη αναδιαμόρφωση του εκπαιδευτικού υλικού.
- Το έργο υλοποιείται στο πλαίσιο του Επιχειρησιακού Προγράμματος «Εκπαίδευση και Δια Βίου Μάθηση» και συγχρηματοδοτείται από την Ευρωπαϊκή Ένωση (Ευρωπαϊκό Κοινωνικό Ταμείο) και από εθνικούς πόρους.



**Σημειώματα**

# Σημείωμα αδειοδότησης

•Το παρόν υλικό διατίθεται με τους όρους της άδειας χρήσης Creative Commons Αναφορά, Μη Εμπορική Χρήση, Όχι Παράγωγο Έργο 4.0 [1] ή μεταγενέστερη, Διεθνής Έκδοση. Εξαιρούνται τα αυτοτελή έργα τρίτων π.χ. φωτογραφίες, διαγράμματα κ.λ.π., τα οποία εμπεριέχονται σε αυτό και τα οποία αναφέρονται μαζί με τους όρους χρήσης τους στο «Σημείωμα Χρήσης Έργων Τρίτων».



[1] <http://creativecommons.org/licenses/by-nc-nd/4.0/>

•Ως **Μη Εμπορική** ορίζεται η χρήση:

- που δεν περιλαμβάνει άμεσο ή έμμεσο οικονομικό όφελος από την χρήση του έργου, για το διανομέα του έργου και αδειοδόχο
- που δεν περιλαμβάνει οικονομική συναλλαγή ως προϋπόθεση για τη χρήση ή πρόσβαση στο έργο
- που δεν προσπορίζει στο διανομέα του έργου και αδειοδόχο έμμεσο οικονομικό όφελος (π.χ. διαφημίσεις) από την προβολή του έργου σε διαδικτυακό τόπο

•Ο δικαιούχος μπορεί να παρέχει στον αδειοδόχο ξεχωριστή άδεια να χρησιμοποιεί το έργο για εμπορική χρήση, εφόσον αυτό του ζητηθεί.

# Σημείωμα Αναφοράς

Copyright Πανεπιστήμιο Κρήτης, Δημήτρης Πλεξουσάκης. «**Συστήματα Διαχείρισης Βάσεων Δεδομένων. Εισαγωγή**». Έκδοση: 1.0. Ηράκλειο/Ρέθυμνο 2015. Διαθέσιμο από τη δικτυακή διεύθυνση: <http://www.csd.uoc.gr/~hy460/>