



ΕΛΛΗΝΙΚΗ ΔΗΜΟΚΡΑΤΙΑ
ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΡΗΤΗΣ

Εισαγωγή στις Βάσεις Δεδομένων II

Ενότητα: Εισαγωγή II - Σχεδίαση και Αρχιτεκτονική

Διδάσκων: Πηγουνάκης Κωστής
ΤΜΗΜΑ ΟΙΚΟΝΟΜΙΚΩΝ ΕΠΙΣΤΗΜΩΝ
ΣΧΟΛΗ ΚΟΙΝΩΝΙΚΩΝ ΕΠΙΣΤΗΜΩΝ



Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο



Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



Άδειες Χρήσης

- Το παρόν εκπαιδευτικό υλικό υπόκειται στην άδεια χρήσης **Creative Commons** και ειδικότερα ***Αναφορά – Μη εμπορική Χρήση – Όχι Παράγωγο Έργο 3.0 Ελλάδα*** (***Attribution – Non Commercial – Non-derivatives 3.0 Greece***)



[ή επιλογή ενός άλλου από τους έξι συνδυασμούς]

[και αντικατάσταση λογότυπου άδειας όπου αυτό έχει μπει (σελ. 1, σελ. 2 και τελευταία)]

- Εξαιρείται από την ως άνω άδεια υλικό που περιλαμβάνεται στις διαφάνειες του μαθήματος, και υπόκειται σε άλλου τύπου άδεια χρήσης. Η άδεια χρήσης στην οποία υπόκειται το υλικό αυτό αναφέρεται ρητώς.

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.



Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο



ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ

Εισαγωγή στις Βάσεις Δεδομένων II

Εισαγωγή στις Σχεσιακές Βάσεις Δεδομένων (ΕΒΔΟ100)

Κωστής Πηγουνάκης

Τελευταία ενημέρωση : 17/2/2013

Αρχές καλής σχεδίασης

- **Ακεραιότητα**
- **Αξιοπιστία**
- **Αποτελεσματικότητα**
- **Ταχύτητα**
- **Ασφάλεια**

Χαρακτηριστικά σχεδίασης 1

- Δεν θα πρέπει να περιλαμβάνει **περιττά δεδομένα (redundancy)**. Αυτό σημαίνει ότι τα ίδια δεδομένα δεν θα πρέπει να καταχωρούνται στη βάση δύο ή περισσότερες φορές.
- **Διαφορετικά :**
 - **σπαταλούμε άσκοπα αποθηκευτικό χώρο**
 - **υπάρχει ο κίνδυνος δημιουργίας ασυνεπών δεδομένων (inconsistency)**
- Για το λόγο αυτό, ένας από τους πρώτους ελέγχους που πραγματοποιούμε στη βάση αμέσως μετά το σχεδιασμό της, είναι ο **έλεγχος παρουσίας επαναλαμβανόμενων πεδίων, και η απομάκρυνσή τους, εφ' όσον υπάρχουν.**

Χαρακτηριστικά σχεδίασης 2

- Η βάση θα πρέπει να είναι σχεδιασμένη έτσι ώστε τα δεδομένα που περιλαμβάνει να ανακτώνται **εύκολα** και **γρήγορα**.
- **Διαφορετικά** : η βάση είναι εξαιρετικά δυσκίνητη και αναποτελεσματική.
- Η σωστή σχεδίαση επιτυγχάνεται με εφαρμογή επί της δομής της βάσης, μιας τεχνικής, η οποία ονομάζεται **κανονικοποίηση (normalization)**.

Χαρακτηριστικά σχεδίασης 3

- Το ΣΔΒΔ θα πρέπει να χαρακτηρίζεται από **σύστημα ασφάλειας (security system)** που να απαγορεύει την πρόσβαση στα δεδομένα μη εξουσιοδοτημένων ατόμων.
- Η τεχνική που συνήθως εφαρμόζεται, συνίσταται στον καθορισμό **ομάδων χρηστών (user groups)** με διαφορετικά δικαιώματα πρόσβασης στον καθένα από αυτούς.
- Ο κάθε χρήστης λαμβάνει ένα **κωδικό πρόσβασης (password)** και τα καθήκοντα που μπορεί να επιτελέσει είναι εντελώς συγκεκριμένα και καθορισμένα εκ των προτέρων.

Χαρακτηριστικά σχεδίασης 4

- Το ΣΔΒΔ θα πρέπει να μπορεί να διαχειρίζεται **ταυτόχρονες προσπελάσεις** πάνω στα ίδια δεδομένα (**concurrency control**), ώστε να αποτρέψει την καταχώρηση πληροφοριών στην ίδια θέση αποθήκευσης.
- Αυτό επιτυγχάνεται με το «κλείδωμα» των δεδομένων τη στιγμή που δοθεί μια εντολή εγγραφής/αλλαγής σε αυτά.

Χαρακτηριστικά σχεδίασης 5

- Το ΣΔΒΔ θα πρέπει να διαθέτει σύστημα δημιουργίας **αντιγράφων ασφαλείας (backups)** των δεδομένων που είναι καταχωρημένα σε αυτή.
- Η ταυτόχρονη αποθήκευση των δεδομένων σε περισσότερα από ένα σημεία, είναι μια εργασία επιβεβλημένη, προκειμένου να είναι δυνατή η ανάκτησή τους σε περιπτώσεις κατάρρευσης της βάσης για οποιοδήποτε λόγο.

Αρχιτεκτονικές Σ.Δ.Β.Δ.

Οι κοινοί τρόποι σχεδιασμού του λογισμικού σε ό,τι αφορά τις επιμέρους μονάδες (*modules*) και τη μεταξύ τους επικοινωνία χαρακτηρίζονται ως **αρχιτεκτονικά μοντέλα** ή **αρχιτεκτονικές**.

Οι βασικές αρχιτεκτονικές των σύγχρονων ΣΔΒΔ κατηγοριοποιούνται ανάλογα με τις **βαθμίδες (tiers)** που εμπλέκονται.

Ο όρος των βαθμίδων ανάγεται στη λογική **αρχιτεκτονική P-A-D** (*Presentation-Application-Data*)

Βασίζεται στην αλληλεπίδραση των τριών λογικών διαδικασιών:

- **Παρουσίαση – Presentation** : η διεπαφή χρήστη (user interface)
- **Εφαρμογή – Application** : η επεξεργασία ή λογική εφαρμογής
- **Δεδομένα – Data** : η διαχείριση

Εξυπηρετητές και Πελάτες

Η ανάπτυξη μεγάλων εφαρμογών, επικεντρωμένων στην ανταλλαγή δεδομένων μέσω δικτύου, οδήγησε στο **περιβάλλον Πελάτη/Εξυπηρετητή (Client/Server)** στη δεκαετία του 1980.

Εξυπηρετητής (Server) : μια κεντρική εφαρμογή με την οποία συνδέονται και ανταλλάσσουν πληροφορίες άλλες εφαρμογές που βρίσκονται στους υπολογιστές των χρηστών και χαρακτηρίζονται ως **Πελάτες (Clients)** .

Η ανταλλαγή της πληροφορίας γίνεται μέσω *συναλλαγών (transactions)*, δηλαδή μέσω καλά ορισμένων *αιτημάτων (requests)* και *αποκρίσεων (responses)*.



Αδύνατοι και Παχείς

Το περιβάλλον Πελάτη/Εξυπηρετητή αφορά στη φυσική αρχιτεκτονική και εμφανίζει επιμέρους αρχιτεκτονικές.

Όταν το μεγαλύτερο μέρος της επεξεργασίας (λογική της εφαρμογής) γίνεται στον Εξυπηρετητή και ο Πελάτης αναλαμβάνει κυρίως την παρουσίαση αποτελεσμάτων της διαδικασίας, τότε έχουμε την περίπτωση *Αδύνατου Πελάτη (Thin Client) – Παχύ Εξυπηρετητή (Fat Server)*.

Όταν ο Πελάτης αναλαμβάνει το μεγαλύτερο μέρος της επεξεργασίας μιλάμε για *Παχύ Πελάτη (Fat Client) – Αδύνατο Εξυπηρετητή (Thin Server)*.

Στην ειδική περίπτωση του Διαδικτύου, όπου αποκλειστικά και μόνο το επίπεδο της παρουσίασης είναι στο φυλλομετρητή (browser), ενώ το σύνολο της επεξεργασίας είναι στον Εξυπηρετητή, μιλάμε για *Υπέρ-Παχύ Εξυπηρετητή (Super-Fat Server)*.

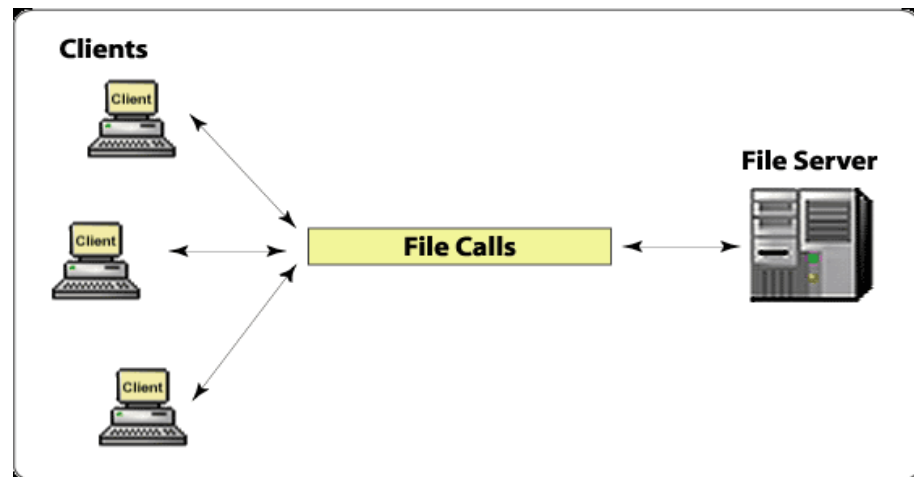


Αρχιτεκτονική 1-tier

Ο υπολογιστής που φιλοξενεί την εφαρμογή είναι ο ίδιος με το οποίο ο χρήστης επικοινωνεί με την εφαρμογή.

Τυπική περίπτωση είναι η ανάπτυξη μιας βάσης στο ΣΔΒΔ MS Access.

Μια επέκταση είναι όταν ένας file server φιλοξενεί τα αρχεία αποθήκευσης του ΣΔΒΔ (αλλά όχι κατ' ανάγκη το ίδιο) και βρίσκεται σε τοπικό δίκτυο με άλλους υπολογιστές.



Τα + και τα – του 1-tier

Πλεονεκτήματα:

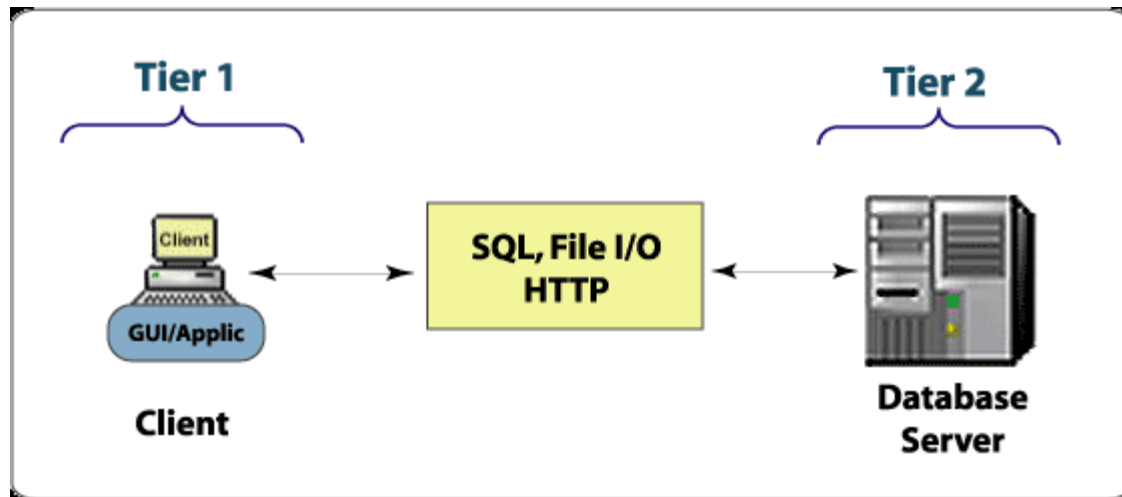
- Ο διαμοιρασμός δεδομένων σε πολλαπλούς χρήστες γίνεται με απλό τρόπο, αν και οι δυνατότητες είναι περιορισμένες.
- Το κόστος αποθήκευσης είναι χαμηλό και μοιράζεται στους διασυνδεδεμένους χρήστες.
- Επειδή η ίδια εφαρμογή ΣΔΒΔ εγκαθίσταται πολλές φορές, ενδεχομένως επιτυγχάνεται χαμηλότερη τιμή αγοράς ανά εγκατάσταση (site license).

Μειονεκτήματα:

- Μικρή δυνατότητα κλιμάκωσης.
- Αδυναμία ταυτόχρονης χρήσης.

Αρχιτεκτονική 2-tier

Ένα κεντρικό σύστημα, πολλές φορές με αποκλειστική αρμοδιότητα (server), παρέχει τη λειτουργικότητα για πολλά συστήματα χρηστών (clients). Το ΣΔΒΔ είναι στον εξυπηρετητή, με τον οποίο επικοινωνούν οι εφαρμογές – πελάτες.



Τα + και τα – του 2-tier

Πλεονεκτήματα:

- Κλιμάκωση, οριζόντια (αύξηση πελατών) και κάθετη (αύξηση εξυπηρετητών).
- Αποτελεσματική χρήση των υπολογιστικών πόρων με καλό επίπεδο ασφάλειας.
- Εύκολη ενσωμάτωση νέας τεχνολογίας (π.χ. κινητές συσκευές).
- Μαζί με τον client μπορεί να υπάρχουν και άλλες εφαρμογές.

Μειονεκτήματα:

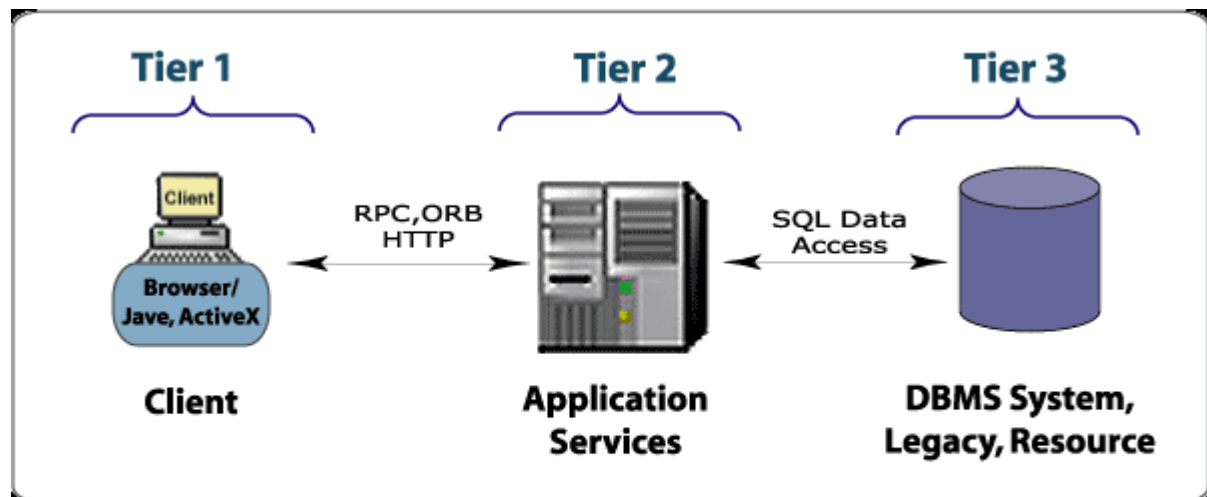
- Η client εφαρμογή εγκαθίσταται σε πολλά συστήματα, πράγμα που προκαλεί δυσκολία αλλαγής της (π.χ. αναβάθμισης) και, για το λόγο αυτό, το κόστος συντήρησης αυξάνεται σημαντικά.

Αρχιτεκτονική N-tier

Η λειτουργικότητα της εφαρμογής του ΣΔΒΔ χωρίζεται σε διαφορετικά επίπεδα που διασυνδέονται λογικά και μοιράζονται πληροφορίες.

Κάθε ενδιάμεσο επίπεδο αποτελεί ένα μοντέλο client/server.

Η πιο συνήθης υλοποίηση είναι αυτή των τριών βαθμίδων (3-tier), όπου υπάρχει σαφής φυσικός διαχωρισμός της επεξεργασίας από τη βάση δεδομένων και την παρουσίαση.



Τα + και τα – του N-tier

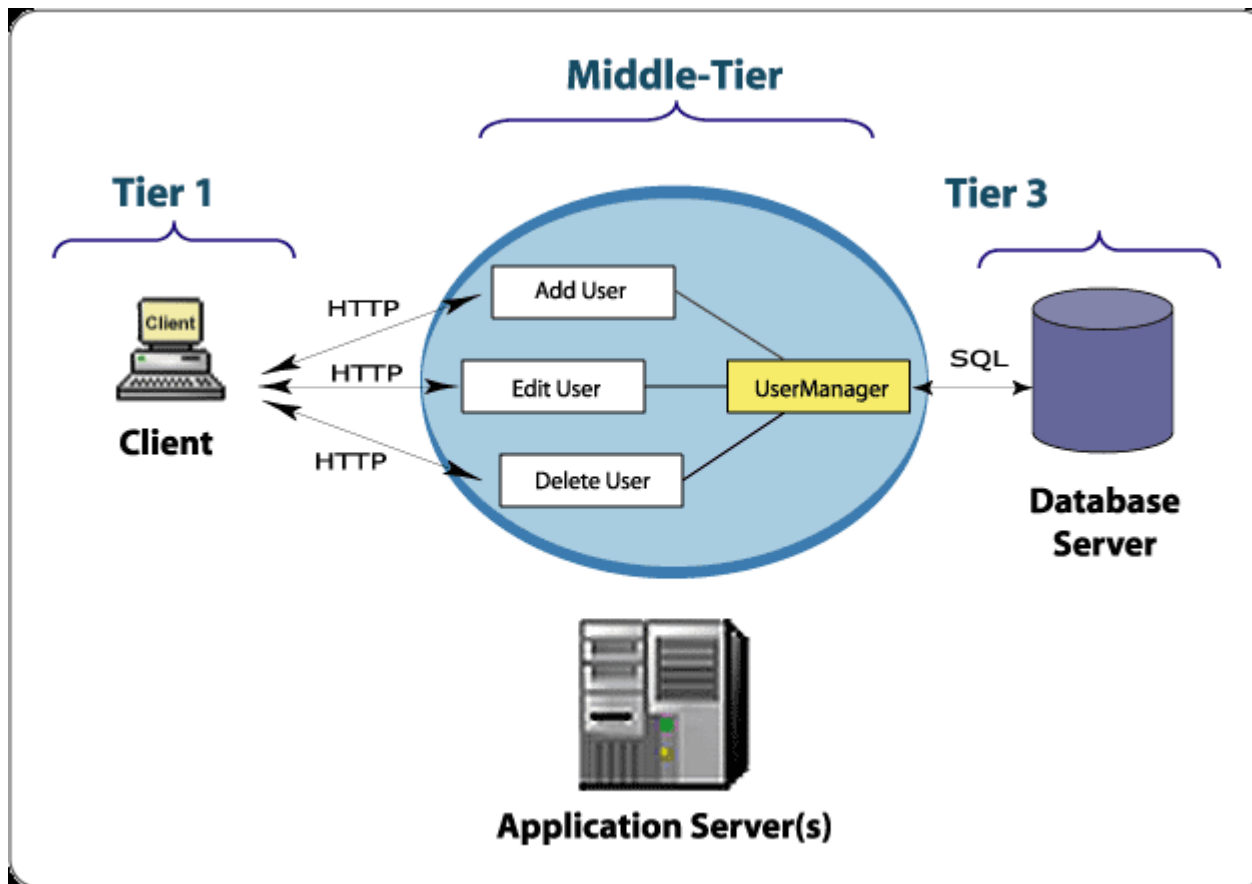
Πλεονεκτήματα:

- Αυξημένη ασφάλεια, αφού ο χρήστης δεν προσπελαίνει άμεσα τη βάση δεδομένων
- Αυξημένη ευελιξία και έλεγχος λόγω των ενδιάμεσων επιπέδων και, επομένως, καλύτερη απόδοση
- Ελαχιστοποιημένη διαχείριση σε σχέση με το μοντέλο client/server

Μειονεκτήματα:

- Πολυπλοκότητα της σχεδίασης
- Αυξημένο αρχικό κόστος υλοποίησης

Αρχιτεκτονική διαδικτύου



Τα + και τα – του διαδικτύου

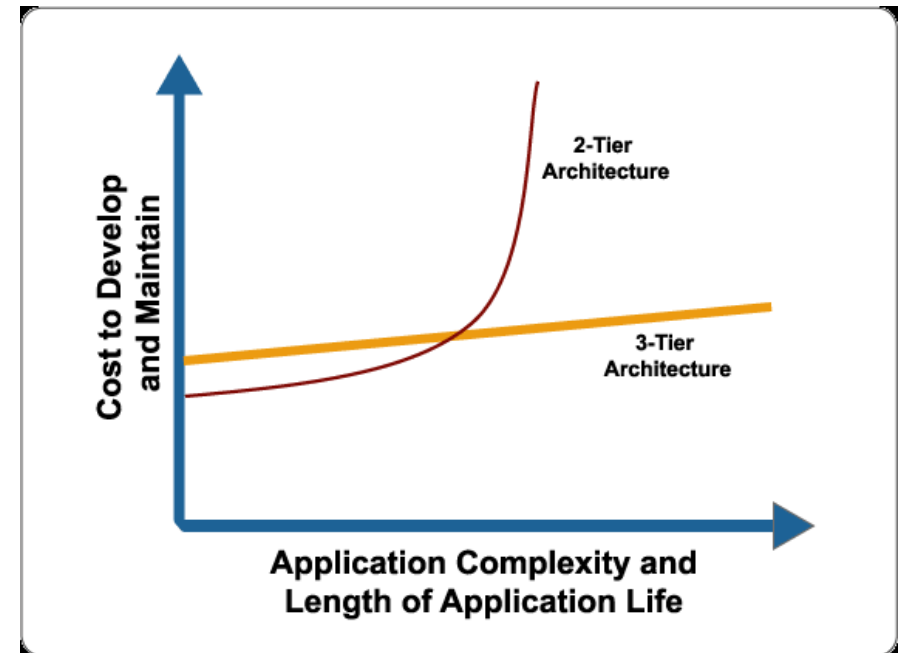
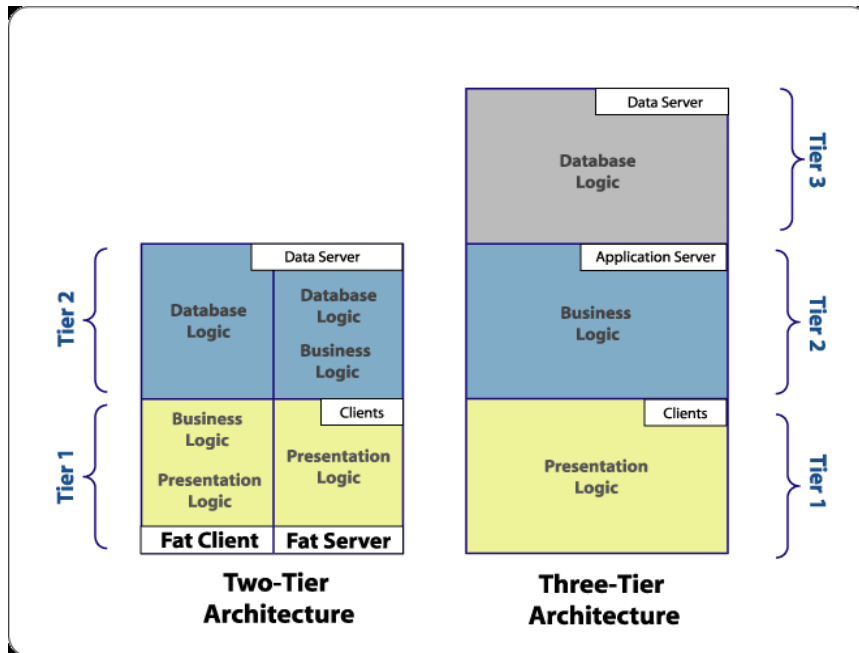
Πλεονεκτήματα

- Αυξημένη ασφάλεια, αφού ο χρήστης δεν προσπελαύνει άμεσα τη βάση δεδομένων
- Αυξημένη ευελιξία και έλεγχος και, επομένως, καλύτερη απόδοση
- Ελαχιστοποιημένη διαχείριση σε σχέση με το μοντέλο client/server
- Χαμηλές απαιτήσεις για την πλευρά του χρήστη

Μειονεκτήματα

- Αυξημένοι κίνδυνοι που αντιμετωπίζονται μέσω ισχυρών πολιτικών ασφάλειας (firewalls, λογισμικό ανίχνευσης εισβολέων, κ.ά.).

Σύγκριση 2-tier και 3-tier



Στάδια Ζωής Βάσεων Δεδομένων

- ♦ **Στάδιο 1:** Σχεδίαση και υλοποίηση της βάσης
- ♦ **Στάδιο 2:** Καταχώρηση των δεδομένων στη βάση (data entry)
- ♦ **Στάδιο 3:** Διαχείριση της βάσης δεδομένων και επεξεργασία των δεδομένων

Επίπεδα Σχεδίασης Β.Δ.

- **Φυσικό επίπεδο (physical level):** σχεδιάζεται και αναλύεται ο τρόπος με τον οποίο τα δεδομένα της βάσης θα αποθηκεύονται στο ΣΔΒΔ. Στο επίπεδο αυτό, καθορίζονται, μεταξύ άλλων, οι τύποι δεδομένων που θα χρησιμοποιηθούν.
- **Εννοιολογικό ή λογικό επίπεδο (conceptual / logical level):** καθορίζεται ο τρόπος που τα δεδομένα και οι σχέσεις που υφίστανται ανάμεσά τους θα μοντελοποιηθούν στη βάση του συστήματος.
- **Επίπεδο όψης (view level):** καθορίζεται ο τρόπος που ένα μέρος των δεδομένων θα προβάλλεται στο χρήστη σύμφωνα με καθορισμένα κριτήρια που εκείνος θέτει με άμεσο ή έμμεσο τρόπο.

Εμείς επικεντρωνόμαστε στο λογικό επίπεδο γιατί εκεί γίνεται κυρίως η μοντελοποίηση της βάσης δεδομένων.

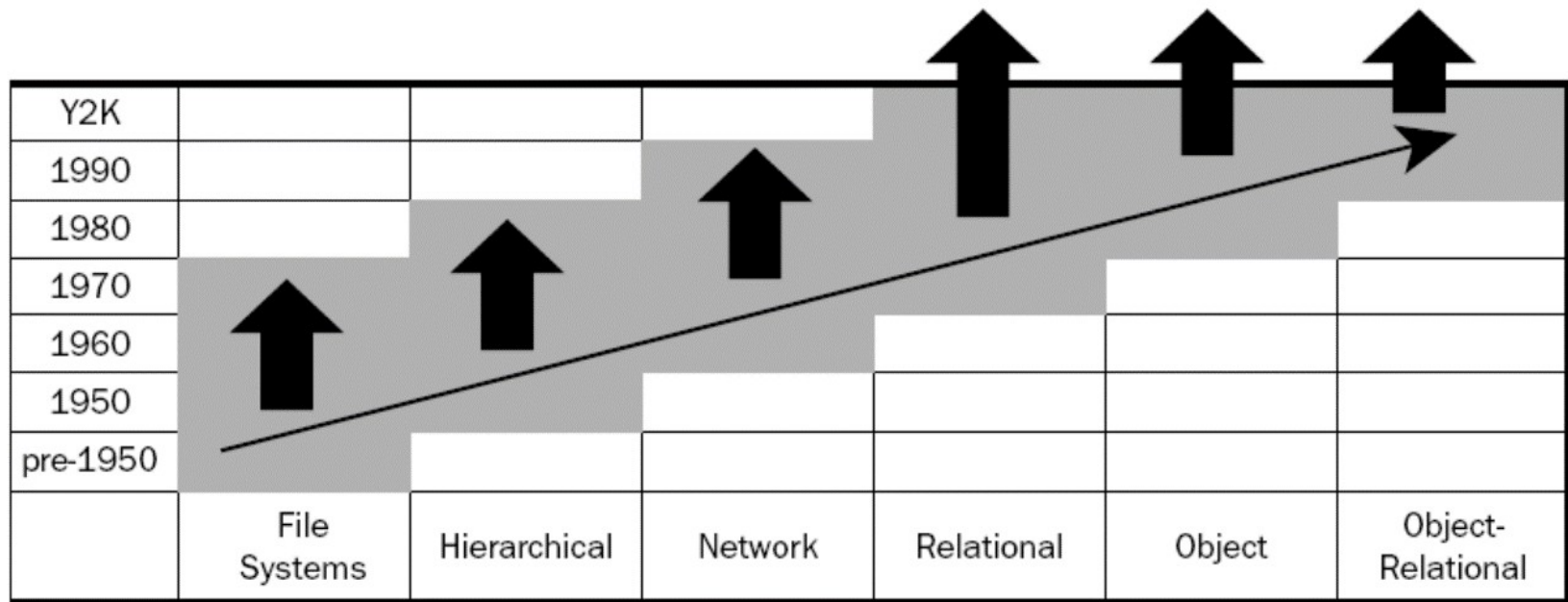
Μοντέλα λογικού επιπέδου

Τα μοντέλα σχεδίασης εννοιολογικού-λογικού επιπέδου δεν αφορούν απλά στους τρόπους οργάνωσης δεδομένων, αλλά προχωρούν στο να ορίσουν και τις πράξεις που τελούνται στα δεδομένα.

Η ιστορική εξέλιξη των μοντέλων είναι:

- το ιεραρχικό μοντέλο (hierarchical model),
- το δικτυωτό μοντέλο (network model),
- το σχεσιακό μοντέλο (relational model) και
- το αντικειμενοστραφές μοντέλο (object-oriented model).

Μοντέλα λογικού επιπέδου



Μοντέλο Επίπεδου Αρχείου

Αρχείο κειμένου (text file) που περιέχει συλλογή ακολουθιών δεδομένων, διαχωρισμένα με ειδικούς χαρακτήρες, που μπορούν να διερευνηθούν **σειριακά** για τον εντοπισμό τους.

Η σειριακή αναζήτηση δεδομένων είναι αναποτελεσματική, αλλά μπορεί να είναι χρήσιμη για μικρούς καταλόγους και για απλές δομές δεδομένων.

Η διαχείριση γίνεται μέσω προγραμμάτων που διαβάζουν και γράφουν στα αρχεία.

Προβλήματα :

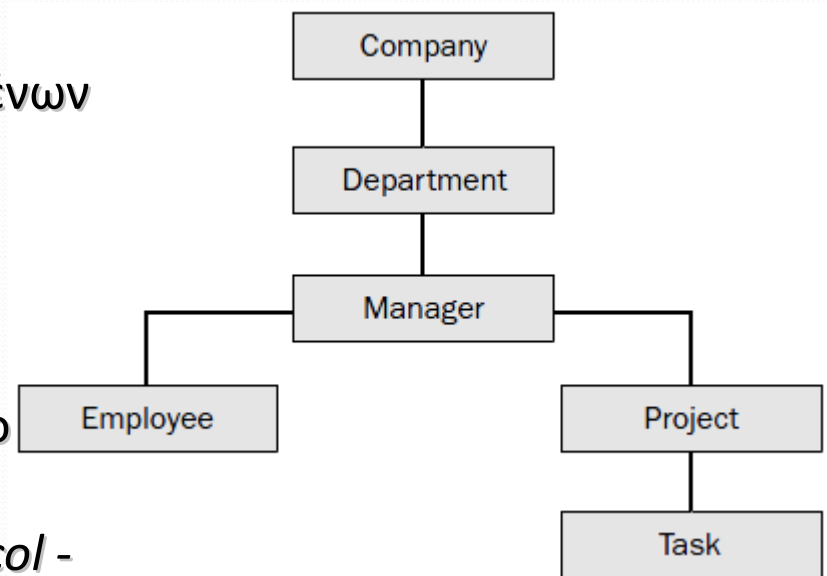
- ταυτόχρονη εγγραφή από περισσότερες από μία εφαρμογές,
- απώλεια δεδομένων λόγω κατάρρευσης μιας εφαρμογής κατά τη διάρκεια προσπέλασης του αρχείου.

Ιεραρχικό μοντέλο (1)

- Τα δεδομένα οργανώνονται σε δομές που ακολουθούν τη λογική του **ανεστραμμένου δένδρου**.
- Τις δομές αυτές μπορούμε να τις θεωρήσουμε **πίνακες καταχωρήσεων - record tables**, οι οποίοι χαρακτηρίζονται από **δείκτες - pointers**.
- Οι μεταξύ των πινάκων σχέσεις είναι του τύπου **πρόγονος/απόγονος - parent/child** δηλ. δημιουργούνται σχέσεις «ένα προς πολλά». Αυτό καθίσταται δυνατό μέσω των δεικτών που φέρουν οι πίνακες.
- Ο πρώτος πίνακας-πρόγονος καλείται **ρίζα-root**, και έχει έναν ή περισσότερους πίνακες – απογόνους.
- Κάθε πίνακας, μπορεί να έχει έναν ή περισσότερους απογόνους, αλλά **μόνο έναν πρόγονο**.

Ιεραρχικό μοντέλο (2)

- Η αναζήτηση είναι **μονοσήμαντη** μέσω των «κλαδιών», ξεκινώντας από τη ρίζα.
- Η ακεραιότητα αναφοράς των δεδομένων είναι **εγγενής** και δεν χρειάζεται ιδιαίτερος μηχανισμός για να επιτευχθεί.
- Ένα τυπικό παράδειγμα οργάνωσης σύμφωνα με την ιεραρχικό μοντέλο είναι το πρότυπο καταλόγου *Lightweight Directory Access Protocol - LDAP*.



Ιεραρχικό μοντέλο (3)

Τα σοβαρότερα μειονεκτήματα είναι :

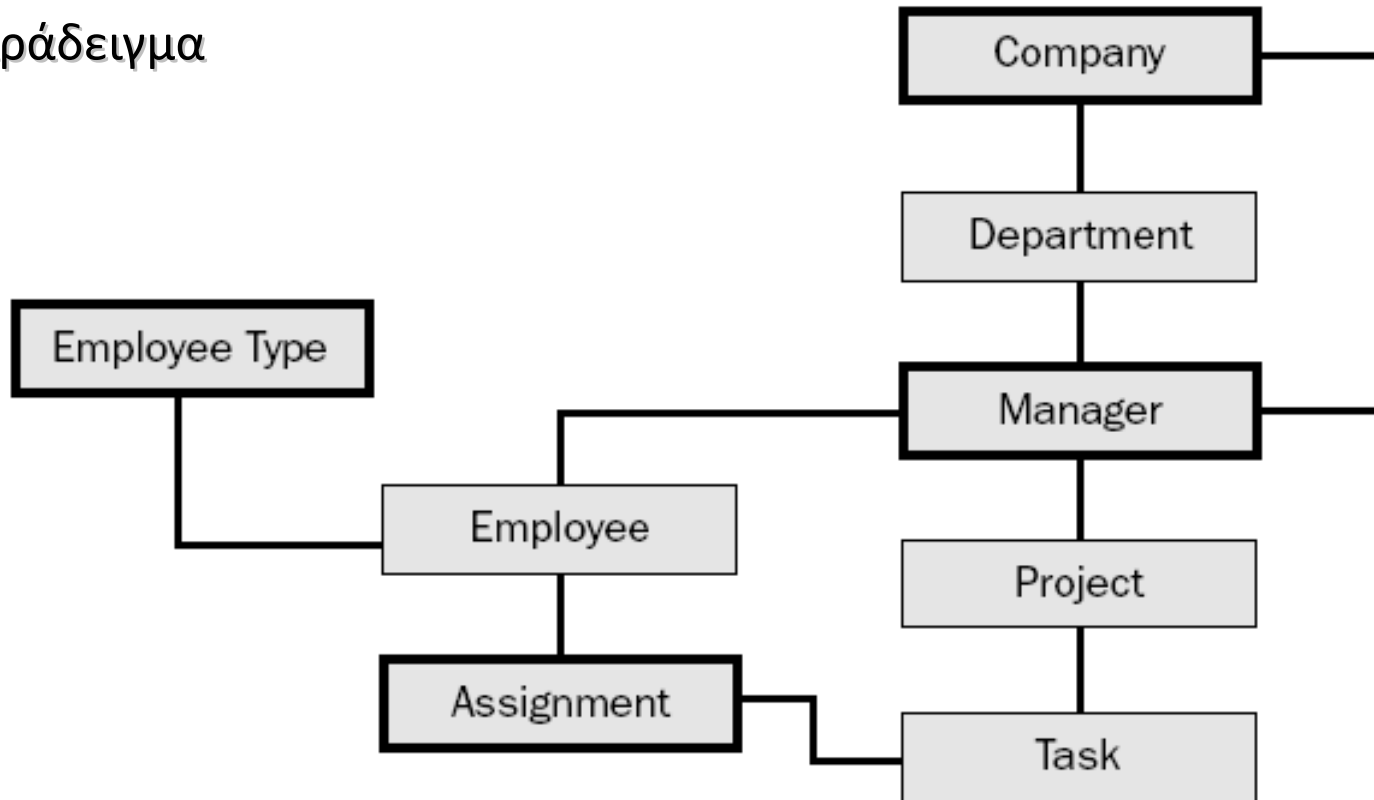
- η ανάγκη εκκίνησης της αναζήτησης από τη ρίζα
- η μεγάλη δυσκολία αλλαγής της δομής του «δέντρου», η οποία επηρεάζει, όχι μόνο την ίδια τη βάση δεδομένων, αλλά και τις εφαρμογές που τη χειρίζονται.
- λόγω της στενής σχέσης των πινάκων, όταν διαγράφεται ένας πρόγονος, πρέπει να διαγραφούν και οι απόγονοί του.

Δικτυακό μοντέλο (1)

- Αποτελεί βελτίωση του ιεραρχικού μοντέλου
- Επιτρέπει σε κάθε απόγονο να έχει πολλούς προγόνους, δηλ. επιτρέπει τη δημιουργία σχέσεων «πολλά προς πολλά».
- Οι πίνακες συνδέονται μεταξύ τους δημιουργώντας ένα **δίκτυο συνδέσεων**, και ουσιαστικά συνδυάζονται μεμονωμένες ιεραρχικές βάσεις δεδομένων σε μια μεγαλύτερη.
- Η πλοήγηση δεν απαιτεί την έναρξη από κάποια ρίζα, αλλά **μπορεί να αρχίσει από οποιαδήποτε πίνακα**.
-

Δικτυακό μοντέλο (2)

Παράδειγμα



Σχεσιακό Μοντέλο

- Βασίζεται στις σχέσεις που αναπτύσσονται μεταξύ των πινάκων, ενώ αλλάζει η έννοια της ιεραρχίας και δεν αποτελεί το κυρίαρχο χαρακτηριστικό.
- Τα δεδομένα οργανώνονται σε πίνακες που καλούνται **οντότητες (entities)**.
- Οι **συσχετίσεις (relations)** μεταξύ των οντοτήτων αποδίδονται επίσης με πίνακες.
- Οι οντότητες και οι συσχετίσεις καλούνται συνολικά **ΣΧΕΣΕΙΣ**
- Η αναζήτηση στις σχεσιακές βάσεις δεδομένων γίνεται **απ' ευθείας στον πίνακα** που έχει τα δεδομένα που μας ενδιαφέρουν, χωρίς να πρέπει να ακολουθηθεί μια δενδρική διαδρομή.

Αντικειμενοστραφές μοντέλο

- Ενώ οι σχεσιακές ΒΔ προορίζονται για την ανεύρεση δεδομένων σε δύο διαστάσεις (πίνακες σειρών), οι **αντικειμενοστραφείς** είναι αποδοτικές στην εξεύρεση ενός μοναδικού στοιχείου. Αυτό τις καθιστά **αποδοτικότερες σε νέους τύπους δεδομένων** όπως ήχου και εικόνας (σταθερής ή video).
- Οι αντικειμενοστραφείς ΒΔ **προσπελαύνονται άμεσα από γλώσσες** όπως η Java και η C++, πράγμα που τις κάνει **10 έως 1000 φορές ταχύτερες** από τις αντίστοιχες σχεσιακές.
- Αν και σχεδιαστικά οι αντικειμενοστραφείς ΒΔ ξεπερνούν πολλές αδυναμίες των σχεσιακών, πρακτικά παρουσιάζουν το μειονέκτημα ότι η διείσδυσή τους στην αγορά είναι μικρή σε σύγκριση με τις σχεσιακές βάσεις. Αυτό σημαίνει **λιγότερους εξειδικευμένους προγραμματιστές και μη διασφάλιση εκτέλεσης των εφαρμογών σε διαφορετικές πλατφόρμες.**

Μοντέλο Συσχέτισης Αντικειμένων

- **Υβριδικό μοντέλο** Συσχέτισης Αντικειμένων – *Object Related* της Oracle8, που εισήχθησαν στο τέλος της δεκαετίας του 1990.
- Ενσωματώνει αντικειμενοστραφή χαρακτηριστικά σε μια σχεσιακή βάση δεδομένων.
- Επιτρέπει τη δημιουργία νέων τύπων δεδομένων, αν και η βασική αρχιτεκτονική παραμένει σχεσιακή.
- Μια τέτοια βάση δεδομένων χρησιμοποιεί το FBI για καταχώρηση και ανάλυση δακτυλικών αποτυπωμάτων

Είδη λειτουργιών και γλωσσών

- Λειτουργίες που τροποποιούν τη **δομή** : γλώσσα ορισμού δεδομένων (**Data Definition Language, DDL**),
- Λειτουργίες που τροποποιούν το **περιεχόμενο** : γλώσσα διαχείρισης δεδομένων (**Data Manipulation Language, DML**)

Η γλώσσα SQL

- **Δομημένη Γλώσσα Ερωτοαποκρίσεων - Structured Query Language, SQL**
- Απαντάται σε όλα ανεξαιρέτως τα συστήματα διαχείρισης σχεσιακών βάσεων δεδομένων
- Εμφανίζεται σε πολλές παραλλαγές αλλά η βασική ιδέα που περιγράφει τη λειτουργία της, είναι ουσιαστικά η ίδια.
- Οι βασικές λειτουργίες της διαχωρίζονται σε 4 κατηγορίες:
 - η εισαγωγή δεδομένων (data insertion),
 - η διαγραφή δεδομένων (data deletion),
 - η ανάκτηση δεδομένων (data retrieval) και
 - η τροποποίηση δεδομένων (data update).

<http://w3schools.com/sql/default.asp>

Τέλος Ενότητας



Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο



Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

