
1

Εισαγωγή στις Βάσεις Δεδομένων

Το βασικό χαρακτηριστικό των σύγχρονων εφαρμογών είναι η απαίτηση για την αποτελεσματική και αποδοτική διαχείριση της αποθηκευμένης πληροφορίας. Αρχικά, η διαχείριση της πληροφορίας γινόταν από τις ίδιες τις εφαρμογές, χρησιμοποιώντας το **σύστημα αρχείων** (file system) του λειτουργικού συστήματος, ενώ οι μέθοδοι επεξεργασίας και ανάκτησης των δεδομένων ήταν ενσωματωμένες στον κώδικα της εκάστοτε εφαρμογής. Ο τρόπος αυτός διαχείρισης, αν και δίνει στον προγραμματιστή την ευελιξία να υλοποιήσει τις μεθόδους επεξεργασίας που αυτός επιθυμεί, ωστόσο δημιουργεί προβλήματα καθώς αυξάνεται ο όγκος των δεδομένων και η πολυπλοκότητα της πληροφορίας. Για παράδειγμα, αν στόχος της εφαρμογής είναι η διαχείριση αριθμητικών δεδομένων και υπάρχει η ανάγκη για αποθήκευση και διαχείριση μερικών εκατοντάδων αριθμών, τότε αυτό μπορεί να υλοποιηθεί εύκολα αποθηκεύοντας τους αριθμούς σε ένα αρχείο του λειτουργικού συστήματος, και υλοποιώντας τις απαραίτητες λειτουργίες υπολογισμών (π.χ., μέσος όρος, ελάχιστη τιμή, μέγιστη τιμή) στη λογική της εφαρμογής. Ας θεωρήσουμε όμως μία πολυπλοκότερη εφαρμογή που στόχο έχει την οργάνωση μίας ηλεκτρονικής βιβλιοθήκης, όπου απαιτείται η αναζήτηση άρθρων ή βιβλίων με βάση το συγγραφέα, τον τίτλο, την ημερομηνία έκδοσης, ενώ ακόμη απαιτείται η αναζήτηση άρθρων με βάση λέξεις-κλειδιά (keywords) από όλο το κείμενο. Είναι

προφανές ότι θα απαιτηθεί αρκετή προσπάθεια από τον προγραμματιστή για την υλοποίηση της εφαρμογής λαμβάνοντας υπ' όψη ότι:

- Αναμένεται το πλήθος των άρθρων ή των βιβλίων να είναι μεγάλος, με αποτέλεσμα να επιβάλλεται η χρήση αποδοτικών μεθόδων αναζήτησης.
- Η δομή της πληροφορίας είναι περίπλοκη, διότι ένας συγγραφέας μπορεί να έχει συμβάλει στη συγγραφή πολλών άρθρων, ενώ στη συγγραφή ενός άρθρου μπορεί να έχουν συμβάλει πολλοί συγγραφείς.
- Πρέπει να βρεθεί ένας αποτελεσματικός και αποδοτικός τρόπος αναζήτησης των λέξεων μέσα στο κείμενο. Προφανώς, η εξέταση κάθε κειμένου ξεχωριστά έχει απαγορευτικό κόστος λόγω του μεγάλου αριθμού των κειμένων λαμβάνοντας μάλιστα υπ' όψη ότι ένα κείμενο μπορεί να είναι αρκετά εκτενές.
- Υπάρχει η ανάγκη για εισαγωγή νέων στοιχείων, όπως για παράδειγμα η ενημέρωση των δεδομένων όταν υπάρχει ένα νέο άρθρο που πρέπει να καταχωρισθεί.
- Αναμένεται ότι πολλοί χρήστες θα χρησιμοποιούν την εφαρμογή ταυτόχρονα, με αποτέλεσμα να απαιτούνται ειδικοί μηχανισμοί προσπέλασης και συγχρονισμού των προσπελάσεων από διαφορετικούς χρήστες.

Ας εξετάσουμε με περισσότερη λεπτομέρεια ποιά είναι τα βασικά μειονεκτήματα της αποθήκευσης και διαχείρισης των δεδομένων απ' ευθείας σε αρχεία του λειτουργικού συστήματος. Στη συνέχεια θα εξετάσουμε την εναλλακτική λύση που προσφέρει η οργανωμένη διαχείριση των δεδομένων από εξειδικευμένα συστήματα, δίνοντας τους απαραίτητους ορισμούς.

- Ο προγραμματιστής της εφαρμογής είναι υπεύθυνος για τη σωστή ενημέρωση των αρχείων των δεδομένων, αναλόγως με τις εισαγωγές και διαγραφές που πραγματοποιούνται. Για παράδειγμα, έστω ότι στο αρχείο A1 αποθηκεύονται τα στοιχεία ενός άρθρου (π.χ., τίτλος, αριθμός σελίδων), ενώ στο αρχείο A2 αποθηκεύονται τα ονόματα των συγγραφέων κάθε άρθρου. Κατά την εισαγωγή ενός νέου άρθρου θα πρέπει να ενημερωθεί το αρχείο A1 με τα απαραίτητα στοιχεία, ενώ το αρχείο A2 θα πρέπει να ενημερωθεί με τα στοιχεία των συγγραφέων του άρθρου. Επίσης, κατά την αναζήτηση στοιχείων θα πρέπει σε πρώτο στάδιο να προσδιορισθούν τα απαιτούμενα αρχεία, ενώ σε δεύτερο στάδιο πρέπει να γίνει προσεκτική ανάγνωση των δεδομένων, συνδυάζοντας κατάλληλα τις πληροφορίες που

περιέχουν τα διαφορετικά αρχεία. Η κατάσταση δυσκολεύει όσο αυξάνεται ο αριθμός των αρχείων δεδομένων και όσο αυξάνεται η πολυπλοκότητα των ερωτημάτων.

- Η δομή του κάθε αρχείου καθορίζεται από την ίδια την εφαρμογή. Αυτό έχει ως αποτέλεσμα να είναι δύσκολη η αλλαγή της δομής ενός αρχείου. Πολλές φορές παρουσιάζεται η ανάγκη να προστεθεί ένα νέο χαρακτηριστικό στα δεδομένα. Για παράδειγμα, μπορεί να απαιτηθεί η προσθήκη στο αρχείο A1 ενός ακόμη πεδίου που να περιέχει το πλήθος των βιβλιογραφικών αναφορών (citations) που περιέχονται στο άρθρο. Σε μία τέτοια περίπτωση πρέπει αφ' ενός να μεταβληθεί η δομή του αρχείου A1, έτσι ώστε να δημιουργηθεί χώρος για το νέο πεδίο, και αφ' ετέρου να μεταβληθεί η λογική της εφαρμογής ώστε να ληφθεί το νέο πεδίο υπ' όψη κατά την αναζήτηση, εισαγωγή και διαγραφή δεδομένων. Οι αλλαγές αυτές είναι χρονοβόρες και επικίνδυνες για την ακεραιότητα των αποθηκευμένων δεδομένων.
- Εφ' όσον ο τρόπος δημιουργίας των αρχείων καθορίζεται από την εφαρμογή, η μορφοποίηση των αρχείων εξαρτάται από την εκάστοτε γλώσσα προγραμματισμού που χρησιμοποιεί ο προγραμματιστής (όπως C, C++, Java, Visual Basic). Το γεγονός αυτό δημιουργεί δυσκολίες όταν απαιτείται η χρήση των αρχείων από διαφορετικές εφαρμογές ή όταν απαιτείται η κατασκευή των αρχείων από διαφορετικούς προγραμματιστές που χρησιμοποιούν διαφορετικές γλώσσες προγραμματισμού σε διαφορετικά λειτουργικά συστήματα (όπως Linux, Windows, MacOS). Το πρόβλημα είναι ιδιαίτερα έντονο όταν τα δεδομένα αποθηκεύονται σε δυαδική (binary) μορφή και όχι σε αλφαριθμητική (alphanumeric, text) μορφή.
- Σε πολλές περιπτώσεις η εφαρμογή εξυπηρετεί πολλούς χρήστες ταυτόχρονα. Η διασφάλιση της προστασίας και της ακεραιότητας των δεδομένων κατά την ταυτόχρονη αναζήτηση, εισαγωγή και διαγραφή στοιχείων είναι αρκετά περίπλοκη. Επίσης, σε συγκεκριμένες λειτουργίες πρέπει να διασφαλίζεται η ατομικότητα της εκτέλεσης. Με τον όρο "ατομικότητα" εννοούμε ότι ένα σύνολο λειτουργιών είτε θα εκτελεσθεί συνολικά, είτε θα απορριφθεί συνολικά. Χαρακτηριστικό παράδειγμα αποτελεί η ανάληψη χρημάτων από μία τραπεζική αυτόματη ταμειακή μηχανή (ATM). Εδώ διακρίνουμε δύο ξεχωριστές λειτουργίες: α) ανάληψη ενός ποσού και β) ενημέρωση του λογαριασμού με το νέο υπόλοιπο. Αν αποτύχει μία από τις δύο λειτουργίες, τότε θα αποτύχει και το σύνολό τους. Σε διαφορετική περίπτωση θα δημιουργηθεί σοβαρό πρόβλημα ακεραιότητας δεδομένων καθώς

τα δεδομένα του συστήματος θα είναι λανθασμένα (π.χ., ενώ έγινε η ανάληψη δεν αφαιρέθηκε το ποσό από το υπόλοιπο του λογαριασμού).

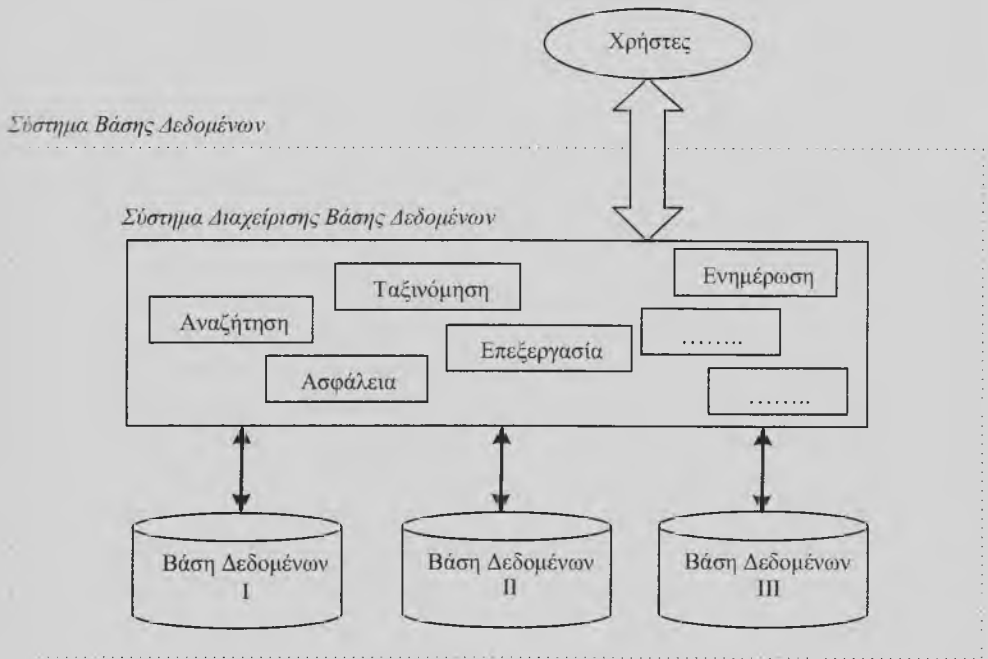
- Στη γενική περίπτωση, κάθε χρήστης της εφαρμογής έχει διαφορετικά δικαιώματα ως προς την προσπέλαση των δεδομένων. Για παράδειγμα, μία ομάδα χρηστών μπορεί να έχει μόνο δικαιώματα ανάγνωσης των δεδομένων, ενώ μία άλλη ομάδα χρηστών μπορεί να έχει δικαιώματα ανάγνωσης και εισαγωγής νέων δεδομένων αλλά όχι διαγραφής. Η εφαρμογή των κανόνων προστασίας των δεδομένων ώστε να αποφεύγονται δυσάρεστες καταστάσεις απώλειας δεδομένων (είτε κατά λάθος, είτε εσκεμμένα), στηριζόμενοι στα εργαλεία του λειτουργικού συστήματος, επιτυγχάνεται δύσκολα. Επιπλέον, είναι πολύ δύσκολο να ορίσουμε διαφορετικά δικαιώματα μέσα στο ίδιο αρχείο.
- Εφ' όσον η λογική της αναζήτησης στοιχείων από τα αρχεία δεδομένων υλοποιείται στον κώδικα της εφαρμογής, οι διαφορετικοί τύποι ερωτημάτων που μπορεί να δεχθεί η εφαρμογή είναι περιορισμένοι και πρέπει να υλοποιηθούν εκ των προτέρων. Αυτό αποτελεί αρκετά δεσμευτικό παράγοντα λαμβάνοντας υπόψη ότι είναι σχεδόν αδύνατο να προβλέψουμε όλα τα δυνατά ερωτήματα που μπορεί να επιθυμεί να θέσει ένας χρήστης. Η υποστήριξη νέων ερωτημάτων απαιτεί αλλαγή στον κώδικα της εφαρμογής, κάτι που γενικά δεν είναι επιθυμητό διότι οδηγεί σε νέες εκδόσεις (versions) του λογισμικού.

1.1 Συστήματα Βάσεων Δεδομένων

Οι περιορισμοί που αναφέρθηκαν οφείλονται κυρίως στο ότι η διαχείριση των δεδομένων είναι ενσωματωμένη στον κώδικα της εφαρμογής, εισάγοντας σημαντικές δυσκολίες στην επεξεργασία και στην αλλαγή της δομής των δεδομένων. Η προσέγγιση αυτή διαχωρίζει τα δεδομένα από τις μεθόδους επεξεργασίας, προσφέρει ευελιξία και επιπλέον αποδεσμεύει τον προγραμματιστή από την ανάγκη συγχρονισμού των δεδομένων και την υλοποίηση της λογικής προσπέλασης των δεδομένων. Η προσέγγιση αυτή στηρίζεται σε ένα **Σύστημα Βάσης Δεδομένων** (database system) το οποίο αναλαμβάνει αποκλειστικά την αποθήκευση, προστασία και επεξεργασία των δεδομένων, παρέχοντας εξελιγμένους μηχανισμούς πρόσβασης, δικαιωμάτων και ενημέρωσης των δεδομένων. Ένα Σύστημα Βάσης Δεδομένων αποτελείται από τα εξής:

- Τη **Βάση Δεδομένων** (database), η οποία αποτελεί μία συλλογή στοιχείων σχετικών μεταξύ τους, που είναι δομημένα και καταχωρισμένα με κατάλληλο τρόπο, και

- Το **Σύστημα Διαχείρισης (ή Διοίκησης) Βάσης Δεδομένων** (database management system), το οποίο είναι ένα λογισμικό σύστημα που υλοποιεί όλες τις λειτουργίες που πρέπει να υποστηριχθούν, όπως αναζήτηση, εισαγωγή, διαγραφή, συγχρονισμός προσπελάσεων, προστασία και πολλές ακόμη οι οποίες θα αναλυθούν σε επόμενα κεφάλαια. Σημειώνεται ότι το Σύστημα Διαχείρισης Βάσης Δεδομένων (ΣΔΒΔ) είναι σε θέση να διαχειρίζεται πολλές Βάσεις Δεδομένων (ΒΔ) ταυτόχρονα, αναλόγως με τις απαιτήσεις των χρηστών.



Σχήμα 1.1: Βάση Δεδομένων και Σύστημα Διαχείρισης.

Στο Σχήμα 1.1 παρουσιάζεται η σχέση μεταξύ της ΒΔ και του ΣΔΒΔ. Παρατηρούμε ότι οι χρήστες δεν έχουν άμεση πρόσβαση στα δεδομένα. Η προσπέλαση των δεδομένων επιτυγχάνεται μέσω των μηχανισμών που έχουν υλοποιηθεί στο ΣΔΒΔ. Με τον τρόπο αυτόν, ο έλεγχος των δεδομένων ανήκει στο ΣΔΒΔ και μόνο αυτό είναι υπεύθυνο για τη διαχείριση και την προστασία τους.

Ας προσέξουμε αναλυτικά τα πλεονεκτήματα που προσφέρει η δομή αυτή σε σχέση τόσο με τις δυνατότητες διαχείρισης των δεδομένων όσο και με την υποστήριξη των τελικών χρηστών (end-users).

- **Περιγραφή Δεδομένων.** Το πρώτο βασικό πλεονέκτημα είναι ότι το Σύστημα της Βάσης Δεδομένων εκτός από τα δεδομένα περιέχει και βοηθητικές πληροφορίες για την περιγραφή τους. Το χαρακτηριστικό αυτό επιτρέπει τη μεταβολή της δομής και της οργάνωσης των δεδομένων σύμφωνα με τις απαιτήσεις των χρηστών. Σημειώνεται ότι ο προγραμματιστής δεν χρειάζεται να παρέμβει, διότι όλοι οι μηχανισμοί διαχείρισης των δεδομένων είναι υλοποιημένοι στο ΣΔΒΔ, που επεμβαίνει απ' ευθείας στην αποθηκευμένη πληροφορία. Στην υλοποίηση με αρχεία, οι πληροφορίες για τη δομή και την οργάνωση των δεδομένων βρίσκονται στον κώδικα της εφαρμογής με αποτέλεσμα η αλλαγή τους να απαιτεί σημαντική εργασία από την πλευρά του προγραμματιστή.
- **Ανεξαρτησία Δεδομένων και Λειτουργιών.** Τα δεδομένα διαχωρίζονται από τις λειτουργίες που μπορούν να εφαρμοσθούν σε αυτά, σε αντίθεση με την προσέγγιση της υλοποίησης με αρχεία. Έτσι προσφέρεται ευελιξία, διότι η μεταβολή της δομής των δεδομένων δεν απαιτεί τη μεταβολή των προγραμμάτων εφαρμογής. Επίσης, η υποστήριξη νέων λειτουργιών επί των δεδομένων πραγματοποιείται ευκολότερα, καθώς επιβάλλονται μόνο προσθήκες στις ήδη υπάρχουσες λειτουργίες. Ο διαχωρισμός μεταξύ δεδομένων και λειτουργιών διευκολύνει, ακόμη, την αποθήκευση των δεδομένων σε διαφορετικές διαμορφώσεις (format) και διαφορετικά μέσα αποθήκευσης, χωρίς να απαιτείται αλλαγή των προγραμμάτων εφαρμογής. Επιπλέον, διευκολύνεται η ανάπτυξη εφαρμογών σε διαφορετικές γλώσσες προγραμματισμού, καθώς αυτό επιτυγχάνεται μέσω του ΣΔΒΔ που διαχειρίζεται όλα τα δεδομένα. Απεναντίας, στην υλοποίηση με αρχεία, μεταβολές στον τρόπο αποθήκευσης των δεδομένων συνεπάγονται τη μεταβολή του κώδικα της εφαρμογής.
- **Αποδοτικότερη Διαχείριση Δεδομένων.** Η υλοποίηση όλων των λειτουργιών επί των δεδομένων βρίσκεται μέσα στο ΣΔΒΔ. Οι κατασκευαστές ΣΔΒΔ φροντίζουν ώστε ο κώδικας να είναι όσο το δυνατόν αποδοτικότερος και η επεξεργασία των ερωτημάτων να γίνεται με όσο πραγματοποιείται με όσο δυνατόν καλύτερο τρόπο. Επομένως, ο προγραμματιστής εφαρμογών αποδεδεσμεύεται από τη μελέτη, υλοποίηση και εφαρμογή αποδοτικών αλγορίθμων και μεθόδων προσπέλασης και επικεντρώνεται στην υλοποίηση λειτουργιών που αφορούν μόνο στην εκάστοτε εφαρμογή και όχι στα αποθηκευμένα δεδομένα. Η προσπέλαση των δεδομένων και η μεταφορά αυτών στο χρήστη ή την εφαρμογή είναι αρμοδιότητα του ΣΔΒΔ.

- **Προστασία Δεδομένων και Δικαιώματα Χρηστών.** Το ΣΔΒΔ υλοποιεί μηχανισμούς προστασίας δεδομένων με στόχο την αποφυγή διαγραφής ή ενημέρωσης των δεδομένων χωρίς την απαραίτητη δικαιοδοσία. Εκτός από τον παραδοσιακό τρόπο κλειδώματος με χρήση ονόματος χρήστη (username) και κωδικού πρόσβασης (password), το ΣΔΒΔ επιτρέπει την απόδοση συγκεκριμένων δικαιωμάτων σε συγκεκριμένους χρήστες του συστήματος. Για παράδειγμα, σε ένα χρήστη μπορεί να επιτραπεί η ανάγνωση ενός συνόλου δεδομένων αλλά να αποτραπεί η διαγραφή ή η ενημέρωσή τους, για λόγους ασφαλείας. Παρομοίως, η ενημέρωση και η εισαγωγή νέων στοιχείων στη ΒΔ μπορεί να επιτραπεί μόνο σε ένα μικρό σύνολο εξουσιοδοτημένων χρηστών. Επειδή η απόδοση των δικαιωμάτων γίνεται με δυναμικό και όχι με στατικό τρόπο, τα δικαιώματα ενός χρήστη μπορούν να μεταβάλλονται αναλόγως με τις ανάγκες από το διαχειριστή του συστήματος.
- **Μηχανισμοί Ταυτόχρονης Προσπέλασης.** Η δυνατότητα ταυτόχρονης υποστήριξης πολλών χρηστών είναι πολύ σημαντική στις σύγχρονες εφαρμογές. Αν πολλοί χρήστες έχουν δικαιώματα ενημέρωσης του ίδιου τμήματος των δεδομένων, τότε υπάρχει μεγάλη πιθανότητα δύο ή περισσότεροι χρήστες να προσπαθήσουν ταυτόχρονα να μεταβάλλουν τα δεδομένα. Αντιστοίχως, υπάρχει η περίπτωση ένας χρήστης να προσπαθήσει να μεταβάλλει κάποια δεδομένα που τη συγκεκριμένη χρονική στιγμή προσπελούνται για ανάγνωση από άλλους χρήστες. Σε τέτοιες περιπτώσεις απαιτείται ένας μηχανισμός ελέγχου και συγχρονισμού των διαφορετικών λειτουργιών ώστε να αποφευχθεί καταστροφή των δεδομένων. Συνήθως χρησιμοποιούνται μηχανισμοί κλειδώματος (locking) έτσι ώστε δεδομένα που είναι ήδη σε χρήση (όπως για ανάγνωση ή ενημέρωση) να μην επιτρέπεται να μεταβληθούν. Οι μηχανισμοί αυτοί προσφέρονται από το ΣΔΒΔ αποδεσμεύοντας τους τελικούς χρήστες και τους προγραμματιστές από το δύσκολο έργο του συγχρονισμού λειτουργιών.
- **Επεκτασιμότητα.** Τα σύγχρονα ΣΔΒΔ επιτρέπουν την ενσωμάτωση νέων μεθόδων επεξεργασίας, νέων τύπων δεδομένων και νέων δομών οργάνωσης δεδομένων. Η δυνατότητα αυτή προσφέρει πολλά πλεονεκτήματα, διότι το σύστημα μπορεί να προσαρμόζεται αναλόγως με τις απαιτήσεις των χρηστών. Το βασικό χαρακτηριστικό της δυνατότητας αυτής είναι ότι οι νέες τεχνικές που υλοποιούνται γίνονται μέρος του ΣΔΒΔ, κι έτσι δεν απαιτείται η υλοποίηση ξεχωριστών προγραμμάτων εφαρμογής για την υποστήριξή τους.

Από την προηγούμενη ανάλυση φαίνεται καθαρά ότι η χρήση ενός ΣΔΒΔ πλεονεκτεί ως προς την προσέγγιση με υλοποίηση των αρχείων. Δικαίως λοιπόν τα ΣΔΒΔ έχουν υιοθετηθεί από μεγάλες εταιρείες και οργανισμούς για την αποθήκευση των δεδομένων τους. Επίσης, σήμερα κάθε μεγάλη εφαρμογή συνοδεύεται από υποστήριξη της Βάσης Δεδομένων για γρήγορη προσπέλαση και προστασία των δεδομένων.

1.2 Κλασικές και Σύγχρονες Εφαρμογές

Τα πρώτα ολοκληρωμένα Συστήματα Βάσεων Δεδομένων εμφανίσθηκαν στο τέλος της δεκαετίας του '60 και υποστήριζαν εφαρμογές που απαιτούσαν τη διαχείριση πολλών μικρών τμημάτων πληροφορίας και την επεξεργασία πολλών ερωτημάτων ή ενημερώσεων. Το βασικό χαρακτηριστικό τους είναι η άμεση ταυτόχρονη διαχείριση πολλών συναλλαγών. Για το λόγο αυτό οι εφαρμογές αυτές καλούνται εφαρμογές Άμεσης Επεξεργασίας Συναλλαγών (On-Line Transaction Processing, OLTP). Χαρακτηριστικά παραδείγματα τέτοιων εφαρμογών είναι: α) κρατήσεις θέσεων σε αεροπορικές εταιρείες, β) εφαρμογές τραπεζικών συναλλαγών και γ) διαχείριση δεδομένων σε μεγάλες εταιρείες ή οργανισμούς. Αναλυτικότερα:

- **Κρατήσεις Θέσεων.** Οι εφαρμογές αυτές υποστηρίζουν λειτουργίες, όπως κράτηση θέσης για μία συγκεκριμένη πτήση, αναζήτηση πληροφοριών για διαθέσιμες πτήσεις με βάση την αφετηρία και τον προορισμό, αναζήτηση πληροφοριών σχετικά με τις τιμές και τη διαθεσιμότητα των εισιτηρίων. Η λειτουργία της εφαρμογής επιτρέπει την απάντηση ερωτημάτων που αφορούν στην ώρα αναχώρησης και άφιξης συγκεκριμένων πτήσεων και την αποδοτική κράτηση θέσεων και έλεγχο της διαθεσιμότητας. Ακόμη και σήμερα οι εφαρμογές αυτές είναι πολύτιμες για ταξιδιωτικούς πράκτορες και αεροπορικές εταιρείες λόγω των πολλών διευκολύνσεων που παρέχουν.
- **Τραπεζικές Συναλλαγές.** Στις εφαρμογές αυτές η πληροφορία αποτελείται από ονόματα πελατών, διευθύνσεις, αριθμούς τραπεζικών λογαριασμών, υπόλοιπο λογαριασμών, δεδομένα που αφορούν σε δάνεια, πιστωτικές κάρτες και γενικά από ο,τιδήποτε σχετίζεται με τη λειτουργία μίας τράπεζας. Στόχος των εφαρμογών αυτών είναι η ταχύτερη εξυπηρέτηση των πελατών και η αποδοτικότερη λειτουργία της τράπεζας. Η εφαρμογή πρέπει να υποστηρίζει ταυτόχρονες προσπελάσεις στα δεδομένα από πολλούς χρήστες.
- **Διαχείριση Εταιρικών Δεδομένων.** Η καλή οργάνωση των δεδομένων μίας μεγάλης εταιρείας συμβάλει αποφασιστικά στην αποτελεσματική και αποδοτική λειτουργία της, διότι διευκολύνονται και επιταχύνονται οι εσωτερικές

διεργασίες της εταιρείας, οι οποίες πολλές φορές είναι πολύπλοκες και χρονοβόρες. Η μεθοδική οργάνωση των δεδομένων επιδρά σε όλα τα τμήματα της εταιρείας (όπως τμήμα πωλήσεων, τμήμα ανάπτυξης, τμήμα ανθρώπινου δυναμικού, κ.λ.π.). Τα δεδομένα σχετίζονται με τα προσωπικά στοιχεία των υπαλλήλων, όπως όνομα, διεύθυνση, μισθός, επιδόματα αδειάς. Η δομημένη οργάνωση της πληροφορίας επιτρέπει τη γρήγορη αναζήτηση στοιχείων που σχετίζονται με έναν υπάλληλο και την εύκολη ενημέρωση των στοιχείων σε περίπτωση μεταβολών (π.χ., αλλαγή θέσης ενός υπαλλήλου).

Οι σύγχρονες εφαρμογές απαιτούν την υποστήριξη νέων τύπων δεδομένων και νέων μεθόδων επεξεργασίας, ενώ ταυτόχρονα σε πολλές από αυτές η ταχύτητα επεξεργασίας είναι κρίσιμη. Στόχος των σύγχρονων ΣΔΒΔ είναι η αποτελεσματική και αποδοτική υποστήριξη των εφαρμογών αυτών. Οι ιδιαιτερότητες που εμφανίζουν οι σύγχρονες εφαρμογές σε σχέση με τις παραδοσιακές εφαρμογές Βάσεων Δεδομένων αγγίζουν σχεδόν όλα τα υποσυστήματα ενός ΣΔΒΔ. Η αδυναμία των παραδοσιακών Συστημάτων Βάσεων Δεδομένων να καλύψουν τις σύγχρονες ανάγκες οδήγησε τους ερευνητές στην αναζήτηση νέων μεθόδων επεξεργασίας. Στη συνέχεια παραθέτουμε μερικά παραδείγματα σύγχρονων εφαρμογών και τον τρόπο υποστήριξής τους από τα σύγχρονα Συστήματα Βάσεων Δεδομένων.

- **Πολυμεσικές Εφαρμογές.** Τα παλαιότερα Συστήματα Βάσεων Δεδομένων είχαν την ικανότητα να χειρίζονται μόνο αριθμητικά και αλφαριθμητικά δεδομένα (κείμενο). Σήμερα ωστόσο, ένα σημαντικό ποσοστό της πληροφορίας βρίσκεται σε μορφές όπως ήχος, εικόνα και video. Η υποστήριξη των τύπων αυτών απαιτεί ταχύτητα στην επεξεργασία και περισσότερο χώρο αποθήκευσης λόγω του μεγάλου όγκου των δεδομένων. Οι Βάσεις Δεδομένων πολυμέσων (multimedia databases) έχουν τη δυνατότητα χειρισμού των σύνθετων αυτών τύπων δεδομένων χρησιμοποιώντας βελτιστοποιημένες τεχνικές αναζήτησης. Έτσι, σήμερα έχουμε τη δυνατότητα να θέτουμε ερωτήσεις όπως: *"Να βρεθούν οι εικόνες που μοιάζουν περισσότερο με την εικόνα Χ"* ή *"Να βρεθούν τα μουσικά κομμάτια που περιέχουν το εξής μοτίβο ή κάποιο παρόμοιο"*, κάτι που θα φαινόταν εξαιρετικά πολύπλοκο στο παρελθόν.
- **Διαχείριση Γεωγραφικών Δεδομένων.** Εκτός από αριθμητικές και αλφαριθμητικές συνιστώσες, τα γεωγραφικά δεδομένα αποτελούνται από γεωμετρικά αντικείμενα (όπως σημεία, ευθύγραμμα τμήματα, πολύγωνα, περιοχές κ.λ.π.), που απαιτούν νέους τρόπους αποθήκευσης και επεξεργασίας λόγω της χωρικής (spatial) φύσης τους. Για παράδειγμα, για τα δεδομένα αυτά λόγω της δισδιάστατης (ή γενικότερα πολυδιάστατης) φύσης τους δεν

ισχύουν οι σχέσεις διάταξης (π.χ., στους ακεραίους και τους χαρακτήρες υπάρχει η έννοια του προηγούμενου-επόμενου, στους πραγματικούς αριθμούς η έννοια του μικρότερου-μεγαλύτερου κ.λ.π.). Ισχύουν όμως τοπολογικές σχέσεις (όπως αριστερά-δεξιά, βόρεια-νότια, εντός-εκτός, τομή, απόσταση κ.λ.π.). Έτσι, το ΣΔΒΔ, πιθανώς συνεργαζόμενο με ένα Σύστημα Γεωγραφικών Πληροφοριών (Geographical Information System, GIS), στοχεύοντας στην αποδοτική επεξεργασία των ερωτημάτων είναι εξοπλισμένο με εξειδικευμένες τεχνικές διαχείρισης, οι οποίες βασίζονται στην περιοχή της Υπολογιστικής Γεωμετρίας (Computational Geometry).

- **Αποθήκες Δεδομένων και Αναλυτική Επεξεργασία.** Μία άλλη σύγχρονη εφαρμογή που προέκυψε ως ανάγκη λόγω της ύπαρξης τεράστιου όγκου δεδομένων σε μεγάλες επιχειρήσεις και οργανισμούς είναι τα **Συστήματα Στήριξης Αποφάσεων** (decision support systems). Για τη λήψη αποφάσεων σε θέματα ανάλυσης αγοράς, οικονομικού σχεδιασμού, marketing κ.λ.π., ένας αναλυτής μίας επιχείρησης ενδιαφέρεται να υποβάλλει πολύπλοκα ερωτήματα περισσότερο σε συγκεντρωτικά δεδομένα παρά σε ατομικές συναλλαγές. Η υποβολή και η επεξεργασία τέτοιου τύπου ερωτημάτων είναι μία διαδικασία από δύσκολη μέχρι και αδύνατη, καθώς συχνά τα δεδομένα των μεγάλων φορέων είναι διασκορπισμένα σε πολλά ετερογενή συστήματα, που μπορεί να είναι και γεωγραφικά κατανομημένα. Στις περιπτώσεις αυτές, τα πρωτογενή δεδομένα με ειδικές διαδικασίες ομογενοποιούνται και αποθηκεύονται σε ειδικές βάσεις με συνοπτικό τρόπο χρησιμοποιώντας διαφορετικές οπτικές γωνίες και διαφορετικές κλίμακες. Αυτές οι ειδικές βάσεις λέγονται Αποθήκες Δεδομένων (data warehouses), ενώ η επεξεργασία τους έχει την ειδική ονομασία Άμεση Αναλυτική Επεξεργασία (On-Line Analytical Processing, OLAP) σε αντίθεση με τα παραδοσιακά Συστήματα Βάσεων Δεδομένων, όπου μιλούμε για Άμεση Επεξεργασία Συναλλαγών.
- **Εξόρυξη Δεδομένων.** Η Εξόρυξη Δεδομένων (data mining) είναι μία άλλη σύγχρονη περιοχή των Βάσεων Δεδομένων, που προέκυψε ως ανάγκη λόγω του τεράστιου όγκου δεδομένων (συχνά πολυμεσιικών), που συλλέγονται αυτοματοποιημένα από δορυφόρους, κάμερες παρακολούθησης, συσκευές μέτρησης αλλά ακόμη μπορεί να συλλέγονται με παραδοσιακό τρόπο από ένα Σύστημα Βάσεων Δεδομένων. Επειδή τέτοιους όγκους δεδομένων δεν μπορούμε να τους χειρισθούμε αποδοτικά με βάση τις συμβατικές μεθόδους επεξεργασίας, σε τέτοιες περιπτώσεις εφαρμόζουμε ειδικές αυτοματοποιημένες μεθόδους επεξεργασίας, οι οποίες αναλαμβάνουν να "σκάψουν" στα δεδομένα αναζητώντας πληροφορία χρήσιμη, προηγουμένως άγνωστη.