ΤΕΧΝΟΛΟΓΙΚΟ ΕΚΠΑΙΔΕΥΤΙΚΟ ΙΔΡΥΜΑ ΠΑΤΡΑΣ

Τμήμα Διοίκησης Επιχειρήσεων

Σημειώσεις του μαθήματος: «Βάσεις Δεδομένων και Ευφυή Πληροφοριακά Συστήματα»

ΕΞΑΜΗΝΟ: ΣΤ

Δρ. Κωνσταντίνος Χ. Γιωτόπουλος

Πάτρα, Δεκέμβριος 2010

Περιεχόμενα

ΠΕΡΙΕΧΟΜΕΝΑ ---------- 2

1 ΕΙΣΑΓΩΓΗ ΣΤΙΣ ΒΑΣΙΚΕΣ ΕΝΝΟΙΕΣ ΤΩΝ ΒΑΣΕΩΝ ΔΕΔΟΜΕΝΩΝ---------- 4

1.1 ΔΕΔΟΜΕΝΑ ΚΑΙ ΠΛΗΡΟΦΟΡΙΑ ---------- 4
1.2 ΒΑΣΗ ΔΕΔΟΜΕΝΩΝ---------- 4
1.3 ΙΣΤΟΡΙΚΗ ΑΝΑΔΡΟΜΗ---------- 5

2 ΣΥΣΤΗΜΑ ΔΙΑΧΕΙΡΙΣΗΣ ΒΑΣΕΩΝ ΔΕΔΟΜΕΝΩΝ ---------- 7

2.1 ΔΙΑΦΟΡΑ ΒΑΣΗΣ ΔΕΔΟΜΕΝΩΝ ΜΕ ΣΥΣΤΗΜΑ ΔΙΑΧΕΙΡΙΣΗΣ ΒΑΣΕΩΝ ΔΕΔΟΜΕΝΩΝ ---------- 7
2.2 ΔΟΜΗ ΕΝΟΣ DBMS ---------- 8
2.3 ΤΙ ΠΑΡΕΧΕΙ ΕΝΑ ΣΥΣΤΗΜΑ ΔΙΑΧΕΙΡΙΣΗΣ ΒΑΣΕΩΝ ΔΕΔΟΜΕΝΩΝ ---------- 9
2.4 ΤΙ ΠΑΡΕΧΕΙ Η ΒΑΣΗ ΔΕΔΟΜΕΝΩΝ ---------- 11

3 ΑΝΑΣΚΟΠΗΣΗ ΤΗΣ ΔΙΑΔΙΚΑΣΙΑΣ ΣΧΕΔΙΑΣΜΟΥ ΒΑΣΕΩΝ ΔΕΔΟΜΕΝΩΝ --------13

3.1 Ο ΡΟΛΟΣ ΤΩΝ ΠΛΗΡΟΦΟΡΙΑΚΩΝ ΣΥΣΤΗΜΑΤΩΝ ΣΤΟΥΣ ΟΡΓΑΝΙΣΜΟΥΣ ---------- 13
3.2 ΚΥΚΛΟΣ ΖΩΗΣ ΠΛΗΡΟΦΟΡΙΑΚΩΝ ΣΥΣΤΗΜΑΤΩΝ---------- 16
3.3 ΚΥΚΛΟΣ ΖΩΗΣ ΣΥΣΤΗΜΑΤΟΣ ΒΑΣΗΣ ΔΕΔΟΜΕΝΩΝ ---------- 17
3.4 ΔΙΑΔΙΚΑΣΙΑ ΣΧΕΔΙΑΣΜΟΥ ΤΗΣ ΒΑΣΗΣ ΔΕΔΟΜΕΝΩΝ ---------- 18
3.4.1 Φάση 1: Συλλογή και Ανάλυση Απαιτήσεων---------- 21
3.4.2 Φάση 2: Εννοιολογικός Σχεδιασμός Βάσεων Δεδομένων ---------- 22
3.4.3 Φάση 3: Επιλογή ενός ΣΔΒΔ ---------- 30
3.4.4 Φάση 4: Απεικόνιση στο Μοντέλο Δεδομένων ---------- 33
3.4.5 Φάση 5: Φυσικός Σχεδιασμός Βάσεων Δεδομένων---------- 33
3.4.6 Φάση 6: Υλοποίηση του Συστήματος Βάσης Δεδομένων ---------- 34
3.5 ΠΑΡΑΓΟΝΤΕΣ ΠΟΥ ΕΠΗΡΕΑΖΟΥΝ ΤΟ ΦΥΣΙΚΟ ΣΧΕΔΙΑΣΜΟ ΒΑΣΕΩΝ ΔΕΔΟΜΕΝΩΝ ---------- 36

4 ΈΝΝΟΙΕΣ ΕΠΕΞΕΡΓΑΣΙΑΣ ΔΟΣΟΛΗΨΙΩΝ----------38

4.1 ΕΙΣΑΓΩΓΗ ΣΤΗΝ ΕΠΕΞΕΡΓΑΣΙΑ ΔΟΣΟΛΗΨΙΩΝ---------- 38
4.1.1 Μονοχρηστικά Συστήματα Έναντι Πολυχρηστικών Συστημάτων ---------- 38
4.1.2 Οι Πράξεις Ανάγνωσης και Εγγραφής μιας Δοσοληψίας---------- 40
4.1.3 Γιατί είναι απαραίτητος ο Έλεγχος Συνδρομικότητας ---------- 41
4.1.4 Γιατί είναι απαραίτητη η Δυνατότητα Ανάκαμψης ---------- 43
4.2 ΕΝΝΟΙΕΣ ΔΟΣΟΛΗΨΙΩΝ ΚΑΙ ΣΥΣΤΗΜΑΤΟΣ---------- 44
4.2.1 Καταστάσεις Δοσοληψίας και Πρόσθετες Πράξεις---------- 45
4.2.2 Ημερολόγιο Συστήματος (System Log ή System Journal)---------- 46
4.2.3 Σημείο Επικύρωσης μιας Δοσοληψίας ---------- 47
4.2.4 Σημεία Ελέγχου στο Ημερολόγιο Συστήματος ---------- 48
4.3 ΕΠΙΘΥΜΗΤΕΣ ΙΔΙΟΤΗΤΕΣ ΤΩΝ ΔΟΣΟΛΗΨΙΩΝ ---------- 49
4.4 ΧΡΟΝΟΠΡΟΓΡΑΜΜΑΤΑ ΚΑΙ ΔΥΝΑΤΟΤΗΤΑ ΑΝΑΚΑΜΨΗΣ---------- 50
4.4.1 Χρονοπρογράμματα (Χρονικά) Δοσοληψιών ---------- 50
4.4.2 Χαρακτηρισμός των Χρονοπρογραμμάτων με Βάση τη Δυνατότητα Ανάκαμψης---------- 52
4.5 ΣΕΙΡΙΟΠΟΙΗΣΙΜΟΤΗΤΑ ΤΩΝ ΧΡΟΝΟΠΡΟΓΡΑΜΜΑΤΩΝ---------- 53
4.5.1 Σειριακά, μη Σειριακά και Σειριοποιήσιμα Βάσει Αντιθέσεων Χρονοπρογράμματα ------ 54
4.5.2 Έλεγχος ενός Χρονοπρογράμματος για Σειριοποιησιμότητα βάσει Συγκρούσεων--------- 57
4.5.3 Χρήσεις της Σειριοποιησιμότητας---------- 61

5 ΑΝΑΓΚΗ ΓΙΑ ΕΥΦΥΗ ΠΛΗΡΟΦΟΡΙΑΚΑ ΣΥΣΤΗΜΑΤΑ----------63

5.1 ΕΥΦΥΗ ΣΥΣΤΗΜΑΤΑ ΠΡΟΣΟΜΟΙΩΣΗΣ ---------- 64
5.2 ΕΥΦΥΗ ΣΥΣΤΗΜΑΤΑ ΠΛΗΡΟΦΟΡΙΩΝ ---------- 65
5.3 ΕΥΦΥΗ ΣΥΣΤΗΜΑΤΑ ΥΠΟΣΤΗΡΙΞΗΣ---------- 66
5.4 ΕΥΦΥΗ ROBOTS ---------- 66
5.5 ΑΝΑΓΚΗ ΓΙΑ ΥΠΟΔΟΜΗ---------- 67
5.5.1 Ευκολία Χρήσης ---------- 67
5.5.2 Ευέλικτες και Προσαρμόσιμες Υπηρεσίες Υποδομής---------- 68
5.5.3 Εργαλεία Ανάπτυξης και Υποστήριξης---------- 69
5.6 ΤΕΧΝΟΛΟΓΙΕΣ ΑΙΧΜΗΣ ΚΑΙ ΤΟΜΕΙΣ ΈΡΕΥΝΑΣ ---------- 69

6 ΕΥΦΥΕΙΣ ΔΙΑΜΕΣΟΛΑΒΗΤΕΣ ----------73

6.1 ΤΙ ΕΙΝΑΙ ΕΝΑΣ ΕΥΦΥΗΣ ΔΙΑΜΕΣΟΛΑΒΗΤΗΣ ----------73

6.2 ΑΡΧΙΤΕΚΤΟΝΙΚΕΣ----------75

6.2.1 Αρχιτεκτονικές Νοημοσύνης----------75

6.2.2 Αρχιτεκτονικές Αντίδρασης----------77

6.2.3 Υβριδικές Αρχιτεκτονικές ----------79

ΒΙΒΛΙΟΓΡΑΦΙΑ ----------81

1 Εισαγωγή στις βασικές έννοιες των Βάσεων Δεδομένων

1.1 Δεδομένα και Πληροφορία

Πριν προχωρήσουμε στην έννοια της Βάσης Δεδομένων πρέπει να διακρίνουμε μεταξύ των εννοιών δεδομένα (data) και πληροφορία (information). Από τα πρώτα χρόνια της χρήσης των Ηλεκτρονικών Υπολογιστών οι έννοιες δεδομένα και πληροφορία συγχέονταν μεταξύ τους.

Έτσι τα δεδομένα μπορούν να είναι λέξεις π.χ. {"Νίκος", "Μιχάλης", "Μαρία", "Θάλασσα", "Αυτοκίνητο"}, αριθμοί, π.χ. {1, 2, 3, 5, 78}, συμβολοσειρές (strings) π.χ. {"Κώστας", "5621", "ΤΡ–882", "6&5 #1", "<>+–/*"}, ή σύμβολα που έχουν νόημα σε συγκεκριμένο πλαίσιο αναφοράς. Ένα τέτοιο παράδειγμα, δεδομένων είναι ψηφιοποιημένες εικόνες με κάποια καθορισμένη τυποποίηση. Αυτές οι εικόνες θα έχουν νόημα μόνο με τη χρήση ενός συγκεκριμένου προγράμματος απεικόνισης, δηλαδή μέσα στο πλαίσιο το προγράμματος απεικόνισης, ενώ έξω από αυτό είναι ακατανόητες συλλογές συμβόλων.

Έτσι όπως ορίσαμε τα δεδομένα είναι φανερό ότι είναι κατάλληλα για αποθήκευση σε ηλεκτρονικό υπολογιστή. Οι άνθρωποι, όμως, χρησιμοποιούσαν δεδομένα –πολύ πριν από την ανακάλυψη των ηλεκτρονικών υπολογιστών– στα πλαίσια ενός πεδίου αναφοράς για την εξαγωγή συμπερασμάτων ή την λήψη αποφάσεων. Για παράδειγμα, από τα δεδομένα {85, 210, 515} και τα δεδομένα {"Κόρινθος", "Πάτρα", "Θεσσαλονίκη"}, αλλά και από τη γνώση (πεδίο αναφοράς) ότι τα πρώτα δεδομένα περιγράφουν αποστάσεις από την Αθήνα για τις πόλεις που ορίζονται στα δεύτερα δεδομένα, κάποιος μπορεί να καταλάβει ότι «η Πάτρα είναι πιο μακριά από την Αθήνα από ότι η Κόρινθος, αλλά πολύ κοντύτερα στην Αθήνα από ότι η Θεσσαλονίκη». Αυτό το συμπέρασμα είναι πληροφορία.

Συνοψίζοντας μπορούμε να πούμε ότι τα δεδομένα είναι κωδικοποιημένη αναπαράσταση πληροφορίας, ενώ η πληροφορία εξάγεται από τα δεδομένα με βάση το πεδίο αναφοράς των δεδομένων ή συγκεκριμένη γνώση για αυτά.

1.2 Βάση Δεδομένων

Ένα βασικό χαρακτηριστικό των δεδομένων, έτσι όπως τα παρουσιάσαμε είναι ότι είναι κατάλληλα για να αποθηκευτούν σε ηλεκτρονικό υπολογιστή. Αυτό οδηγεί στο επόμενο βήμα που είναι η δημιουργία και χρήση Βάσεων Δεδομένων (databases).

Η σχετικότητα των δεδομένων είναι σημαντική για τον ορισμό μιας Βάσης Δεδομένων. Δεδομένα που δε σχετίζονται μεταξύ τους και απλά έχουν αποθηκευτεί σε έναν ηλεκτρονικό υπολογιστή δεν αποτελούν μια Βάση Δεδομένων. Μια Βάση Δεδομένων πρέπει να αντικατοπτρίζει ένα περιβάλλον του πραγματικού κόσμου. Τα δεδομένα που αποθηκεύονται στη Βάση Δεδομένων πρέπει να έχουν λογική συνέχεια και νόημα. Επίσης οι λειτουργίες που παρέχει η Βάση Δεδομένων είναι σημαντικές για τον ορισμό μιας Βάσης Δεδομένων. Συλλογές δεδομένων χωρίς τη δυνατότητα ενός αυτόματου, κοινού και κεντρικού τρόπου χειρισμού των δεδομένων αυτών, δεν αποτελούν Βάση Δεδομένων. Απλά είναι δεδομένα αποθηκευμένα στον ηλεκτρονικό υπολογιστή.

Η Βάση Δεδομένων έχει ένα σκοπό. Υλοποιείται για να απεικονίσει στον υπολογιστή ένα πρόβλημα και να διευκολύνει τη λύση του. Αυτό συνήθως σημαίνει ότι τα δεδομένα που χειρίζεται δεν είναι στατικά αλλά δυναμικά, δηλαδή αλλάζουν

συνεχώς. Η Βάση Δεδομένων περιέχει τα δεδομένα που περιγράφουν τη συγκεκριμένη χρονική στιγμή το «πρόβλημα», μπορεί όμως να περιέχει και δεδομένα από την ιστορία του προβλήματος.

Παράδειγμα 1.1

Ας δούμε μία αρκετά απλή Βάση Δεδομένων: Έστω ότι θέλουμε να έχουμε στον ηλεκτρονικό υπολογιστή τη θερμοκρασία 5 μεγάλων ευρωπαϊκών πόλεων κάθε μέρα στις 12 το μεσημέρι. Θα πρέπει λοιπόν να αποθηκεύσουμε στον υπολογιστή δεδομένα που θα είναι (για λόγους απλότητας) της μορφής {“Αθήνα”, 24, “15–1–98”}, {“Ρώμη”, 29, “12–1–98”} κτλ. Προσέξτε ότι χρειαζόμαστε και την πληροφορία για την ημερομηνία. Επίσης, όπως ορίσαμε παραπάνω, θα πρέπει για να έχουμε Βάση Δεδομένων να έχουμε και τη δυνατότητα χειρισμού αυτών των δεδομένων. Έτσι θα πρέπει να μπορούμε να αντλήσουμε πληροφορίες όπως «ποια είναι η θερμοκρασία που είχε η Αθήνα στις 25 Μαρτίου του 1998;» ή «που έκανε πιο πολύ ζέστη, στην Αθήνα ή στην Ρώμη στις 31 Αυγούστου 1997;». Τέλος θα πρέπει να μπορούμε να χειριστούμε τα δεδομένα, δηλαδή να εισάγουμε νέα δεδομένα στη Βάση Δεδομένων (όπως είναι οι σημερινές θερμοκρασίες), ή να σβήσουμε κάποια παλαιά δεδομένα, που πιθανόν δεν χρειαζόμαστε άλλο.

Βέβαια για να μπορεί η Βάση Δεδομένων να μας παρέχει σωστά δεδομένα (που θα μας οδηγούν σε σωστές πληροφορίες) πρέπει να αναλάβουμε την καθημερινή εισαγωγή δεδομένων σε αυτή, αλλά και να ελέγχουμε την ορθότητα των δεδομένων που εισάγουμε.

Η Βάση Δεδομένων που περιγράφουμε στο παράδειγμα 1.1 έχει όλα τα στοιχεία που καθορίζουν μια Βάση Δεδομένων. Είναι μια πολύ απλή Βάση Δεδομένων με μετεωρολογικές πληροφορίες. Για να μπορεί όμως να αναπαριστά δυναμικά τον πραγματικό κόσμο θα πρέπει κάποιοι άνθρωποι να είναι επιφορτισμένοι με το καθήκον να εισάγουν δεδομένα σε αυτή. Οι άνθρωποι αυτοί χρησιμοποιούν τη Βάση Δεδομένων και καλούνται χρήστες (users). Οι λειτουργίες που επιτελούν σχετίζονται όχι μόνο με την εισαγωγή δεδομένων, αλλά γενικότερα φροντίζουν για την εύρυθμη λειτουργία της Βάσης Δεδομένων. Αυτή η διαδικασία ονομάζεται συντήρηση της Βάσης Δεδομένων. Χρήστες είναι επίσης και όσοι ανατρέχουν στη Βάση Δεδομένων για πληροφορίες, π.χ. για να μάθουν τι θερμοκρασία είχε η Αθήνα σήμερα.

1.3 Ιστορική Αναδρομή

Οι Βάσεις Δεδομένων δεν προϋποθέτουν την ύπαρξη ηλεκτρονικού υπολογιστή. Οι πρώτες Βάσεις Δεδομένων δεν βασίζονταν σε ηλεκτρονικό υπολογιστή, αλλά σε χαρτί και οργανώνονταν με φακέλους ή καρτέλες. Αρκετές υπηρεσίες δυστυχώς λειτουργούν με παρόμοιο τρόπο ακόμα και σήμερα. Η Βάση Δεδομένων του παραδείγματος θα μπορούσε κάλλιστα να χρησιμοποιεί καρτέλες της μορφής του παραδείγματος 1.2.

Παράδειγμα 1.2

Πάτρα – Δεκέμβριος 2009									
1	17 °C	2	18 °C	3	19 °C	4	17 °C	5	15 °C
6	12 °C	7	13 °C	8	15 °C	9	10 °C	10	9 °C
11	17 °C	12	8 °C	13	11 °C	14	9 °C	15	7 °C
16	12 °C	17	7 °C	18	8 °C	19	7 °C	20	10 °C
21	7 °C	22	9 °C	23	7 °C	24	10 °C	25	7 °C
26	5 °C	27	7 °C	28	10 °C	29	7 °C	30	10 °C
31	4 °C								

Μια Βάση Δεδομένων που θα βασιζόταν σε τέτοιες καρτέλες θα μπορούσε να παρέχει τις ίδιες πληροφορίες με μια Βάση Δεδομένων σε ηλεκτρονικό υπολογιστή, όμως ο χειρισμός της θα ήταν πολύ πιο δύσκολος. Έτσι για παράδειγμα θα μπορούσαμε εύκολα να ανακτήσουμε την πληροφορία «θερμοκρασία στην Πάτρα στις 12 το μεσημέρι στις 16 Δεκεμβρίου 2009» τόσο από τον ηλεκτρονικό υπολογιστή, όσο και από τη Βάση Δεδομένων με τις καρτέλες. Στον υπολογιστή όμως, πληροφορίες όπως «ποιες ημέρες στην Αθήνα έκανε περισσότερη ζέστη από τη Πάτρα;» ή «ποια πόλη ήταν η πιο ζεστή για κάθε ημέρα του χρόνου;» μπορούν να υλοποιηθούν εύκολα σε αντίθεση με τις καρτέλες. Σκεφτείτε πόσοι άνθρωποι θα έπρεπε να δουλέψουν με τις καρτέλες για να ετοιμασθούν οι απαντήσεις για τις δύο παραπάνω ερωτήσεις εάν είχαμε δεδομένα για τα δέκα προηγούμενα έτη! Αυτή η ανάγκη για αξιοποίηση των δυνατοτήτων του ηλεκτρονικού υπολογιστή οδήγησε στην δημιουργία των πρώτων Βάσεων Δεδομένων. Οι βάσεις αυτές απλά υλοποιούσαν στον ηλεκτρονικό υπολογιστή σχήματα όπως το παραπάνω με τις καρτέλες και εκμεταλλεύονταν τις δυνατότητες του υπολογιστή για ταχύτατη και μαζική επεξεργασία δεδομένων και εξαγωγή απαντήσεων σε ερωτήσεις, δηλαδή εξαγωγή πληροφορίας.

Οι Βάσεις Δεδομένων εξελίχτηκαν πολύ τα χρόνια που πέρασαν. Από μικρές Βάσεις Δεδομένων που χειρίζονταν μικρούς αριθμούς δεδομένων παρουσιάστηκαν Βάσεις Δεδομένων που χειρίζονται τεράστιο όγκο πολύπλοκων δεδομένων. Σήμερα μιλάμε για Βάσεις Δεδομένων της τάξης των terabytes[1] με δεδομένα τα οποία περιέχουν εικόνες, ή βίντεο και Συστήματα Διαχείρισης Βάσεων Δεδομένων με δυνατότητες ενσωμάτωσης λειτουργιών για καλύτερο χειρισμό των δεδομένων, καθώς και για τρόπους εναλλακτικής οργάνωσης των δεδομένων, όπως οι αντικειμενοστραφείς Βάσεις Δεδομένων.

[1] 1terabyte = 1012 bytes ή 1.000.000.000.000 bytes

2 Σύστημα Διαχείρισης Βάσεων Δεδομένων

Η εξέλιξη των Βάσεων Δεδομένων και οι ανάγκες για δημιουργία όλο και περισσότερων Βάσεων Δεδομένων, οδήγησαν στη δημιουργία των Συστημάτων Διαχείρισης Βάσεων Δεδομένων (Database Management Systems ή DBMS). Το Σύστημα Διαχείρισης Βάσεων Δεδομένων είναι ένα εργαλείο το οποίο διευκολύνει τους χρήστες να εργάζονται με Βάσεις Δεδομένων. Με τη χρήση του Συστήματος Διαχείρισης Βάσεων Δεδομένων οι χρήστες μπορούν να κατασκευάσουν και να χρησιμοποιήσουν Βάσεις Δεδομένων. Ακολουθεί ο ορισμός του Συστήματος Διαχείρισης Βάσεων Δεδομένων.

Ένα Σύστημα Διαχείρισης Βάσεων Δεδομένων κατά κανόνα «φιλοξενεί» πολλές Βάσεις Δεδομένων που έχουν κατασκευαστεί από διαφορετικούς χρήστες. Οι δυνατότητες που παρέχει ένα Σύστημα Διαχείρισης Βάσεων Δεδομένων στους χρήστες συνοψίζονται στις παρακάτω:

- Ορισμός της Βάσης Δεδομένων
- Κατασκευή της Βάσης Δεδομένων
- Διαγραφή της Βάσης Δεδομένων
- Χρήση της Βάσης Δεδομένων

Στον ορισμό της Βάσης Δεδομένων ο χρήστης μπορεί να καθορίσει το μοντέλο της Βάσης Δεδομένων, να ορίσει τους τύπους δεδομένων που θα χρησιμοποιήσει και να ελέγξει τη Βάση Δεδομένων χωρίς να προχωρήσει σε κατασκευή της. Στην κατασκευή της Βάσης Δεδομένων οι τύποι των δεδομένων και τα δεδομένα αποθηκεύονται στο υλικό (hardware) του ηλεκτρονικού υπολογιστή με διαδικασίες που ελέγχονται από το Σύστημα Διαχείρισης Βάσεων Δεδομένων και δεν απασχολούν τον χρήστη. Στη διαγραφή της Βάσης Δεδομένων ο χρήστης αποφασίζει τον τερματισμό μιας Βάσης Δεδομένων και την απομάκρυνση των δεδομένων από το υλικό. Τέλος, κατά τη χρήση της Βάσης Δεδομένων ο χρήστης, είτε χειρίζεται τα δεδομένα (εισάγει νέα δεδομένα, τροποποιεί δεδομένα, ή διαγράφει δεδομένα), είτε υποβάλει ερωτήσεις στη Βάση Δεδομένων με στόχο την εξαγωγή πληροφοριών.

2.1 Διαφορά Βάσης Δεδομένων με Σύστημα Διαχείρισης Βάσεων Δεδομένων

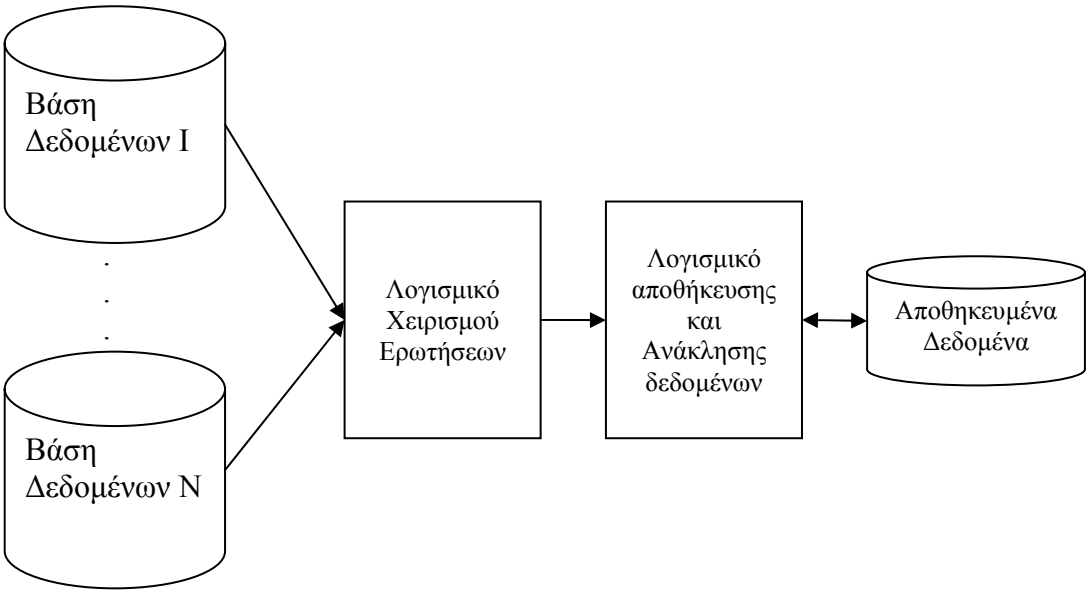
Σε αυτό το σημείο πρέπει να είναι ξεκάθαρη σε εσάς η διαφορά μεταξύ ενός Συστήματος Διαχείρισης Βάσεων Δεδομένων και μίας Βάσης Δεδομένων. Το Σύστημα Διαχείρισης Βάσεων Δεδομένων είναι λογισμικό το οποίο διευκολύνει τους χρήστες να υλοποιήσουν Βάσεις Δεδομένων. Αντίθετα η Βάση Δεδομένων υλοποιείται με τη βοήθεια ενός Συστήματος Διαχείρισης Βάσεων Δεδομένων που χειρίζεται δεδομένα τα οποία αποθηκεύονται στο υλικό του υπολογιστή. Η όλη διαδικασία χειρισμού και αποθήκευσης των δεδομένων διευκολύνεται από το Σύστημα Διαχείρισης Βάσεων Δεδομένων, το οποίο αναλαμβάνει τη μετατροπή των εντολών του χρήστη σε εντολές προς τον υπολογιστή και λειτουργίες που σχετίζονται με το χειρισμό των δεδομένων στο υλικό του υπολογιστή.

Ο χρήστης μιας Βάσης Δεδομένων επιτελεί λειτουργίες που μπορούν να ομαδοποιηθούν στις παρακάτω: εισαγωγή δεδομένων, διαγραφή δεδομένων, αλλαγή δεδομένων και ανάκληση δεδομένων. Όλες οι παραπάνω λειτουργίες προϋποθέτουν

επικοινωνία με το υλικό του υπολογιστή (π.χ. το μαγνητικό δίσκο του υπολογιστή). Αυτή η επικοινωνία γίνεται μέσω του λογισμικού του Συστήματος Διαχείρισης Βάσεων Δεδομένων. Η επικοινωνία αυτή συνήθως δεν είναι ορατή από τον χρήστη της Βάσης Δεδομένων, ο οποίος απλά βλέπει τα αποτελέσματα από τις λειτουργίες που ζήτησε να επιτελούνται. Αυτό είναι και το σημείο που κάνει αρκετούς χρήστες να συγχέουν τις έννοιες Βάση Δεδομένων και Σύστημα Διαχείρισης Βάσεων Δεδομένων.

2.2 Δομή ενός DBMS

Ένα Σύστημα Διαχείρισης Βάσεων Δεδομένων μπορεί να απεικονισθεί όπως στο σχήμα 1. Στο δεξιό μέρος του σχήματος 1 παρουσιάζεται το είδος (μέσο αποθήκευσης) του υλικού (hardware) του υπολογιστή στον οποίο αποθηκεύονται τα δεδομένα. Σε μικρές Βάσεις Δεδομένων το μέσο αυτό συνήθως είναι ο σκληρός δίσκος του υπολογιστή, ενώ σε μεγάλες Βάσεις Δεδομένων χρησιμοποιούνται πολλαπλά επίπεδα αποθήκευσης που ποικίλουν από σκληρούς δίσκους διαφορετικής ταχύτητας και χωρητικότητας, juke boxes από CDs, μαγνητικές ταινίες κτλ.



Σχήμα 1: Σύστημα Διαχείρισης Βάσεων Δεδομένων

Ο χειρισμός των δεδομένων στο υλικό του υπολογιστή γίνεται από το λογισμικό του Συστήματος Διαχείρισης Βάσεων Δεδομένων που στο σχήμα 1 ονομάζεται «Λογισμικό Αποθήκευσης και Ανάκλησης Δεδομένων». Αυτό το λογισμικό αναλαμβάνει να μεταφέρει όλες τις εντολές του χρήστη προς το υλικό. Το λογισμικό αυτό συνεργάζεται με το άλλο τμήμα του Συστήματος Διαχείρισης Βάσεων Δεδομένων που είναι το «Λογισμικό Χειρισμού Ερωτήσεων». Όλα τα Συστήματα Διαχείρισης Βάσεων Δεδομένων παρέχουν μία γλώσσα υποβολής ερωτήσεων προς τη Βάση Δεδομένων. Αυτές οι ερωτήσεις επεξεργάζονται από το Λογισμικό Χειρισμού Ερωτήσεων και στη συνέχεια μετατρέπονται σε εντολές για ανάκληση συγκεκριμένων δεδομένων από το υλικό και μεταφέρονται στο Λογισμικό Αποθήκευσης και Ανάκλησης Δεδομένων.

Στο αριστερό μέρος του σχήματος 1 απεικονίζονται Ν Βάσεις Δεδομένων. (Με το σύστημα μπορούν να υλοποιηθούν αρκετές Βάσεις Δεδομένων που ο αριθμός τους στο σχήμα 1 ορίζεται ως Ν.) Ο χρήστης μίας Βάσης Δεδομένων δεν αντιλαμβάνεται τη διαδικασία, όπως την υλοποιεί το Σύστημα Διαχείρισης Βάσεων Δεδομένων. Συνήθως, δεν γνωρίζει που βρίσκονται τα δεδομένα (δηλαδή σε ποιο υλικό είναι αποθηκευμένα). Ο χρήστης απλά εκτελεί λειτουργίες (εισαγωγή, διαγραφή, ή τροποποίηση) στα δεδομένα και υποβάλει ερωτήσεις. Βλέπει μόνο τη Βάση Δεδομένων χωρίς να τον απασχολούν οι λειτουργίες του Συστήματος Διαχείρισης Βάσεων Δεδομένων.

2.3 Τι παρέχει ένα Σύστημα Διαχείρισης Βάσεων Δεδομένων

Όπως αναφέραμε προηγουμένως μία Βάση Δεδομένων μπορεί να υλοποιηθεί και αυτόνομα χωρίς να χρησιμοποιηθεί ένα Σύστημα Διαχείρισης Βάσεων Δεδομένων. Όμως κατά κανόνα οι παροχές ενός Συστήματος Διαχείρισης Βάσεων Δεδομένων είναι αυτές που τελικά ωθούν τους χρήστες να το χρησιμοποιήσουν. Στην ενότητα αυτή θα περιγράψουμε τι παρέχει ένα Σύστημα Διαχείρισης Βάσεων Δεδομένων.

Ευκολία στη Σχεδίαση και στην Υλοποίηση

Όπως ήδη γνωρίζετε ένας χρήστης μίας Βάσης Δεδομένων θέλει αρχικά να σχεδιάσει τη Βάση Δεδομένων και να ορίσει τους τύπους δεδομένων που θα χρησιμοποιεί, μετά να δημιουργήσει τη Βάση Δεδομένων και στη συνέχεια να χρησιμοποιεί τη Βάση Δεδομένων ή να επιτρέπει και σε άλλους χρήστες να τη χρησιμοποιούν ταυτόχρονα με αυτόν. Η υλοποίηση του λογισμικού που πραγματοποιεί αυτές τις διαδικασίες δε χρειάζεται να γίνει, μια και όλα τα παραπάνω παρέχονται από το Σύστημα Διαχείρισης Βάσεων Δεδομένων. Επίσης το σύστημα παρέχει στον χρήστη ένα ενιαίο τρόπο για να ορίζει τις Βάσεις Δεδομένων του και μια φιλοσοφία που τον καθοδηγεί στις ενέργειές του.

Επίπεδα Χρηστών και Έλεγχος Πρόσβασης

Επειδή μία Βάση Δεδομένων μπορεί να χρησιμοποιείται από πολλούς χρήστες με διαφορετικές ανάγκες και γνώσεις, το Σύστημα Διαχείρισης Βάσεων Δεδομένων παρέχει τη δυνατότητα ορισμού διαφορετικών κατηγοριών χρηστών. Για κάθε κατηγορία χρηστών ορίζονται τα όρια εξουσιοδότησης που παρέχονται σε κάθε επίπεδο. Κάποια δεδομένα είναι πιθανόν να είναι εμπιστευτικά και να μην επιτρέπεται να έχουν πρόσβαση σε αυτά όλοι οι χρήστες. Επίσης, κάθε κατηγορία χρηστών θα πρέπει να έχει το δικαίωμα να επιτελεί διαφορετικές λειτουργίες από τις άλλες κατηγορίες. Κάποιοι χρήστες για παράδειγμα θα μπορούν μόνο να διαβάζουν δεδομένα, ενώ άλλοι θα μπορούν να εισάγουν ή να κάνουν αλλαγές στα δεδομένα. Σκεφτείτε το παράδειγμα 1.1 με τη Μετεωρολογική Βάση Δεδομένων. Εισαγωγή και τροποποίηση δεδομένων θα μπορεί να κάνει καθένας που θα θέλει να χρησιμοποιήσει τη Βάση Δεδομένων; Προφανώς όχι. Η δυνατότητα αυτή θα πρέπει να δίνεται μόνο στο εξουσιοδοτημένο προσωπικό (Μετεωρολόγους), ενώ οι υπόλοιποι θα πρέπει να μπορούν μόνο να βλέπουν τα δεδομένα.

Το Σύστημα Διαχείρισης Βάσεων Δεδομένων παρέχει διαφορετικά επίπεδα χρηστών ανάλογα με τις δυνατότητες και απαιτήσεις καθενός. Επίσης το σύστημα φροντίζει για την τήρηση των κανόνων ασφαλείας και ελέγχει την πρόσβαση σε αυτό και στις Βάσεις Δεδομένων και δεν επιτρέπει παραβάσεις των επιπέδων πρόσβασης. Τα επίπεδα πρόσβασης δεν ισχύουν μόνο για τις Βάσεις Δεδομένων, αλλά και για το ίδιο το Σύστημα Διαχείρισης Βάσεων Δεδομένων.

Προστασία από Βλάβες Υλικού

Το Σύστημα Διαχείρισης Βάσεων Δεδομένων παρέχει δυνατότητες για τήρηση αντιγράφων ασφαλείας καθώς και δυνατότητες ανάληψης της Βάσης Δεδομένων μετά από βλάβες. Έτσι εξασφαλίζει καλύτερη προστασία των δεδομένων. Η ανάληψη (ή ανάνηψη) των δεδομένων είναι η διαδικασία κατά την οποία το Σύστημα Διαχείρισης Βάσεων Δεδομένων είτε αυτόματα (χωρίς τη συμβολή του διαχειριστή) είτε χειροκίνητα (με τη συμβολή του διαχειριστή) ανακτά τα δεδομένα που χάθηκαν έπειτα από μία βλάβη.

Δυνατότητα Περιγραφής Περιορισμών Ορθότητας

Σε μία Βάση Δεδομένων μπορούν να τεθούν περιορισμοί ορθότητας. Τέτοιοι περιορισμοί σχετίζονται με την οργάνωση των δεδομένων στη Βάση, ή τις τιμές που μπορούν να λάβουν κάποια δεδομένα.

Για παράδειγμα στη Μετεωρολογική Βάση Δεδομένων που αναφέραμε στο παράδειγμα 1.1, θα μπορούσαμε να ορίσουμε ότι η τιμή της θερμοκρασίας θα κυμαίνεται από –50 έως +50 βαθμούς. Με αυτόν τον τρόπο αν κάποιος εισήγαγε κατά λάθος την τιμή «177» αντί για «17» το Σύστημα Διαχείρισης Βάσεων Δεδομένων θα εντόπιζε το λάθος και δε θα δεχόταν την τιμή. (Θα αντιλαμβανόταν δηλαδή ότι η τιμή 177 δεν μπορεί να είναι πραγματική τιμή για θερμοκρασία.)

Με αυτό τον τρόπο το σύστημα υλοποιεί ελέγχους και προστατεύει τη Βάση Δεδομένων από λαθεμένες εισαγωγές δεδομένων. Προσέξτε όμως: Ακόμα και με τον παραπάνω περιορισμό, κάποιος θα μπορούσε να κάνει λάθος και αντί για «17» να εισάγει «27». Αυτό το λάθος δεν μπορεί να εντοπιστεί από το Σύστημα Διαχείρισης Βάσεων Δεδομένων, αλλά μόνο από τους χρήστες.

Δυνατότητα Ελέγχου Πλεονασμών

Το Σύστημα Διαχείρισης Βάσεων Δεδομένων πραγματοποιεί και ελέγχους για πλεονασμό δεδομένων. Υποθέστε ότι, αντί για Σύστημα Διαχείρισης Βάσεων Δεδομένων χρησιμοποιούσαμε μία Βάση Δεδομένων με καρτέλες όπως στο παράδειγμα 1.2, και ότι για την τήρηση των αρχείων ήταν επιφορτισμένοι δύο υπάλληλοι. Από κάποιο λάθος λοιπόν έγραψαν και οι δύο καρτέλα για την «Πάτρα – Δεκέμβριος 2009». Εκτός από το πρόβλημα ότι θα είχαμε διπλά τα δεδομένα, θα μπορούσαν να προκύψουν μεγαλύτερα προβλήματα που σχετίζονται με την συμβατότητα των δεδομένων. Υποθέστε ότι εντοπίστηκε ένα λάθος στις μετρήσεις και η τιμή για την ημερομηνία 16 Δεκεμβρίου πρέπει να αλλάξει και από 12 °C πρέπει να γίνει 15 °C. Εάν κάποιος αναλάβει να κάνει την αλλαγή θα την κάνει σε μία μόνο καρτέλα, ενώ η άλλη θα συνεχίζει να έχει την λάθος τιμή. Αυτό σημαίνει ότι η Βάση Δεδομένων θα περιέχει λάθος δεδομένα, παρά τις διορθώσεις. Στην περίπτωση του Συστήματος Διαχείρισης Βάσεων Δεδομένων οι πλεονασμοί ελέγχονται αυτόματα από το σύστημα και η συμβατότητα των δεδομένων προστατεύεται από το ίδιο το Σύστημα Διαχείρισης Βάσεων Δεδομένων απλοποιώντας την εργασία του χρήστη.

2.4 Τι παρέχει η Βάση Δεδομένων

Ταυτόχρονη Προσπέλαση

Ένα βασικό κέρδος από την χρήση Βάσης Δεδομένων είναι η δυνατότητα που παρέχει σε πολλούς χρήστες να έχουν ταυτόχρονη προσπέλαση στα δεδομένα, αλλά και η αυτοματοποίηση των ελέγχων που συνεπάγεται αυτή η ταυτόχρονη προσπέλαση. Για παράδειγμα, το Σύστημα Διαχείρισης της Βάσης Δεδομένων φροντίζει να αποκλείει περιπτώσεις ταυτόχρονης αλλαγής ενός δεδομένου από δύο ή περισσότερους χρήστες.

Επίσης, το Σύστημα Διαχείρισης της Βάσης Δεδομένων φροντίζει για το έλεγχο των εξουσιοδοτημένων ή μη προσπελάσεων στα δεδομένα, παρέχοντας σε κάθε χρήστη μόνο τα δεδομένα αυτά για τα οποία έχει εξουσιοδότηση, και επιτρέποντάς του μόνο τις πράξεις (π.χ. διαγραφή, ή αλλαγή) αυτές για τις οποίες έχει αρμοδιότητα.

Ταχύτατη εξαγωγή απαντήσεων

Ένα μεγάλο πρόβλημα της οργάνωσης ενός μοντέλου του πραγματικού κόσμου χωρίς τη χρήση Βάσης Δεδομένων είναι η αδυναμία αποφυγής αρκετής εργασίας από τον άνθρωπο για να αντληθεί η απαιτούμενη πληροφορία, προκειμένου να εξαχθούν απαντήσεις σε απλές ερωτήσεις. Ένα σύστημα αρχείων οργανώνει την πληροφορία σειριακά σε αρχεία, χωρίς τη δυνατότητα αναζήτησης παρά μόνο σειριακά (δηλαδή ανάγνωση ενός τμήματος του αρχείου στη μνήμη, έλεγχος και μετά ανάγνωση του επόμενου τμήματος, κτλ).

Σε ένα παραδοσιακό σύστημα αρχείων σε ηλεκτρονικό υπολογιστή η απάντηση στην ερώτηση «Δώσε μας την πόλη που είχε την υψηλότερη θερμοκρασία για κάθε ημέρα από το 1990 μέχρι σήμερα», θα σήμαινε άνοιγμα εκατοντάδων αρχείων και ανθρώπινη εργασία για την καταγραφή και σύγκριση δεδομένων. Ακριβώς επειδή τα δεδομένα συνήθως συλλέγονται για την απόκτηση πληροφορίας με την μορφή απαντήσεων σε τέτοιου είδους ερωτήσεις, η χρήση παραδοσιακών μοντέλων οργάνωσης δεν είναι εξυπηρετική, αφού η παροχή τέτοιου είδους πληροφορίας είναι πρακτικά ανέφικτη. Ο λόγος είναι ότι η οργάνωση των δεδομένων δεν υποστηρίζει την αυτόματη εξαγωγή τέτοιους είδους απαντήσεων σε ερωτήσεις.

Αντίθετα, με τη χρήση Βάσης Δεδομένων τέτοιου είδους πληροφορίες μπορούν να είναι διαθέσιμες ταχύτατα στο χρήστη. Ο τρόπος οργάνωσης των δεδομένων δίνει τη δυνατότητα στις Βάσεις Δεδομένων να μπορούν να δίνουν απαντήσεις αυτόματα και ταχύτατα τόσο σε πολύπλοκες ερωτήσεις, όσο και σε απλές ερωτήσεις οι οποίες όμως θα ήταν πολύ δύσκολο να απαντηθούν εάν είχε ακολουθηθεί οργάνωση βασισμένη σε ένα παραδοσιακό σύστημα (είτε με χειρόγραφα, είτε με μη οργανωμένα σε Βάση Δεδομένων αρχεία στο ηλεκτρονικό υπολογιστή).

Ευελιξία

Πολλές φορές το μοντέλο του πραγματικού κόσμου αλλάζει. Αυτό κατά κανόνα απαιτεί αλλαγές που τις περισσότερες φορές είναι πολύ δύσκολο να υλοποιηθούν σε ένα παραδοσιακό μοντέλο. Αντίθετα, ο τρόπος οργάνωσης των δεδομένων σε μία Βάση Δεδομένων, κατά κανόνα, επιδέχεται αλλαγές και μπορεί πολύ πιο εύκολα να ακολουθεί το μοντέλο του πραγματικού κόσμου.

Αυτό σημαίνει ότι οι χρήστες της Βάσης Δεδομένων μπορούν να προσαρμόσουν την μορφή των δεδομένων ευκολότερα και οι αλλαγές αυτές να εφαρμοστούν σε όλα τα δεδομένα, σε αντίθεση με το παραδοσιακό μοντέλο.

Υψηλή Ποιότητα Δεδομένων

Η Βάση Δεδομένων χρησιμοποιεί τις δυνατότητες που παρέχει το Σύστημα Διαχείρισης Βάσεων Δεδομένων και ελέγχει για πλεονασμούς και παραβιάσεις της ορθότητας των δεδομένων, καθώς και για ταυτόχρονες προσπελάσεις και μη εξουσιοδοτημένες εγγραφές. Έτσι τα δεδομένα προστατεύονται από ένα μηχανισμό που δεν θα μπορούσε να υπάρξει σε ένα παραδοσιακό σύστημα οργάνωσης. Με αυτό τον τρόπο διασφαλίζεται η ποιότητα των δεδομένων, δηλαδή ο χρήστης έχει μεγαλύτερη ασφάλεια ότι τα δεδομένα που παίρνει ως απάντηση στις ερωτήσεις του είναι σωστά.

3 Ανασκόπηση της Διαδικασίας Σχεδιασμού Βάσεων Δεδομένων

Στο κεφάλαιο αυτό εξετάζουμε τις διάφορες φάσεις της διαδικασίας σχεδιασμού βάσεων δεδομένων. Για μικρές βάσεις δεδομένων, που θα χρησιμοποιηθούν από περιορισμένο αριθμό χρηστών, ο σχεδιασμός της βάσης δεδομένων δε χρειάζεται να είναι πολύπλοκος. Ωστόσο, όταν σχεδιάζονται μεσαίου ή μεγάλου μεγέθους βάσεις δεδομένων που θα αποτελέσουν μέρος του πληροφοριακού συστήματος ενός μεγάλου οργανισμού, ο σχεδιασμός καθίσταται αρκετά πολύπλοκος. Αυτό συμβαίνει επειδή το σύστημα πρέπει να ικανοποιεί τις απαιτήσεις πολλών διαφορετικών χρηστών. Προσεκτικός σχεδιασμός και φάσεις ελέγχου είναι αναγκαίες προϋποθέσεις για να εξασφαλίσουμε ότι όλες αυτές οι απαιτήσεις θα ικανοποιηθούν.

Μεσαίες και μεγάλες βάσεις δεδομένων χρησιμοποιούνται από περίπου 25 έως και μερικές εκατοντάδες χρήστες, περιέχουν εκατομμύρια bytes πληροφορίας και περιλαμβάνουν εκατοντάδες ερωτήσεις και προγράμματα εφαρμογών. Τέτοιες βάσεις δεδομένων θα βρούμε στη βιομηχανία, στο δημόσιο και σε μεγάλους γενικότερα φορείς. Υπάρχουν βέβαια και σχετικές περιπτώσεις (τράπεζες, ξενοδοχεία, τηλεπικοινωνίες, κλπ) που η λειτουργία της επιχείρησης εξαρτάται απόλυτα από την απρόσκοπτη λειτουργία των βάσεων δεδομένων τους σε 24ωρη βάση. Τα συστήματα αυτού του τύπου των εφαρμογών ονομάζονται συχνά *συστήματα επεξεργασίας δοσοληψιών*, λόγω του μεγάλου αριθμού των δοσοληψιών που διεξάγονται κάθε ημέρα. Θα εστιάσουμε εδώ στη διαδικασία σχεδιασμού για τέτοιες μεσαίου και μεγάλου μεγέθους βάσεις δεδομένων.

Στο υπόλοιπο μέρος του κεφαλαίου ο όρος σχεδιασμός βάσεων δεδομένων αναφέρεται στη διαδικασία σχεδιασμού για ένα τέτοιο περιβάλλον.

3.1 Ο ρόλος των Πληροφοριακών Συστημάτων στους Οργανισμούς

Πλαίσιο χρήσης Συστημάτων Βάσεων Δεδομένων

Εξετάζουμε εν συντομία πως τα συστήματα βάσεων δεδομένων έχουν γίνει τμήμα των Πληροφοριακών Συστημάτων (ΠΣ). Στα ΠΣ της δεκαετίας του 60 επικρατούσαν τα συστήματα αρχείων. Οι οργανισμοί μετακινούνται σταδιακά από αυτά σε συστήματα βάσεων δεδομένων από τις αρχές της δεκαετίας του 70. Πολλοί οργανισμοί έχουν συστήσει τμήματα υπό τη διοίκηση ενός διαχειριστή βάσης δεδομένων για την επίβλεψη και τον έλεγχο των δραστηριοτήτων του κύκλου ζωής των βάσεων δεδομένων. Παρόμοια, η διαχείριση των πληροφοριακών πόρων (information resource management - IRM) αναγνωρίζεται από μεγάλους οργανισμούς ως κλειδί για επιτυχή διοίκηση της επιχείρησης. Υπάρχουν πολλοί λόγοι γι' αυτό:

- Περισσότερες λειτουργίες στους οργανισμούς μηχανογραφούνται, αυξάνοντας την ανάγκη να είναι διαρκώς διαθέσιμοι μεγάλοι όγκοι δεδομένων, οι οποίοι θα είναι ενημερωμένοι με όλες τις αλλαγές, ακόμη και του τελευταίου λεπτού.
- Καθώς αυξάνει η πολυπλοκότητα των δεδομένων και των εφαρμογών, χρειάζεται να μοντελοποιηθούν και να συντηρηθούν πολύπλοκες συσχετίσεις μεταξύ των δεδομένων.
- Υπάρχει μια τάση για ενοποίησης των πληροφοριακών πηγών σε πολλούς οργανισμούς.

Τα συστήματα βάσεων δεδομένων ικανοποιούν κατά μεγάλο μέρος τις τρεις παραπάνω απαιτήσεις. Δύο επιπλέον χαρακτηριστικά των συστημάτων βάσεων δεδομένων είναι επίσης πολύτιμα στο σχεδιασμό και τη διαχείριση μεγάλων βάσεων δεδομένων:

- Η ανεξαρτησία των δεδομένων προστατεύει τα προγράμματα εφαρμογών από αλλαγές στη λογική οργάνωση, τις διαδρομές φυσικής προσπέλασης και τις δομές αποθήκευσης.
- Τα εξωτερικά σχήματα (όψεις) επιτρέπουν να χρησιμοποιούνται τα ίδια δεδομένα από πολλές εφαρμογές, με την κάθε εφαρμογή να έχει τη δική της όψη των δεδομένων.

Ένας επιπλέον λόγος για μεταπήδηση σε συστήματα βάσεων δεδομένων είναι το χαμηλό κόστος ανάπτυξης νέων εφαρμογών, σε σχέση με το κόστος σε παλαιότερα συστήματα αρχείων. Αυτό συχνά δικαιολογεί το υψηλό αρχικό κόστος σχεδιασμού της βάσεως δεδομένων και της μετατροπής του συστήματος. Η ύπαρξη γλωσσών υψηλού επιπέδου για προσπέλαση δεδομένων απλοποιεί το έργο της συγγραφής εφαρμογών βάσεων δεδομένων, ενώ οι υψηλού επιπέδου γλώσσες ερωτήσεων καθιστούν εφικτή για τα υψηλόβαθμα στελέχη την υποβολή μη τυποποιημένων ερωτήσεων για ανάκτηση πληροφορίας.

Από τις αρχές της δεκαετίας του ’70 έως τα μέσα της δεκαετίας του ’80 η τάση ήταν προς δημιουργία μεγάλων ταμιευτήρων δεδομένων, τους οποίους διαχειριζόταν ένα συγκεντρωτικό ΣΔΒΔ. Τελευταία η τάση αυτή έχει αντιστραφεί λόγω των κάτωθι εξελίξεων:

1. Οι προσωπικοί υπολογιστές και προϊόντα λογισμικού που μοιάζουν με βάσεις δεδομένων όπως τα VISICALC, LOTUS 1-2-3, SYMPHONY, PARADOX και DBASE IV και V χρησιμοποιούνται ευρέως από χρήστες που πριν ανήκαν στην κατηγορία των περιπτωσιακών χρηστών βάσεων δεδομένων. Πολλοί διοικητικοί, μηχανικοί, αρχιτέκτονες κ.λ.π. ανήκουν στην κατηγορία αυτή. Ως αποτέλεσμα, η πρακτική της δημιουργίας προσωπικών βάσεων δεδομένων γίνεται δημοφιλής. Είναι σήμερα δυνατόν να εξαχθεί αντίγραφο ενός μέρους μιας μεγάλης βάσης δεδομένων από ένα κεντρικό υπολογιστή ή έναν εξυπηρετητή μιας βάσης δεδομένων, να γίνει επεξεργασία σ΄ έναν προσωπικό σταθμό εργασίας και το επεξεργασμένο τμήμα να αποθηκευθεί ξανά στον κεντρικό υπολογιστή. Παρόμοια, οι χρήστες μπορούν να σχεδιάζουν και να δημιουργούν τις δικές τους βάσεις δεδομένων και στη συνέχεια να τις συνενώνουν, δημιουργώντας μια μεγαλύτερη.
2. πρόοδος των κατανεμημένων συστημάτων διαχείρισης βάσεων δεδομένων δημιουργεί δυνατότητες για κατανομή της βάσης δεδομένων σε πολλά υπολογιστικά συστήματα, για καλύτερο τοπικό έλεγχο και ταχύτερη τοπική επεξεργασία. Την ίδια στιγμή, οι τοπικοί χρήστες μπορούν να προσπελάσουν απομακρυσμένα δεδομένα, χρησιμοποιώντας τις δυνατότητες που παρέχει το κατανεμημένο σύστημα διαχείρισης βάσεων δεδομένων.
3. Πολλοί οργανισμοί σήμερα χρησιμοποιούν συστήματα λεξικών δεδομένων, τα οποία είναι μικρά ΣΔΒΔ που διαχειρίζονται μετά-δεδομένα για ένα σύστημα βάσεων δεδομένων – δηλαδή δεδομένα που περιγράφουν τη δομή της βάσης δεδομένων, περιορισμούς, εφαρμογές, δικαιώματα χρήσης κ.ο.κ. Αυτά συχνά χρησιμοποιούνται ως ολοκληρωμένο εργαλείο για διαχείριση των πόρων. Ένα χρήσιμο λεξικό δεδομένων πρέπει να αποθηκεύει και να διαχειρίζεται τους ακόλουθους τύπους πληροφοριών:

3.1. Περιγραφές των οχημάτων του συστήματος βάσεων δεδομένων.

3.2. Λεπτομερείς πληροφορίες για το φυσικό σχεδιασμό της βάσης δεδομένων όπως δομές αποθήκευσης, διαδρομές προσπέλασης και μεγέθη αρχείων και εγγραφών.

3.3. Περιγραφές των χρηστών της βάσης δεδομένων, των καθηκόντων τους και των δικαιωμάτων προσπέλασης που έχουν.

3.4. Υψηλού επιπέδου περιγραφή των δοσοληψιών και εφαρμογών της βάσης δεδομένων και της συσχέτισης μεταξύ χρηστών και δοσοληψιών.

3.5. Τη συσχέτιση μεταξύ δοσοληψιών στη βάση δεδομένων και των δεδομένων που χρησιμοποιούνται από αυτές. Αυτό χρησιμεύει για να είναι γνωστό ποιες δοσοληψίες επηρεάζονται όταν οι ορισμοί κάποιων δεδομένων αλλάξουν.

3.6. Στατιστικά χρήσης, όπως συχνότητα ερωτήσεων και δοσοληψιών και μετρήσεις προσπελάσεων σε διάφορα τμήματα της βάσης δεδομένων.

Όταν οι πληροφορίες αυτές είναι διαθέσιμες στους διαχειριστές των βάσεων, τους σχεδιαστές και τους εξουσιοδοτημένους χρήστες ως τεκμηρίωση του συστήματος και είναι άμεσα προσβάσιμες (on-line). Αυτό βελτιώνει τον έλεγχο του πληροφοριακού συστήματος από τους διαχειριστές της βάσης δεδομένων και την κατανόηση και χρήσης του συστήματος από τους χρήστες. Σε πολλούς μεγάλους οργανισμούς, ένα σύστημα λεξικού δεδομένων θεωρείται τόσο σημαντικό, όσο και ένα ΣΔΒΔ. Τελευταία έχει δοθεί μεγάλη έμφαση σε συστήματα επεξεργασίας δοσοληψιών υψηλών επιδόσεων, τα οποία απαιτούν 24ωρη αδιάλειπτη λειτουργία και χρησιμοποιούνται στη βιομηχανία υπηρεσιών. Στις βάσεις αυτές γίνεται προσπέλαση από εκατοντάδες δοσοληψίες στο λεπτό, τόσο από απομακρυσμένα όσο και από τοπικά τερματικά στις εφαρμογές αυτές είναι κρίσιμη η απόδοση των δοσοληψιών, όσον αφορά το μέσο αριθμό δοσοληψιών ανά λεπτό και το μέσο και μέγιστο χρόνο απόκρισης των δοσοληψιών. Σε τέτοιους τύπους συστημάτων είναι απαραίτητος ένας προσεκτικός φυσικός σχεδιασμός των βάσεων δεδομένων έτσι ώστε να καλύπτει τις απαιτήσεις του οργανισμού σε επεξεργασία δοσοληψιών.

Μερικοί οργανισμοί είναι δεσμευμένοι σε συγκεκριμένα προϊόντα για ΣΔΒΔ και λεξικό δεδομένων. Οι επενδύσεις τους στο σχεδιασμό και την υλοποίηση μεγάλων και πολύπλοκων συστημάτων καθιστούν πολύ δύσκολη γι΄ αυτούς την αντικατάσταση του τρέχοντος από νεότερα προϊόντα ΣΔΒΔ. Έτσι οι οργανισμοί εγκλωβίζονται στα υπάρχοντα ΣΔΒΔ. σε σχέση με τέτοιες μεγάλες και πολύπλοκες βάσεις δεδομένων, η σημασία του προσεκτικού σχεδιασμού, που λαμβάνει υπόψη την αναγκαιότητα για πιθανές μελλοντικές τροποποιήσεις του συστήματος λόγω αλλαγών των απαιτήσεων, είναι ιδιαίτερα μεγάλη. Το κόστος μπορεί να είναι πολύ υψηλό αν ένα μεγάλο και πολύπλοκο σύστημα δεν μπορεί να εξελιχθεί και καταστεί απαραίτητη η μεταφορά σε άλλα προϊόντα συστημάτων διαχείρισης βάσεων δεδομένων.

3.2 Κύκλος Ζωής Πληροφοριακών Συστημάτων

Σε ένα μεγάλο οργανισμό, το σύστημα διαχείρισης βάσεων δεδομένων τυπικά αποτελεί μέρος ενός πολύ μεγαλύτερου πληροφοριακού συστήματος που χρησιμοποιείται για την διαχείριση των πληροφοριακών πόρων του οργανισμού. Ένα πληροφοριακό σύστημα περιλαμβάνει όλους τους πόρους στον οργανισμό που εμπλέκονται στη συλλογή, τη διαχείριση, τη χρήση και τη διάχυση των πληροφοριών. Σε ένα μηχανογραφημένο περιβάλλον οι πόροι αυτοί περιλαμβάνουν τα δεδομένα, το λογισμικό του ΣΔΒΔ, το υλικό του υπολογιστικού συστήματος και τα μέσα αποθήκευσης, το προσωπικό που χρησιμοποιεί και διαχειρίζεται τα δεδομένα (ΔΒΔ, τελικούς χρήστες, παραμετρικούς χρήστες κ.ο.κ.), το λογισμικό των εφαρμογών που ενημερώνουν και προσπελαύνουν τα δεδομένα και τους προγραμματιστές εφαρμογών που αναπτύσσουν τις εφαρμογές αυτές. Επομένως, το σύστημα της βάσης δεδομένων δεν είναι παρά μόνο μέρος ενός πολύ ευρύτερου πληροφοριακού συστήματος του οργανισμού.

Στα Εδάφιο αυτό εξετάζουμε τον τυπικό κύκλο ζωής ενός πληροφοριακού συστήματος και το πως το σύστημα της βάσης δεδομένων εντάσσεται στο κύκλο αυτό. Ο κύκλος ζωής του πληροφοριακού συστήματος συχνά ονομάζεται μακρόκυκλος ζωής, ενώ ο κύκλος ζωής του συστήματος της βάσης δεδομένων αναφέρεται ως μικρόκυκλος ζωής. Ο μακρόκυκλος ζωής τυπικά περιλαμβάνει τις ακόλουθες φάσεις:

1) Ανάλυση σκοπιμότητας: αφορά την ανάλυση πιθανών περιοχών εφαρμογών, εκτέλεση προκαταρκτικών μελετών κόστους- απόδοσης, και καθορισμό προτεραιοτήτων μεταξύ των εφαρμογών.

2) Συλλογή και ανάλυση απαιτήσεων: συλλέγονται λεπτομερείς απαιτήσεις σε συνεργασία με ενδεχόμενους χρήστες, για το προσδιορισμό ειδικών προβλημάτων και αναγκών.

3) Σχεδιασμός: αυτή η φάση έχει δύο πλευρές, το σχεδιασμό του συστήματος της βάσης δεδομένων και το σχεδιασμό των συστημάτων (προγραμμάτων) εφαρμογών που χρησιμοποιούν και επεξεργάζονται τη βάση δεδομένων.

4) Υλοποίηση: το πληροφορικό σύστημα υλοποιείται, η βάση δεδομένων φορτώνεται και οι δοσοληψίες της βάσης δεδομένων υλοποιούνται και ελέγχονται.

5) Πιστοποίηση και έλεγχος αποδοχής: πιστοποιείται ότι το σύστημα ικανοποιεί τις απαιτήσεις των χρηστών και τα κριτήρια απόδοσης. Το σύστημα ελέγχεται βάσει κριτηρίων απόδοσης και συμπεριφοράς.

6) Λειτουργία: η μετάπτωση των χρηστών από ένα παλαιότερο σύστημα καθώς και η εκπαίδευση των χρηστών μπορούν να προηγηθούν αυτής της φάσης. Η φάση λειτουργίας αρχίζει όταν όλες οι λειτουργίες του συστήματος είναι έτοιμες και έχουν πιστοποιηθεί. Όταν προκύπτουν νέες απαιτήσεις ή εφαρμογές, διέρχονται από τις προηγούμενες φάσεις μέχρι να πιστοποιηθούν και να ενταχθούν στο σύστημα. Κατά τη διάρκεια της φάσης λειτουργίας η παρακολούθηση της απόδοσης του συστήματος και η συντήρηση του συστήματος είναι σημαντικές λειτουργίες.

3.3 Κύκλος Ζωής Συστήματος Βάσης Δεδομένων

Οι δραστηριότητες που σχετίζονται με τον (μικρό) κύκλο ζωής του συστήματος εφαρμογών της βάση δεδομένων περιλαμβάνουν τις ακόλουθες φάσεις:

1) Ορισμός του συστήματος: ορίζονται η εμβέλεια του συστήματος της βάσης δεδομένων, οι χρήστες του και οι εφαρμογές του.
2) Σχεδιασμός: στο τέλος αυτής της φάσης είναι έτοιμος ένας πλήρης λογικός και φυσικός σχεδιασμός του συστήματος της βάσης δεδομένων για το ΣΔΒΔ που επιλέχθηκε.
3) Υλοποίηση: αυτή η φάση αποτελείται από την συγγραφή των εννοιολογικών, των εξωτερικών και των εσωτερικών ορισμών της βάσης δεδομένων, τη δημιουργία μιας βάσης δεδομένων με κενά αρχεία και την υλοποίηση του λογισμικού των εφαρμογών.
4) Φόρτωση ή μετατροπή δεδομένων: στη βάση δεδομένων καταχωρούνται δεδομένα είτε άμεσα, είτε με μετατροπή υπαρχόντων αρχείων στη μορφή που απαιτεί το σύστημα της βάσης δεδομένων.
5) Μετατροπή Εφαρμογών: κάθε λογισμικό εφαρμογής από το προηγούμενο σύστημα προσαρμόζεται στο νέο σύστημα.
6) Έλεγχος και πιστοποίηση: το νέο σύστημα ελέγχεται και πιστοποιείται.
7) Λειτουργία: τίθενται σε λειτουργία το σύστημα της βάσης δεδομένων και οι εφαρμογές του.
8) Παρακολούθηση και συντήρηση: κατά τη φάση της λειτουργία το σύστημα παρακολουθείται και συντηρείται συνεχώς. Τόσο το περιεχόμενο των δεδομένων όσο και το λογισμικό των εφαρμογών μπορούν να επαυξηθούν και να επεκταθούν. Από καιρό σε καιρό μπορεί να απαιτηθούν σημαντικές τροποποιήσεις και αναδιοργανώσεις.

Οι δραστηριότητες 2,3 και 4 ως σύνολο αποτελούν μέρος των φάσεων σχεδιασμού και υλοποίησης του μακρόκυκλου ζωής του πληροφοριακού συστήματος. Το επόμενο Εδάφιο επικεντρώνεται στη δραστηριότητα 2, που καλύπτει την φάση σχεδιασμού της βάσης δεδομένων. Οι περισσότερες βάσεις δεδομένων σε οργανισμούς περνούν από όλες τις παραπάνω δραστηριότητες του κύκλου ζωής. Τα βήματα μετατροπής (4 και 5) δεν εφαρμόζονται όταν τόσο η βάση δεδομένων όσο και τα δεδομένα είναι νέα. Όταν ένας οργανισμός αντικαθιστά ένα καθιερωμένο παλαιό σύστημα με ένα νέο, οι δραστηριότητες 4 και 5 τείνουν να είναι πιο χρονοβόρες και η προσπάθεια που απαιτείται για την υλοποίηση τους υποτιμάται. Γενικά υπάρχει ανάδραση μεταξύ των διαφορετικών βημάτων, διότι συχνά εμφανίζονται νέες απαιτήσεις σε κάθε φάση.

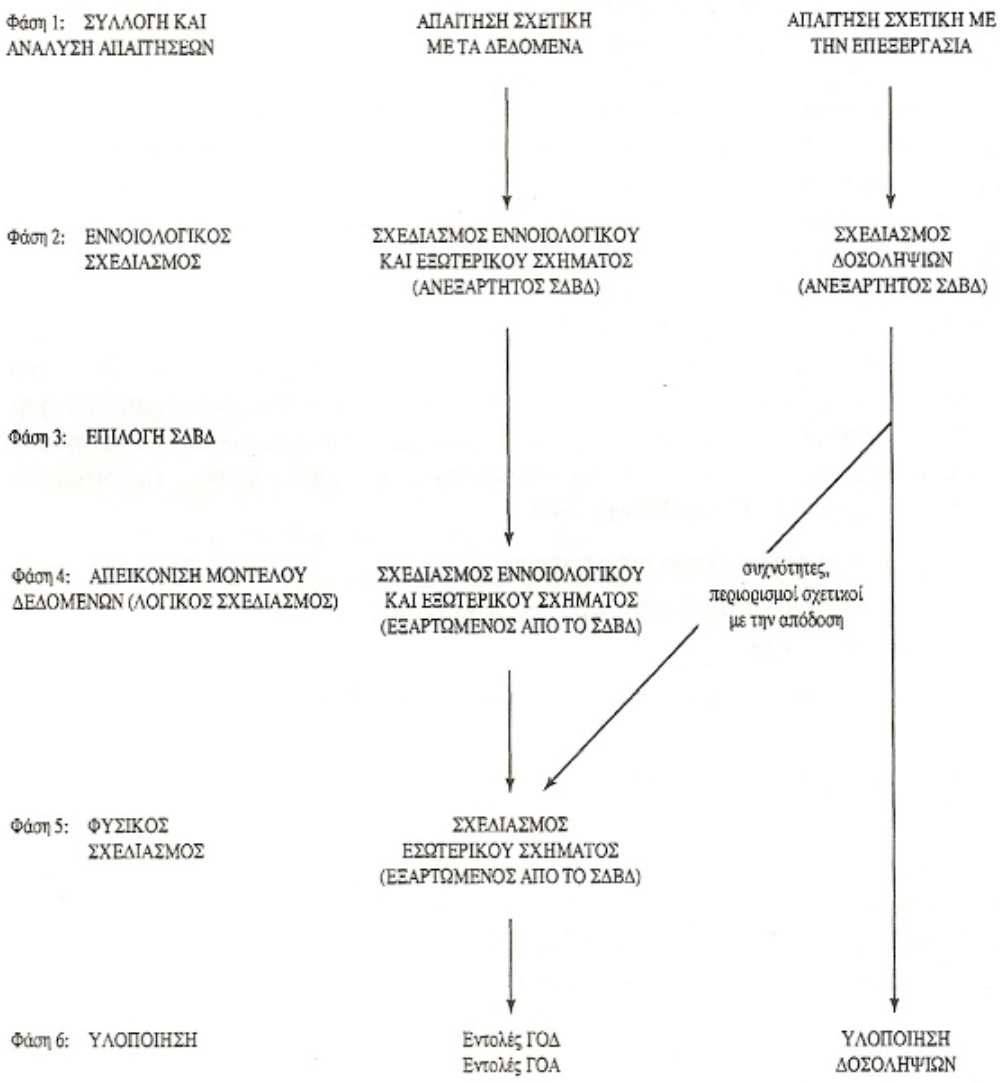
3.4 Διαδικασία Σχεδιασμού της Βάσης Δεδομένων

Εστιάζουμε τώρα στο δεύτερο βήμα του κύκλου ζωής του συστήματος εφαρμογών της βάσης δεδομένων, το οποίο ονομάζουμε σχεδιασμό της βάσης δεδομένων. Το πρόβλημα του σχεδιασμού βάσεων δεδομένων μπορεί να τεθεί ως ακολούθως: σχεδιασμός λογικής και φυσικής δομής μίας ή περισσοτέρων βάσεων δεδομένων για την κάλυψη των πληροφοριακών αναγκών των χρηστών σε έναν οργανισμό για καθορισμένο σύνολο εφαρμογών.

Οι στόχοι του σχεδιασμού βάσεων δεδομένων είναι πολλαπλοί: α) να ικανοποιηθούν οι απαιτήσεις καθορισμένων χρηστών και εφαρμογών ως προς το περιεχόμενο των πληροφοριών, β) να παρέχεται μια φυσική και ευκολονόητη δόμηση των πληροφοριών και γ) να υποστηρίζονται οι απαιτήσεις σε επεξεργασία και όλοι οι στόχοι απόδοσης όπως ο χρόνος απόκρισης, ο χρόνος επεξεργασίας και ο χώρος αποθήκευσης. Οι στόχοι αυτοί είναι πολύ δύσκολο να επιτευχθούν και να μετρηθούν. Το πρόβλημα επιδεινώνεται επειδή ο σχεδιασμός των βάσεων δεδομένων αρχίζει με πολύ άτυπες και ανεπαρκώς ορισμένες απαιτήσεις. Αντίθετα, το αποτέλεσμα της δραστηριότητας του σχεδιασμού είναι ένα αυστηρά ορισμένο σχήμα βάσης δεδομένων που δεν μπορεί να τροποποιηθεί εύκολα όταν υλοποιηθεί η βάση δεδομένων. Μπορούμε να προσδιορίσουμε έξι κύριες φάσεις της διαδικασίας σχεδιασμού της βάσης δεδομένων:

1) Συλλογή και ανάλυση απαιτήσεων.
2) Εννοιολογικός σχεδιασμός βάσης δεδομένων.
3) Επιλογή ΣΔΒΔ.
4) Απεικόνιση στο μοντέλο δεδομένων (ονομάζεται και λογικός σχεδιασμός βάσης δεδομένων).
5) Φυσικός σχεδιασμός βάσης δεδομένων.
6) Υλοποίηση συστήματος βάσης δεδομένων.

Η διαδικασία σχεδιασμού αποτελείται από δυο παράλληλες δραστηριότητες, όπως απεικονίζεται στην Εικόνα 3.1. Η πρώτη δραστηριότητα περιλαμβάνει το σχεδιασμό του περιεχομένου και της δομής της βάσης δεδομένων. Η δεύτερη σχετίζεται με το σχεδιασμό της επεξεργασίας της βάσης δεδομένων και του λογισμικού των εφαρμογών. Αυτές οι δύο δραστηριότητες είναι στενά συνυφασμένες. Για παράδειγμα, μπορούμε να προσδιορίσουμε δεδομένα που θα αποθηκευθούν στη βάση δεδομένων, αναλύοντας τις εφαρμογές της βάσης δεδομένων. Επιπλέον, η φάση του φυσικού σχεδιασμού της βάσης δεδομένων, κατά την οποία επιλέγουμε τις δομές αποθήκευσης και τις διαδρομές προσπέλασης στα αρχεία της βάσης δεδομένων, εξαρτάται από τις εφαρμογές που θα χρησιμοποιήσουν αυτά τα αρχεία. Από την άλλη πλευρά, συνήθως προσδιορίσουμε το σχεδιασμό των εφαρμογών της βάσης δεδομένων αναφερόμενοι στις δομές του σχήματος της βάσης δεδομένων, οι οποίες καθορίζονται κατά την πρώτη δραστηριότητα. Σαφώς, αυτές οι δυο δραστηριότητες επηρεάζουν πολύ η μια την άλλη. Παραδοσιακά, οι μεθοδολογίες σχεδιασμού βάσεων δεδομένων έχουν κυρίως εστιάσει στη μια μόνον από αυτές τις δραστηριότητες και ο σχεδιασμός βάσεων δεδομένων έχουν κυρίως εστιάσει στη μια μόνον από αυτές τις δραστηριότητες και ο σχεδιασμός βάσεων δεδομένων μπορεί να είναι καθοδηγούμενος από τα δεδομένα ή καθοδηγούμενος από τις διαδικασίες. Σήμερα αναγνωρίζεται ότι οι δύο δραστηριότητες πρέπει να πάνε χέρι- χέρι.



Εικόνα 3.1: Φάσεις του σχεδιασμού βάσεων δεδομένων για μεγάλες βάσεις

Οι έξι φάσεις που αναφέρονται πιο πάνω δεν είναι απαραίτητο να ακολουθηθούν αυστηρά με αυτή τη σειρά. Σε πολλές περιπτώσεις μπορεί να χρειασθεί να τροποποιήσουμε το σχεδιασμό μιας φάσης κατά τη διάρκεια μιας μεταγενέστερης φάσης. Αυτοί οι βρόχοι ανάδρασης μεταξύ φάσεων – καθώς και μέσα σε φάσεις- είναι συνηθισμένοι κατά το σχεδιασμό μιας βάσης δεδομένων. Για να αποφύγουμε πολύπλοκα διαγράμματα δεν περιλαμβάνουμε βρόχους ανάδρασης στην Εικόνα 3.1. η φάση 1 στην Εικόνα 3.1, περιλαμβάνει συλλογή πληροφοριών για την προτιθέμενη χρήση της βάσης δεδομένων, ενώ η φάση 6 αφορά την υλοποίηση της βάσης δεδομένων. Οι φάσεις 1 και 6 μερικές φορές θεωρείται ότι δεν αποτελούν μέρος αυτού καθεαυτού του σχεδιασμού μια ς βάσης δεδομένων, αλλά μέρος του γενικότερου κύκλου ζωής του πληροφοριακού συστήματος. Η καρδιά της διαδικασίας του σχεδιασμού βάσεων δεδομένων είναι οι φάσεις 2,4 και 5, τις οποίες περιγράφουμε περιληπτικά εδώ:

- Εννοιολογικός σχεδιασμός βάσης δεδομένων (φάση 2): Ο στόχος αυτής της φάσης είναι να δημιουργηθεί ένα εννοιολογικό σχήμα για τη βάση δεδομένων, το οποίο να είναι ανεξάρτητο από κάποιο συγκεκριμένο ΣΔΒΔ. Συχνά στη φάση αυτή χρησιμοποιείται ένα μοντέλο δεδομένων υψηλού επιπέδου όπως το μοντέλο ΟΣ ή το επεκτεταμένο μοντέλο ΟΣ. Επιπλέον, προσδιορίζουμε όσο το δυνατόν περισσότερες από τις γνωστές εφαρμογές και δοσοληψίες της

βάσης δεδομένων, χρησιμοποιώντας έναν συμβολισμό που είναι ανεξάρτητος από κάποιο συγκεκριμένο ΣΔΒΔ.

- Απεικόνιση στο μοντέλο δεδομένων (φάση 4): Ονομάζεται και λογικός σχεδιασμός βάσεων δεδομένων. Κατά τη φάση αυτή απεικονίζουμε (ή μετασχηματίζουμε) το εννοιολογικό σχήμα από το μοντέλο δεδομένων υψηλού επιπέδου που χρησιμοποιείται στη φάση 2 στο μοντέλο δεδομένων του ΣΔΒΔ που έχει επιλεγεί στη φάση 3. Μπορούμε να αρχίσουμε αυτή τη φάση μετά την επιλογή ενός μοντέλου δεδομένων υλοποίησης, αντί να περιμένουμε την επιλογή συγκεκριμένου ΣΔΒΔ- για παράδειγμα, αν αποφασίσουμε να χρησιμοποιήσουμε ένα σχεσιακό ΣΔΒΔ, αλλά δεν έχουμε αποφασίσει ακόμα ποιο συγκεκριμένα. Ονομάζουμε την προσέγγιση αυτή λογικό σχεδιασμό ανεξάρτητο συστήματος (αλλά εξαρτώμενο από το μοντέλο δεδομένων). Με όρους της αρχιτεκτονικής ΣΔΒΔ τριών επιπέδων, το αποτέλεσμα αυτής της φάσης είναι ένα εννοιολογικό σχήμα στο επιλεγμένο μοντέλο δεδομένων. Επιπλέον, κατά τη διάρκεια αυτής της φάσης συχνά λαμβάνει χώρα και ο σχεδιασμός των εξωτερικών σχημάτων (όψεων).
- Φυσικός σχεδιασμός βάσης δεδομένων (φάση 5): κατά τη φάση αυτή σχεδιάζουμε τις προδιαγραφές για την αποθηκευμένη βάση δεδομένων σε ότι αφορά τις φυσικές δομές αποθήκευσης, την τοποθέτηση των εγγραφών και τις διαδρομές προσπέλασης. Αυτό, με όρους τις τριεπίπεδης αρχιτεκτονικής, αντιστοιχεί στο σχεδιασμό του εσωτερικού σχήματος.

Στα επόμενα υποεδάφια εξετάζουμε κάθε μια από τις έξι φάσεις του σχεδιασμού βάσεων δεδομένων.

3.4.1 Φάση 1: Συλλογή και Ανάλυση Απαιτήσεων

Για να σχεδιάσουμε αποτελεσματικά μια βάση δεδομένων πρέπει να ξέρουμε τις προσδοκίες των χρηστών και τις προτιθέμενες χρήσεις της βάσης δεδομένων, με όσο το δυνατόν περισσότερες λεπτομέρειες. Η διαδικασία προσδιορισμού και ανάλυσης της χρήσης λέγεται συλλογή και ανάλυση απαιτήσεων. Για να προσδιορίζουμε τις απαιτήσεις πρέπει πρώτα να ορισθούν τα άλλα μέρη του πληροφοριακού συστήματος που θα αλληλεπιδρούν με το σύστημα της βάσης δεδομένων. Αυτά περιλαμβάνουν νέους και υπάρχοντες χρήστες και εφαρμογές. Οι απαιτήσεις αυτών των χρηστών και των εφαρμογών συλλέγονται και αναλύονται. Στην τυπική περίπτωση, οι κάτωθι δραστηριότητες αποτελούν μέρος αυτής της φάσης:

- Ορίζονται οι κύριες περιοχές εφαρμογών και οι ομάδες χρηστών που θα χρησιμοποιήσουν τη βάση δεδομένων. Επιλέγονται τα σημαντικά άτομα κάθε ομάδας ως κυρίως συμμετέχοντες στα επόμενα βήματα του προσδιορισμού και της συλλογής των απαιτήσεων.
- Μελετάται και αναλύεται η υπάρχουσα τεκμηρίωση που αφορά τις εφαρμογές. Γίνεται επισκόπηση της λοιπής τεκμηρίωσης – όπως εγχειρίδια τακτικής, φόρμες, αναφορές και πίνακες του οργανισμού- για να αποφασισθεί αν επηρεάζουν τη διαδικασία προσδιορισμού και συλλογής απαιτήσεων.
- Μελετάται το υπάρχον λειτουργικό περιβάλλον και η σχεδιαζόμενη χρήση των πληροφοριών. Αυτό περιλαμβάνει ανάλυση των τύπων των δοσοληψιών και των συχνοτήτων τους, καθώς και τη ροή των πληροφοριών στο σύστημα. Προσδιορίζονται τα δεδομένα εισόδου / εξόδου για τις δοσοληψίες.
- Συλλέγονται γραπτές απαντήσεις, από ενδεχόμενους χρήστες της βάσης δεδομένων, σε σύνολα ερωτήσεων. Οι ερωτήσεις αυτές περιλαμβάνουν τις προτεραιότητες των χρηστών και τη σπουδαιότητα που δίνουν στις διάφορες εφαρμογές. Μπορεί να γίνουν συνεντεύξεις με άτομα-κλειδιά για να αξιολογηθούν οι πληροφορίες και να τεθούν προτεραιότητες.

Οι παραπάνω τρόποι συλλογής των απαιτήσεων καταλήγουν σε αποτυπώσεις των απαιτήσεων, που δεν είναι καλά δομημένες και είναι – ως επί το πλείστον- άτυπες. Οι αποτυπώσεις αυτές στη συνέχεια μετατρέπονται σε μια καλύτερα δομημένη μορφή με χρήση μιας ή περισσοτέρων τεχνικών προσδιορισμού απαιτήσεων. Αυτές περιλαμβάνουν τις HIPO (hierarchical input process output), SADT (structured analysis and design technique (τεχνική δομημένης ανάλυσης και σχεδιασμού)), DFD (data flow diagrams (διαγράμματα ροής δεδομένων)), διαγράμματα Orr- Warnier, και διαγράμματα Nassi- Schneiderman. Όλες αυτές είναι διαγραμματικές μέθοδοι για την οργάνωση και παρουσίαση των απαιτήσεων πληροφοριών- επεξεργασίας. Τα διαγράμματα μπορεί να έχουν τη μορφή ιεραρχιών, διαγραμμάτων ροής, δομών ακολουθίας και επανάληψης, ή κάποια άλλη μορφή που εξαρτάται από τη μέθοδο. Συνήθως τα διαγράμματα αυτά συνοδεύονται από επιπλέον τεκμηρίωση με τη μορφή κειμένου, πινάκων, άλλων διαγραμμάτων (charts) και απαιτήσεων σχετικών με τη λήψη αποφάσεων. Υπάρχουν ολόκληρα βιβλία που εξετάζουν τη συλλογή των απαιτήσεων και τη φάση της ανάλυσης. Αυτή η φάση μπορεί να είναι αρκετά χρονοβόρα, αλλά είναι αποφασιστική για τη μελλοντική επιτυχία του πληροφοριακού συστήματος.

Έχουν προταθεί μερικές τεχνικές με βοήθεια υπολογιστή, για την αντιμετώπιση της συλλογής απαιτήσεων και της ανάλυσης. Οι τεχνικές αυτές περιλαμβάνουν αυτοματοποιημένα εργαλεία ανάλυσης απαιτήσεων, τα οποία ελέγχουν τη συνέπεια και την πληρότητα των προδιαγραφών. Οι απαιτήσεις καταχωρούνται σε ένα

ταμιευτήρα που συνήθως λέγεται βάση δεδομένων του σχεδιασμού και μπορεί να παρουσιασθεί και να ενημερωθεί κατά την πρόοδο του σχεδιασμού. Από τα πρώτα παραδείγματα είναι η PROBLEM STATEMENT LANGUAGE (Γλώσσα Διατύπωσης Προβλήματος) και ο PROBLEM STATEMENT ANALYZER (αναλυτής διατύπωσης προβλήματος) (PSL/PSA) που αναπτύχθηκαν στα πλαίσια του προγράμματος ISDOS στο Πανεπιστήμιο του Michigan..

3.4.2 Φάση 2: Εννοιολογικός Σχεδιασμός Βάσεων Δεδομένων

Η δεύτερη φάση του σχεδιασμού βάσεων δεδομένων περιλαμβάνει δυο παράλληλες δραστηριότητες. Η πρώτη δραστηριότητα, ο σχεδιασμός εννοιολογικού σχήματος, εξετάζει τις σχετικές με τα δεδομένα απαιτήσεις που προέκυψαν ως αποτέλεσμα της φάσης 1 και παράγει ένα εννοιολογικό σχήμα της βάσης δεδομένων. Η δεύτερη δραστηριότητα, ο σχεδιασμός δοσοληψιών, εξετάζει τις εφαρμογές της βάσης δεδομένων που αναλύθηκαν στη φάση 1 και παράγει προδιαγραφές υψηλού επιπέδου για τις δοσοληψίες αυτές.

Φάση 2α: Σχεδιασμός Εννοιολογικού Σχήματος.

Το εννοιολογικό σχήμα που παράγεται από τη φάση αυτή συνήθως περιέχεται σε ένα μοντέλο δεδομένων υψηλού επιπέδου που είναι ανεξάρτητο από το ΣΔΒΔ και επομένως δεν μπορεί να χρησιμοποιηθεί άμεσα για υλοποίηση της βάσης δεδομένων. Η σπουδαιότητα ενός εννοιολογικού μοντέλου ανεξάρτητου από το ΣΔΒΔ είναι ιδιαίτερα μεγάλη για τους ακόλουθους λόγους:

1) Ο στόχος σχεδιασμού του εννοιολογικού σχήματος είναι η πλήρης κατανόηση της δομής της βάσης δεδομένων, της σημασιολογίας, των συσχετίσεων μεταξύ δεδομένων και των περιορισμών. Αυτό επιτυγχάνεται καλύτερα όταν γίνεται ανεξάρτητα από το συγκεκριμένο ΣΔΒΔ, γιατί κάθε ΣΔΒΔ έχει ιδιαιτερότητες και περιορισμούς που δεν θα πρέπει να επιτρέπεται να επηρεάζουν το σχεδιασμό του εννοιολογικού σχήματος.

2) Το εννοιολογικό σχήμα είναι σημαντικό ως σταθερή περιγραφή του περιεχομένου της βάσης δεδομένων. Η επιλογή του ΣΔΒΔ και οι σχεδιαστικές αποφάσεις που θα ακολουθήσουν μπορεί να αλλάξουν, χωρίς να αλλάξει το εννοιολογικό σχήμα που είναι ανεξάρτητο από το ΣΔΒΔ.

3) Η καλή αντίληψη του εννοιολογικού σχήματος είναι κρίσιμη για τους χρήστες της βάσης δεδομένων και τους σχεδιαστές εφαρμογών. Επομένως είναι σημαντικής η χρήση ενός μοντέλου δεδομένων υψηλού επιπέδου, που είναι πιο εκφραστικό και γενικό από μοντέλα δεδομένων συγκεκριμένων ΣΔΒΔ.

4) Η διαγραμματική περιγραφή του εννοιολογικού σχήματος μπορεί να λειτουργήσει άριστα σαν μέσο επικοινωνίας μεταξύ χρηστών της βάσης δεδομένων, σχεδιαστών και αναλυτών. Επειδή τα μοντέλα δεδομένων υψηλού επιπέδου συνήθως στηρίζονται σε πιο ευκολονόητες έννοιες από ότι τα χαμηλού επιπέδου μοντέλα δεδομένων των ΣΔΒΔ, κάθε επικοινωνία που αφορά το σχεδιασμό του σχήματος γίνεται πιο ακριβής.

Στη φάση αυτή του σχεδιασμού βάσεων δεδομένων είναι σημαντικό να χρησιμοποιηθεί ένα μοντέλο δεδομένων υψηλού επιπέδου- γνωστό και ως σημασιολογικό ή εννοιολογικό μοντέλο δεδομένων- που έχει τα ακόλουθα χαρακτηριστικά:

5) Εκφραστικότητα: το μοντέλο δεδομένων πρέπει να είναι εκφραστικό ώστε να ξεχωρίζουν οι διάφοροι τύποι δεδομένων, συσχετίσεων και περιορισμών.

6) Απλότητα: το μοντέλο πρέπει να είναι απλό ώστε οι μέσοι , μη εξειδικευμένοι χρήστες να κατανοούν και να χρησιμοποιούν τις έννοιές του.

7) Ελαχιστότητα: το μοντέλο πρέπει να έχει μικρό πλήθος βασικών εννοιών που θα έχουν διακριτή και μη επικαλυπτόμενη σημασία.

8) Διαγραμματική παράσταση: το μοντέλο πρέπει να έχει διαγραμματικό και εύκολο στην κατανόηση συμβολισμό για την απεικόνιση του εννοιολογικού σχήματος.

9) Τυπικότητα: ένα εννοιολογικό μοντέλο που εκφράζεται με όρους του μοντέλου δεδομένων πρέπει να αποτελεί έναν τυπικό προσδιορισμό των δεδομένων χωρίς ασάφειες. Επομένως, οι έννοιες του μοντέλου πρέπει να ορίζονται με ακρίβεια και χωρίς ασάφειες.

Οι απαιτήσεις αυτές είναι μερικές φορές συγκρουόμενες. Ειδικά η απαίτηση 1 βρίσκεται σε αντίθεση με τις άλλες απαιτήσεις. Πολλά εννοιολογικά μοντέλα υψηλού επιπέδου έχουν προταθεί για το σχεδιασμό βάσεων δεδομένων. Σε ότι ακολουθεί, θα χρησιμοποιήσουμε το Επεκτεταμένο Μοντέλο Οντοτήτων- Συσχετίσεων (ΕΟΣ) και θα υποθέσουμε ότι χρησιμοποιείται στη φάση αυτή.

ΠΡΟΣΕΓΓΙΣΕΙΣ ΓΙΑ ΤΟ ΣΧΕΔΙΑΣΜΟ ΤΟΥ ΕΝΝΟΙΟΛΟΓΙΚΟΥ ΣΧΗΜΑΤΟΣ:

Για το σχεδιασμό του εννοιολογικού σχήματος πρέπει να προσδιορίσουμε τις βασικές συνιστώσες του σχήματος: τους τύπους οντοτήτων, τους τύπους συσχετίσεων και τα γνωρίσματά τους. Πρέπει επίσης να προσδιορίσουμε τα γνωρίσματα- κλειδιά, την πληθικότητα και τους περιορισμούς συμμετοχής στις συσχετίσεις, τους μη ισχυρούς τύπους οντοτήτων και τις ιεραρχίες και τα πλέγματα ειδικεύσεων / γενικεύσεων (αν χρειάζεται). Ο σχεδιασμός αυτός παράγεται από τις απαιτήσεις που συλλέγονται κατά τη φάση 1.

Υπάρχουν δυο προσεγγίσεις για το σχεδιασμό εννοιολογικού σχήματος. Με τον πρώτο, που καλείται συγκεντρωτική προσέγγιση σχεδιασμού σχήματος, συγχωνεύονται οι απαιτήσεις των διαφόρων εφαρμογών και των ομάδων χρηστών από τη φάση 1 σε ένα σύνολο απαιτήσεων, πριν αρχίσει ο σχεδιασμός του σχήματος. Στη συνέχεια σχεδιάζεται ένα σχήμα που αντιστοιχεί στο σύνολο αυτό των απαιτήσεων. Όταν υπάρχουν πολλοί χρήστες και εφαρμογές, η συγχώνευση όλων των απαιτήσεων μπορεί να είναι πολύ επίπονη και χρονοβόρα διαδικασία. Αυτή η προσέγγιση στηρίζεται στο ότι μια κεντρική εξουσία, Ο ΔΒΔ, είναι υπεύθυνος για να αποφασίσει πως θα συγχωνευθούν οι απαιτήσεις διάφορων χρηστών και εφαρμογών, καθώς και για το σχεδιασμό του εννοιολογικού σχήματος για ολόκληρη τη βάση δεδομένων. Μόλις σχεδιασθεί και παγιωθεί το εννοιολογικό σχήμα, μπορούν προσδιορισθούν από τον ΔΒΔ τα εξωτερικά σχήματα για τις διάφορες ομάδες χρηστών και εφαρμογών.

Στη δεύτερη προσέγγιση που ονομάζουμε προσέγγιση με ενοποίηση των όψεων, δεν συγχωνεύουμε τις απαιτήσεις, αλλά σχεδιάζεται ένα σχήμα (όψη) για κάθε ομάδα χρηστών ή εφαρμογή με βάση μόνον τις απαιτήσεις της. Ως αποτέλεσμα, αναπτύσσουμε ένα σχήμα υψηλού επιπέδου (όψη) για κάθε ομάδα χρηστών ή εφαρμογή. Στη συνέχεια, σε μια φάση ενοποίησης όψεων, αυτά τα σχήματα συγχωνεύονται ή ενοποιούνται σ' ένα καθολικό εννοιολογικό σχήμα για όλα τη βάση δεδομένων. Οι επιμέρους όψεις μπορούν να αναπαραχθούν ως εξωτερικά σχήματα μετά την ενοποίηση των σχημάτων.

Η κύρια διαφορά μεταξύ των δυο προσεγγίσεων βρίσκεται στον τρόπο και στο στάδιο στο οποίο οι πολλαπλές όψεις ή απαιτήσεις πολλών χρηστών και εφαρμογών ενοποιούνται και συγχωνεύονται. Στη συγκεντρωτική προσέγγιση η ενοποίηση γίνεται χειρονακτικά από το προσωπικό με αρμοδιότητες ΔΒΔ πριν τον σχεδιασμό των σχημάτων και εφαρμόζεται άνετα στις απαιτήσεις που συλλέγονται στη φάση 1. Η προσέγγιση αυτή έχει χρησιμοποιηθεί παραδοσιακά για την επίλυση των διαφορών και των συγκρούσεων μεταξύ ομάδων χρηστών, παρά το φόρτο που προσθέτει στο προσωπικό με αρμοδιότητες ΔΒΔ, έτσι ώστε το προσωπικό αυτό να επιτύχει ξεκάθαρο ορισμό των καθολικών απαιτήσεων πριν επιχειρηθεί ο σχεδιασμός του σχήματος. Επειδή υπάρχουν δυσκολίες για να επιτευχθεί αυτός ο στόχος, η προσέγγιση της ενοποίησης όψεων κερδίζει σήμερα ευρύτερη αποδοχή στην προσέγγιση με ενοποίηση όψεων, κάθε ομάδα χρηστών ή εφαρμογή στην πραγματικότητα σχεδιάζει το δικό της εννοιολογικό (ΕΟΣ) σχήμα από τις απαιτήσεις της. Στη συνέχεια εφαρμόζεται μια διαδικασία ενοποίησης στα σχήματα (όψεις) αυτά από τον ΔΒΔ, για το σχηματισμό ενός ενοποιημένου καθολικού σχήματος. Αν και η ενοποίηση όψεων μπορεί να μην είναι αυτοματοποιημένη, η εφαρμογή της σε μια μεγάλη βάση δεδομένων που περιλαμβάνει δεκάδες ομάδες χρηστών απαιτεί μια μεθοδολογία και χρήση αυτόματων εργαλείων, για να βοηθήσουν στην επίτευξη αυτής της ενοποίησης. Οι αντιστοιχίες μεταξύ γνωρισμάτων, τύπων οντοτήτων και τύπων συσχετίσεων σε διάφορες όψεις πρέπει να ορισθούν πριν εφαρμοστεί η ενοποίηση. Επιπρόσθετα, πρέπει να επιλυθούν προβλήματα όπως η ενοποίηση συγκρουόμενων όψεων και η πιστοποίηση της συμβατότητας των αντιστοιχιών μεταξύ διαφορετικών σχημάτων (όψεων).

ΣΤΡΑΤΗΓΙΚΕΣ ΣΧΕΔΙΑΣΜΟΥ ΣΧΗΜΑΤΟΣ: Όταν δίδεται ένα σύνολο από απαιτήσεις είτε για ένα χρήστη είτε για μια κοινότητα χρηστών, πρέπει να δημιουργήσουμε ένα εννοιολογικό σχήμα που να ικανοποιεί τις απαιτήσεις αυτές. Υπάρχουν διάφορες στρατηγικές για το σχεδιασμό ενός τέτοιου σχήματος. Οι περισσότερες από τις στρατηγικές αυτές ακολουθούν μια αυξητική προσέγγιση, δηλαδή αρχίζουν με κάποιες δομές του σχήματος που παράγονται από τις απαιτήσεις και στη συνέχεια διαδοχικά τροποποιούνται, εκλεπτύνονται ή χρησιμοποιούνται ως δομικά υλικά. Εξετάζουμε τώρα μερικές από τις στρατηγικές αυτές:

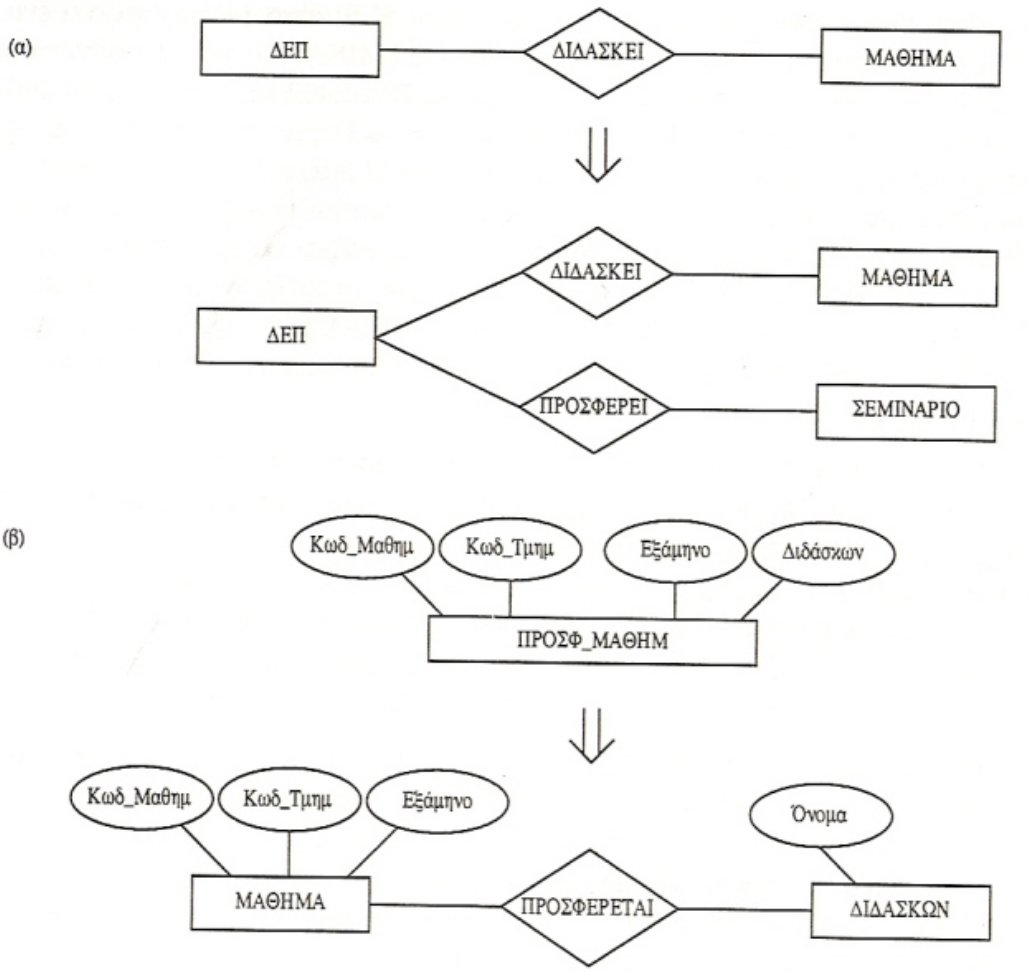
1) Καθοδική στρατηγική (top-down): Αρχίζουμε με ένα σχήμα που περιέχει αφαιρέσεις υψηλού επιπέδου και στη συνέχεια εφαρμόζουμε καθοδικά εκλεπτύνσεις. Για παράδειγμα, μπορούμε να αρχίσουμε προσδιορίζοντας λίγους τύπους οντοτήτων υψηλού επιπέδου. Στη συνέχεια, καθώς προσδιορίζουμε γνωρίσματά τους, τις διαχωρίζουμε σε τύπους οντοτήτων χαμηλότερου επιπέδου και σε συσχετίσεις. Ένα άλλο παράδειγμα καθοδικής στρατηγικής σχεδιασμού είναι η διαδικασία ειδίκευσης για εκλέπτυνση των τύπων οντοτήτων σε υποκλάσεις.

2) Ανοδική στρατηγική (bottom- up): Αρχίζουμε με ένα σχήμα που περιέχει βασικές αφαιρέσεις και στη συνέχεια συνδυάζουμε αυτές τις αφαιρέσεις ή προσθέτουμε αφαιρέσεις. Για παράδειγμα, μπορούμε να αρχίζουμε με τα γνωρίσματα και να τα ομαδοποιούμε σε τύπους οντοτήτων και σε συσχετίσεις. Μπορούμε προοδευτικά κατά τον σχεδιασμό να προσθέσουμε νέες συσχετίσεις μεταξύ τύπων οντοτήτων. Ένα άλλο παράδειγμα στρατηγική ανοδικού σχεδιασμού είναι η διαδικασία της γενίκευσης υποκλάσεων σε γενικευμένες κλάσεις υψηλότερου επιπέδου.

3) Στρατηγική από τον πυρήνα προς την περιφέρεια (inside-out): Αυτή είναι μια ειδική περίπτωση της ανοδικής στρατηγικής, όπου εστιάζουμε σε ένα κεντρικό σύνολο από έννοιες που είναι οι πιο προφανείς. Στη συνέχεια η

μοντελοποίηση επεκτείνεται με τη θεώρηση νέων εννοιών γύρω από τις υπάρχουσες. Για παράδειγμα, θα μπορούσαμε να προσδιορίσουμε μερικούς πολύ προφανείς τύπους οντοτήτων στο σχήμα και να συνεχίσουμε προσθέτοντας άλλους τύπους οντοτήτων και συσχετίσεων που σχετίζονται με καθέναν από τους αρχικούς.

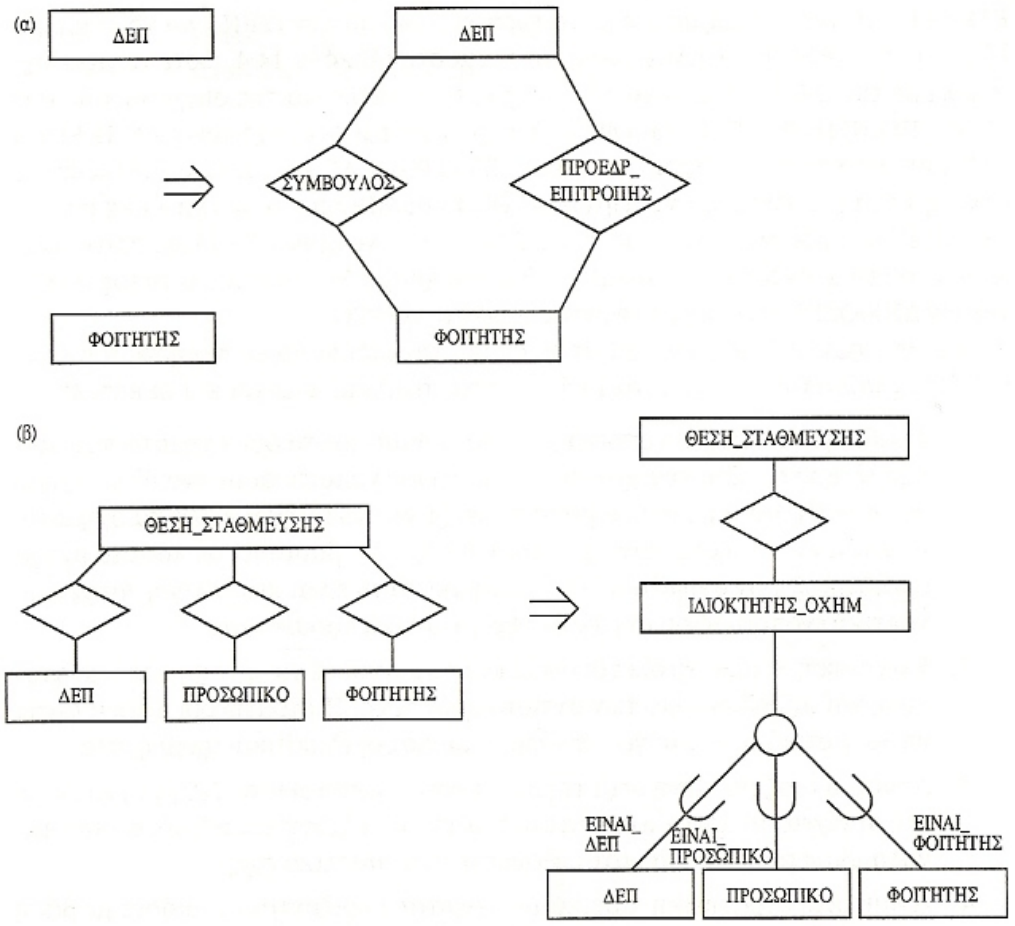
4) Ανάμεικτη στρατηγική: Αντί να ακολουθήσουμε μια συγκεκριμένη στρατηγική κατά το σχεδιασμό, διαμερίζουμε τις απαιτήσεις σύμφωνα με την καθοδική στρατηγική και σχεδιάζουμε μέρος του σχήματος με την ανοδική στρατηγική για κάθε διαμέριση. Στη συνέχεια, τα διάφορα τμήματα του σχήματος συνδυάζονται μεταξύ τους.



Εικόνα 3.2: Παραδείγματα καθοδικής εκλέπτυνσης, (α) Δημιουργία ενός νέου τύπου οντοτήτων και β) Αποσύνθεση ενός τύπου οντοτήτων σε δύο τύπους και μια συσχέτιση

Οι Εικόνες 3.2 και 3.3 δείχνουν την καθοδική και ανοδική εκλέπτυνση, αντίστοιχα. Ένα παράδειγμα πρωταρχικής καθοδικής εκλέπτυνσης είναι η αποσύνθεση ενός τύπου οντοτήτων σε διάφορους τύπους οντοτήτων. Η Εικόνα 3.2(α) δείχνει ένα ΜΑΘΗΜΑ που εκλεπτύνεται σε ΜΑΘΗΜΑ και ΣΕΜΙΝΑΡΙΟ, και η συσχέτιση ΔΙΔΑΣΚΕΙ διασπάται αντίστοιχα σε ΔΙΔΑΣΚΕΙ και ΠΡΟΣΦΕΡΕΙ. Η Εικόνα 3.2 (β) δείχνει ένα τύπο οντοτήτων ΠΡΟΣΦ_ΜΑΘΗΜ που εκλεπτύνεται σε δυο τύπους οντοτήτων και μια συσχέτιση μεταξύ τους. Η Εικόνα 3.3 (α) δείχνει την πρωταρχική μοναδική εκλέπτυνση για τη δημιουργία νέων συσχετίσεων μεταξύ τύπων οντοτήτων. Η ανοδική εκλέπτυνση με χρήση γενίκευσης απεικονίζεται στην Εικόνα 3.3 (β), όπου «ανακαλύπτεται» από τους υπάρχοντες τύπους οντοτήτων ΔΕΠ (Διδακτικό και

Επιστημονικό Προσωπικό), ΦΟΙΤΗΤΗΣ και ΠΡΟΣΩΠΙΚΟ, η νέα έννοια ΙΔΙΟΚΤΗΤΗΣ_ΟΧΗΜ..



Εικόνα 3.3: Παραδείγματα εκλέπτυνσης από κάτω προς τα άνω. (α) Εντοπισμός και πρόσθεση νέων συσχετίσεων και (β) Εντοπισμός ενός νέου, γενικευμένου τύπου οντοτήτων και συσχετισμός του.

ΕΝΟΠΟΙΗΣΗ ΣΧΗΜΑΤΩΝ (ΟΨΕΩΝ): Για μεγάλες βάσεις δεδομένων με πολλούς χρήστες και εφαρμογές, μπορεί να χρησιμοποιηθεί η προσέγγιση με ενοποίηση όψεων, με σχεδιασμό επιμέρους σχημάτων και στη συνέχεια τη συγχώνευσή τους. Επειδή οι επιμέρους όψεις μπορούν να κρατηθούν μικρές σε μέγεθος, ο σχεδιασμός των σχημάτων μπορεί να απλουστευθεί. Ωστόσο, απαιτείται μια μεθοδολογία ενοποίησης των όψεων σε ένα καθολικό σχήμα βάσης δεδομένων. Η ενοποίηση των σχημάτων μπορεί να χωρισθεί στις παρακάτω επιμέρους εργασίες:

1) Εντοπισμός των αντιστοιχιών και των συγκρούσεων μεταξύ των σχημάτων: επειδή τα σχήματα σχεδιάζονται ξεχωριστά, είναι απαραίτητο να προσδιορισθούν δομές στα σχήματα που παριστάνουν την ίδια έννοια του πραγματικού κόσμου. Αυτές οι αντιστοιχίες πρέπει να προσδιορισθούν πριν προχωρήσει η ενοποίηση. Κατά τη διαδικασία αυτή μπορεί να εντοπισθούν διάφοροι τύποι συγκρούσεων μεταξύ των σχημάτων:

   a) Συγκρούσεις ονομάτων: υπάρχουν δυο τύποι τέτοιων συγκρούσεων: συνώνυμα και ομώνυμα. Ένα συνώνυμο προκύπτει όταν δυο σχήματα χρησιμοποιούν διαφορετικά ονόματα για να περιγράψουν την ίδια έννοια. Για παράδειγμα, ο τύπος οντοτήτων ΦΟΙΤΗΤΗΣ σε ένα σχήμα μπορεί να περιγράφει τον ίδιο τύπο οντοτήτων που καλείται ΣΠΟΥΔΑΣΤΗΣ σε ένα άλλο σχήμα. Ένα ομώνυμο προκύπτει όταν δυο σχήματα χρησιμοποιούν το

ίδιο όνομα για να περιγράψουν διαφορετικές έννοιες. Για παράδειγμα, ο τύπος οντότητας ΑΝΤΑΛΛΑΚΤΙΚΟ μπορεί να παριστάνει ανταλλακτικά υπολογιστών σε ένα σχήμα και αυτοκινήτων σε ένα άλλο.

b) Συγκρούσεις τύπων: η ίδια έννοια μπορεί να παριστάνεται σε δύο σχήματα με διαφορετικές δομές μοντελοποίησης. Για παράδειγμα η έννοια ΤΜΗΜΑ μπορεί σε ένα σχήμα να είναι τύπος οντοτήτων και γνώρισμα σε ένα άλλο.

c) Συγκρούσεις πεδίου ορισμού (συνόλου τιμών): ένα γνώρισμα μπορεί να έχει διαφορετικά πεδία ορισμού σε δυο σχήματα. Για παράδειγμα, ο ΑΡ_ΜΗΤΡΩΟΥ μπορεί να δηλώνεται ως ακέραιος σε ένα άλλο σχήμα και ως συμβολοσειρά σε ένα άλλο. Σύγκρουση στη μονάδα μέτρησης προκύπτει αν ένα σχήμα μετρά το ΒΑΡΟΣ σε κιλά και ένα άλλο σε λίβρες.

d) Συγκρούσεις μεταξύ περιορισμών: δυο σχήματα μπορεί να θέτουν διαφορετικούς περιορισμούς. Για παράδειγμα, το κλειδί ενός τύπου οντοτήτων μπορεί να είναι διαφορετικό σε κάθε σχήμα. ένα άλλο παράδειγμα περιλαμβάνει διαφορετικούς δομικούς περιορισμούς σε μια συσχέτιση όπως η ΔΙΔΑΣΚΕΙ, που σε ένα σχήμα μπορεί να παριστάνεται ως 1:N (ένα μάθημα έχει ένα διδάσκονται), ενώ σε άλλο σχήμα παριστάνεται ως M:N (ένα μάθημα μπορεί να διδάσκεται από περισσότερους του ενός διδάσκοντες).

2) Τροποποίηση των όψεων για να προσαρμοστεί η μια στην άλλη: μερικά σχήματα τροποποιούνται έτσι ώστε να προσαρμόζονται σε άλλα σχήματα. Στο βήμα αυτό επιλύονται μερικές από τις συγκρούσεις που αναφέρονται στο βήμα 1.

3) Συγχώνευση όψεων: το καθολικό σχήμα δημιουργείται με συγχώνευση των επιμέρους σχημάτων. Αντίστοιχες έννοιες, που εμφανίζονται σε περισσότερα του ενός σχήματα, παριστάνονται μόνο μια φορά στο καθολικό σχήμα, και καθορίζονται οι απεικονίσεις μεταξύ όψεων και καθολικού σχήματος.

4) Αναδόμηση: ως τελευταίο προαιρετικό βήμα, το καθολικό σχήμα μπορεί να αναλυθεί και να αναδομηθεί για να απομακρυνθούν πλεονασμοί ή περιττή πολυπλοκότητα.

Μερικές από τις ιδέες αυτές απεικονίζονται από το μάλλον απλό παράδειγμα των εικόνων 3.4 και 3.5. Στην εικόνα 3.4 συγχωνεύονται δύο όψεις για να δημιουργηθεί μια βιβλιογραφική βάση δεδομένων. Κατά τον εντοπισμό των αντιστοιχιών μεταξύ δυο όψεων, ανακαλύπτουμε ότι τα ΕΡΕΥΝΗΤΗΣ και ΣΥΓΓΡΑΦΕΑΣ είναι συνώνυμα (όσον αφορά τη βάση δεδομένων), όπως και τα ΣΥΝΕΙΣΦΕΡΕΙ και ΕΓΡΑΨΕ. Επιπλέον, αποφασίζουμε να τροποποιήσουμε τη ΟΨΗ1 για να συμπεριλάβει και το ΑΡΘΡΟ το ΘΕΜΑ, όπως φαίνεται στην Εικόνα 3.4, ώστε να προσαρμοσθεί με την ΟΨΗ2. η Εικόνα 3.5 δείχνει το αποτέλεσμα της συγχώνευσης των ΤΡΟΠΟΠΟΙΗΜΕΝΗ ΟΨΗ1 και ΟΨΗ2. συγχωνεύονται οι συσχετίσεις ΑΝΗΚΕΙ και ΕΓΡΑΨΕ, όπως και οι τύποι οντοτήτων ΣΥΓΓΡΑΦΕΑΣ και ΘΕΜΑ. Γενικεύουμε επίσης τους τύπους οντοτήτων ΑΡΘΡΟ και ΒΙΒΛΙΟ στον τύπο οντοτήτων ΕΚΔΟΣΗ, με το κοινό γνώρισμα τους Τίτλος. Το γνώρισμα Εκδότης απαντάται μόνο στο τύπο οντοτήτων ΒΙΒΛΙΟ, ενώ το γνώρισμα Μέγεθος και ο τύπος συσχετίσεων ΔΗΜΟΣΙΕΥΘΗΚΕ απαντώνται μόνο στο ΑΡΘΡΟ.

Έχουν προταθεί διάφορες στρατηγικές για τη διαδικασία ενοποίησης των όψεων. Όπως φαίνεται στην Εικόνα 3.6, σ' αυτές περιλαμβάνονται και οι κάτωθι:

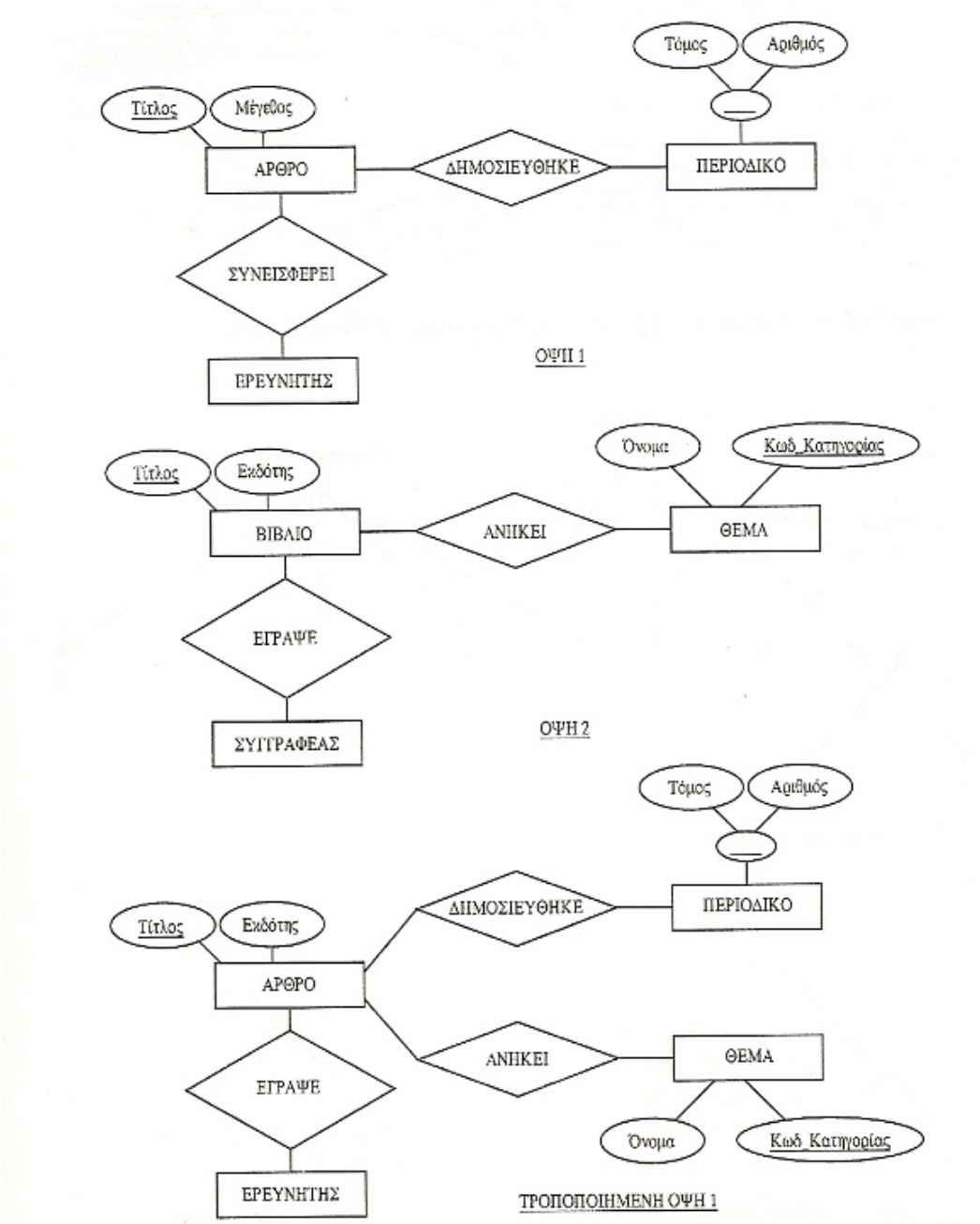
1) Δυαδική κλιμακωτή ενοποίηση: ενοποιούνται πρώτα δυο σχήματα που μοιάζουν αρκετά. Στη συνέχεια το αποτέλεσμα ενοποιείται με ένα άλλο σχήμα και η διαδικασία επαναλαμβάνεται μέχρι να ενοποιηθούν όλα τα σχήματα. Η σειρά

ενοποίησης των σχημάτων μπορεί να βασισθεί σε κάποιο μέτρο ομοιότητας των σχημάτων. Η πρακτική αυτή είναι κατάλληλη για χειρωνακτική ενοποίηση επειδή ακολουθεί βηματική προσέγγιση.

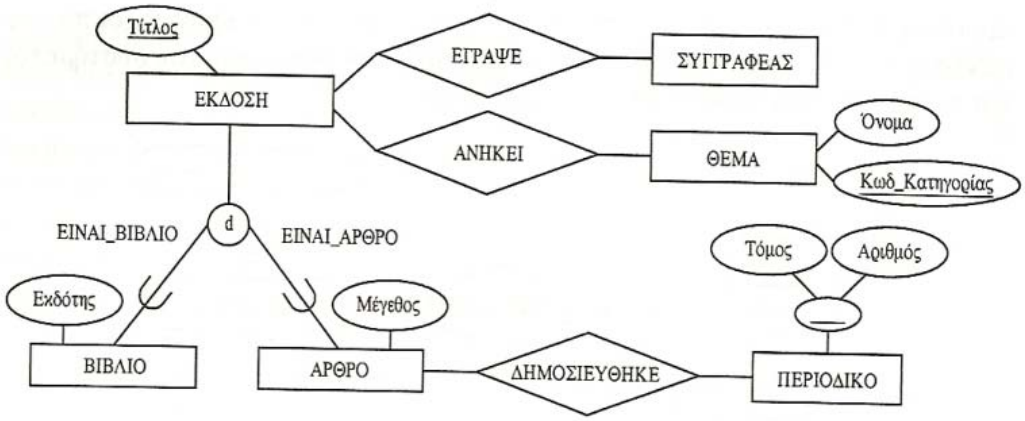
2) Ενοποίηση Ν- τάξεως: όλες οι όψεις ενοποιούνται σε μία φάση, μετά από ανάλυση και προσδιορισμό των αντιστοιχιών τους. Η στρατηγική αυτή απαιτεί υπολογιστικά εργαλεία για μεγάλης κλίμακας σχεδιαστικά προβλήματα.

3) Δυαδική ισοζυγισμένη στρατηγική: πρώτα ενοποιούνται ζεύγη σχημάτων στη συνέχεια τα αποτελέσματα ενοποιούνται σε ζεύγη και η διαδικασία επαναλαμβάνεται μέχρι να καταλήξουμε σε ένα καθολικό σχήμα.

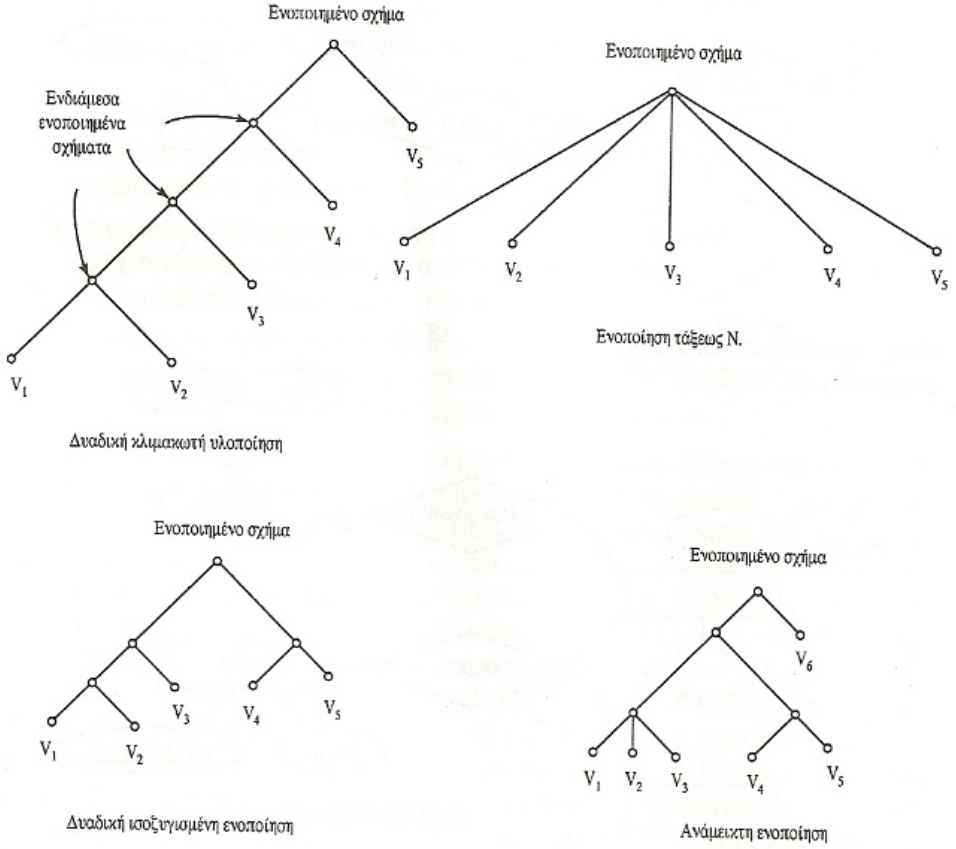
4) Ανάμεικτη στρατηγική: αρχικά τα σχήματα χωρίζονται σε ομάδες με βάση την ομοιότητά τους και κάθε ομάδα ενοποιείται ξεχωριστά. Τα ενδιάμεσα σχήματα ομαδοποιούνται ξανά και ενοποιούνται κ.ο.κ.



Εικόνα 3.4: Τροποποίηση όψεων ώστε να καταστούν συμβατές, πριν την ενοποίηση



Εικόνα 3.5: Ενοποιημένο σχήμα, μετά την συγχώνευση των όψεων 1 και 2.



Εικόνα 3.6: Διαφορετικές στρατηγικές για ενοποίηση όψεων

Φάση 2β: Σχεδιασμός Δοσοληψιών.

Ο σκοπός της φάσης 2β, που εκτελείται παράλληλα με τη φάση 2α, είναι να σχεδιαστούν τα χαρακτηριστικά γνωστών δοσοληψιών κατά τρόπο ανεξάρτητο από κάποιο συγκεκριμένο ΣΔΒΔ. Όταν σχεδιάζεται ένα σύστημα βάσης δεδομένων, οι σχεδιαστές είναι ενήμεροι για πολλές γνωστές εφαρμογές (ή δοσοληψίες) που θα εκτελούνται στη βάση δεδομένων μόλις αυτή υλοποιηθεί. Ένα σημαντικό μέρος του σχεδιασμού της βάσης δεδομένων είναι να προσδιορισθούν τα λειτουργικά χαρακτηριστικά αυτών των δοσοληψιών όσο το δυνατόν νωρίτερα στη διαδικασία σχεδιασμού. Αυτό εξασφαλίζει ότι το σχήμα της βάσης δεδομένων θα περιλαμβάνει όλες τις πληροφορίες που απαιτούνται απ' αυτές τις δοσοληψίες. Επιπλέον, η γνώση

της σχετικής σπουδαιότητας των διαφόρων δοσοληψιών και της αναμενόμενης συχνότητας ενοποίησης τους παίζει αποφασιστικό ρόλο στο φυσικό σχεδιασμό της βάσης δεδομένων (φάση 5). Συνήθως, μόνο μερικές δοσοληψίες της βάσης δεδομένων είναι γνωστές κατά τη φάση του σχεδιασμού. Μετά την υλοποίηση του συστήματος της βάσης δεδομένων, προσδιορίζονται και υλοποιούνται συνεχώς νέες δοσοληψίες. Ωστόσο, οι πιο σημαντικές δοσοληψίες είναι συνήθως γνωστές πριν την υλοποίηση του συστήματος και πρέπει να προσδιορισθούν στα πρώτα στάδια.

Μια συνηθισμένη τεχνική προσδιορισμού των δοσοληψιών σε εννοιολογικό επίπεδο είναι να προσδιορίζεται η είσοδος και η έξοδός τους και η λειτουργική συμπεριφορά τους. Με τον προσδιορισμό των δεδομένων εισόδου, των δεδομένων εξόδου και της εσωτερικής λειτουργικής ροής ελέγχου, οι σχεδιαστές μπορούν να ορίσουν μια δοσοληψία με εννοιολογικό και ανεξάρτητο συστήματος τρόπο. Οι δοσοληψίες συνήθως ομαδοποιούνται σε τρεις κατηγορίες: δοσοληψίες ανάκτησης, δοσοληψίες ενημέρωσης και ανάμεικτες δοσοληψίες. Οι δοσοληψίες ανάκτησης χρησιμοποιούνται για ανάκτηση δεδομένων, προκειμένου για παρουσίαση σε οθόνη ή για παραγωγή αναφορών. Οι δοσοληψίες ενημέρωσης χρησιμοποιούνται για την εισαγωγή νέων δεδομένων ή την τροποποίηση υπαρχόντων δεδομένων στη βάση δεδομένων. Οι ανάμεικτες δοσοληψίες χρησιμοποιούνται για πιο πολύπλοκες εφαρμογές που εκτελούν κάποια ανάκτηση και κάποια ενημέρωση. Για παράδειγμα, ας υποθέσουμε ότι σχεδιάζουμε μια βάση δεδομένων κρατήσεων μια αεροπορικής εταιρείας. Ένα παράδειγμα δοσοληψίας ανάκτησης είναι η εύρεση όλων των πρωινών πτήσεων μεταξύ δυο πόλεων σε μια συγκεκριμένη ημερομηνία. Ένα παράδειγμα δοσοληψίας ενημέρωσης είναι η κράτηση μιας θέσης σε συγκεκριμένη πτήση. Μια ανάμεικτη δοσοληψία μπορεί να βρει πρώτα κάποια δεδομένα, όπως την κράτηση ενός πελάτη σε κάποια πτήση και στη συνέχεια να ενημερώσει τη βάση δεδομένων, π.χ. ακυρώνοντας την κράτηση με διαγραφή της καταχώρισης.

Διάφορες τεχνικές για τον προσδιορισμό απαιτήσεων περιλαμβάνουν συμβολισμό για τον προσδιορισμό διαδικασιών, οι οποίες είναι πιο πολύπλοκες λειτουργίες και είναι δυνατόν να αποτελούνται από διάφορες δοσοληψίες. Ο σχεδιασμός των δοσοληψιών είναι τόσο σημαντικός όσο και ο σχεδιασμός του σχήματος. Δυστυχώς, πολλές σημερινές μεθοδολογίες σχεδιασμού δίνουν έμφαση στον έναν ή τον άλλο. Θα πρέπει κανείς να εκτελέσει παράλληλα τις φάσεις 2α και 2β, χρησιμοποιώντας βρόχους ανάδρασης, μέχρι να καταλήξει σε ένα σταθερό σχεδιασμό σχήματος και δοσοληψιών.

3.4.3 Φάση 3: Επιλογή ενός ΣΔΒΔ

Πλήθος παραγόντων παίζουν αποφασιστικό ρόλο στην επιλογή ενός ΣΔΒΔ. Μερικοί παράγοντες είναι τεχνικοί, άλλοι είναι οικονομικοί και άλλοι σχετίζονται με τις πολιτικές του οργανισμού. Οι τεχνικοί παράγοντες αφορούν την καταλληλότητα του ΣΔΒΔ για το συγκεκριμένο έργο. Θέματα που πρέπει να εξετασθούν εδώ είναι ο τύπος του ΣΔΒΔ (σχεσιακό, δικτυωτό, ιεραρχικό, αντικειμενοστραφές, άλλο), οι δομές αποθήκευσης και οι διαδρομές προσπέλασης που υποστηρίζει το ΣΔΒΔ, οι διαθέσιμες διεπαφές για χρήστες και προγραμματιστές, οι τύποι γλωσσών ερωτήσεων υψηλού επιπέδου κ.ο.κ..

Στο εδάφιο αυτό εστιάζουμε στην εξέταση των οικονομικών και οργανωτικών παραγόντων που επηρεάζουν την επιλογή του ΣΔΒΔ. Τα ακόλουθα έξοδα πρέπει να εξετασθούν όταν επιλέγεται ένα ΣΔΒΔ:

1. Κόστος απόκτησης λογισμικού: είναι το κόστος αγοράς του λογισμικού και συμπεριλαμβάνει τις γλώσσες που θα αγοραστούν, τις διάφορες διεπαφές όπως φόρμες και οθόνες, επιλογές ανάκαμψης και τήρησης εφεδρικών αντιγράφων, ειδικές μεθόδους προσπέλασης και τεκμηρίωση. Πρέπει να επιλεγεί η σωστή έκδοση ΣΔΒΔ για ένα συγκεκριμένο λειτουργικό σύστημα.
2. Κόστος συντήρησης: είναι το διαρκές κόστος για λήψη υπηρεσιών συντήρησης από τον κατασκευαστή και για την αναβάθμιση σε νέες εκδόσεις, όταν αυτές κυκλοφορούν.
3. Κόστος απόκτησης υλικού: Μπορεί να χρειασθεί νέο υλικό όπως επιπλέον μνήμη, τερματικά, μονάδες δίσκων ή εξειδικευμένη αποθήκευση για το ΣΔΒΔ.
4. Κόστος δημιουργίας βάσης δεδομένων και μετατροπών: είναι το κόστος είτε για τη δημιουργία του συστήματος της βάσης δεδομένων από την αρχή, είτε για τη μετατροπή του υπάρχοντος συστήματος στο νέο λογισμικό ΣΔΒΔ. Στην τελευταία περίπτωση συνηθίζεται να λειτουργεί το υπάρχον σύστημα παράλληλα με το νέο σύστημα, μέχρι να υλοποιηθούν και να ελεγχθούν πλήρως όλες οι νέες εφαρμογές. Το κόστος αυτό είναι δύσκολο να προβλεφθεί και συχνά υποτιμάται.
5. Κόστος προσωπικού: η εγκατάσταση λογισμικού ΣΔΒΔ για πρώτη φορά σε έναν οργανισμό συχνά συνοδεύεται από αναδιοργάνωση του τμήματος επεξεργασίας δεδομένων. Στις περισσότερες εταιρείες δημιουργούνται νέες θέσεις προσωπικού και διαχειριστή της βάσης δεδομένων (ΔΒΔ).
6. Κόστος εκπαίδευσης: επειδή τα ΣΔΒΔ συχνά είναι πολύπλοκα συστήματα, πρέπει συνήθως το προσωπικό να εκπαιδευτεί στη χρήση και τον προγραμματισμό του ΣΔΒΔ.
7. Λειτουργικό κόστος: το κόστος συνεχούς λειτουργίας του συστήματος βάσης δεδομένων δεν εκτιμάται όταν εξετάζονται εναλλακτικές λύσεις, διότι ο οργανισμός θα το επωμισθεί ανεξαρτήτως του ΣΔΒΔ που θα επιλεχθεί.

Τα πλεονεκτήματα που προκύπτουν από την απόκτηση ενός ΣΔΒΔ δεν είναι εύκολο να μετρηθούν και να προσδιορισθούν ποσοτικά. Ένα ΣΔΒΔ έχει μερικά σαφή πλεονεκτήματα σε σχέση με τα παραδοσιακά συστήματα αρχείων, όπως εύκολη χρήση, αυξημένη διαθεσιμότητα των δεδομένων και ταχύτερη προσπέλαση στις πληροφορίες. Πιο απτά πλεονεκτήματα περιλαμβάνουν το μειωμένο κόστος ανάπτυξης εφαρμογών, το μειωμένο πλεονασμό δεδομένων και τον καλύτερο έλεγχο και ασφάλεια. Με βάση την ανάλυση πλεονεκτημάτων- κόστους, ένας οργανισμός πρέπει να αποφασίσει πότε θα στραφεί σε ΣΔΒΔ. Η κίνηση αυτή γενικά καθοδηγείται από τους ακόλουθους παράγοντες:

- Πολυπλοκότητα δεδομένων: όταν οι συσχετίσεις των δεδομένων γίνονται πολύπλοκες, γίνεται αισθητή η ανάγκη για ΣΔΒΔ.
- Διαμοιρασμός μεταξύ εφαρμογών: όσο περισσότερο διαμερίζονται οι εφαρμογές τα δεδομένα, τόσο περισσότερος πλεονασμός υπάρχει μεταξύ των αρχείων και επομένως εμφανίζεται ανάγκη για ένα ΣΔΒΔ.
- Δυναμική ανάπτυξη και αύξηση των δεδομένων: αν τα δεδομένα αλλάζουν συνεχώς, είναι ευκολότερο να αντιμετωπισθούν αυτές οι αλλαγές με χρήση ΣΔΒΔ απ’ ότι με χρήση συστήματος αρχείων.
- Συχνότητα μη τυποποιημένων ερωτήσεων στα δεδομένα: τα συστήματα αρχείων δεν είναι κατάλληλα για μη τυποποιημένη ανάκτηση δεδομένων.

- Όγκος δεδομένων και αναγκαιότητα ελέγχου: ο μεγάλος όγκος των δεδομένων και η ανάγκη για έλεγχό τους, μερικές φορές καθιστούν αναγκαίο ένα ΣΔΒΔ.

Τέλος, διάφοροι οικονομικοί και οργανωτικοί παράγοντες επηρεάζουν την επιλογή ενός ΣΔΒΔ, σε σχέση με ένα άλλο:

1. Δομή των δεδομένων: αν τα δεδομένα που θα αποθηκευθούν στη βάση δεδομένων ακολουθούν ιεραρχική δομή, πρέπει να εξετασθεί η περίπτωση ενός ΣΔΒΔ ιεραρχικού τύπου. Για δεδομένα με πολλές συσχετίσεις μεταξύ τους, ένα δικτυωτό ή ένα σχεσιακό σύστημα μπορεί να είναι πιο κατάλληλο. Η σχεσιακή τεχνολογία κερδίζει διαρκώς σε δημοτικότητα. Για σύνθετες δομές και σύνθετους τύπους δεδομένων, τα αντικειμενοστραφή συστήματα μπορεί να είναι κατάλληλα.
2. Εξοικείωση του προσωπικού με το σύστημα: αν το προσωπικό μηχανογράφησης σε έναν οργανισμό είναι εξοικειωμένο με ένα συγκεκριμένο ΣΔΒΔ, μπορεί να προτιμηθεί αυτό, προκειμένου να ελαττωθεί το κόστος εκπαίδευσης και ο χρόνος εκμάθησης.
3. Διαθεσιμότητα των υπηρεσιών του κατασκευαστή: η ύπαρξη δυνατότητας παροχής υπηρεσιών από τον κατασκευαστή είναι επιθυμητή, για να διευκολυνθεί η αντιμετώπιση προβλημάτων με το σύστημα. Η μετάπτωση από περιβάλλον χωρίς ΣΔΒΔ σε ένα με ΣΔΒΔ είναι γενικά μεγάλο εγχείρημα και απαιτεί βοήθεια από τον κατασκευαστή στην αρχή.

Πριν αγορασθεί ένα ΣΔΒΔ, ο οργανισμός πρέπει να λάβει υπόψη τη σύνθεση υλικού / λογισμικού που απαιτείται για να λειτουργήσει το ΣΔΒΔ και τη μεταφερσιμότητά του σε διαφορετικούς τύπους υλικού. Πολλά εμπορικά ΣΔΒΔ σήμερα σχεδιάζονται ως ολικές λύσεις στις ανάγκες επεξεργασίας πληροφοριών και διαχείρισης πληροφοριακών πόρων στους οργανισμούς. Οι περισσότεροι κατασκευαστές ΣΔΒΔ προσφέρουν τα προϊόντα τους με τις κατωτέρω επιλογές ή ενσωματωμένα χαρακτηριστικά, που συχνά αναφέρονται ως 4GL (γλώσσες τέταρτης γενιάς) από τους κατασκευαστές:

- Επεξεργαστές κειμένου και εργαλεία πλοήγησης (browsers).
- Γεννήτριες αναφορών και βοηθητικά προγράμματα για παραγωγή λιστών.
- Λογισμικό επικοινωνιών (συχνά καλούνται επόπτες τηλεπεξεργασίας (teleprocessing monitors))
- Ευκολίες εισαγωγής και παρουσίασης δεδομένων όπως φόρμες, οθόνες και μενού με αυτοματοποιημένες ευκολίες επεξεργασίας.
- Γραφικά εργαλεία σχεδιασμού.

Σε μερικές περιπτώσεις μπορεί να μην ενδείκνυται η χρήση ενός ΣΔΒΔ αλλά μπορεί να είναι προτιμότερο να αναπτυχθεί λογισμικό για τις εφαρμογές του οργανισμού. Αυτό μπορεί να συμβεί αν οι εφαρμογές είναι καλά ορισμένες και είναι όλες γνωστές από την αρχή. Στην περίπτωση αυτή, ένα σύστημα ένα σύστημα σχεδιασμένο και προσαρμοσμένο για τον οργανισμό μπορεί να είναι κατάλληλο για υλοποίηση γνωστών εφαρμογών κατά τον πιο αποδοτικό τρόπο. Ωστόσο, στις περισσότερες περιπτώσεις, νέες εφαρμογές που δεν είχαν προβλεφθεί κατά το σχεδιασμό εμφανίζονται μετά την υλοποίηση του συστήματος. Γι' αυτόν ακριβώς το λόγο, τα ΣΔΒΔ έχουν γίνει πολύ δημοφιλή: διευκολύνουν την ενσωμάτωση νέων εφαρμογών στο υπάρχον σύστημα χωρίς σημαντικές αλλαγές.

3.4.4 Φάση 4: Απεικόνιση στο Μοντέλο Δεδομένων

Η επόμενη φάση του σχεδιασμού βάσεων δεδομένων είναι να δημιουργηθούν ένα εννοιολογικό σχήμα και εξωτερικά σχήματα στο μοντέλο δεδομένων του επιλεχθέντος ΣΔΒΔ. Αυτό γίνεται απεικονίζοντας το εννοιολογικό και εξωτερικό σχήμα, που παρήχθησαν στη φάση 2α από το μοντέλο δεδομένων υψηλού επιπέδου, στο μοντέλο δεδομένων του ΣΔΒΔ. Η απεικόνιση μπορεί να γίνει σε δυο στάδια:

1. Απεικόνιση ανεξάρτητη του συστήματος: στο βήμα αυτό, η απεικόνιση στο μοντέλο δεδομένων του ΣΔΒΔ δεν λαμβάνει υπόψη συγκεκριμένα χαρακτηριστικά ή ειδικές περιπτώσεις που έχουν σχέση με την υλοποίηση του μοντέλου δεδομένων στο συγκεκριμένο ΣΔΒΔ.
2. Προσαρμογή των σχημάτων σε συγκεκριμένο ΣΔΒΔ: τα ΣΔΒΔ υλοποιούν ένα μοντέλο δεδομένων χρησιμοποιώντας συγκεκριμένους τρόπους μοντελοποίησης και περιορισμούς. Μπορεί να χρειαστεί να προσαρμοστούν τα σχήματα που παρήχθησαν στο βήμα 1, έτσι ώστε να συμφωνούν με τα χαρακτηριστικά υλοποίησης μοντέλων δεδομένων που χρησιμοποιούνται στο επιλεγμένο ΣΔΒΔ.

Το αποτέλεσμα αυτής της φάσης πρέπει να είναι εντολές ΓΟΔ στη γλώσσα του επιλεγμένου ΣΔΒΔ, οι οποίες ορίζουν το εννοιολογικό και τα εξωτερικά σχήματα του συστήματος της βάσης δεδομένων. Ωστόσο, αν οι εντολές της ΓΟΔ περιλαμβάνουν μερικές παραμέτρους φυσικού σχεδιασμού, ο πλήρης ορισμός με εντολές ΓΟΔ θα παραχθεί μετά την ολοκλήρωση της φάσης του φυσικού σχεδιασμού της βάσης δεδομένων. Πολλά αυτόματα εργαλεία σχεδιασμού CASE (computer- assisted software engineering) μπορούν να δημιουργήσουν εντολές ΓΟΔ για εμπορικά συστήματα από έναν εννοιολογικό σχεδιασμό σχήματος.

3.4.5 Φάση 5: Φυσικός Σχεδιασμός Βάσεων Δεδομένων

Ο φυσικός σχεδιασμός βάσεων δεδομένων είναι η διαδικασία επιλογής συγκεκριμένων δομών αποθήκευσης και διαδρομών προσπέλασης για τα αρχεία της βάσης δεδομένων, προκειμένου να επιτευχθεί καλή απόδοση για τις διάφορες εφαρμογές της. Κάθε ΣΔΒΔ παρέχει μια ποικιλία από επιλογές για οργάνωση αρχείων και διαδρομές προσπέλασης. Αυτές συνήθως περιλαμβάνουν διάφορους τύπους ευρετηριασμού, δημιουργίας συστάδων με σχετιζόμενες εγγραφές σε μπλοκ δίσκου, χρήση δεικτών για σύνδεση εγγραφών που σχετίζονται και διάφορους τύπους κατακερματισμού. Μόλις επιλεγεί ένα ΣΔΒΔ, η διαδικασία φυσικού σχεδιασμού της βάσης δεδομένων περιορίζεται στην επιλογή, μεταξύ των δυνατοτήτων που προσφέρει το συγκεκριμένο ΣΔΒΔ, των πιο κατάλληλων δομών για τα αρχεία της βάσης δεδομένων. Στα εδάφιο αυτό δίνουμε οδηγίες για τις αποφάσεις που πρέπει να ληφθούν για το φυσικό σχεδιασμό σε διάφορους τύπους ΣΔΒΔ. Τα παρακάτω κριτήρια χρησιμοποιούνται συχνά για να καθοδηγήσουν τη διαδικασία φυσικού σχεδιασμού της βάσης δεδομένων:

1. Χρόνος απόκρισης: είναι ο χρόνος που μεσολαβεί μεταξύ της υποβολής μιας δοσοληψίας για εκτέλεση και λήψης της απάντησης. Το τμήμα του χρόνου απόκρισης που εξαρτάται από το ΣΔΒΔ επηρεάζεται σημαντικά από το χρόνο προσπέλασης στα δεδομένα της βάσης που χρησιμοποιούνται από τη δοσοληψία. Ο χρόνος απόκρισης επηρεάζεται επίσης από παράγοντες που δεν βρίσκονται υπό τον έλεγχο του ΣΔΒΔ, όπως ο φόρτος του συστήματος, η χρονοδρομολόγηση από το λειτουργικό σύστημα και οι καθυστερήσεις επικοινωνίας.

2. Χρησιμοποίηση του χώρου: πρόκειται για το μέγεθος του χώρου αποθήκευσης που χρησιμοποιείται από τα αρχεία της βάσης δεδομένων και τις δομές προσπέλασής τους.
3. Ρυθμός διεκπεραίωσης δοσοληψιών: είναι ο μέσος αριθμός δοσοληψιών που μπορεί να επεξεργαστεί το σύστημα της βάσης δεδομένων ανά λεπτό. Πρόκειται για κρίσιμη παράμετρο για συστήματα δοσοληψιών όπως αυτά που χρησιμοποιούνται σε κρατήσεις αεροπορικών εταιρειών και σε τραπεζικά συστήματα. Ο αριθμός διεκπεραίωσης δοσοληψιών πρέπει να μετριέται σε συνθήκες αιχμής στο σύστημα.

Στην τυπική περίπτωση καθορίζονται ως μέρος των απαιτήσεων απόδοσης του συστήματος οι οριακές τιμές για τη μέση και τη χειρότερη περίπτωση, για όλες τις παραπάνω παραμέτρους. Χρησιμοποιούνται αναλυτικές ή πειραματικές τεχνικές, οι οποίες μπορεί να περιλαμβάνουν πρωτοτυποποίηση και προσομοίωση για την εκτίμηση της μέσης και της χειρότερης τιμής για διαφορετικούς φυσικούς σχεδιασμούς, προκειμένου να διαπιστωθεί αν ικανοποιούν τις απαιτήσεις απόδοσης που έχουν τεθεί.

Η απόδοση εξαρτάται από το μέγεθος της εγγραφής και το πλήθος των εγγραφών του αρχείου. Επομένως πρέπει να εκτιμηθούν αυτές οι παράμετροι για κάθε αρχείο. Επιπλέον πρέπει να εκτιμήσουμε τα πρότυπα (patterns) ενημέρωσης και ανάκτησης όλων των δοσοληψιών συμβολικά για κάθε αρχείο. Τα γνωρίσματα που χρησιμοποιούνται για την επιλογή εγγραφών θα πρέπει να έχουν πρωτεύουσες διαδρομές προσπέλασης και θα πρέπει να κατασκευαστούν γι' αυτά δευτερεύοντα ευρετήρια. Κατά το φυσικό σχεδιασμό, πρέπει να ληφθούν υπόψη και εκτιμήσεις για την αύξηση του μεγέθους του αρχείου είτε λόγω της αύξησης του μεγέθους εγγραφής, καθώς προστίθενται νέα γνωρίσματα, είτε λόγω αύξησης του πλήθους εγγραφών.

Τα αποτέλεσμα της φάσης του φυσικού σχεδιασμού είναι ένας αρχικός προσδιορισμός των δομών αποθήκευσης και των διαδρομών προσπέλασης για τα αρχεία της βάσης δεδομένων. Είναι σχεδόν πάντα απαραίτητο να ρυθμιστεί (time) ο σχεδιασμός, με βάσή την απόδοση που παρατηρείται μετά την υλοποίηση του συστήματος της βάσης δεδομένων. Τα περισσότερα συστήματα περιλαμβάνουν ένα βοηθητικό πρόγραμμα παρακολούθησης για συλλογή στατιστικών απόδοσης, τα οποία καταχωρούνται στο κατάλογο του συστήματος ή στο λεξικό δεδομένων για μελλοντική χρήση. Αυτά περιλαμβάνουν στατιστικές για το πλήθος των κλήσεων προκαθορισμένων δοσοληψιών ή ερωτήσεων, μετρήσεις εισόδων/ εξόδων για αρχεία, πλήθος σελίδων αρχείου ή εγγραφών ευρετηρίου και συχνότητα χρήσης ευρετηρίου. Καθώς οι απαιτήσεις του συστήματος της βάσης δεδομένων αλλάζουν, συχνά καθίσταται αναγκαία η αναδιοργάνωση μερικών αρχείων με την κατασκευή νέων ευρετηρίων, ή με την αλλαγή πρωτευουσών μεθόδων προσπέλασης.

3.4.6 Φάση 6: Υλοποίηση του Συστήματος Βάσης Δεδομένων

Μετά την ολοκλήρωση του λογικού και του φυσικού σχεδιασμού, μπορούμε να υλοποιήσουμε το σύστημα της βάσης δεδομένων. Εντολές της ΓΟΔ (γλώσσα ορισμού δεδομένων) και της ΓΟΑ (γλώσσα ορισμού αποθήκευσης) του επιλεγμένου ΣΔΒΔ μεταγλωττίζονται και χρησιμοποιούνται για τη δημιουργία των σχημάτων της βάσης δεδομένων και των (κενών) αρχείων της βάσης δεδομένων. Στη συνέχεια η βάση δεδομένων μπορεί να φορτωθεί με δεδομένα. Αν τα δεδομένα πρέπει να μετατραπούν από προηγούμενο υπολογιστικό σύστημα, μπορεί να χρειασθούν ρουτίνες μετατροπής για να μορφοποιηθούν τα δεδομένα για φόρτωμα στη βάση δεδομένων.

Οι δοσοληψίες στη βάση δεδομένων πρέπει να υλοποιηθούν στη φάση αυτή από τους προγραμματιστές εφαρμογών. Εξετάζονται οι εννοιολογικοί ορισμοί των δοσοληψιών και συγγράφεται και ελέγχεται αντίστοιχος κώδικας προγραμμάτων με εμφυτευμένες εντολές (γλώσσα χειρισμού δεδομένων- DML). Μόλις ετοιμασθούν οι δοσοληψίες και φορτωθούν τα δεδομένα στη βάση δεδομένων, ολοκληρώνεται η φάση σχεδιασμού και υλοποίησης και αρχίζει η φάση λειτουργίας του συστήματος της βάσης δεδομένων.

3.5 Παράγοντες που επηρεάζουν το φυσικό σχεδιασμό βάσεων δεδομένων

Στο εδάφιο αυτό εξετάζουμε πράτα τους παράγοντες του φυσικού σχεδιασμού που επηρεάζουν την απόδοση των εφαρμογών και των δοσοληψιών.

Ο φυσικός σχεδιασμός είναι μια δραστηριότητα με στόχο όχι μόνον την κατάλληλη δόμηση των δεδομένων στον αποθηκευτικό χώρο, αλλά και την εξασφάλιση καλής απόδοσης μέσω αυτής της δόμησης. Για κάποιο εννοιολογικό σχήμα υπάρχουν πολλοί εναλλακτικοί φυσικοί σχεδιασμοί σε κάποιο ΣΔΒΔ. Δεν είναι δυνατόν να ληφθούν σημαντικές αποφάσεις φυσικού σχεδιασμού και να γίνουν αναλύσεις απόδοσης, αν δεν ξέρουμε τις ερωτήσεις, τις δοσοληψίες και τις εφαρμογές που αναμένεται να εκτελεσθούν στη βάση δεδομένων. Πρέπει να αναλύσουμε αυτές τις εφαρμογές, τις αναμενόμενες συχνότητες εκτέλεσης, τους περιορισμούς όσον αφορά το χρόνο εκτέλεσης και την αναμενόμενη συχνότητα των ενημερώσεων. Στα επόμενα εξετάζουμε καθέναν από αυτούς τους παράγοντες.

Ανάλυση των Ερωτήσεων και Δοσοληψιών στη Βάση Δεδομένων. Πριν προχωρήσουμε στο φυσικό σχεδιασμό μιας βάσης δεδομένων, πρέπει να έχουμε κατανοήσει καλά πριν προτιθέμενη χρήση της βάσης δεδομένων έχοντας ορίσει, σε υψηλό επίπεδο, τις ερωτήσεις και τις δοσοληψίες που περιμένουμε να εκτελεσθούν στη βάση δεδομένων. Για κάθε ερώτηση πρέπει να προσδιορίσουμε τα ακόλουθα:

1. Τα αρχεία που θα προσπελασθούν από την ερώτηση.
2. Τα πεδία που συμμετέχουν στις συνθήκες επιλογής.
3. Τα πεδία που συμμετέχουν στις συνθήκες συνένωσης ή συνθήκες σύνδεσης πολλαπλών τύπων εγγραφών για την ερώτηση.
4. Τα πεδία των οποίων οι τιμές θα ανακτηθούν από την ερώτηση.

Τα πεδία που αναφέρονται στα 2 και 3 παραπάνω, είναι υποψήφια για ορισμό δομών προσπέλασης. Για κάθε δοσοληψία ενημέρωσης και για κάθε πράξη ενημέρωσης πρέπει να προσδιορίσουμε τα παρακάτω:

Τα αρχεία που θα ενημερωθούν.

1. Τον τύπο της πράξης ενημέρωσης σε κάθε αρχείο (εισαγωγή, τροποποίηση ή διαγραφή).
2. Τα πεδία που συμμετέχουν σε συνθήκες επιλογής για την πράξη της διαγραφής ή της τροποποίησης.
3. Τα πεδία των οποίων οι τιμές θα αλλάξουν από μια πράξη τροποποίησης.

Τα πεδία που αναφέρονται στο 3 παραπάνω είναι επίσης υποψήφια για δομές προσπέλασης. Από την άλλη πλευρά, στα πεδία που αναφέρονται στο 4 είναι καλό να αποφύγουμε τον ορισμό δομών προσπέλασης, διότι η τροποποίησή τους θα απαιτεί ενημέρωση των δομών αυτών.

Ανάλυση της Αναμενόμενης Συχνότητας Κλήσης Ερωτήσεων και Δοσοληψιών. Εκτός από τον προσδιορισμό των χαρακτηριστικών των αναμενόμενων ερωτήσεων και δοσοληψιών, πρέπει να εξετάσουμε τους αναμενόμενους ρυθμούς κλήσης τους. Αυτές οι πληροφορίες συχνοτήτων, μαζί με πληροφορίες για πεδία που συλλέγονται για κάθε ερώτηση και δοσοληψία, χρησιμοποιούνται για την σύνταξη μιας συγκεντρωτικής λίστας των αναμενόμενων συχνοτήτων χρήσης όλων των ερωτήσεων και των δοσοληψιών. Αυτό εκφράζει την αναμενόμενη συχνότητα χρήσης κάθε πεδίου σε κάθε αρχείο, ως πεδίο επιλογής ή συνένωσης, από όλες τις ερωτήσεις και

τις δοσοληψίες. Γενικά, για μεγάλους όγκους επεξεργασίας, εφαρμόζεται ο άτυπος κανόνας «80-20». Αυτός ο κανόνας λεει ότι κατά προσέγγιση το 80% της επεξεργασίας γίνεται από το 20% μόνον των ερωτήσεων και δοσοληψιών. Επομένως, σε πρακτικές καταστάσεις, σπάνια είναι απαραίτητη η συλλογή λεπτομερών στατιστικών και ρυθμών κλήσης για όλες τις ερωτήσεις και τις δοσοληψίες.

Αρκεί να προσδιορίσουμε το 20% περίπου , αυτό που αφορά τις πιο σημαντικές.

Ανάλυση των Περιορισμών Χρόνου για Ερωτήσεις και Δοσοληψίες. Μερικές ερωτήσεις και δοσοληψίες μπορεί να έχουν αυστηρούς περιορισμούς απόδοσης. Για παράδειγμα, σε κάποια δοσοληψία μπορεί να εμφανίζεται ο περιορισμός ότι στο 95% των περιπτώσεων που καλείται πρέπει να τερματίζει σε 5 δευτερόλεπτα και ότι δεν μπορεί ποτέ να ξεπερνά τα 20 δευτερόλεπτα. Τέτοιοι πρόσθετοι περιορισμοί απόδοσης μπορούν να χρησιμοποιηθούν για να τεθούν περαιτέρω προτεραιότητες στα πεδία που είναι υποψήφια για δευτερεύουσες οργανώσεις. Τα πεδία επιλογής που χρησιμοποιούνται από ερωτήσεις και δοσοληψίες με περιορισμούς χρόνου αποκτούν υψηλότερη προτεραιότητα για δευτερεύουσα οργάνωση.

Ανάλυση των Αναμενόμενων Συχνοτήτων των Πράξεων Ενημέρωσης. Για ένα αρχείο που ενημερώνεται συχνά πρέπει να ορίζεται ο ελάχιστος αριθμός από δευτερεύουσες οργανώσεις, επειδή η ενημέρωση των οργανώσεων αυτών καθυστερεί τις πράξεις ενημέρωσης. Μόλις συλλέξουμε τις ανωτέρω πληροφορίες, μπορούμε να πάρουμε αποφάσεις για το φυσικό σχεδιασμό της βάσης δεδομένων, που κυρίως περιλαμβάνουν τις αποφάσεις για τις δομές αποθήκευσης και τις δευτερεύουσες οργανώσεις για τα αρχεία της βάσης δεδομένων. Σπάνια είναι γνωστές κατά το φυσικό σχεδιασμό μιας βάσης δεδομένων όλες οι ερωτήσεις και οι δοσοληψίες στη βάση δεδομένων. Νέες εφαρμογές που απαιτούν τον προσδιορισμό νέων ερωτήσεων και δοσοληψιών εμφανίζονται συχνά μετά την υλοποίηση του συστήματος της βάσης δεδομένων. Σε τέτοιες περιπτώσεις, συχνά είναι απαραίτητο να τροποποιηθούν αποφάσεις για το φυσικό σχεδιασμό της βάσης δεδομένων, προκειμένου να ενσωματωθούν στο σύστημα οι νέες εφαρμογές. Αυτό λέγεται ρύθμιση (tunning) του φυσικού σχεδιασμού. Αν οι περιορισμοί στους χρόνους απόκρισης για κάποιες δοσοληψίες ή ερωτήσεις δεν ικανοποιούνται, χρειάζονται τροποποιήσεις στον αρχικό σχεδιασμό για να βελτιώσουν την αποτελεσματικότητα των δοσοληψιών.

4 Έννοιες Επεξεργασίας Δοσοληψιών

Στο παρόν κεφάλαιο παρουσιάζουμε την έννοια της ατομικής δοσοληψίας, η οποία χρησιμοποιείται για να αναπαρασταθεί μια λογική ενότητα επεξεργασίας στη βάση δεδομένων. Εξετάζουμε το πρόβλημα του ελέγχου συνδρομικότητας (concurrency control problem), το οποίο εμφανίζεται όταν δημιουργούνται αλληλοπαρεμβολές μεταξύ πολλαπλών δοσοληψιών που υποβάλλονται από διάφορους χρήστες, πράγμα που οδηγεί σε λανθασμένα αποτελέσματα. Εξετάζουμε επίσης τη δυνατότητα ανάκαμψης από αποτυχίες των δοσοληψιών. Το Εδάφιο 4.1 εξετάζει άτυπα γιατί ο έλεγχος συνδρομικότητας και η ανάκαμψη είναι απαραίτητα στοιχεία ενός συστήματος βάσεων δεδομένων. Το Εδάφιο 4.2 εισάγει την έννοια της *ατομικής δοσοληψίας* και εξετάζει πρόσθετες έννοιες που σχετίζονται με την επεξεργασία δοσοληψιών σε συστήματα βάσεων δεδομένων. Το Εδάφιο 4.3 παρουσιάζει τις έννοιες της ατομικότητας (atomicity), της συνέπειας (consistency), της απομόνωσης (isolation) και της διάρκειας (durability) -αυτές είναι γνωστές στη βιβλιογραφία ως *ιδιότητες ACID* και είναι επιθυμητές για τις δοσοληψίες. Το Εδάφιο 4.4 εισάγει την έννοια των *χρονοπρογραμμάτων* (schedules) ή *χρονικών* (histories) εκτέλεσης δοσοληψιών και εξετάζει τη δυνατότητα ανάκαμψης των χρονοπρογραμμάτων. Το Εδάφιο 4.5 εξετάζει την έννοια της σειριοποιησιμότητας (serializability) της συνδρομικής (concurrent) εκτέλεσης δοσοληψιών, η οποία μπορεί να χρησιμοποιηθεί για να ορισθούν οι σωστές ακολουθίες εκτέλεσης ταυτόχρονων δοσοληψιών.

4.1 ΕΙΣΑΓΩΓΗ ΣΤΗΝ ΕΠΕΞΕΡΓΑΣΙΑ ΔΟΣΟΛΗΨΙΩΝ

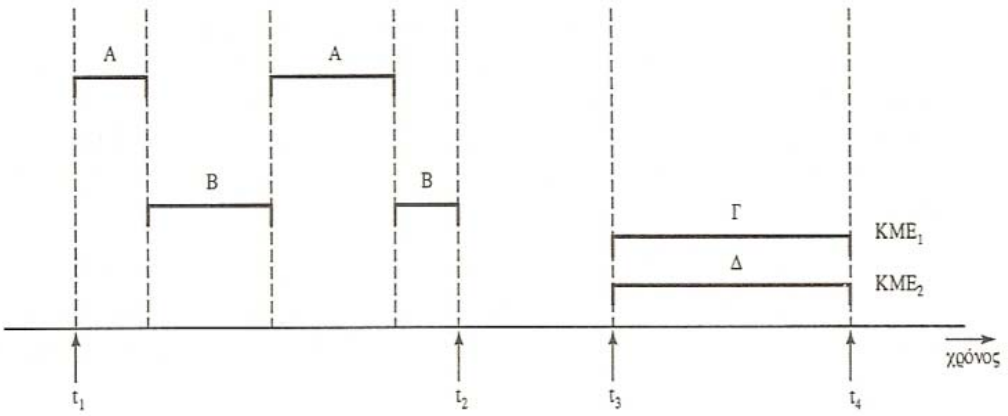
Στο παρόν Εδάφιο εισάγουμε άτυπα τις έννοιες της ταυτόχρονης εκτέλεσης δοσοληψιών και της ανάκαμψης από αποτυχίες δοσοληψιών. Στο Εδάφιο 4.1.1 συγκρίνουμε μονοχρηστικά (single-user) και πολυχρηστικά (multi-user) συστήματα βάσεων δεδομένων και δείχνουμε πώς μπορεί να υλοποιηθεί η συνδρομική εκτέλεση δοσοληψιών σε πολυχρηστικά συστήματα. Το Εδάφιο 4.1.2 παρουσιάζει ένα απλό μοντέλο εκτέλεσης δοσοληψιών που βασίζεται στις πράξεις ανάγνωσης και εγγραφής στη βάση δεδομένων. Το μοντέλο αυτό χρησιμοποιείται για τον τυπικό ορισμό των εννοιών του ελέγχου συνδρομικότητας και της ανάκαμψης. Το Εδάφιο 4.1.3 δείχνει, με άτυπα παραδείγματα, γιατί απαιτούνται τεχνικές ελέγχου συνδρομικότητας σε πολυχρηστικά συστήματα. Τέλος, το Εδάφιο 4.1.4 εξετάζει γιατί απαιτούνται τεχνικές που επιτρέπουν την ανάκαμψη μετά από λάθη, παρουσιάζοντας τους διαφορετικούς λόγους αποτυχίας μιας δοσοληψίας κατά την εκτέλεση.

4.1.1 Μονοχρηστικά Συστήματα Έναντι Πολυχρηστικών Συστημάτων

Ένα κριτήριο διαχωρισμού των συστημάτων βάσεων δεδομένων είναι το πλήθος των χρηστών που μπορούν να χρησιμοποιούν το σύστημα ταυτόχρονα. Ένα ΣΔΒΔ είναι *μονοχρηστικά* αν το πολύ ένας χρήστης μπορεί να χρησιμοποιεί το σύστημα ανά πάσα στιγμή και είναι *πολυχρηστικά* αν πολλοί χρήστες μπορούν να χρησιμοποιούν το σύστημα ταυτόχρονα. Τα μονοχρηστικά ΣΔΒΔ απαντώνται κυρίως σε μερικά συστήματα μικροϋπολογιστών, ενώ τα περισσότερα άλλα ΣΔΒΔ είναι πολυχρηστικά. Για παράδειγμα, ένα σύστημα κρατήσεων μιας αεροπορικής εταιρείας χρησιμοποιείται από εκατοντάδες ταξιδιωτικούς πράκτορες και υπαλλήλους εκδοτηρίων ταυτόχρονα. Συστήματα σε τράπεζες, ασφαλιστικούς πράκτορες,

χρηματιστηριακά γραφεία κ.ο.κ. λειτουργούν με πολλούς χρήστες, οι οποίοι υποβάλλουν ταυτόχρονες δοσοληψίες στο σύστημα.

Οι χρήστες μπορούν να χρησιμοποιούν υπολογιστικά συστήματα ταυτόχρονα, λόγω του *πολυπρογραμματισμού,* ο οποίος επιτρέπει την επεξεργασία πολλών προγραμμάτων (ή δοσοληψιών) την ίδια στιγμή. Αν υπάρχει μόνο μία κεντρική μονάδα επεξεργασίας (ΚΜΕ-CPU), αυτή μπορεί να επεξεργασθεί στην πραγματικότητα μόνο ένα πρόγραμμα ανά πάσα στιγμή. Ωστόσο, τα *λειτουργικά συστήματα πολυπρογραμματισμού* εκτελούν μερικές εντολές από ένα πρόγραμμα, στη συνέχεια αναστέλλουν την εκτέλεση αυτού του προγράμματος και εκτελούν κάποιες εντολές από το επόμενο πρόγραμμα κ.ο.κ. Ένα πρόγραμμα συνεχίζει από το σημείο όπου ανεστάλη όταν έρθει ξανά η σειρά του να χρησιμοποιήσει την κεντρική μονάδα επεξεργασίας. Επομένως η συνδρομική εκτέλεση των προγραμμάτων είναι στην πραγματικότητα *διαπεπλεγμένη* (interleaved) όπως δείχνει η Εικόνα 4.1 (ίδια με την Εικόνα 4.3). Στην εικόνα αυτή παρουσιάζονται δύο προγράμματα Α και Β τα οποία εκτελούνται συνδρομικά, με διαπεπλεγμένο τρόπο. Η διαπλοκή εκτέλεσης διατηρεί απασχολημένη την ΚΜΕ όταν ένα εκτελούμενο πρόγραμμα απαιτεί πράξεις εισόδου / εξόδου, όπως ανάγνωση από τον δίσκο.



Εικόνα 4.1: Διαπεπλεγμένη Έναντι Ταυτόχρονης Συνδρομικότητας

Η ΚΜΕ, αντί να παραμένει αδρανής κατά την εκτέλεση της πράξης εισόδου / εξόδου, μετάγεται στην εκτέλεση κάποιου άλλου προγράμματος.

Αν το υπολογιστικό σύστημα διαθέτει πολλαπλές μονάδες επεξεργασίας, είναι δυνατή η *ταυτόχρονη* επεξεργασία πολλών προγραμμάτων, επιτυγχάνοντας έτσι ταυτόχρονη, αντί διαπεπλεγμένης, συνδρομικότητα. Το μεγαλύτερο μέρος της θεωρίας που αφορά τον έλεγχο συνδρομικότητας σε βάσεις δεδομένων έχει αναπτυχθεί έχοντας υπόψη τη διαπεπλεγμένη συνδρομικότητα, μπορεί όμως να προσαρμοσθεί στην ταυτόχρονη συνδρομικότητα. Στο υπόλοιπο του κεφαλαίου θεωρούμε το *διαπεπλεγμένο μοντέλο ταυτόχρονης εκτέλεσης.*

Σε ένα πολυχρηστικό ΣΔΒΔ, τα δεδομένα που αποθηκεύονται είναι οι πρωτεύοντες πόροι που μπορούν να προσπελασθούν ταυτόχρονα από τα προγράμματα των χρηστών, τα οποία συνεχώς ανακτούν πληροφορίες και ενημερώνουν τη βάση δεδομένων. Η *εκτέλεση ενός προγράμματος* που προσπελαύνει ή τροποποιεί το περιεχόμενο της βάσης δεδομένων καλείται *δοσοληψία* (transaction).

4.1.2 Οι Πράξεις Ανάγνωσης και Εγγραφής μιας Δοσοληψίας

Στην εξέταση του ελέγχου συνδρομικότητας και τεχνικών ανάκαμψης θα θεωρήσουμε δοσοληψίες σε επίπεδο στοιχειωδών δεδομένων και μπλοκ δίσκου. Στο επίπεδο αυτό, οι πράξεις προσπέλασης στη βάση δεδομένων που μπορεί να περιέχει μια δοσοληψία είναι:

- *ανάγνωση_στοιχείου(Χ):* διαβάζει ένα στοιχείο της βάσης δεδομένων που ονομάζεται Χ και το αποθηκεύει σε μια μεταβλητή του προγράμματος. Για απλούστευση των συμβολισμών, υποθέτουμε ότι η *μεταβλητή του προγράμματος ονομάζεται επίσης Χ.*
- *εγγραφή_στοιχείου(Χ):* καταχωρεί την τιμή μιας μεταβλητής προγράμματος Χ στο στοιχείο της βάσης δεδομένων που ονομάζεται Χ.

Η βασική μονάδα μεταφοράς δεδομένων από το δίσκο στην κύρια μνήμη του υπολογιστή είναι ένα μπλοκ. Γενικά, ένα στοιχειώδες δεδομένο θα είναι ένα πεδίο κάποιας εγγραφής στη βάση δεδομένων, αν και είναι δυνατόν να πρόκειται για μεγαλύτερη μονάδα, όπως μια εγγραφή ή ακόμη κι ένα μπλοκ. Η εκτέλεση μιας εντολής ανάγνωση_στοιχείου(Χ) περιλαμβάνει τα ακόλουθα βήματα:

1. Εύρεση της διεύθυνσης του μπλοκ δίσκου που περιέχει το στοιχείο Χ.
2. Αντιγραφή του συγκεκριμένου μπλοκ σε μια ενδιάμεση περιοχή στην κύρια μνήμη (αν το μπλοκ δίσκου δεν βρίσκεται ήδη σε κάποια ενδιάμεση περιοχή στην κύρια μνήμη).
3. Αντιγραφή του στοιχείου Χ από την ενδιάμεση περιοχή στη μεταβλητή του προγράμματος Χ.
4. Η εκτέλεση μιας εντολής εγγραφή_στοιχείου(Χ) περιλαμβάνει τα ακόλουθα βήματα:
5. Εύρεση της διεύθυνσης του μπλοκ δίσκου που περιέχει το στοιχείο Χ.
6. Αντιγραφή του συγκεκριμένου μπλοκ σε μια ενδιάμεση περιοχή στην κύρια μνήμη (αν το μπλοκ δίσκου δεν βρίσκεται ήδη σε κάποια ενδιάμεση περιοχή στην κύρια μνήμη).
7. Αντιγραφή του στοιχείου Χ από τη μεταβλητή του προγράμματος που ονομάζεται Χ στη σωστή του θέση, μέσα στην ενδιάμεση περιοχή της κύριας μνήμης.
8. Αποθήκευση της ενημερωμένης σελίδας στο δίσκο (είτε αμέσως είτε κάποια στιγμή αργότερα).

Το βήμα 4 είναι εκείνο που στην πραγματικότητα ενημερώνει τη βάση δεδομένων στον δίσκο. Σε πολλές περιπτώσεις, αν πρόκειται να γίνουν και άλλες αλλαγές στην ενδιάμεση μνήμη, η σελίδα δεν αποθηκεύεται αμέσως στο δίσκο. Συνήθως, η απόφαση για το πότε θα αποθηκευθεί ένα τροποποιημένο μπλοκ δίσκου που βρίσκεται στην κύρια μνήμη λαμβάνεται από τον διαχειριστή ανάκαμψης ή από το λειτουργικό σύστημα.

Μια δοσοληψία θα περιλαμβάνει πράξεις ανάγνωση_στοιχείου και εγγραφή_στοιχείου για προσπέλαση στη βάση δεδομένων. Η Εικόνα 42 παρουσιάζει ως παραδείγματα δύο πολύ απλές δοσοληψίες. Οι μηχανισμοί ελέγχου συνδρομικότητας και ανάκαμψης αφορούν κυρίως τις εντολές προσπέλασης της βάσης δεδομένων σε μια δοσοληψία.

```
(α) ανάγνωση_στοιχείου(X);
    X:=X-N;
    εγγραφή_στοιχείου(X);
    ανάγνωση_στοιχείου(Y);
    Y:=Y+N;
    εγγραφή_στοιχείου(Y);

(β) ανάγνωση_στοιχείου(X);
    X:=X+M;
    εγγραφή_στοιχείου(X);
```

Εικόνα 4.2: Δύο παραδείγματα δοσοληψιών (α) Δοσοληψία Τ1 (β) Δοσοληψία Τ2

Δοσοληψίες που υποβάλλονται από διάφορους χρήστες μπορούν να εκτελούνται ταυτόχρονα και είναι δυνατόν να προσπελαύνουν και να ενημερώνουν τα ίδια στοιχεία της βάσης δεδομένων. Αν η ταυτόχρονη εκτέλεση δεν είναι ελεγχόμενη, μπορεί να οδηγήσει σε προβλήματα όπως σε *ασυνεπή βάση δεδομένων*. Στο επόμενο Εδάφιο εισάγουμε *άτυπα* τρία από τα προβλήματα που μπορεί να προκύψουν όταν εκτελούνται ταυτόχρονες δοσοληψίες κατά μη ελεγχόμενο τρόπο.

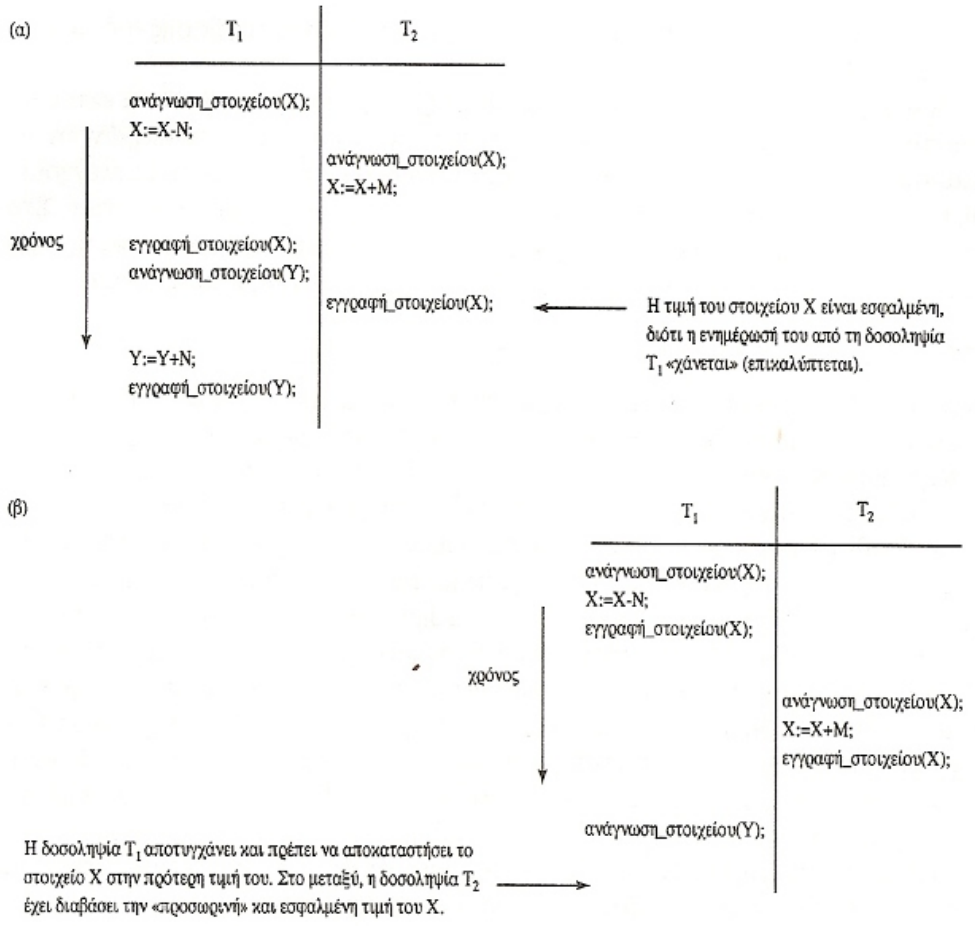
4.1.3 Γιατί είναι απαραίτητος ο Έλεγχος Συνδρομικότητας

Διάφορα προβλήματα μπορούν να προκύψουν όταν εκτελούνται συνδρομικές δοσοληψίες κατά μη ελεγχόμενο τρόπο. Παρουσιάζουμε μερικά από τα προβλήματα αυτά χρησιμοποιώντας ως παράδειγμα μια απλή βάση δεδομένων κρατήσεων μιας αεροπορικής εταιρείας, όπου αποθηκεύεται μία εγγραφή για κάθε πτήση της εταιρείας. Κάθε εγγραφή περιλαμβάνει, μεταξύ άλλων πληροφοριών, το πλήθος των θέσεων που έχουν κρατηθεί στη συγκεκριμένη πτήση. Το πλήθος αυτό αποτελεί ένα *ονοματισμένο στοιχειώδες δεδομένο* (named data item). Η Εικόνα 42(α) παρουσιάζει μια δοσοληψία Τ1 που *ακυρώνει* Ν κρατήσεις από μια πτήση, της οποίας το πλήθος των κρατημένων θέσεων αποθηκεύεται στο στοιχείο της βάσης δεδομένων Χ και *κρατά* τον ίδιο αριθμό θέσεων σε μια άλλη πτήση, της οποίας το πλήθος των κρατημένων θέσεων αποθηκεύεται στο στοιχείο της βάσης δεδομένων Υ. Η Εικόνα 42(β) παρουσιάζει μια απλούστερη δοσοληψία, την Τ2, η οποία *κρατά* Μ θέσεις στην πρώτη πτήση που αναφέρεται στην δοσοληψία Τ1. Για απλοποίηση του παραδείγματος δεν δείχνουμε άλλα τμήματα της δοσοληψίας, όπως ο έλεγχος ύπαρξης επαρκών ελεύθερων θέσεων πριν κρατηθούν επιπλέον θέσεις.

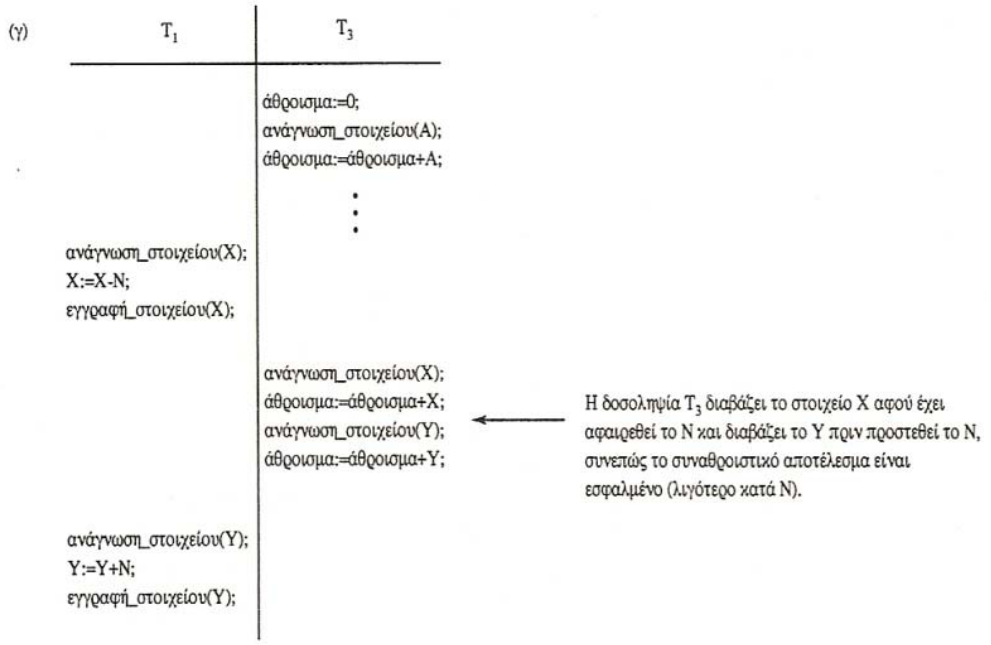
Όταν συγγράφεται ένα πρόγραμμα βάσης δεδομένων, οι αριθμοί των πτήσεων, οι ημερομηνίες τους και το πλήθος των θέσεων που θα κρατηθούν μοντελοποιούνται ως παράμετροι. Επομένως, το ίδιο πρόγραμμα μπορεί να χρησιμοποιηθεί για να εκτελεσθούν πολλές δοσοληψίες, κάθε μία από τις οποίες θα χρησιμοποιεί διαφορετικούς αριθμούς πτήσεων και θέσεων. Σε ό,τι αφορά τον έλεγχο συνδρομικότητας, μια δοσοληψία είναι μία *συγκεκριμένη εκτέλεση* ενός προγράμματος για συγκεκριμένη ημερομηνία, πτήση και πλήθος θέσεων. Στην Εικόνα 4.2(α) και (β), οι δοσοληψίες Τ1 και Τ2 είναι *συγκεκριμένες εκτελέσεις* των προγραμμάτων. Οι εκτελέσεις αυτές αναφέρονται στις πτήσεις των οποίων ο αριθμός των κρατημένων θέσεων αποθηκεύεται στα στοιχειώδη δεδομένα Χ και Υ της βάσης δεδομένων. Εξετάζουμε τώρα τους τύπους των προβλημάτων που μπορεί να αντιμετωπίσουμε με αυτές τις δύο δοσοληψίες.

Το Πρόβλημα της Απώλειας Ενημερώσεων. Αυτό εμφανίζεται όταν οι πράξεις δύο δοσοληψιών που προσπελαύνουν τα ίδια στοιχειώδη δεδομένα, διαπλέκονται κατά τρόπο που καθιστά την τιμή κάποιου στοιχειώδους δεδομένου λανθασμένη. Ας

υποθέσουμε ότι οι δοσοληψίες Τ1 και Τ2 υποβάλλονται περίπου την ίδια στιγμή και ότι οι πράξεις τους διαπλέκονται από το λειτουργικό σύστημα όπως φαίνεται στην Εικόνα 4.3(α).



Εικόνα 4.3: Μερικά προβλήματα που εμφανίζονται όταν δεν υπάρχει έλεγχος ταυτόχρονης προσπέλασης (α) Το πρόβλημα της απώλειας ενημερώσεων. (β) Το πρόβλημα της προσωρινής ενημέρωσης. (γ) Το πρόβλημα της λανθασμένης συνάθροισης



Στην περίπτωση αυτή, η τελική τιμή του στοιχείου X είναι λανθασμένη διότι η δοσοληψία T2 διαβάζει το X πριν η δοσοληψία T1 αλλάξει την τιμή του στη βάση των δεδομένων και επομένως *η τροποποιημένη τιμή που προκύπτει από την* T1 *χάνεται*. Για παράδειγμα, αν στην αρχή ισχύει X=80 (αρχικά υπήρχαν 80 κρατήσεις στην πτήση), N=5 (η T1 ακυρώνει 5 θέσεις στην πτήση που αντιστοιχεί στο X και τις κρατά στην πτήση που αντιστοιχεί στο Y) και M=4 (η T2 κρατά 4 θέσεις στην πτήση που αντιστοιχεί στο X), το τελικό αποτέλεσμα πρέπει να είναι X=79. Ωστόσο, στο χρονοπρόγραμμα της Εικόνας 4.3(α) έχουμε X=84, διότι *χάθηκε* η ενημέρωση που ακύρωσε 5 θέσεις.

Το Πρόβλημα της Προσωρινής Ενημέρωσης. Αυτό εμφανίζεται όταν μια δοσοληψία ενημερώνει ένα στοιχείο της βάσης δεδομένων και στη συνέχεια αποτυγχάνει για κάποιο λόγο. Μια άλλη δοσοληψία προσπελαύνει το στοιχείο που έχει ενημερωθεί, πριν αυτό επανέλθει στην αρχική του τιμή. Η Εικόνα 4.3(β) παρουσιάζει ένα παράδειγμα, όπου η T1 ενημερώνει το στοιχείο X και στη συνέχεια αποτυγχάνει πριν ολοκληρωθεί και συνεπώς το σύστημα πρέπει να επαναφέρει το X στην αρχική του τιμή. Ωστόσο, πριν γίνει η επαναφορά, η δοσοληψία T2 διαβάζει την «προσωρινή» τιμή του X, η οποία δεν θα καταχωρηθεί μόνιμα στη βάση δεδομένων, διότι η T1 αποτυγχάνει. Η τιμή του στοιχείου X που διαβάζεται από την T2 καλείται *ασταθές δεδομένο* (dirty data), διότι έχει δημιουργηθεί από μια δοσοληψία που δεν έχει ακόμη ολοκληρωθεί και επικυρωθεί (comitted). Για το λόγο αυτό, το συγκεκριμένο πρόβλημα είναι γνωστό ως πρόβλημα *ασταθούς ανάγνωσης* (dirty read).

Το Πρόβλημα της Λανθασμένης Συνάθροισης. Αν μια δοσοληψία υπολογίζει μια συναθροιστική συνάρτηση σε ένα πλήθος εγγραφών ενόσω άλλες δοσοληψίες ενημερώνουν κάποιες από αυτές τις εγγραφές, η συναθροιστική συνάρτηση μπορεί να χρησιμοποιήσει κάποιες τιμές πριν ενημερωθούν και άλλες αφού ενημερωθούν. Για παράδειγμα, υποθέστε ότι μια δοσοληψία T3 υπολογίζει το συνολικό πλήθος κρατήσεων σε όλες τις πτήσεις. Παράλληλα εκτελείται η T1. Αν η διαπλοκή των πράξεων λάβει χώρα όπως φαίνεται στην Εικόνα 4.3(γ), το αποτέλεσμα της T3 θα είναι μικρότερο από το σωστό κατά N, διότι η T3 διαβάζει την τιμή του X *αφού* αφαιρεθούν N θέσεις από αυτό, αλλά διαβάζει την τιμή του Y *πριν* αυτές οι N θέσεις προστεθούν σ' αυτό.

Ένα άλλο πρόβλημα που μπορεί να εμφανιστεί είναι η *μη επαναλήψιμη ανάγνωση,* όπου μια δοσοληψία T διαβάζει ένα στοιχείο δύο φορές και το στοιχείο έχει τροποποιηθεί από μία δοσοληψία T΄ μεταξύ των δύο αναγνώσεων. Επομένως, η T διαβάζει δύο *διαφορετικές τιμές για* το ίδιο στοιχείο.

4.1.4 Γιατί είναι απαραίτητη η Δυνατότητα Ανάκαμψης

Όταν υποβάλλεται μια δοσοληψία στο ΣΔΒΔ για εκτέλεση, το σύστημα είναι υπεύθυνο να εξασφαλίσει ότι είτε (α) όλες οι πράξεις της δοσοληψίας θα ολοκληρωθούν με επιτυχία και το αποτέλεσμά τους θα καταχωρηθεί μόνιμα στη βάση δεδομένων είτε (β) η δοσοληψία δεν έχει καμιά επίδραση στη βάση δεδομένων ή σε κάποια άλλη δοσοληψία. Το ΣΔΒΔ δεν πρέπει να επιτρέπει την εφαρμογή στη βάση δεδομένων ορισμένων πράξεων μιας δοσοληψίας T, όταν άλλες πράξεις της δοσοληψίας T δεν εφαρμοσθούν. Αυτό μπορεί να συμβεί αν μια δοσοληψία *αποτύχει* αφού εκτελέσει μερικές από τις πράξεις της αλλά πριν τις εκτελέσει όλες.

Είδη Αποτυχιών. Υπάρχουν διάφοροι πιθανοί λόγοι για τους οποίους μια δοσοληψία μπορεί να αποτύχει στο μέσον της εκτέλεσης της:

1. Αποτυχία του υπολογιστή (κατάρρευση συστήματος-system crash): εμφανίζεται σφάλμα υλικού ή λογισμικού στο υπολογιστικό σύστημα κατά την εκτέλεση μιας δοσοληψίας. Αν πρόκειται για κατάρρευση υλικού, είναι δυνατόν να χαθούν τα περιεχόμενα της μνήμης του υπολογιστή.
2. Σφάλμα δοσοληψίας ή σφάλμα συστήματος: κάποια πράξη στη δοσοληψία μπορεί να προκαλέσει αποτυχία, όπως υπερχείλιση ακεραίου ή διαίρεση με το μηδέν. Μια δοσοληψία μπορεί επίσης να αποτύχει λόγω λανθασμένων τιμών σε παραμέτρους ή λόγω προγραμματιστικού λάθους. Επιπλέον είναι δυνατόν να διακόψει ο χρήστης σκόπιμα την εκτέλεση της δοσοληψίας -για παράδειγμα πατώντας τα πλήκτρα Control-C σε περιβάλλον VAX/VMS ή UNIX.
3. Τοπικά σφάλματα και συνθήκες εξαίρεσης που εντοπίζονται από τη δοσοληψία: κατά την εκτέλεση μιας δοσοληψίας είναι δυνατόν να εμφανιστούν διάφορες συνθήκες που να καθιστούν αναγκαία την ακύρωση της δοσοληψίας. Για παράδειγμα, μπορεί να μην βρεθούν τα δεδομένα που χρειάζεται η δοσοληψία. Μια συνθήκη, π.χ. ανεπαρκές υπόλοιπο λογαριασμού σε μια τραπεζική βάση δεδομένων, μπορεί να προκαλέσει την ακύρωση μιας δοσοληψίας, π.χ. την ανάληψη χρημάτων από το συγκεκριμένο λογαριασμό. Η ακύρωση μπορεί να προκληθεί με μια εντολή ABORT, η οποία εισάγεται στον κώδικα της δοσοληψίας.
4. Εφαρμογή ελέγχου συνδρομικότητας: η μέθοδος ελέγχου συνδρομικότητας μπορεί να αποφασίσει την ακύρωση μιας δοσοληψίας η οποία θα επανεκκινηθεί αργότερα, επειδή παραβιάζει τη σειριοποιησιμότητα ή επειδή διάφορες δοσοληψίες βρίσκονται σε κατάσταση αδιεξόδου.
5. Πρόβλημα δίσκου: είναι δυνατόν να απολεσθούν τα δεδομένα που έχουν αποθηκευθεί σε κάποια μπλοκ δίσκου λόγω σφάλματος ανάγνωσης ή εγγραφής, ή λόγω προβλήματος στην κεφαλή ανάγνωσης-εγγραφής. Κάτι τέτοιο μπορεί να συμβεί κατά την εκτέλεση μιας πράξης ανάγνωσης ή εγγραφής της δοσοληψίας.
6. Φυσικά προβλήματα και καταστροφές: εδώ αναφερόμαστε σε μια ατέλειωτη λίστα προβλημάτων που περιλαμβάνουν διακοπές παροχής ρεύματος, βλάβες κλιματισμού, πυρκαγιές, κλοπές, δολιοφθορές, επικάλυψη των δεδομένων ενός δίσκου ή ταινίας από άλλα δεδομένα ή τοποθέτηση λάθος ταινίας από τον χειριστή.

Οι αποτυχίες του τύπου 1, 2, 3 και 4 είναι πιο κοινές απ' ό,τι των τύπων 5 ή 6. Οποτεδήποτε συμβεί μια αποτυχία τύπου 1 έως 4, το σύστημα πρέπει να κρατά ικανές πληροφορίες ώστε να ανακάμψει από την αποτυχία. Προβλήματα δίσκων και άλλες καταστροφικές αποτυχίες τύπου 5 ή 6 δεν συμβαίνουν συχνά, αν όμως συμβούν η ανάκαμψη είναι ιδιαίτερα επίπονη.

Η έννοια της ατομικής δοσοληψίας είναι θεμελιώδης σε πολλές τεχνικές ελέγχου συνδρομικότητας και ανάκαμψης από σφάλματα. Στο επόμενο Εδάφιο παρουσιάζουμε την έννοια της δοσοληψίας και εξηγούμε γιατί είναι σημαντική.

4.2 ΕΝΝΟΙΕΣ ΔΟΣΟΛΗΨΙΩΝ ΚΑΙ ΣΥΣΤΗΜΑΤΟΣ

Όπως αναφέραμε προηγουμένως, η *εκτέλεση ενός προγράμματος* που περιλαμβάνει πράξεις προσπέλασης σε βάση δεδομένων ονομάζεται *δοσοληψία βάσης δεδομένων* ή απλά *δοσοληψία.* Αν οι πράξεις μιας δοσοληψίας στη βάση δεδομένων δεν ενημερώνουν δεδομένα αλλά περιορίζονται σε ανακτήσεις, η δοσοληψία λέγεται

δοσοληψία ανάγνωσης (read only transaction). Στο παρόν κεφάλαιο ενδιαφερόμαστε κυρίως για δοσοληψίες που εκτελούν κάποιες ενημερώσεις στη βάση δεδομένων, συνεπώς ο όρος *δοσοληψία* αναφέρεται στην εκτέλεση ενός προγράμματος που *ενημερώνει τη βάση δεδομένων,* εκτός αν αναφέρεται ρητώς κάτι διαφορετικό.

4.2.1 Καταστάσεις Δοσοληψίας και Πρόσθετες Πράξεις

Μια δοσοληψία είναι μια ατομική μονάδα εργασίας που είτε εκτελείται στο σύνολό της είτε καθόλου. Για λόγους ανάκαμψης, το σύστημα πρέπει να φυλάσσει στοιχεία όπως πότε μια δοσοληψία αρχίζει, τερματίζει και επικυρώνεται (commits) ή ακυρώνεται (aborts-βλ. κατωτέρω). Επομένως, ο διαχειριστής ανάκαμψης φυλάσσει στοιχεία για τις κατωτέρω πράξεις:

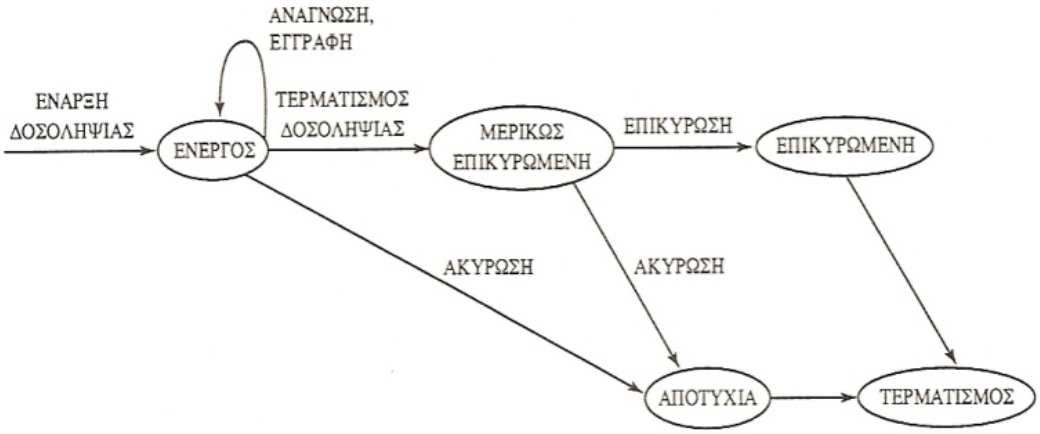
- ΕΝΑΡΞΗ_ΔΟΣΟΛΗΨΙΑΣ: αυτή η πράξη σηματοδοτεί την αρχή εκτέλεσης μιας δοσοληψίας.
- ΑΝΑΓΝΩΣΗ ή ΕΓΓΡΑΦΗ: προσδιορίζουν πράξεις ανάγνωσης και εγγραφής στοιχείων της βάσης δεδομένων, οι οποίες εκτελούνται στα πλαίσια μιας δοσοληψίας.
- ΤΕΡΜΑΤΙΣΜΟΣ_ΔΟΣΟΛΗΨΙΑΣ: η πράξη αυτή προσδιορίζει ότι οι εντολές ΑΝΑΓΝΩΣΗ και ΕΓΓΡΑΦΗ της δοσοληψίας τελείωσαν και σηματοδοτεί το πέρας της εκτέλεσης της δοσοληψίας. Ωστόσο, στο σημείο αυτό μπορεί να είναι απαραίτητο να ελεγχθεί αν οι αλλαγές που εισήγαγε η δοσοληψία μπορούν να εφαρμοσθούν μόνιμα στη βάση δεδομένων ή πρέπει να ακυρωθεί η δοσοληψία διότι παραβιάζει τον έλεγχο συνδρομικότητας ή για κάποιον άλλο λόγο.
- ΕΠΙΚΥΡΩΣΗ_ΔΟΣΟΛΗΨΙΑΣ: αυτή η πράξη σηματοδοτεί ένα *επιτυχημένο πέρας* της δοσοληψίας δηλώνοντας ότι οποιεσδήποτε αλλαγές (ενημερώσεις) εκτελέσθηκαν από τη δοσοληψία μπορούν να *επικυρωθούν* (καταχωρηθούν μόνιμα) στη βάση δεδομένων και ότι δεν πρόκειται να αναιρεθούν.
- ΑΝΑΚΛΗΣΗ (ή ΑΚΥΡΩΣΗ): η πράξη αυτή σηματοδοτεί ότι η δοσοληψία *περατώθηκε ανεπιτυχώς,* συνεπώς τυχόν αλλαγές ή επιπτώσεις που είναι δυνατόν να επέφερε η δοσοληψία στη βάση δεδομένων πρέπει να *αναιρεθούν.*

Πέραν των παραπάνω πράξεων, μερικές τεχνικές ανάκαμψης απαιτούν πρόσθετες πράξεις που περιλαμβάνουν τις κατωτέρω:

- *ΑΝΑΙΡΕΣΗ: είναι παρόμοια με την ΑΝΑΚΛΗΣΗ, με τη διαφορά ότι εφαρμόζεται σε μία πράξη, αντί σε ολόκληρη δοσοληψία.*
- *ΕΠΑΝΑΛΗΨΗ: αυτή η πράξη προσδιορίζει ότι κάποιες* πράξεις μιας δοσοληψίας *πρέπει να* εκτελεσθούν ξανά, *για να εξασφαλισθεί ότι όλες οι πράξεις της δοσοληψίας που έχει επικυρωθεί εφαρμόσθηκαν με επιτυχία στη βάση δεδομένων.*

Η Εικόνα 4.4 παρουσιάζει ένα διάγραμμα μετάπτωσης καταστάσεων το οποίο περιγράφει πώς μεταπίπτει μια δοσοληψία μεταξύ των καταστάσεων εκτέλεσης. Μια δοσοληψία οδηγείται στην *ενεργό* κατάσταση αμέσως μόλις αρχίσει η εκτέλεσή της. Στην κατάσταση αυτή μπορεί να εκτελέσει πράξεις ΑΝΑΓΝΩΣΗ και ΕΓΓΡΑΦΗ. Όταν η δοσοληψία περατώνεται, μεταπίπτει στην κατάσταση *μερικής επικύρωσης.* Στο σημείο αυτό μερικές τεχνικές ελέγχου συνδρομικότητας απαιτούν να διεξαχθούν κάποιοι *έλεγχοι* για να εξασφαλισθεί ότι δεν υπήρξαν αλληλοπαρεμβολές μεταξύ της

συγκεκριμένης δοσοληψίας και άλλων εκτελούμενων δοσοληψιών. Επιπλέον, σε μερικά πρωτόκολλα ανάκαμψης πρέπει να εξασφαλισθεί ότι μια αποτυχία συστήματος δεν θα έχει ως αποτέλεσμα την αδυναμία μόνιμης καταχώρησης των αλλαγών που εισήγαγε η δοσοληψία (συνήθως καταγράφοντας τις αλλαγές στο ημερολόγιο συστήματος, το οποίο εξετάζεται στο επόμενο εδάφιο). Μόλις και οι δύο έλεγχοι ολοκληρωθούν επιτυχώς, η δοσοληψία έχει φτάσει στο σημείο επικύρωσής της και μεταπίπτει στην *κατάσταση επικύρωσης*.



Εικόνα 4.4: Διάγραμμα μετάπτωσης καταστάσεων για την εκτέλεση δοσοληψιών

Μόλις μια δοσοληψία μεταπέσει στην κατάσταση επικύρωσης, έχει ολοκληρώσει την εκτέλεσή της με επιτυχία.

Μια δοσοληψία μπορεί ωστόσο να μεταπέσει σε *κατάσταση αποτυχίας* (failed state) αν κάποιος έλεγχος αποτύχει ή αν ακυρωθεί ενόσω βρίσκεται σε ενεργό κατάσταση. Τότε μπορεί να χρειασθεί να ανακληθεί για να αναιρέσει τις επιπτώσεις των πράξεων ΕΓΓΡΑΦΗ, σε ό,τι αφορά τη βάση δεδομένων. Η *κατάσταση τερματισμού* (terminated sate) αντιστοιχεί στην απομάκρυνση της δοσοληψίας από το σύστημα. Δοσοληψίες που αποτυγχάνουν ή ακυρώνονται μπορεί να *επανεκκινηθούν* αργότερα, είτε αυτόματα είτε αφού υποβληθούν ξανά ως *νέες* δοσοληψίες.

4.2.2 Ημερολόγιο Συστήματος (System Log ή System Journal)

Προκειμένου να είναι δυνατή η ανάκαμψη από αποτυχίες δοσοληψιών, το σύστημα διατηρεί ένα *ημερολόγιο* (log ή journal). Στο ημερολόγιο καταχωρούνται πληροφορίες για όλες τις πράξεις των δοσοληψιών που επηρεάζουν τις τιμές των στοιχείων της βάσης δεδομένων. Αυτές οι πληροφορίες χρειάζονται για να είναι δυνατή η ανάκαμψη από αποτυχίες δοσοληψιών. Το ημερολόγιο φυλάσσεται στο δίσκο και έτσι δεν επηρεάζεται από κανένα τύπο αποτυχίας, εκτός από πρόβλημα δίσκου ή καταστροφή. Επιπλέον, κατά περιόδους λαμβάνονται εφεδρικά αντίγραφα του ημερολογίου σε ταινίες, για εξασφάλιση έναντι καταστροφικών αποτυχιών. Παραθέτουμε τώρα τους τύπους των καταχωρήσεων που εισάγονται στο ημερολόγιο και τη δράση καθεμιάς. Στις καταχωρήσεις αυτές το γράμμα Τ συμβολίζει μία μοναδική *ταυτότητα δοσοληψίας* (transaction-id), η οποία παράγεται αυτόματα από το σύστημα και χρησιμοποιείται για την αναγνώριση κάθε δοσοληψίας:

1. [έναρξη_δοσοληψίας, Τ]: Καταχωρεί ότι ξεκίνησε η εκτέλεση της δοσοληψίας Τ.

2. [εγγραφή_στοιχείου, T, X, παλαιά_τιμή, νέα_τιμή]: Καταχωρεί ότι η δοσοληψία T τροποποίησε την τιμή του στοιχείου X της βάσης δεδομένων από παλαιά_τιμή σε νέα_τιμή.
3. [ανάγνωση_στοιχείου, T, X]: Καταχωρεί ότι η δοσοληψία T έχει διαβάσει την τιμή του στοιχείου X της βάσης δεδομένων.
4. [επικύρωση, T]: Καταχωρεί ότι η δοσοληψία T έχει ολοκληρωθεί με επιτυχία και επιβεβαιώνει ότι η επίδραση της μπορεί να καταχωρηθεί μόνιμα στη βάση δεδομένων.
5. [ακύρωση, T]: Καταχωρεί ότι η δοσοληψία T έχει ακυρωθεί.

Τα πρωτόκολλα ανάκαμψης που αποφεύγουν διάδοση της ανάκλησης δεν απαιτούν να καταγράφονται οι πράξεις ΑΝΑΓΝΩΣΗ στο ημερολόγιο, ενώ άλλα πρωτόκολλα απαιτούν αυτές τις καταχωρήσεις για λόγους ανάκαμψης. Στην πρώτη περίπτωση ελαττώνεται η επιβάρυνση καταγραφής των πράξεων στο ημερολόγιο μια και καταχωρούνται λιγότερες πράξεις μόνο οι πράξεις ΕΓΓΡΑΦΗ. Επιπλέον, τα αυστηρά πρωτόκολλα απαιτούν απλούστερες καταχωρήσεις ΕΓΓΡΑΦΗ, οι οποίες δεν περιλαμβάνουν την τιμή νέα_τιμή.

Σημειώνουμε ότι εδώ υποτίθεται πως οι δοσοληψίες δεν μπορούν να εμφωλευθούν. Επίσης υποθέτουμε ότι *όλες* οι μόνιμες αλλαγές στη βάση δεδομένων γίνονται στα πλαίσια δοσοληψιών, επομένως η έννοια της ανάκαμψης από αποτυχία δοσοληψίας ισοδυναμεί με την αναίρεση ή την επανάληψη κάθε *ανακάμψιμης* (recoverable) πράξης της συγκεκριμένης δοσοληψίας, με τη βοήθεια του ημερολογίου. Αν καταρρεύσει το σύστημα, μπορούμε να επαναφέρουμε τη βάση δεδομένων σε συνεπή κατάσταση εξετάζοντας το ημερολόγιο. Δεδομένου ότι το ημερολόγιο περιέχει μια καταχώρηση για κάθε πράξη ΕΓΓΡΑΦΗ που αλλάζει την τιμή κάποιου στοιχείου της βάσης δεδομένων, είναι δυνατόν να *αναιρεθεί* (undo) η επίδραση των πράξεων ΕΓΓΡΑΦΗ της δοσοληψίας T, ανατρέχοντας στο ημερολόγιο και επαναφέροντας τα στοιχεία που τροποποιήθηκαν από πράξεις ΕΓΓΡΑΦΗ στην παλαιά τιμή τους. Μπορούμε επίσης να *επαναλάβουμε* (redo) την επίδραση των πράξεων ΕΓΓΡΑΦΗ μιας δοσοληψίας T σαρώνοντας το ημερολόγιο και θέτοντας τα στοιχεία που τροποποιήθηκαν από κάθε πράξη ΕΓΓΡΑΦΗ της δοσοληψίας T στις νέες τους τιμές. Η επανάληψη των πράξεων μιας δοσοληψίας μπορεί να απαιτηθεί αν όλες οι ενημερώσεις της έχουν καταχωρηθεί στο ημερολόγιο, αλλά συμβεί κάποιο σφάλμα πριν διασφαλισθεί ότι όλες οι νέες τιμές γράφτηκαν μόνιμα στη βάση δεδομένων.

4.2.3 Σημείο Επικύρωσης μιας Δοσοληψίας

Μια δοσοληψία T φθάνει στο *σημείο επικύρωσής* της (commit point) όταν όλες οι πράξεις της που προσπελαύνουν τη βάση δεδομένων έχουν εκτελεσθεί επιτυχώς *και* η επίδραση όλων των πράξεων της δοσοληψίας στη βάση δεδομένων έχει καταχωρηθεί στο ημερολόγιο. Μια δοσοληψία που πέρασε το σημείο επικύρωσης καλείται *επικυρωμένη* (commited) και θεωρείται ότι η επίδρασή της έχει *καταχωρηθεί μόνιμα* στη βάση δεδομένων. Η δοσοληψία τότε εισάγει μια καταχώρηση [επικύρωση, T] στο ημερολόγιο. Αν παρουσιαστεί αποτυχία συστήματος, διατρέχουμε το ημερολόγιο από το τέλος προς την αρχή αναζητώντας τις δοσοληψίες T που έχουν εισάγει μια καταχώρηση [έναρξη_δοσοληψίας, T] στο ημερολόγιο αλλά όχι και την καταχώρηση [επικύρωση, T]. Οι δοσοληψίες αυτές πρέπει να *ανακληθούν,* έτσι ώστε να αναιρεθούν οι επιδράσεις τους στη βάση δεδομένων. Αυτό αποτελεί τμήμα της διαδικασίας ανάκαμψης. Οι δοσοληψίες που έχουν εισάγει την καταχώρηση

επικύρωσής τους στο ημερολόγιο πρέπει να έχουν εισάγει επίσης και όλες τις πράξεις ΕΓΓΡΑΦΗ που εκτέλεσαν (στην αντίθετη περίπτωση δεν θα ήταν επικυρωμένες), συνεπώς οι επιδράσεις τους στη βάση δεδομένων μπορούν να *επαναληφθούν* με βάση τις καταχωρήσεις του ημερολογίου.

Σημειώνουμε ότι το αρχείο του ημερολόγιου πρέπει να αποθηκεύεται στο δίσκο. Η ενημέρωση ενός αρχείου στο δίσκο περιλαμβάνει αντιγραφή του κατάλληλου μπλοκ από το δίσκο σε μια ενδιάμεση περιοχή μνήμης, ενημέρωση της ενδιάμεσης περιοχής και αντιγραφή της ενδιάμεσης περιοχής από τη μνήμη στο δίσκο. Συνηθίζεται να παραμένουν τα μπλοκ του αρχείου ημερολόγιου στην κύρια μνήμη μέχρι να γεμίσουν από καταχωρήσεις έτσι ώστε να γράφονται μία φορά στο δίσκο, αντί να γράφονται στο δίσκο κάθε φορά που προστίθεται μια καταχώρηση στο ημερολόγιο. Με τον τρόπο αυτό αποφεύγεται η επιβάρυνση της πολλαπλής εγγραφής στο δίσκο του ίδιου μπλοκ αρχείου. Αν καταρρεύσει το σύστημα, κατά τη διαδικασία ανάκαμψης θα ληφθούν υπόψη μόνο οι καταχωρήσεις ημερολογίου που έχουν *γραφεί στο δίσκο,* διότι τα περιεχόμενα της μνήμης μπορεί να χαθούν. Επομένως, *πριν μια* δοσοληψία φθάσει στο σημείο επικύρωσής της, τμήματα του ημερολογίου που δεν έχουν γραφεί ακόμη στο δίσκο πρέπει να γραφούν άμεσα. Η διαδικασία αυτή καλείται *εξαναγκαστική εγγραφή* (force-writing) του αρχείου ημερολόγιου πριν επικυρωθεί μια δοσοληψία.

4.2.4 Σημεία Ελέγχου στο Ημερολόγιο Συστήματος

Ένα άλλο είδος καταχώρησης στο ημερολόγιο καλείται σημείο ελέγχου (checkpoint). Μια εγγραφή [σημείο_ελέγχου] εισάγεται στο ημερολόγιο περιοδικά, όταν το σύστημα καταχωρεί στα αρχεία της βάσης δεδομένων την επίδραση όλων των πράξεων ΕΓΓΡΑΦΗ των επικυρωμένων δοσοληψιών. Επομένως, για όλες τις δοσοληψίες που έχουν εισάγει στο ημερολόγιο την καταχώρηση [επικύρωση, Τ] πριν από μια καταχώρηση [σημείο_ελέγχου] δεν είναι απαραίτητο να επαναληφθούν οι πράξεις ΕΓΓΡΑΦΗ σε περίπτωση κατάρρευσης του συστήματος.

Ο διαχειριστής ανάκαμψης του ΣΔΒΔ πρέπει να καθορίζει τα διαστήματα εισαγωγής σημείων ελέγχου. Το διάστημα μπορεί να αντιστοιχεί σε χρόνο – π.χ. κάθε λ λεπτά – ή σε ένα πλήθος δ από δοσοληψίες που επικυρώθηκαν από τη στιγμή εισαγωγής του τελευταίου σημείου ελέγχου. Οι τιμές των λ και δ αποτελούν παραμέτρους του συστήματος. Η εισαγωγή ενός σημείου ελέγχου συνίσταται στις ακόλουθες ενέργειες:

1. Προσωρινή αναστολή εκτέλεσης των δοσοληψιών.
2. Εξαναγκαστική εγγραφή όλων των πράξεων ενημέρωσης των επικυρωμένων δοσοληψιών από τις ενδιάμεσες περιοχές της κύριας μνήμης στο δίσκο.
3. Εισαγωγή μιας εγγραφής [σημείο_ελέγχου] στο ημερολόγιο και εξαναγκαστική εγγραφή του ημερολογίου στο δίσκο.
4. Συνέχιση της εκτέλεσης των δοσοληψιών.

Μια καταχώρηση σημείου ελέγχου στο ημερολόγιο μπορεί να περιλαμβάνει επίσης και άλλες πληροφορίες, όπως μια λίστα με τις ταυτότητες των ενεργών δοσοληψιών και τις θέσεις (διευθύνσεις) της πρώτης και της πιο πρόσφατης (τελευταίας) εγγραφής στο ημερολόγιο για κάθε ενεργό δοσοληψία. Αυτό μπορεί να βοηθήσει στην ακύρωση των πράξεων μιας δοσοληψίας, σε περίπτωση που η δοσοληψία πρέπει να ανακληθεί.

4.3 ΕΠΙΘΥΜΗΤΕΣ ΙΔΙΟΤΗΤΕΣ ΤΩΝ ΔΟΣΟΛΗΨΙΩΝ

Οι ατομικές δοσοληψίες πρέπει να έχουν διάφορες ιδιότητες που συχνά αναφέρονται ως ιδιότητες *ACID*. Οι ιδιότητες ACID πρέπει να διασφαλίζονται από τις μεθόδους ελέγχου συνδρομικότητας και ανάκαμψης του ΣΔΒΔ, και είναι οι εξής:

1. *Ατομικότητα* (Atomicity): μια δοσοληψία είναι μια ατομική μονάδα επεξεργασίας, η οποία είτε εκτελείται εξ ολοκλήρου είτε δεν εκτελείται καθόλου.

2. *Διατήρηση της συνέπειας* (Consistency preservation): η σωστή εκτέλεση μιας δοσοληψίας οδηγεί τη βάση δεδομένων από μια συνεπή κατάσταση σε μια άλλη.

3. *Απομόνωση* (Isolation): μια δοσοληψία δεν κοινοποιεί τις ενημερώσεις της στις άλλες δοσοληψίες πριν επικυρωθεί. Όταν η ιδιότητα αυτή επιβάλλεται αυστηρά, λύνεται το πρόβλημα της προσωρινής ενημέρωσης και δεν είναι αναγκαία η διάδοση των ακυρώσεων των δοσοληψιών.

4. *Διάρκεια* (Durability) ή *μονιμότητα:* αν μια δοσοληψία τροποποιήσει τη βάση δεδομένων και οι αλλαγές επικυρωθούν, οι αλλαγές αυτές δεν είναι δυνατόν να χαθούν λόγω μεταγενέστερης αποτυχίας.

Η ιδιότητα της ατομικότητας απαιτεί την εκτέλεση των δοσοληψιών μέχρι ολοκλήρωσης. Η εξασφάλιση της ατομικότητας είναι υπευθυνότητα της μεθόδου ανάκαμψης. Αν μια δοσοληψία αποτύχει να ολοκληρωθεί για κάποιο λόγο, όπως κατάρρευση του συστήματος κατά τη διάρκεια εκτέλεσης της δοσοληψίας, η μέθοδος ανάκαμψης πρέπει να αναιρέσει κάθε επίδραση της δοσοληψίας στη βάση δεδομένων.

Η ιδιότητα της διατήρησης της συνέπειας θεωρείται γενικά υπευθυνότητα των προγραμματιστών, οι οποίοι γράφουν τα προγράμματα βάσης δεδομένων ή του τμήματος του ΣΔΒΔ που επιβάλλει τους περιορισμούς ακεραιότητας. Υπενθυμίζουμε ότι μια *κατάσταση της βάσης δεδομένων είναι* μια συλλογή όλων των αποθηκευμένων δεδομένων (τιμών) στη βάση, σε κάποια δεδομένη χρονική στιγμή. Μια *συνεπής κατάσταση* της βάσης δεδομένων ικανοποιεί τους περιορισμούς που ορίζονται στο σχήμα, καθώς και κάθε άλλον περιορισμό που πρέπει να ισχύει στη βάση δεδομένων. Ένα πρόγραμμα βάσης δεδομένων πρέπει να γραφεί κατά τρόπο που να εξασφαλίζει ότι, αν η βάση δεδομένων βρίσκεται σε συνεπή κατάσταση πριν εκτελεσθεί η δοσοληψία, θα βρίσκεται σε συνεπή κατάσταση και μετά την *ολοκλήρωση* της εκτέλεσης της δοσοληψίας, υποθέτοντας ότι δεν υπάρχουν *αλληλοπαρεμβολές με άλλες δοσοληψίες.*

Η απομόνωση διασφαλίζεται από τη μέθοδο ελέγχου συνδρομικότητας. Έχουν γίνει προσπάθειες να ορισθεί *το επίπεδο* ή *ο βαθμός απομόνωσης* μιας δοσοληψίας. Μια δοσοληψία έχει βαθμό απομόνωσης 0 (μηδέν) αν δεν επικαλύπτει (overwrite) τις ασταθείς αναγνώσεις (dirty reads) δοσοληψιών με υψηλότερο βαθμό. Μια δοσοληψία με βαθμό απομόνωσης 1 (ένα) δεν παρουσιάζει απώλειες ενημερώσεων. Στο βαθμό απομόνωσης 2 δεν εμφανίζονται απώλειες ενημερώσεων ούτε ασταθείς αναγνώσεις. Τέλος, ο βαθμός απομόνωσης 3 (που ονομάζεται επίσης *πραγματική απομόνωση)* περιλαμβάνει, πέραν των ιδιοτήτων του επιπέδου 2, επαναλήψιμες αναγνώσεις (repeatable reads). Η ιδιότητα της διάρκειας αποτελεί υπευθυνότητα της μεθόδου ανάκαμψης.

4.4 ΧΡΟΝΟΠΡΟΓΡΑΜΜΑΤΑ ΚΑΙ ΔΥΝΑΤΟΤΗΤΑ ΑΝΑΚΑΜΨΗΣ

Όταν οι δοσοληψίες εκτελούνται συνδρομικά με διαπεπλεγμένο τρόπο, η διάταξη εκτέλεσης των πράξεων των διαφόρων δοσοληψιών σχηματίζει μια ακολουθία γνωστή ως *χρονοπρόγραμμα (ή χρονικό)* δοσοληψιών. Στο Εδάφιο αυτό ορίζουμε πρώτα την έννοια του χρονοπρογράμματος και στη συνέχεια χαρακτηρίζουμε τους τύπους των χρονοπρογραμμάτων, ανάλογα με το πόσο διευκολύνουν την ανάκαμψη σε περίπτωση εμφάνισης αποτυχίας. Στο επόμενο Εδάφιο χαρακτηρίζουμε τα χρονοπρογράμματα με βάση την αλληλοπαρεμβολή των δοσοληψιών που συμμετέχουν, εισάγοντας τις έννοιες της σειριοποιησιμότητας και των σειριοποιήσιμων χρονοπρογραμμάτων.

4.4.1 Χρονοπρογράμματα (Χρονικά) Δοσοληψιών

Χρονοπρόγραμμα (ή χρονικό) S των δοσοληψιών T1, T2, ..., Tn είναι μια διάταξη των πράξεων των δοσοληψιών η οποία υπόκειται στον περιορισμό ότι, για κάθε δοσοληψία Ti που συμμετέχει στο S, οι πράξεις της Tj στο S πρέπει να εμφανισθούν με την ίδια σειρά που εμφανίζονται στην Ti. Σημειώνουμε, ωστόσο, ότι πράξεις από άλλες δοσοληψίες Tj είναι δυνατόν να διαπλέκονται με τις πράξεις της Tj στο S. Για την ώρα, θεωρούμε ότι η διάταξη των πράξεων στο S είναι μια *ολική διάταξη,* αν και είναι δυνατόν να χρησιμοποιήσουμε χρονοπρογράμματα των οποίων οι πράξεις σχηματίζουν *μερικές διατάξεις* (όπως εξετάζεται στη συνέχεια).

Η Εικόνα 4.3(α) και (β) παρουσιάζει δύο πιθανά χρονοπρογράμματα για τις δοσοληψίες T1 και T2. Για λόγους ανάκαμψης και ελέγχου συνδρομικότητας ενδιαφερόμαστε κυρίως για τις πράξεις ανάγνωση_στοιχείου και εγγραφή_στοιχείου των δοσοληψιών, καθώς και για τις πράξεις ΕΠΙΚΥΡΩΣΗ και ΑΚΥΡΩΣΗ. Ένας στενογραφικός συμβολισμός για την περιγραφή ενός χρονοπρογράμματος χρησιμοποιεί τα σύμβολα r, w, c και a για τις πράξεις ανάγνωση_στοιχείου (read_item), εγγραφή_στοιχείου (write_item), επικύρωση (commit) και ακύρωση (abort) αντίστοιχα, επισυνάπτοντας ως δείκτη την ταυτότητα της δοσοληψίας σε κάθε πράξη στο χρονοπρόγραμμα. Σύμφωνα με το συμβολισμό αυτό, το στοιχείο X της βάσης δεδομένων που διαβάζεται ή γράφεται ακολουθεί τις πράξεις r και w εντός παρενθέσεων. Για παράδειγμα, το χρονοπρόγραμμα της Εικόνας 4.3(α), το οποίο θα συμβολίσουμε με Sα, μπορεί να γραφεί ως ακολούθως όταν συμπληρωθεί με τις πράξεις επικύρωσης:

Sa: r1(X); r2(X); wl(X); r1(Y); w2(X); c2; w1(Y); c1;

Παρομοίως, το χρονοπρόγραμμα της Εικόνας 4.3(β), το οποίο θα το συμβολίσουμε Sβ, μπορεί να γραφεί ως ακολούθως αν υποθέσουμε ότι η δοσοληψία T1 ακυρώνεται μετά την πράξη ανάγνωση_στοιχείου(Y):

Sβ: r1(X); w1(X); r2(X); w2(X); c2; r1(Y); a1;

Δύο πράξεις σ' ένα χρονοπρόγραμμα λέμε ότι *συγκρούονται* αν ανήκουν σε διαφορετικές δοσοληψίες, προσπελαύνουν το ίδιο στοιχείο X και τουλάχιστον μία από τις δύο πράξεις είναι η εγγραφή_στοιχείου (X). Για παράδειγμα, στο χρονοπρόγραμμα Sα οι πράξεις r1(X) και w2(X) συγκρούονται, όπως και οι πράξεις r2(X) και w1(X) και οι πράξεις W1(X) και w2(X). Ωστόσο οι πράξεις r1(X) και r2(X) δεν συγκρούονται, διότι και οι δύο είναι πράξεις ανάγνωσης. Επίσης, οι πράξεις w2(X) και w1(Y) δεν συγκρούονται επειδή εφαρμόζονται σε διαφορετικά δεδομένα, τα X και Y.

Ένα χρονοπρόγραμμα S των δοσοληψιών T1,T2,...,Tn καλείται *πλήρες χρονοπρόγραμμα* αν ισχύουν οι κατωτέρω συνθήκες:

1. Οι πράξεις στο S είναι ακριβώς οι πράξεις στις T1,T2,...,Tn συμπεριλαμβανομένης μιας πράξης επικύρωσης ή ακύρωσης ως τελευταίας πράξης για κάθε δοσοληψία στο χρονοπρόγραμμα.
2. Για κάθε ζεύγος πράξεων από την ίδια δοσοληψία Ti, η σειρά εμφάνισής τους στο S είναι ίδια με τη σειρά εμφάνισής τους στην Ti.
3. Για κάθε ζεύγος συγκρουόμενων πράξεων, μία από τις δύο πρέπει να βρίσκεται πριν από την άλλη στο χρονοπρόγραμμα.

Οι ανωτέρω συνθήκες επιτρέπουν σε δύο *μη συγκρουόμενες* πράξεις να εμφανίζονται στο χρονοπρόγραμμα χωρίς να ορίζεται ποια θα είναι πρώτη, οδηγώντας έτσι στον ορισμό του χρονοπρογράμματος ως *μερική διάταξη* των πράξεων των δοσοληψιών. Ωστόσο, πρέπει να ορισθεί στο χρονοπρόγραμμα μια ολική διάταξη για κάθε ζεύγος συγκρουόμενων πράξεων (συνθήκη 3) και για κάθε ζεύγος πράξεων από την ίδια δοσοληψία (συνθήκη 2). Η συνθήκη 1 απλά ορίζει ότι όλες οι πράξεις στις δοσοληψίες πρέπει να εμφανίζονται σε ένα πλήρες χρονοπρόγραμμα. Δεδομένου ότι κάθε δοσοληψία είτε επικυρώνεται είτε ακυρώνεται, ένα πλήρες χρονοπρόγραμμα δεν περιέχει ενεργές δοσοληψίες.

Γενικά, είναι δύσκολο να διαμορφωθούν πλήρη χρονοπρογράμματα σε ένα σύστημα επεξεργασίας δοσοληψιών, διότι υποβάλλονται συνεχώς νέες δοσοληψίες στο σύστημα. Επομένως, είναι χρήσιμο να ορίσουμε την έννοια της *επικυρωμένης προβολής* C(S) ενός χρονοπρογράμματος S η οποία περιλαμβάνει μόνο τις πράξεις στο S που ανήκουν σε επικυρωμένες δοσοληψίες -δηλαδή, τις δοσοληψίες Ti για τις οποίες η πράξη επικύρωσης c_i ανήκει στο S.

4.4.2 Χαρακτηρισμός των Χρονοπρογραμμάτων με Βάση τη Δυνατότητα Ανάκαμψης

Για μερικά χρονοπρογράμματα είναι εύκολη η ανάκαμψη από αποτυχίες δοσοληψιών ενώ για άλλα χρονοπρογράμματα η διαδικασία ανάκαμψης μπορεί να είναι αρκετά περίπλοκη. Επομένως, είναι σημαντικό να χαρακτηρισθούν οι τύποι των χρονοπρογραμμάτων για τους οποίους είναι δυνατή η ανάκαμψη, καθώς και εκείνοι για τους οποίους η ανάκαμψη είναι σχετικά απλή. Πρώτα θα θέλαμε να εξασφαλίσουμε ότι αν μια δοσοληψία Τ επικυρωθεί δεν θα χρειασθεί ποτέ να ανακληθεί. Τα χρονοπρογράμματα που ικανοποιούν αυτό το κριτήριο καλούνται *χρονοπρογράμματα με δυνατότητα ανάκαμψης.*

Ένα χρονοπρόγραμμα S λέγεται ότι έχει *δυνατότητα ανάκαμψης* αν καμία δοσοληψία Τ στο S δεν επικυρώνεται έως ότου επικυρωθούν όλες οι δοσοληψίες Τ΄ οι οποίες τροποποίησαν ένα δεδομένο που διαβάζει η Τ. Λέμε ότι μια δοσοληψία Τ *διαβάζει από* μια δοσοληψία Τ΄ σε ένα χρονοπρόγραμμα S, αν κάποιο στοιχείο Χ γράφεται πρώτα από την Τ΄ και στη συνέχεια διαβάζεται από την Τ. Στο χρονοπρόγραμμα S η Τ΄ δεν πρέπει να ακυρωθεί πριν η Τ διαβάσει το στοιχείο Χ και δεν πρέπει να υπάρχουν δοσοληψίες που γράφουν το στοιχείο Χ αφού το γράψει η Τ΄ και πριν το διαβάσει η Τ (εκτός αν οι δοσοληψίες αυτές, εφόσον υπάρχουν, ακυρωθούν πριν η Τ διαβάσει το Χ).

Το χρονοπρόγραμμα Sα που παρουσιάστηκε στο προηγούμενο Εδάφιο έχει δυνατότητα ανάκαμψης. Αντίθετα, θεωρήστε τα δύο (μερικά) χρονοπρογράμματα Sγ και Sδ που ακολουθούν:

Sγ: r1(X); wl(X); r2(X); r1(Y); w2(X); c2; a1;

Sδ: r1(X); wl(X); r2(X); r1(Y); w2(X); wl(Y); cl; c2;

Το *Sγ* δεν έχει δυνατότητα ανάκαμψης διότι η δοσοληψία Τ2 διαβάζει το στοιχείο Χ από την Τ1 και ακολούθως η Τ2 επικυρώνεται πριν επικυρωθεί η Τ1. Αν η Τ1 ακυρωθεί μετά την πράξη c2 στο Sγ, τότε η τιμή του Χ που διάβασε η Τ2 δεν είναι πλέον έγκυρη και η Τ2 πρέπει να ακυρωθεί *αφού έχει ήδη επικυρωθεί,* με αποτέλεσμα να μην έχει το χρονοπρόγραμμα δυνατότητα ανάκαμψης. Για να έχει δυνατότητα ανάκαμψης το χρονοπρόγραμμα πρέπει η πράξη c2 στο Sγ να αναβληθεί έως ότου επικυρωθεί η Τ1 όπως φαίνεται στο Sδ.

Σε ένα χρονοπρόγραμμα με δυνατότητα ανάκαμψης δεν παρουσιάζεται ποτέ ανάγκη ανάκλησης μιας επικυρωμένης δοσοληψίας. Μπορεί ωστόσο να παρουσιαστεί ένα φαινόμενο που είναι γνωστό ως *διαδιδόμενη ανάκληση* (ή *διαδιδόμενη ακύρωση),* όπου μια μη επικυρωμένη δοσοληψία πρέπει να ανακληθεί, διότι διάβασε κάποιο στοιχείο από μια δοσοληψία που απέτυχε. Αυτό συμβαίνει στο χρονοπρόγραμμα Sε, όπου η δοσοληψία Τ2 πρέπει να ανακληθεί διότι διάβασε το στοιχείο Χ από την Τ1 και στη συνέχεια η Τ1 ακυρώθηκε:

Sε: r1(X); w1(X); r2(X); r1(Y); w2(X); w1(Y); a1;

Δεδομένου ότι η διάδοση της ανάκλησης μπορεί να είναι αρκετά χρονοβόρα –αφού πολλές δοσοληψίες μπορεί να ανακληθούν- είναι σημαντικό να προσδιορισθούν τα χρονοπρογράμματα όπου εξασφαλίζεται ότι το φαινόμενο αυτό δεν θα εμφανιστεί. Ένα χρονοπρόγραμμα λέμε ότι αποφεύγει τη *διάδοση ανακλήσεων* αν κάθε δοσοληψία στο χρονοπρόγραμμα διαβάζει μόνο στοιχεία που έχουν γραφεί από *επικυρωμένες* δοσοληψίες. Στην περίπτωση αυτή, όλα τα στοιχεία που διαβάζονται θα είναι επικυρωμένα και επομένως δεν θα εμφανιστεί διάδοση ανακλήσεων. Για να ικανοποιηθεί αυτό το κριτήριο, η εντολή r2(X) στο χρονοπρόγραμμα Sε πρέπει να

αναβληθεί μέχρι να επικυρωθεί η Τ1 (ή να ακυρωθεί), καθυστερώντας έτσι την Τ2, αλλά εξασφαλίζοντας ότι δεν θα εμφανιστεί διάδοση ανακλήσεων αν ακυρωθεί η Τ1.

Τέλος υπάρχει ένα τρίτο, πιο περιοριστικό είδος χρονοπρογράμματος που καλείται *αυστηρό χρονοπρόγραμμα* και στο οποίο οι δοσοληψίες δεν μπορούν *ούτε να διαβάσουν ούτε να γράψουν* ένα στοιχείο Χ έως ότου επικυρωθεί (ή ακυρωθεί) η τελευταία δοσοληψία που έγραψε το Χ. Τα αυστηρά χρονοπρογράμματα απλοποιούν τη διαδικασία αναίρεσης των πράξεων εγγραφής, ανάγοντάς τη σε ζήτημα αποκατάστασης του πρότερου αποτυπώματος (before image) ενός δεδομένου Χ, το οποίο είναι η τιμή που είχε το Χ πριν από την ακυρωθείσα πράξη εγγραφής. Αυτή η απλή διαδικασία είναι δυνατόν να μην οδηγεί σε σωστά αποτελέσματα αν το χρονοπρόγραμμα δεν είναι αυστηρό. Για παράδειγμα, θεωρήστε το χρονοπρόγραμμα Sστ:

Sστ: w1(X,5); w2(X,8); α1;

Υποθέστε ότι η τιμή του Χ ήταν αρχικά 9, που είναι το πρότερο αποτύπωμα που βρίσκεται αποθηκευμένο στο ημερολόγιο του συστήματος μαζί με την πράξη w1(X, 5). Αν η Τ1, ακυρωθεί, όπως συμβαίνει στο Sστ, η διαδικασία ανάκαμψης που αποκαθιστά το πρότερο αποτύπωμα μιας ακυρωμένης πράξεως εγγραφής θα επαναφέρει την τιμή του Χ σε 9, παρ' ότι έχει ήδη αλλαχθεί σε 8 από τη δοσοληψία Τ2, με αποτέλεσμα να οδηγηθούμε τελικά σε λανθασμένα αποτελέσματα. Μολονότι το Sστ αποφεύγει τη διάδοση των ανακλήσεων, δεν είναι αυστηρό χρονοπρόγραμμα, αφού επιτρέπει στην Τ2 να γράψει το στοιχείο Χ παρ' ότι η δοσοληψία Τ1 η οποία είναι η τελευταία που έγραψε το Χ, δεν έχει επικυρωθεί (ή ακυρωθεί) ακόμη. Ένα αυστηρό χρονοπρόγραμμα δεν παρουσιάζει αυτό το πρόβλημα: έχει δυνατότητα ανάκαμψης και αποφεύγει τη διάδοση ανακλήσεων.

Χαρακτηρίσαμε τα χρονοπρογράμματα με βάση τη δυνατότητα ανάκαμψης. στο επόμενο Εδάφιο χαρακτηρίζουμε τους τύπους των χρονοπρογραμμάτων που θεωρούνται σωστοί όταν εκτελούνται ταυτόχρονες δοσοληψίες. Αυτά τα χρονοπρογράμματα καλούνται *σειριοποιήσιμα χρονοπρογράμματα.*

4.5 ΣΕΙΡΙΟΠΟΙΗΣΙΜΟΤΗΤΑ ΤΩΝ ΧΡΟΝΟΠΡΟΓΡΑΜΜΑΤΩΝ

Υποθέστε ότι δύο χρήστες -υπάλληλοι κρατήσεων αεροπορικής εταιρείας- υποβάλλουν στο ΣΔΒΔ τις δοσοληψίες Τ1 και Τ2 της Εικόνας 4.2 περίπου την ίδια χρονική στιγμή. Αν δεν επιτρέπεται διαπλοκή, υπάρχουν μόνο δύο δυνατοί τρόποι διάταξης των πράξεων των δύο δοσοληψιών που θα εκτελεστούν:

1. Εκτέλεση όλων των πράξεων της Τ1 (σε σειρά) ακολουθούμενη από την εκτέλεση όλων των πράξεων της δοσοληψίας Τ2 (σε σειρά).
2. Εκτέλεση όλων των πράξεων της Τ2 (σε σειρά) ακολουθούμενη από την εκτέλεση όλων των πράξεων της δοσοληψίας Τ1 (σε σειρά).

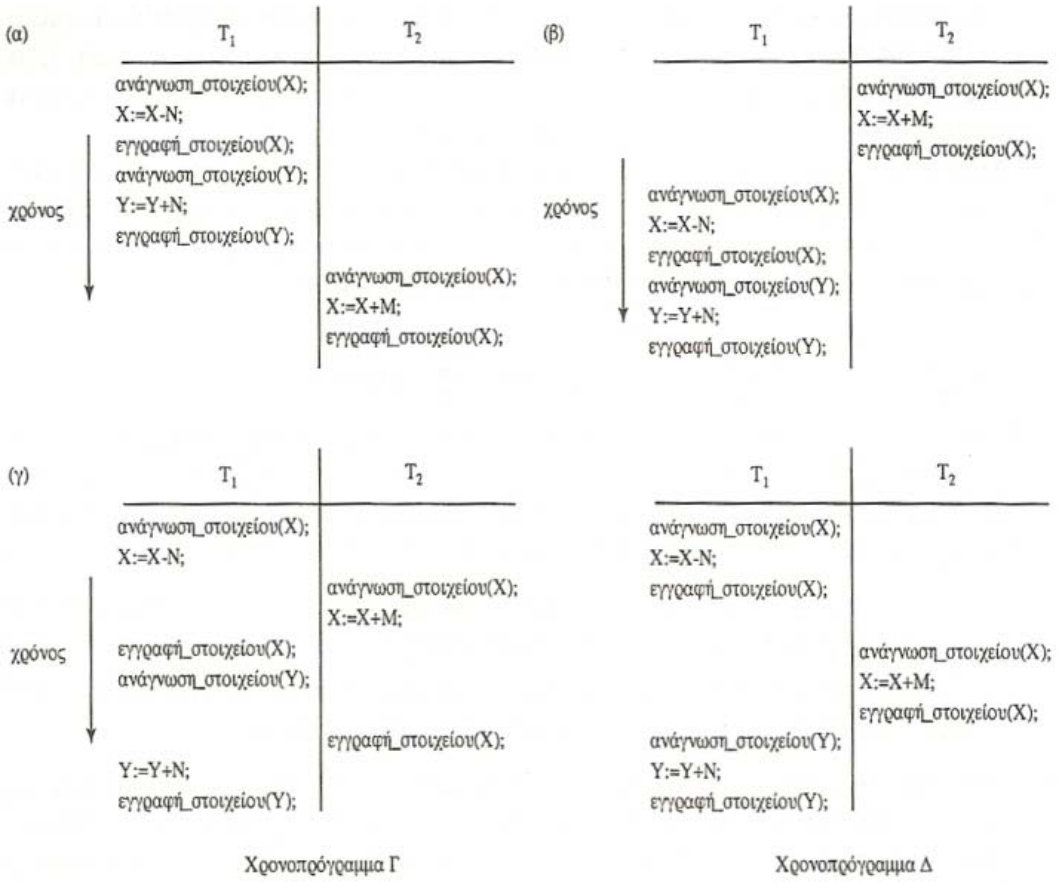
Οι δύο αυτοί εναλλακτικοί τρόποι απεικονίζονται στην Εικόνα 4.5(α) και (β) αντίστοιχα. Αν επιτρέπεται διαπλοκή των πράξεων, υπάρχουν πολλές πιθανές διατάξεις σύμφωνα με τις οποίες το σύστημα μπορεί να εκτελέσει τις επιμέρους πράξεις των δοσοληψιών. Δύο πιθανά χρονοπρογράμματα παρουσιάζονται στην Εικόνα 4.5(γ).

Ένας σημαντικός τομέας του ελέγχου συνδρομικότητας, που ονομάζεται *θεωρία σειριοποιησιμότητας,* προσπαθεί να προσδιορίσει ποια χρονοπρογράμματα είναι «σωστά» και ποια όχι, και να αναπτύξει τεχνικές που διαμορφώνουν μόνο σωστά

χρονοπρογράμματα. Το παρόν Εδάφιο ορίζει τη σειριοποιησιμότητα των χρονοπρογραμμάτων, παρουσιάζει τμήματα της θεωρίας αυτής και εξετάζει πώς μπορεί να χρησιμοποιηθεί στην πράξη.

4.5.1 Σειριακά, μη Σειριακά και Σειριοποιήσιμα Βάσει Αντιθέσεων Χρονοπρογράμματα

Η Εικόνα 4.5 παρουσιάζει διάφορα πιθανά χρονοπρογράμματα για τις δοσοληψίες Τ1 και Τ2 της Εικόνας 4.2. Τα χρονοπρογράμματα Α και Β στην Εικόνα 4.5(α) και (β) καλούνται *σειριακά χρονοπρογράμματα,* διότι οι πράξεις κάθε δοσοληψίας εκτελούνται διαδοχικά, χωρίς παρεμβολή πράξεων από την άλλη δοσοληψία. Σε ένα σειριακό χρονοπρόγραμμα ολόκληρες δοσοληψίες εκτελούνται με σειριακή διάταξη: η Τ1 ακολουθούμενη από την Τ2 στην Εικόνα 4.5(α) και η Τ2 ακολουθούμενη από την Τ1 στην Εικόνα 4.5(β). Τα χρονοπρογράμματα Γ και Δ στην Εικόνα 4.5(γ) καλούνται *μη σειριακά,* διότι σε κάθε ακολουθία διαπλέκονται πράξεις από τις δύο δοσοληψίες.



Εικόνα 4.5: Μερικά χρονοπρογράμματα που περιλαμβάνουν τις δοσοληψίες Τ1 και Τ2. (α) Χρονοπρόγραμμα Α: η Τ1 ακολουθούμενη από την Τ2 (β) Χρονοπρόγραμμα Β: η Τ2 ακολουθούμενη από την Τ1 (γ) Δυο χρονοπρογράμματα με διαπλεκόμενες πράξεις.

Τυπικά ένα χρονοπρόγραμμα S είναι *σειριακό* αν, για κάθε δοσοληψία Τ που συμμετέχει στο χρονοπρόγραμμα, όλες οι πράξεις της Τ εκτελούνται διαδοχικά στο χρονοπρόγραμμα. Στην αντίθετη περίπτωση το χρονοπρόγραμμα καλείται *μη σειριακό*. Μια λογική υπόθεση που μπορούμε να κάνουμε, αν θεωρήσουμε τις δοσοληψίες *ανεξάρτητες,* είναι ότι *κάθε σειριακό χρονοπρόγραμμα θεωρείται σωστό.* Αυτό ισχύει διότι υποθέτουμε ότι κάθε δοσοληψία είναι σωστή αν εκτελεσθεί μεμονωμένα (βάσει της ιδιότητας διατήρησης της συνέπειας) και ότι δεν υπάρχουν

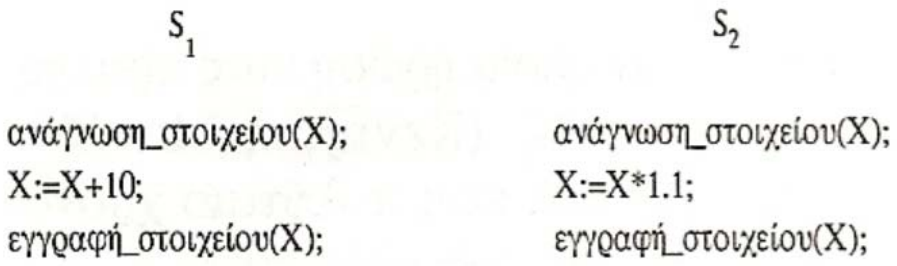
αλληλεξαρτήσεις μεταξύ των δοσοληψιών. Επομένως, δεν έχει σημασία ποια δοσοληψία εκτελείται πρώτη: αν κάθε δοσοληψία εκτελείται από την αρχή μέχρι το τέλος χωρίς την παρεμβολή πράξεων από άλλες δοσοληψίες, έχουμε σωστό τελικό αποτέλεσμα στη βάση δεδομένων. Το πρόβλημα με τα σειριακά χρονοπρογράμματα είναι ότι περιορίζουν τη συνδρομικότητα ή τη διαπλοκή των πράξεων. Αν σε ένα σειριακό χρονοπρόγραμμα μια δοσοληψία περιμένει την ολοκλήρωση μιας πράξης εισόδου / εξόδου, δεν μπορούμε να μετάγουμε την ΚΜΕ (Κεντρική Μονάδα Επεξεργασίας) σε κάποια άλλη δοσοληψία, σπαταλώντας έτσι πολύτιμο χρόνο ΚΜΕ και καθιστώντας τα σειριακά χρονοπρογράμματα μη αποδεκτά.

Ως παράδειγμα, θεωρείστε τα χρονοπρογράμματα στην Εικόνα 4.5 και υποθέστε ότι οι αρχικές τιμές των στοιχείων της βάσης δεδομένων είναι X=90. Y=90 και ότι N=3 και M=2. Μετά την εκτέλεση των δοσοληψιών T1 και T2 θα περιμέναμε ότι οι τιμές στη βάση δεδομένων θα είναι X=89 και Y=93, σύμφωνα με τη σημασιολογία των δοσοληψιών. Είναι βέβαιο ότι εκτελώντας οποιοδήποτε από τα χρονοπρογράμματα Α ή Β προκύπτουν σωστά αποτελέσματα. Τώρα θεωρήστε τα μη σειριακά χρονοπρογράμματα Γ και Δ. Το χρονοπρόγραμμα Γ (όμοιο με αυτό της Εικόνας 4.3(α)) δίνει τα αποτελέσματα X=92 και Y=93, όπου η τιμή του X είναι λάθος, ενώ το χρονοπρόγραμμα Δ δίνει σωστά αποτελέσματα.

Το χρονοπρόγραμμα Γ οδηγεί σε λάθος αποτέλεσμα λόγω του προβλήματος της απώλειας ενημέρωσης. Η δοσοληψία T2 διαβάζει την τιμή του X *πριν* αλλαχθεί από τη δοσοληψία T1, επομένως μόνον η επίδραση της T2 στο X αντικατοπτρίζεται στη βάση δεδομένων. Η επίδραση της T1 στο X *χάνεται* όταν επικαλύπτεται από την T2, οδηγώντας σε λανθασμένο αποτέλεσμα για το στοιχείο X. Ωστόσο, μερικά μη σειριακά χρονοπρογράμματα, όπως το χρονοπρόγραμμα Δ, δίνουν το αναμενόμενο σωστό αποτέλεσμα. Θα θέλαμε να προσδιορίσουμε ποια από τα μη σειριακά χρονοπρογράμματα δίνουν *πάντα* σωστό αποτέλεσμα και ποια μπορεί να οδηγήσουν σε λάθος αποτελέσματα. Για το συγκεκριμένο διαχωρισμό των χρονοπρογραμμάτων χρησιμοποιείται η έννοια της σειριοποιησιμότητας των χρονοπρογραμμάτων.

Ένα χρονοπρόγραμμα S από n δοσοληψίες είναι *σειριοποιήσιμο* αν είναι *ισοδύναμο* με *κάποιο σειριακό χρονοπρόγραμμα* των ίδιων n δοσοληψιών. Θα ορίσουμε στη συνέχεια την έννοια της ισοδυναμίας δοσοληψιών. Σημειώνουμε ότι για n δοσοληψίες υπάρχουν (n)! πιθανά σειριακά χρονοπρογράμματα και πολύ περισσότερα πιθανά μη σειριακά χρονοπρογράμματα. Μπορούμε να σχηματίσουμε δύο ξένες μεταξύ τους ομάδες μη σειριακών χρονοπρογραμμάτων: αυτά που είναι ισοδύναμα με ένα σειριακό χρονοπρόγραμμα (ή και περισσότερα) και επομένως είναι σειριοποιήσιμα και αυτά που δεν είναι ισοδύναμα με κάποιο σειριακό χρονοπρόγραμμα και επομένως δεν είναι σειριοποιήσιμα.

Ο χαρακτηρισμός ενός μη σειριακού χρονοπρογράμματος ως σειριοποιήσιμου είναι ισοδύναμος με το χαρακτηρισμό του ως σωστού, διότι είναι ισοδύναμο με κάποιο σειριακό χρονοπρόγραμμα το οποίο θεωρείται σωστό. Πότε όμως δύο χρονοπρογράμματα θεωρούνται «ισοδύναμα»; Υπάρχουν διάφοροι τρόποι ορισμού της ισοδυναμίας χρονοπρογραμμάτων. Ο απλούστερος, αλλά και λιγότερο ικανοποιητικός, ορισμός της ισοδυναμίας χρονοπρογραμμάτων συνίσταται στη σύγκριση της επίδρασης των χρονοπρογραμμάτων στη βάση δεδομένων. Δύο χρονικά καλούνται *ισοδυνάμου αποτελέσματος* αν παράγουν την ίδια τελική κατάσταση της βάσης δεδομένων. Ωστόσο, είναι δυνατόν δύο διαφορετικά χρονοπρογράμματα να παράγουν τυχαία την ίδια τελική κατάσταση.



Εικόνα 4.6: Δυο χρονοπρογράμματα που είναι ισοδύναμα για αρχική τιμή X=100, αλλά γενικά δεν είναι ισοδύναμα

Για παράδειγμα, στην Εικόνα 4.6 τα χρονοπρογράμματα S1 και S2 θα οδηγήσουν στην ίδια τελική κατάσταση της βάσης δεδομένων αν η αρχική τιμή του X είναι 100, ενώ για οποιαδήποτε άλλη αρχική τιμή του X τα χρονοπρογράμματα *δεν* είναι ισοδυνάμου αποτελέσματος. Επιπλέον, τα δύο αυτά χρονοπρογράμματα εκτελούν διαφορετικές δοσοληψίες, επομένως είναι σαφές ότι δεν πρέπει να θεωρηθούν ισοδύναμα. Συνεπώς, η ισοδυναμία αποτελέσματος *δεν* χρησιμοποιείται για τον ορισμό ισοδυναμίας χρονοπρογραμμάτων.

Η ασφαλέστερη και γενικότερη προσέγγιση για τον ορισμό της ισοδυναμίας χρονοπρογραμμάτων είναι να μην γίνει καμία υπόθεση για τους τύπους των πράξεων που περιλαμβάνονται στις δοσοληψίες. Για να είναι δύο χρονοπρογράμματα ισοδύναμα, οι πράξεις που εφαρμόζονται σε κάθε στοιχειώδες δεδομένο που επηρεάζεται από τα χρονοπρογράμματα πρέπει να εφαρμόζονται σ' αυτό *με την ίδια σειρά* και στα δύο χρονοπρογράμματα. Γενικά είναι παραδεκτοί δύο ορισμοί της ισοδυναμίας χρονοπρογραμμάτων: *ισοδυναμία συγκρούσεων και ισοδυναμία όψεων.* Και οι δύο ορισμοί εξετάζονται στη συνέχεια.

Δύο χρονοπρογράμματα καλούνται *ισοδυνάμων συγκρούσεων* αν η διάταξη κάθε ζεύγους *συγκρουόμενων πράξεων* είναι η ίδια και στα δύο χρονοπρογράμματα. Υπενθυμίζουμε τον ορισμό ότι δύο πράξεις σε ένα χρονοπρόγραμμα είναι *συγκρουόμενες αν* ανήκουν σε διαφορετικές δοσοληψίες, προσπελαύνουν το ίδιο στοιχειώδες δεδομένο της βάσης δεδομένων και αν τουλάχιστον μία από τις δύο πράξεις είναι η εγγραφή_στοιχείου. Αν δύο πράξεις συγκρουόμενες εφαρμοσθούν με *διαφορετική σειρά* σε δύο χρονοπρογράμματα η επίδραση των χρονοπρογραμμάτων μπορεί να είναι διαφορετική, είτε στις δοσοληψίες είτε στη βάση δεδομένων και επομένως τα χρονοπρογράμματα δεν είναι ισοδυνάμων συγκρούσεων. Για παράδειγμα, αν μια πράξη ανάγνωσης και μια πράξη εγγραφής εμφανίζονται με τη σειρά r1(X), w2(X) σε ένα χρονοπρόγραμμα Sα και με αντίθετη σειρά w2(X), r1(X) σε ένα χρονοπρόγραμμα Sβ, η τιμή που διαβάζεται από την πράξη r1(X) μπορεί να είναι διαφορετική στα δύο χρονοπρογράμματα. Παρομοίως, αν δύο πράξεις εγγραφής εμφανίζονται με τη σειρά w1(X), w2(X) σε ένα χρονοπρόγραμμα Sα και με αντίθετη σειρά w2(X), w1(X) σε ένα άλλο χρονοπρόγραμμα Sβ, η επόμενη πράξη r(X) στα δύο χρονοπρογράμματα θα διαβάσει ενδεχομένως διαφορετικές τιμές ή, αν αυτές είναι οι τελευταίες πράξεις προσπέλασης του στοιχείου X στα δύο χρονοπρογράμματα, η τελική τιμή του στοιχείου X θα είναι διαφορετική για τα δύο χρονοπρογράμματα.

Χρησιμοποιώντας την έννοια της ισοδυναμίας συγκρούσεων, ορίζουμε ότι ένα χρονοπρόγραμμα S είναι *σειριοποιήσιμο βάσει συγκρούσεων* αν είναι ισοδύναμο (βάσει συγκρούσεων) με κάποιο σειριακό χρονοπρόγραμμα S΄. Σε μια τέτοια περίπτωση μπορούμε να αναδιατάξουμε τις μη συγκρουόμενες πράξεις στο S μέχρι

να σχηματίσουμε το ισοδύναμο σειριακό χρονοπρόγραμμα S΄. Σύμφωνα με τον ορισμό αυτό, το χρονοπρόγραμμα Δ της Εικόνας 4.5(γ) είναι ισοδύναμο με το σειριακό χρονοπρόγραμμα Α της Εικόνας 4.5(α). Και στα δύο αυτά χρονοπρογράμματα η πράξη ανάγνωση_στοιχείου(Χ) της Τ2 διαβάζει την τιμή του Χ που καταχωρείται από την Τ1, ενώ οι άλλες πράξεις ανάγνωση_στοιχείου διαβάζουν τις τιμές της αρχικής κατάστασης της βάσης δεδομένων. Επιπλέον, η Τ1 είναι η τελευταία δοσοληψία που γράφει το στοιχείο Υ και η Τ2 είναι η τελευταία δοσοληψία που γράφει το στοιχείο Χ και στα δύο χρονοπρογράμματα. Δεδομένου ότι το Α είναι ένα σειριακό χρονοπρόγραμμα και το Δ είναι ισοδύναμο με το Α, το Δ είναι ένα *σειριοποιήσιμο χρονοπρόγραμμα.* Σημειώνουμε ότι οι πράξεις r1(Y) και w1(Y) του χρονοπρογράμματος Δ δεν συγκρούονται με τις πράξεις r2(X) και w2(X), διότι προσπελαύνουν διαφορετικά στοιχειώδη δεδομένα. Επομένως μπορούμε να μετακινήσουμε τις πράξεις r1(Y), w1(Y) πριν από τις r2(X), w2(X), καταλήγοντας στο ισοδύναμο σειριακό χρονοπρόγραμμα Τ1, Τ2.

Το χρονοπρόγραμμα Γ της Εικόνας 4.5(γ) δεν είναι ισοδύναμο με κανένα από τα δύο δυνατά σειριακά χρονοπρογράμματα Α και Β. Δεν είναι ισοδύναμο με το Α διότι η πράξη ανάγνωση_στοιχείου(Χ) στο Τ2 διαβάζει την τιμή που καταχωρείται από την Τ1 στο χρονοπρόγραμμα Α, ενώ στο χρονοπρόγραμμα Γ διαβάζει την αρχική τιμή του Χ, παραβιάζοντας έτσι τον πρώτο κανόνα ισοδυναμίας. Επίσης, το χρονοπρόγραμμα Γ δεν είναι ισοδύναμο με το Β, επομένως το χρονοπρόγραμμα Γ *δεν είναι σειριοποιήσιμο.* Αν προσπαθήσουμε να αναδιατάξουμε τις πράξεις του χρονοπρογράμματος Γ για να καταλήξουμε σε ένα ισοδύναμο σειριακό χρονοπρόγραμμα θα αποτύχουμε, διότι οι πράξεις r2(X) και w1(X) είναι συγκρουόμενες και έτσι δεν μπορούμε να μετακινήσουμε την πράξη r1(X) προς τα κάτω για να οδηγηθούμε στο σειριακό χρονοπρόγραμμα Τ1, Τ2. Παρομοίως, λόγω του ότι οι πράξεις w1(X) και w2(X) είναι συγκρουόμενες, δεν μπορούμε να μετακινήσουμε την πράξη w1(X) προς τα κάτω ώστε να οδηγηθούμε στο σειριακό χρονοπρόγραμμα Τ2, Τ1.

4.5.2 Έλεγχος ενός Χρονοπρογράμματος για Σειριοποιησιμότητα βάσει Συγκρούσεων

Υπάρχει ένας απλός αλγόριθμος για να προσδιορισθεί αν ένα χρονοπρόγραμμα είναι σειριοποιήσιμο βάσει συγκρούσεων. Στην πραγματικότητα, οι περισσότερες μέθοδοι ελέγχου συνδρομικότητας *δεν* ελέγχουν για σειριοποιησιμότητα. Αντί ελέγχου, αναπτύσσονται πρωτόκολλα ή κανόνες που εγγυώνται ότι ένα χρονοπρόγραμμα θα είναι σειριοποιήσιμο. Εξετάζουμε εδώ τον αλγόριθμο ελέγχου των χρονοπρογραμμάτων για σειριοποιησιμότητα βάσει συγκρούσεων.

Ο αλγόριθμος 4.1 μπορεί να χρησιμοποιηθεί για να ελεγχθεί αν ένα χρονοπρόγραμμα είναι σειριοποιήσιμο βάσει συγκρούσεων. Ο αλγόριθμος θεωρεί μόνον τις πράξεις ανάγνωση_στοιχείου και εγγραφή_στοιχείου στο χρονοπρόγραμμα για να κατασκευάσει ένα *γράφο προήγησης* (precedence graph) ή *γράφο σειριοποίησης* (serialization graph). Ο γράφος προήγησης είναι ένας *κατευθυνόμενος γράφος* G=(N,E) ο οποίος αποτελείται από ένα σύνολο κόμβων N={Tl,T2,..., Tn} και ένα σύνολο ακμών E= {α1,α2, ..., αm}. Υπάρχει ένας κόμβος στο γράφο για κάθε δοσοληψία Ti του χρονοπρογράμματος. Κάθε ακμή αi στο γράφο έχει τη μορφή (Tj→Tk), l≤j≤n, l≤k≤n, όπου ο κόμβος Tj καλείται *αρχικός κόμβος* της αi και ο Tk καλείται *τελικός κόμβος* της αi. Η ύπαρξη μιας ακμής (Tj→Tk) στο γράφο σημαίνει

ότι μια από τις πράξεις της Tj εμφανίζεται στο χρονοπρόγραμμα πριν από κάποια πράξη της Tk, με την οποία *συγκρούεται*.

Αλγόριθμος 4.1: Έλεγχος σειριοποιησιμότητας βάσει συγκρούσεων ενός χρονοπρογράμματος S.

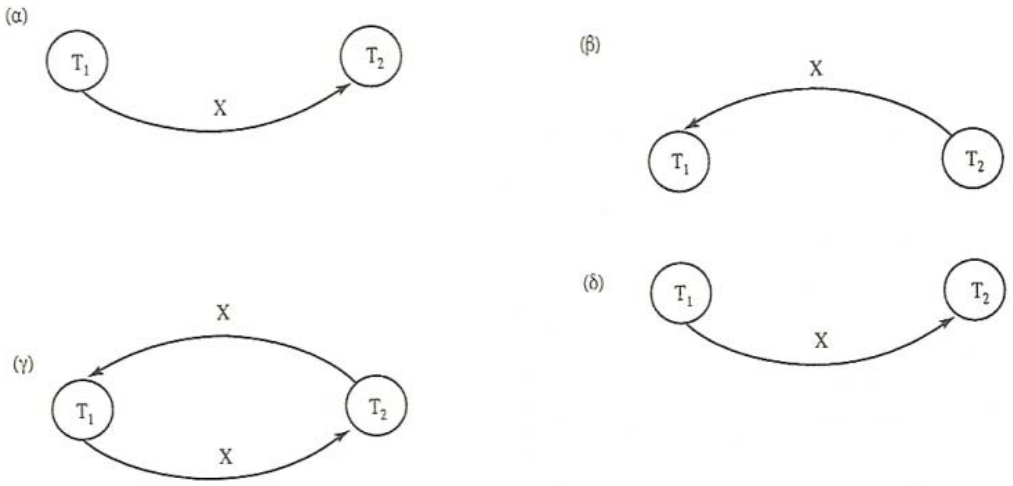
1. Για κάθε δοσοληψία Ti που συμμετέχει στο χρονοπρόγραμμα S δημιούργησε έναν κόμβο Ti στο γράφο προτεραιοτήτων.
2. για κάθε περίπτωση στο S όπου η Tj εκτελεί μια πράξη ανάγνωση_στοιχείου(X) μετά από μια εντολή εγγραφή_στοιχείου(X) που εκτελείται από την Ti δημιούργησε μια ακμή (Ti→Tj) στο γράφο προήγησης.
3. για κάθε περίπτωση στο S όπου η Tj εκτελεί μια πράξη εγγραφή_στοιχείου(X) μετά από μια εντολή ανάγνωση_στοιχείου(X) που εκτελείται από την Ti δημιούργησε μια ακμή (Ti→Tj) στο γράφο προήγησης.
4. για κάθε περίπτωση στο S όπου η Tj εκτελεί μια εντολή εγγραφή_στοιχείου(X) αφού η Ti εκτελέσει μια εντολή εγγραφή_στοιχείου(X) δημιούργησε μια ακμή (Ti→Tj) στο γράφο προήγησης.
5. Το χρονοπρόγραμμα S είναι σειριοποιήσιμο αν και μόνον αν ο γράφος προήγησης δεν έχει κυκλώματα.

Ο γράφος προήγησης κατασκευάζεται σύμφωνα με τον αλγόριθμο 4.1. Αν υπάρχει κύκλωμα στο γράφο προήγησης, το χρονοπρόγραμμα S δεν είναι σειριοποιήσιμο (βάσει συγκρούσεων), ενώ αν δεν υπάρχει κύκλωμα το S το χρονοπρόγραμμα είναι σειριοποιήσιμο. *Κύκλωμα* σε έναν κατευθυνόμενο γράφο είναι μια *ακολουθία* ακμών C=((Tj→Tk), (Tk→Tp),...,(Ti→Tj)) με την ιδιότητα ότι ο αρχικός κόμβος κάθε ακμής -εκτός της αρχικής ακμής- είναι ίδιος με τον τελικό κόμβο της προηγούμενης ακμής και ο αρχικός κόμβος της πρώτης ακμής είναι ο ίδιος με τον τελικό κόμβο της τελευταίας ακμής. Επομένως, η ακολουθία ξεκινάει και τερματίζει στον ίδιο κόμβο.

Στο γράφο προήγησης, μια ακμή από τον κόμβο Ti στον Tj σημαίνει ότι η δοσοληψία Ti πρέπει να προηγείται της Tj σε κάθε σειριακό χρονοπρόγραμμα που είναι ισοδύναμο με το S, διότι κάθε ζεύγος συγκρουόμενων πράξεων εμφανίζεται στο χρονοπρόγραμμα με τη συγκεκριμένη σειρά. Αν δεν υπάρχει κύκλωμα στο γράφο προήγησης, μπορούμε να δημιουργήσουμε ένα *ισοδύναμο σειριακό χρονοπρόγραμμα* S΄ το οποίο είναι ισοδύναμο με το S, διατάσσοντας με τον ακόλουθο τρόπο τις δοσοληψίες που συμμετέχουν στο S: όταν υπάρχει μια ακμή στο γράφο προήγησης από τον κόμβο Ti στον Tj, η δοσοληψία Ti πρέπει να προηγείται της Tj στο ισοδύναμο σειριακό χρονοπρόγραμμα S΄. Αυτή η διαδικασία είναι γνωστή ως τοπολογική ταξινόμηση. Σημειώνουμε ότι σε κάθε ακμή (Ti→Tj) του γράφου προήγησης μπορεί προαιρετικά να επισυναφθεί ως ετικέτα το όνομα (ή τα ονόματα) του δεδομένου (των δεδομένων) που οδήγησε(αν) στη δημιουργία της ακμής.

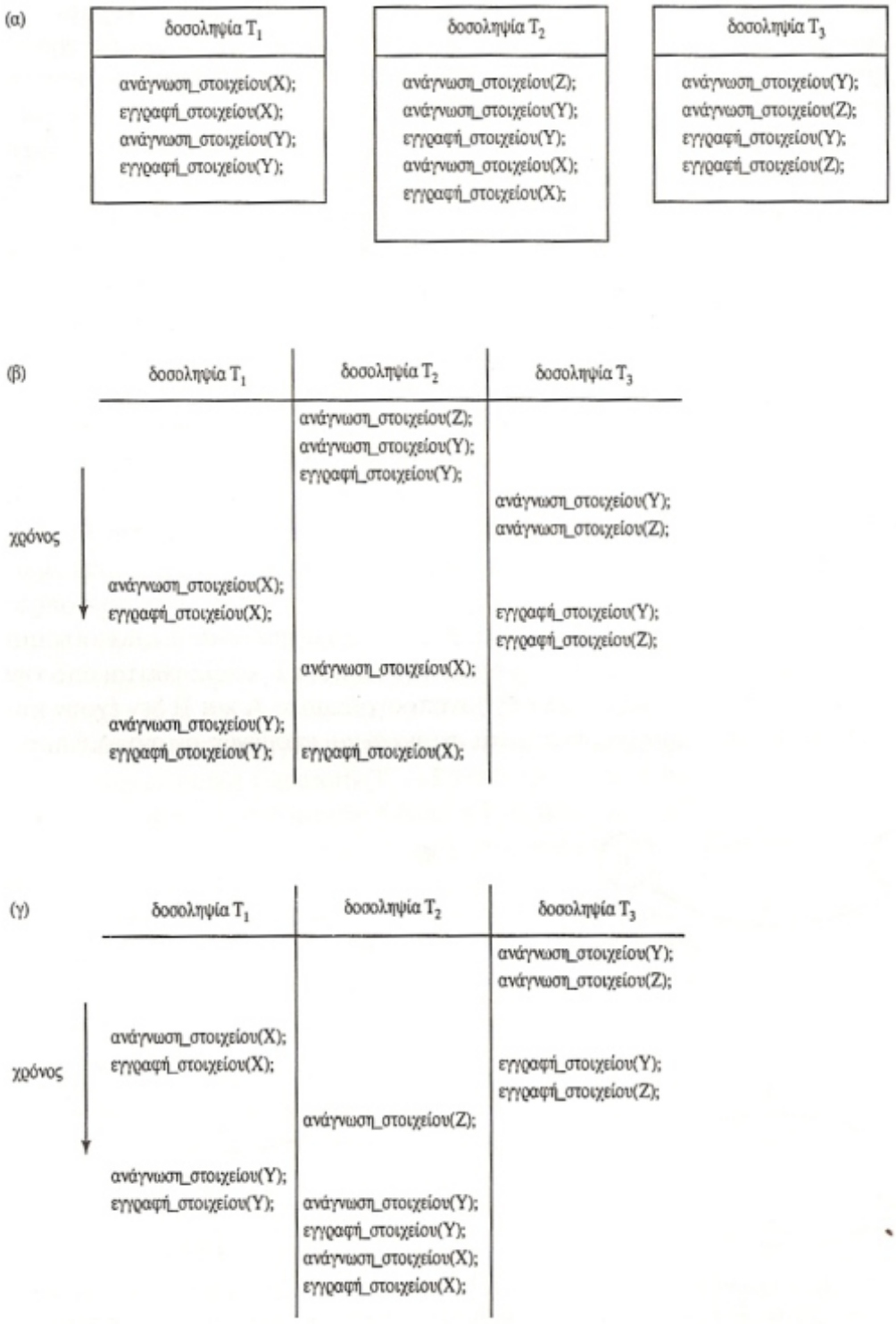
Γενικά, διάφορα σειριακά χρονοπρογράμματα μπορούν να είναι ισοδύναμα με το S αν ο γράφος προήγησης δεν έχει κυκλώματα. Αντιθέτως, αν ο γράφος προήγησης έχει ένα κύκλωμα, είναι εύκολο να δείξουμε ότι δεν μπορούμε να δημιουργήσουμε κανένα ισοδύναμο σειριακό χρονοπρόγραμμα, συνεπώς το χρονοπρόγραμμα S δεν είναι σειριοποιήσιμο. Οι γράφοι προήγησης που δημιουργήθηκαν για τα χρονοπρογράμματα Α έως Δ της Εικόνας 4.5, αντίστοιχα, παρουσιάζονται στην Εικόνα 4.7(α) έως (δ). Ο γράφος προήγησης για το χρονοπρόγραμμα Γ έχει ένα κύκλωμα, συνεπώς το χρονοπρόγραμμα δεν είναι σειριοποιήσιμο. Ο γράφος για το χρονοπρόγραμμα Δ δεν έχει κυκλώματα, συνεπώς το χρονοπρόγραμμα είναι σειριοποιήσιμο και στο ισοδύναμο σειριακό χρονοπρόγραμμα η δοσοληψία Τι

ακολουθείται από την Τ2. Όπως αναμένεται, οι γράφοι για τα χρονοπρογράμματα Α και Β δεν έχουν κυκλώματα διότι τα χρονοπρογράμματα είναι *σειριακά* και επομένως σειριοποιήσιμα.

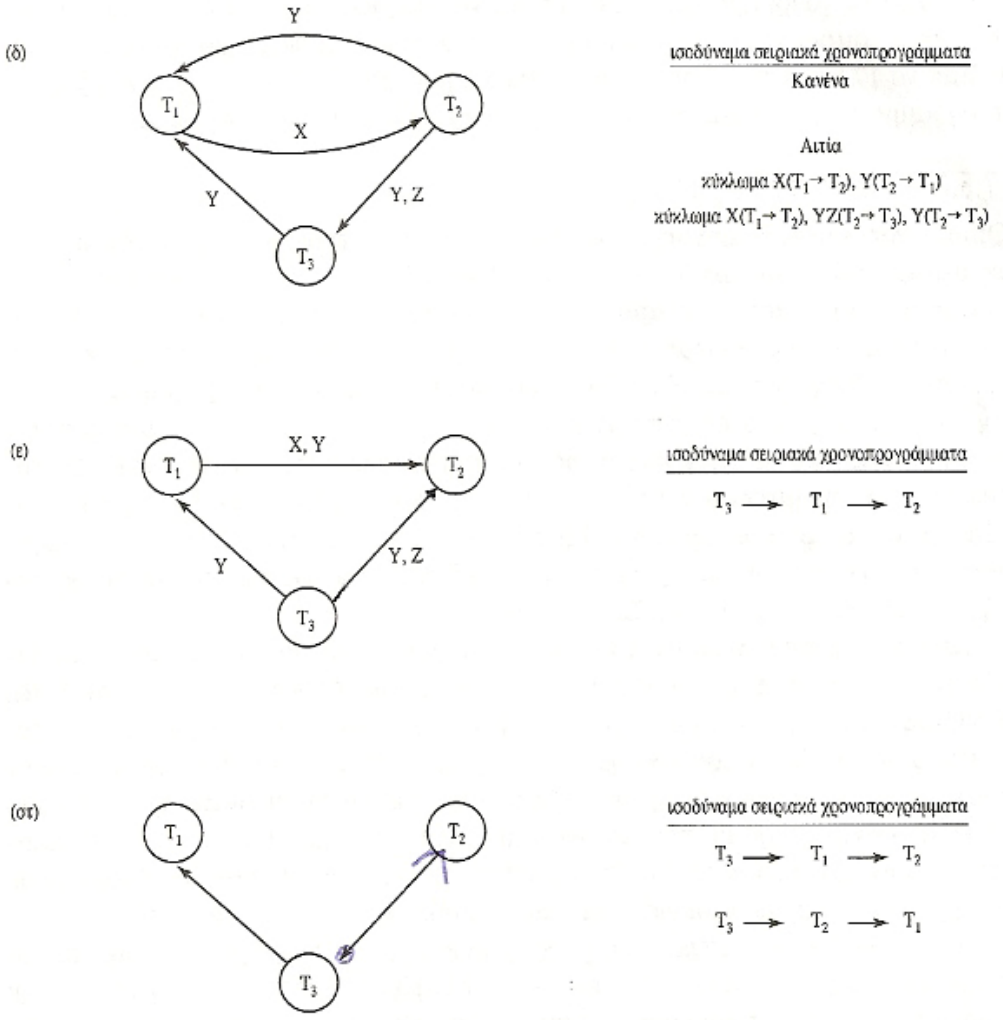


Εικόνα 4.7: Κατασκευή των γράφων προήγησης για τα χρονοπρογράμματα Α έως Δ, προκειμένου να ελεγχθεί η σειριοποιησιμότητα βάσει συγκρούσεων. (α) γράφος προήγησης για το χρονοπρόγραμμα Α της Εικόνας 4.5 (β) γράφος προήγησης για το χρονοπρόγραμμα Β της Εικόνας 4.5 (γ) γράφος προήγησης για το χρονοπρόγραμμα Γ της Εικόνας 4.5 (δ) γράφος προήγησης για το χρονοπρόγραμμα Δ της Εικόνας 4.5 (σειριοποιήσιμο, ισοδύναμο με το χρονοπρόγραμμα Α).

Ένα άλλο παράδειγμα, στο οποίο συμμετέχουν τρεις δοσοληψίες, φαίνεται στην Εικόνα 4.8. Η Εικόνα 4.8(a) παρουσιάζει τις πράξεις ανάγνωση_στοιχείου και εγγραφή_στοιχείου κάθε δοσοληψίας. Για τις δοσοληψίες αυτές, δύο χρονοπρογράμματα Ε και Στ παρουσιάζονται στην Εικόνα 4.8(β) και (γ), αντίστοιχα. Οι γράφοι προήγησης για τα χρονοπρογράμματα Ε και Στ παρουσιάζονται στην Εικόνα 4.8(δ) και (ε). Το χρονοπρόγραμμα Ε δεν είναι σειριοποιήσιμο διότι στον αντίστοιχο γράφο προήγησης εμφανίζεται ένα κύκλωμα. Το χρονοπρόγραμμα Στ είναι σειριοποιήσιμο και η Εικόνα 4.8(ε) απεικονίζει το σειριακό χρονοπρόγραμμα που είναι ισοδύναμο με το Στ. Αν και υπάρχει μόνον ένα σειριακό ισοδύναμο για το χρονοπρόγραμμα Στ, γενικά μπορεί να υπάρχουν περισσότερα από ένα ισοδύναμα σειριακά χρονοπρογράμματα για ένα σειριοποιήσιμο χρονοπρόγραμμα. Η Εικόνα 4.8(στ) παρουσιάζει ένα γράφο προήγησης που παριστάνει ένα χρονοπρόγραμμα το οποίο έχει δύο ισοδύναμα σειριακά χρονοπρογράμματα.



Εικόνα 4.8: Ένα άλλο παράδειγμα ελέγχου σειριοποιησιμότητας. (α) Οι πράξεις ΑΝΑΓΝΩΣΗ και ΕΓΓΡΑΦΗ τριών δοσοληψιών (β) το χρονοπρόγραμμα Ε των δοσοληψιών Τ1 , Τ2 και Τ3 (γ) το χρονοπρόγραμμα ΣΤ των δοσοληψιών Τ1 , Τ2 και Τ3.



Εικόνα 4.8: (δ) γράφος προήγησης για το χρονοπρόγραμμα Ε (ε) Γράφος προήγησης για το χρονοπρόγραμμα ΣΤ. (στ) ένας γράφος προήγησης με δυο ισοδύναμα σειριακά χρονοπρογράμματα.

4.5.3 Χρήσεις της Σειριοποιησιμότητας

Όπως αναφέρθηκε προηγουμένως, ο χαρακτηρισμός ενός χρονοπρογράμματος S ως σειριοποιήσιμου (βάσει συγκρούσεων) -με άλλα λόγια ως ισοδύναμου (βάσει συγκρούσεων) με κάποιο σειριακό χρονοπρόγραμμα- είναι ταυτόσημος με το χαρακτηρισμό του ως σωστού. Ωστόσο, οι χαρακτηρισμοί *σειριοποιήσιμο* και *σειριακό* είναι διαφορετικοί. Η επεξεργασία ενός σειριακού χρονοπρογράμματος δεν είναι αποδοτική, διότι δεν επιτρέπεται διαπλοκή των πράξεων των διαφορετικών δοσοληψιών. Αυτό μπορεί να οδηγήσει σε χαμηλά ποσοστά χρήσης της ΚΜΕ στα διαστήματα που μια δοσοληψία περιμένει την ολοκλήρωση κάποιας πράξεως εισόδου / εξόδου στο δίσκο, με αποτέλεσμα τη σημαντική επιβράδυνση της επεξεργασίας. Ένα σειριοποιήσιμο χρονοπρόγραμμα έχει τα πλεονεκτήματα της συνδρομικής εκτέλεσης, χωρίς να θυσιάζει την ορθότητα.

Στην πράξη είναι δύσκολο να ελεγχθεί ένα χρονοπρόγραμμα για σειριοποιησιμότητα. Η διαπλοκή των πράξεων των ταυτόχρονων δοσοληψιών αποφασίζεται συνήθως από το χρονοπρογραμματιστή του λειτουργικού συστήματος. Παράγοντες όπως ο φόρτος του συστήματος, ο χρόνος υποβολής της δοσοληψίας και οι προτεραιότητες των

δοσοληψιών λαμβάνονται υπόψη για τη διάταξη των πράξεων ενός χρονοπρογράμματος από το λειτουργικό σύστημα. Επομένως, είναι πρακτικά αδύνατον να προσδιορισθεί εκ των προτέρων η διαπλοκή των πράξεων σε ένα χρονοπρόγραμμα, έτσι ώστε να εξασφαλισθεί η σειριοποιησιμότητα.

Αν οι δοσοληψίες εκτελούνται αυθαίρετα και το χρονοπρόγραμμα που προκύπτει ελέγχεται για σειριοποιησιμότητα εκ των υστέρων, πρέπει να ακυρώσουμε την επίδραση του χρονοπρογράμματος αν διαπιστωθεί ότι δεν είναι σειριοποιήσιμο. Αυτό είναι ένα σοβαρό πρόβλημα που καθιστά τη μέθοδο αυτή πρακτικά άχρηστη. Συνεπώς, η προσέγγιση που ακολουθείται στις περισσότερες υλοποιήσεις συστημάτων είναι να προσδιορίζονται μέθοδοι που εξασφαλίζουν τη σειριοποιησιμότητα, χωρίς να απαιτείται ο έλεγχος των χρονοπρογραμμάτων για σειριοποιησιμότητα μετά την εκτέλεσή τους. Μια τέτοια μέθοδος χρησιμοποιεί τη θεωρία της σειριοποιησιμότητας για να προσδιορίσει *πρωτόκολλα* ή σύνολα κανόνων που, αν ακολουθηθούν από *κάθε* δοσοληψία ή αν επιβληθούν από το υποσύστημα ελέγχου συνδρομικότητας ενός ΣΔΒΔ, θα εξασφαλίσουν τη σειριοποιησιμότητα *όλων των χρονοπρογραμμάτων στα οποία συμμετέχουν οι δοσοληψίες*. Επομένως, ακολουθώντας αυτή την προσέγγιση, δεν χρειάζεται να ασχοληθούμε με τα χρονοπρογράμματα.

Ένα άλλο πρόβλημα είναι ότι, καθώς υποβάλλονται συνεχώς δοσοληψίες στο σύστημα, είναι δύσκολο να προσδιορισθεί πότε αρχίζει και πότε τελειώνει ένα χρονοπρόγραμμα. Η θεωρία της σειριοποιησιμότητας μπορεί να προσαρμοσθεί έτσι ώστε να αντιμετωπίζει αυτό το πρόβλημα, εξετάζοντας μόνο την επικυρωμένη προβολή ενός χρονοπρογράμματος S. Υπενθυμίζουμε τον ορισμό, σύμφωνα με τον οποίο η επικυρωμένη προβολή C(S) ενός χρονοπρογράμματος S περιλαμβάνει μόνο τις πράξεις του S που ανήκουν σε επικυρωμένες δοσοληψίες. Μπορούμε να ορίσουμε ότι ένα χρονοπρόγραμμα S είναι σειριοποιήσιμο αν η επικυρωμένη προβολή του, C(S), είναι ισοδύναμη με κάποιο σειριακό χρονοπρόγραμμα, εφόσον μόνον οι επικυρωμένες δοσοληψίες τελούν υπό την εγγύηση του ΣΔΒΔ.

Ένας άλλος παράγοντας που επηρεάζει τον έλεγχο συνδρομικότητας είναι η *διακριτότητα* των στοιχείων -δηλαδή τι μέρος της βάσης δεδομένων αντιπροσωπεύει ένα στοιχείο. Το μέγεθος ενός στοιχείου μπορεί να είναι από πολύ μικρό -π.χ. μια μεμονωμένη τιμή- έως πολύ μεγάλο -π.χ. ολόκληρη η βάση δεδομένων. Φυσικά, στην τελευταία περίπτωση επιτρέπεται πολύ περιορισμένη συνδρομικότητα.

5 Ανάγκη για ευφυή πληροφοριακά συστήματα

Τα τελευταία χρόνια τα υπολογιστικά συστήματα έχουν εισχωρήσει στη ζωή μας, τόσο στον επαγγελματικό τομέα όσο και σε προσωπικό επίπεδο, και αυτό το φαινόμενο θα είναι ακόμα πιο έντονο τα επόμενα χρόνια. Ήδη παίζουν σημαντικό ρόλο σε πλήθος ιδιωτικών επιχειρήσεων, σε δημόσιους οργανισμούς, σε τράπεζες, στη διαχείριση και κατανομή ηλεκτρικής ενέργειας, στις τηλεπικοινωνίες, στον στρατιωτικό τομέα, στον τομέα υγείας, στην εκπαίδευση κ.ά. Ωστόσο η συσσώρευση ολοένα και μεγαλύτερου όγκου πληροφορίας, οι συνεχείς και πολλές φορές απρόβλεπτες αλλαγές του περιβάλλοντος λειτουργίας, οι συνεχώς αυξανόμενες απαιτήσεις και διαφορετικές ανάγκες των χρηστών αλλά και πλήθος άλλων παραγόντων διαφορετικών μεταξύ τους και ξεχωριστών για κάθε τομέα εφαρμογής, καθιστούν αναγκαία τη δυνατότητα προσαρμογής των υπολογιστικών συστημάτων στις νέες συνθήκες λειτουργίας και στις νέες ανάγκες που προκύπτουν. Είναι αναγκαία, δηλαδή, η ανάπτυξη υπολογιστικών συστημάτων με ευφυή συμπεριφορά.

Η Τεχνητή Νοημοσύνη (Artificial Intelligence) είναι ο κλάδος της επιστήμης των υπολογιστών, ο οποίος μελετά τα φαινόμενα ευφυούς συμπεριφοράς χρησιμοποιώντας εργαλεία θεωρητικά και πειραματικά, με στόχο την κατασκευή ευφυών συστημάτων.

Αν και ο όρος «ευφυία» έχει αποτελέσει αντικείμενο συζήτησης πολλών φιλοσόφων και ερευνητών, μία ανάλυσή του ξεφεύγει από τους στόχους του παρόντος. Ωστόσο, ένας γενικός ορισμός σχετικά με το τι είναι ευφυία που καλύπτει τις ανάγκες του συγγράμματος είναι αυτός του J. McCarthy, σύμφωνα με τον οποίο «ευφυία είναι το υπολογιστικό μέρος της ικανότητας για την επίτευξη στόχων στον κόσμο» [3].

Η έρευνα στην Τεχνητή Νοημοσύνη τόσο σε θεωρητικό όσο και πειραματικό επίπεδο έχει δώσει πλήθος μεθόδων και τεχνικών που βοηθούν στην κατανόηση των μηχανισμών εκδήλωσης ευφυούς συμπεριφοράς για συγκεκριμένα έργα. Η ίδια έρευνα έχει καταστήσει δυνατή την ανάπτυξη ευφυών συστημάτων σε ένα ευρύ φάσμα εφαρμογών, από ελεγκτές εναέριας κυκλοφορίας έως οικιακές συσκευές.

Σε μία έκθεση της Αμερικανικής Εταιρείας Τεχνητής Νοημοσύνης (American Association for Artificial Intelligence) [1], γίνεται λόγος για τους τρόπους με τους οποίους η τεχνολογία που έχει αναπτυχθεί στην Τεχνητή Νοημοσύνη μπορεί να δώσει νέες και σημαντικές δυνατότητες σε πλήθος συστημάτων σε όλους τους τομείς της δημόσιας και ιδιωτικής δραστηριότητας, δυνατότητες οι οποίες θα έχουν επίδραση στη βελτίωση των προσφερομένων υπηρεσιών, την αύξηση της παραγωγικότητας και την αναβάθμιση της ποιότητας ζωής. Έτσι αναφέρεται η ανάγκη για επένδυση σε τέσσερις μεγάλες οικογένειες εφαρμογών ευρείας κλίμακας:

- Ευφυή Συστήματα Προσομοίωσης
- Ευφυή Συστήματα Πληροφοριών
- Ευφυή Συστήματα Υποστήριξης
- Ευφυή Robots

5.1 Ευφυή Συστήματα Προσομοίωσης

Τα συστήματα προσομοίωσης είναι συστήματα που αναπαριστούν επιλεγμένα χαρακτηριστικά συμπεριφοράς ενός άλλου φυσικού ή αφηρημένου συστήματος, όπως για παράδειγμα η προσομοίωση μιας κατάστασης ανάγκης σε μία αεροπορική πτήση. Αν και έχουν αναπτυχθεί αρκετά συστήματα προσομοίωσης μέχρι σήμερα, αυτά παρουσιάζουν περιορισμένες δυνατότητες στη ρεαλιστική αναπαράσταση των φυσικών αντικειμένων και έλλειψη δυνατοτήτων στη ρεαλιστική αναπαράσταση ανθρώπων.

Η χρήση των τεχνικών της Τεχνητής Νοημοσύνης μπορεί να υποστηρίξει την ανάπτυξη προγραμμάτων για τη μοντελοποίηση σύνθετων καταστάσεων και να δώσει ακριβείς αναπαραστάσεις στη συμπεριφορά των μερών που συμμετέχουν σε μία προσομοίωση μεγάλης κλίμακας. Τα διάφορα μέρη που συμμετέχουν μπορούν να σχεδιάζουν και να ενεργούν αυτόνομα, να συνεργάζονται μεταξύ τους, να συντονίζουν τις ενέργειές τους προκειμένου να επιτύχουν κάποιο κοινό στόχο, να αποκτούν εμπειρία από προηγούμενες καταστάσεις και να προσαρμόζουν τη συμπεριφορά τους και τις ενέργειές τους ανάλογα με τις συνθήκες του περιβάλλοντος όπου λειτουργούν. Η δυνατότητα των ευφυών συστημάτων προσομοίωσης να παράγουν ακριβείς και ρεαλιστικούς κόσμους τα καθιστά πολύτιμα σε εφαρμογές κατάρτισης και εκπαίδευσης ομάδων ατόμων σε πραγματικές καταστάσεις, όπως και σε βιομηχανικές και στρατιωτικές εφαρμογές.

Στον τομέα της κατάρτισης, στελέχη επιχειρήσεων μπορούν να εκπαιδευτούν σε καταστάσεις λήψης αποφάσεων, καταστάσεις κατά τις οποίες μία στρατηγική απόφασή τους θα μπορούσε να αποβεί καταστροφική οικονομικά για την επιχείρηση. Μία ομάδα γιατρών μπορεί να εκπαιδευτεί στην περίθαλψη ασθενών με σπάνιες ή περίπλοκες ασθένειες και να δει τα αποτελέσματα των διαφόρων θεραπευτικών αγωγών που ακολουθήθηκαν. Μία χειρουργική ομάδα μπορεί να εκπαιδευτεί σε μία δύσκολη εγχείριση κατά την οποία είναι πιθανόν να ανακύψουν διάφορες περιπλοκές. Χειριστές πολύπλοκων και ακριβών συσκευών και μηχανημάτων μπορούν να εκπαιδευτούν στη χρήση τους σε πραγματικές και απρόβλεπτες καταστάσεις. Στρατιωτικοί μπορούν να εκπαιδευτούν σε συνθήκες μάχης, να δοκιμάσουν νέες τακτικές, να βελτιώσουν την απόδοση των κινήσεων ανάλογα με τις συνθήκες του περιβάλλοντος των επιχειρήσεων και να παρακολουθήσουν τα αποτελέσματα των αποφάσεών τους.

Στον τομέα της εκπαίδευσης, οι μαθητές μπορούν να μεταφερθούν σε διάφορες ιστορικές εποχές και να χρησιμοποιήσουν τα υπάρχοντα μέσα της εποχής για να κατασκευάσουν γεωργικά εργαλεία, ή να εξασκηθούν σε μία ξένη γλώσσα, για παράδειγμα, σε ένα περιβάλλον που προσομοιώνει ένα εμπορικό κέντρο μιας πόλης της αντίστοιχης χώρας στο οποίο πρέπει να έρθουν σε επαφή με πολίτες, εμπόρους, υπαλλήλους, οδηγούς ταξί, κ.ά.

Στον τομέα των βιομηχανικών και στρατιωτικών εφαρμογών, ένα προϊόν (π.χ. ένα αεροπλάνο, ένα αυτοκίνητο, μία ιατρική συσκευή κτλ.), αφού σχεδιαστεί, μπορεί να δοκιμαστεί σε ένα προσομοιωμένο περιβάλλον ως προς την απόδοσή του, τη συμπεριφορά του, την αξιοπιστία του, την ευχρηστία του, τη συντηρησιμότητά του και να υποστεί τροποποιήσεις και βελτιώσεις, πριν ακόμα κατασκευαστεί, αποφεύγοντας έτσι ένα τεράστιο κόστος ανάπτυξης και δοκιμής πρωτοτύπων.

5.2 Ευφυή Συστήματα Πληροφοριών

Ο όγκος πληροφοριών που συσσωρεύεται στις βάσεις δεδομένων, στις ηλεκτρονικές βιβλιοθήκες ή βρίσκεται στο Διαδίκτυο (Internet) είναι ήδη τεράστιος και, όπως είναι φυσιολογικό, συνέχεια θα αυξάνεται. Το γεγονός ότι καθημερινά όλο και περισσότεροι χρήστες έχουν πρόσβαση στο Διαδίκτυο (από τη δουλειά, το σχολείο, το σπίτι) κάνει διαθέσιμες αυτές τις πληροφορίες στο ευρύ κοινό. Ωστόσο, ο κάθε χρήστης που αναζητά πληροφορίες για κάποιο ορισμένο σκοπό θα πρέπει να γνωρίζει που θα τις βρει και με ποιο τρόπο να τις ψάξει. Αν και υπάρχουν αρκετά προγράμματα αναζήτησης, τις περισσότερες φορές τα αποτελέσματα δεν εξυπηρετούν το στόχο ενώ επιπλέον οι χρήστες θα πρέπει να γνωρίζουν τη σύνταξη κάποιας —έστω και απλής— γλώσσας ερωτημάτων (query language) για τα συγκεκριμένα προγράμματα. Ένα άλλο πρόβλημα, που προκύπτει, είναι ο τρόπος με τον οποίο οι χρήστες θα ενημερώνονται για νέες πληροφορίες ή υπηρεσίες που είναι διαθέσιμες και εμπίπτουν στα ενδιαφέροντά τους.

Τα ευφυή συστήματα πληροφοριών καλούνται να καλύψουν τα παραπάνω προβλήματα υποστηρίζοντας με αποτελεσματικό τρόπο τη διαχείριση αυτού του τεράστιου όγκου πληροφοριών, σύμφωνα με τις ανάγκες των χρηστών[2]. Τέτοια συστήματα θα μπορούν να προσαρμόζονται στις ανάγκες και τις ικανότητες του καθενός, να αναζητούν τις κατάλληλες πληροφορίες για κάθε στόχο ή να αναζητούν πληροφορίες που εμπίπτουν στα ενδιαφέροντα του χρήστη και να επικοινωνούν μαζί του με απλό και φυσικό τρόπο προκειμένου να υποστηριχθούν οι χρήστες με περιορισμένες γνώσεις υπολογιστών. Επιπλέον, θα μπορούν να λαμβάνουν πρωτοβουλία και να εκτελούν απλές συνηθισμένες καθημερινές ενέργειες, θα μπορούν να ανακαλύπτουν νέες προσφερόμενες υπηρεσίες χρήσιμες για τον αντίστοιχο χρήστη και να ανακαλύπτουν άτομα με κοινά ενδιαφέροντα. Τέλος, θα μπορούν να αναγνωρίζουν το περιεχόμενο των πληροφοριών ανεξάρτητα από τη μορφή που αυτές είναι αποθηκευμένες (π.χ. κείμενο, εικόνα, video).

Για παράδειγμα, ένα ευφυές σύστημα διαχείρισης ηλεκτρονικού ταχυδρομείου θα μπορεί παρακολουθώντας και μαθαίνοντας από τις συνήθειες του χρήστη να παραλαμβάνει, ταξινομεί και παρουσιάζει τα εισερχόμενα μηνύματα ανάλογα με τη σπουδαιότητά τους. Ένα σύστημα πλοήγησης στο Διαδίκτυο θα ελέγχει τους τόπους του Παγκόσμιου Ιστού (World Wide Web sites) που επισκέπτεται πιο συχνά ο χρήστης, εξετάζοντας αν έχουν ανανεωθεί και παρουσιάζοντας στο χρήστη τις νέες ενδιαφέρουσες πληροφορίες. Ένα ευφυές σύστημα αναζήτησης θα είναι σε θέση να δέχεται ένα ευρύ φάσμα ασαφών ερωτημάτων σε διάφορες μορφές και θα μπορεί να συνεργάζεται με άλλα αντίστοιχα συστήματα ειδικού σκοπού, παρουσιάζοντας μετά το πέρας της αναζήτησης τα αποτελέσματα ομαδοποιημένα με ένα συνεκτικό τρόπο, σύμφωνα με το περιεχόμενό τους, σχολιάζοντάς τα και προτείνοντας εναλλακτικές λύσεις.

[2] Στο υπόλοιπο του κειμένου, ως «ευφυή συστήματα πληροφοριών» αναφέρονται τα συστήματα που επικεντρώνονται στην αναζήτηση και διαχείριση πληροφοριών σε αντίθεση με τα «ευφυή πληροφοριακά συστήματα» που αποτελούν γενικότερο όρο και αναφέρονται γενικά στα συστήματα της Πληροφορικής.

5.3 Ευφυή Συστήματα Υποστήριξης

Τα ευφυή συστήματα υποστήριξης είναι προγράμματα που βοηθούν στην ανάπτυξη και λειτουργία σύνθετων συστημάτων. Ο ρόλος τους είναι α) η υποστήριξη ταχείας πρωτοτυποποίησης (rapid prototyping), β) η χρήση τους ως συνθετικά περιβάλλοντα πειραματικού ελέγχου (synthetic environment testbeds) ([1], [2]), και γ) η παροχή βοήθειας για τη διαχείριση έργων (project management aid).

Ένα ευφυές σύστημα υποστήριξης ταχείας πρωτοτυποποίησης βοηθά καταλυτικά στον καθορισμό των προδιαγραφών και στο σχεδιασμό ενός συστήματος, κάνοντας χρήση τεχνικών αναπαράστασης γνώσης και εννοιών υψηλού επιπέδου. Επίσης, είναι πολύτιμο στον έλεγχο (testing) και στην επικύρωση (validation) των προδιαγραφών και του σχεδιασμού, αφού μπορεί να ανακαλύψει ασάφειες και ασυνέπειες χρησιμοποιώντας μηχανισμούς απόδειξης θεωρημάτων και αλγόριθμους λογικών συμπερασμάτων. Η ταχεία ανάπτυξη του πρωτοτύπου μπορεί να γίνει με τη χρήση τεχνικών αυτόματης ή ημιαυτόματης δημιουργίας κώδικα και με την επαναχρησιμοποίηση δομικών στοιχείων (modules) που ήδη ικανοποιούν προδιαγραφές παρόμοιες με το υπό ανάπτυξη σύστημα.

Η χρήση ενός ευφυούς συστήματος υποστήριξης ως συνθετικό περιβάλλον πειραματικού ελέγχου το κάνει πολύτιμο εργαλείο στον έλεγχο της λειτουργικότητας και της απόδοσης του υπό ανάπτυξη προϊόντος, στην ανακάλυψη, διάγνωση και πρόληψη αστοχιών, ενώ μπορεί να λειτουργήσει ταυτόχρονα ως σύστημα προσομοίωσης.

Τέλος, στη διαχείριση έργων, ένα ευφυές σύστημα υποστήριξης μπορεί να βοηθήσει στην καταγραφή και χρησιμοποίηση της γνώσης που αποκτάται σε όλη τη διάρκεια του έργου. Για παράδειγμα, μπορεί να καταγράψει τις αλλαγές στις διάφορες εκδόσεις των προδιαγραφών, του σχεδιασμού και της υλοποίησης ενός συστήματος, σε όλη τη διάρκεια του κύκλου ζωής του, και κατόπιν να εξηγήσει και να αιτιολογήσει τους λόγους για οποίους ελήφθησαν οι διάφορες αποφάσεις βοηθώντας στη διατήρηση της γνώσης και στη διαρκή παροχή της σε μία ομάδα ανάπτυξης που πιθανόν αλλάζει κατά διαστήματα. Κατά τον ίδιο τρόπο μπορεί να υποστηρίξει και μελλοντικά παρόμοια έργα, προσφέροντας έτοιμες λύσεις ή εμπειρία σε παρόμοια προβλήματα.

5.4 Ευφυή Robots

Τα robots είναι μηχανές που, αφού προγραμματιστούν, εκτελούν ενέργειες (χειρισμού, κίνησης κτλ.) υπό συνθήκες αυτομάτου ελέγχου. Ωστόσο, για να είναι ένα robot ευφυές πρέπει να έχει ικανότητες μεγαλύτερες και διαφορετικές από αυτές των απλών robots. Θα πρέπει να μπορεί να ενεργεί με ασφαλή και αποτελεσματικό τρόπο στο φυσικό περιβάλλον των ανθρώπων. Επίσης, αν μετακινείται, μπορεί να αναγνωρίζει αντικείμενα καθώς και ποια από αυτά είναι σταθερά ή ποια μπορούν να μετακινηθούν, ποια είναι επικίνδυνα ή ποια είναι εύθραυστα, να μπορεί να σχεδιάζει τις ενέργειές του, να αποφασίζει ποιες είναι πιο αποτελεσματικές ανάλογα με τους στόχους που πρέπει να επιτευχθούν, να αντιλαμβάνεται ήχους, χειρονομίες και φυσική γλώσσα, να μπορεί να επικοινωνήσει με τον άνθρωπο ή με άλλα robots. Επιπλέον, ένα πολύ σημαντικό χαρακτηριστικό είναι η δυνατότητα συνεργασίας με άλλα ευφυή robots. Έτσι, ενώ μεμονωμένα robots μπορεί να έχουν συγκεκριμένες και περιορισμένες ικανότητες, ενεργώντας ως ομάδες μπορεί να έχουν επίσης, την ικανότητα εκτέλεσης σύνθετων έργων και να δίνουν λύσεις σε προβλήματα όπου απαιτείται συνδυασμός γνώσης, εμπειρίας και μηχανικής ισχύος.

Τα ευφυή robots μπορούν να χρησιμοποιηθούν σε επιχειρήσεις που είναι επικίνδυνες ή αδύνατες για τον άνθρωπο όπως υπόγειες, υποβρύχιες ή διαστημικές αποστολές, καθαρισμό του περιβάλλοντος σε περιπτώσεις τοξικής ή ραδιενεργούς μόλυνσης, κατάσβεση πυρκαγιών, ή καταστροφή ναρκών, βομβών και πυρομαχικών. Μπορούν να χρησιμοποιηθούν για δουλειές οι οποίες, ενώ είναι απλές, καθημερινές και τετριμμένες για τον άνθρωπο, ωστόσο απαιτούν τη χρήση αισθήσεων, σχεδιασμό ενεργειών και τη δυνατότητα χειρισμού και κίνησης. Έτσι, μπορεί να ικανοποιηθεί πλήθος αναγκών στην βιομηχανία, στα εργαστήρια, ακόμα και στο σπίτι.

5.5 Ανάγκη για υποδομή

Τα ευφυή πληροφοριακά συστήματα μπορούν να προσφέρουν νέες δυνατότητες, ευκολίες και υπηρεσίες και να συμβάλλουν με πολλαπλούς τρόπους σε όποιον τομέα της ανθρώπινης δραστηριότητας εφαρμοστούν. Ωστόσο, για να πραγματοποιηθούν και να αποδώσουν τα όσα αναφέρθηκαν στην προηγούμενη ενότητα απαιτείται να ξεπεραστούν διάφορα τεχνικά προβλήματα τα οποία έχουν ομαδοποιηθεί και σύμφωνα με μία άλλη έκθεση της Αμερικανικής Εταιρείας Τεχνητής Νοημοσύνης [2] επικεντρώνονται

- στην ευκολία χρήσης και πιο συγκεκριμένα στους τρόπους αλληλεπίδρασης ανθρώπου μηχανής,
- στις ευέλικτες και προσαρμόσιμες υπηρεσίες υποδομής, που αφορούν κυρίως νέους τρόπους αλληλεπίδρασης μεταξύ των προγραμμάτων,
- στα εργαλεία ανάπτυξης και υποστήριξης των νέων τεχνολογιών.

Και σε αυτή την περίπτωση, οι τεχνικές της Τεχνητής Νοημοσύνης και κατ' επέκταση τα ήδη υπάρχοντα ευφυή πληροφοριακά συστήματα μπορούν να παίξουν σημαντικό ρόλο για την πραγματοποίηση των μακρόπνοων στόχων και των αναμενόμενων αποτελεσμάτων.

5.5.1 Ευκολία Χρήσης

Όσο οι εφαρμογές γίνονται πιο σύνθετες, όσο ο όγκος των διαθέσιμων πληροφοριών και υπηρεσιών αυξάνεται και όσο όλο περισσότεροι νέοι και μη έμπειροι χρήστες πρέπει να τις χρησιμοποιήσουν τόσο γίνεται πιο απαιτητική η ανάγκη για νέους ευκολότερους και φυσικότερους τρόπους επικοινωνίας με τους υπολογιστές. Το μέσο που είναι υπεύθυνο για την αλληλεπίδραση του ανθρώπου με τον υπολογιστή είναι το σύστημα διεπαφής χρήστη (user interface). Η σημερινή κυρίαρχη μορφή του είναι η γραφική παραθυρική (window–based graphical interface), η οποία έχει απλοποιήσει σημαντικά τη χρήση του υπολογιστή προσφέροντας εύκολους, πολλαπλούς και πολυμεσικούς τρόπους επικοινωνίας.

Ωστόσο, οι νέες ανάγκες που έχουν προκύψει καθιστούν πολλές φορές ανεπαρκή αυτόν τον τρόπο αλληλεπίδρασης. Για παράδειγμα, σήμερα ο μέσος χρήστης που έχει πρόσβαση στο Διαδίκτυο, έχει στη διάθεσή του ένα τεράστιο όγκο πηγών πληροφοριών και ηλεκτρονικών υπηρεσιών όπως ψυχαγωγικά, επιστημονικά ή δημοσιογραφικά άρθρα, φωτογραφίες, ήχους ή videos σχετικά με την εργασία ή τα χόμπι του, ταινίες, παιγνίδια, εφαρμογές πολυμέσων, υπηρεσίες επικοινωνίας με άλλους χρήστες, υπηρεσίες εμπορικών συναλλαγών ή ηλεκτρονικού εμπορίου κ.ά. Κατά συνέπεια, αυτό που χρειάζεται είναι νέοι τρόποι αλληλεπίδρασης, που θα καθιστούν εφικτή και αποτελεσματική τη χρήση των νέων προσφερόμενων αγαθών.

Τις νέες αυξημένες ανάγκες μπορούν να καλύψουν τα ευφυή συστήματα διεπαφής χρήστη (intelligent user interfaces), όπου η αλληλεπίδραση με το χρήστη θα είναι σε ένα ανώτερο και πιο φυσικό επίπεδο. Τέτοια συστήματα θα μπορούν να λειτουργήσουν ως προσωπικοί βοηθοί που θα προσαρμόζονται στις ανάγκες και στις ιδιαιτερότητες του κάθε χρήστη και θα μαθαίνουν τις συνήθειές του, παρατηρώντας τις ενέργειές του. Θα κατανοούν τις ερωτήσεις, τις οδηγίες, τις εντολές ή τις αιτήσεις του χρήστη, οι οποίες θα μπορούν να τεθούν σε μία ποικιλία μορφών, όπως κείμενο, λόγος, χειρονομίες. Σε περίπτωση που η διατύπωση είναι λανθασμένη ή ασαφής αντί να παρουσιάζουν τα γνωστά παθητικά μηνύματα λάθους, θα συνεργάζονται μαζί του προκειμένου να διατυπωθεί αυτό που επιθυμεί ο χρήστης με κατανοητό και αποδεκτό τρόπο. Τέλος, οι προσωπικοί βοηθοί θα καθορίζουν μόνοι τους τις ενέργειες που πρέπει να εκτελεστούν για την επίτευξη ενός στόχου χωρίς τη συνεχή παρέμβαση του χρήστη. Το μόνο που θα χρειάζεται να κάνει ο χρήστης είναι να περιγράφει αυτά που θέλει να επιτύχει.

5.5.2 Ευέλικτες και Προσαρμόσιμες Υπηρεσίες Υποδομής

Η ανάγκη νέων τρόπων αλληλεπίδρασης δεν περιορίζεται μόνο μεταξύ ανθρώπου και υπολογιστή αλλά επεκτείνεται και μεταξύ των προγραμμάτων. Κάτι τέτοιο είναι φυσικό και αναμενόμενο διότι, πρώτον, για να παρασχεθούν νέες ευκολίες απαιτούνται νέα μέσα και νέες τεχνολογίες, και δεύτερον, εφόσον ο άνθρωπος απαλλάσσεται από το βάρος κάποιων ενεργειών θα πρέπει κάποιος άλλος να το επωμιστεί. Η αλληλεπίδραση μεταξύ των προγραμμάτων αφορά τον τρόπο που αυτά θα επικοινωνούν μεταξύ τους για να ανταλλάξουν πληροφορίες, να ζητήσουν υπηρεσίες ή να συνεργαστούν για να επιτύχουν κάποιο κοινό στόχο. Ωστόσο, για να είναι αποτελεσματική μία τέτοια επικοινωνία απαιτούνται υπηρεσίες υποδομής για

- τη διαχείριση των δεδομένων και της γνώσης έτσι ώστε να επιτρέπεται η γρήγορη αναζήτηση και ανεύρεση των σχετικών πληροφοριών και πηγών μέσα από ένα τεράστιο όγκο ετερογενών και κατανεμημένων δεδομένων,
- τη μετάφραση / μετατροπή και ενοποίηση πληροφοριών διαφόρων μορφών και μέσων σύμφωνα με τη σημασιολογία τους.

Ευφυή συστήματα αναζήτησης πληροφοριών ειδικευμένα σε συγκεκριμένα αντικείμενα και εφοδιασμένα με νέους, ισχυρούς και αποτελεσματικούς αλγόριθμους αναζήτησης αλλά και ευρετικές μεθόδους[3] μπορούν να λειτουργήσουν ως διαμεσολαβητές μεταξύ του χρήστη και των πηγών πληροφορίας. Επιπλέον, μπορούν να επικοινωνούν με άλλα αντίστοιχα συστήματα για την ανεύρεση θεμάτων σχετικών με το αντικείμενο αναζήτησης ή νεοεκδιδόμενων πληροφοριών και υπηρεσιών.

Ωστόσο, η αποτελεσματικότητα τέτοιων συστημάτων απαιτεί νέες μορφές αναπαράστασης της γνώσης και υποστήριξη πολλαπλών ταξινομήσεων και σχημάτων κατηγοριοποίησης και μεθόδους αντίληψης του περιεχομένου της ανταλλασσόμενης πληροφορίας.

[3] Με τον όρο «ευρετικές μέθοδοι» ή «ευρετικά» εννοούμε εμπειρικούς κανόνες που αποτελούν τεχνικές οι οποίες βελτιώνουν την αποτελεσματικότητα της αναζήτησης, δίνοντας αποδεκτές λύσεις αλλά όχι αναγκαστικά τις βέλτιστες. Οι ευρετικές μέθοδοι εφαρμόζονται κυρίως όταν υπάρχει πληροφορία για το συγκεκριμένο πρόβλημα η οποία βοηθάει στην αναζήτηση.

5.5.3 Εργαλεία Ανάπτυξης και Υποστήριξης

Η ανάπτυξη και η υποστήριξη των συστημάτων που θα επιτρέπουν και θα χρησιμοποιούν τις νέες μορφές αλληλεπίδρασης, απαιτούν τα αντίστοιχα εργαλεία και τις τεχνικές, εκείνες οι οποίες θα καθιστούν εφικτή την πραγματοποίηση αυτών των στόχων. Τέτοια εργαλεία είναι απαραίτητα στην ανάπτυξη προδιαγραφών για την περιγραφή του τρόπου σκέψης και συμπεριφοράς του συστήματος, στην υλοποίηση αρχιτεκτονικών συστημάτων που δρουν αυτόνομα και θα περιέχουν μηχανισμούς παρατήρησης, μάθησης και προσαρμογής στο περιβάλλον λειτουργίας, στην ανάπτυξη δομών διαχείρισης γνώσης και σχεδιασμού και επιλογής ενεργειών, καθώς και στην ανάπτυξη δομών επικοινωνίας και συνεργασίας.

Επιπλέον, η υποστήριξη τέτοιων συστημάτων απαιτεί τη χρήση ανοιχτών αρχιτεκτονικών δυναμικής δραστηριοποίησης πολλαπλών αυτόνομων συστημάτων, δηλαδή αρχιτεκτονικών που επιτρέπουν την προσθήκη ή την απομάκρυνση στοιχείων τους χωρίς να μεταβάλλεται η καλή λειτουργία του όλου συστήματος. Επίσης, απαιτεί την ανάπτυξη πρωτοκόλλων και γλωσσών επικοινωνίας που θα επιτρέπουν την ανταλλαγή πληροφοριών και γενικότερα τη διαπραγμάτευση και συνεννόηση μεταξύ των συστημάτων και τέλος τη δυνατότητα επαναχρησιμοποίησης των κατάλληλων εργαλείων για την επίτευξη ενός στόχου.

5.6 Τεχνολογίες Αιχμής και Τομείς Έρευνας

Στις προηγούμενες ενότητες εξετάστηκαν περιοχές εφαρμογών και θέματα υποδομής όπου η Τεχνητή Νοημοσύνη μπορεί να έχει καθοριστική συμβολή προσφέροντας σημαντικές λύσεις. Ωστόσο, αν και οι τεχνολογίες της Τεχνητής Νοημοσύνης παρέχουν μία στέρεα βάση αποτελεσματικών τεχνικών και εργαλείων, η μεμονωμένη χρήση και εφαρμογή τους δεν μπορεί να καλύψει τις αυξημένες ανάγκες ούτε των ευφυών υπολογιστικών συστημάτων, ούτε της απαιτούμενης υποδομής. Αυτό που χρειάζεται είναι ο συνδυασμός αυτών μέσα από μία καινοτόμο αντίληψη ανάπτυξης λογισμικού και κατ' επέκταση ευφυών συστημάτων. Στη συνέχεια θα παρουσιαστούν οι τομείς έρευνας στο πεδίο της Τεχνητής Νοημοσύνης, οι οποίοι συνδυαζόμενοι μπορούν να αποδώσουν τα αναμενόμενα αποτελέσματα.

Οι ευφυείς διαμεσολαβητές (intelligent agents) και ο προσανατολισμένος στους διαμεσολαβητές προγραμματισμός (agent oriented programming) είναι ίσως η πιο επαναστατική μέθοδος αντίληψης, σχεδίασης και ανάπτυξης λογισμικού των τελευταίων χρόνων. Γενικά, ένας ευφυής διαμεσολαβητής είναι ένα αυτόνομο πρόγραμμα που είναι συνεχώς ενεργό, παρατηρεί το περιβάλλον όπου ενεργεί και αντιδρά στα γεγονότα που το αφορούν χωρίς την παρέμβαση του χρήστη. Επιπλέον, ενεργεί σύμφωνα με κάποιους στόχους που έχει, οι οποίοι μπορεί να είναι μέρος ενός απώτερου στόχου. Προκειμένου να επιτευχθούν αυτοί οι στόχοι μπορεί να επικοινωνήσει και να συνεργαστεί με άλλους διαμεσολαβητές[4].

Τα περισσότερα από τα συστήματα, που περιγράφηκαν στις δύο προηγούμενες ενότητες, είναι δυνατό να υλοποιηθούν ως διαμεσολαβητές ή να περιέχουν έναν ή περισσότερους διαμεσολαβητές ως συστατικά τους. Για παράδειγμα, ένα ευφυές σύστημα προσομοίωσης μπορεί να αποτελείται από διάφορους διαμεσολαβητές, οι οποίοι παίζουν το ρόλο των διαφόρων οντοτήτων που συμμετέχουν στην προσομοίωση.

[4] Αυτός ο ορισμός είναι κάπως γενικός και περιγραφικός και χρησιμοποιείται εδώ για να καλύψει τις ανάγκες της συγκεκριμένης ενότητας.

Ένας προσωπικός βοηθός (personal assistant) μπορεί να είναι ένας διαμεσολαβητής που βρίσκεται συνέχεια στο πλάι του χρήστη και τον βοηθά όπου χρειάζεται ή αναλαμβάνει να εκτελέσει εργασίες εκ μέρους του. Διαμεσολαβητές μπορούν να είναι τα ευφυή συστήματα αναζήτησης πληροφορίας (intelligent information brokers) τα οποία, ενώ βρίσκονται απομακρυσμένα σε διάφορους τόπους στο Διαδίκτυο, συνεργάζονται μεταξύ τους, για να ανευρεθεί η αναζητούμενη πληροφορία.

Οι διαμεσολαβητές μπορεί να έχουν από πολλές και σύνθετες ικανότητες έως απλές και περιορισμένες δυνατότητες και μέσα από τη συνεργασία με άλλους διαμεσολαβητές να πραγματοποιούν πολύπλοκα έργα. Το γεγονός αυτό επιτρέπει την ενσωμάτωση, σύνθεση και ολοκλήρωση διαφόρων τεχνολογιών της Τεχνητής Νοημοσύνης για την υλοποίηση ενός διαμεσολαβητή ή ενός συστήματος από πολλούς διαμεσολαβητές (multi agent system). Έτσι ανάλογα με την εφαρμογή, μπορούν να χρησιμοποιηθούν τεχνικές αναπαράστασης γνώσης, μέθοδοι ανάκτησης πληροφορίας και ευρετικά, σχεδιασμός και επιλογή ενεργειών, τεχνικές εκμάθησης και προσαρμογής, επεξεργασία εικόνων και φυσικής γλώσσας κ.ά.

Σε ένα σύστημα πολλαπλών διαμεσολαβητών καθοριστικό ρόλο παίζει η οργανωτική μορφή της κοινωνίας των διαμεσολαβητών (agent society), καθώς και η γνώση που έχει ο καθένας για τις ικανότητες των υπολοίπων. Η επικοινωνία, ο συντονισμός των ενεργειών και η συνεργασία των διαφόρων διαμεσολαβητών εξαρτάται σε μεγάλο βαθμό από αυτούς τους δύο παράγοντες. Για παράδειγμα, σε ένα εκπαιδευτικό σύστημα πολλαπλών διαμεσολαβητών με αντικείμενο τη διδασκαλία ενός ιατρικού θέματος, ο εκπαιδευόμενος αλληλεπιδρά με τον κατάλληλο διαμεσολαβητή, ο οποίος γνωρίζει πώς διδάσκεται το αντικείμενο και έχει παιδαγωγικές αρχές και εκπαιδευτικούς στόχους. Τα ειδικά τεχνικά ζητήματα του αντικειμένου επιλύονται και παρέχονται σε αυτόν από άλλους διαμεσολαβητές που είναι αρμόδιοι και ειδικοί στα συγκεκριμένα θέματα, ενώ επιπλέον, αυτοί μπορούν να επικοινωνήσουν και να συνεργαστούν όταν υπάρχει επικαλυπτόμενη ή διαπλεκόμενη γνώση. Σε ένα αντίστοιχο σύστημα αναζήτησης πληροφοριών, ο διαμεσολαβητής που επικοινωνεί με το χρήστη μπορεί να μη γνωρίζει πού βρίσκεται η ζητούμενη πληροφορία, μπορεί ωστόσο να γνωρίζει άλλους διαμεσολαβητές, που έχουν τη δυνατότητα να τον πληροφορήσουν σχετικά ή να αναλάβουν οι ίδιοι την αναζήτηση.

Ένας επιπλέον παράγοντας, που σχετίζεται με την αλληλεπίδραση των συστημάτων πολλαπλών διαμεσολαβητών και γενικότερα των ευφυών συστημάτων, είναι η κατανόηση του νοήματος της ανταλλασσόμενης πληροφορίας. Για να επιτευχθεί αυτό, το σύστημα πρέπει να έχει γνώση για τον τομέα όπου εφαρμόζεται. Η γνώση αυτή αναφέρεται ως οντολογία (ontology) του τομέα (π.χ. Οικονομία, Αρχιτεκτονική, Πληροφορική, κτλ.) και σκοπό έχει να περιγράψει το νόημα και τις σχέσεις των οντοτήτων που ανήκουν σε αυτόν τον τομέα. Οι οντολογίες αποτελούν πολύ χρήσιμα εργαλεία διότι αφενός καλύπτουν πλήθος φαινομένων και καταστάσεων και αφετέρου είναι επαναχρησιμοποιήσιμες από διάφορα συστήματα, όποτε αυτό είναι απαραίτητο. Έτσι, σχετικά με το παραπάνω παράδειγμα, το εκπαιδευτικό σύστημα χρησιμοποιώντας τις ίδιες εκπαιδευτικές στρατηγικές και τις αντίστοιχες οντολογίες, θα μπορούσε να διδάξει δερματολογία, καρδιολογία, ορθοπεδική, κτλ. Επιπλέον, ένα ευφυές σύστημα διεπαφής χρήστη ή ένα ευφυές robot θα μπορούσε να κατανοήσει και να επικοινωνήσει σε φυσικό επίπεδο με τον άνθρωπο.

Σχεδόν όλα τα ευφυή πληροφοριακά συστήματα χρησιμοποιούν και διαχειρίζονται κάποιο είδος γνώσης (συμβολικό ή αριθμητικό). Για αυτόν το λόγο οι τεχνικές αναπαράστασης γνώσης και ο τρόπος σκέψης (λογισμός) βασισμένος στη γνώση αποτελούν βασικούς τομείς έρευνας και εφαρμογής για την Τεχνητή Νοημοσύνη. Ο στόχος είναι η ανάπτυξη μεθόδων και μορφών που αναπαριστούν ένα εύρος από

πληροφορίες, από απλές και καθημερινές έως ειδικές και τεχνικές, και επιτρέπουν την κατανόηση και ταχεία παραγωγή συμπερασμάτων. Όταν η αναπαράσταση είναι σε χαμηλό επίπεδο, δηλαδή πιο κοντά στο επίπεδο της μηχανής (π.χ. βάσεις δεδομένων), τότε η ταχύτητα ανάκλησης είναι μεγάλη και επιτρέπεται η γρήγορη παραγωγή συμπερασμάτων. Ωστόσο, μόνο περιορισμένα και συγκεκριμένα κομμάτια γνώσης μπορούν να κωδικοποιηθούν με αυτή τη μορφή, ενώ παρουσιάζεται έλλειψη εκφραστικότητας και τα συμπεράσματα είναι περιορισμένα. Αντίθετα, όταν η αναπαράσταση είναι σε υψηλό επίπεδο, δηλαδή πιο κοντά στον άνθρωπο (π.χ. κείμενο φυσικής γλώσσας), η εκφραστικότητα μεγιστοποιείται, υπάρχει η δυνατότητα κωδικοποίησης μεγάλης ποσότητας κοινής γνώσης (commonsense knowledge) και τα συμπεράσματα που παράγονται είναι πιο λεπτομερή και αναλυτικά. Ωστόσο, η ανάκληση είναι κατά κανόνα πιο αργή, η διαχείρισή της πιο δύσκολη, ενώ τα συμπεράσματα πιθανόν να είναι ασαφή ή ελλιπή.

Ο συνδυασμός των δύο άκρων στην αναπαράσταση γνώσης είναι απαραίτητος, καθώς οι ευφυείς διαμεσολαβητές και γενικότερα τα ευφυή πληροφοριακά συστήματα αλληλεπιδρούν τόσο με ανθρώπους όσο και μεταξύ τους ή με άλλα μηχανικά μέρη. Σε πραγματικά συστήματα, η γνώση είναι συνήθως περιορισμένη ή μη πλήρης, ενώ το περιβάλλον και οι συνθήκες λειτουργίας πολλές φορές απαιτούν απόκριση σε περιορισμένο χρόνο και οι πληροφορίες που παρέχονται μπορεί να είναι ασαφείς, ελλιπείς ή λανθασμένες. Στην κάλυψη αυτών των αναγκών επικεντρώνεται η ανάπτυξη λογισμών με απώτερο στόχο την αντιμετώπιση του προβλήματος της αναπαράστασης, της κατανόησης και του ελέγχου της συμπεριφοράς των ευφυών συστημάτων. Ενδεικτικά αναφέρονται ο μη μονοτονικός λογισμός (non monotonic reasoning) όπου μπορούν να παραχθούν συμπεράσματα από ελλιπείς πληροφορίες και στη συνέχεια να αναιρεθούν ή να συμπληρωθούν όταν νέες πληροφορίες γίνουν διαθέσιμες, ο χρονικός λογισμός (temporal reasoning) όπου η παραγωγή συμπερασμάτων γίνεται με παραμέτρους χρόνου, ο λογισμός περιπτώσεων (case based reasoning) όπου η επίλυση προβλημάτων βασίζεται σε μεθόδους επίλυσης παλαιότερων και παρόμοιων προβλημάτων, ή τέλος τεχνικές που χρησιμοποιούν αρχές της Ασαφούς Λογικής (Fuzzy Logic) και της Θεωρίας Πιθανοτήτων (π.χ. Bayesian Networks). Όμως, προκειμένου να είναι δυνατή η συνεργασία και ο συντονισμός των διαφόρων διαμεσολαβητών σε ένα κοινό περιβάλλον λειτουργίας, πρέπει οι λογισμοί που χρησιμοποιούνται για τη διαχείριση της γνώσης και της συμπεριφοράς να επεκταθούν και να συμπληρωθούν με δυνατότητες επικοινωνίας και τρόπους σκέψης σχετικά με την νοητική κατάσταση τόσο του ίδιου (introspective reasoning) όσο και των υπολοίπων διαμεσολαβητών (social reasoning).

Ο μηχανισμός σκέψης (λογισμός) ενός διαμεσολαβητή έχει άμεση σχέση με τον τρόπο που αυτός σχεδιάζει τις ενέργειές του προκείμενου να πετύχει κάποιους στόχους. Η δυνατότητα σχεδιασμού και καθορισμού ενεργειών, η εκτέλεσή τους και η παρακολούθηση των αποτελεσμάτων τους είναι απαραίτητη σε κάθε ευφυές σύστημα ή διαμεσολαβητή και απαιτεί εξελιγμένες ικανότητες αναπαράστασης και συλλογισμού σχετικά με τις ενέργειες, το χρόνο, την αντίληψη και τη νοητική κατάσταση άλλων διαμεσολαβητών που πιθανόν συμμετέχουν στην εκπλήρωση κάποιων κοινών στόχων. Για παράδειγμα, ένα ευφυές σύστημα αναζήτησης πληροφοριών, χρειάζεται να σχεδιάσει τον καλύτερο τρόπο για να αποκτήσει μία πληροφορία εξισορροπώντας ταυτόχρονα το χρόνο που απαιτείται για την ανεύρεση, το κόστος προσπέλασης των διαφόρων βάσεων δεδομένων και την επιτυχία της αναζήτησης. Αντίστοιχα, σε ένα σύστημα προσομοίωσης, όπου συμμετέχουν διάφοροι διαμεσολαβητές και αλληλεπιδρούν τόσο μεταξύ τους όσο και με ανθρώπους, πρέπει να αντιμετωπιστούν σύνθετες καταστάσεις, που προκύπτουν από το συνδυασμό των ενεργειών αυτών όσων συμμετέχουν στην προσομοίωση,

δημιουργώντας ένα δυναμικό περιβάλλον. Παρόμοιες πολύπλοκες καταστάσεις προκύπτουν στο φυσικό περιβάλλον, οι οποίες θα κληθούν να λειτουργήσουν ευφυή robots όπου ο τυχαίος και απρόβλεπτος παράγοντας παίζει πολύ σημαντικό ρόλο.

Το μεγαλύτερο και δυσκολότερο θέμα που προκύπτει είναι ότι το σύστημα πρέπει να επιλέξει το κατάλληλο για την περίπτωση πλάνο προκειμένου να πετύχει τα καλύτερα αποτελέσματα σε συγκεκριμένο χρόνο με δεδομένες τις τρέχουσες συνθήκες του περιβάλλοντος. Γι' αυτό μεγάλη ερευνητική προσπάθεια επικεντρώνεται σε τρόπους παραγωγής και επιλογής πλάνων ανάλογα με το στόχο που πρέπει να επιτευχθεί. Αυτό απαιτεί συνδυασμό τεχνικών της Τεχνητής Νοημοσύνης, της Θεωρίας Αποφάσεων και της Θεωρίας Παιγνίων.

Ένα ευφυές σύστημα μπορεί να έχει αυξημένες επιδόσεις και μεγαλύτερη ευελιξία, αν έχει δυνατότητες μάθησης και προσαρμογής σε νέες καταστάσεις. Η μηχανική μάθηση (machine learning) επικεντρώνεται κυρίως στην ανάπτυξη λογισμικού που βελτιώνεται αυτόματα μέσω εμπειρίας και στην εξαγωγή κανόνων από ένα σύνολο συγκεκριμένων δεδομένων [2]. Οι μέθοδοι μηχανικής μάθησης μπορούν να εφαρμοστούν στις περισσότερες κατηγορίες συστημάτων, που έχουν αναφερθεί παραπάνω, όπως σε προσωπικούς βοηθούς, σε συστήματα αναζήτησης πληροφοριών, σε συστήματα εκπαίδευσης και κατάρτισης, σε συστήματα υποστήριξης, σε ευφυή robots κτλ.

Ήδη αρκετές μέθοδοι, όπως επαγωγή από δέντρα αποφάσεων, νευρωνικά δίκτυα, γενετικοί αλγόριθμοι κ.ά., έχουν εφαρμοστεί σε ένα ευρύ φάσμα πραγματικών προβλημάτων με αξιοσημείωτα αποτελέσματα. Ωστόσο, είναι απαραίτητες νέες μέθοδοι που θα μπορούν να εκμεταλλευτούν τις δυνατότητες που παρέχει το Διαδίκτυο, συνδυάζοντας τις πολλαπλές πληροφορίες των διαφόρων βάσεων δεδομένων και τις πληροφορίες που παρέχονται από τους χρήστες. Επιπλέον, ο συνδυασμός και η ενοποίηση των μεθόδων μηχανικής μάθησης, των μεθόδων παραγωγής και επιλογής πλάνων, καθώς και των μεθόδων επεξεργασίας φυσικής γλώσσας και εικόνας που περιγράφονται στη συνέχεια, θα αποτελέσει έναν ισχυρό μηχανισμό για την συμπεριφορά των ευφυών συστημάτων και την αλληλεπίδρασή τους με τους ανθρώπους.

Σε πολλές από τις περιπτώσεις που έχουν περιγραφεί παραπάνω, τα ευφυή πληροφοριακά συστήματα και οι ευφυείς διαμεσολαβητές είναι απαραίτητο να έχουν αυξημένες και ιδιαίτερες ικανότητες επικοινωνίας με τους χρήστες και αντίληψης του περιβάλλοντος. Για παράδειγμα, τα ευφυή robots για να λειτουργήσουν ρεαλιστικά χρειάζονται όραση, χρήση και κατανόηση φυσικής γλώσσας, ίσως και αφή. Αντίστοιχα οι ευφυείς προσωπικοί βοηθοί θα πρέπει να μπορούν να χρησιμοποιούν και να καταλαβαίνουν τη φυσική γλώσσα ή ακόμα να αντιλαμβάνονται χειρονομίες ή εκφράσεις του προσώπου. Τα συστήματα προσομοίωσης, τα συστήματα κατάρτισης και εκπαίδευσης, τα συστήματα βοηθείας για τη διαχείριση έργων θα πρέπει να έχουν δυνατότητες επεξεργασίας και σύνθεσης λόγου και εικόνας. Ο συνδυασμός της Τεχνητής Νοημοσύνης με αρχές σχετικές με τους τρόπους αντίληψης και επικοινωνίας που αφορούν τον άνθρωπο έχουν δώσει αρκετά και ικανοποιητικά αποτελέσματα και έχουν δημιουργήσει ένα υπόβαθρο τόσο στην επεξεργασία λόγου και γλώσσας όσο και στην επεξεργασία εικόνας. Ωστόσο, τα περισσότερα από τα ήδη υπάρχοντα συστήματα δίνουν τα επιθυμητά αποτελέσματα συνήθως κάτω από συγκεκριμένες και πολλές φορές ιδανικές συνθήκες λειτουργίας. Προκειμένου να υπάρξουν συστήματα που να μπορούν να αντιλαμβάνονται και να επικοινωνούν με ρεαλιστικό τρόπο χρειάζεται περισσότερη έρευνα, βελτίωση των μεθόδων που έχουν προταθεί και ενίσχυσή τους με τη χρήση μεθόδων μηχανικής μάθησης.

6 Ευφυείς Διαμεσολαβητές

6.1 Τι είναι ένας Ευφυής Διαμεσολαβητής

Η συχνή χρήση του όρου διαμεσολαβητής (agent) σε επιστήμες διαφορετικές μεταξύ τους και με διαφορετικούς τρόπους έχει ως συνέπεια να μην υπάρχει ξεκάθαρη εικόνα για το τι σημαίνει αυτός ο όρος, αν δε δοθεί το ευρύτερο πλαίσιο μέσα στο οποίο χρησιμοποιείται. Για να χρησιμοποιηθεί στην επιστήμη των υπολογιστών θα πρέπει να υπάρχει μία κοινή αναφορά για το τι είναι και τι κάνει ένας διαμεσολαβητής, έτσι ώστε να μπορεί να γίνει κατανοητή η χρήση του. Η γενική εξήγηση του ρου, όπως αυτή μπορεί να βρεθεί σε ένα λεξικό όρων [31] είναι ότι ένας διαμεσολαβητής είναι:

- ένα πρόσωπο ή πράγμα που ενεργεί ή καταβάλλει προσπάθεια,
- κάποια φυσική δύναμη, που δρα κατά περίπτωση προκαλώντας κάποια αποτελέσματα,
- ένα πρόσωπο (ή εταιρεία) εξουσιοδοτημένο να διεκπεραιώσει μια δουλειά για κάποιον άλλον,
- ένα πρόσωπο που ενεργεί εκ μέρους κάποιου άλλου για να πετύχει μια νόμιμη σχέση μεταξύ αυτού του άλλου και κάποιου τρίτου.

Στα παρακάτω θα δοθούν οι διάφορες απόψεις στο πεδίο της Τεχνητής Νοημοσύνης. Πολλοί ερευνητές χρησιμοποιούν κάποια από τις παραπάνω γενικές και απλές ερμηνείες. Για παράδειγμα ο Marvin Minsky αναφέρει ότι στο βιβλίο του «The society of Mind» [32] χρησιμοποίησε τη λέξη διαμεσολαβητής για να αναφερθεί σε μία μηχανή που επιτυγχάνει κάτι, χωρίς να χρειάζεται να γνωρίζει κανείς τον τρόπο που δουλεύει [33], ο N. Shardlow λεει ότι οι διαμεσολαβητές κάνουν διάφορα πράγματα / ενεργούν [40], ενώ η P. Maes σε μία συνέντευξή της αναφέρει ότι ένας διαμεσολαβητής είναι το λογισμικό που ενεργεί για λογαριασμό των ανθρώπων [29].

Ωστόσο, τέτοιοι γενικοί και αόριστοι ορισμοί δεν είναι και πολύ κατατοπιστικοί και εύλογα πολλοί επιστήμονες διερωτώνται αν αυτό είναι κάτι το διαφορετικό από το μέχρι σήμερα συμβατικό και «χαζό» λογισμικό [38]. Σε μία προσπάθεια να δοθεί συγκεκριμένος ορισμός για το τι είναι ένας διαμεσολαβητής, σύμφωνα με τη σειρά προτύπων που έχει δημοσιευτεί από τη FIPA (Foundation for Intelligent Physical Agents) [14], ένας διαμεσολαβητής ορίζεται ως εξής:

«Ένας διαμεσολαβητής είναι η θεμελιώδης μονάδα δράσης (actor) σε μία περιοχή. Συνδυάζει μία ή περισσότερες ικανότητες υπηρεσιών σε ένα ενοποιημένο και ενιαίο μοντέλο εκτέλεσης, το οποίο μπορεί να περιλαμβάνει προσπέλαση σε εξωτερικό λογισμικό, χρήστες (ανθρώπους) και μηχανισμούς επικοινωνίας» [14].

Σε μία άλλη προσπάθεια και προκειμένου να δοθεί μία πιο σαφής εικόνα για το τι είναι ένας διαμεσολαβητής, γίνεται αναφορά στις ιδιότητες που πρέπει να έχει ένα τέτοιο σύστημα. Ένας ορισμός που ουσιαστικά καλύπτει τις βασικές και γενικότερα αποδεκτές ιδιότητες, τις οποίες πρέπει να έχει ένας διαμεσολαβητής, είναι αυτός που χρησιμοποιούν οι Wooldridge και Jennings [46]. Ως διαμεσολαβητή αναφέρουν ένα σύστημα (hardware ή software) που παρουσιάζει ιδιότητες όπως αυτονομία (autonomy), κοινωνικότητα (social ability), αποκρισιμότητα (reactivity), και ανάληψη πρωτοβουλιών (pro–activeness). Γενικά, η πιο αποδεκτή και περισσότερο χρησιμοποιούμενη ιδιότητα είναι η αυτονομία, με την έννοια ότι ένας διαμεσολαβητής δεν χρειάζεται τη συνεχή καθοδήγηση και διαμεσολάβηση του ανθρώπου, ενώ ταυτόχρονα έχει κάποιον έλεγχο των ενεργειών του. Η κοινωνικότητα εκφράζει την ικανότητα ενός διαμεσολαβητή να επικοινωνεί με άλλους

διαμεσολαβητές προκειμένου να ζητήσει κάποια πληροφορία ή κάποια ενέργεια ή γενικότερα να επιτύχουν μαζί ένα κοινό στόχο. Η αποκρισιμότητα δηλώνει ότι ο διαμεσολαβητής παρατηρεί το περιβάλλον του και αντιδρά ανάλογα στα γεγονότα που τον αφορούν, ενώ η ανάληψη πρωτοβουλιών εκφράζει επιπλέον τη συμπεριφορά του διαμεσολαβητή ανάλογα με τους στόχους του και τη δραστηριοποίησή του, όποτε αυτός κρίνει απαραίτητο.

Άλλες ιδιότητες που αφορούν κυρίως τη συμπεριφορά των διαμεσολαβητών και έχουν χρησιμοποιηθεί είναι η εξατομίκευση (personalization) και η προσαρμογή (adaptation) [29], οι οποίες αναφέρονται κυρίως στους διαμεσολαβητές ως προσωπικούς βοηθούς που προσαρμόζονται ανάλογα με τις ανάγκες και ιδιαιτερότητες του χρήστη, η ικανότητα μετακίνησης (mobility) δηλαδή η δυνατότητα να μεταφέρεται ο διαμεσολαβητής σε απομακρυσμένους σταθμούς εργασίας προκειμένου να βρει κάποια διαθέσιμη υπηρεσία, η ειλικρίνεια (veracity) που δηλώνει ότι ο διαμεσολαβητής δε στέλνει εσκεμμένα ψευδείς πληροφορίες, η καλοπροαίρεση (benevolence) που αναφέρεται στο γεγονός ότι ο διαμεσολαβητής πάντα προσπαθεί να κάνει κάτι που θα του ζητηθεί και ο ορθολογισμός (rationality) δηλαδή το γεγονός ότι ο διαμεσολαβητής ενεργεί με τρόπο τέτοιο ώστε να πετύχει τους στόχους του και όχι με τρόπο τέτοιο ώστε να τους αποτρέψει.

Πολλοί επιστήμονες, ανάλογα με τις μεθόδους που χρησιμοποιούν για να περιγράψουν τον τρόπο συμπεριφοράς ενός διαμεσολαβητή, χρησιμοποιούν έννοιες που αποδίδονται κυρίως σε ανθρώπους, όπως γνώση, πεποιθήσεις, επιθυμίες, προθέσεις, δεσμεύσεις, ικανότητες, επιλογές ([4], [42]) ακόμα και συναισθήματα [2]. Τέτοιες έννοιες είναι περισσότερο κατανοητές στον άνθρωπο αφού είναι καθημερινής χρήσης και το νόημά τους είναι άμεσα αντιληπτό. Έτσι, προκειμένου να περιγραφούν πολύπλοκα συστήματα των οποίων η λειτουργία και συμπεριφορά δεν μπορούν να εκφραστούν εύκολα, γίνεται χρήση της λεγόμενης λαϊκής ψυχολογίας (folk psychology) που ο καθένας μπορεί να καταλάβει χωρίς δυσκολία. Η θεώρηση ενός διαμεσολαβητή ως συστήματος προθέσεων (intentional system) βασίζεται στην περιγραφή του φιλοσόφου Daniel Dennett, ο οποίος με τον όρο αυτό περιγράφει οντότητες «… των οποίων η συμπεριφορά μπορεί να προβλεφθεί αποδίδοντάς τους πεποιθήσεις, επιθυμίες και λογική οξυδέρκεια» [12].

Η παραπάνω θεώρηση είναι ευρέως διαδεδομένη και χρησιμοποιούμενη στις τυπικές προδιαγραφές των διαμεσολαβητών. Αμέσως, όμως, τίθεται το θέμα αν η απόδοση τέτοιων χαρακτηριστικών στους διαμεσολαβητές (και γενικότερα στις μηχανές) είναι θεμιτή και δικαιολογημένη και ακόμα αν είναι χρήσιμη. Οι περιπτώσεις, κατά τις οποίες αυτή η θεώρηση είναι θεμιτή και χρήσιμη έχουν συζητηθεί από τον J. McCarthy:

«Η απόδοση χαρακτηριστικών όπως πεποιθήσεων, ελεύθερης βούλησης, προθέσεων, συνείδησης, ικανοτήτων ή επιθυμιών σε μία μηχανή ή ένα πρόγραμμα είναι θεμιτή, όταν μια τέτοια απόδοση εκφράζει την ίδια πληροφορία σχετικά με τη μηχανή όπως θα εξέφραζε και για ένα άτομο. Είναι χρήσιμη όταν αυτή η απόδοση μας βοηθάει να κατανοήσουμε τη δομή της μηχανής, την προηγούμενη ή μελλοντική της συμπεριφορά, ή τον τρόπο για να την επισκευάσουμε ή να τη βελτιώσουμε» [30].

Αργότερα, διάφοροι επιστήμονες απέδειξαν ότι λίγο πολύ κάθε αντικείμενο μπορεί να περιγραφεί χρησιμοποιώντας αυτή τη θεώρηση. Χαρακτηριστικό είναι το παράδειγμα που χρησιμοποιεί ο Shoham στο [41] με το διακόπτη ηλεκτρικού ρεύματος.

Αναφέρει ότι είναι απόλυτα λογικό να θεωρήσει κανείς ένα διακόπτη ηλεκτρικού ρεύματος ως έναν διαμεσολαβητή με βούληση μετάδοσης ρεύματος ανάλογα αν πιστεύει ότι θέλουμε να πραγματοποιηθεί αυτή η μετάδοση ή όχι. Έτσι, το πάτημα

του διακόπτη είναι ο τρόπος επικοινωνίας μαζί του και ο τρόπος έκφρασης των επιθυμιών μας. Βέβαια, με τη χρήση τέτοιων εννοιών για την περιγραφή ενός διακόπτη ηλεκτρικού ρεύματος δεν υπάρχει ιδιαίτερο κέρδος, αφού ο διακόπτης είναι μια πολύ απλή συσκευή και η λειτουργία του αρκετά κατανοητή. Όμως, δεν ισχύει το ίδιο για πολύπλοκα συστήματα όπως για παράδειγμα τα robots, όπου η χρήση τέτοιων εννοιών κάνει ευκολότερη την κατανόηση της λειτουργία τους.

Όπως αναφέρεται παραπάνω η θεώρηση των διαμεσολαβητών ως συστημάτων προθέσεων επικεντρώνεται στην περιγραφή της συμπεριφοράς τους. Αυτό βοηθάει στην αποφυγή λογικών παραδόξων και διαισθητικών ασυμβατοτήτων. Ένα βασικό μειονέκτημα είναι ότι δεν υποδηλώνεται τίποτα σχετικά με την εσωτερική κατάσταση ή αναπαράσταση του περιγραφόμενου συστήματος, ούτε μπορεί να γίνει καθορισμός

του τρόπου υλοποίησης, καθώς μία συμπεριφορά μπορεί να υλοποιηθεί με διάφορους τρόπους. Ωστόσο, από σχεδιαστική άποψη, επιτρέπει την καλύτερη κατανόηση και συνεπώς την περιγραφή πολύπλοκων συστημάτων και αυτό το γεγονός την καθιστά ιδιαίτερα χρήσιμη.

Καταλήγοντας οι Wooldridge και Jennings [46] θεωρούν ότι τα κατάλληλα χαρακτηριστικά για την αναπαράσταση των διαμεσολαβητών κατηγοριοποιούνται σε αυτά που σχετίζονται με την πληροφορία που έχει ένας διαμεσολαβητής σχετικά με το περιβάλλον όπου βρίσκεται όπως πεποιθήσεις και γνώση (information attitudes), και σε αυτά που κατά κάποιο τρόπο καθοδηγούν της ενέργειες του διαμεσολαβητή (pro–attitudes), όπως επιθυμίες, προθέσεις, δεσμεύσεις, επιλογές κ.ά.

6.2 Αρχιτεκτονικές

Με τον όρο αρχιτεκτονική ενός διαμεσολαβητή (και κατ' επέκταση ενός συστήματος πολλαπλών διαμεσολαβητών) εννοούμε την μέθοδο κατασκευής του και τον τρόπο που το σύνολο των δομικών του στοιχείων (modules) σχετίζονται και επικοινωνούν μεταξύ τους και με το περιβάλλον. Στο χώρο της Τεχνητής Νοημοσύνης υπάρχουν δύο βασικές σχολές με αντιδιαμετρικές απόψεις σχετικά με το ποια είναι αυτά τα χαρακτηριστικά που κάνουν έναν διαμεσολαβητή ευφυή και αντίστοιχα δύο κατηγορίες αρχιτεκτονικών, οι αρχιτεκτονικές νοημοσύνης (deliberative architectures) και οι αρχιτεκτονικές αντίδρασης (reactive architectures). Ενδιάμεσα βρίσκονται οι υβριδικές αρχιτεκτονικές (hybrid architectures), που συνδυάζουν χαρακτηριστικά των άλλων δύο.

6.2.1 Αρχιτεκτονικές Νοημοσύνης

Η παραδοσιακή και επικρατούσα άποψη θεωρεί τους διαμεσολαβητές ως συστήματα συμβόλων και με αυτόν τον τρόπο κωδικοποιείται η νοητική τους κατάσταση. Η νοημοσύνη τους προκύπτει, μέσω των μηχανισμών διαχείρισης της νοητικής κατάστασης, που παρέχουν δυνατότητα λήψης αποφάσεων, παραγωγής συμπερασμάτων από τη γνώση τους, επιλογής των ενεργειών τους και ανανέωσης της γνώσης τους.

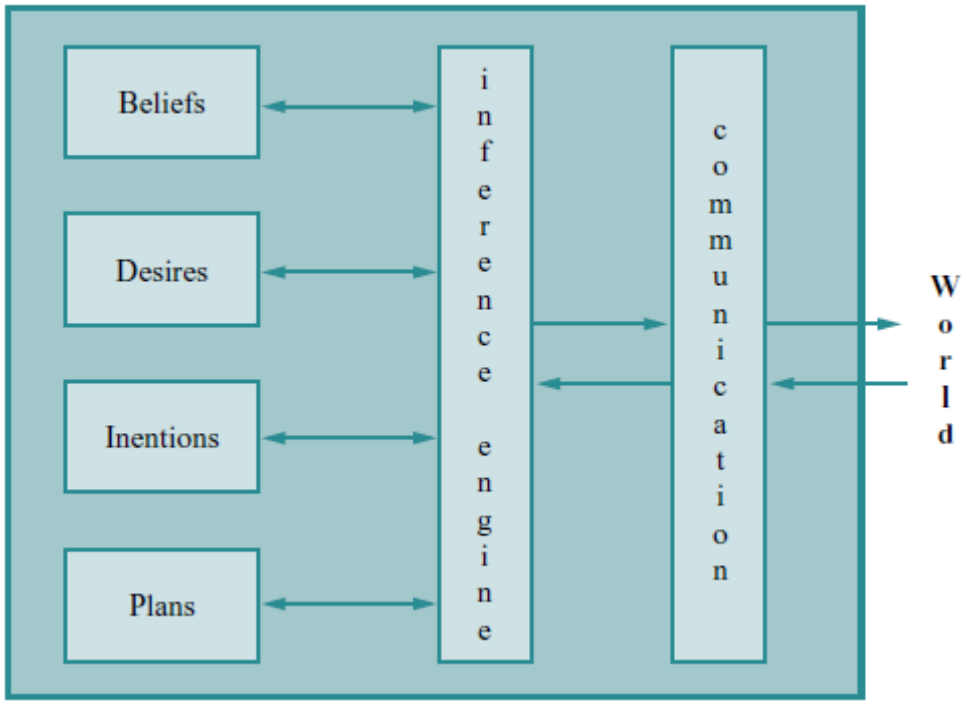
Μία τέτοια αρχιτεκτονική καλείται αρχιτεκτονική νοημοσύνης (deliberative architecture). Ένας διαμεσολαβητής κατασκευασμένος σύμφωνα με την αρχιτεκτονική νοημοσύνης περιέχει σε συμβολική αναπαράσταση το μοντέλο του κόσμου, όπου δραστηριοποιείται, μοντέλο το οποίο αντιστοιχεί στις πεποιθήσεις του, και ο τρόπος σκέψης και απόφασης βασίζεται σε κάποιο μηχανισμό παραγωγής

λογικών συμπερασμάτων. Η λειτουργία του ακολουθεί έναν κύκλο τριών βημάτων, σύμφωνα με τον οποίο παρατηρεί το περιβάλλον και για όσα γεγονότα τον αφορούν σκέπτεται (deliberates) και κατόπιν ενεργεί βασιζόμενος στα συμπεράσματα που έφθασε και τις αποφάσεις που πήρε. Η μεγάλη πλειοψηφία των συστημάτων διαμεσολαβητών που έχουν αναπτυχθεί βασίζονται σε αρχιτεκτονικές νοημοσύνης.

Η αρχιτεκτονική BDI που ανήκει στην παραπάνω κατηγορία και χρησιμοποιείται ευρέως τελευταία, όχι μόνο περιέχει τα βασικά συστατικά των συστημάτων προθέσεων αλλά ικανοποιεί και τις αρχές του πρακτικού λογισμού (practical reasoning).

Ο πρακτικός λογισμός αποτελεί αντικείμενο μελέτης από φιλοσόφους και σχετίζεται με θεωρίες που απευθύνονται στην ανθρώπινη συμπεριφορά. Με αυτόν τον όρο απλά εννοούμε τον πραγματικό λογισμό που χρησιμοποιούμε για να αποφασίσουμε τι να κάνουμε. Η ονομασία της αρχιτεκτονικής BDI οφείλεται στα αρχικά των λέξεων beliefs (B), desires (D), intentions (I), στοιχεία που αποτελούν τα βασικά χαρακτηριστικά της αρχιτεκτονικής.

Οι πεποιθήσεις, οι επιθυμίες και οι προθέσεις αναπαριστούνται ρητά ως μεταβλητές των γλωσσών προγραμματισμού ή ως δομές δεδομένων. Επιπλέον, διατηρείται μία βιβλιοθήκη πλάνων (plan library) που περιγράφει τους τρόπους με τους οποίους μπορούν να επιτευχθούν οι διάφοροι στόχοι. Ουσιαστικά, ένα πλάνο είναι μία μερική δέσμευση στον τρόπο που θα επιτευχθεί μία επιθυμία. Στο Σχήμα 2.1 παρουσιάζεται η εσωτερική δομή ενός διαμεσολαβητή σύμφωνα με τη αρχιτεκτονική BDI. Ο μηχανισμός παραγωγής συμπερασμάτων (inference engine) αναπαριστά και διαχειρίζεται την νοητική κατάσταση του διαμεσολαβητή όπως επίσης και τις ενέργειες επικοινωνίας με το περιβάλλον. Το συστατικό επικοινωνίας (communication component) αναπαριστά ένα ξεχωριστό μηχανισμό υπεύθυνο για την επικοινωνία του διαμεσολαβητή. Ωστόσο, θα μπορούσε και να παραλειφθεί, αφού σε αρκετές περιπτώσεις, κυρίως ομοιογενών συστημάτων πολλαπλών διαμεσολαβητών, η επικοινωνία γίνεται μέσω των άλλων συστατικών.



Σχήμα 2: Η εσωτερική δομή ενός διαμεσολαβητή σύμφωνα με την αρχιτεκτονική BDI

Πολλά συστήματα διαμεσολαβητών έχουν βασιστεί σε αυτή την αρχιτεκτονική και κάποια αναφέρονται πιο αναλυτικά παρακάτω.

Στις ίδιες αρχές βασίζεται και η αρχιτεκτονική IRMA (Intelligent Resource–bounded Machine Architecture) [5]. Περιέχει βιβλιοθήκη πλάνων και κατάλληλες δομές δεδομένων για την αναπαράσταση των πεποιθήσεων, των επιθυμιών και των προθέσεων. Επίσης, περιέχει ένα σύνολο άλλων υποσυστημάτων για την παραγωγή συμπερασμάτων και επιλογών, για τον καθορισμό των προθέσεων και των ενεργειών και την παρατήρηση του περιβάλλοντος.

Οι αρχιτεκτονικές GRATE* [20] και DESIRE [6] είναι επίσης βασισμένες στην αρχιτεκτονική BDI. Είναι αρχιτεκτονικές πολλαπλών επιπέδων (multi layered architectures) και εκτός από τις πεποιθήσεις, τις επιθυμίες και τις προθέσεις, η νοητική κατάσταση των διαμεσολαβητών χαρακτηρίζεται και από κοινές προθέσεις (joint intentions). Τα διάφορα επίπεδα είναι υπεύθυνα για την οργάνωση, τη συνεργασία και τον καθορισμό των ενεργειών. Έχουν εφαρμοστεί στη διαχείριση μεταφοράς ηλεκτρικού ρεύματος και το όλο σύστημα χωρίζεται σε διάφορα υποσυστήματα αποτελούμενα από αυτόνομους διαμεσολαβητές που επικοινωνούν και ενεργούν σε συνεργασία μεταξύ τους προκειμένου να διασφαλίσουν τη σωστή και αδιάκοπη λειτουργία ενός σταθμού ηλεκτρικού ρεύματος. Επίσης, μπορούν να κάνουν σωστή και άμεση διάγνωση βλαβών και να εξασφαλίσουν άμεση επιδιόρθωση.

6.2.2 Αρχιτεκτονικές Αντίδρασης

Σε αντίθεση με τις προηγούμενες απόψεις έρχονται οι απόψεις του R. Brooks [8], οι οποίες υιοθετήθηκαν αργότερα και από άλλους ερευνητές. Γενικά, αυτές οι απόψεις απορρίπτουν οποιαδήποτε συμβολική αναπαράσταση και μηχανισμούς διαχείρισης αυτής. Οι αρχιτεκτονικές που ανήκουν σε αυτή την κατηγορία καλούνται αρχιτεκτονικές αντίδρασης (reactive architectures)[5].

Η βασική ιδέα αυτών των αρχιτεκτονικών είναι ότι η έξυπνη συμπεριφορά ενός συστήματος είναι κάτι υποκειμενικό και κρίνεται από τον παρατηρητή του συστήματος, ενώ ταυτόχρονα δεν είναι απαραίτητη ούτε η συμβολική αναπαράσταση ούτε οι διαδικασίες συλλογισμού που εφαρμόζονται σε αυτή. Αντίθετα, η έξυπνη συμπεριφορά μπορεί να προκύψει από τον συνδυασμό πολλών στοιχειωδών ενεργειών και από τον τρόπο που αλληλεπιδρά ένας διαμεσολαβητής με το περιβάλλον του.

Πιο συγκεκριμένα, τρία είναι τα βασικά στοιχεία που χαρακτηρίζουν τους διαμεσολαβητές που στηρίζονται σε αυτές τις αρχιτεκτονικές, οι οποίοι καλούνται διαμεσολαβητές αντίδρασης (reactive agents) [26]:

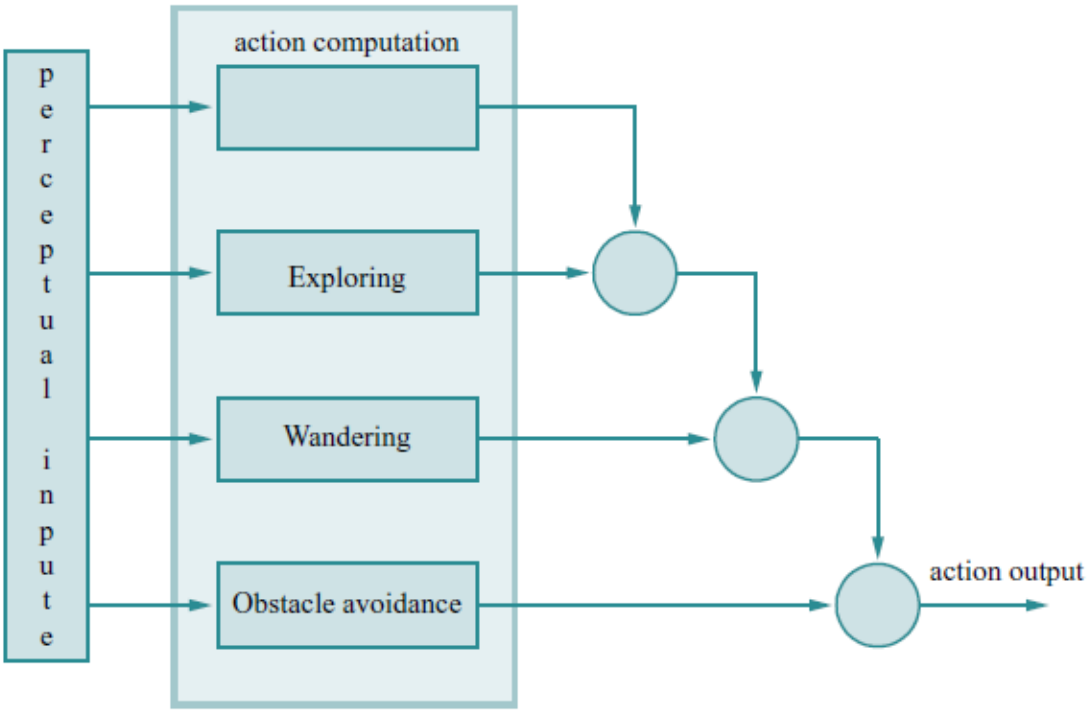
- Οι διαμεσολαβητές είναι σχετικά απλοί και αλληλεπιδρούν μεταξύ τους με στοιχειώδη τρόπο. Έτσι, δεν υπάρχουν εκ των προτέρων προδιαγραφές της

[5] Ο όρος «reactive architecture» και κατ' επέκταση «reactive agent» αποδίδεται από άλλους επιστήμονες δικαιολογημένα ως «αρχιτεκτονική αλληλεπίδρασης» και αντίστοιχα «διαμεσολαβητές που αλληλεπιδρούν». Όπως φαίνεται και στη συνέχεια της ενότητας, μία τέτοια απόδοση στηρίζεται στην αλληλεπίδραση των βασικών συστατικών μίας τέτοιας αρχιτεκτονικής και αντίστοιχα ενός τέτοιου διαμεσολαβητή. Όμως, μία τέτοια απόδοση μπορεί να οδηγήσει στην παρανόηση ότι οι άλλες αρχιτεκτονικές (και οι διαμεσολαβητές) δεν έχουν χαρακτηριστικά αλληλεπίδρασης. Στο παρόν σύγγραμμα έχει προτιμηθεί η απόδοση του όρου ως «αντίδραση», η οποία αντανακλά τη βασική ιδιότητα των αρχιτεκτονικών και των διαμεσολαβητών αυτού του είδους, δηλαδή την άμεση αντίδραση στα ερεθίσματα (stimuli) του περιβάλλοντος.

συμπεριφοράς των διαμεσολαβητών. Ωστόσο, από αυτές τις αλληλεπιδράσεις προκύπτει σύνθετη συμπεριφορά, από μία καθολική παρατήρηση του συστήματος.

- Ένας διαμεσολαβητής θεωρείται ως μία συλλογή διαφόρων συστατικών που λειτουργούν αυτόνομα και είναι υπεύθυνα για συγκεκριμένα έργα. Η επικοινωνία μεταξύ αυτών των συστατικών είναι η ελάχιστη δυνατή και αρκετά χαμηλού επιπέδου (δηλαδή κοντά σε επίπεδο μηχανής). Η συνολική συμπεριφορά του διαμεσολαβητή προκύπτει από τα συστατικά του, αφού δεν υπάρχει κάποιο μοντέλο αυτής στους διαμεσολαβητές.
- Οι διαμεσολαβητές λειτουργούν βασισμένοι στην επεξεργασία στοιχειωδών αναπαραστάσεων, όπως τα δεδομένα που προέρχονται από διάφορους αισθητήρες.

Αρχικά, η υλοποίηση συστημάτων βασισμένων σε τέτοιου είδους αρχιτεκτονικές αφορούσε κυρίως robots, ωστόσο δεν περιορίστηκε εκεί. Για να αποδείξει τους ισχυρισμούς του ο Brooks προχώρησε στην υλοποίηση συστημάτων (κυρίως hardware) βασισμένων στην αρχιτεκτονική υπαγωγής (subsumption architecture) [7]. Η αρχιτεκτονική υπαγωγής αποτελείται από ένα σύνολο στοιχειωδών συμπεριφορών που συνδυάζονται μεταξύ τους με αποτέλεσμα μία έξυπνη δραστηριότητα για την επίτευξη ενός έργου. Η αρχιτεκτονική είναι πολυεπίπεδη (multi layer architecture) όπου κάθε επίπεδο περιλαμβάνει συστατικά συμπεριφοράς (behavior modules) για την αντίληψη γεγονότων στο περιβάλλον και την εκτέλεση αντίστοιχων στοιχειωδών ενεργειών (Σχήμα 3). Στα κατώτερα επίπεδα αναπαριστώνται περισσότερο συγκεκριμένες συμπεριφορές, όπως η αποφυγή αντικειμένων, ενώ στα ανώτερα επίπεδα οι συμπεριφορές είναι πιο αφηρημένες. Ωστόσο, ο τρόπος που θα ενεργήσει το σύστημα εξαρτάται από τις αλληλεπιδράσεις των συστατικών συμπεριφοράς και αυτό διότι μία συμπεριφορά μπορεί να εμποδίσει κάποια άλλη. Παρόμοια προσέγγιση ακολούθησε και ο Steels στο σύστημα «Mars explorer» [43].



Σχήμα 3: Αρχιτεκτονική Υπαγωγής

Το σύστημα PENGI [1] κατασκευασμένο σύμφωνα με τις αρχιτεκτονικές αντίδρασης είναι ένα παιγνίδι εξομοίωσης και περιέχει καταγραμμένες σε ψηφιακά κυκλώματα αρκετές καθημερινές απλές δραστηριότητες, οι οποίες εκτελούνται μηχανικά και χωρίς ιδιαίτερη σκέψη. Ωστόσο, για να μπορέσει το σύστημα να εκτελέσει νέες ενέργειες που δεν έχει καταγραμμένες, θα πρέπει αυτές να κωδικοποιηθούν ξανά στο κύκλωμα. Με παρόμοιο τρόπο τα situated automata [21] περιέχουν καταγραμμένη σε κυκλώματα τη συμπεριφορά ενός διαμεσολαβητή όπως αυτή έχει περιγραφεί προηγουμένως με τυπικές προδιαγραφές σε όρους των συστημάτων προθέσεων. Κάπως διαφορετική είναι η προσέγγιση στην Αρχιτεκτονική Δικτύου Διαμεσολαβητών (Agent Network Architecture – ANA) [27], όπου ένα σύνολο δομικών στοιχείων συνδυάζονται σε μορφή δικτύου ανάλογα με τις συνθήκες που επικρατούν κάθε στιγμή στο περιβάλλον. Η συμπεριφορά κάθε συστατικού καθορίζεται από το σχεδιαστή ως ένα σύνολο προ– και μετα– συνθηκών, ενώ ένα επίπεδο ενεργοποίησης (activation level) εκτιμά ποιο συστατικό θα δραστηριοποιηθεί. Τα συστατικά με τα ανώτερα στην ιεραρχία επίπεδα δραστηριοποίησης είναι περισσότερο πιθανά να δραστηριοποιηθούν.

6.2.3 Υβριδικές Αρχιτεκτονικές

Οι υβριδικές αρχιτεκτονικές (hybrid architectures) είναι συνήθως πολυεπίπεδες και χρησιμοποιούν συνήθως συμβολική αναπαράσταση και μηχανισμούς για τη γνώση, την παραγωγή συμπερασμάτων και τη λήψη αποφάσεων (αρχιτεκτονικές νοημοσύνης) στα ανώτερα επίπεδα της ιεραρχίας, ενώ διαθέτουν μηχανισμούς (κυρίως hardware) για την παρατήρηση του περιβάλλοντος και την άμεση αντίδραση σε κρίσιμα γεγονότα (αρχιτεκτονικές αντίδρασης) στα κατώτερα επίπεδα της ιεραρχίας. Έτσι, χρησιμοποιώντας κατάλληλα στοιχεία και από τις δύο παραπάνω αρχιτεκτονικές, εκμεταλλεύονται τα πλεονεκτήματα αυτών αυξάνοντας την απόδοση του συστήματος.

Ένα τέτοιο σύστημα είναι το Procedural Reasoning System (PRS) [15] βασισμένο στην αρχιτεκτονική BDI. Οι πεποιθήσεις εκφράζονται σε λογική πρώτης τάξεως, ενώ οι επιθυμίες αποτελούν δυναμικές αναπαραστάσεις της συμπεριφοράς του συστήματος. Τα πλάνα είναι μερικώς επεξεργασμένα και μπορούν να ενεργοποιηθούν κατά περίπτωση, δηλαδή είτε για την επίτευξη κάποιου απώτερου στόχου (αρχιτεκτονική νοημοσύνης) είτε για άμεση απόκριση στις αλλαγές του περιβάλλοντος (αρχιτεκτονική αντίδρασης). Τα ενεργά πλάνα αποτελούν τις προθέσεις του συστήματος. Το PRS έχει εφαρμοστεί στην εξομοίωση των διαδικασιών συντήρησης των διαστημικών λεωφορείων. Παρόμοιες τεχνικές ακολουθούνται και στο COSY [17] που είναι άλλη μία αρχιτεκτονική BDI που όμως περιλαμβάνει αισθητήρες και άλλες συσκευές για την αλληλεπίδραση με το περιβάλλον.

Τα TOURINGMACHINES [13] είναι μία υβριδική αρχιτεκτονική τριών επιπέδων, με το ένα επίπεδο υπεύθυνο για την αλληλεπίδραση με το περιβάλλον (reactive layer) και τα άλλα δύο για τη μοντελοποίηση της νοητικής και γνωστικής κατάστασης και για το σχεδιασμό των ενεργειών των διαμεσολαβητών (modeling και planning layers).

Σε υβριδική αρχιτεκτονική τριών επιπέδων βασίζεται και το INTERRAP [35], με ιεράρχηση διαφόρων βάσεων γνώσης και των μοντέλων του κόσμου (αρχιτεκτονική νοημοσύνης) και της συμπεριφοράς (αρχιτεκτονική αντίδρασης). Στο κατώτερο επίπεδο υλοποιούνται οι ικανότητες απόκρισης του συστήματος σε μορφή κανόνων ενεργειών ανάλογα με την τρέχουσα κατάσταση. Στο μεσαίο επίπεδο υπάρχουν τα

πλάνα για το σχεδιασμό των ενεργειών ανάλογα με τη νοητική κατάσταση του διαμεσολαβητή, ενώ στο ανώτερο επίπεδο παρέχεται η δυνατότητα ενεργειών επικοινωνίας και συνεργασίας με άλλους διαμεσολαβητές. Το INTERRAP έχει για την εξομοίωση robots που λειτουργούν σε περιβάλλον τοποθέτησης φορτίων σε αποβάθρες.

Βιβλιογραφία

Γενική Βιβλιογραφία

1] R. Elmasri, S.B. Navathe "Θεμελιώδεις Αρχές Συστημάτων Βάσεων Δεδομένων", Α' και Β' τόμος- 5η έκδοση αναθεωρημένη.

2] «Συστήματα Βάσεων Δεδομένων - Η Πλήρης Θεωρία των Βάσεων Δεδομένων - Silberschatz, Korth, Sudarshan - Εκδόσεις Μ.Γκιούρδας -4η Έκδοση»

3] Ευφυή Πληροφοριακά Συστήματα και Ευρετικές Μέθοδοι, Δρ. Ιωάννης Ζαχαράκης, Πανεπιστημιακές Εκδόσεις ΕΑΠ.

4] Intelligent Information Systems: Meeting the Challenge of the Knowledge Era, Sue A. Davis, Alan Rowe, 1996

5] Advances in Intelligent Information and Database Systems (Studies in Computational Intelligence), Ngoc Thanh Nguyen (Editor), Radoslaw Katarzyniak (Editor), Shyi-Ming Chen (Editor), poster session organized during the 2nd Asian Conference on Intelligent Information and Database Systems (24-26 March 2010 in Hue, Vietnam).

6] Distributed Intelligent Systems: A Coordination Perspective, Abdellah Bedrouni (Author), Ranjeev Mittu (Author), Abdeslem Boukhtouta (Author), Jean Berger (Author), Springer; 1 edition (August 21, 2009), ISBN-13: 978-0387777016.

Βιβλιογραφία Κεφαλαίων 5 και 6

Τα κεφάλαια 5 και 6 προέρχονται από το σύγγραμμα «Ευφυή Πληροφοριακά Συστήματα και Ευρετικές Μέθοδοι, Δρ. Ιωάννης Ζαχαράκης, Πανεπιστημιακές Εκδόσεις ΕΑΠ, Πάτρα 2001»

Κεφάλαιο 5

1] Grosz, B., and Davis, R. (1994). A report to ARPA on twenty–first century intelligent systems, AI Magazine 15(3):10–20.

2] Weld, D. (ed.) (1997). The Role of Intelligent Systems in the National Information Infrastructure, The American Association for Artificial Intelligence, (available in http://www.aaai.org/Policy/Papers/nii.html).

3] McCarthy, J. (1998). What is Artificial Intelligence? Computer Science Department, Stanford University, Stanford CA 94305, (available in http://www–formal.stanford.edu/jmc/).

Κεφάλαιο 6

1] Agre, P. and Chapman, D. (1987). PENGI: An Implementation of a Theory of Activity. In Proceedings of the Sixth National Conference on Artificial Intelligence (AAAI–87), pp. 268–272, Seattle, WA.

2] Bates, J. (1994). The role of emotion in believable agents. Communications of the ACM, 37(7): 122–125.

3] Baujard, O., Garbay, C. (1993). KISS, a Multi–Agent Segmentation System. Special Issue From Numeric to Symbolic Image Processing: Systems and Applications, SPIE Optical Engineering, 32(6): 1235–1249.

4] Baujard O., Pesty S. and Garbay C. (1993). MAPS: a Language for MultiAgent System Design. Expert Systems, 11(2): 89–97.

5] Bratman, M. E., Israel, D. J., and Pollack, M. E. (1988). Plans and Resource bounded Practical Reasoning. Computational Intelligence, 4:349–355.

6] Brazier F. M. T., Dunin–Keplicz B. M., Jennings N. R., and Treur J.(1996): DESIRE: Modelling Multi–Agent Systems in a Compositional Formal Framework. International Journal of Cooperative Information Systems, 6(1): 67–94.

7] Brooks, R. A. (1986). A Robust Layered Control System for a Mobile Robot. IEEE Journal of Robotics and Automation, 2(1):14–23.

8] Brooks, R. A. (1991). Intelligence without representation. Artificial Intelligence, 47:139–159.

9] Chavez, A., Maes, P. (1996). Kasbah: An Agent Marketplace for Buying and Selling Goods. Proceedings of the First International Conference on the Practical Application of Intelligent Agents and Multi–Agent Systems, London, UK.

10] Cheikes, B. A. (1995). GIA: An Agent–Based Architecture for Intelligent Tutoring Systems. In Proceedings of the CIKM'95 Workshop on Intelligent Information Agents.

11] Cohen, P. R. and Levesque, H. J. (1990). Intention is Choice with Commitment. Artificial Intelligence, 42:213–261.

12] Dennett, D. C. (1987). The Intentional Stance. The MIT Press: Cambridge, MA.

13] Ferguson, I. A. (1992). TouringMachines: An Architecture for Dynamic, Rational, Mobile Agents. Ph.D. Thesis, Clare Hall, University of Cambridge, UK