Θέματα Μοντελοποίησης και Επερώτησης Πολυδιάστατων Βάσεων Δεδομένων

Π. Βασιλειάδης, Σ. Σκιαδόπουλος, Τ. Σελλής

Εθνικό Μετσόβιο Πολυτεχνείο

Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών, Τομέας Πληροφορικής

Εργαστήριο Συστημάτων Βάσεων Γνώσεων και Δεδομένων

Ζωγράφου 15773, Αθήνα, Ελλάδα

Τηλ: +301-772-1402, Fax: +301-772-1442

{pvassil,spiros,timos}@dbnet.ece.ntua.gr

Περίληψη

Η Σύγχρονη Αναλυτική Επεξεργασία Δεδομένων (On-Line Analytical Processing - OLAP) είναι μια τάση στην τεχνολογία των βάσεων δεδομένων, που στηρίζεται στη θεώρηση της πληροφορίας με πολυδιάστατο τρόπο στο επίπεδο των πελατών. Παρά την κοινή αποδοχή των πολυδιάστατων κύβων σαν το κεντρικό λογικό μοντέλο για OLAP και την πληθώρα των ερευνητικών προτάσεων, υπάρχει μικρή συμφωνία στην εύρεση μιας κοινής ορολογίας και σημασιολογίας για το λογικό μοντέλο δεδομένων. Στο άρθρο αυτό προτείνεται ένα επιπλέον λογικό μοντέλο για κύβους, με βάση την παρατήρηση ότι ένας κύβος δεν είναι μια αυθύπαρκτη οντότητα, αλλά μια όψη πάνω σε ένα υποκείμενο σύνολο δεδομένων. Το προτεινόμενο μοντέλο είναι αρκετά ισχυρό στο να καλύπτει όλες τις συνηθισμένες πράξεις OLAP όπως επιλογή, συναθροιστική άνοδος και αναλυτική κάθοδος σε επίπεδα αδρομέρειας, μέσω μιας συνεπούς και πλήρης άλγεβρας. Δείχνεται επίσης πώς αυτό το μοντέλο μπορεί να χρησιμοποιηθεί σαν η βάση για την επεξεργασία λειτουργιών στους κύβους και παρουσιάζονται συντακτικοί χαρακτηρισμοί για τα προβλήματα της χρησιμότητας κύβων (ήτοι, του προβλήματος χρησιμοποιήσεως δεδομένων από κάποιον κύβο για να υπολογιστεί ένας άλλος κύβος).

1. ΕΙΣΑΓΩΓΗ

Η Σύγχρονη Αναλυτική Επεξεργασία Δεδομένων (On-Line Analytical Processing - OLAP) είναι μια τάση στην τεχνολογία των βάσεων δεδομένων, που στηρίζεται στη θεώρηση της πληροφορίας με πολυδιάστατο τρόπο στο επίπεδο των πελατών. Υπάρχουν διάφορα εννοιολογικά (π.χ. [BaSa98],[Kimb96]) και φυσικά (π.χ. [Sara97]) μοντέλα. Τα παραδοσιακά λογικά μοντέλα, όμως, όπως για παράδειγμα το σχεσιακό, δεν φαίνονται ιδιαίτερα ικανά να παράσχουν αποδοτικά τη λειτουργικότητα που χρειάζεται για τη συνάθροιση, θεώρηση και διαχείριση της πληροφορίας που υπάρχει στις συγκεντρωτικές αποθήκες δεδομένων [JLVV00]. Παρά τη συμφωνία στον κεντρικό ρόλο των πολυδιάστατων κύβων και την πληθώρα των ερευνητικών προτάσεων, υπάρχει μικρή συμφωνία στην εύρεση μιας κοινής ορολογίας και σημασιολογίας για το λογικό μοντέλο δεδομένων.

Υπάρχουν διάφορα σχετικά πρότυπα ήδη [OLAP97a, TPC99, Meta97, Micr98], αλλά εκτός από το τελευταίο, κανένα δε φαίνεται να παρέχει ένα καλά ορισμένο μοντέλο για λειτουργίες OLAP. Στον ακαδημαϊκό χώρο υπάρχουν διάφορες προτάσεις για τη μοντελοποίηση κύβων [AgGS95, LiWa96, GyLa97, BaPT97, CaTo97, Lehn98, Vass98]. Παρά τις προσπάθειες αυτές, πιστεύουμε ότι διάφορα βασικά χαρακτηριστικά των κύβων δεν έχουν τονιστεί, ούτε από τον ακαδημαϊκό χώρο, ούτε από το χώρο της παραγωγής. Για το λόγο αυτό, παρουσιάζουμε ένα λογικό μοντέλο πολυδιάστατων βάσεων δεδομένων. Το μοντέλο επεκτείνει την πρόταση του [Vass98] με τρόπο πιο συστηματικό και κομψό. Το μοντέλο αφορά όλες τις συχνά συναντώμενες οντότητες ενός πολυδιάστατου μοντέλου (ιεραρχίες διαστάσεων, κύβοι και λειτουργίες τους) χωρίς να περιορίζεται από τη φυσική τους αναπαράσταση (π.χ., ROLAP ή MOLAP αρχιτεκτονικές). Μια από τις βασικές μας παρατηρήσεις είναι ότι *ένας κύβος δεν είναι μια αυθύπαρκτη οντότητα, αλλά μια όψη* (αποθηκευμένη ή όχι) *πάνω σε ένα υποκείμενο σύνολο δεδομένων*. Η ιδιότητα αυτή μας επιτρέπει να αναπτύξουμε σύνθετες λειτουργίες που δεν υποστηρίζονται από άλλα μοντέλα (π.χ. η πράξη drill-down και η αλλαγή αθροιστικής συνάρτησης).

Εξ' όσων γνωρίζουμε, τα εργαλεία OLAP συμπεριφέρονται με τρόπο προσανατολισμένο στα δεδομένα. Οι κύβοι αντιμετωπίζονται απλά σαν σύνολα δεδομένων, αγνοώντας ότι παράγονται σαν ερωτήσεις πάνω στο υποκείμενο σύνολο δεδομένων. Το μοντέλο που προτείνουμε εδώ, ακολουθεί μια διαφορετική στρατηγική: κρατάμε την ιστορία των επιλογών που έχουν γίνει και έτσι μπορούμε να παράγουμε ένα κύβο από τη δηλωτική του περιγραφή. Έτσι, μπορούμε να ορίσουμε πιο σύνθετες λειτουργίες και σειρές λειτουργιών. Το μοντέλο συνοδεύεται από μια ισχυρή άλγεβρα που μοντελοποιεί τις συνήθεις λειτουργίες OLAP όπως (α) *επιλογή* πάνω σε ένα κύβο, (β) *συναθροιστική άνοδος* (*roll-up*), που σημαίνει συνάθροιση ενός κύβου σε υψηλότερο επίπεδο και (γ) *αναλυτική κάθοδος* (*drill-down*), που αφορά την ανάλυση σε πιο λεπτομερή παρουσίαση της πληροφορίας.

Η συμβολή δεν περιορίζεται στην εκφραστικότητα των αποτελεσμάτων, αλλά επεκτείνεται και σε αποτελέσματα βελτιστοποίησης. Ερευνήσαμε το πρόβλημα της *χρησιμότητας κύβων* (*cube usability problem*), μια παραλλαγή του προβλήματος της χρησιμότητας όψεων, δηλαδή. Συνοδεύουμε το μοντέλο με τεχνικές που επεκτείνουν γνωστές σχεσιακές τεχνικές για τον εγκλεισμό συνθηκών επιλογής [Ullm89]. Παρατηρήσαμε ότι αν και πολύ δουλειά έχει υπάρξει στο χώρο του εγκλεισμού επερωτήσεων και χρησιμότητας όψεων στις σχεσιακές βάσεις δεδομένων [DJLS96, GuHQ95, NuSS98, CKPS95], δεν υπάρχουν αποτελέσματα που να λαμβάνουν υπόψη τους την ύπαρξη ιεραρχιών διαστάσεων, όπως στις πολυδιάστατες βάσεις δεδομένων. Παρουσιάζουμε, λοιπόν, αποτελέσματα σε δύο ζητήματα. Πρώτα αντιμετωπίζουμε το ζήτημα του εγκλεισμού δύο συνθηκών επιλογής σε σχέση με τις οριακές τους συνθήκες. Κατά δεύτερον παρουσιάζουμε ένα σύνολο αξιωμάτων που αποκρίνονται για τον εγκλεισμό συνθηκών επιλογής που περιλαμβάνουν διαφορετικά επίπεδα της ίδιας ιεραρχίας, επεκτείνοντας τα αποτελέσματα που υπάρχουν για το σχεσιακό μοντέλο, αλλά για ένα πεδίο ορισμού μόνο [Ullm89]. Τα αποτελέσματά μας έχουν δημοσιευθεί στα [VaSk00] και [VaSe99]. Η πλήρης εκδοχή τους (που περιλαμβάνει και όλες τις αποδείξεις και επιπλέον παραδείγματα) βρίσκεται στο [VaSk99].

Το άρθρο αυτό οργανώνεται ως εξής: στην ενότητα 2 παρουσιάζουμε τον χώρο της τεχνολογίας OLAP, στην ενότητα 3 παρουσιάζουμε το λογικό μοντέλο κύβων και στην ενότητα 4 παρουσιάζουμε τα αποτελέσματα βελτιστοποίησης. Τέλος στην ενότητα 5 συνοψίζουμε και παρουσιάζουμε περαιτέρω ερευνητικά προβλήματα.

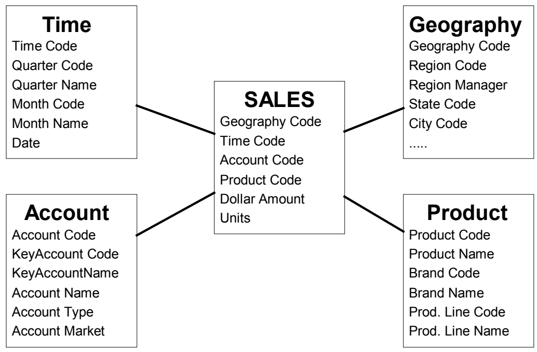
2. ΒΑΣΙΚΕΣ ΕΝΝΟΙΕΣ ΣΥΓΧΡΟΝΗΣ ΑΝΑΛΥΤΙΚΗΣ ΕΠΕΞΕΡΓΑΣΙΑΣ ΔΕΔΟΜΕΝΩΝ

Ένας καλός ορισμός του όρου OLAP βρίσκεται στο [OLAP97]: «…Η Σύγχρονη Αναλυτική Επεξεργασία Δεδομένων (On-Line Analytical Processing -OLAP) είναι μια κατηγορία λογισμικού που επιτρέπει σε αναλυτές και διοικητικά στελέχη να αποκτήσουν γνώση των δεδομένων μέσω μιας γρήγορης, συνεπούς και αξιόπιστης πρόσβασης σε μια μεγάλη ποικιλία όψεων της πληροφορίας που έχει μετασχηματιστεί από απλά δεδομένα, ώστε να αναπαριστά τη πολυδιάστατη θεώρηση ενός οργανισμού, όπως γίνεται αντιληπτή από το χρήστη. Η λειτουργικότητα του OLAP χαρακτηρίζεται από τη δυναμική πολυδιάστατη ανάλυση συναθροισμένων δεδομένων του οργανισμού, που υποστηρίζουν τον τελικό χρήστη στις αναλυτικές πλοηγήσεις του, συμπεριλαμβάνοντας υπολογισμούς και μοντελοποιήσεις που βασίζονται σε διαστάσεις, ιεραρχίες και επίπεδα, ανάλυση τάσεων σε συνεχείς χρονικές περιόδους, τεμαχισμός υποσυνόλων για προβολές στην οθόνη, ανάλυση σε χαμηλότερα επίπεδα συνάθροισης, περιστροφή για νέες συγκρίσεις διαστάσεων κλπ…». Μια τυπική ορολογία για OLAP παρέχεται από το OLAP Council στο [OLAP97].

Τα εργαλεία OLAP εστιάζουν στην παροχή πολυδιάστατης ανάλυσης της πληροφορίας. Για να επιτύχουν το στόχο τους, χρησιμοποιούν πολυδιάστατη μοντελοποίηση και αποθήκευση των δεδομένων. Τα δεδομένα οργανώνονται σε *κύβους* ή αλλιώς *υπερκύβους* (*cubes* και *hypercubes* αντίστοιχα), που ορίζονται σε πολυδιάστατους χώρους, αποτελούμενους από πολλές διαστάσεις. Κάθε διάσταση αποτελείται από πολλά επίπεδα συνάθροισης. Τυπικές λειτουργίες OLAP περιλαμβάνουν την συνάθροιση ή ανάλυση της πληροφορίας (μέσω των πράξεων *roll-up* και *drill-down*), την επιλογή (*selection*) τμημάτων της και περιστροφής της παρουσίασης της, με βάση τις διαστάσεις της, στην οθόνη (*pivoting)*.

Υπάρχουν δύο πόλοι γύρω από τους οποίους συγκεντρώνονται τα εργαλεία OLAP και οι οποίοι έχουν να κάνουν με τη φυσική αποθήκευση των δεδομένων. Από τη μία πλευρά υπάρχει η *αρχιτεκτονική ROLAP* (Relational On-Line Analytical Processing) [MStr97, Info97, RBSI97], υποστηριζόμενη και από την άλλη υπάρχει η *αρχιτεκτονική MOLAP* (Multidimensional On-Line Analytical Processing) [Arbo96]. Το πλεονέκτημα της αρχιτεκτονικής MOLAP είναι ότι παρέχει μια άμεση πολυδιάστατη όψη των δεδομένων, ενώ το πλεονέκτημα της αρχιτεκτονικής ROLAP είναι απλά μια πολυδιάστατη διαπροσωπεία σε σχεσιακά συστήματα βάσεων δεδομένων. Η αρχιτεκτονική ROLAP έχει δύο πλεονεκτήματα: (α) μπορεί να ενσωματωθεί εύκολα σε υπάρχοντα σχεσιακά συστήματα, και (β) τα σχεσιακά δεδομένα μπορούν να αποθηκευθούν πιο αποδοτικά από τα πολυδιάστατα δεδομένα.

Σε μια αρχιτεκτονική ROLAP, τα δεδομένα οργανώνονται σε *σχήμα αστέρα* ή *νιφάδας* (*star* ή *snowflake* schema) (Σχήμα 1). Ένα σχήμα αστέρα αποτελείται από ένα κεντρικό *πίνακα πληροφοριών* (*fact* table) και διάφορους αποκανονικοποιημένους *πίνακες διάστασης* (*dimension* tables). Τα *μέτρα* (*measures*) της πληροφορίας αποθηκεύονται στον πίνακα πληροφοριών (π.χ. Dollar Amount, Units στον πίνακα SALES). Για κάθε διάσταση του πολυδιάστατου μοντέλου, υπάρχει και ένας πίνακας διάστασης (π.χ. Geography, Product, Time, Account) με όλα τα επίπεδα συνάθροισης και τις επιπλέον ιδιότητες των επιπέδων αυτών. Η κανονικοποιημένη εκδοχή του σχήματος αστέρα είναι το σχήμα νιφάδας όπου κάθε επίπεδο μιας διάστασης έχει το δικό του πίνακα.



Σχήμα 1. Σχήμα αστέρα [STGI96]

Τα πολυδιάστατα συστήματα βάσεων δεδομένων (Multidimensional database systems -MDBMS) αποθηκεύουν τα δεδομένα σε πολυδιάστατους πίνακες. Κάθε διάσταση του πίνακα αναπαριστά μια διάσταση του κύβου. Τα περιεχόμενα του πίνακα είναι τα μέτρα του κύβου. Τα συστήματα MDBMS απαιτούν τον προϋπολογισμό όλων των πιθανών συναθροίσεων, γι' αυτό και είναι πιο γρήγορα από τα τυπικά σχεσιακά συστήματα [Coll96], αλλά πιο απαιτητικά στην ανανέωση της πληροφορίας και τη διαχείρισή της.

Ο αναγνώστης παραπέμπεται στο [VaSe99] για μια αναλυτική παρουσίαση των σχετικών προσπαθειών στο χώρο των λογικών μοντέλων για πολυδιάστατες βάσεις δεδομένων, σε σχέση με τα εργαλεία που χρησιμοποιούνται και τις ακαδημαϊκές προσπάθειες, οι οποίες αφορούν επεκτάσεις του σχεσιακού μοντέλου, αλλά και προσεγγίσεις προσανατολισμένες στον πολυδιάστατο χαρακτήρα των κύβων.

3. ΚΥΒΟΙ ΓΙΑ ΠΟΛΥΔΙΑΣΤΑΤΕΣ ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ

Στην ενότητα αυτή παρουσιάζουμε τις βασικές οντότητες και λειτουργίες του προτεινόμενου μοντέλου. Οι οντότητες αυτές συμπεριλαμβάνουν διαστάσεις, σύνολα δεδομένων και κύβους. Οι λειτουργίες περιλαμβάνουν επιλογές και αλλαγή στη λεπτομέρεια των δεδομένων. Το μοντέλο επεκτείνει προηγούμενες προσεγγίσεις και συγκεκριμένα τις [Vass98, CaTo97, Lehn98].

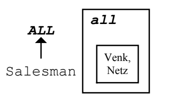
Ένα από τα κύρια χαρακτηριστικά των OLAP εφαρμογών είναι η πολυδιάστατη θεώρηση των δεδομένων (*multidimensional view of data*) σε ότι αφορά τον τρόπο με τον οποίο τα αντιμετωπίζει ο χρήστης. Εν γένει, σε ότι αφορά το χρήστη, τα δεδομένα θεωρούνται αποθηκευμένα σε ένα *πολυδιάστατο πίνακα* (*multi-dimensional array*), ο οποίος αποκαλείται και *κύβος* ή *υπερκύβος* (*Cube* και *HyperCube* αντίστοιχα). Ο κύβος είναι μια ομάδα από *κελιά δεδομένων* (*data cells*). Κάθε κελί χαρακτηρίζεται μονοσήμαντα από τις αντίστοιχες τιμές των διαστάσεων του κύβου. Τα περιεχόμενα του κελιού ονομάζονται *μέτρα* (*measures*) και αναπαριστούν τις αποτιμώμενες αξίες του πραγματικού κόσμου. Τα μέτρα είναι συναρτησιακά εξαρτημένα, με τη σχεσιακή έννοια, από τις διαστάσεις του κύβου.

Μια *διάσταση* (*dimension*) ορίζεται ως "ένα δομικό χαρακτηριστικό ενός κύβου, που αποτελείται από μια λίστα τιμών, οι οποίες είναι όλες του ιδίου τύπου σε ότι αφορά την αντίληψη των δεδομένων από το χρήστη" [OLAP97]. Με άλλα λόγια, μια διάσταση μοντελοποιεί όλους τους τρόπους με τους οποίους τα δεδομένα μπορούν να συναθροιστούν σε σχέση με μια συγκεκριμένη παράμετρο του περιεχομένου τους. Κάθε διάσταση έχει μια σχετική *ιεραρχία επιπέδων* συνάθροισης των δεδομένων (*hierarchy of levels*). Αυτό σημαίνει, με απλά λόγια, ότι η διάσταση μπορεί να θεωρηθεί από πολλά επίπεδα λεπτομέρειας. Τυπικά, μια *διάσταση* D είναι ένα δίκτυο (lattice) $(\mathbf{L},\prec)$: $\mathbf{L}=(\mathrm{L}_1,\ldots,\mathrm{L}_n,\mathrm{ALL})$. Απαιτούμε το ανώτερο επίπεδο του δικτύου να είναι πάντα το επίπεδο ALL, έτσι ώστε να μπορούμε να ομαδοποιήσουμε όλες τις τιμές της διάστασης σε μία τιμή 'all'. Το κάτω όριο του δικτύου ονομάζεται *λεπτομερές επίπεδο* (*detailed level*) της διάστασης. Για παράδειγμα, ας θεωρήσουμε τη διάσταση Date του σχήματος 6. Τα επίπεδα της διάστασης Date είναι Day, Week, Month, Year και ALL. Το επίπεδο Day είναι το πλέον λεπτομερές επίπεδο. Το επίπεδο ALL είναι το πλέον υψηλό επίπεδο συνάθροισης για όλες τις διαστάσεις. Το να συναθροίζουμε την πληροφορία στο επίπεδο ALL μιας διάστασης σημαίνει ότι πρακτικά αγνοούμε τη διάσταση κατά τη συνάθροιση (συναθροίζουμε, δηλαδή, τα δεδομένα με βάση όλες τις άλλες διαστάσεις, πλην αυτής).

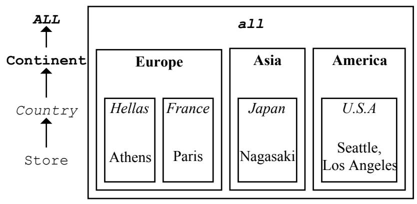
Η σχέση μεταξύ των τιμών των επιπέδων επιτυγχάνεται με τη χρήση ενός συνόλου συναρτήσεων της μορφής $\mathrm{anc}_{\mathrm{L1}}^{\mathrm{L2}}$. Μια συνάρτηση $\mathrm{anc}_{\mathrm{L1}}^{\mathrm{L2}}$ αντιστοιχίζει μια τιμή του επιπέδου L_2 στο επίπεδο L_1. Για παράδειγμα, $\mathrm{anc}_{\mathrm{Month}}^{\mathrm{Year}}(\mathrm{Feb{-}97})=1997$. Θα αποκαλούμε τις συναρτήσεις αυτές και *συναρτήσεις προγόνου* (ancestor functions).

Day	Title	Salesman	Store	Sales
6-Feb-97	Symposium	Netz	Paris	7
18-Feb-97	Karamazof brothers	Netz	Seatle	5
11-May-97	Ace of Spades	Netz	Los Angeles	20
3-Sep-97	Zarathustra	Netz	Nagasaki	50
3-Sep-97	Report to El Greco	Netz	Nagasaki	30
1-Jul-97	Ace of Spades	Venk	Athens	13
1-Jul-97	Piece of Mind	Venk	Athens	34

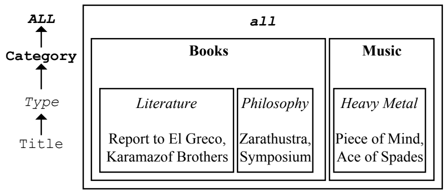
Σχήμα 2: Λεπτομερές Σύνολο Δεδομένων DS^0



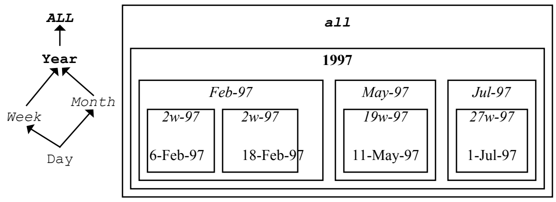
Σχήμα 3: Διάσταση Person



Σχήμα 4: Διάσταση Location



Σχήμα 5: Διάσταση Product



Σχήμα 6: Διάσταση Date

Οι κυριότερες λειτουργίες πολυδιάστατων βάσεων είναι η *επιλογή* (*selection*) και η *πλοήγηση* (*navigation*). Η επιλογή χρησιμοποιείται οπουδήποτε ένα κριτήριο εφαρμόζεται στα δεδομένα με βάση τις τιμές μιας διάστασης, με σκοπό να περιοριστεί το σύνολο των παρουσιαζομένων δεδομένων. Η πλοήγηση είναι ένας όρος για να περιγράψουμε τις διαδικασίες που χρησιμοποιούν οι χρήστες για να εξερευνούν την πληροφορία ενός κύβου με αλληλεπιδραστικό τρόπο, αλλάζοντας το επίπεδο λεπτομέρειας της πληροφορίας, όπως αυτή τους παρουσιάζεται [JLVV00, OLAP97]. Πιθανές πράξεις πλοήγησης είναι: (α) *Συναθροιστική Άνοδος* (*Roll-up*) που αντιστοιχεί στη συνάθροιση των δεδομένων από χαμηλότερο σε υψηλότερο επίπεδο λεπτομέρειας στην ιεραρχία μιας διάστασης, (β) *Αναλυτική Κάθοδος* (*Drill-Down*) που είναι η αντίστροφη λειτουργία της συναθροιστικής ανόδου και επιτρέπει την ανάλυση της πληροφορίας από υψηλότερο σε χαμηλότερο επίπεδο λεπτομέρειας, και (γ) *Τεμαχισμός* (*Slicing)* που αντιστοιχεί στην συνάθροιση των δεδομένων σε σχέση με ένα υποσύνολο μόνο των διαστάσεων του κύβου. Για παράδειγμα, ας θεωρήσουμε τη διάσταση `Date`: συναθροίζοντας από το επίπεδο `Month` στο επίπεδο `Year` είναι μια πράξη συναθροιστικής ανόδου και αναλύοντας από το επίπεδο `Month` στο επίπεδο `Day` είναι μια πράξη αναλυτικής καθόδου. Στο μοντέλο μας, ο τεμαχισμός είναι μια πράξη συναθροιστικής ανόδου στο επίπεδο `ALL`.

Στο προτεινόμενο μοντέλο, ορίζουμε την έννοια *σύνολο δεδομένων* (*data set*) σαν ένα σύνολο πλειάδων κάτω από ένα συγκεκριμένο σχήμα. Επιπλέον, υποθέτουμε την ύπαρξη ενός *λεπτομερούς συνόλου δεδομένων* (*detailed data set*), ενός συνόλου δεδομένων, δηλαδή, που ορίζεται στα πλέον λεπτομερή επίπεδα όλων των διαστάσεων του σχήματός του. Αυτό το λεπτομερές σύνολο δεδομένων είναι η κεντρική πηγή πληροφορίας, η οποία θα παρέχει δεδομένα σε όλους τους κύβους που θα παραχθούν κατά τη διάρκεια μιας OLAP συνόδου (π.χ., μπορεί να είναι ο πίνακας πληροφοριών σε μια συγκεντρωτική αποθήκη δεδομένων).

Μία από τις βασικές παρατηρήσεις μας είναι ότι ο κύβος δεν είναι μια ανεξάρτητη οντότητα (όπως συνήθως παρουσιάζεται στη βιβλιογραφία) αλλά μια όψη πάνω στο λεπτομερές σύνολο δεδομένων. Ως συνήθως, μια όψη (και άρα, ένας κύβος) μπορεί είτε να είναι αποθηκευμένη ή όχι. Κατά συνέπεια, μπορούμε να αντιμετωπίσουμε με δύο τρόπους τον κύβο: απλά ως σύνολο δεδομένων, ή σαν μια επερώτηση. Στο μοντέλο μας κρατάμε αυτό το δυϊσμό τυπικά: ένας κύβος δεν είναι απλά ένα σύνολο πλειάδων, αλλά έχει και ένα ορισμό, μέσω μιας επερώτησης που ανάγει τον υπολογισμό του κύβου σε μια σειρά πράξεις πάνω στο αποθηκευμένο λεπτομερές σύνολο δεδομένων.

Τυπικά ένας κύβος c ορισμένος πάνω στο σχήμα $[L_1,\dots,L_n,M_1,\dots,M_m]$, είναι μια έκφραση της μορφής $c=(DS^0,\varphi,[L_1,\dots,L_n,M_1,\dots,M_m],[agg_1(M_1^0),\dots,agg_m(M_m^0)])$, όπου το DS^0 είναι ένα λεπτομερές σύνολο δεδομένων πάνω στο σχήμα $S=[L_1^0,\dots,L_n^0,M_1^0,\dots,M_k^0]$, $m\leq k$, φ είναι μια συνθήκη επιλογής πάνω σε λεπτομερή επίπεδα, $M_1^0,\dots,M_m^0$ είναι λεπτομερή μέτρα, $M_1,\dots,M_m$ είναι συναθροισμένα μέτρα, L_i^0 και L_i είναι επίπεδα τέτοια ώστε $L_i^0 \prec L_i$, $1\leq i\leq n$ και agg_i, $1\leq i\leq m$ είναι αθροιστικές συναρτήσεις από το σύνολο `{sum,min,max,count}`.

Αν θέλουμε να περιγράψουμε διαισθητικά τον υπολογισμό ενός κύβου, ξεκινάμε με την εφαρμογή της συνθήκης επιλογής στα λεπτομερή δεδομένα. Στη συνέχεια, αντικαθιστούμε τις τιμές των επιπέδων των πλειάδων του αποτελέσματος με τις αντίστοιχες τιμές των προγόνων τους (στα επίπεδα του σχήματος του κύβου) και συναθροίζουμε τις πλειάδες, παράγοντας μία μόνο τιμή για κάθε μέτρο, μέσω της αντίστοιχης αθροιστικής συνάρτησης. Σημειώστε ότι ένα σύνολο δεδομένων μπορεί να εκφραστεί σαν κύβος, χρησιμοποιώντας την εκφυλισμένη συνθήκη επιλογής `true`. Για παράδειγμα, ο κύβος του λεπτομερούς συνόλου δεδομένων DS^0 του σχήματος 2 εκφράζεται ως: c^0`=(`DS^0`,true,[day,day,item,salesman,city,sales],sum(sales))`.

Η προσέγγιση αυτή εισάγει ένα ισχυρό μηχανισμό έκφρασης, ικανό να μοντελοποιήσει απ' ευθείας πράξεις όπως η αναλυτική κάθοδος και η αλλαγή της αθροιστικής συνάρτησης, με απώτερο στόχο τη μοντελοποίηση σειρών πράξεων, όπως συμβαίνει στα OLAP συστήματα. Εξ' όσων γνωρίζουμε, δεν υπάρχει άλλο μοντέλο που να το κάνει αυτό άμεσα. Η αναγωγή του ορισμού του κύβου σε μια κανονικοποιημένη μορφή φαίνεται να είναι η μόνη εναλλακτική λύση που προσφέρει αυτή τη δυνατότητα.

Τυπικά, το μοντέλο αποτελείται από τα παρακάτω στοιχεία:

- Κάθε *διάσταση* (*dimension*) D είναι ένα δίκτυο (lattice) $(\mathbf{L},\prec)$ τέτοιο ώστε: $\mathbf{L}=(L_1,\ldots,L_n,ALL)$ είναι ένα πεπερασμένο υποσύνολο από *επίπεδα* (*levels*) και $\prec$ είναι μια σχέση μερικής διάταξης ορισμένη μεταξύ των επιπέδων του $\mathbf{L}$, έτσι ώστε $L_1\prec L_i\prec ALL$ για κάθε $1\leq i\leq n$.
- Μια οικογένεια συναρτήσεων anc_{L1}^{L2} που ικανοποιεί τις παρακάτω συνθήκες (επεκτείνοντας την πρόταση του [CaTo97]):
    1. Για κάθε ζεύγος επιπέδων L_1 και L_2 έτσι ώστε $L_1\prec L_2$, η συνάρτηση anc_{L1}^{L2} απεικονίζει κάθε στοιχείο του $dom(L_1)$ σε ένα στοιχείο του $dom(L_2)$.
    2. Δοθέντων των επιπέδων L_1, L_2 και L_3 έτσι ώστε $L_1\prec L_2\prec L_3$, η συνάρτηση $anc_{L_1}^{L_3}$ ισοδυναμεί με τη σύνθεση $anc_{L1}^{L2}\circ anc_{L_2}^{L_3}$.
    3. Για κάθε ζεύγος επιπέδων L_1 και L_2 έτσι ώστε $L_1\prec L_2$, η συνάρτηση anc_{L1}^{L2} είναι μονότονη, ήτοι, $\forall x,y\in dom(L_1),L_1\prec L_2: x<y \Rightarrow anc_{L1}^{L2}(x)\leq anc_{L1}^{L2}(y)$.
    4. Για κάθε ζεύγος επιπέδων L_1 και L_2 η συνάρτηση anc_{L1}^{L2} καθορίζει ένα σύνολο από πεπερασμένες κλάσεις ισοδυναμίες X_i έτσι ώστε: $\forall x,y\in dom(L_1),L_1\prec L_2$: $anc_{L1}^{L2}(x)=anc_{L1}^{L2}(y)\Rightarrow x,y$ ανήκουν στο ίδιο X_i.
    5. Η σχέση $desc_{L1}^{L2}$ είναι η αντίστροφη της συνάρτησης anc_{L1}^{L2} -τουτέστιν, $desc_{L1}^{L2}(l)=\{x\in dom(L):anc_{L1}^{L2}(x)=l\}$.
- Κάθε *σύνολο δεδομένων* (*data set*) DS ορισμένο πάνω σε ένα σχήμα $S=[L_1,\ldots,L_n,M_1,\ldots,M_m]$ είναι ένα πεπερασμένο σύνολο πλειάδων ορισμένων πάνω στο S έτσι ώστε το $[L_1,\ldots,L_n]$ να είναι πρωτεύον κλειδί (με τη συνήθη έννοια του όρου).
- Κάθε *συνθήκη επιλογής* (*selection condition*) φ είναι μια έκφραση σε διαζευκτική κανονική μορφή. Ένα *άτομο* (*atom*) της συνθήκης επιλογής είναι της μορφής $true$, $false$ ή $x\ \theta\ y$, όπου θ είναι ένας τελεστής από το σύνολο (>, <, =, ≥, ≤, ≠) και κάθε ένα εκ των x και y μπορεί να είναι ένα από τα παρακάτω: (α) ένα επίπεδο L, (β) μια τιμή l, (γ) μια έκφραση της μορφής $anc_{L1}^{L2}(L_1)$ όπου $L_1\prec L_2$ και (δ) μια έκφραση της μορφής $anc_{L1}^{L2}(l)$ όπου $L_1\prec L_2$ και $l\in dom(L_1)$. Το *λεπτομερές ισοδύναμο της* φ (*detailed equivalent of* φ), συμβολιζόμενο με φ^0, είναι μια συνθήκη επιλογής που προκύπτει από την εξής διαδικασία: κάθε στιγμιότυπο του ονόματος ενός επιπέδου L στη φ, αντικαθίσταται από την ισοδύναμη έκφραση $anc_{L^0}^{L}(L^0)$, όπου το L^0 είναι το λεπτομερές επίπεδο της διάστασης στην οποία ανήκει το L.
- Κάθε κύβος (*cube*) c ορισμένος πάνω στο σχήμα $[L_1,\ldots,L_n,M_1,\ldots,M_m]$, είναι μια έκφραση της μορφής: $c=(DS^0,\varphi,[L_1,\ldots,L_n,M_1,\ldots,M_m],[agg_1(M_1^0),\ldots,agg_m(M_m^0)])$, όπου το DS^0 είναι ένα λεπτομερές σύνολο δεδομένων ορισμένο πάνω στο σχήμα $S=[L_1^0,\ldots,L_n^0,M_1^0,\ldots,M_k^0]$, $m\leq k$, φ είναι μια λεπτομερής συνθήκη επιλογής, $M_1^0,\ldots,M_m^0$ είναι λεπτομερή μέτρα, $M_1,\ldots,M_m$ είναι συναθροισμένα μέτρα, L_i^0 και L_i είναι επίπεδα τέτοια ώστε $L_i^0\prec L_i$, $1\leq i\leq n$ και agg_i, $1\leq i\leq m$ είναι

αθροιστικές συναρτήσεις από το σύνολο $\{sum, min, max, count\}$. Η έκφραση που χαρακτηρίζει τον κύβο έχει την παρακάτω σημασιολογία:

$$c=\{x\in dom(L_1)\times\ldots\times dom(L_n)\times dom(M_1)\times\ldots\times dom(M_m) \mid \exists y\in \varphi(DS^0), x[L_i]=anc_{L_i^0}^{L_i}(y[L_i^0]), 1\le i\le n, x[M_j]=agg_j(\{q \mid \exists z\in \varphi(DS^0), x[L_i]=anc_{L_i^0}^{L_i}(z[L_i^0]), 1\le i\le n, q=z[M_j^0]\}), 1\le j\le m\}.$$

Η *Άλγεβρα Κύβων* (*Cube Algebra* -CA) αποτελείται από τρεις λειτουργίες:

1. *Πλοήγηση* (*Navigate*): Έστω $S=[L_1, \ldots, L_n, M_1, \ldots, M_m]$ ένα σχήμα και $agg_1, \ldots, agg_m$ αθροιστικές συναρτήσεις. Εάν L_i^a και L_i ανήκουν στην ίδια διάσταση D_i και $agg_i\in\{sum, min, max, count\}$ τότε η πλοήγηση ορίζεται ως ακολούθως:
   - $nav(c^a, S, agg_1, \ldots, agg_m) = (DS^0, \varphi^a, S, [agg_1(M_1^0), \ldots, agg_m(M_m^0)])$.
2. *Επιλογή* (*Selection*): Έστω φ μια συνθήκη επιλογής εφαρμόσιμη στο c^a. Τότε, ορίζουμε την πράξη επιλογής ως:

   $$\sigma_\varphi(c^a) = (DS^0, \varphi^a \wedge \varphi^0, [L_1^a, \ldots, L_n^a, M_1^a, \ldots, M_m^a], [agg_1(M_1^0), \ldots, agg_m(M_m^0)])$$

   όπου το φ^0 είναι το λεπτομερές ισοδύναμο της συνθήκης επιλογής φ.
3. *Αποκόλληση Μέτρου* (*Split measure*): Έστω M ένα μέτρο του σχήματος του κύβου c. Χωρίς βλάβη της γενικότητας, ας υποθέσουμε ότι το M είναι το M_m. Τότε, η αποκόλληση μέτρου ορίζεται ως εξής:

   $$\pi_{M_m}(c^a) = (DS^0, \varphi^a, [L_1^a, \ldots, L_n^a, M_1^a, \ldots, M_{m-1}^a], [agg_1(M_1^0), \ldots, agg_m(M_{m-1}^0)]).$$

Παράδειγμα 3.1. Για να υποστηρίξουμε τη συζήτηση, προσαρμόζουμε το παράδειγμα του [Micr98] σε μια διεθνή εκδοτική εταιρεία με ταξιδεύοντες πωλητές που πουλάνε βιβλία και CD σε βιβλιοπωλεία σε όλο τον κόσμο. Η βάση δεδομένων (σχήματα 2-6) αποθηκεύει πληροφορία για τις πωλήσεις που ένας πωλητής πέτυχε σε μια πόλη, μια δεδομένη χρονική στιγμή. Οι διαστάσεις του παραδείγματος είναι Person (σχήμα 3), Location (σχήμα 4), Product (σχήμα 5) και Date (σχήμα 6). Το μέτρο Sales είναι συναρτησιακά εξαρτημένο από τις διαστάσεις Date, Product, Person και Location.

Η οργάνωση της πληροφορίας σε διαφορετικά επίπεδα συνάθροισης (διαστάσεις, δηλαδή) είναι απαραίτητη για τον απλό λόγο ότι οι χρήστες είναι μάλλον απίθανο να κάνουν ερωτήσεις απ' ευθείας στα λεπτομερή δεδομένα. Αντίθετα, ενδιαφέρονται περισσότερο για την συναθροισμένη πληροφορία που τους δίνει μια γενικότερη εικόνα της κατάστασης, και ζητούν την περαιτέρω ανάλυσή της μόνο σε ειδικές περιπτώσεις (π.χ. στα καταστήματα εκείνων των πόλεων που έκαναν συνολικά τις υψηλότερες ή χαμηλότερες πωλήσεις).

Στη συνέχεια παρουσιάζουμε τρεις επερωτήσεις και την αντίστοιχη αλγεβρική αναπαράστασή τους. Οι επερωτήσεις αυτές θα μπορούσαν κάλλιστα να αποτελούν μια σειρά πράξεων σε μια OLAP σύνοδο. Τα αποτελέσματα των επερωτήσεων φαίνονται στα σχήματα 7, 8 και 9, αντίστοιχα.

Επερώτηση 1. Βρες τις μέγιστες πωλήσεις ανά μήνα, κατηγορία προϊόντος, πωλητή και χώρα.

$$c^1=nav(DS^0, [Month, Category, Salesman, Country, Max\_val], max(sales)) = (DS^0, true, [Month, Category, Salesman, Country, Max\_val], max(sales)).$$

Επερώτηση 2. Βρες τις μέγιστες πωλήσεις εκτός Αμερικανικής ηπείρου ανά μήνα, κατηγορία προϊόντος, πωλητή και χώρα.

$$c^2=\sigma_{\text{'}anc_{country}^{continent}(country)\neq\text{'America'}}(c^1)=(DS^0,anc_{city}^{continent}(City)\neq\text{'America'},$$

$$[Month,Category,Salesman,Country,Max\_val],max(sales)).$$

Επερώτηση 3. Βρες το άθροισμα των πωλήσεων εκτός Αμερικανικής ηπείρου ανά μήνα, τύπο προϊόντος και χώρα.

$$c^3=nav(c^2,[Month,Type,ALL,Country,Sum\_val],sum(Sales))=$$

$$(DS^0,anc_{city}^{continent}(City)\neq\text{'America'},[Month,Type,ALL,Country,Sum\_val],$$

$$sum(sales)).$$

Στη διάρκεια αυτής της συνόδου, ο χρήστης έκανε τα εξής:

1. μια συναθροιστική άνοδος από το λεπτομερές σύνολο δεδομένων,
2. μια επιλογή,
3. ένας τεμαχισμός (της διάστασης `Person`) συνδυασμένος με μια αναλυτική κάθοδο (από το επίπεδο `Category` στο επίπεδο `Type`) και μια αλλαγή αθροιστικής συνάρτησης (από `max` σε `sum`).

Στην πρώτη λειτουργία βλέπουμε ότι η σημασιολογία της πλοήγησης μας επιτρέπει να χρησιμοποιήσουμε ένα οποιοδήποτε όνομα (π.χ., `Max_val`) για το μέτρο που υπολογίζει τη μέγιστη τιμή ανά ομάδα συνάθροισης.

Στη δεύτερη λειτουργία, παρατηρήστε ότι η έκφραση $anc_{country}^{continent}(Country)$ που είναι απευθείας εφαρμόσιμη στο σχήμα (και τα δεδομένα) του κύβου c^1 μετασχηματίζεται στην ισοδύναμή της $anc_{city}^{continent}(City)$, η οποία είναι εφαρμόσιμη στο λεπτομερές σύνολο δεδομένων DS^0, μέσω του ορισμού της έννοιας της λεπτομερούς συνθήκης επιλογής.

Το μοντέλο αυτό υπογραμμίζει την ιδέα *ότι ο κύβος μπορεί να αντιμετωπισθεί ταυτόχρονα και σαν επερώτηση και σαν σύνολο πλειάδων*. Πιστεύουμε ότι αυτή η πλευρά του OLAP έχει παραμεληθεί στις προηγούμενες προσεγγίσεις. Στο παράδειγμα που μόλις αναφέραμε είναι εμφανές ότι ήταν το γεγονός ότι κρατήσαμε την ιστορία των επιλογών, που μας επέτρεψε να κάνουμε αναλυτική κάθοδο και να αλλάξουμε αθροιστική συνάρτηση. Η εναλλακτική αντιμετώπιση για την αναλυτική κάθοδο θα ήταν κάποια πράξη σύνδεσης του c^2 με το DS^0. Το ίδιο ισχύει επίσης και για την αλλαγή της αθροιστικής συνάρτησης. Χρησιμοποιώντας την ιστορία των επιλογών μπορούμε (α) να αποφύγουμε την εκτέλεση μιας δαπανηρής πράξης σύνδεσης και (β) να βελτιστοποιήσουμε πιθανώς περισσότερο την εκτέλεση μιας λειτουργίας με τη χρήση ήδη υπολογισμένων κύβων. Τη δεύτερη αυτή πιθανότητα θα τη διερευνήσουμε στην ενότητα 4.

Όπως έχουμε ήδη τονίσει αυτό είναι ένα *λογικό* μοντέλο κύβων. Δεν υποστηρίζουμε ότι ο *φυσικός* υπολογισμός των αποτελεσμάτων θα έπρεπε να εκτελείται από το λεπτομερή κύβο. Ενώ η αναλυτική κάθοδος και η αλλαγή αθροιστικής συνάρτησης δεν ξεφεύγουν από αυτό τον κανόνα, οι επιλογές και οι συναθροιστικές αναβάσεις μπορούν αντίθετα να εκτελεστούν τοπικά. Στην περίπτωση επιλογής αρκεί να περάσουν τα δεδομένα από το φίλτρο της συνθήκης επιλογής. Στην περίπτωση της συναθροιστικής ανόδου σε υψηλότερα επίπεδα λεπτομέρειας αρκεί η κατάλληλη ομαδοποίηση των πλειάδων και η εφαρμογή της κατάλληλης αθροιστικής συνάρτησης. Οι απλές αυτές διαπιστώσεις γενικεύονται στην ενότητα 4 με μια ισχυρότερη προσέγγιση ικανή να εντοπίσει εάν ένας κύβος μπορεί να υπολογισθεί από τα δεδομένα ενός άλλου κύβου απλά συγκρίνοντας τους ορισμούς τους.

Month	Category	Salesman	Country	Max val
Feb 97	Books	Netz	France	7
Feb 97	Books	Netz	USA	5
May 97	Music	Netz	USA	20
Sept 97	Books	Netz	Japan	50
July 97	Music	Venk	Greece	34

Σχήμα 7: Κύβος c^1 - Πλοήγηση

Month	Category	Salesman	Country	Max val
Feb 97	Books	Netz	France	7
Sept 97	Books	Netz	Japan	50
July 97	Music	Venk	Greece	34

Σχήμα 8: Κύβος c^2 – Επιλογή

Day	Type	ALL	Country	Sum_val
Feb 97	Philosophy	All	France	7
Sep 97	Philosophy	All	Japan	50
Sep 97	Literature	All	Japan	30
Jul 97	Heavy Metal	All	Greece	47

Σχήμα 9: Κύβος c^3 – Σύνθετη σειρά πράξεων

Θεώρημα 3.1. Η Άλγεβρα Κύβων CA είναι *συνεπής* (το αποτέλεσμα, δηλαδή, όλων των λειτουργιών είναι πάντα κύβος) και *πλήρης* (κάθε κύβος, δηλαδή, μπορεί να υπολογιστεί από ένα συνδυασμό των πράξεων της άλγεβρας CA). ■

4. ΤΟ ΠΡΟΒΛΗΜΑ ΤΗΣ ΧΡΗΣΙΜΟΤΗΤΑΣ ΤΩΝ ΚΥΒΩΝ

Περιγραφή του προβλήματος. Υπάρχουν περιπτώσεις όπου υπάρχει η ανάγκη να αποφασίσουμε αν μια όψη μπορεί να χρησιμοποιηθεί για να υπολογίσουμε μια άλλη όψη. Για να δώσουμε δύο γνωστά παραδείγματα, (α) οι χρήστες OLAP εργαλείων κάνουν αλληλεπιδραστικές πλοηγήσεις στα δεδομένα, με αλλαγές στη λεπτομέρεια της παρουσίασής τους και (β) ο σχεδιαστής της συγκεντρωτικής αποθήκης δεδομένων έχει να επιλέξει, ανάμεσα σε πολλές υποψήφιες, ποιες όψεις θα υλοποιήσει. Στην πρώτη περίπτωση, ο χρήστης επιλέγει κάποια δεδομένα και κάνει κάποια πράξη πάνω τους. Το αποτέλεσμα της νέας λειτουργίας μπορεί φυσικά να υπολογισθεί από τα λεπτομερή δεδομένα (όπως δείξαμε στην προηγούμενη ενότητα), αλλά είναι δυνατόν και να υπολογισθεί από κύβους προϋπολογισμένους ή προσωρινώς αποθηκευμένους στη λανθάνουσα μνήμη. Στη δεύτερη περίπτωση, ο σχεδιαστής της συγκεντρωτικής αποθήκης δεδομένων χρειάζεται κάποιους αλγόριθμους για να αποφασίσει αν θα αποθηκεύσει επιπλέον όψεις (πιθανά επικαλυπτόμενες) ώστε οι ερωτήσεις των χρηστών να απαντώνται πιο γρήγορα. Ορισμένες φορές, ο πλεονασμός όψεων μπορεί να επιταχύνει και τη διαδικασία ανανέωσης [ThLS99, LSTV99, Gupt97]. Τμήμα του αλγορίθμου σχεδίασης είναι και μια μέθοδος που αποφασίζει αν μια όψη μπορεί να χρησιμοποιηθεί για να υπολογίσει μια άλλη όψη (ή εν γένει μια ερώτηση). Γενικεύοντας τα παραπάνω, μπορεί κανείς να πει ότι το πρόβλημα έγκειται στην απόκριση του κατά πόσον μπορεί ένας κύβος να υπολογισθεί από ένα ενδιάμεσο επίπεδο συνάθροισης αντί από το λεπτομερές σύνολο δεδομένων.

Τυπικά, έστω DS^0 ένα λεπτομερές σύνολο δεδομένων. Έστω επίσης c^{old} και c^{new} δύο κύβοι ορισμένοι πάνω στο DS^0. Εξ' ορισμού, οι κύβοι c^{old} και c^{new} μπορούν να υπολογισθούν από το DS^0. Το *πρόβλημα της χρησιμότητας κύβων* (*cube usability problem*) έγκειται στην απόφαση, εάν οι πλειάδες του c^{old} μπορούν να χρησιμοποιηθούν για τον υπολογισμό του c^{new}. Είναι σαφές ότι πρόβλημα είναι υποπερίπτωση του προβλήματος της *υπαγωγής όψεων* (*view subsumption*), που έχει ήδη εξερευνηθεί αρκετά στο χώρο των σχεσιακών βάσεων δεδομένων [Ullm97].

Προβλήματα των υπαρχόντων προσεγγίσεων. Έχει προϋπάρξει σημαντική δουλειά στο παρελθόν για την επίλυση του προβλήματος της υπαγωγής όψεων και της επανεγγραφής ερωτήσεων όταν είναι παρούσες αποθηκευμένες όψεις [NuSS98, CoNS99, DJLS96, CKPS95, GuHQ95, LMSS95, ChSh96, YaLa85]. Παρ' όλα αυτά, οι προηγούμενες προσεγγίσεις είναι προσαρμοσμένες στο σχεσιακό μοντέλο και αδυνατούν να εκμεταλλευθούν τα ιδιαίτερα χαρακτηριστικά της πολυδιάστατης μοντελοποίησης κύβων. Θα δώσουμε δύο παραδείγματα για να αναδείξουμε τα προβλήματα αυτά.

Παράδειγμα 4.1. Διαισθητικά, θα περίμενε κανείς ότι για να λύσουμε το πρόβλημα της χρησιμότητας κύβων ο νέος κύβος c^{new} θα έπρεπε:

1. Να είναι ορισμένος στις ίδιες διαστάσεις με τον c^{old} και σε υψηλότερο ή ίσο επίπεδο.
2. Να είναι ορισμένος στο ίδιο μέτρο του DS^0 και επιπλέον η αθροιστικές συναρτήσεις agg^{new} και agg^{old} να είναι ίδιες.
3. Να έχει μια συνθήκη επιλογής πιο περιορισμένη από το c^{old}, τουτέστιν η φ^{new} να εγκλείεται στην φ^{old} με το συνήθη σχεσιακό τρόπο.

Ο έλεγχος των συνθηκών 1 και 2 είναι εύκολος, φυσικά. Για να κάνουμε όμως τη σύγκριση της συνθήκης 3, πρέπει να μετασχηματίσουμε της συνθήκες επιλογής των δύο κύβων ώστε να τις αντιμετωπίσουμε σα συζευκτικές επερωτήσεις [Ullm89]. Θα δείξουμε ότι οι υπάρχουσες σχεσιακές τεχνικές δεν επαρκούν για λύσουν το πρόβλημα.

Cube c^1

Month	Sales
Feb	5
May	20
Sep	80

Cube c^2

Month	Sales
Feb	**12**
May	20
Sep	80

Cube c^3

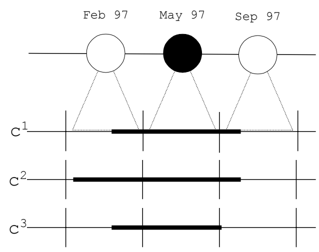
Month	Sales
Feb	5
May	20

φ_1=18-Feb-97≤day≤3-Sep-97 ∧Salesman=Netz

φ_2=6-Feb-97≤day≤3-Sep-97∧Salesman=Netz

φ_3=18-Feb-97≤day≤31-May-97∧Salesman=Netz

(α)



(β)

Σχήμα 10: Προβλήματα χρησιμότητας κύβων

Έστω το λεπτομερές σύνολο δεδομένων DS^0 του σχήματος 2. Έστω c^i, $1\le i\le 3$ κύβοι ορισμένοι ως c^i=[DS^0, φ_i, [Month,ALL,ALL,ALL,ALL,ALL,ALL,Sales],sum(sales)]. Το σχήμα 10a περιγράφει το επίπεδο Month, το μέτρο Sales και τις συνθήκες επιλογής για κάθε κύβο. Το πρόβλημα είναι αν ο (νέος) κύβος c^3 μπορεί να υπολογισθεί από τις πλειάδες των (ήδη υπαρχόντων) κύβων c^1 και c^2. Αφού οι συνθήκες 1, 2 και 3 ισχύουν, θα μπορούσε κανείς να ισχυρισθεί ότι αυτό είναι εφικτό. Όμως, όπως φαίνεται και στο σχήμα 10α, μόνο ο c^1 μπορεί να χρησιμοποιηθεί για τον υπολογισμό του c^3. Η διαισθητική επεξήγηση του προβλήματος φαίνεται στο σχήμα 10β. Υπάρχουν τρεις οριζόντιοι άξονες ορισμένοι στο επίπεδο day για κάθε έναν από τους κύβους c^1, c^2 και c^3. Οι έντονες γραμμές δείχνουν το σύνολο των ημερών που συμμετέχουν στον υπολογισμό του αντίστοιχου κύβου. Ο κύβος c^3 είναι ορισμένος στο επίπεδο `month` και κατά συνέπεια διαχωρίζουμε τους τρεις άξονες σε σχέση με τη συνάρτηση anc_{day}^{month}. Όπως μπορούμε να δούμε έχουμε τρία τμήματα διαχωρισμού: `Feb'97`, `May'97` και `Sep'97`. Ο κύβος c^3 μπορεί να υπολογισθεί από τον c^1 επειδή για όλα τα τμήματα διαχωρισμού του c^3 (i.e., `Feb'97`, `May'97`), οι κύβοι c^1 και c^3 καλύπτουν ακριβώς τις ίδιες μέρες. Αυτό δεν ισχύει για τους κύβους c^1 και c^2. □

Παράδειγμα 4.2. Έστω η περίπτωση που ο κύβος c^1 έχει μια συνθήκη επιλογής φ_1=`arr.year<dep.year` (όπου `arr` είναι συντομογραφία για τη διάσταση `arrival date` και `dep` για τη διάσταση `departure date`). Έστω ακόμα ότι ο κύβος c^2 είναι ορισμένος στο επίπεδο μήνα και έχει μια συνθήκη επιλογής φ_2=`arr.month<dep.month`. Μπορούμε να δούμε ότι ο κύβος c^1 μπορεί να υπολογισθεί από τον c^2. Αυτό σημαίνει ότι αν ο c^2 είναι αποθηκευμένος, μπορούμε να χρησιμοποιήσουμε τις πλειάδες του για τον υπολογισμό του c^1. Μπορούμε να κάνουμε αυτού του είδους το συλλογισμό εκμεταλλευόμενοι τη σχέση μηνών και χρόνων, η οποία εκφράζεται από την ιεραρχία της χρονικής διάστασης μέσω των συναρτήσεων `anc`. Σε ότι γνωρίζουμε, δεν υπάρχει άλλη προσπάθεια στο χώρο της υπαγωγής όψεων που να χρησιμοποιεί αντίστοιχη πληροφορία.

Στο σχήμα 11, παρουσιάζουμε μια γραφική απεικόνιση του προβλήματος. Οι πλειάδες του λεπτομερούς κύβου αναπαριστώνται σαν κελιά στο διδιάστατο χώρο. Ο οριζόντιος άξονας αναπαριστά τη διάσταση `departure date` και ο κάθετος άξονας τη διάσταση `arrival date` (χάριν ευκολίας αγνοούμε τις υπόλοιπες διαστάσεις του παραδείγματος). Όπως μπορεί να παρατηρήσει κανείς, οι πλειάδες του λεπτομερούς συνόλου δεδομένων που πληρούν την συνθήκη `arr.month<dep.month` είναι γνήσιο υποσύνολο των πλειάδων που πληρούν τη συνθήκη `arr.month<dep.month`. ■

01Jan96								
...								
01Feb96								
...								
01Jan97								
...								
01Feb97								
arr / dep	01Jan96	...	01Feb96	...	01Jan97	...	01Feb97

Legend

	Denotes the tuples of the detailed data set fulfilling only the condition `arr.month<dep.month`
	Denotes the tuples of the detailed data set fulfilling only the condition `arr.year<dep.year`
	Denotes the tuples of the detailed data set fulfilling both the conditions

Σχήμα 11. Γραφική αναπαράσταση της αξιολόγησης των δεδομένων για το πρόβλημα της χρησιμότητας κύβων

Συνεισφορά. Στην ενότητα αυτή θα δείξουμε ότι το πρόβλημα χρησιμότητας κύβων ανάγεται σε απλούς ελέγχους και πράξεις. Διαφορετικοί έλεγχοι χρησιμοποιούνται για διαφορετικές κλάσεις ερωτήσεων. Θα ερευνήσουμε τις συνθήκες επιλογής δύο κατηγοριών: (α) συνθήκες επιλογής με άτομα που περιέχουν τιμές (της μορφής, δηλαδή, $L\theta l$, $L\theta anc_{L1}^{L2}(l)$, κλπ.) και (β) συνθήκες επιλογής με άτομα που περιέχουν μόνο επίπεδα (της μορφής, δηλαδή, $L_1\theta L_2$, $L\theta anc_{L1}^{L2}(L_1)$, κλπ). Θα εξετάσουμε ζητήματα βελτιστοποίησης για τις πρώτες στην υποενότητα 4.1 και για τις δεύτερες στην υποενότητα 4.2. Στην υποενότητα 4.3 θα δώσουμε ένα θεώρημα με ικανά κριτήρια και τον αντίστοιχο αλγόριθμο επανεγγραφής και για τις δύο περιπτώσεις αυτές.

Στο υπόλοιπο κεφάλαιο, για λόγους απλότητας θα θεωρήσουμε ότι οι κύβοι έχουν μόνο ένα μέτρο. Όλα τα αποτελέσματα επεκτείνονται εύκολα σε κύβους με περισσότερα μέτρα [NuSS98]. Έστω $c^{new}=(DS^0, \varphi^{new}, [\mathbf{L}^{new}, M^{new}], agg^{new}(M))$ ο νέος κύβος και $c^{old}=(DS^0, \varphi^{old}, [\mathbf{L}^{old}, M^{old}],$

$\mathtt{agg}^{old}\mathtt{(M))}$ ο υποψήφιος κύβος, όπου $\mathbf{L}^{new}$ και $\mathbf{L}^{old}$ είναι σύνολα επιπέδων από τα σύνολα διαστάσεων $\mathbf{D}^{new}$ και $\mathbf{D}^{old}$ αντίστοιχα, $\mathtt{M}^{new}$ και $\mathtt{M}^{old}$ είναι μέτρα, και τέλος, $\mathtt{agg}^{new}$ και $\mathtt{agg}^{old}$ είναι αθροιστικές συναρτήσεις.

4.1 Ισοδύναμοι μετασχηματισμοί για άτομα που εμπλέκουν τιμές

Έστω δύο επίπεδα $\mathtt{L}^{old}$ και $\mathtt{L}^{new}$, τέτοια ώστε $\mathtt{L}^{old} \prec \mathtt{L}^{new}$. Η συνάρτηση $\mathtt{anc}_{L^{old}}^{L^{new}}$ ορίζει ένα διαχωρισμό των τιμών του $\mathtt{L}^{old}$ σε σχέση με τις τιμές του $\mathtt{L}^{new}$ (π.χ. ο διαχωρισμός με βάση το `year` στο επίπεδο `month`). Έστω ακόμα δύο άτομα $\mathtt{a}_1$ και $\mathtt{a}_2$ ορισμένα στο $\mathtt{L}^{old}$, όπως στην περίπτωση του σχήματος 10. Για να κάνουμε μια συνάθροιση στο $\mathtt{L}^{new}$, τα δύο άτομα πρέπει να έχουν τα ίδια εύρη τιμών για κάθε τμήμα διαχωρισμού που ορίζει το $\mathtt{L}^{new}$ πάνω στο $\mathtt{L}^{old}$. Γενικεύοντας την παρατήρηση αυτή, στην περίπτωση που δύο συνθήκες επιλογής περιέχουν ένα μεγαλύτερο αριθμό ατόμων, πρέπει:

(1) Να μετασχηματίσουμε τις συνθήκες επιλογής σε συνεχή εύρη τιμών για κάθε διάσταση.

(2) Να ανάξουμε τα άτομα στο ίδιο επίπεδο χρησιμοποιώντας τους κατάλληλους μετασχηματισμούς (ώστε να μπορούν να συγκριθούν).

(3) Να ελέγξουμε αν η ευρύτερη συνθήκη επιλογής είναι κατάλληλα ορισμένη για τις οριακές συνθήκες της άλλης συνθήκης επιλογής

Ο παρακάτω βοηθητικός ορισμός εισάγει την έννοια του *διαστήματος διάστασης* (*dimension interval*), που είναι ένα συμπαγές εύρος τιμών πάνω στο πεδίο ορισμού ενός επιπέδου.

Ορισμός 4.1: Ένα *διάστημα διάστασης* (*dimension interval* -DI) είναι ένα από τα παρακάτω (α) `true`, (β) `false` και (γ) μια έκφραση της μορφής $\mathtt{l}_1 \le \mathtt{L} \le \mathtt{l}_2$, όπου το `L` είναι μια μεταβλητή που αναπαριστά ένα επίπεδο μιας διάστασης και $\mathtt{l}_1$ και $\mathtt{l}_2$ είναι τιμές. ■

Atom	Dimension Interval	Atom	Dimension Interval
True	true	$\mathtt{anc}_L^{L'}\mathtt{(L)<l}$	$-\infty<\mathtt{L}\le\mathtt{max(desc}_L^{L'}\mathtt{(prev(l)))}$
False	false	$\mathtt{l}\le\mathtt{anc}_L^{L'}\mathtt{(L)}$	$\mathtt{min(desc}_L^{L'}\mathtt{(l))}\le\mathtt{L}<+\infty$
$\mathtt{anc}_L^{L'}\mathtt{(L)=l}$	$\mathtt{min(desc}_L^{L'}\mathtt{(l))}\le\mathtt{L}\le\mathtt{max(desc}_L^{L'}\mathtt{(l))}$	$\mathtt{l<anc}_L^{L'}\mathtt{(L)}$	$\mathtt{min(desc}_L^{L'}\mathtt{(next(l)))}\le\mathtt{L}<+\infty$
$\mathtt{anc}_L^{L'}\mathtt{(L)}\le\mathtt{l}$	$-\infty<\mathtt{L}\le\mathtt{max(desc}_L^{L'}\mathtt{(l))}$		

Σχήμα 12: Μετασχηματίζοντας άτομα σε διαστήματα διάστασης

Το σχήμα 12 δείχνει πώς απλά άτομα μπορούν να μετασχηματιστούν σε διαστήματα διάστασης. Οι τιμές $-\infty$ και $+\infty$ έχουν την προφανή σημασιολογία. Οι συναρτήσεις `prev` και `next` επιστρέφουν την προηγούμενη και την επόμενη τιμή του `l` στο πεδίο ορισμού του `L` αντίστοιχα.

Εν γένει, για να αποφασίσουμε αν ένας κύβος $\mathtt{c}^{old}$ μπορεί να χρησιμοποιηθεί για τον υπολογισμό του $\mathtt{c}^{new}$, πρέπει να διαχωρίσουμε το λεπτομερές επίπεδο κάθε διάστασης με βάση το αντίστοιχο επίπεδο του $\mathtt{c}^{new}$. Αν για κάθε τμήμα διαχωρισμού του $\mathtt{c}^{new}$, υπάρχει ένα ταυτοτικό τμήμα διαχωρισμού του $\mathtt{c}^{old}$, τότε ο $\mathtt{c}^{old}$ μπορεί να χρησιμοποιηθεί για να υπολογίσουμε το $\mathtt{c}^{new}$. Τυποποιούμε τη σχέση αυτή, μέσω του Ορισμού 4.2.

Ορισμός 4.2. Εγκλεισμός $\mathbf{L}$: Έστω $\mathbf{D}$ ένα σύνολο διαστάσεων και φ^{old}, φ^{new} δύο συνθήκες επιλογής που περιλαμβάνουν επίπεδα μόνο από το $\mathbf{D}$. Έστω $\mathbf{L}$ ένα σύνολο επιπέδων που το καθένα ανήκει σε διαφορετική διάσταση του $\mathbf{D}$. Έστω ακόμα και δύο κύβοι $\mathtt{c}^{new}\mathtt{=(DS}^0\mathtt{,}\varphi^{new}\mathtt{,[}\mathbf{L}\mathtt{,M],agg(M))}$ και $\mathtt{c}^{old}\mathtt{=(DS}^0\mathtt{,}\varphi^{old}\mathtt{,[}\mathbf{L}\mathtt{,M],agg(M))}$, ορισμένοι πάνω σε ένα τυχαίο σύνολο δεδομένων $\mathtt{DS}^0$. Η συνθήκη επιλογής φ^{new} *εγκλείεται κατά* **L** (**L**-*contained*) στη φ^{old} (που συμβολίζεται $\varphi^{new} \subseteq_{\mathbf{L}} \varphi^{old}$) αν $c^{new} \subseteq c^{old}$ για οποιοδήποτε σύνολο δεδομένων DS^0. ■

Algorithm `Check_Atoms_Usability`.
Input: Two conjunctions of atoms **a** και **b** involving only values, και a set of levels **L′**.
Output: true if $\mathbf{a} \subseteq_{\mathbf{L'}} \mathbf{b}$, false otherwise.

1. Write all atoms of **a** και **b** as DI's using the transformations of Figure 5.12.
2. Group all DI's of **a** και **b** by dimension level και produce for every set a single DI having the most restrictive boundaries. Let **a′** και **b′** be the result, respectively.
3. <u>For</u> every DI a of **a′**
4.   <u>If</u> a is defined over dimension level $D_i.L^0$ that does not exist in any DI of **b′** <u>Then</u>
5.   Introduce DI $-\infty \le D_i.L^0 \le \infty$ to **b′**.
6. <u>EndFor</u>
7. `flag = false`
8. <u>For</u> every DI a of **a′**
9.   `flag = false`
10.   <u>For</u> every DI b of **b′**
11.   <u>For</u> every dimension level L′ of **L′** involved in b
12.   <u>Case</u> $A_s < B_s$ or $B_e < A_e$ or b=false
13.   `flag = true`
14.   <u>Case</u> $L \neq L'$ και $A_s \neq \mathrm{min}(desc_L^{L'}(anc_L^{L'}(A_s)))$ και $A_s \neq B_s$
15.   `flag = false`
16.   <u>Case</u> $L \neq L'$ και $A_e \neq \mathrm{max}(desc_L^{L'}(anc_L^{L'}(A_e)))$ και $A_e \neq B_e$
17.   `flag = false`
18.   <u>Default</u>
19.   `flag = true`
20.   <u>EndFor</u>
21.   <u>EndFor</u>
22.   <u>If</u> `flag = false` <u>Then</u>
23.   <u>Return</u> `false`
24. <u>EndFor</u>
25. <u>Return</u> `true`

Σχήμα 13: Αλγόριθμος `Check_Atoms_Usability`

Για να αντιμετωπίσουμε το πρόβλημα της χρησιμότητας κύβων μεταξύ κύβων διαφορετικών επιπέδων λεπτομέρειας, αρχίζουμε τον έλεγχο από τον εγκλεισμό των συνθηκών επιλογής τους. Η ανάλυσή μας προς το παρόν δεν καλύπτει την περίπτωση του ≠, αλλά αυτό θα αντιμετωπισθεί στην υποενότητα 4.3.
Ο Αλγόριθμος `Check_Atoms_Usability` του σχήματος 13 παίρνει ως είσοδο δύο συζεύξεις ατόμων, **a** και **b**, που αφορούν μόνο τιμές. Ο αλγόριθμος επιστρέφει `true` αν το **a** εγκλείει κατά **L** το **b** σε σχέση με το σύνολο επιπέδων **L′**, και `false` σε κάθε άλλη περίπτωση. Ο αλγόριθμος προχωρεί ως ακολούθως. Αρχικά, ο Αλγόριθμος `Check_Atoms_Usability` επανεγγράφει όλα τα άτομα των **a** και **b** σε διαστήματα διάστασης χρησιμοποιώντας τους μετασχηματισμούς του σχήματος 13 (Γραμμή 1). Έπειτα ομαδοποιεί όλα τα διαστήματα διάστασης των **a** και **b** ανά επίπεδο και παράγει για κάθε σύνολο, ένα μόνο διάστημα διάστασης με τα πλέον περιορισμένα άκρα. Το αποτέλεσμα αποθηκεύεται στα σύνολα διαστημάτων διάστασης **a′** και **b′** αντίστοιχα (Γραμμή 2). Οι Γραμμές 3-6 ελέγχουν εάν υπάρχει επίπεδο $D_i.L$ του διαστήματος διάστασης a που ανήκει στο **a′**, το οποίο δεν ανήκει σε κανένα διάστημα διάστασης του **b′**. Στην περίπτωση αυτή, ο αλγόριθμος εισάγει το διάστημα $-\infty \le D_i.L^0 \le +\infty$ στο **b′** (Γραμμή 5). Τέλος οι γραμμές 7-24 ελέγχουν αν για κάθε διάστημα διάστασης b του **b** υπάρχει ένα διάστημα διάστασης a στο **a** τέτοιο ώστε $b \subseteq_{L'} a$ για κάποιο επίπεδο $L' \in \mathbf{L'}$. Ειδικότερα, οι γραμμές 12-21 ελέγχουν κατά πόσον το διάστημα διάστασης a είναι

εγκλεισμένο κατά **L** σε σχέση με το διάστημα διάστασης b. Οι γραμμές 12-13 ελέγχουν αν το διάστημα διάστασης b είναι ευρύτερο του a. Ο αλγόριθμος επιστρέφει true αν τα οριακά τμήματα διαχωρισμού είναι ταυτοτικά ίσα (Γραμμές 14-17).

Στο παράδειγμα 4.1 αν χρησιμοποιήσουμε τον Αλγόριθμο Check_Atoms_Usability θα συνάγουμε ότι η φ_1 εγκλείει κατά **L** τη φ_3 (σε σχέση με το επίπεδο Month), ενώ η φ_2 όχι. Ακόμα, είναι ενδιαφέρον να παρατηρήσει κανείς ότι σε σχέση με το επίπεδο year, ούτε η φ_1 ούτε η φ_2 εγκλείουν κατά **L** τη φ_3.

4.2 Ισοδύναμοι μετασχηματισμοί για άτομα που εμπλέκουν μόνο επίπεδα

Ακολουθώντας το [Ullm89], υποθέτουμε την ύπαρξη δύο άπειρων, καθολικά ταξινομημένων πεδίων **L** και **L′** ισομορφικά στους ακέραιους. Έστω επίσης η συνάρτηση f η οποία είναι καθολική και μονότονη πάνω στο **L**, και η οποία απεικονίζει τις τιμές του **L** σε τιμές του **L′**. Η οικογένεια των συναρτήσεων anc πληροί αυτά τα κριτήρια.

Υποθέτουμε ότι μας δίνεται ένα σύνολο από ανισότητες της μορφής X<Y, X≤Y, X≠Y, f(X)<f(Y), f(X)≤f(Y), f(X)≠f(Y) και ισότητες της μορφής f(X)=f(Y). Δεν επιτρέπουμε ισότητες της μορφής X=Y. Αν ένας τέτοιος *υπο-στόχος* (*subgoal*) βρεθεί σε μία ερώτηση, αντικαθιστούμε κάθε εμφάνιση του X με Y. Επίσης αντικαθιστούμε κάθε ζεύγος ανισοτήτων f(X)≤f(Y) και f(Y)≤f(X), όπου X,Y είναι διακριτές μεταβλητές με f(X)=f(Y).

Θα χρησιμοποιήσουμε το παρακάτω σύνολο αξιωμάτων για αυτές τις ανισότητες:

A1	X≤X	A8	X≤Z, Z≤Y, X≤W, W≤Y και W≠Z συνάγουν X≠Y
A2	X<Y συνάγει X≤Y	A9	X≤Y συνάγει f(X)≤f(Y)
A3	X<Y συνάγει X≠Y	A10	f(X)<f(Y) συνάγει X<Y
A4	X≤Y και X≠Y συνάγουν X<Y	A11	f(X)≠f(Y) συνάγει X≠Y
A5	X≠Y συνάγει Y≠X	A12	f(X)≤f(Y) και f(Y)≤f(X) συνάγει f(X)=f(Y)
A6	X<Y και Y<Z συνάγουν X<Z	A13	f(X)=f(Y) και f(Y)≤f(Z) συνάγει f(X)≤f(Z)
A7	X≤Y και Y≤Z συνάγουν X≤Z	A14	f(X)=f(Y) και f(Y)≠f(Z) συνάγει f(X)≠f(Z)
		A15	f(X)=f(Y) συνάγει f(X)≤f(Y)

Σχήμα 14: Αξιώματα για έλεγχο εγκλεισμού κατά L.

Υποθέτουμε ότι τα μοντέλα μας είναι αναθέσεις ακεραίων σε μεταβλητές. Εκφράσεις της μορφής f(X) αντιμετωπίζονται επίσης σαν μεταβλητές. Για τις μεταβλητές της μορφής X επιβάλλουμε τα αξιώματα A1 ως A9 και για τις μεταβλητές της μορφής f(X) τα αξιώματα A1 ως A15.

Θεώρημα 4.1: Τα αξιώματα είναι συνεπή και πλήρη. ■

Για να ελέγξουμε εάν ένα σύνολο ανισοτήτων T συνάγεται από ένα άλλο σύνολο ανισοτήτων S υπολογίζουμε το κλείσιμο του S^+ εφαρμόζοντας τα αξιώματα A1-A15 μέχρι να μην παράξουν νέες ανισότητες. Τότε, ελέγχουμε εάν το T είναι υποσύνολο του S^+.

4.3 Ελέγχοντας τη χρησιμότητα (Usability) των κύβων

Στην υποενότητα αυτή θα συνδυάσουμε τα αποτελέσματα των υποενοτήτων 4.1 και 4.2 για να κατασκευάσουμε τρόπο ελέγχου για το πρόβλημα χρησιμότητας των κύβων. Μπορούμε να χρησιμοποιήσουμε λογικούς μετασχηματισμούς για να μετατρέψουμε οποιαδήποτε έκφραση σε μια ισοδύναμη έκφραση που αποτελείται από διαζεύξεις συζεύξεων που δεν συμπεριλαμβάνουν $\neq$ και $\neg$ [Ende72]. Το Θεώρημα 4.2 παρέχει ικανά κριτήρια για τη δυνατότητα χρήσης ενός κύβου c^{old} στον υπολογισμό ενός κύβου c^{new}. Ο Αλγόριθμος `Cube_Usability` περιγράφει τα συγκεκριμένα βήματα που απαιτούνται για τον υπολογισμό αυτό.

Θεώρημα 4.2: Έστω ένα λεπτομερές σύνολο δεδομένων $DS^0=[L_1^0,\ldots,L_n^0,M^0]$ και δύο κύβους $c^{old}=(DS^0,\varphi_{old},[L_1^{old},\ldots,L_n^{old},M_{old}],agg_{old}(M^0))$ και $c^{new}=(DS^0,\varphi_{new},[L_1^{new},\ldots,L_n^{new},M_{new}],agg_{new}(M^0))$. Εάν $agg_{old}=agg_{new}$, $L_i^{old}\prec L_i^{new}$, $1\leq i\leq n$, και μία από τις παρακάτω περιπτώσεις ισχύει για τα φ_{old} και φ_{new}:

- φ_{old} και φ_{new} περιέχουν συζεύξεις ατόμων μόνο της μορφής $L_i\theta L_j$, όλα τα επίπεδα L_i, L_j είναι υψηλότερα από τα αντίστοιχα επίπεδα του σχήματος του c^{old} (i.e. $L_{i,j}^{old}\prec L_{i,j}$) και τέλος, το φ_{old} ανήκει στο κλείσιμο του φ_{new}, ή,
- φ_{old} και φ_{new} αφορούν συζεύξεις ατόμων της μορφής $L\theta l$ και $\varphi_{new}\subseteq$**Error!**φ_{old},

τότε ο Αλγόριθμος `Cube_Usability` υπολογίζει σωστά c^{new} από τις πλειάδες του c^{old}. ■

Algorithm `Cube_Usability`.

Input: A detailed data set $DS^0=[L_1^0,\ldots,L_n^0,M^0]$ και two cubes $c^{old}=(DS^0,\varphi_{old},[L_1^{old},\ldots,L_n^{old},M_{old}],agg_{old}(M^0))$ και $c^{new}=(DS^0,\varphi_{new},[L_1^{new},\ldots,L_n^{new},M_{new}],agg_{new}(M^0))$ such that φ_{old} και φ_{new} involve either (a) conjunctions of atoms of the form $L\theta l$ or (b) conjunctions of atoms of the form $L\theta L'$ where L και L' are levels και l is a value.

Output: A rewriting that calculates cube c^{new} from the tuples of c^{old}.

<u>If</u> all atoms of φ_{old} και φ_{new} involve conjunctions of atoms of the form $L\theta l$ <u>Then</u>
    <u>For</u> every atom $a=anc_{L^0}^{L}(L^0)\,\theta\, l$ in φ_{new} (or equivalent to this form)
        <u>If</u> L^{old} is the respective level in the schema of c^{old} και $L^{old}\prec L$ <u>Then</u>
            Transform a to $anc_{L^{old}}^{L}(L^{old})\,\theta\, l$
        <u>EndIf</u>
        <u>ElseIf</u> L^{old} is the respective level in the schema of c^{old} και $L\prec L^{old}$ <u>Then</u>
            Transform a to $L^{old}\theta' anc_{L}^{L^{old}}(l)$ where $\theta'=\theta$ except for two cases:
                (a) $a=anc_{L^0}^{L}(L^0)<l$ και $l\neq min(desc_{L}^{L^{old}}(anc_{L}^{L^{old}}(l)))$ where $\theta'=\leq$,
                (b) $a=anc_{L^0}^{L}(L^0)>l$ και $l\neq max(desc_{L}^{L^{old}}(anc_{L}^{L^{old}}(l)))$ where $\theta'=\geq$
        <u>EndIf</u>
    <u>EndFor</u>
<u>EndIf</u>
<u>If</u> all atoms of φ_{old} και φ_{new} involve conjunctions of atoms of the form $a=anc_{L^0}^{L}(L^0)\,\theta\, anc_{L^{0\prime}}^{L'}(L^{0\prime})$ (or equivalent to this form), where both L και L' are higher than the respective levels of c^{old} <u>Then</u>
    <u>For</u> every atom $a=anc_{L^0}^{L}(L^0)\,\theta\, anc_{L^{0\prime}}^{L'}(L^{0\prime})$ in φ_{new}
        Transform a to $anc_{L^{old}}^{L}(L^{old})\,\theta\, anc_{L^{old\prime}}^{L'}(L^{old\prime})$
    <u>EndFor</u>
<u>EndIf</u>
Apply the transformed selection condition to c^{old} και derive a new data set DS^1.
Replace all the values of DS^1 with their ancestor values at the levels of c^{new}, resulting in a new data set DS^2.
Aggregate ("group by" in the relational semantics) on the tuples of DS^2, so that we produce c^{new}.

Σχήμα 15: Αλγόριθμος `Cube_Usability`

Το Θεώρημα 4.2 ελέγχει για πιθανή χρησιμότητα ζεύγη κύβων που ενέχουν συζευκτικές συνθήκες επιλογής που δεν περιέχουν $\neq$ και $\neg$. Κύβοι που περιλαμβάνουν διαζευκτικές συνθήκες επιλογής μπορούν να αντιμετωπισθούν με το γνωστό τρόπο [Ullm89].

Σημειώστε επίσης ότι το αντίστροφο του θεωρήματος ('και μόνο εάν') δεν ισχύει. Υποθέστε την περίπτωση μιας συγκεκριμένης διάστασης `D`, που περιλαμβάνει δύο επίπεδα `low` και `high`, όπου η σχέση desc είναι συνάρτηση (που σημαίνει ότι η συνάρτηση $\text{anc}_{\text{low}}^{\text{high}}$ έχει αντίστροφη και η αντιστοιχήσει από λεπτομερείς σε υψηλότερου επιπέδου τιμές είναι 1:1). Τότε, αν και η συνθήκη (2) του Θεωρήματος 4.2 παραβιάζεται, ένας κύβος στο επίπεδο high μπορεί να χρησιμοποιηθεί και για τον υπολογισμό ενός κύβου στο επίπεδο low. Ακόμα, είναι εύκολο να κατασκευάσουμε ένα παράδειγμα που να δείχνει ότι οι παραπάνω τεχνικές δεν εφαρμόζονται στην κλάση ερωτήσεων που περιέχουν άτομα και των δύο κατηγοριών του Θεωρήματος 4.2

Παράδειγμα 4.3. Έστω c^{new} και c^{old} δύο κύβοι πάνω στο DS^0 του σχήματος 16, ορισμένοι ως εξής:

c^{old}=(DS^0, φ_{old}, [Month,Country,Type,Salesman,Sum_old],sum(Sales)) και

c^{new}=(DS^0, φ_{new}, [Month,Country,Categoty,Salesman,Sum_new],sum(Sales))

where φ_{old}=18-Feb-97≤day ∧ day≤3-Sep-97 ∧ $\text{anc}_{\text{Item}}^{\text{Category}}$(Item)= "Books" και

φ_{new}=1-Mar-97≤day ∧ day≤3-Sep-97 ∧ "Literature"≤$\text{anc}_{\text{Item}}^{\text{Type}}$(Item) ∧ $\text{anc}_{\text{Item}}^{\text{Type}}$(Item)≤ "Philosophy".

Για να ελέγξουμε αν ο c^{new} μπορεί να υπολογισθεί από τον c^{old} εφαρμόζουμε το Θεώρημα 4.2. Τα σχήματα και οι αθροιστικές συναρτήσεις των δύο κύβων είναι συμβατά (συνθήκες (1) και (2) του Θεωρήματος 4.2). Επιπλέον, η φ_{new} είναι **L**-επικαλυπτόμενη από τη φ_{old} σε σχέση με τα επίπεδα του c^{new}. Ακολουθώντας τις γραμμές 2-10 του Αλγόριθμου Cube_Usability, μετασχηματίζουμε τη φ_{new} ώστε να είναι εφαρμόσιμη στο σχήμα του κύβου c^{old}. Οι μετασχηματισμοί των γραμμών 3-8 καταλήγουν στην

$\varphi_{n@o}$=Mar-97≤Month ∧ Month≤Sep-97∧"Literature"≤Type ∧ Type≤"Philosophy". Εφαρμόζουμε τη μετασχηματισμένη συνθήκη επιλογής στον c^{old} (όπως φαίνεται στο σχήμα 16a) και παράγουμε ένα νέο σύνολο δεδομένων DS^1 (όπως φαίνεται στο σχήμα 16b). Έπειτα αντικαθιστούμε όλες τις τιμές του DS^1 με τις αντίστοιχές του στα επίπεδα του c^{new} (Γραμμή 19), καταλήγοντας σε ένα νέο σύνολο δεδομένων DS^2 (όπως φαίνεται στο σχήμα 16c). Τέλος, συναθροίζουμε τις πλειάδες του DS^2 και παράγουμε το c^{new} (όπως φαίνεται στο σχήμα 16d). ■

Month	Type	Salesman	Country	Sum_old
Feb-97	Literature	Netz	USA	5
Sep-97	Philosophy	Netz	Japan	50
Sep-97	Literature	Netz	Japan	30

(a)

Month	Type	Salesman	Country	Sum_1
Sep-97	Philosophy	Netz	Japan	50
Sep-97	Literature	Netz	Japan	30

(b)

Month	Category	Salesman	Country	Sum_2
Sep-97	Book	Netz	Japan	50
Sep-97	Book	Netz	Japan	30

(c)

Month	Category	Salesman	Country	Sum_new
Sep-97	Book	Netz	Japan	80

(d)

Σχήμα 16: Υπολογίζοντας το c^{new} από το c^{old}.

5. ΣΥΜΠΕΡΑΣΜΑΤΑ

Στο άρθρο αυτό κάναμε μια παρουσίαση του χώρου της τεχνολογίας OLAP. Στη συνέχεια, παρουσιάσαμε ένα λογικό μοντέλο για κύβους, βασιζόμενοι στην παρατήρηση ότι ένας κύβος δεν είναι μια αυθύπαρκτη οντότητα, αλλά μια όψη πάνω σε ένα υποκείμενο σύνολο δεδομένων. Το προτεινόμενο μοντέλο είναι αρκετά ισχυρό στο να καλύπτει όλες τις συνηθισμένες πράξεις OLAP όπως επιλογή, συναθροιστική άνοδος και αναλυτική κάθοδος, μέσω μιας συνεπούς και πλήρης άλγεβρας. Δείξαμε επίσης πώς αυτό το μοντέλο μπορεί να χρησιμοποιηθεί σαν η βάση για την επεξεργασία λειτουργιών στους κύβους και παρουσιάσαμε συντακτικούς χαρακτηρισμούς για τα προβλήματα της χρησιμότητα κύβων. Το Θεώρημα 4.2 που μας δίνει αυτούς τους συντακτικούς χαρακτηρισμούς είναι σημαντικό για τις συνήθεις λειτουργίες του μοντέλου. Δύο επιφανείς περιπτώσεις είναι (α) μετάβαση από ένα κύβο c σε ένα άλλο κύβο που έχει όλα του επίπεδα υψηλότερα ή ίσα από τα επίπεδα του c και (β) επιλογή πάνω σε ένα κύβο c όπου όλα τα επίπεδα που συμμετέχουν στην έκφραση της συνθήκης είναι υψηλότερα ή ίσα από τα επίπεδα του c.

Φυσικά, η εφαρμογή του Θεωρήματος 4.2 δεν περιορίζεται σε αυτές τις δύο απλές περιπτώσεις. Συνήθως μια οθόνη ενός OLAP εργαλείου περιέχει παραπάνω από ένα κύβο [Micr98]. Έτσι, μια αλληλεπιδραστική σύνοδος OLAP παράγει πολλούς κύβους οι οποίοι πιθανά επικαλύπτονται. Ο υπολογισμός ενός νέου συνόλου κύβων μπορεί πιθανά να επιτευχθεί με τη χρήση κύβων που είναι ήδη υπολογισμένοι και αποθηκευμένοι στην λανθάνουσα μνήμη (δεδομένου, φυσικά, ότι πληρούν τα κριτήρια του Θεωρήματος 4.2). Κατά συνέπεια, τα αποτελέσματα στο πρόβλημα της χρησιμότητας κύβων μπορούν να χρησιμοποιηθούν και για τη βελτιστοποίηση ερωτήσεων και για τις διαδικασίες διαχείρισης λανθάνουσας μνήμης. Τα αποτελέσματα αυτά μπορούν να χρησιμοποιηθούν και στο πρόβλημα της σχεδίασης της συγκεντρωτικής αποθήκης δεδομένων [ThLS99], όπου το βέλτιστο σύνολο όψεων (σε σχέση με τα κόστη επερώτησης και συντήρησης) πρέπει να εξαχθεί. Ελέγχοντας για τη χρησιμότητα κύβων, μπορούμε να αποφύγουμε επικαλύψεις στο τελικό σχήμα της συγκεντρωτικής αποθήκης δεδομένων και να βελτιώσουμε το χρόνο εκτέλεσης της διαδικασίας σχεδίασης [LSTV99].

Υπάρχουν διάφορα ανοικτά θέματα σαν συνέχεια της δουλειάς αυτής. Τα κομμάτια μοντελοποίησης μπορούν να επεκταθούν, επίσης, ώστε να συμπεριλάβουν διάφορες παραλλαγές της δομής των ιεραρχιών (μερικές συναρτήσεις προγόνου, ιεραρχίες που δεν συμμορφώνονται με το μοντέλο δικτύου [LeSh97], κλπ.). Τα θεωρητικά αποτελέσματα στην επεξεργασία ερωτήσεων μπορούν να επεκταθούν στο να διαχειρίζονται ένα μεγαλύτερο σύνολο από συνθήκες επιλογής, μερική επανεγγραφή και βελτιστοποίηση της φυσικής εκτέλεσης των λειτουργιών του κύβου.

ΑΝΑΦΟΡΕΣ

[AgGS95] R. Agrawal, A. Gupta, S. Sarawagi. Modeling Multidimensional Databases. IBM Research Report, IBM Almaden Research Center, September 1995.

[Arbo96] Arbor Software Corporation. Arbor Essbase.http://www.arborsoft.com/essbase.html, 1996.

[BaSa98] F. Baader and U. Sattler. Description Logics with Concrete Domains and Aggregation. Proceedings of the 13th European Conference on Artificial Intelligence (ECAI-98), pp. 336-340, 1998.

[BaPT97] E. Baralis, S. Paraboschi, E. Teniente. Materialized View Selection in a Multidimensional Database. In Proceedings of the 23rd International Conference on Very Large Databases (VLDB), Athens, August 1997.

[ChSh96] S. Chaudhuri, K. Shim: Optimizing Queries with Aggregate Views. In Proc. of EDBT 1996. Proceedings of the 5th International Conference on Extending Database Technology (EDBT-96), Avignon, France, March 25-29, 1996.

[CKPS95] S. Chaudhuri, S. Krishnamurthy, S. Potamianos, and K. Shim. Optimizing queries with materialized views. Proceedings of the 11th International Conference on Data Engineering (ICDE), IEEE Computer Society, pp. 190-200, Taipei, March 1995.

[Coll96] G. Colliat. OLAP, Relational, and Multidimensional Database Systems. SIGMOD Record, vol. 25, No.3, September 1996.

[CoNS99] S. Cohen, W. Nutt, A. Serebrenik: Rewriting Aggregate Queries Using Views. Proceedings of the 18th ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems (PODS), Philadelphia, Pennsylvania. ACM Press, 1999.

[CaTo97] L. Cabbibo and R. Torlone. Querying Multidimesional Databases. 6th International Workshop on Database Programming Languages (DBPL6), 1997.

[CaTo98a] L. Cabbibo, R. Torlone. A Logical Approach to Multidimensional Databases. In 6th EDBT, 1998.

[CaTo98b] L. Cabibbo, R. Torlone. From a Procedural to a Visual Query Language for OLAP. Proceedings of 10th International Conference on Scientific and Statistical Database Management (SSDBM), Capri, Italy, July 1998.

[DJLS96] S. Dar, H.V.Jagadish, A. Levy, D. Srivastava. Answering queries with aggregation using views. In Proc. of 22nd International Conference on Very Large Data Bases (VLDB), Mumbai, India, 1996.

[Ende72] H.B. Enderton, A Mathematical Introduction to Logic. Academic Press, 1972.

[GBLP96] J. Gray, A. Bosworth, A. Layman, H. Pirahesh. Data Cube. A Relational Aggregation Operator Generalizing Group-By, Cross-Tabs, and Sub-Totals. Proceedings of the 12th International Conference on Data Engineering (ICDE '96), New Orleans, February 1996. Also Microsoft Technical Report, MSR-TR-95-22, available at http://www.research.microsoft.com/~gray.

[GuHQ95] A. Gupta, V. Harinarayan, and D. Quass. Aggregate query processing in data warehouses. Proceedings of the 21st International Conference on Very Large Data Bases (VLDB), Zurich, Switzerland, Morgan Kaufmann Publishers, August 1995.

[GeJJ97] M. Gebhardt, M Jarke and S. Jacobs. A toolkit for negotiation support on multi-dimensional data. Proceedings of ACM SIGMOD International Conference on Management of Data. Tucson, Arizona, 1997.

[GyLa97] M. Gyssens, L.V.S. Lakshmanan. A Foundation for Multi-Dimensional Databases. Proceedings of the 23rd International Conference on Very Large Databases (VLDB), Athens, August 1997.

[GiLa98] F. Gingras, L. Lakshmanan. nD-SQL: A Multi-dimensional Language for Interoperability and OLAP. Proceedings of the 24th International Conference on Very Large Databases (VLDB), N. York, August 1998.

[Gupt97] H. Gupta. Selection of Views to Materialize in a Data Warehouse. In Proceedings of the 6th International Conference on Database Theory (ICDT-97), Delfi, Greece, 1997

[Info97] Informix, Inc.: The INFORMIX-MetaCube Product Suite. http://www.informix.com/informix/products/new_plo/metabro/metabro2.htm, 1997.

[JLVV00] M. Jarke, M. Lenzerini, Y. Vassiliou, P. Vassiliadis (eds.). Fundamentals of Data Warehouses. Springer-Verlag, 2000.

[Kimb96] R. Kimball. The Data Warehouse Toolkit: Practical techniques for building dimensional data warehouses. John Wiley. 1996.

[LeAW98] W. Lehner, J. Albrect, H. Wedekind. Normal Forms for Multidimensional Databases. Proceedings of 10th International Conference on Scientific and Statistical Database Management (SSDBM), Capri, Italy, July 1998.

[Lehn98] W. Lehner. Modeling Large Scale OLAP Scenarios. Proceedings of the 6th International Conference of Extending Database Technology (EDBT-98), 1998.

[LMSS95] A.Y. Levy, A.O. Mendelzon, Y. Sagiv, and D. Srivastava. Answering queries using views. Proceedings of the 14th Symposium on Principles of Database Systems (PODS), pp. 95-104, San Jose (California, USA), May 1995. ACM Press.

[LeSh97] H. Lenz, A. Shoshani. Summarizability in OLAP and Statistical databases. In Proceedings of 9th International Conference on Scientific and Statistical Database Management (SSDBM), 1997.

[LSTV99] S.Ligoudistianos, T.Sellis, D.Theodoratos, and Y.Vassiliou. Heuristic Algorithms for Designing the Data Warehouse with SPJ Views. Proceedings of the First International Conference on Data Warehousing and Knowledge Discovery, (DaWaK), Lecture Notes in Computer Science, Vol. 1676, Springer, 1999.

[LiWa96] C. Li, X. Sean Wang. A Data Model for Supporting On-Line Analytical Processing. Proceedings of the International Conference on Information and Knowledge Management (CIKM), 1996.

[Meta97] Metadata Coalition. Metadata Interchange Specification (MDIS v. 1.1). http://www.metadata.org/standards/toc.html 1997.

[Micr98] Microsoft Corp. OLEDB for OLAP February 1998. Available at http://www.microsoft.com/data/oledb/olap/

[MStr97] MicroStrategy, Inc. MicroStrategy's 4.0 Product Line. http://www.strategy.com/launch/4_0_arc1.htm, 1997.

[NuSS98] W.Nutt, Y.Sagiv, S. Surin. Deciding Equivalences among Aggregate Queries. In Proc. 17th ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems (PODS), Seattle, USA, 1998

[OLAP97] OLAP Council. OLAP AND OLAP Server Definitions. 1997 Available at http://www.olapcouncil.org/research/glossaryly.htm

[OLAP97a] OLAP Council. The APB-1 Benchmark. 1997. Available at http://www.olapcouncil.org/research/bmarkly.htm

[RBSI97] Red Brick Systems, Inc. Red Brick Warehouse 5.0. http://www.redbrick.com/rbs-

g/html/whouse50.html, 1997.
[Sara97] Sunita Sarawagi. Indexing OLAP Data. Data Engineering Bulletin 20(1): 36-43 (1997).
[Shos97] A. Shoshani. OLAP and Statistical Databases: Similarities and Differences. Tutorials of 16th Symposium on Principles Of Database Systems (PODS), 1997.
[STGI96] Stanford Technology Group, Inc. Designing the Data Warehouse on Relational Databases. http://www.informix.com/informix/corpinfo/zines/whitpprs/stg/metacube.htm, 1996.
[ThLS99] D.Theodoratos, S.Ligoudistianos, and T.Sellis. Designing the Global DW with SPJ Queries. In Proc. of the 11th Conference on Advanced Information Systems Engineering (CAiSE), pages 180--194, June 1999.
[TPC99] TPC: TPC Benchmark H and TPC Benchmark R. Transcation Processing Council. June 1999. Available at http://www.tpc.org/
[Ullm89] J. Ullman. "Principles of Database and Knowledge-Base Systems. Volume II: The New Technologies. Computer Science Press. 1989.
[Ullm97] J.D. Ullman. Information integration using logical views. Proceedings of the 6th International Conference on Database Theory (ICDT-97), Lecture Notes in Computer Science, pp. 19-40. Springer-Verlag, 1997.
[VaSe99] P. Vassiliadis, T. Sellis. “A Survey on Logical Models for OLAP Databases”. SIGMOD Record, vol. 28, no. 4, December 1999.
[VaSk00] P. Vassiliadis, S. Skiadopoulos. Modelling and Optimization Issues for Multidimensional Databases. In Proc. 12th Conference on Advanced Information Systems Engineering (CAiSE '00), pp. 482-497, Stockholm, Sweden, 5-9 June 2000. Lecture Notes in Computer Science, Vol. 1789, Springer, 2000.
[VaSk99] P. Vassiliadis, S. Skiadopoulos. Modeling and Optimization Issues for Multidimensional Databases, Extended version, Technical Report, KDBSL 1999. Available at http://www.dblab.ece.ntua.gr/~pvassil/publications/cube99.ps.gz
[Vass98] P. Vassiliadis. Modeling Multidimensional Databases, Cubes and Cube Operations. Proceedings of 10th International Conference on Scientific and Statistical Database Management (SSDBM), Capri, Italy, July 1998.
[YaLa85] P. Larson, H. Z. Yang: Computing Queries from Derived Relations. In Proc. of VLDB 1985. Proceedings of the 11th International Conference on Very Large Data Bases (VLDB), Stockholm, Sweden, Morgan Kaufmann Publishers, August 1985.

ΓΛΩΣΣΑΡΙ

Στο εν λόγω γλωσσάρι αναφέρουμε τη μετάφραση των αγγλικών όρων όπως την προτείνουμε εμείς, το βιβλίο Elmasri & Navathe, και τα διαδικτυακά λεξικά του ΕΚΠΑ και ΟΠΑ. Όπου οι όροι δεν παρουσιάζονται αυτούσιοι, αναφέρουμε τη μετάφραση σχετικών όρων.

Αγγλικός όρος	**Μετάφραση**	**Elmasri & Navathe (Μτφρ. Χατζόπουλος)**	**ΕΚΠΑ[1]**	**ΟΠΑ[2]**
On-Line Analytical Proecssing (OLAP)	Σύγχρονη Αναλυτική Επεξεργασία Δεδομένων	- (On-Line = άμεση)	-	- (On-Line = ‘επί της γραμμής’ ή ‘επιγραμμική’)
Cube (Hypercube)	Κύβος (Υπερκύβος)	Κύβος	-	-
Measure	Μέτρο	-	-	-
Dimension	Διάσταση	Διάσταση	-	Διάσταση
Navigation	Πλοήγηση	-	Πλοήγηση	Πλοήγηση
Roll-up	Συναθροιστική Άνοδος	Ανοδική Παρουσίαση/ Άνοδος	-	- (rolling = κατακόρυφη κύλιση)
Drill-down	Αναλυτική Κάθοδος	Καθοδική Παρουσίαση/ Κάθοδος	-	-
Pivoting	Περιστροφή	Περιστροφή	-	-
Selection	Επιλογή	Επιλογή	Επιλογή	Επιλογή
Slicing	Τεμαχισμός	Τεμαχισμός	-	Τεμαχισμός
Star Schema	Σχήμα Αστέρα	Αστεροειδές Σχήμα	-	- (star network = αστεροειδές δίκτυο)
Snowflake Schema	Σχήμα Νιφάδας	Σχήμα Χιονονιφάδας	-	-
Fact Table	Πίνακας Πληροφοριών	Πίνακας Γεγονότων	-	- (fact = δεδομένο)
Dimension Table	Πίνακας Διάστασης	Πίνακας Διάστασης	-	-

[1] http://www.di.uoa.gr/~infodict/English/dict_search.html

[2] http://www.cs.aueb.gr/eleto-INFORTERM/indexgr.htm