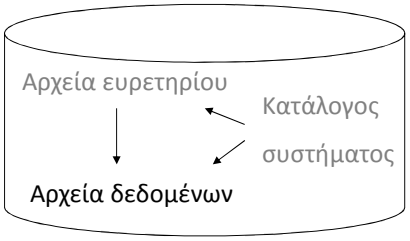
Εισαγωγή στην

Επεξεργασία Ερωτήσεων

Εισαγωγή

ΣΔΒΔ

Σύνολο από προγράμματα για τη διαχείριση της ΒΔ

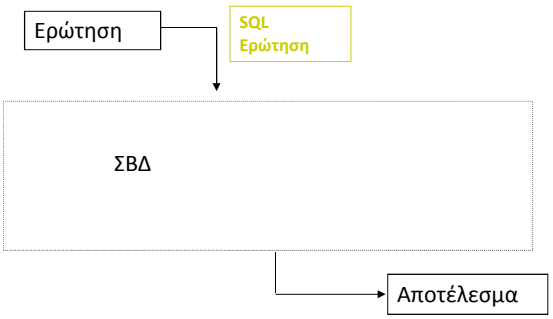


ΒΑΣΗ ΔΕΔΟΜΕΝΩΝ

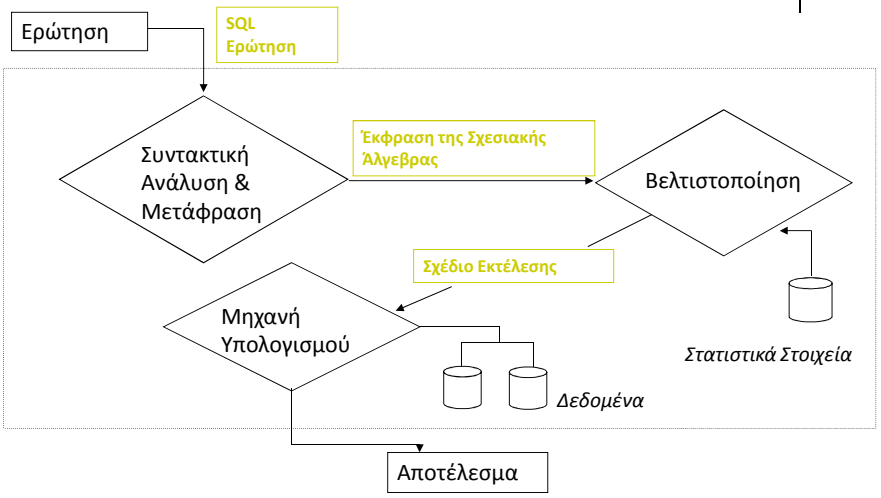
Σύστημα Βάσεων Δεδομένων (ΣΒΔ)

Επεξεργασία Ερωτήσεων

Θα δούμε την «πορεία» μιας SQL ερώτησης (πως εκτελείται)



Επεξεργασία Ερωτήσεων



Επεξεργασία Ερωτήσεων

Τα βασικά βήματα στην επεξεργασία μιας ερώτησης είναι

1. Συντακτική Ανάλυση & Μετάφραση
2. Βελτιστοποίηση
3. Υπολογισμός

Συντακτική Ανάλυση & Μετάφραση

1. Συντακτική Ανάλυση (Parsing) & Μετάφραση

Η SQL ερώτηση μεταφράζεται σε μια εσωτερική μορφή αφού γίνει ο απαραίτητος συντακτικός και σημασιολογικός έλεγχος (π.χ., τα ονόματα που αναφέρονται είναι ονόματα σχέσεων που υπάρχουν)

Αντικατάσταση των όψεων από τον ορισμό τους

Σε ποια εσωτερική μορφή; Έκφραση της σχεσιακής άλγεβρας

select $A_1, A_2, \ldots, A_n$

from $R_1, R_2, \ldots, R_m$

where P

$\boldsymbol{\pi}_{A1, A2, \ldots, An} (\boldsymbol{\sigma}_{P} (R_1 \mathbf{x} R_2 \mathbf{x} \ldots \mathbf{x} R_m))$

2. Βελτιστοποίηση

Μια SQL ερώτηση μπορεί να μεταφραστεί σε διαφορετικές (ισοδύναμες) εκφράσεις της σχεσιακής άλγεβρας

select balance
from account
where balance < 25000

- $\pi_{\text{balance}} (\sigma_{\text{balance} < 2500} (\text{account}))$
- $\sigma_{\text{balance} < 2500} (\pi_{\text{balance}}(\text{account}))$

Με ποιο κριτήριο γίνεται η επιλογή της έκφρασης;

- *Η βελτιστοποίηση είναι το πιο «δύσκολο» βήμα*

Κάθε πράξη της σχεσιακής άλγεβρας μπορεί να υλοποιηθεί με *διαφορετικούς αλγορίθμους*:

π.χ., για την υλοποίηση της επιλογής μπορεί

είτε να σαρώσουμε (scan – σειριακή αναζήτηση) όλο το αρχείο ελέγχοντας κάθε εγγραφή αν ικανοποιεί τη συνθήκη

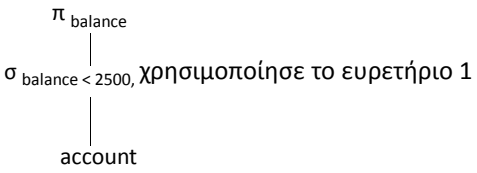
είτε αν υπάρχει π.χ., ένα B^+ ευρετήριο στο γνώρισμα balance να χρησιμοποιήσουμε το ευρετήριο

Άρα δεν αρκεί ο προσδιορισμός της πράξης - πρέπει να προσδιορίζεται **και** ο αλγόριθμος που θα χρησιμοποιηθεί για την υλοποίησή της

Βελτιστοποίηση

βασικές (primitive) πράξεις: πράξη + αλγόριθμος

Σχέδιο εκτέλεσης (execution plan): μια ακολουθία από βασικές πράξεις

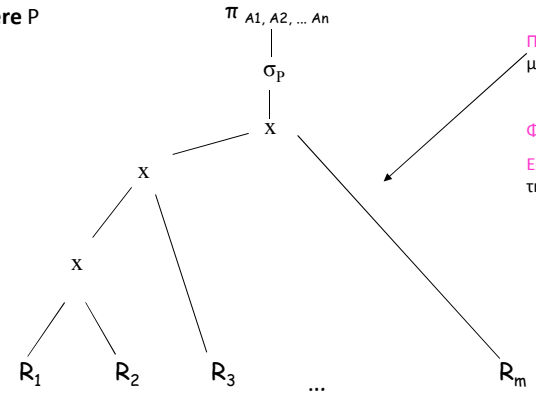


Βελτιστοποίηση

select $A_1, A_2, \ldots, A_n$
from $R_1, R_2, \ldots, R_m$
where P

Μετάφραση

$\pi_{A1, A2, \ldots, An}(\sigma_P(R_1 \times R_2 \times \ldots \times R_m))$



Πλάνο εκτέλεσης (ποιες πράξεις και με ποιον αλγόριθμο)

Φύλλα: σχέσεις

Εσωτερικοί κόμβοι: βασικές πράξεις της σχεσιακής άλγεβρας

Βελτιστοποίηση του πλάνου

Βελτιστοποίηση

- Τα διαφορετικά σχέδια εκτέλεσης έχουν και διαφορικό κόστος
- Βελτιστοποίηση: η διαδικασία επιλογής του σχεδίου εκτέλεσης που έχει το μικρότερο κόστος
- Εκτίμηση του κόστους (συνήθως χρήση στατιστικών στοιχείων)

Βελτιστοποίηση Ερωτήσεων

Μερικοί ευριστικοί κανόνες

Γενική ιδέα: εκτέλεση πρώτα των πράξεων με μικρή επιλεξικότητα ώστε να περιοριστεί το μέγεθος των ενδιάμεσων αποτελεσμάτων

1. Διάσπαση των πράξεων επιλογής με συζευκτικές συνθήκες σε ακολουθίες πράξεων επιλογής

2. Μετατοπίζουμε την *πράξη επιλογής όσο πιο κάτω* επιτρέπεται από τα γνωρίσματα που περιλαμβάνονται στη συνθήκη

3. Επανα-διευθέτηση των φύλλων ώστε να εκτελούνται πρώτα οι σχέσεις που έχουν τις πιο περιοριστικές πράξεις επιλογής

Βελτιστοποίηση Ερωτήσεων

4. Συνδυασμός μιας πράξης καρτεσιανού γινομένου με μια πράξη επιλογής που ακολουθεί

5. Διάσπαση και *μετακίνηση των λιστών προβολής όσο πιο κάτω* γίνεται στο δέντρο

6. Εντοπισμός υποδέντρων με ομάδες πράξεων που μπορεί να εκτελεστούν με κοινό αλγόριθμο

Εκτέλεση

3. Εκτέλεση

Μηχανή εκτέλεσης που εκτελεί τις βασικές πράξεις

Εκτέλεση

- Υπάρχουν υλοποιημένοι μια σειρά από αλγόριθμοι για κάθε βασική πράξη (π.χ., που χρησιμοποιούν ή όχι ευρετήρια κλπ)

- Γενικά, το ΣΔΒΔ κάνει μια *εκτίμηση του κόστους* και *επιλέγει τον αλγόριθμο* για κάθε πράξη με τον μικρότερο (με βάση την εκτίμηση) κόστος

- Η εκτίμηση του κόστους γίνεται με βάση στατιστικά στοιχεία που αποθηκεύονται στη βάση δεδομένων για αυτό το σκοπό

Αλγόριθμοι Εκτέλεσης Βασικών Πράξεων

Για να επιλέξουμε ποιόν αλγόριθμο θα χρησιμοποιήσουμε, διατηρούμε στατιστικά στοιχεία

Παράδειγμα

Για ένα *αρχείο δεδομένων* μιας σχέσης R, μπορεί να διατηρούμε στοιχεία όπως:

- n_R: αριθμός πλειάδων της σχέσης R
- b_R : αριθμός blocks της σχέσης R
- s_R: μέγεθος σε bytes κάθε πλειάδας της σχέσης R
- f_R : παράγοντας ομαδοποίησης (αριθμός εγγραφών ανά block)

αν μη εκτεινόμενη, $f_R = \lfloor B / s_R \rfloor$ και $b_R = \lceil n_R / f_R \rceil$

Ενημέρωση στατιστικών στοιχείων;

Αλγόριθμοι Εκτέλεσης Βασικών Πράξεων

Άλλα στατιστικά στοιχεία;

- V(A, R): πλήθος των διαφορετικών τιμών που παίρνει το γνώρισμα A

  $|\pi_A(R)|$ -- αν το A κλειδί;

- SC(A, R): μέσος αριθμός πλειάδων που ικανοποιεί μια συνθήκη (δεδομένου ότι υπάρχει μια τουλάχιστον που την ικανοποιεί)

  1 αν κλειδί, αν ομοιόμορφη;

Αλγόριθμοι Εκτέλεσης Βασικών Πράξεων

Στατιστικά στοιχεία επίσης για το *αρχείο ευρετηρίου* (αν υπάρχει)

- f_i : παράγοντας διακλάδωσης,
  - πολυεπίπεδο f_0, B^+ δέντρο ~ τάξη
- H_i: αριθμός επιπέδων
- LB_i: αριθμός block φύλλων

Με βάση τα στατιστικά επιλέγεται ο αλγόριθμος με το μικρότερο κόστος

I/O Κόστος (Αριθμό blocks που μεταφέρονται)

- ✓ Στη συνέχεια, θα δούμε κάποιους αλγορίθμους για την *εκτέλεση βασικών πράξεων της σχεσιακής άλγεβρας* και κάποια εκτίμηση του κόστους τους

Διαφορετικοί αλγόριθμοι ανάλογα με το αν το αρχείο είναι ή όχι διατεταγμένο, αν υπάρχει ή όχι ευρετήριο και από το είδος του ευρετηρίου

Επιλογή

Επιλογή

Πιθανοί *αλγόριθμοι εκτέλεσης* για την επιλογή:

E1: Σειριακή αναζήτηση

E2: Δυαδική αναζήτηση (αν το αρχείο είναι ταξινομημένο)

E3: Χρήση πρωτεύοντος ευρετηρίου/κατακερματισμού (αν υπάρχει)

E4: Χρήση δευτερεύοντος ευρετηρίου/κατακερματισμού (αν υπάρχει)

Αν υπάρχει κάποιο ευρετήριο, λέμε ότι έχουμε μονοπάτι προσπέλασης (access path)

Επιλογή: Συνθήκη Ισότητας

Επιλογή - συνθήκη ισότητας $\sigma_{A = \alpha}(R)$

E1 Σειριακή αναζήτηση

Διάβασμα (scan) όλου του αρχείου

b_R: *αριθμός blocks της σχέσης R*

b_R

$b_R/2$ (μέσος όρος) αν το A υποψήφιο κλειδί (οπότε το αποτέλεσμα έχει μόνο μία πλειάδα, σταματάμε την αναζήτηση μόλις τη βρούμε)

Μπορεί να χρησιμοποιηθεί σε οποιοδήποτε αρχείο

Επιλογή: Συνθήκη Ισότητας

E2 Δυαδική αναζήτηση

b_R: αριθμός blocks της σχέσης R
SC(A, R): μέσος αριθμός πλειάδων που ικανοποιεί μια συνθήκη («ταιριάσματα»)
f_R : παράγοντας ομαδοποίησης

Μπορεί να χρησιμοποιηθεί μόνο αν το αρχείο είναι *διατεταγμένο* με βάση το Α (δηλαδή, το γνώρισμα της επιλογής)

$\lceil \log (b_R) \rceil$ ← Εύρεση της πρώτης

\+

$\lceil SC(A, r)/f_R \rceil - 1$ ← Εύρεση των υπόλοιπων

Αν το Α υποψήφιο κλειδί;

Επιλογή: Συνθήκη Ισότητας

E3 Χρήση πρωτεύοντος δεντρικού ευρετηρίου

Πρωτεύον ευρετήριο σημαίνει ταξινομημένο αρχείο

b_R: αριθμός blocks της σχέσης R
SC(A, R): μέσος αριθμός πλειάδων που ικανοποιεί μια συνθήκη
f_R: παράγοντας ομαδοποίησης
HT_i: αριθμός επιπέδων (ύψος)

Μπορεί να χρησιμοποιηθεί μόνο αν υπάρχει τέτοιο ευρετήριο στο Α

$HT_i + 1$ ← Εύρεση και μεταφορά της πρώτης

Αν το Α δεν είναι υποψήφιο κλειδί -- ευρετήριο συστάδων

$HT_i + \lceil SC(A, R)/f_R \rceil$ ← Εύρεση και των υπόλοιπων

ΣΗΜΕΙΩΣΗ: Πρωτεύον ευρετήριο στο Α, σημαίνει ότι οι εγγραφές του αρχείου δεδομένων είναι ταξινομημένες (διατεταγμένες) ως προς Α άρα οι υπόλοιπες εγγραφές με την ίδια τιμή (αν υπάρχουν) βρίσκονται σε γειτονικά blocks του αρχείου δεδομένων

Επιλογή: Συνθήκη Ισότητας

E4 Χρήση δευτερεύοντος δεντρικού ευρετηρίου

b_R: αριθμός blocks της σχέσης R
SC(A, R): μέσος αριθμός πλειάδων που ικανοποιεί μια συνθήκη
f_R: παράγοντας ομαδοποίησης
HT_i: αριθμός επιπέδων

Μπορεί να χρησιμοποιηθεί μόνο αν υπάρχει τέτοιο ευρετήριο στο Α

Αν το Α είναι υποψήφιο κλειδί

$HT_i + 1$ ⟵ Εύρεση και μεταφορά της πρώτης

Αν το Α δεν είναι υποψήφιο κλειδί + κόστος για την εύρεση των υπολοίπων

HT_i + *ενδιάμεσο επίπεδο*

+SC(A, R) ⟵ Εύρεση και των υπόλοιπων

Στη χειρότερη περίπτωση κάθε εγγραφή που ικανοποιεί τη συνθήκη σε **διαφορετικό block**

Επιλογή: Συνθήκη με Σύγκριση

Επιλογή - συνθήκη με σύγκριση

$\sigma_{A \leq u}(R)$ ή $\sigma_{A \geq u}(R)$

Επιλογή - συνθήκη με σύγκριση

$\sigma_{A \leq u}(R)$

Έστω αύξουσα διάταξη

Από το 1ο block του αρχείου έως την πρώτη εγγραφή με A > u

Κόστος?

Επιλογή - συνθήκη με σύγκριση

$\sigma_{A \leq u}(R)$

Έστω αρχείου σωρού (δεν υπάρχει διάταξη) και B+ δέντρο

Εύρεση στο B+ δέντρο της τιμής u
Χρήση εγγραφών στο φύλλο για τις υπόλοιπες τιμές

Κόστος?

Επιλογή: Συνθήκη Σύζευξης

Συζευκτική επιλογή $\sigma_{P1 \text{ AND } \ldots \text{ AND } .. \text{ PM}}(R)$

Υπάρχει διαδρομή προσπέλασης για *ένα* από τα γνωρίσματα που εμφανίζονται σε οποιαδήποτε συνθήκη

Επιλογή του γνωρίσματος συνθήκη με τη *μικρότερη* επιλεκτικότητα (γιατί;)

Χρήση μιας από τις προηγούμενες μεθόδους για την ανάκτηση των εγγραφών που ικανοποιούν αυτήν την συνθήκη και

έλεγχος για κάθε επιλεγμένη εγγραφή αν ικανοποιεί και τις υπόλοιπες συνθήκες

Επιλογή: Συνθήκη Διάζευξης

Επιλογή - συνθήκη διάζευξης

$\sigma_{P1 \text{ OR } P2 \ldots \text{ OR } Pn}(R)$

Αν έστω και μία από τις συνθήκες δεν έχει διαδρομή προσπέλασης -> σάρωση όλου του αρχείου

Αλγόριθμους εκτέλεσης βασικών πράξεων

- Επιλογή
- Συνένωση
- Πράξεις συνόλων

Συνένωση

$$R \triangleright\!\triangleleft_{R.A\ op\ S.B} S$$

- Σ1 Εμφωλευμένος (εσωτερικός - εξωτερικός) βρόγχος
- Σ2 Χρήση μιας δομής προσπέλασης
- Σ3 Ταξινόμηση-Συγχώνευση

Έχει σημασία πόσο χώρο μνήμης κάθε χρονική στιγμή (buffers) μπορούμε να χρησιμοποιήσουμε για τις σχέσεις – δηλαδή, πόσα blocks στην μνήμη

Αρχικά, ας υποθέσουμε ότι έχουμε **μόνο 2 blocks**

Συνένωση (εμφωλευμένος βρόγχος)

Σ1 Εμφωλευμένος (εσωτερικός-εξωτερικός) βρόγχος

```
Για κάθε εγγραφή t της R
        Για κάθε εγγραφή s της S
                Αν t[A] op s[B] πρόσθεσε το t s στο αποτέλεσμα
```

Αγνοώντας την εγγραφή των blocks του αποτελέσματος

$b_r + n_R * b_S$

Συνένωση (εμφωλευμένος βρόγχος)

```
Για κάθε block B_r της R
    Για κάθε block B_s της S
        Για κάθε εγγραφή t του B_r
            Για κάθε εγγραφή s του B_s
                Αν t[A] op s[B] πρόσθεσε το t s στο αποτέλεσμα
```

Αγνοώντας την εγγραφή των blocks του αποτελέσματος

$b_R + b_R * b_S$

Συμφέρει η τοποθέτηση της *μικρότερης* σχέσης στον εξωτερικό βρόγχο

Συνένωση (χρήση ευρετηρίου)

Σ2 Χρήση μιας δομής προσπέλασης

Η σχέση για την οποία υπάρχει ευρετήριο τοποθετείται στον *εσωτερικό* βρόγχο. Έστω ότι υπάρχει ευρετήριο για το γνώρισμα B της σχέσης S

Για κάθε block B_r της R

Για κάθε εγγραφή t του B_r

Χρησιμοποίησε το ευρετήριο στο B για να βρεις τις εγγραφές s της S τέτοιες ώστε t[A] op s[B]

$b_R + n_R * C$ όπου C το κόστος μιας επιλογής στο S (δηλαδή της εύρεσης της εγγραφής (εγγραφών) του S που ικανοποιούν τη συνθήκη)

Συνένωση (ταξινόμηση-συγχώνευση)

Σ3 Ταξινόμηση - Συγχώνευση

Έστω συνθήκη ισότητας

Ταξινόμησε τις πλειάδες της R στο γνώρισμα A

Ταξινόμησε τις πλειάδες της S στο γνώρισμα B

i : = 1; j := 1;

while ($i \leq n_R$ and $j \leq n_S$)

if ($R_i[A] < S_j[B]$)

i := i + 1; (*προχώρησε το δείκτη στην R *)

if ($R_i[A] > S_j[B]$)

j := j + 1; (* προχώρησε το δείκτη στην S*)

Συνένωση (ταξινόμηση-συγχώνευση)

```
else   (*  Ri[A] = Sj[B]  *)
        πρόσθεσε το Ri . Sj στο αποτέλεσμα
        k := j + 1;  (* γράψε και τις άλλες πλειάδες της S που ταιριάζουν, αν υπάρχουν *)
                while ((k ≤ nS) and (Ri[A] = Sk[B]))
                        πρόσθεσε το Ri . Sk στο αποτέλεσμα
                    k := k + 1;
      m := i + 1;  (* γράψε και τις άλλες πλειάδες της R που ταιριάζουν,
                                        αν υπάρχουν *)
                while ((m ≤ nR)  and (Rm[A] = Sj[B]))
                        πρόσθεσε το Rm . Sj στο αποτέλεσμα
                        m := m + 1;
       i := m; j := k;
```


Συνένωση (ταξινόμηση-συγχώνευση)

Αν αγνοήσουμε τη ταξινόμηση για τη *συγχώνευση* (merge) απλή σάρωση των δύο αρχείων:

$b_R + b_S$

Ταξινόμηση: $b_R * \log(b_R) + b_S * \log(b_S)$

Πράξεις συνόλου

- R ∪ S (ένωση)
- R ∩ S (τομή)
- R – S (διαφορά)

Θα δούμε έναν αλγόριθμο βασισμένο σε merge-sort (ταξινόμηση-συγχώνευση)

Ταξινόμησε τις πλειάδες της R σε ένα γνώρισμα (έστω A)

Ταξινόμησε τις πλειάδες της S στο ίδιο γνώρισμα

i : = 1; j := 1;

while ($i \leq n_R$ and $j \leq n_S$)

if ($R_i[A] > S_j[A]$)

Τομή	Ένωση	Διαφορά
τίποτα	γράψε το S_j στο αποτέλεσμα	τίποτα

j := j + 1

else if ($R_i[A] < S_j[A]$)

Τομή

τίποτα

Ένωση

γράψε το R_i στο αποτέλεσμα

Διαφορά

γράψε το R_i στο αποτέλεσμα

i := i + 1

else (* $R_i[A] = S_j[A]$ *)

Τομή

γράψε το R_i στο αποτέλεσμα

i := i + 1;

j := j + 1;

Ένωση

i := i + 1;

Διαφορά

i := i + 1;

j := j + 1;

Αν υπάρχουν ακόμα εγγραφές για κάποιο αρχείο:

Ένωση

while ($i \leq n_R$)

γράψε το R_i στο αποτέλεσμα

i: = i + 1;

while ($j \leq n_S$)

γράψε το S_j στο αποτέλεσμα

j: = j + 1;

Διαφορά

while ($i \leq n_R$)

γράψε το R_i στο αποτέλεσμα

i: = i + 1;

Τέλος