

Βελτιστοποίηση Σχεσιακών Αιτημάτων

Κεφάλαιο 15

Ο Βελτιστοποιητής του System R

- ❖ Επίδραση:
 - Ο πιο δημοφιλής – δουλεύει καλά με < 10 συζεύξεις.
- ❖ Υπολογισμός Κόστους: Η τέχνη της προσέγγισης.
 - Στατιστικά στοιχεία: διατηρούνται στους καταλόγους του συστήματος - χρησιμοποιούνται για την εκτίμηση του κόστους των πράξεων και του μεγέθους των αποτελεσμάτων.
 - Λαμβάνεται υπόψη ο συνδυασμός CPU και I/O κόστους.
- ❖ Ο Χώρος των Σχεδίων: Πολύ μεγάλος, πρέπει να “κλαδευτεί”.
 - Θεωρούμε μόνο το χώρο των *σχεδίων αριστερού βάθους*.
 - Τα σχέδια αριστερού βάθους επιτρέπουν τη *διασωλήνωση* της εξόδου της κάθε πράξης στην επόμενη πράξη χωρίς την ανάγκη αποθήκευσής της σε προσωρινό πίνακα.
 - Αποφεύγονται τα Καρτεσιανά γινόμενα.

Επισκόπηση της Βελτιστοποίησης Αιτημάτων

- ❖ **Σχέδιο:** Δέντρο με πράξεις Σχεσιακής Άλγεβρας, με επιλογή αλγορίθμου για κάθε πράξη.
 - Κάθε πράξη υλοποιείται χρησιμοποιώντας μια διεπαφή “λήψης”: όταν μια πράξη “ενεργοποιείται” για να παράξει τις εγγραφές εξόδου, αυτή “λαμβάνει” τις εγγραφές εισόδου και υπολογίζει την έξοδο της.
- ❖ Δυο βασικά ζητήματα:
 - Για ένα αίτημα, **ποια σχέδια εξετάζονται;**
 - Αλγόριθμος αναζήτησης στο χώρο των σχεδίων για το φθηνότερο (κατ' εκτίμηση) σχέδιο.
 - Πώς **εκτιμάται το κόστος ενός σχεδίου;**
- ❖ **Ιδανικά:** Θέλουμε να βρούμε το καλύτερο σχέδιο.
Πρακτικά: Πρέπει να αποφύγουμε τα χειρότερα σχέδια!
- ❖ Θα μελετήσουμε την προσέγγιση του System R.

Σχήμα Παραδειγμάτων

Sailors (sid: integer, sname: string, rating: integer, age: real)

Reserves (sid: integer, bid: integer, day: dates, rname: string)

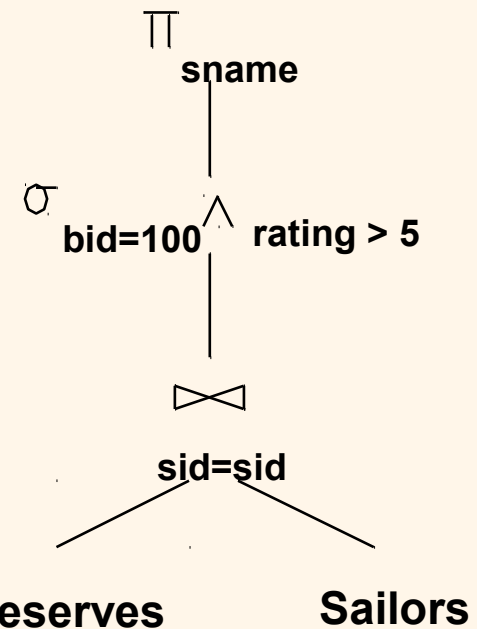
- ❖ Παρόμοιο με το παλιό σχήμα – προστέθηκε το *rname* για παραλλαγές στα αιτήματα.
- ❖ Reserves:
 - Κάθε εγγραφή είναι 40 bytes, 100 εγγραφές ανά σελίδα, 1000 σελίδες.
- ❖ Sailors:
 - Κάθε εγγραφή είναι 50 bytes, 80 εγγραφές ανά σελίδα, 500 σελίδες.

Παράδειγμα

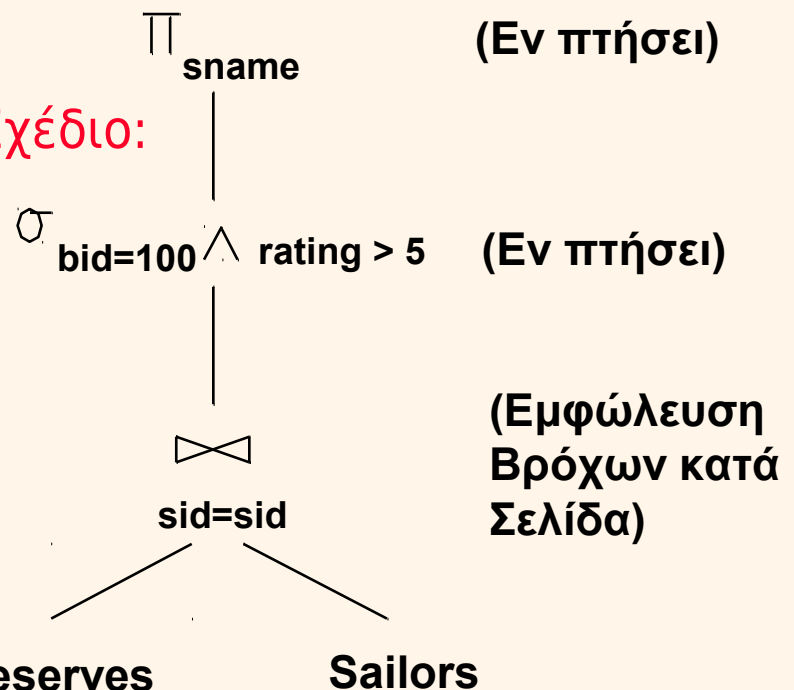
```
SELECT S.sname
FROM Reserves R, Sailors S
WHERE R.sid=S.sid AND
      R.bid=100 AND S.rating>5
```

- ❖ **Κόστος: $500+500*1000$ I/O**
- ❖ Σε καμιά περίπτωση το χειρότερο σχέδιο!
- ❖ Πολλές χαμένες ευκαιρίες: οι επιλογές θα μπορούσαν να είχαν “προωθηθεί” νωρίτερα, δεν χρησιμοποιούνται τα διαθέσιμα ευρετήρια, κλπ.
- ❖ *Στόχος της βελτιστοποίησης: Να βρεθούν πιο αποτελεσματικά σχέδια που υπολογίζουν την ίδια απάντηση.*

Δέντρο
Σχεσιακής
Άλγεβρας:



Σχέδιο:



Εναλλακτικό Σχέδιο 1 (Χωρίς Ευρετήρια)

❖ **Βασική διαφορά:** προώθηση των επιλογών.

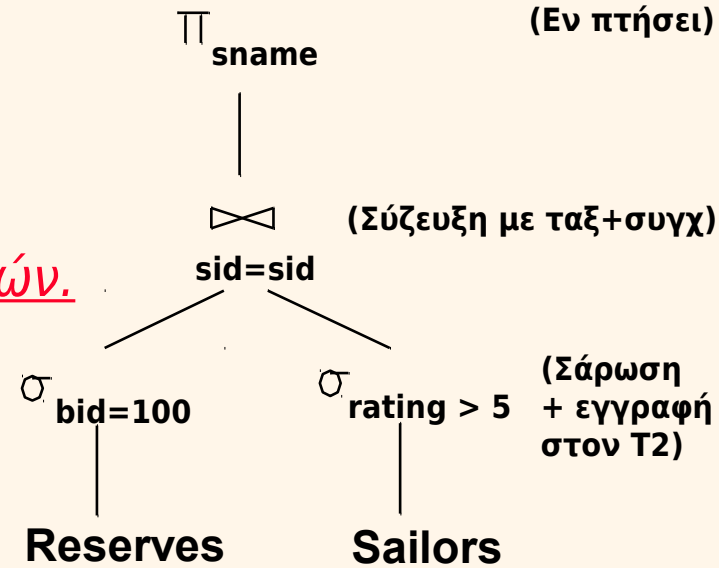
❖ Με 5 πλαίσια, κόστος σχεδίου:

- Σάρωση Reserves (1000) + εγγραφή προσωρινού T1 (10 σελίδες, αν έχουμε 100 βάρκες, ομοιόμορφη κατανομή).
- Σάρωση Sailors (500) + εγγραφή προσωρινού T2 (250 σελίδες, αν έχουμε 10 ratings).
- Ταξινόμηση T1 ($2 \times 2 \times 10$), ταξινόμηση T2 ($2 \times 3 \times 250$), συγχώνευση (10+250)
- **Σύνολο: 3560 I/O.**

❖ Αν είχαμε χρησιμοποιήσει σύζευξη με εμφ. βρόχων κατά μπλοκ, κόστος σύζευξης = $10 + 4 \times 250$, συνολικό κόστος = 2770.

❖ Αν είχαμε “προωθήσει” τις προβολές, ο T1 έχει μόνο το *sid*, ο T2 μόνο το *sid* και το *sname*:

- Ο T1 χωρά σε 3 pages, το κόστος της σύζευξης με εμφ. βρόχων κατά μπλοκ πέφτει κάτω από 250 σελίδες, **σύνολο < 2000.**



Εναλλακτικό Σχέδιο 2 (Με ευρετήρια)

❖ Με συγκροτημένο ευρετήριο στο *bid* του Reserves, έχουμε $100,000/100 = 1000$ εγγραφές σε $1000/100 = 10$ σελίδες.

❖ Σύζευξη με εμφ. βρόχους και ευρετήριο με **διασωλήνωση** (εξωτερικός δεν αποθηκεύεται).

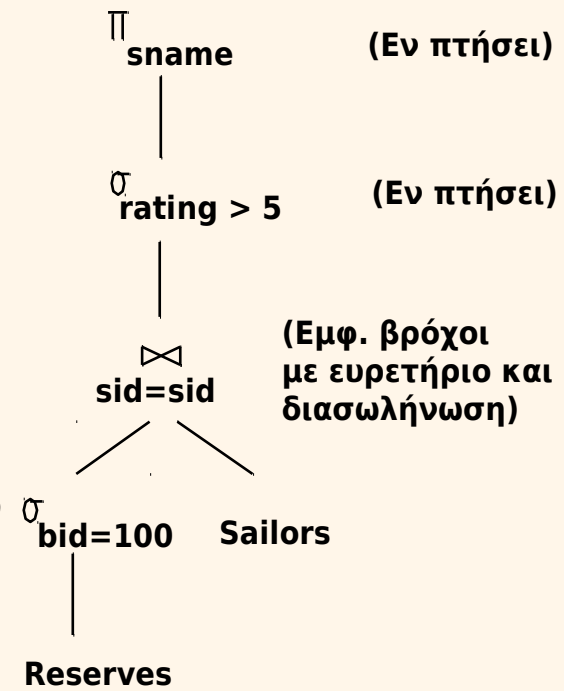
- Εξάλειψη άχρηστων πεδίων με προβολή δεν βοηθά.

❖ Το πεδίο σύζευξης *sid* είναι κλειδί του Sailors.

- Το πολύ μια ταιριαστή εγγραφή, μη-συγκρ. ευρετήριο στο *sid* είναι OK.

❖ Η απόφαση μη προώθησης του $rating > 5$ πριν τη σύζευξη βασίστηκε στη διαθεσιμότητα του ευρετηρίου στο *sid* του Sailors.

❖ **Κόστος:** Επιλογή των εγγραφών του Reserves (10 I/O). Για κάθεμιά, πρέπει να βρούμε την ταιριαστή εγγραφή του Sailors ($1000 * 1.2$); σύνολο **1210 I/O**.



Μπλοκ αιτήματος: Η μονάδα βελτιστοποίησης

- ❖ Ένα αίτημα SQL αναλύεται σε μια συλλογή από μπλοκ τα οποία βελτιστοποιούνται ένα μπλοκ τη φορά.
- ❖ Τα εμφωλευμένα μπλοκ συνήθως αντιμετωπίζονται ως κλήσεις υπορουτινών, μια ανά καθεμιά εξωτερική εγγραφή. (Αυτό συνιστά απλούστευση, αλλά προς το παρόν μας εξυπηρετεί.)

```
SELECT S.sname
FROM Sailors S
WHERE S.age IN
    (SELECT MAX (S2.age)
     FROM Sailors S2
     GROUP BY S2.rating)
```

Εξωτερικό μπλοκ

Εμφωλευμένο μπλοκ

- ❖ Για κάθε μπλοκ, θεωρούμε τα εξής σχέδια:
 - Όλες τις διαθέσιμες διαδρομές πρόσβασης για κάθε πίνακα στο FROM.
 - Όλα τα δέντρα συζεύξεων αριστερού βάθους (δηλ., έναν-έναν όλους τους τρόπους σύζευξης πινάκων του FROM, θεωρώντας όλους τους συνδυασμούς πινάκων και μεθόδων σύζευξης.)

Ισοδυναμίες Σχεσιακής Άλγεβρας

- ❖ Μας επιτρέπουν να επιλέξουμε διαφορετική σειρά συζεύξεων και να “προωθήσουμε προς τα κάτω” επιλογές και προβολές, πριν τις συζεύξεις.
 - ❖ Επιλογές: $\sigma_{c_1 \wedge \dots \wedge c_n}(R) \equiv \sigma_{c_1}(\dots \sigma_{c_n}(R))$ (Σύνθεση)
 $\sigma_{c_1}(\sigma_{c_2}(R)) \equiv \sigma_{c_2}(\sigma_{c_1}(R))$ (Αντιμετάθεση)
 - ❖ Προβολές: $\pi_{a_1}(R) \equiv \pi_{a_1}(\dots(\pi_{a_n}(R)))$ (Σύνθεση)
 - ❖ Συζεύξεις: $R \bowtie (S \bowtie T) \equiv (R \bowtie S) \bowtie T$ (Προσεταιρισμός)
 $(R \bowtie S) \equiv (S \bowtie R)$ (Αντιμετάθεση)
- ☞ Δείξτε ότι: $R \bowtie (S \bowtie T) \equiv (T \bowtie R) \bowtie S$

Περισσότερες Ισοδυναμίες

- ❖ Μια προβολή αντιμετωπίζεται με μια επιλογή που χρησιμοποιεί μόνο πεδία που διατηρεί η προβολή.
- ❖ Επιλογή ανάμεσα σε πεδία πινάκων ενός καρτεσιανού γινομένου, το μετατρέπει σε σύζευξη.
- ❖ Επιλογή μόνο σε πεδία του R αντιμετωπίζεται με το $R \bowtie S$. (δηλ., $\sigma(R \bowtie S) \equiv \sigma(R) \bowtie S$)
- ❖ Παρόμοια, αν μια προβολή έπεται μιας σύζευξης $R \bowtie S$, μπορούμε να την “προωθήσουμε” κρατώντας μόνο τα πεδία του R (και του S) που χρειάζονται για τη σύζευξη ή διατηρεί η προβολή.

Απαρίθμηση Εναλλακτικών Σχεδίων

- ❖ Υπάρχουν δυο περιπτώσεις:
 - Σχέδια με έναν πίνακα
 - Σχέδια με πολλούς πίνακες
- ❖ Για αιτήματα με έναν πίνακα, έχουμε ένα συνδυασμό επιλογών, προβολών και συναθροίσεων:
 - Εξετάζεται κάθε διαθέσιμη διαδρομή πρόσβασης (σάρωση αρχείου/ευρετήριο), και επιλέγεται αυτή με το μικρότερο εκτιμώμενο κόστος.
 - Ουσιαστικά, οι πράξεις εκτελούνται μαζί (π.χ., αν χρησιμοποιείται ένα ευρετήριο για την επιλογή, η προβολή γίνεται σε κάθε ανακτημένη εγγραφή, και οι εγγραφές που προκύπτουν **διασωληνώνονται** στον υπολογισμό της συνάθροισης).

Εκτίμηση Κόστους

- ❖ Για κάθε εξεταζόμενο σχέδιο πρέπει να εκτιμηθεί το κόστος:
 - Πρέπει να **εκτιμηθεί το κόστος** καθεμιάς πράξης του δέντρου σχεδίου.
 - Εξαρτάται από τους πληθαιθμούς των εισόδων.
 - Έχουμε ήδη συζητήσει πως εκτιμάται το κόστος των πράξεων (σειριακή σάρωση, σάρωση με ευρετήριο, συζεύξεις, κλπ.)
 - Πρέπει επίσης να **εκτιμηθεί το μέγεθος του αποτελέσματος** για κάθε πράξη του δέντρου!
 - Χρησιμοποιείται η πληροφορία που υπάρχει για τους πίνακες.
 - Για τις επιλογές και τις συζεύξεις, υποθέτουμε ανεξαρτησία των κατηγορημάτων.

Στατιστικά και Κατάλογοι

- ❖ Χρειαζόμαστε πληροφορίες για τους εμπλεκόμενους πίνακες και ευρετήρια. Συνήθως, οι **Κατάλογοι** περιέχουν τουλάχιστον:
 - # εγγραφών (NTuples) and # σελίδων (NPages) για κάθε πίνακα.
 - # διακριτών τιμών κλειδιών (NKeys) και # σελίδων (NPages) για κάθε ευρετήριο.
 - Ύψος ευρετηρίου, χαμηλότερη/υψηλότερη τιμή κλειδιού (Low/High) για κάθε δεντρικό ευρετήριο.
- ❖ Οι Κατάλογοι ενημερώνονται περιοδικά.
 - Το να ενημερώνονται κάθε φορά που αλλάζουν τα δεδομένα έχει πολύ υψηλό κόστος. Έτσι κι αλλιώς οι υπολογισμοί γίνονται κατά προσέγγιση, οπότε δεν πειράζει να υπάρχει λίγη ασυνέπεια.
- ❖ Μερικές φορές αποθηκεύεται πιο αναλυτική πληροφορία (π.χ., ιστογράμματα των τιμών κάποιου πεδίου).

Διαδρομές πρόσβασης

- ❖ Μια διαδρομή πρόσβασης είναι μια μέθοδος ανάκτησης εγγραφών:
 - Σάρωση αρχείου, ή ευρετήριο που ταιριάζει με μια επιλογή (του αιτήματος)
- ❖ Ένα δεντρικό ευρετήριο ταιριάζει με (ένα συνδυασμό από) όρους που εμπλέκουν μόνο πεδία κάποιου προθέματος του κλειδιού αναζήτησης.
 - Π.χ., Το δεντρικό ευρετήριο στο $\langle a, b, c \rangle$ ταιριάζει με τις επιλογές $a=5 \text{ AND } b=3$, και $a=5 \text{ AND } b>6$, αλλά όχι με την $b=3$.
- ❖ Ένα ευρετήριο κατακερματισμού ταιριάζει με (ένα συνδυασμό από) όρους που έχει τον όρο πεδίο = τιμή για κάθε πεδίο στο κλειδί αναζήτησης του ευρετηρίου.
 - Π.χ., Το ευρετήριο κατακερματισμού στο $\langle a, b, c \rangle$ ταιριάζει με την $a=5 \text{ AND } b=3 \text{ AND } c=5$, αλλά δεν ταιριάζει με την $b=3$, ή την $a=5 \text{ AND } b=3$, ή το $a>5 \text{ AND } b=3 \text{ AND } c=5$.

Περί Σύνθετων Επιλογών

$(day < 8/9/94 \text{ AND } rname = 'Paul') \text{ OR } bid = 5 \text{ OR } sid = 3$

- ❖ Οι συνθήκες επιλογής πρώτα μετατρέπονται σε κανονική μορφή συνένωσης - conjunctive normal form (CNF):

$(day < 8/9/94 \text{ OR } bid = 5 \text{ OR } sid = 3) \text{ AND } (rname = 'Paul' \text{ OR } bid = 5 \text{ OR } sid = 3)$

- ❖ Θα ασχοληθούμε μόνο με την περίπτωση που δεν περιέχει OR.

Εκτίμηση Κόστους για Σχέδια με ένα Πίνακα

- ❖ Ευρετήριο I στο κύριο κλειδί που ταιριάζει με την επιλογή:
 - Το κόστος είναι $Height(I)+1$ για *B+tree*, περίπου *1.2* για ευρετήριο κατακερματισμού.
- ❖ Συγκροτημένο ευρετήριο I που ταιριάζει με μια ή περισσότερες επιλογές:
 - $(NPages(I)+NPages(R)) * \text{γινόμενο των } RF \text{ των ταιριαστών επιλογών.}$
- ❖ Μη-συγκροτημένο ευρετήριο I που ταιριάζει με μια ή περισσότερες επιλογές:
 - $(NPages(I)+NTuples(R)) * \text{γινόμενο των } RF \text{ των ταιριαστών επιλογών.}$
- ❖ Σειριακή σάρωση αρχείου:
 - $NPages(R)$.
- ☞ **Σημείωση:** Συνήθως, *δεν εξαλείφονται τα διπλότυπα στις προβολές!* (Εξαίρεση: αν υπάρχει *DISTINCT*.)

Παράδειγμα

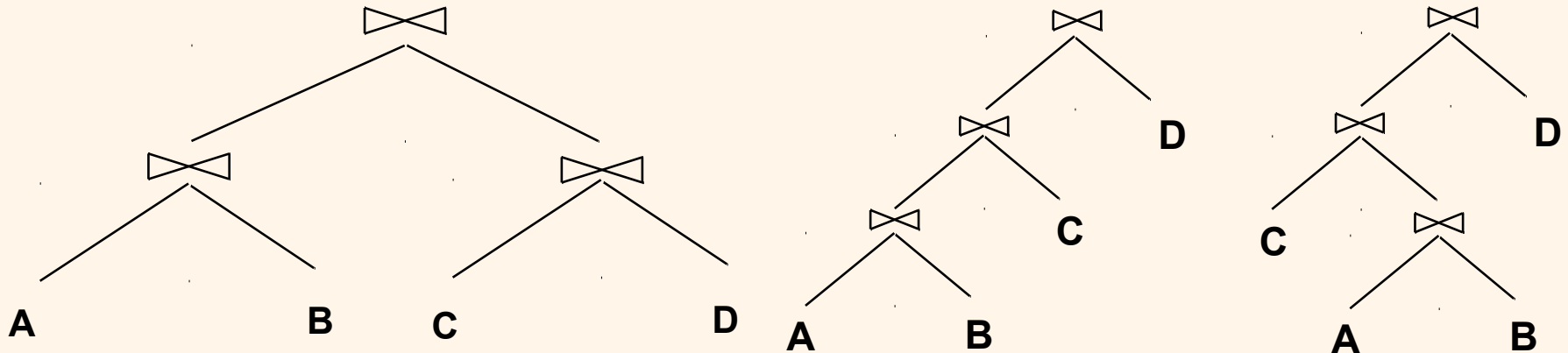
```
SELECT S.sid
FROM Sailors S
WHERE S.rating=8
```

- ❖ Αν έχουμε ένα **ευρετήριο στο *rating***:
 - $(1/NKeys(I)) * NTuples(R) = (1/10) * 40000$ εγγραφές.
 - **Συγκροτημένο ευρετήριο:** $(1/NKeys(I)) * (NPages(I) + NPages(R)) = (1/10) * (50+500)$ σελίδες.
(Αυτό είναι το **κόστος**.)
 - **Μη-συγκροτημένο ευρετήριο:** $(1/NKeys(I)) * (NPages(I) + NTuples(R)) = (1/10) * (50+40000)$ σελίδες.
- ❖ Αν έχουμε ένα **ευρετήριο στο *sid***:
 - Θα πρέπει να ανακτήσουμε όλες τις εγγραφές/σελίδες.
Με ένα **συγκροτημένο** ευρετήριο, το **κόστος είναι 50+500**,
με **μη-συγκροτημένο** ευρετήριο, **50+40000**.
- ❖ Με **σάρωση αρχείου**:
 - Ανακτούμε όλες τις σελίδες (**500**).

Αιτήματα με πολλούς πίνακες

❖ Θεμελιώδης απόφαση στο System R: εξετάζονται μόνο δέντρα συζεύξεων αριστερού βάθους.

- Καθώς ο αριθμός των συζεύξεων αυξάνει, ο αριθμός των εναλλακτικών σχεδίων αυξάνει ραγδαία. Πρέπει να περιοριστεί ο χώρος αναζήτησης.
- Τα δέντρα αριστερού βάθους μας επιτρέπουν να παράξουμε όλα τα πλήρως διασωληνωμένα σχέδια.
 - Τα ενδιάμεσα αποτελέσματα δεν αποθηκεύονται σε προσωρινά αρχεία.
 - Δεν είναι όλα τα δέντρα αριστερού βάθους πλήρως διασωληνωμένα (π.χ., σύζευξη με ταξινόμηση και συγχώνευση).



Απαρίθμηση των Σχεδίων αριστερού βάρθους

- ❖ Τα σχέδια αριστερού βάρθους διαφέρουν μόνο ως προς τη σειρά των πινάκων, τη μέθοδο πρόσβασης του κάθε πίνακα, και τη μέθοδο σύζευξης για κάθε σύζευξη.
- ❖ Απαρίθμηση σε N περάσματα (σε σύζευξη N πινάκων):
 - **Πέρασμα 1:** Βρες το καλύτερο σχέδιο ενός πίνακα για κάθε πίνακα.
 - **Πέρασμα 2:** Βρες τον καλύτερο τρόπο σύζευξης του αποτελέσματος του κάθε σχεδίου ενός πίνακα (ως εξωτερικού) με έναν άλλο πίνακα. *(Όλα τα σχέδια 2-πινάκων.)*
 - **Πέρασμα N :** Βρες τον καλύτερο τρόπο σύζευξης του αποτελέσματος ενός σχεδίου $(N-1)$ -πινάκων (ως εξωτερικού) με τον Νιοστό πίνακα. *(Όλα τα σχέδια N -πινάκων.)*
- ❖ Για κάθε υποσύνολο πινάκων, διατήρησε μόνο:
 - Το συνολικά φθηνότερο σχέδιο, συν
 - Το φθηνότερο σχέδιο για κάθε **ενδιαφέρουσα σειρά** των εγγραφών.

Απαρίθμηση των Σχεδίων (συνέχεια)

- ❖ Τα **ORDER BY, GROUP BY, συναθροίσεις** κλπ. αντιμετωπίζονται ως τελικό βήμα, χρησιμοποιώντας είτε ένα σχέδιο με “ενδιαφέρουσα σειρά” ή μια επιπλέον πράξη ταξινόμησης.
- ❖ Ένα σχέδιο με N-1 πίνακες δεν συνδυάζεται με επιπλέον πίνακα εκτός και αν υπάρχει συνθήκη σύζευξης ανάμεσά τους, ή εκτός αν έχουν χρησιμοποιηθεί όλα τα κατηγορήματα στο WHERE.
 - δηλ., **αποφεύγουμε τα καρτεσιανά γινόμενα** ει δυνατόν.
- ❖ Παρά το “κλάδεμα” του χώρου αναζήτησης, η προσέγγιση αυτή είναι **ακόμα εκθετική** ως προς το πλήθος των πινάκων.

Εκτίμηση Κόστους για Σχέδια με πολλούς πίνακες

- ❖ Έστω ένα μπλοκ αιτήματος:
- ❖ Ο μέγιστος # εγγραφών του αποτελέσματος είναι το γινόμενο των πληθάριθμων των πινάκων στο FROM.
- ❖ Ο *Συντελεστής μείωσης - Reduction Factor (RF)* που συσχετίζεται με κάθε *όρο* αντικατοπτρίζει την επίδραση του όρου στη μείωση του μεγέθους του αποτελέσματος.
*Πληθάριθμος αποτελέσματος = μέγιστο # εγγραφών * γινόμενο όλων των RF.*
- ❖ Σχέδια με πολλούς πίνακες κατασκευάζονται μέσω της σύζευξης ενός επιπλέον πίνακα κάθε φορά.
 - Το κόστος της μεθόδου σύζευξης, συν η εκτίμηση του πληθάριθμου της σύζευξης μας δίνει εκτίμηση και του κόστους και του μεγέθους του αποτελέσματος.

```
SELECT attribute list
FROM relation list
WHERE term1 AND ...
      AND termk
```

Παράδειγμα

Sailors:

B+ tree στο *rating*

Ευρ. Κατακερματισμού στο *sid*

Reserves:

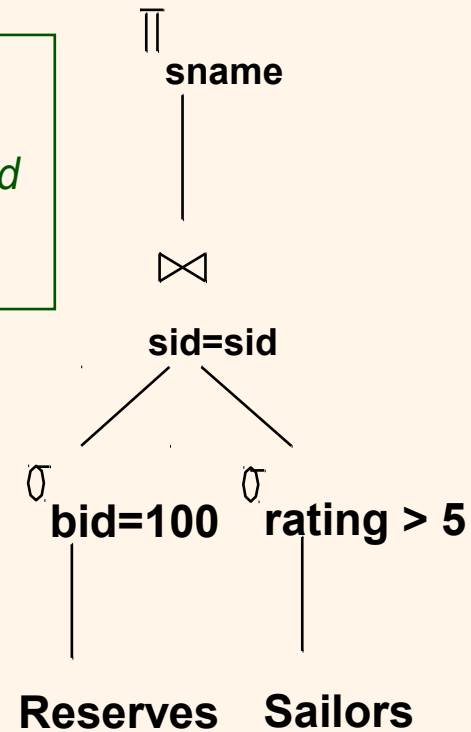
B+ tree στο *bid*

❖ Πέρασμα 1:

- **Sailors:** Υπάρχει B+tree που ταιριάζει με το $rating > 5$, και πιθανόν είναι το φθηνότερο. Όμως, η επιλογή αυτή αναμένεται να ανακτήσει πολλές εγγραφές, και καθώς το ευρετήριο είναι μη-συγκροτημένο, η σάρωση είναι φθηνότερη.
 - Παρόλα αυτά, κρατάμε το σχέδιο με το B+tree (επειδή οι εγγραφές είναι σε σειρά σύμφωνα με το *rating*).
- **Reserves:** Το B+tree στο *bid* ταιριάζει με το $bid=100$ και είναι το φθηνότερο.

❖ Πέρασμα 2:

- Θεωρούμε κάθε σχέδιο που διατηρούμε από το Πέρασμα 1 ως εξωτερικό, και εξετάζουμε πως μπορεί να γίνει η σύζευξη με τον (μοναδικό) άλλο πίνακα.
 - π.χ., με τον **Reserves ως εξωτερικό**: μπορεί να χρησιμοποιηθεί το ευρετήριο κατακερματισμού για την εύρεση των εγγραφών του Sailors που ικανοποιούν τη συνθήκη $sid = \text{τιμή του } sid \text{ του εξωτερικού πίνακα}$.



Εμφωλευμένα Αιτήματα

- ❖ Τα εμφωλευμένα μπλοκ βελτιστοποιούνται ανεξάρτητα, θεωρώντας ότι η εξωτερική εγγραφή μας δίνει μια συνθήκη επιλογής.
- ❖ Το εξωτερικό μπλοκ βελτιστοποιείται λαμβάνοντας υπόψη για κάθε “κλήση” το κόστους υπολογισμού του εμφωλευμένου μπλοκ.
- ❖ Η έμμεση σειρά των μπλοκ σημαίνει ότι μπορεί να μην εξεταστούν μερικές καλές στρατηγικές. *Η μη-εμφωλευμένη έκδοση του αιτήματος συνήθως μπορεί να βελτιστοποιηθεί καλύτερα.*

```
SELECT S.sname
FROM Sailors S
WHERE EXISTS
  (SELECT *
   FROM Reserves R
   WHERE R.bid=103
   AND R.sid=S.sid)
```

Εμφωλευμένο μπλοκ προς βελτιστοποίηση:

```
SELECT *
FROM Reserves R
WHERE R.bid=103
      AND R.sid= outer value
```

Ισοδύναμο μη-εμφωλευμένο αίτημα:

```
SELECT S.sname
FROM Sailors S, Reserves R
WHERE S.sid=R.sid
      AND R.bid=103
```

Σύνοψη (1)

- ❖ Η Βελτιστοποίηση Αιτημάτων αποτελεί μια σημαντική λειτουργία σε ένα σχεσιακό DBMS.
- ❖ Πρέπει να κατανοούμε τη βελτιστοποίηση ώστε να καταλαβαίνουμε την επίδραση στις επιδόσεις μιας συγκεκριμένης υλοποίησης μιας βάσης (πίνακες, ευρετήρια) πάνω σε ένα υπολογιστικό φόρτο (σύνολο αιτημάτων).
- ❖ Σε ένα αίτημα βελτιστοποιούμε δυο πράγματα:
 - Θεωρούμε ένα σύνολο εναλλακτικών σχεδίων.
 - Κλαδεύουμε τον χώρο αναζήτησης - συνήθως μόνο αριστερού βάθους σχέδια.
 - Πρέπει να εκτιμήσουμε το κόστος του κάθε εξεταζόμενου σχεδίου.
 - Πρέπει να υπολογίσουμε το μέγεθος του αποτελέσματος και το κόστος του κάθε κόμβου του σχεδίου.
 - *Θέματα κλειδιά*: Στατιστικά, ευρετήρια, υλοποιήσεις πράξεων.

Σύνοψη (2)

❖ Αιτήματα ενός πίνακα:

- Θεωρούμε όλες τις διαδρομές πρόσβασης και επιλέγουμε τη φθηνότερη.
- *Θέματα*: Επιλογές που *ταιριάζουν* με ευρετήριο, αν το ευρετήριο έχει όλα τα απαιτούμενα πεδία και/ή επιστρέφει τις εγγραφές με επιθυμητή σειρά.

❖ Αιτήματα πολλών πινάκων:

- Πρώτα απαριθμούνται όλα τα σχέδια ενός πίνακα.
 - Επιλογές/Προβολές εξετάζονται όσο το δυνατό νωρίτερα.
- Μετά, για κάθε σχέδιο με ένα πίνακα, εξετάζονται όλοι οι τρόποι σύζευξης με έναν άλλο πίνακα (ως εσωτερικό).
- Μετά, για κάθε σχέδιο με δυο πίνακες που “διατηρείται”, εξετάζονται όλοι οι τρόποι σύζευξης με άλλο πίνακα (ως εσωτερικό), κλπ.
- Σε κάθε επίπεδο, για κάθε υποσύνολο πινάκων, “διατηρείται” μόνο το καλύτερο σχέδιο για κάθε ενδιαφέρουσα σειρά των εγγραφών.