Υπολογισμός Σχεσιακών Πράξεων: Άλλες Τεχνικές

Κεφάλαιο 14 (μέρος Β)

Απλές Επιλογές

```
SELECT  *
FROM    Reserves R
WHERE   R.rname < 'C%'
```

- Είναι της μορφής $\sigma_{R.attr\ op\ value}(R)$
- Το μέγεθος του αποτελέσματος προσεγγίζεται ως *μέγεθος του R * συντελεστή μείωσης.* Αργότερα, θα δούμε πως εκτιμάμε τους συντελεστές αυτούς.
- Χωρίς ευρετήριο, μη-ταξινομημένος: Ουσιαστικά πρέπει να σαρωθεί όλος ο πίνακας. Το κόστος είναι M (#σελίδων του R).
- Με ευρετήριο στο πεδίο επιλογής: Χρησιμοποιούμε το ευρετήριο για να βρούμε τις κατάλληλες καταχωρίσεις δεδομένων και μετά ανακτούμε τις αντίστοιχες εγγραφές.

Χρήση Ευρετηρίου για τις Επιλογές

- Το κόστος εξαρτάται από το #εγγραφών του αποτελέσματος και τη συγκρότηση.
  - Το κόστος εύρεσης των καταχωρίσεων δεδομένων του αποτελέσματος (συνήθως μικρό) συν το κόστος ανάκτησης των εγγραφών (μπορεί να είναι μεγάλο χωρίς συγκρότηση).
  - Στο παράδειγμα, υποθέτοντας ομοιόμορφη κατανομή ονομάτων, περίπου 10% των εγγραφών θα είναι στο αποτέλεσμα (100 σελίδες, 10000 εγγραφές). Με συγκροτημένο ευρετήριο, το κόστος είναι λίγο περισσότερο από 100 I/O. Με μη-συγκροτημένο, μέχρι 10000 I/O!
- *Σημαντική βελτίωση για μη-συγκροτημένα ευρετήρια*:
  1. Βρες τις καταχωρίσεις δεδομένων του αποτελέσματος.
  2. Ταξινόμησε τα rid των εγγραφών που θα ανακτηθούν.
  3. Ανάκτησε τα rids με τη σειρά. Έτσι, κάθε σελίδα δεδομένων κλειδώνεται μόνο μια φορά (παρόλο που το # τέτοιων σελίδων είναι πιθανό να είναι υψηλότερο από ότι όταν έχουμε συγκρότηση).

Δυο προσεγγίσεις για Γενικές Επιλογές

- Πρώτη προσέγγιση: Βρες την *πιο επιλεκτική διαδρομή πρόσβασης*, ανάκτησε τις εγγραφές με τη βοήθειά της, και εφάρμοσε τους υπόλοιπους όρους που δεν ταιριάζουν με το ευρετήριο:
  - *Πιο επιλεκτική διαδρομή πρόσβασης:* Ένα ευρετήριο ή μια σάρωση αρχείου που εκτιμούμε ότι θα απαιτήσει τις λιγότερες I/O.
  - Όροι που ταιριάζουν με ευρετήριο ελλατώνουν το πλήθος των εγγραφών που *ανακτώνται.* Οι υπόλοιποι όροι χρησιμοποιούνται για να απορρίψουν μερικές από τις ανακτημένες εγγραφές, αλλά δεν επηρεάζουν το πλήθος των εγγραφών/σελίδων που ανακτώνται.
  - Θεωρείστε το *day<8/9/94 AND bid=5 AND sid=3*. Μπορεί να χρησιμοποιηθεί ένα B+tree στο *day.* Μετά για κάθε ανακτημένη εγγραφή πρέπει να ελεγχθεί το *bid=5* and *sid=3*. Παρόμοια, μπορεί να χρησιμοποιηθεί ένα ευρετήριο κατακερματισμού στο <*bid, sid*>. Μετά, πρέπει να ελεγχθεί το *day<8/9/94*.

Τομή συνόλων από rid

- Δεύτερη προσέγγιση (Αν έχουμε 2 ή περισσότερα ταιριαστά ευρετήρια Εναλλακτικών (2) ή (3) για τις καταχωρίσεις δεδομένων):
  - Πάρε τα σύνολα των rid των εγγραφών για κάθε ταιριαστό ευρετήριο.
  - Μετά πάρε την *τομή* αυτών των συνόλων από rid (θα μιλήσουμε για την τομή σύντομα!)
  - Ανάκτησε τις εγγραφές και εφάρμοσε τους υπόλοιπους όρους.
  - Θεωρείστε το *day<8/9/94 AND bid=5 AND sid=3.* Αν έχουμε ένα B+tree στο *day* και ένα άλλο ευρετήριο στο *sid*, και τα δυο Εναλλακτικής (2), μπορούμε με το πρώτο να ανακτήσουμε τα rid των εγγραφών που ικανοποιούν το *day<8/9/94*, με το δεύτερο τα rid των εγγραφών που ικανοποιούν το *sid=3*, να πάρουμε την τομή, να ανακτήσουμε τις εγγραφές και να ελέγξουμε το *bid=5.*

Η πράξη της Προβολής

```
SELECT   DISTINCT
         R.sid, R.bid
FROM     Reserves R
```

- Μια προσέγγιση βασισμένη στην ταξινόμηση:
  - Τροποποιούμε το Πέρασμα 0 της εξωτερικής ταξινόμησης για να εξαλείψουμε τα άχρηστα πεδία. Έτσι, παράγονται συρμοί 2B σελίδων, αλλά με εγγραφές μικρότερες από αυτές των πινάκων εισόδου. (Ο λόγος μεγέθους εξαρτάται από το # και το μέγεθος των πεδίων που εξαλείφονται.)
  - Τροποποιούμε τα περάσματα συγχώνευσης για να εξαλείψουμε τα διπλότυπα. Έτσι, το πλήθος των εγγραφών του αποτελέσματος είναι μικρότερο από την είσοδο. (Η διαφορά εξαρτάται από το # των διπλοτύπων.)
  - Κόστος: Στο Πέρασμα 0, διαβάζουμε τον αρχικό πίνακα (μεγέθους M), και γράφουμε στην έξοδο ίδιο πλήθος μικρότερων εγγραφών. Στα περάσματα συγχώνευσης, γράφονται στην έξοδο λιγότερες εγγραφές σε κάθε πέρασμα. Στο παράδειγμα με τον Reserves, οι 1000 σελίδες εισόδου ελλατώνονται σε 250 στο Πέρασμα 0 αν ο λόγος μεγέθους είναι 0.25

Προβολή με Κατακερματισμό

- *Φάση Διαμερισμού*: Διάβασε τον R χρησιμοποιώντας ένα πλαίσιο εισόδου. Για κάθε εγγραφή, εξάλειψε τα άχρηστα πεδία, και εφάρμοσε τη συνάρτηση κατακερματισμού *h1* για να επιλέξεις ένα από τα B-1 πλαίσια εξόδου.
  - Το αποτέλεσμα είναι B-1 διαμερίσματα (εγγραφών χωρίς άχρηστα πεδία). 2 εγγραφές από διαφορετικά διαμερίσματα είναι εγγυημένα διαφορετικές.
- *Φάση εξάλειψης διπλότυπων*: Για κάθε διαμέρισμα, διάβασέ το και κατασκεύασε στη μνήμη έναν πίνακα κατακερματισμού, χρησιμοποιώντας τη συνάρτηση *h2* (<> *h1*) σε όλα τα πεδία, ενώ εξαλείφεις τα διπλότυπα.
  - Αν το διαμέρισμα δε χωρά στη μνήμη, μπορούμε να εφαρμόσουμε τον ίδιο αλγόριθμο αναδρομικά πάνω στο διαμέρισμα.
- Κόστος: Για το διαμερισμό, διάβασε τον R, και γράψε κάθε εγγραφή, αλλά με λιγότερα πεδία. Αυτές είναι οι εγγραφές που διαβάζονται στην επόμενη φάση.

Συζήτηση της Προβολής

- Πιο συνηθισμένη είναι η προσέγγιση με την ταξινόμηση. Δουλεύει καλύτερα με μη-ομοιόμορφα δεδομένα και το αποτέλεσμα είναι ταξινομημένο.
- Αν ένα ευρετήριο πίνακα περιέχει όλα τα επιθυμητά πεδία στο κλειδί αναζήτησής του, μπορούμε να σαρώσουμε *με ευρετήριο-μόνο*.
  - Εφαρμόζουμε τις τεχνικές προβολής στις καταχωρίσεις δεδομένων (πολύ μικρότερες!)
- Αν ένα ταξινομημένο ευρετήριο (π.χ., δεντρικό) περιέχει όλα τα επιθυμητά πεδία ως *πρόθεμα* του κλειδιού αναζήτησης, ακόμα καλύτερα:
  - Ανάκτησε όλες τις καταχωρίσεις δεδομένων με τη σειρά (σάρωση με ευρετήριο-μόνο), εξάλειψε τα άχρηστα πεδία, σύγκρινε τις γειτονικές εγγραφές για έλεγχο διπλοτύπων.

Πράξεις Συνόλων

- Η Τομή και το Γινόμενο είναι ειδικές περιπτώσεις Σύζευξης.
- Το ίδιο η Ένωση (Distinct) και η Διαφορά. Ας δούμε την Ένωση.
- Ένωση με ταξινόμηση:
  - Ταξινόμησε και τους δυο πίνακες (στο συνδυασμό όλων των πεδίων).
  - Σάρωσε τους ταξινομημένους πίνακες και συγχώνευσέ τους.
  - *Εναλλακτικά*: Συγχώνευσε συρμούς από το Πέρασμα 0 και για τους *δυο* πίνακες.
- Ένωση με κατακερματισμό:
  - Διαμέρισε τους R και S με χρήση της συνάρτησης *h*.
  - Για κάθε διαμέρισμα του S, κατασκεύασε στη μνήμη πίνακα κατακερματισμού (με χρήση της *h2*), σάρωσε αντίστοιχο διαμέρισμα του R και πρόσθεσε εγγραφές στον πίνακα εξαλείφοντας τα διπλότυπα.

Πράξεις Συνάθροισης (AVG, MIN, κλπ.)

- Χωρίς ομαδοποίηση:
  - Γενικά, απαιτεί σάρωση του πίνακα.
  - Αν υπάρχει ευρετήριο με κλειδί αναζήτησης που περιέχει όλα τα πεδία του SELECT ή του WHERE, μπορούμε να σαρώσουμε με ευρετήριο-μόνο.
- Με ομαδοποίηση:
  - Ταξινόμησε πάνω στα πεδία του group-by, μετά σάρωσε τον πίνακα και υπολόγισε τη συνάθροιση για κάθε ομάδα. (Βελτίωση: συνδύασε την ταξινόμηση και τον υπολογισμό της συνάθροισης.)
  - Παρόμοια προσέγγιση: με κατακερματισμό στα πεδία του group-by.
  - Αν υπάρχει δεντρικό ευρετήριο με το κλειδί αναζήτησης να περιέχει όλα τα πεδία των SELECT, WHERE και GROUP BY, μπορούμε να σαρώσουμε με ευρετήριο-μόνο. Αν τα πεδία του group-by σχηματίζουν πρόθεμα του κλειδιού αναζήτησης, μπορούμε να ανακτήσουμε τις καταχωρίσεις δεδομένων/εγγραφές με τη σειρά του group-by.

Επίδραση της Ενδιάμεσης Μνήμης

- Αν υπάρχει ταυτόχρονη εκτέλεση πολλών πράξεων, η εκτίμηση του αριθμού των διαθέσιμων πλαισίων γίνεται στην τύχη.
- Η επανάληψη μοτίβων πρόσβασης αλληλεπιδρά με τη στρατηγική αντικατάστασης του Διαχειριστή Ενδιάμεσης Μνήμης.
  - π.χ., στη Σύζευξη με απλή εμφώλευση βρόχων γίνεται επαναληπτική σάρωση του εσωτερικού πίνακα. Με αρκετά πλαίσια για τον εσωτερικό πίνακα, δεν παίζει ρόλο η στρατηγική αντικατάστασης. Αλλιώς, η MRU είναι η καλύτερη, η LRU η χειρότε*ρη (σειριακή πλημ*μύρα).
  - Παίζει ρόλο η στρατηγική αντακατάστασης για τη σύζευξη με εμφώλευση βρόχων κατά μπλοκ;
  - Τί ισχύει για την εμφώλευση βρόχων με ευρετήριο; Σύζευξη με ταξινόμηση και συγχώνευση;

Μερικές Κοινές Τεχνικές

- Οι αλγόριθμοι υπολογισμού σχεσιακών πράξεων χρησιμοποιούν κατά κόρον μερικές απλές ιδέες:
  - Ευρετηριοποίηση: Χρησιμοποιούνται οι συνθήκες του WHERE ώστε να ανακτηθεί μικρό σύνολο των εγγραφών (επιλογές, συζεύξεις)
  - Επανάληψη: Μερικές φορές είναι ταχύτερο να σαρωθούν όλες οι εγγραφές ακόμη κι αν υπάρχει ευρετήριο. (Και μερικές φορές, μπορούν να σαρωθούν οι καταχωρίσεις δεδομένων σε ένα ευρετήριο αντί των εγγραφών στο πίνακα.)
  - Διαμερισμός: Χρησιμοποιώντας ταξινόμηση ή κατακερματισμό, μπορούμε να διαμερίσουμε τις εγγραφές εισόδου και να αντικαταστήσουμε μια δαπανηρή πράξη από παρόμοιες πράξεις σε μικρότερα υποσύνολα της εισόδου.

* *Θα δούμε αυτές τις τεχνικές στον υπολογισμό αιτημάτων!*

Σύνοψη

- Ένα πλεονέκτημα των σχεσιακών DBMS: *τα αιτήματα είναι σύνθεση μερικών βασικών πράξεων.* Η υλοποίηση αυτών των πράξεων μπορεί να ρυθμιστεί προσεκτικά (και αυτό είναι σημαντικό!).
- Υπάρχουν πολλές εναλλακτικές τεχνικές υλοποίησης της κάθε πράξης. Για τις περισσότερες πράξεις δεν υπάρχει κάποια τεχνική που να είναι η καλύτερη για κάθε περίπτωση.
- Για κάθε πράξη ενός αιτήματος, πρέπει να μελετηθούν οι διαθέσιμες εναλλακτικές και να επιλεγεί η καλύτερη βάσει στατιστικών του συστήματος κλπ. Αυτό αποτελεί μέρος της ευρύτερης διαδικασίας βελτιστοποίησης ενός αιτήματος που αποτελείται από πολλές πράξεις.