Class 18 : Joins I

Summary #pages M $\binom{N}{P_S}$
#rows per page P_R

S (sid, sname, rating, age)
N = 500 P_S = 80

R (sid, bid, day, rname)
M = 1000 P_R = 100

Selection (a) i) cheapest access path
ii) retrieve tuples
iii) apply remaining selection conditions

(b) i) get rids from all matching indexes
ii) intersection rids
iii) retrieve tuples & apply remaining sel. cond.

unsorted M I/Os
sorted $\log_2 M + f \cdot M$
clustered $\log_F M + f \cdot M$
unclustered $\log_F M + f \cdot M \cdot P_R$

Projection
Sort + discard unwanted fields & duplicates
hash + cc

cost : $M + 2 \cdot T$ ← pages after removing unwanted fields

---

Joins : Nested-Loop Joins
Sort-Merge Joins today
Hash Joins
Remaining op (joins + agg)

any interesting query contains a join

SELECT * FROM R, S WHERE R.sid = S.sid

$R \bowtie S$ discuss as # I/Os discard output

$\sigma_{R.sid=S.sid}(R \times S)$

- Simple Nested-Loop Join $R \underset{i=j}{\bowtie} S$

$\forall r \in R$ ← outer

$\forall s \in S$ ← inner

if $r_i == s_j$ then add $\langle r, s \rangle$ to the result

Cost

$(M \cdot p_R) \cdot N + M = (1000 \cdot 100) \cdot 500 + 1000 = 50{,}001{,}000$ # I/Os

28h

rows of R (!!)

1 I/O → 2ms

R M = 1000 → 4MB

S N = 500 → 2MB

swap R with S

$(N \cdot p_S) \cdot M + N = 40{,}000{,}500$ IOs

- Page-oriented Nested-Loop Join

$\forall$ page b_r in R

$\forall$ page b_s in S

$\forall$ tuple r in b_r

$\forall$ tuple s in b_s

if $r_i == s_j$ then add $\langle r, s \rangle$ to the result

Cost

$M \cdot N + M = 1000 \cdot 500 + 1000 = 501,000 \rightarrow 17 \text{ min}$

smaller outer?

$N \cdot M + N = 500 \cdot 1000 + 500 = 500,500$

- Index Nested Loop Join

∀ tuple r in R
  probe index to fetch s such that $s_i == r_j$
    add ⟨r, s⟩ to result

Cost

$M + M \cdot P_R$ · cost of finding matching tuples through the index
  → Hash index 1.2 IOs
  → B+-Tree 2-4 IOs

clustered → 1 I/O per page of matching tuples
unclustered → 1 I/O per matching tuple

Example 1: hash idx on sid of S

Scan R: (M)
  ∀ each tuple in R
    fetch data entry (1.2)
      goto file (1)

$M + M \cdot P_R \cdot (1.2+1) \rightarrow^{1000+} 1000 \cdot 100 (2.2) = 221,000 \rightarrow 7 \text{ min}$

Example 2: hash idx on sid of R

$\frac{M \cdot P_R}{N \cdot P_S} = \frac{100k}{40k} = 2.5 \frac{res}{sailor}$

Scan S (N)

∀ s probe hash idx (1.2)

find matching tuples 2.5

$N = N \cdot P_S (1.2 + 2.5) = 500 + 500 \cdot 80 \cdot (3.7) = \boxed{148,500} \rightarrow \boxed{5 \text{ min}}$

- Bloar Nested Loop Joins

→ 1 page for streaming the inner S

1 page for output

k pages for holding bloars (of k) of outer R

∀ block of k pages of R

∀ page bs in S

∀ tuple r in k pages of R

∀ tuple s in bs

if $r_i == s_j$ add ⟨r, s⟩ to the result

Cost

Scan outer R: M I/Os

Scan inner for each bloar of R

$M + \frac{M}{k} \cdot N \rightarrow 1000 + \frac{500 \cdot 1000}{k}$ k = 100 pages

→ 6000 I/Os 12sec

$N + \frac{N}{k} \cdot M \rightarrow 500 + \frac{500 \cdot 1000}{k} \rightarrow 5500 \rightarrow 11\text{sec}$

• Sort – Merge Join

→ both sorted on the join attribute

useful: ① both or one relations sorted on join attr.

② output should be sorted on join attr.

→ many duplicates may lead to backtracking

Cost: Sort R + Sort S + M + N

worst case? M·N if all is equal

cost: $(M+N) \cdot 2 \cdot \#passes + M + N$

2 passes?

$$\left\lceil \frac{N}{B} \right\rceil = B-1 \approx \frac{N}{B} = B-1 \Rightarrow B^2 - B - N = 0$$

$$B \approx \lceil \sqrt{N} \rceil + 1 = 33$$

$$Cost = (M+N) \cdot 5 = 1500 \cdot 5 = \boxed{7500} \text{ I/Os} \rightarrow \boxed{15 \text{ sec}}$$

BNLJ w/ 33 buffers $M + \frac{M \cdot N}{K} = 1000 + \frac{500 \cdot 1000}{33} \approx \boxed{1000 + 15151}$

$N + \frac{M \cdot N}{K} = 500 + \frac{500 \cdot 1000}{33} \approx \boxed{500 + 15151}$

if K=100 SMJ cannot do better than 7500

BNLJ will do as low as 5500

* Refined Sort-Merge Join

assume $B > \sqrt{M}$ and $B > \sqrt{N}$

after pass 0

$R \rightarrow \frac{M}{B}$ runs $\quad B > \sqrt{M} \Rightarrow \frac{1}{B} < \frac{1}{\sqrt{M}} \Rightarrow \frac{M}{B} < \sqrt{M} < B$

$S \rightarrow \frac{N}{B}$ runs $\quad B > \sqrt{N} \Rightarrow \frac{N}{B} < \sqrt{N} < B$

after pass 0 either R, S # runs $< B$

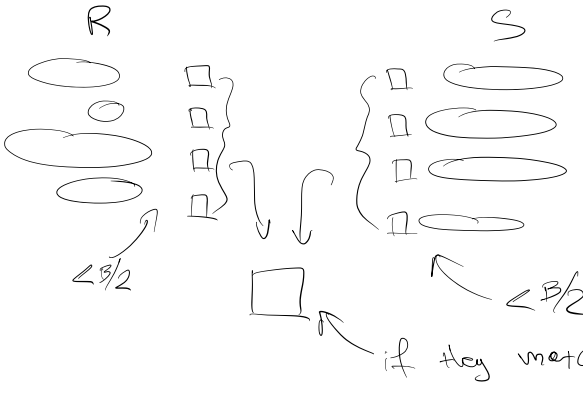
consider using replacement sort

it results to runs with size $\simeq 2B$

sorted runs after pass 0 using replacement sort

$R \rightarrow \frac{M}{2B} < \frac{B}{2} \qquad S \rightarrow \frac{N}{2B} < B/2$

we allocate a buffer per sorted run per file



$cost = (M+N) \cdot 3$

Read R → writing $< B/2$ # runs of R $\quad 2 \cdot M$

Read S → writing $< B/2$ # runs of S $\quad 2 \cdot N$

Read R and S and merge on the fly : $M+N$

$(M+N) \cdot 3 = \boxed{4500}$ I/Os → 9s