Duy Truong

100

13.4

+ The relation resulting from the join of r_1, r_2 and r_3 will be the same no matter which way they're joined, due to the associative and commutative properties of joins. So we will consider the size base on the strategy of $((r_1 \bowtie r_2) \bowtie r_3)$.

We join r_1 and r_2 will yield a relation of at most 1000 tuples, since C is a key for r_2. Likewise, joining that result with r_3 will yield the relation most 1000 because E is a key for r_3. So the final relation will have most 1000 tuples

+ effecient stratagy:

creating index on attribute C for r_2 and on E for r_3. And for each tuple for r_1.

1. look up at most one tuple which matches the C value of r_1 by using index of r_2

2. look up in r_3 at most one tuple which matches the unique value for E in r_2 by using index on E

size of the relation :

each r_1, $\frac{1500}{V(C, r_2)} = \frac{15}{11}$ tuples of r_2 would join with r_1

The intermediate relation would have 15000/11 tuples.

The result of relation join with $r_3 = \frac{15000}{11} \times \frac{750}{100}$

$= 10227$

A good strategy should join r_1 and r_2 first since the intermediate relation is about the same size as r_1 and r_2. Then r_1 will be with result

Sort r and collect the top k tuple The tuples contained in

r ⋈ s since the join is on a foreigin key of r refereing s

b. Execute r ⋈ s using standard join algorithm until the frist k results have been found. After k tuples have been computed in the result set. we exeute the join but discard any tuples from r that have attribute values less than all of the tuples in the result set. if the new tuples t has an attribute value bigger than at least on of the tuples in the result set, replace the lowest-valued tuple in the result set with t.

y 1.

Using hashidex.

we have

$Cost = 1.2 \times 4 = 4.8$.

Query 2.

The number of pointer /node $= \frac{4096 - 4}{20 + 4} + 1 = 171$

The height of B + tree $= \log_{\frac{171}{2}} 100 = 1$

No. of qualifying tuples $= 1000000 * \frac{1}{100} = 10000$

~~No. of leaf cost = 10000 + 1 = 10001.~~

~~No of leaf pages = $1000000 / \left(\frac{171 - 1}{2}\right) * \frac{1}{100} = 117$~~

1

$Cost = 10000 + \cancel{117} = 10001$

Q3.

$$\# \text{ tuple/node} = \frac{4096}{60} = 69$$

The number of pointer/node $= \dfrac{4096 - 4}{20+20+4} + 1 = 94.$

The heigh of B+tree $= \log_{\frac{94}{2}} 10000 = 2.$

No of qualifying tuples $= 10000 * \dfrac{1}{10} = 1000.$

No of leaf pages $= \dfrac{10000}{69} * \dfrac{1}{100} = 15$

$\Rightarrow$ cost $= 15 + 2$
$= 17$

query 4:

using the index on <VID, Date> so the cost = 17

same with Q3.