

## Discussion 05/27

### Discussion 12-1

Two queries  $Q$  and  $Q'$  on a DB schema  $S$  are equivalent if the answer sets of  $Q$  and  $Q'$  are the same in any instances of the DB.

Discuss the implications of changing the underlined expression to 'any legal instances of the DB'.

Legal instance 는 integrity constraints 를 만족하는 instance 들...

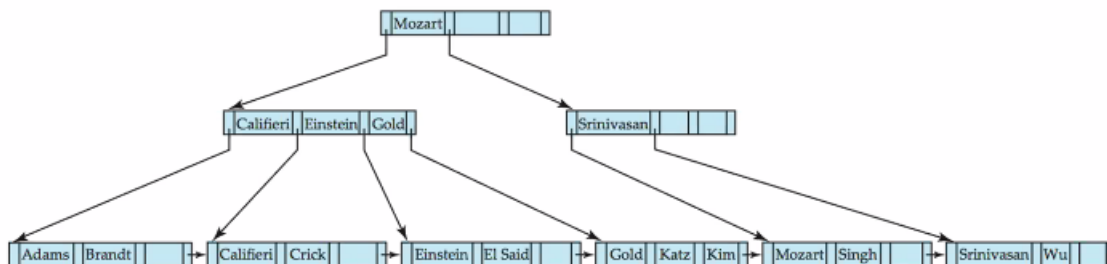
Schema 를 정의하고 record 를 insert, update, delete 하는 과정에서 모든 instance 가 legal 이지 않을까? 그러면 legal instance 만 확인해도 될 것 같고 효율성이 더 높아질 것 같다.

=> query processing 을 하는 시점에서 legal 인지 아닌지의 여부는 이미 판단이 되어 있음. Legal instance 에서 equivalent 한 경우는 훨씬 많아짐. 그래서 선택지가 더 많아지기 때문에 더 많은 선택지 중 cost 가 싼 걸 고르면 됨.

### Discussion 12-2

Suppose the following B+-tree index on *Name* is primary and nonkey. Estimate the number of seeks and block transfers needed to answer the following queries.

- a) **select \* from instructors where name = 'Kim'**
- b) **select \* from instructors where name like 'K%'**
- c) **select \* from instructors where name >= 'Kim'**
- d) **select \* from instructors where name != 'Kim'**



a) 높이가 3 이므로.. estimated seek 은 4 번. Block transfer 는 3 + n 번 . n 은 kim 을 찾는 데 필요한 block 들의 개수.

b) seek 은 3 번. K 로 시작하는 record 가 x 개라고 하면 block transfer 는  $3 + x$  번.

c) seek 는 3 번. KIM 보다 큰 record 가 x 개면 block transfer 는  $3 + X$  번.

=> 확실하지 않음..

## Discussion 12-3

Suppose the *students* table has 20,000 records stored in 1,000 blocks with a primary B+-tree index on *ID* (an index block can hold up to 100 entries).

Estimate the disk access time required to answer the following query, where  $t_T = 10\text{ms}$  and  $t_S = 20\text{ms}$ .

```
select * from students  
where ID = '12121'
```

B+트리의 높이는  $\log_{50}(20000)$  이니까 3. Primary index 이고 key 이기 때문에

Cost =  $(4) * (10\text{ms} + 20\text{ms}) = 120\text{ms}$ .

## Discussion 12-4

Suppose the *students* table has 20,000 records stored in 1,000 blocks with a primary B+-tree index on *ID* (an index block can hold up to 100 entries).

Estimate the disk access time required to answer the following query, where  $t_T = 10\text{ms}$  and  $t_S = 20\text{ms}$ .

```
select * from students  
where ID between '12121' and '42424'
```

위의 문제와 같고, seek 는 12121 에 대해 index 로 수행하면 4 번. 여기서부터 순차적으로 relation 을 스캔하니까 seek 는 여기서 끝. Block transfer 는 트리에서 3 번 + 42424 까지의 개수를 n 이라고 하면 n 개를 돌아야 하므로  $3 + n$ .

Cost =  $4 * 10\text{ms} + (3 + n) * 20\text{ms}$ .

=> n 을 어떻게 estimate 할 것인가? # of distinct value 를 알고 있으면 좋음. 그리고 전체 구간 대비 해당 구간의 비율을 알면 2 만개의 record 에서 비율을 구해서 예상 개수를 구함.

## Discussion 12-5

Perform an *external sort-merge* operation on the relation  $r$  based on the first attribute (numeric). Assume  $M = 3$ .

Extra) Estimate the number of block transfers and the number of seeks.

relation  $r$

3	A
10	B
8	C
6	D
1	E
9	F
8	G
2	H
7	I
2	J
4	K
5	L
3	M
6	N
9	O
1	P

1. 처음에 run 을 만들면

3 개의 block 이 되니까 (1E 3A 6D 8C 9F 10B), (2H 2J 4K 5L 7I 8G), (1P 3M 6N 9O)의 run 들이 만들어짐.

2. N 이 3 이므로 merge pass.

(1E 3A 6D 8C 9F 10B), (2H 2J 4K 5L 7I 8G) 둘이 합쳐져서 (1E 2H 2J 3A 4K 5L 6D 7I 8C 8G 10B)가 되고, 뒤에꺼 그대로 (1P 3M 6N 9O). 그리고 이 둘이 합쳐져서 최종적으로 (1E 1P 2H 2J 3A 3M 4K 5L 6D 6N 7I 8C 8G 9O 10B)가 됨.

Block transfer 는  $b_r (2^{\lceil \log_{M-1}(b_r/M) \rceil} + 1)$  . 즉  $b_r * (2 * \text{pass} + 1)$  인데  $b_r = 8$ ,  $\text{pass} = 2$  이므로 40.

Seek 는  $2^{\lceil b_r/M \rceil} + \lceil b_r/b_b \rceil (2^{\lceil \log_{M-1}(b_r/M) \rceil} - 1)$  .  $2 * [8/3] + [8/1] * (3) = 6 + 24 = 30$ .  $b_b$  는 1.