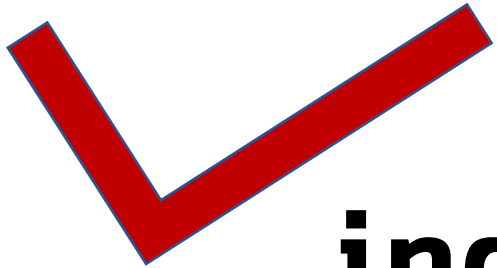


이럴때는



index fast full scan 으로 유도하자 !

1. index fast full scan 이란 ?

index fast full scan은 인덱스 트리 구조를 무시하고 인덱스 세그먼트 전체를 multiblock i/o 방식으로 스캔하는 스캔방법 입니다.

	인덱스 액세스 방법	관련 힌트
1	index range scan	index
2	index unique scan	index
3	index full scan	index
4	index skip scan	index_ss
→ 5	index fast full scan	index_ffs
6	index merge scan	and_equal
7	index bitmap merge scan	index_combine

```
select /*+ gather_plan_statistics */ job, count(*)  
from emp  
group by job;
```

full table scan !

emp 테이블

인덱스가 없다면?

ROWID	EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
AAATc1AAHAAAAHeAAA	7839	KING	PRESIDENT		1981-11-17	5000		10
AAATc1AAHAAAAHeAAB	7698	BLAKE	MANAGER	7839	1981-05-01	2850		30
AAATc1AAHAAAAHeAAC	7782	CLARK	MANAGER	7839	1981-05-09	2450		10
AAATc1AAHAAAAHeAAD	7566	JONES	MANAGER	7839	1981-04-01	2975		20
AAATc1AAHAAAAHeAAE	7654	MARTIN	SALESMAN	7698	1981-09-10	1250	1400	30
AAATc1AAHAAAAHeAAF	7499	ALLEN	SALESMAN	7698	1981-02-11	1600	300	30
AAATc1AAHAAAAHeAAG	7844	TURNER	SALESMAN	7698	1981-08-21	1500	0	30
AAATc1AAHAAAAHeAAH	7900	JAMES	CLERK	7698	1981-12-11	950		30
AAATc1AAHAAAAHeAAI	7521	WARD	SALESMAN	7698	1981-02-23	1250	500	30
AAATc1AAHAAAAHeAAJ	7902	FORD	ANALYST	7566	1981-12-11	3000		20
AAATc1AAHAAAAHeAAK	7369	SMITH	CLERK	7902	1980-12-09	800		20
AAATc1AAHAAAAHeAAL	7788	SCOTT	ANALYST	7566	1982-12-22	3000		20
AAATc1AAHAAAAHeAAM	7876	ADAMS	CLERK	7788	1983-01-15	1100		20
AAATc1AAHAAAAHeAAN	7934	MILLER	CLERK	7782	1982-01-11	1300		10

```
select /*+ index_ffs(emp emp_job) */ job, count(*)
from emp
group by job;
```

직업에 인덱스는 있으나 not null 이 보장되지 않는다면 ?

full table scan !

emp_job 인덱스

JOB	ROWID
ANALYST	AAAXDOAANAAAAHjAAJ
ANALYST	AAAXDOAANAAAAHjAAL
CLERK	AAAXDOAANAAAAHjAAH
CLERK	AAAXDOAANAAAAHjAAK
CLERK	AAAXDOAANAAAAHjAAM
CLERK	AAAXDOAANAAAAHjAAN
MANAGER	AAAXDOAANAAAAHjAAB
MANAGER	AAAXDOAANAAAAHjAAC
MANAGER	AAAXDOAANAAAAHjAAD
PRESIDENT	AAAXDOAANAAAAHjAAA
SALESMAN	AAAXDOAANAAAAHjAAE
SALESMAN	AAAXDOAANAAAAHjAAF
SALESMAN	AAAXDOAANAAAAHjAAG
SALESMAN	AAAXDOAANAAAAHjAAI

emp 테이블

ROWID	EMPNO	ENAME	JOB	MGR	...
AAATc1AAHAAAAHeAAA	7839	KING	PRESIDENT		...
AAATc1AAHAAAAHeAAB	7698	BLAKE	MANAGER	7839	...
AAATc1AAHAAAAHeAAC	7782	CLARK	MANAGER	7839	...
AAATc1AAHAAAAHeAAD	7566	JONES	MANAGER	7839	...
AAATc1AAHAAAAHeAAE	7654	MARTIN	SALESMAN	7698	...
AAATc1AAHAAAAHeAAF	7499	ALLEN	SALESMAN	7698	...
AAATc1AAHAAAAHeAAG	7844	TURNER	SALESMAN	7698	...
AAATc1AAHAAAAHeAAH	7900	JAMES	CLERK	7698	...
AAATc1AAHAAAAHeAAI	7521	WARD	SALESMAN	7698	...
AAATc1AAHAAAAHeAAJ	7902	FORD	ANALYST	7566	...
AAATc1AAHAAAAHeAAK	7369	SMITH	CLERK	7902	...
AAATc1AAHAAAAHeAAL	7788	SCOTT	ANALYST	7566	...
AAATc1AAHAAAAHeAAM	7876	ADAMS	CLERK	7788	...
AAATc1AAHAAAAHeAAN	7934	MILLER	CLERK	7782	...


2. 직업 컬럼에 not null 을 보장하는 방법

- 테이블에 직접 not null 제약을 걸어준다.

```
alter table emp  
  modify job not null;
```

- where 절에 is not null 을 사용한다.

```
select /*+ index_ffs(emp emp_job) */ job, count(*)  
from emp  
where job is not null  
group by job;
```



EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
7839	KING	PRESIDENT		1981-11-17	5000		10
7698	BLAKE	MANAGER	7839	1981-05-01	2850		30
7782	CLARK	MANAGER	7839	1981-05-09	2450		10
7566	JONES	MANAGER	7839	1981-04-01	2975		20
7654	MARTIN	SALESMAN	7698	1981-09-10	1250	1400	30
7499	ALLEN	SALESMAN	7698	1981-02-11	1600	300	30
7844	TURNER	SALESMAN	7698	1981-08-21	1500	0	30
7900	JAMES	CLERK	7698	1981-12-11	950		30
7521	WARD	SALESMAN	7698	1981-02-23	1250	500	30
7902	FORD	ANALYST	7566	1981-12-11	3000		20
7369	SMITH	CLERK	7902	1980-12-09	800		20
7788	SCOTT	ANALYST	7566	1982-12-22	3000		20
7876	ADAMS	CLERK	7788	1983-01-15	1100		20
7934	MILLER	CLERK	7782	1982-01-11	1300		10

```
select /*+ index_ffs(emp emp_job) */ job, count(*)  
from emp  
group by job;
```

직업 컬럼에 not null 이 보장 된다면 ?

index fast full scan !

emp_job 인덱스

JOB	ROWID
ANALYST	AAAXDOAANAAAAHjAAJ
ANALYST	AAAXDOAANAAAAHjAAL
CLERK	AAAXDOAANAAAAHjAAH
CLERK	AAAXDOAANAAAAHjAAK
CLERK	AAAXDOAANAAAAHjAAM
CLERK	AAAXDOAANAAAAHjAAN
MANAGER	AAAXDOAANAAAAHjAAB
MANAGER	AAAXDOAANAAAAHjAAC
MANAGER	AAAXDOAANAAAAHjAAD
PRESIDENT	AAAXDOAANAAAAHjAAA
SALESMAN	AAAXDOAANAAAAHjAAE
SALESMAN	AAAXDOAANAAAAHjAAF
SALESMAN	AAAXDOAANAAAAHjAAG
SALESMAN	AAAXDOAANAAAAHjAAI

결과

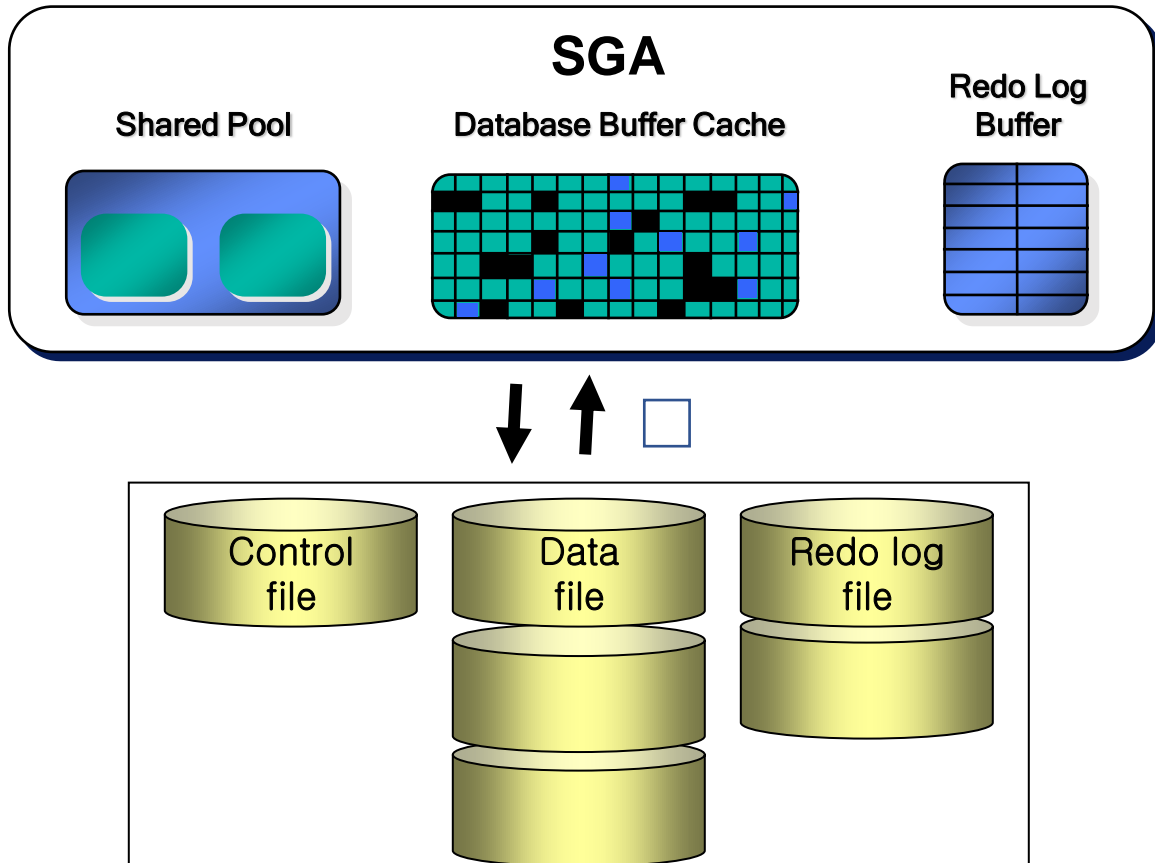
JOB	COUNT(*)
ANALYST	2
CLERK	4
SALESMAN	4
MANAGER	3
PRESIDENT	1

3. index full scan 과 index fast full scan 의 차이

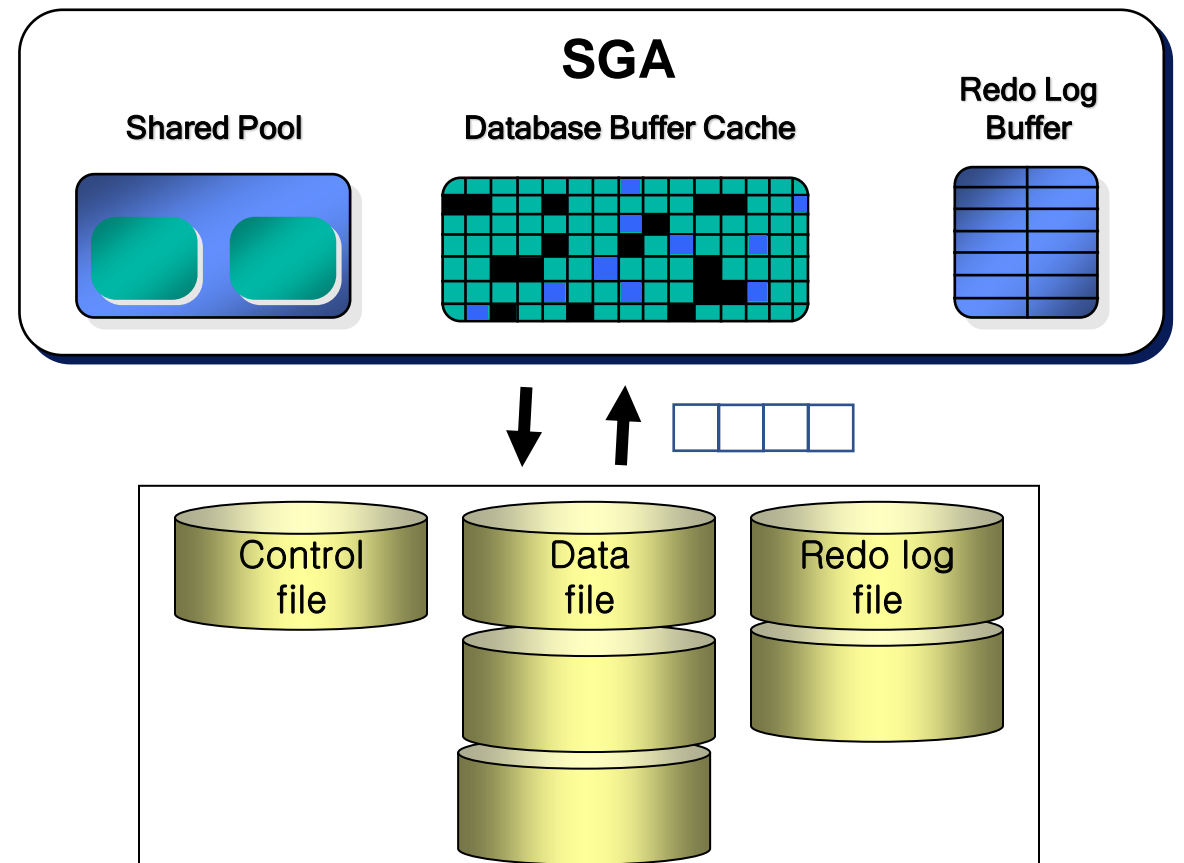
	index full scan	index fast full scan
I/O 방식	single block i/o	multi block i/o
정렬	정렬 보장	정렬 안됨
속도	느림	빠름
병렬읽기	지원 안됨	지원됨

4. single block i/o 와 multi block i/o 의 차이

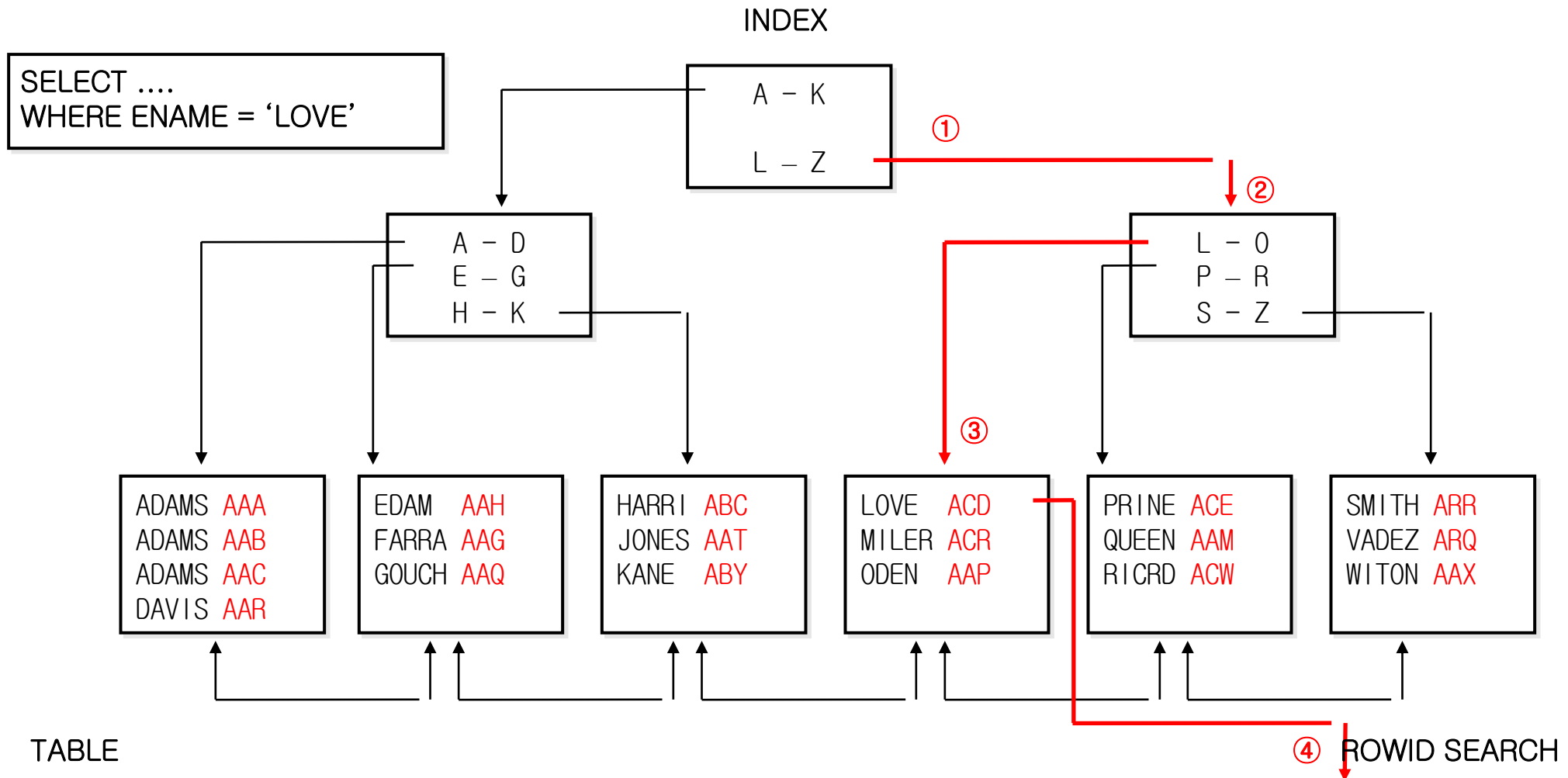
single block i/o



multi block i/o



5. 논리적인 순서에 따라 배치된 index 의 구조

[illegible]

**select /*+
from emp
group by deptno;**

***/ deptno, count(*)**

index_fs(emp emp_deptno) index_ffs(emp emp_deptno)

index full scan !

index fast full scan !

결과

DEPTNO	COUNT(*)
10	3
20	5
30	6

결과

DEPTNO	COUNT(*)
30	6
10	3
20	5