

DBMS-Assignment-4

cs18b055

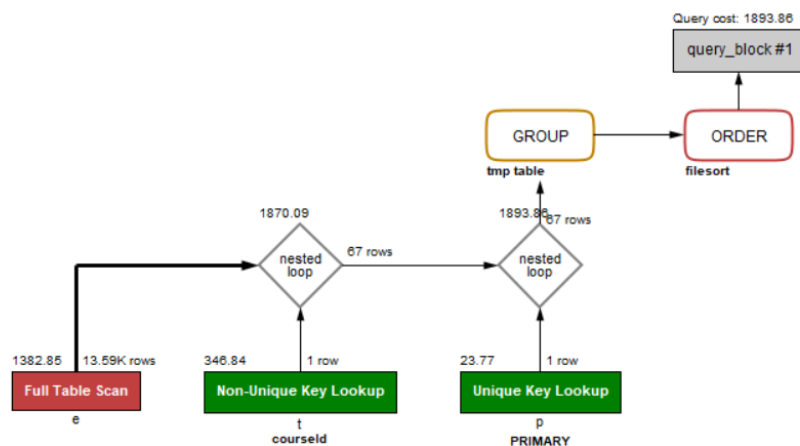
11 November 2020

The following is the query I have choosen

Query

The Number of 'U' grades given by each professor in all the courses he/she taught over all the years in descending order of count of grades.

```
1  use zoo;
2  select p.name, count(e.grade) as 'No of U grades given'
3  from professor p
4  join teaching t
5      on p.empId = t.empId
6  join enrollment e
7      on t.courseId = e.courseId and t.sem = e.sem and t.year = e.year and e.grade = 'U'
8  group by p.empId
9  order by count(e.grade) desc
10 ;
```



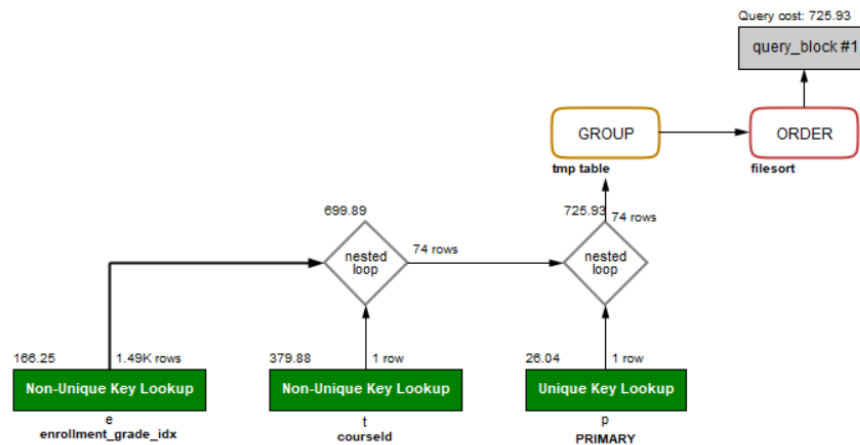
Here my focus is counting No. of 'U' grades , so I thought making an index of enrollment table in which the column 'grade' is sorted might be useful. The following is the code for creating above described index.

```

1  use zoo;
2  create index enrollment_grade_idx
3  on enrollment(grade)
4  ;

```

After adding this index , the query cost has been reduced to 725.93 from 1893.86, which is 61 percent decrease in the time, the index we have added made our query much faster.



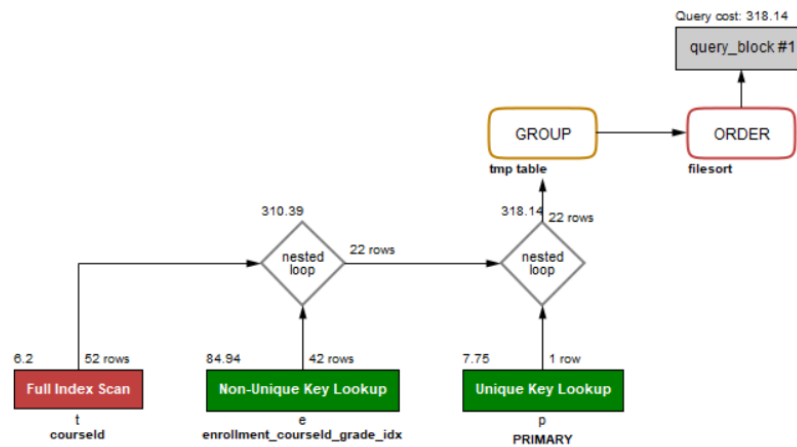
Finally I made one more index including 'courseId' column along with the 'grade' column because we want the grades given to a student in a particular course. The following is the code for creating this index. (I dropped previous index before adding this one).

```

1 use zoo;
2 create index enrollment_courseId_grade_idx
3 on enrollment(courseId, grade)
4 ;

```

After adding this index to the table enrollment , the query cost reduced to 318.14 from 1893.86. This is 83 percent reduction in the query processing time.



Although index take lots of space in table and computing an index takes a bit long time , but it is one time task only , once the index is ready we can use it any number of times and the subsequent queries which uses these indexes becomes faster.