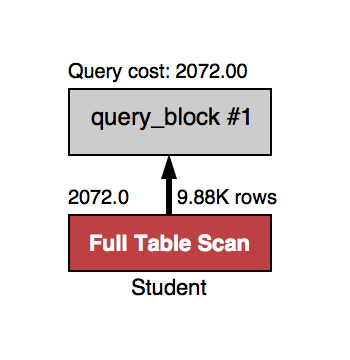
1.List the name of the student with id equal to v1 (id).

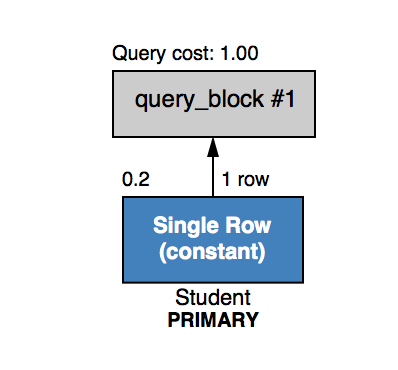
(Before Tuning) SELECT name FROM Student WHERE id = 995293;

Takes 0.0034 to 0.0060 sec to execute.



(After Tuning) SELECT name FROM Student WHERE id = 995293;

Takes 0.00023 to 0.00034 sec to execute.

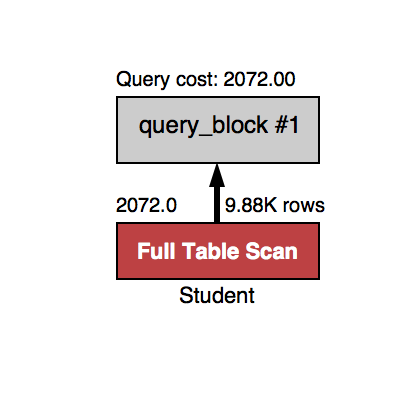


We improved the query by adding id for Student as a primary key.

2.List the names of students with id in the range of v2 (id) to v3 (inclusive).

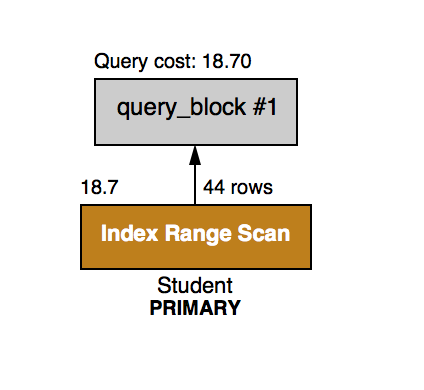
(Before Tuning) SELECT name FROM Student WHERE id >= 131 AND id <= 5379;

Takes 0.0037 to 0.0039 sec to execute.



(After Tuning) SELECT name FROM Student WHERE id BETWEEN 131 AND 5379;

Takes 0.00029 to 0.00035 sec to execute.



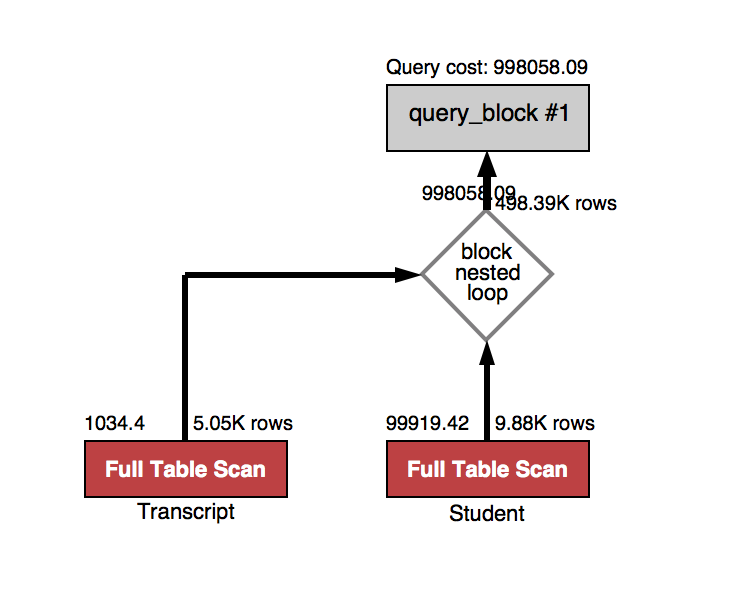
We improved this index by adding id as a primary key for student, studName as an index for name from Student and using BETWEEN for the id intervals.

3.List the names of students who have taken course v4 (crsCode).

(Before Tuning) SELECT Student.name FROM Student, Transcript WHERE Student.id =

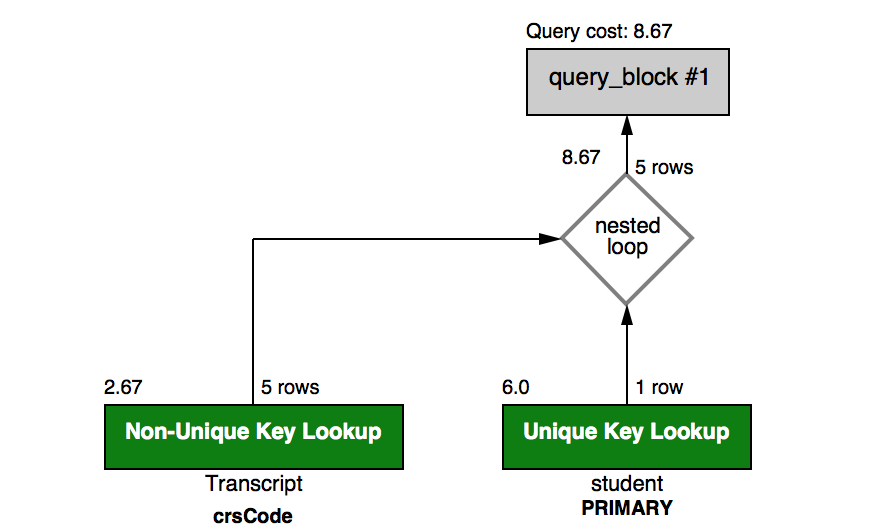
Transcript.studId AND Transcript.crsCode = 'crsCode579449';

Takes 0.0083 to 0.0091 sec to execute.



(After Tuning) SELECT name FROM student INNER JOIN (SELECT studId FROM Transcript WHERE crsCode = 'crsCode579449') AS t ON id = t.studId;

Takes 0.00035 to 0.00041 sec to execute.



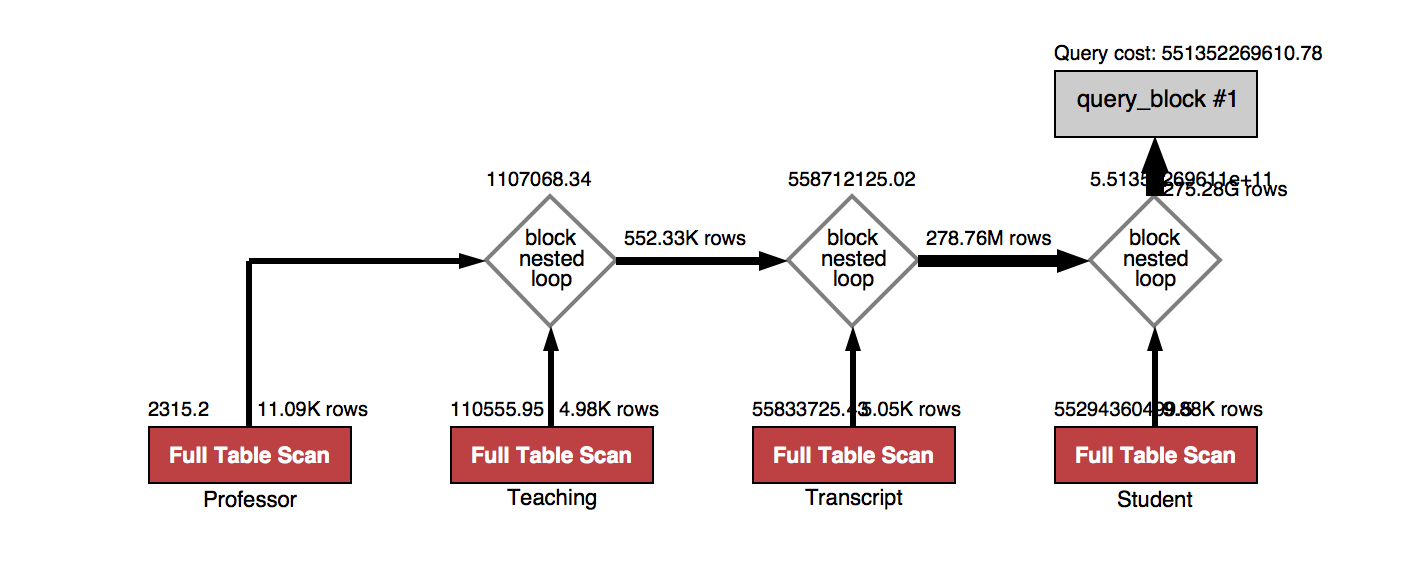
We improved this query by making id a primary key for Student, crsCode with studId as a primary key for Transcript, and using INNER JOIN to loop through the possibilities.

4.List the names of students who have taken a course taught by professor v5 (name).

(Before Tuning) SELECT Student.name, Transcript.crsCode FROM Student, Transcript, Teaching, Professor WHERE Student.id = Transcript.studId AND Professor.id = Teaching.profId AND

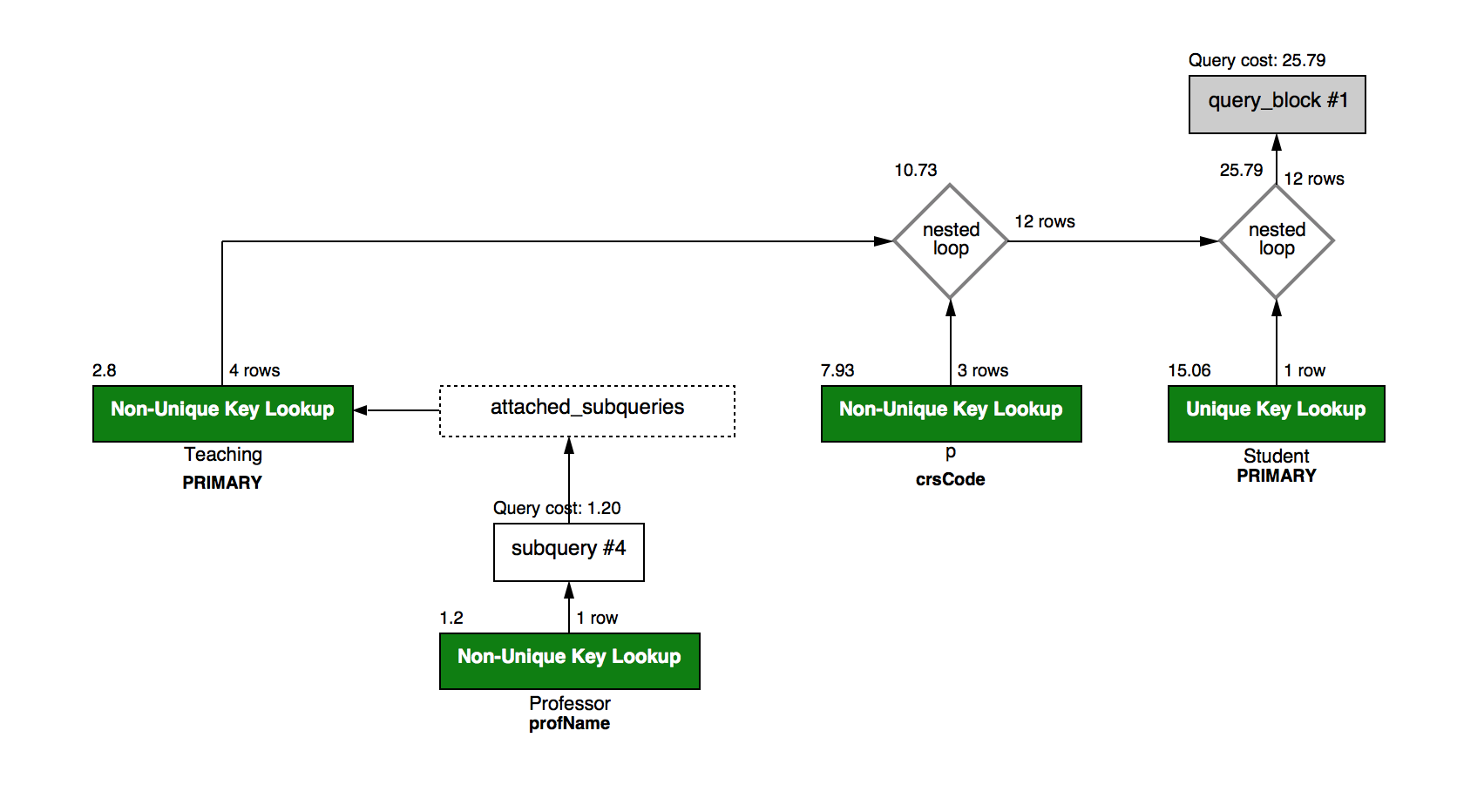
Transcript.crsCode = Teaching.crsCode AND Professor.name = 'name260723';

Takes 0.025 to 0.028 sec to execute.



(After Tuning) SELECT name FROM Student INNER JOIN (SELECT studId FROM Transcript p INNER JOIN (select crsCode FROM Teaching WHERE profId = (SELECT id FROM Professor WHERE name = 'name260723')) AS r ON p.crsCode = r.crsCode ) AS t ON id =t.studId;

Takes 0.00045 to 0.00057 sec to execute.

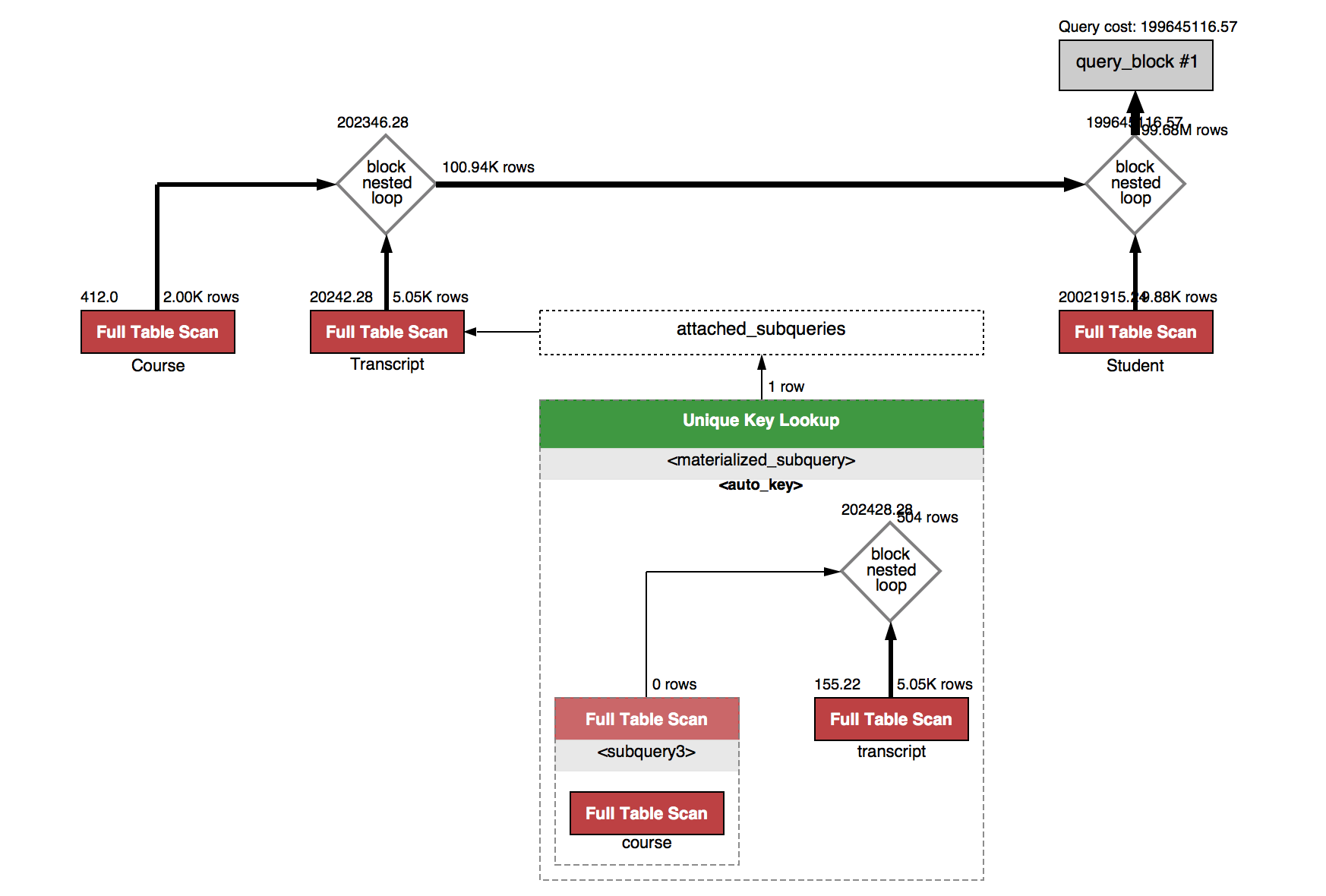


We improved this query by having id be a primary key for Student, studId + crsCode be the primary key for Transcript, profName be the index for name from Professor, crsCode+semester+profId be the primary key for Teaching, nested select statements, and INNER JOINS.

5.List the names of students who have taken a course from department v6 (deptId), but not v7.

(Before Tuning) SELECT Student.name, Transcript.crsCode, Course.deptId FROM Student, Course, Transcript WHERE Student.id = Transcript.studId AND Course.crsCode = Transcript.crsCode AND Course.deptId = "deptId50411" AND student.id not IN (SELECT studId FROM transcript WHERE crsCode IN (SELECT crsCode FROM course WHERE deptId= "deptId563888"));

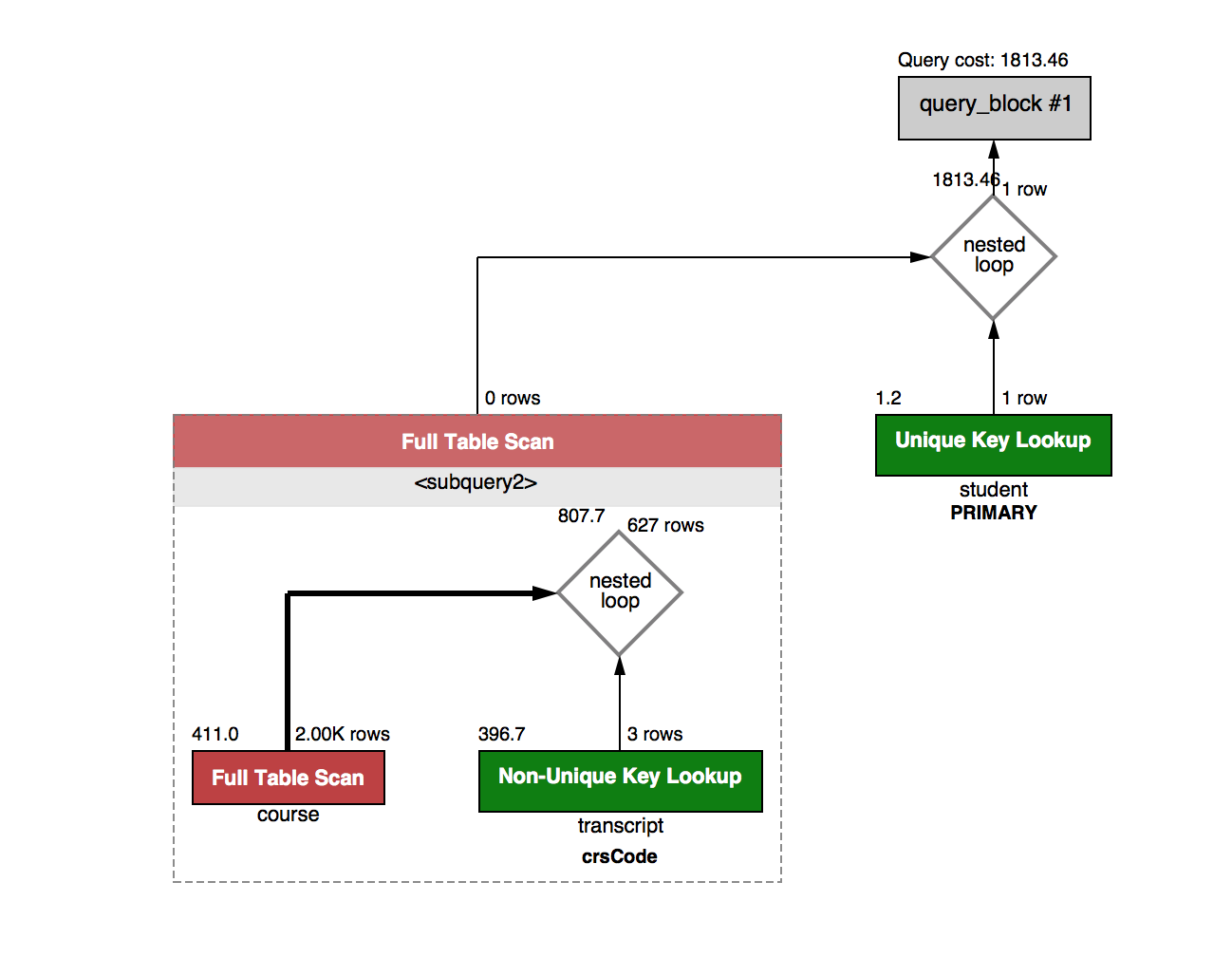
Takes 0.016 to 0.020 sec to execute.



(After Tuning) SELECT name FROM student WHERE id IN (SELECT studId FROM transcript WHERE crsCode IN (SELECT crsCode FROM course WHERE deptId = 'deptId50411')) AND id not IN ((SELECT studId FROM transcript WHERE crsCode IN (SELECT crsCode FROM course WHERE

deptId = 'deptId563888')));

Takes 0.0016 to 0.0024 sec to execute.



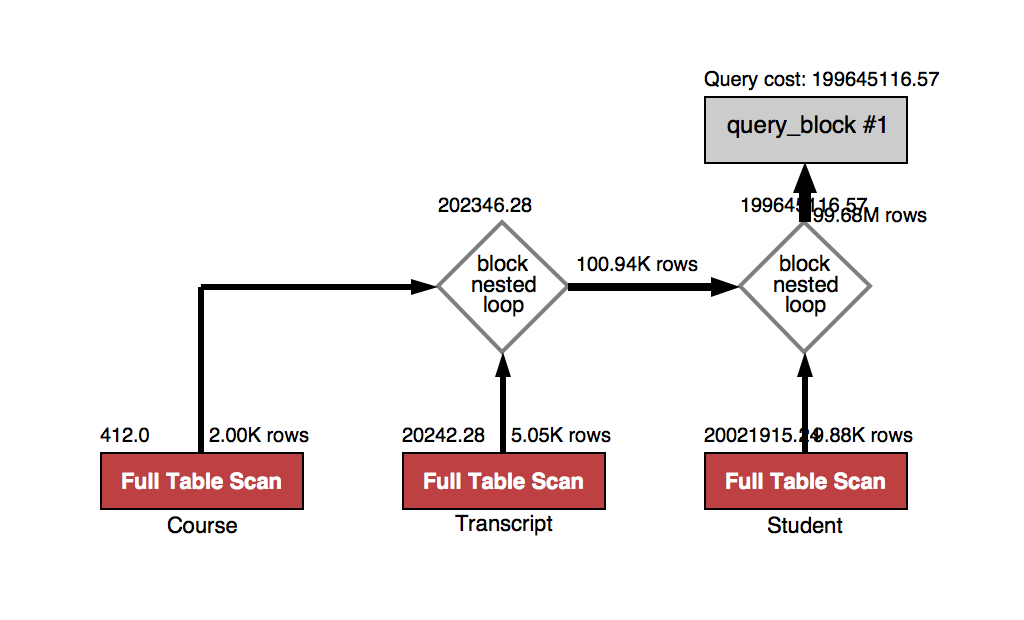
We improved the query by having id as a primary key for Student, studId+crsCode as a primary key for Transcript, nested select statements and not in conditions.

6.List the names of students who have taken all courses offered by department v8 (deptId).

(Before Tuning) SELECT Student.name, Transcript.crsCode, Course.deptId FROM Student, Course, Transcript WHERE Student.id = Transcript.studId AND Course.deptId = "deptId597183"

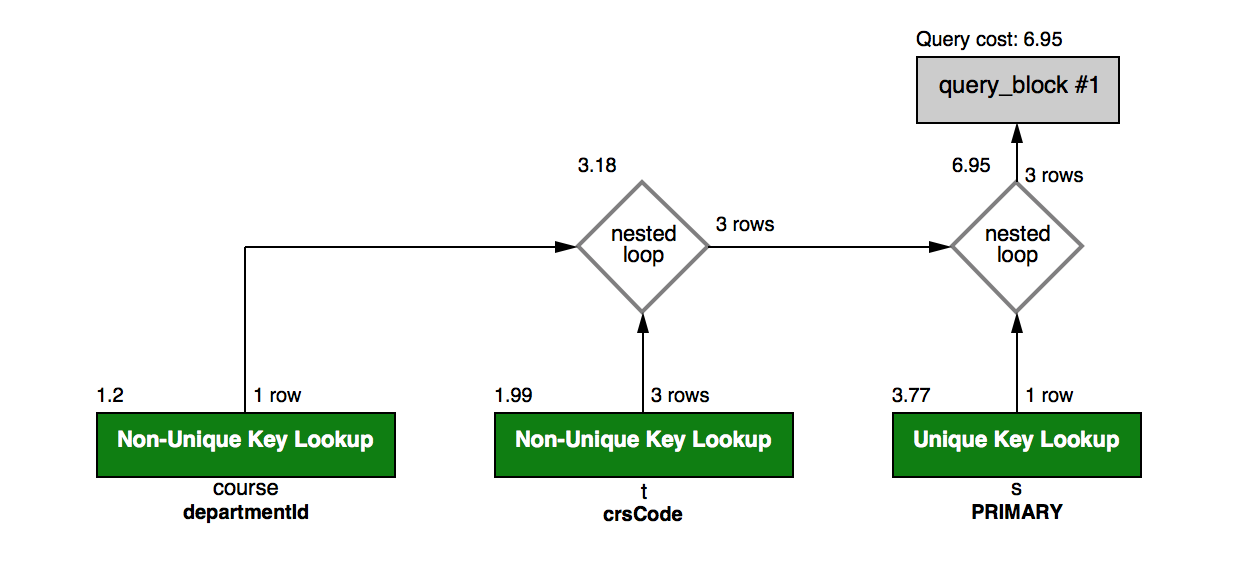
AND course.crsCode = transcript.crsCode;

Takes 0.0083 to 0.0085 to execute.



(After Tuning) SELECT name FROM student s , (SELECT studId FROM transcript t, (SELECT crsCode FROM course WHERE deptId = 'deptId597183' )AS x WHERE t.crsCode = x.crsCode ) AS m WHERE s.id = m.studId;

Takes 0.00035 to 0.00052 sec to execute.



We improved this query by having id as a primary key for Student, studId+crsCode as a primary key for Transcript, an index called departmentId for deptId from Course, and nested select statements.