

# Unifying SQL, Relational Algebra, and the Calculi

Tom Kelliher, CS 318

Mar. 4, 2002

## 1 Administrivia

### Announcements

Formal specification due Friday. What I expect.

### Assignment

Read 8.1–4. Bring hard hats and safety glasses.

### From Last Time

DRC.

### Outline

1. Understanding SQL through TRC.
2. Relationship between relational algebra and calculi.

### Coming Up

Normalization theory, functional dependencies.

## 2 Understanding SQL Through TRC

1. How do we verify that an SQL query will do what we want it to do?

The translation to English isn't all that clear.

2. What does it mean for an SQL query to agree with its English description? Both must hold:

- (a) Every tuple produced by the SQL must satisfy the English description.
- (b) For every tuple,  $t$ , which satisfies the English description, there is a way to assign tuple variable to the SQL query so that the query target is  $t$ .

(Subsets of each other.)

3. Idea: mechanically translate SQL to TRC. TRC is “easy” to understand.

Examples. What's the English translation of each of the following?

1. Simple, single table.

```
SELECT T.Id
FROM Transcript T
WHERE T.CrsCode = 'CS318' AND T.Semester = 'S2002';
```

$$\{T.Id \mid \text{Transcript}(T) \text{ AND } T.CrsCode = 'CS318' \\ \text{AND } T.Semester = 'S2002'\}$$

2. Simple, two tables, single query target.

```
SELECT S.Name
FROM Student S, Transcript T
WHERE S.Id = T.Id AND T.CrsCode = 'CS318' AND T.Semester = 'S2002';
```

$$\{S.Name \mid \text{Student}(S) \text{ AND} \\ \exists T \in \text{Transcript} (S.Id = T.ID \text{ AND} \\ T.CrsCode = 'CS318' \text{ AND } T.Semester = 'S2002')\}$$

How would this change if we added Grade to the query target?

3. More complex, uncorrelated sub-query.

```
SELECT T.Id
FROM Transcript T
WHERE T.CrsCode IN (SELECT C.CrsCode
                    FROM Course C
                    WHERE C.DeptId = 'CS');
```

Translate using a view for the sub-query:

$$CSCourse = \{C.CrsCode \mid Course(C) \text{ AND } C.DeptId = 'CS'\}$$
$$\{T.Id \mid Transcript(T) \text{ AND } \exists C \in CSCourse \\ (T.CrsCode = C.CrsCode)\}$$

Modify to retrieve names.

4. More complex, correlated sub-query.

```
SELECT R.StuId, P.Id, R.CrsCode
FROM Transcript R, Professor P
WHERE R.Semester <> 'S2002'
      AND R.CrsCode IN (SELECT T.CrsCode
                       FROM Teaching T
                       WHERE T.ProfId = P.Id
                       AND T.Semester = 'S2002');
```

Again, use a view:

$$CrsProf = \{T.CrsCode, P.Id \mid Teaching(T) \text{ AND } Professor(P) \text{ AND} \\ T.ProfId = P.ID \text{ AND } T.Semester = 'S2002'\}$$
$$(R.StuId, \\ P.Id, \\ R.CrsCode \mid Transcript(R) \text{ AND } Professor(P) \text{ AND} \\ R.Semester <> 'S2002' \text{ AND} \\ \exists C \in CrsProf (R.CrsCode = C.CrsCode \text{ AND} \\ C.ID = P.Id)\}$$

### 3 Relationship Between Relational Algebra and Calculi

1. Equivalent under domain independence.

In plain English, they're equivalent when queries depend upon only the tuples found in a set of relations, not tuples outside the set.

2. Ullman showed how to reduce calculi queries to relational algebra.

3. Reducing relational algebra queries to TRC:

- (a) Selection: reduce  $\sigma_{Condition}(R)$ .

- (b) Projection: reduce  $\pi_{A,B,C}(R)$ .

- (c) Cartesian Product: reduce  $R \times S$ . (Assume  $R$  has attributes  $A, B, C$  and  $S$  has attributes  $D$  and  $E$ .)

- (d) Union: reduce  $R \cup S$ .

- (e) Set difference: reduce  $R - S$ .

- (f) Division: "No discussion of relational algebra is complete without saying something about the division operator."