CSE 344 Introduction to Data Management

Section 4: Relational Algebra
Outline

• HW3 Check-in
• Relational Algebra Review
• Translate nested SQL Queries to RA
• Translate from RA to SQL
Relational Algebra

• SQL = **WHAT** we want to get from the data
• Relational Algebra = **HOW** to get the data we want
• SQL $\rightarrow$ Relational Algebra $\rightarrow$ Physical Plan
• Relational Algebra = Logical Plan (usually written as a tree)
Relational Algebra Operators

Standard:
- Selection: \( \sigma \)
- Projection: \( \pi \)
- Rename: \( \rho \)

Sets:
- Union: \( \cup \)
- Intersection: \( \cap \)
- Difference: \( - \)

Joins:
- Cartesian Product: \( \times \)
- Join: \( \bowtie \)

Extended:
- Duplicate Elimination: \( \delta \)
- Grouping and Aggregation: \( \gamma \)
- Sorting: \( T \)
SQL to RA Review

• Write a Relational Algebra plan for the following query:

```sql
SELECT d.did, d.name, count(*)
FROM Dancer d, Show s, Role r
WHERE d.did=r.did
AND r.sid=s.sid
AND s.composer='Tchaikovsky'
GROUP BY d.did, d.name
ORDER BY d.name;
```
SELECT d.did, d.name, count(*)
FROM Dancer d, Show s, Role r
WHERE d.did=r.did
AND r.sid=s.sid
AND s.composer='Tchaikovsky'
GROUP BY d.did, d.name
ORDER BY d.name;
SQL to RA Solution

```sql
SELECT d.did, d.name, count(*)
FROM Dancer d, Show s, Role r
WHERE d.did = r.did
AND r.sid = s.sid
AND s.composer = 'Tchaikovsky'
GROUP BY d.did, d.name
ORDER BY d.name;
```
SELECT d.did, d.name, count(*)
FROM Dancer d, Show s, Role r
WHERE d.did=r.did
AND r.sid=s.sid
AND s.composer='Tchaikovsky'
GROUP BY d.did, d.name
ORDER BY d.name;
SQL to RA Solution

```
SELECT d.did, d.name, count(*)
FROM Dancer d, Show s, Role r
WHERE d.did=r.did
AND r.sid=s.sid
AND s.composer='Tchaikovsky'
GROUP BY d.did, d.name
ORDER BY d.name;
```
SQL to RA Solution

SELECT d.did, d.name, count(*)
FROM Dancer d, Show s, Role r
WHERE d.did=r.did
AND r.sid=s.sid
AND s.composer='Tchaikovsky'
GROUP BY d.did, d.name
ORDER BY d.name;
SELECT d.did, d.name, count(*)
FROM Dancer d, Show s, Role r
WHERE d.did=r.did
AND r.sid=s.sid
AND s.composer='Tchaikovsky'
GROUP BY d.did, d.name
ORDER BY d.name;
SELECT w.year, max(w.c) AS m
FROM(SELECT x.name, z.year, count(*) AS c
    FROM Member x, Tagged y, Picture z
    WHERE x.mid = y.mid
    AND y.pid = z.pid
    AND age < 20
    GROUP BY x.name, z.year) w
GROUP BY w.year
HAVING sum(w.c) > 100;
Nested SQL Queries to RA Solution

```sql
SELECT w.year, max(w.c) AS m
FROM (SELECT x.name, z.year, count(*) AS c
     FROM Member x, Tagged y, Picture z
     WHERE x.mid = y.mid
     AND y.pid = z.pid
     AND age < 20
     GROUP BY x.name, z.year) w
GROUP BY w.year
HAVING sum(w.c) > 100;
```
Translate from RA to SQL

- Put tables in FROM clause
- Put join predicates in WHERE clause
- Put selection predicates in WHERE clause
- Translate extended RA symbols to SQL equivalent
- Put selection of aggregates in HAVING clause
- Put projection predicates in SELECT clause
RA to SQL Example

Person(pid, name)
Email(eid, pidFrom, tid, body, length)
EmailTo(eid, pidTo)
RA to SQL Solution

```
SELECT e1.pidFrom, count(*)
FROM Email e1, EmailTo t1, Email e2
WHERE e1.eid = t1.eid
AND t1.pidTo = e2.pidFrom
GROUP BY e1.pidFrom
HAVING max(e2.length) < 1000;
```