Fun With Relational Algebra

Part II

Overview

- Quick review
- Practice with relational algebra queries

Relational Algebra

- What?
  - Simple language for manipulating and retrieving data
  - Operational
- Why?
  - Simple
    - Only need to write code for a few operators
  - Expressive
  - Can be optimized
  - Variants of relational algebra can be used to represent query execution plans
- When?
  - Now

Meet the Operators

- *Selection* (σ) Selects a subset of rows from relation.
- *Projection* (π) Deletes unwanted columns from relation.
- *Union* ($\cup$) Tuples in reln. 1 or in reln. 2.
- *Set-difference* ($-$) Tuples in reln. 1, but not in reln. 2.
- *Cross-product* (**X**) Allows us to combine two relations.

- Additional operators: intersection, join(s), renaming, division.

Selection

S2

sid	sname	rating	age
28	Yuppy	9	35
31	Lubber	8	55.5
44	Guppy	5	35
58	Rusty	10	35.5

$\sigma_{rating > 8}(S2)$

sid	sname	rating	age
28	Yuppy	9	35
58	Rusty	10	35.5

Projection

S2

sid	sname	rating	age
28	Yuppy	9	35
31	Lubber	8	55.5
44	Guppy	5	35
58	Rusty	10	35.5

$\pi_{age}(S2)$

age
55.5
35
35.5

- Eliminate duplicates (why?)

Union

S1

sid	sname	srating	age
22	Dustin	7	45
31	Lubber	8	55.5
58	Rusty	10	35.5

S2

sid	name	rating	age
28	Yuppy	9	35
31	Lubber	8	55.5
44	Guppy	5	35
58	Rusty	10	35.5

S1 ∪ S2

sid	sname	srating	age
22	Dustin	7	45
31	Lubber	8	55.5
28	Yuppy	9	35
44	Guppy	5	35
58	Rusty	10	35.5

- Union Compatible
- Eliminate duplicates

Set Difference

S1

sid	sname	rating	age
22	Dustin	7	45
31	Lubber	8	55.5
58	Rusty	10	35.5

S2

sid	sname	rating	age
28	Yuppy	9	35
31	Lubber	8	55.5
44	Guppy	5	35
58	Rusty	10	35.5

S2 – S1

sid	sname	rating	age
28	Yuppy	9	35
44	Guppy	5	35

- Union Compatible

(Intersection)

S1

sid	sname	rating	age
22	Dustin	7	45
31	Lubber	8	55.5
58	Rusty	10	35.5

S2

sid	sname	rating	age
28	Yuppy	9	35
31	Lubber	8	55.5
44	Guppy	5	35
58	Rusty	10	35.5

S2 – S1

sid	sname	rating	age
28	Yuppy	9	35
44	Guppy	5	35

S2 – (S2 – S1)= S1 ∩ S2

sid	sname	rating	age
31	Lubber	8	55.5
58	Rusty	10	35.5

• Union Compatible

Cross Product

S1

sid	sname	rating	age
22	Dustin	7	45
31	Lubber	8	55.5
58	Rusty	10	35.5

R1

sid	bid	day	age
22	101	10/10/96	1.5
58	103	11/12/96	2

S1 X R1

(sid)	sname	rating	(age)	(sid)	bid	day	(age)
22	Dustin	7	45	22	101	10/10/96	1.5
22	Dustin	7	45	58	103	11/12/96	2
31	Lubber	8	55.5	22	101	10/10/96	1.5
31	Lubber	8	55.5	58	103	11/12/96	2
58	Rusty	10	35.5	22	101	10/10/96	1.5
58	Rusty	10	35.5	58	103	11/12/96	2

Rename

(sid)	sname	rating	(age)	(sid)	bid	day	(age)

$\rho(C(1 \to sid1, 4 \to age1, 5 \to sid2, 8 \to age2), \text{ S1 X R1})$

sid1	sname	rating	age1	sid2	bid	day	age2
22	Dustin	7	45	22	101	10/10/96	1.5
22	Dustin	7	45	58	103	11/12/96	2
31	Lubber	8	55.5	22	101	10/10/96	1.5
31	Lubber	8	55.5	58	103	11/12/96	2
58	Rusty	10	35.5	22	101	10/10/96	1.5
58	Rusty	10	35.5	58	103	11/12/96	2

Joins

- Cross Product A X B
- Join $A \bowtie_{cond} B = \sigma_{cond}(A \times B)$
- Theta join $A \bowtie_{x<y} B$
- Equijoin $A \bowtie_{x} B$
- Natural Join $A \bowtie B$

Join

S1

sid	sname	rating	age
22	Dustin	7	45
31	Lubber	8	55.5
58	Rusty	10	35.5

R1

sid	bid	day	age
22	101	10/10/96	1.5
58	103	11/12/96	2

A ⋈ B
A.sid=B.sid
^ A.age != B.age

(sid)	sname	rating	(age)	(sid)	bid	day	(age)
22	Dustin	7	45	22	101	10/10/96	1.5
58	Rusty	10	35.5	58	103	11/12/96	2

Equijoin

S1

sid	sname	rating	age
22	Dustin	7	45
31	Lubber	8	55.5
58	Rusty	10	35.5

R1

sid	bid	day	age
22	101	10/10/96	1.5
58	103	11/12/96	2

A ⋈ B
sid

sid	sname	rating	(age)	bid	day	(age)
22	Dustin	7	45	101	10/10/96	1.5
58	Rusty	10	35.5	103	11/12/96	2

- Note the schema

Natural Join

S1

sid	sname	rating	age
22	Dustin	7	45
31	Lubber	8	55.5
58	Rusty	10	35.5

R1

sid	bid	day	age
22	101	10/10/96	1.5
58	103	11/12/96	2

A ⋈ B

sid	sname	rating	age	bid	day

- What if no attributes in common?
- Note the schema
- Why is the table empty?

Natural Join

S1

sid	sname	rating	age
22	Dustin	7	45
31	Lubber	8	55.5
58	Rusty	10	35.5

R1

sid	bid	day
22	101	10/10/96
58	103	11/12/96

A ⋈ B

sid	sname	rating	age	bid	day
22	Dustin	7	45	101	10/10/96
58	Rusty	10	35.5	103	11/12/96

Division

Table1

X	Y	A	B

Table2

A	B

- Find all (X, Y) in Table1 that are paired with all (A, B) from Table2
- Example query: "find the sailors who rented all boats"

Division

A

sid	bid
s1	b1
s1	b2
s1	b4
s2	b2
s2	b4
s3	b4
s4	b3

B

bid
b2
b4

A/B

sid
s1
s2

Division

A

sid	bid
s1	b1
s1	b2
s1	b4
s2	b2
s2	b4
s3	b4
s4	b3

B

bid
b2
b4

A/B

sid
s1
s2

(A/B) X B

sid	bid
s1	b2
s1	b4
s2	b2
s2	b4

- (A/B) X B $\subseteq$ A

Division

C

sid
s1
s2

D

bid
b2
b4

C X D

sid	bid
s1	b2
s1	b4
s2	b2
s2	b4

(C X D) / D

sid
s1
s2

- (C X D) / D = C

Division

- (A/B) X B $\subseteq$ A
- (C X D) / D = C

A

sid	bid
s1	b1
s1	b2
s1	b4
s2	b2
s2	b4
s3	b4
s4	b3

B

bid
b2
b4

π_{sid}(A)

sid
s1
s2
s3
s4

π_{sid}(A) X B

sid	bid
s1	b2
s1	b4
s2	b2
s2	b4
s3	b2
s3	b4
s4	b2
s4	b4

(π_{sid}(A) X B) - A

?

A - (π_{sid}(A) X B)

?

Fun with Relational Algebra

A

sid	bid
s1	b1
s1	b2
s1	b4
s2	b2
s2	b4
s3	b4
s4	b3

B

bid
b2
b4

π_{sid}(A)

sid
s1
s2
s3
s4

π_{sid}(A) X B

sid	bid
s1	b2
s1	b4
s2	b2
s2	b4
s3	b2
s3	b4
s4	b2
s4	b4

(π_{sid}(A) X B) - A

sid	bid
s3	b2
s4	b2
s4	b4

A - (π_{sid}(A) X B)

sid	bid
s1	b1
s4	b3

Uses of Subtraction

- $\pi_{sid}((\pi_{sid}(A) \times B) - A)$
  - sid's of Sailors who did not reserve all boats in B
  - $\pi_{sid}(A) - \pi_{sid}((\pi_{sid}(A) \times B) - A)$ = Sailors who reserved ***all*** boats in B = A/B
- $\pi_{sid}(A - (\pi_{sid}(A) \times B))$
  - sid's of Sailors who reserved a boat not in B
  - $\pi_{sid}(A) - \pi_{sid}(A - (\pi_{sid}(A) \times B))$ = Sailors who reserve boats only from B (but they don't have to reserve all boats in B)