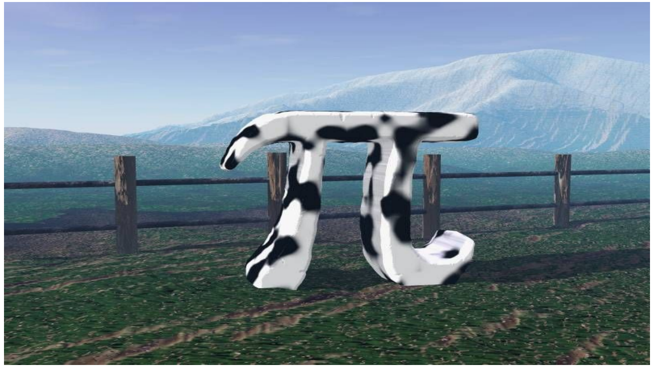
Relational Algebra

Chapter 4.1-4.2
Part II



Complete Chapter 5 worksheet before Monday's class

Division (not an operator in SQL)

- Useful for expressing queries like:
  *Find sailors who have reserved **all** boats.*
- Let A have 2 fields, x and y; B have only field y:
  - $A/B = \{\langle x\rangle \mid \exists \langle x, y\rangle \in A \;\; \forall \langle y\rangle \in B\}$
  - i.e., ***A/B* contains all *x* tuples (sailors) such that for *every y* tuple (boat) in *B*, there is an *xy* tuple in *A*.**
  - If the set of y values (boats) associated with an x value (sailor) in A contains all y values in B, the x value is in A/B.

Examples of Division A/B

sno	pno
s1	p1
s1	p2
s1	p3
s1	p4
s2	p1
s2	p2
s3	p2
s4	p2
s4	p4

A

pno
p2

B1

Examples of Division A/B

sno	pno
s1	p1
s1	p2
s1	p3
s1	p4
s2	p1
s2	p2
s3	p2
s4	p2
s4	p4

A

pno
p2

B1

sno
s1
s2
s3
s4

A/B1

Examples of Division A/B

sno	pno
s1	p1
s1	p2
s1	p3
s1	p4
s2	p1
s2	p2
s3	p2
s4	p2
s4	p4

A

pno
p2

B1

pno
p2
p4

B2

sno
s1
s2
s3
s4

A/B1

Examples of Division A/B

sno	pno
s1	p1
s1	p2
s1	p3
s1	p4
s2	p1
s2	p2
s3	p2
s4	p2
s4	p4

A

pno
p2

B1

pno
p2
p4

B2

sno
s1
s2
s3
s4

A/B1

sno
s1
s4

A/B2

Examples of Division A/B

sno	pno
s1	p1
s1	p2
s1	p3
s1	p4
s2	p1
s2	p2
s3	p2
s4	p2
s4	p4

A

pno
p2

B1

pno
p2
p4

B2

pno
p1
p2
p4

B3

sno
s1
s2
s3
s4

A/B1

sno
s1
s4

A/B2

Examples of Division A/B

sno	pno
s1	p1
s1	p2
s1	p3
s1	p4
s2	p1
s2	p2
s3	p2
s4	p2
s4	p4

A

pno
p2

B1

sno
s1
s2
s3
s4

A1÷B1

pno
p2
p4

B2

sno
s1
s4

A÷B2

pno
p1
p2
p4

B3

sno
s1

A÷B3

Expressing A/B Using Basic Operators

- Division is not essential; it's just a useful shorthand.
  - (Also true of joins, but joins are so common that systems implement joins specially.)

Expressing A/B Using Basic Operators

- *Idea:* For *A/B*, compute all *x* values that are not "disqualified" by some *y* value in *B*.
  - *x* value is *disqualified* if by attaching *y* value from *B*, we obtain an *xy* tuple that is not in *A*.

Disqualified *x* values: $\pi_x((\pi_x(A) \times B) - A)$

A/B: $\pi_x(A)$ $-$ disqualified x values

Relational Algebra Examples

- Assume the following extended schema:
  - ***Sailors(sid: integer, sname: string, rating: integer, age: real)***
  - ***Reserves(sid: integer, bid: integer, day: date)***
  - ***Boat(bid: integer, bname: string, bcolor: string)***

Relational Algebra Examples

- Objective: Write a relational algebra expression whose result instance satisfies the specified conditions
  - May not be unique
  - Some alternatives might be more efficient (in terms of time and/or space)

Names of sailors who reserved boat #103

Solution 1: $\pi_{sname}(\sigma_{bid=103}(Reserves \bowtie Sailors)$

Solution 2: $\pi_{sname}((\sigma_{bid=103}Reserves) \bowtie Sailors$

Solution 3: $\rho(T1, \sigma_{bid=103}Reserves)$

$\rho(T2, T1 \bowtie Sailors)$

$\pi_{sname}T2$

Sailors who reserved boat 103

Names of sailors who reserved a red boat

- Information about boat color only available in Boats; so need an extra join:

Names of sailors who've reserved a red boat

- Information about boat color only available in Boats; so need an extra join:
  $\pi_{sname}((\sigma_{color=red} Boats) \bowtie Reserves \bowtie Sailors)$

- A more efficient solution
  $\pi_{sname}(\pi_{sid}(\pi_{bid}(\sigma_{color=red} Boats) \bowtie Reserves) \bowtie Sailors)$

- A query optimizer could find this from first.

[Sailors who've reserved a red boat]

Sailors who've reserved a red or a green boat

- Can identify all red or green boats, then find sailors who've reserved one of these boats:

Sailors who've reserved a red or a green boat

- Can identify all red or green boats, then find sailors who've reserved one of these boats:

$$\boldsymbol{\rho}(\mathrm{T}, \boldsymbol{\sigma}_{\mathrm{color=red} \vee \mathrm{color=green}} \mathrm{Boats})$$

$$\boldsymbol{\pi}_{\mathrm{sname}}(\mathrm{T} \bowtie \mathrm{Reserves} \bowtie \mathrm{Sailors})$$

Sailors who've reserved a red and a green boat

- Previous approach won't work! Must identify sailors who've reserved red boats, sailors who've reserved green boats, then find the intersection (note that *sid* is a key for Sailors):

Sailors who've reserved a red <u>and</u> a green boat

- Previous approach won't work! Must identify sailors who've reserved red boats, sailors who've reserved green boats, then find the intersection (note that *sid* is a key for Sailors)

$$\rho(T1, \pi_{sid}((\sigma_{color=red} Boats) \bowtie Reserves))$$

$$\rho(T2, \pi_{sid}((\sigma_{color=green} Boats) \bowtie Reserves))$$

$$\pi_{sname}((T1 \cap T2) \bowtie Sailors)$$

[Sailors who reserved a red and green boat]()

Summary

- The relational model has rigorously defined query languages that are simple and powerful.
- Relational algebra is more operational; useful as internal representation for query evaluation plans.
- Several ways of expressing a given query; a query optimizer should choose the most efficient version.