

Relational Algebra Queries

CS430/630
Lecture 3

.forward

Please put your preferred email address in .forward file of your login directory at cs.umb.edu, for example:

```
cat > .forward  
joe@gmail.com  
<control-D>
```

Then email to joe@cs.umb.edu will be forwarded to joe@gmail.com, allowing us to use the class email list cs630-l@cs.umb.edu. In particular, I will use this list to announce new homework, corrections to it, etc.

<control-D> is a single character produced on the keyboard by depressing the control key and then the D key.



Relational Algebra

▶ Basic operations:

- ▶ Selection σ Selects a subset of rows from relation
- ▶ Projection π Deletes unwanted columns from relation
- ▶ Cross-product \times Allows us to combine several relations
- ▶ Join \bowtie Combines several relations using conditions
- ▶ Division \div A bit more complex, will cover later on
- ▶ Set-difference $-$ Union \cup Intersection \cap
- ▶ Renaming ρ Helper operator, does not derive new result, just renames relations and fields

$$\rho(R(F), E)$$

- ▶ F contains *oldname* \rightarrow *newname* pairs
-



Operator Precedence

► In decreasing order of priority:

1. Selection σ Projection π
2. Cross-product \times Join \bowtie
3. Set-difference $-$ Intersection \cap
4. Union \cup

Example: $\sigma_{bid=103} Reserves \bowtie Sailors$

means $(\sigma_{bid=103} Reserves) \bowtie Sailors$

not $\sigma_{bid=103} (Reserves \bowtie Sailors)$



Example Schema

Sailors

<u>sid</u>	sname	rating	age
22	dustin	7	45.0
31	lubber	8	55.5
58	rusty	10	35.0

Boats

<u>bid</u>	name	color
101	interlake	red
103	clipper	green

Reserves

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



Sample Query 0

Sailors

<u>sid</u>	sname	rating	age
------------	-------	--------	-----

Boats

<u>bid</u>	name	color
------------	------	-------

Reserves

<u>sid</u>	<u>bid</u>	<u>day</u>
------------	------------	------------

- ▶ Find names of sailors who have ratings at least 8.
Detail of sailor sid Another sailor detail

$$\pi_{sname}((\sigma_{rating \geq 8} \text{Sailors}))$$

$$\pi_{sname} \sigma_{rating \geq 8} \text{Sailors}$$



Join Reserves ⋈ Sailors

Reserves

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

Sailors

<u>sid</u>	sname	rating	age
22	dustin	7	45.0
31	lubber	8	55.5
58	rusty	10	35.0

Reserves ⋈ Sailors

sid	bid	day	sname	rating	age
22	101	10/10/96	dustin	7	45.0
58	103	11/12/96	rusty	10	35.0

Each sid in Reserves is filled out with Sailor attributes

Sample Query 1

Sailors

<u>sid</u>	sname	rating	age
------------	-------	--------	-----

Boats

<u>bid</u>	name	color
------------	------	-------

Reserves

<u>sid</u>	<u>bid</u>	<u>day</u>
------------	------------	------------

bid only, no Boat details

- ▶ Find names of sailors who've reserved boat #103
Detail of sailor sid sid, bid in reserves table

$$\pi_{sname}((\sigma_{bid=103} \text{Reserves}) \bowtie \text{Sailors})$$

$$\pi_{sname}(\sigma_{bid=103}(\text{Reserves} \bowtie \text{Sailors}))$$



Example Schema

Sailors

<u>sid</u>	sname	rating	age
22	dustin	7	45.0
31	lubber	8	55.5
58	rusty	10	35.0

Boats

<u>bid</u>	name	color
101	interlake	red
103	clipper	green

Reserves

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



Sample Query 2

Sailors

<u>sid</u>	sname	rating	age
------------	-------	--------	-----

Boats

<u>bid</u>	name	color
------------	------	-------

Reserves

<u>sid</u>	<u>bid</u>	<u>day</u>
------------	------------	------------

- ▶ Find names of sailors who've reserved a red boat
Detail of sailor sid sid, bid ... Detail of boat bid

$$\pi_{sname}(\pi_{sid}((\pi_{bid}(\sigma_{color='red'} B)) \bowtie R) \bowtie S)$$
$$\pi_{sname}((\sigma_{color='red'} Boats) \bowtie Reserves \bowtie Sailors)$$



Sample Query 2

- ▶ Find names of sailors who've reserved a red boat
Detail of sailor sid sid, bid ... Detail of boat bid

- ▶ One way that's right:

$$\pi_{sname}((\sigma_{color='red'}Boats) \bowtie Reserves \bowtie Sailors)$$

... but this next is Wrong!: Watch out for precedence!

$$\pi_{sname} \sigma_{color='red'} Boats \bowtie Reserves \bowtie Sailors$$
$$(\pi_{sname}(\sigma_{color='red'}Boats)) \bowtie Reserves \bowtie Sailors$$

empty!



Sample Query 3

Sailors

<u>sid</u>	sname	rating	age
------------	-------	--------	-----

Boats

<u>bid</u>	name	color
------------	------	-------

Reserves

<u>sid</u>	<u>bid</u>	<u>day</u>
------------	------------	------------

- ▶ Find names of sailors who've reserved a red or a green boat
 $\rho (Tempboats, (\sigma_{color='red' \vee color='green'} Boats))$

$\pi_{sname}(Tempboats \bowtie Reserves \bowtie Sailors)$

$\pi_{sname}(\sigma_{color='red' \vee color='green'} B \bowtie R \bowtie S)$



Sample Query 4

Sailors

<u>sid</u>	sname	rating	age
------------	-------	--------	-----

Boats

<u>bid</u>	name	color
------------	------	-------

Reserves

<u>sid</u>	<u>bid</u>	<u>day</u>
------------	------------	------------

- ▶ Find names of sailors who've reserved a red and a green boat

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

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

$\pi_{sname}((Tempred \cap Tempgreen) \bowtie Sailors)$



Sample Query 5

Sailors

<u>sid</u>	sname	rating	age
------------	-------	--------	-----

Boats

<u>bid</u>	name	color
------------	------	-------

Reserves

<u>sid</u>	<u>bid</u>	<u>day</u>
------------	------------	------------

- ▶ Find names of sailors who've reserved only red boats

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

$\rho(\text{Tempothers}, \pi_{sid}((\sigma_{color \neq 'red'} \text{Boats}) \bowtie \text{Reserves}))$

$\pi_{sname}(\text{Tempred} - \text{Tempothers}) \bowtie \text{Sailors}$



Time to try it yourself...

- ▶ Try the exercises on the handed-out sheet
- ▶ You can confer with neighbors—this is not graded
- ▶ Turn in completed paper for the 3 points

- ▶ Lab sheet (Solution) (Solution posted later)

- ▶ Note: you need to attend class to get credit for this work—it is a form of class participation.



An Example of Self-Joins

Sailors

<u>sid</u>	sname	rating	age
22	dustin	7	45.0
31	lubber	8	55.5
58	rusty	10	35.0

- ▶ Find sailors with maximum age
 - ▶ No max operator in RA... (SQL has this)
 - ▶ Need a trick: self join with “left” age smaller than “right” age
 - ▶ This will list rows for all ages for left side *but the max age row(s)*
 - ▶ *Then use all-sailors – this list*



An Example of Self-Join: cross-product with rows eliminated by condition

“LeftHalf”

<u>sid1</u>	sname1	rating1	age1	<u>sid2</u>	sname2	rating2	age2
22	dustin	7	45.0	22	dustin	7	45.0
22	dustin	7	45.0	31	lubber	8	55.5
22	dustin	7	45.0	58	rusty	10	35.0
31	lubber	8	55.5	22	dustin	7	45.0
31	lubber	8	55.5	31	lubber	8	55.5
31	lubber	8	55.5	58	rusty	10	35.0
58	rusty	10	35.0	22	dustin	7	45.0
58	rusty	10	35.0	31	lubber	8	55.5
58	rusty	10	35.0	58	rusty	10	35.0

- ▶ Join condition: “left” age smaller than “right” age



An Example of Self-Join: Max ages

$\rho(S1, Sailors)$

$\rho(S2, Sailors)$

$\rho(TempJoin(1 \rightarrow f1, 2 \rightarrow f2, 3 \rightarrow f3, 4 \rightarrow f4),$
 $S1 \bowtie_{S1.age < S2.age} S2)$

$\rho(LeftHalf, \pi_{f1, f2, f3, f4} TempJoin)$

- ▶ Finally, subtract the resulting left hand side from the initial relation, and you get sailors with maximum ages
- ▶ Final result is

$Sailors - LeftHalf$



More on Natural Joins

Natural Joins match all same-named columns

- ▶ Consider two tables T1 and T2:

$T1(\underline{id1}, attr1, city)$ $T2(\underline{id2}, id1, attr2, city)$

- ▶ Probably want to join on $id1$, a key for T1 showing up in both tables
 - ▶ But T1 and T2 have $id1$ and $city$ in common, so a natural join $T1 \bowtie T2$ matches both
 - ▶ If we don't want non-key columns matched like this
 - ▶ We can use a theta join with an explicit condition:
 $T1 \bowtie_{t1.id1=t2.id1} T2$
 - ▶ Or project out $city$ before one of the joins
-



Consider the Example Schema, modified to have a name attribute for two entities

Sailors

<u>sid</u>	name	rating	age
22	dustin	7	45.0
31	lubber	8	55.5
58	rusty	10	35.0

Boats

<u>bid</u>	name	color
101	interlake	red
103	clipper	green

Reserves

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



Sample Query 2 on modified schema

Sailors

<u>sid</u>	name	rating	age
------------	------	--------	-----

Boats

<u>bid</u>	name	color
------------	------	-------

Reserves

<u>sid</u>	<u>bid</u>	<u>day</u>
------------	------------	------------

- ▶ Find names of sailors who've reserved a red boat
Detail of sailor sid sid, bid ... Detail of boat bid

- ▶ Old solution:

$$\pi_{sname}((\sigma_{color='red'} Boats) \bowtie Reserves \bowtie Sailors)$$

- ▶ Returns an empty relation!
- ▶ It's looking for matches on name as well as bid, sid



Sample Query 2 on modified schema

Sailors

<u>sid</u>	name	rating	age
------------	------	--------	-----

Boats

<u>bid</u>	name	color
------------	------	-------

Reserves

<u>sid</u>	<u>bid</u>	<u>day</u>
------------	------------	------------

- ▶ Find names of sailors who've reserved a red boat
- ▶ Old solution for unmodified schema: returns empty table here
- ▶ $\pi_{sname}((\sigma_{color='red'} Boats) \bowtie Reserves \bowtie Sailors)$
- ▶ Here we can project out boat names before join to Sailors

$$\pi_{name}(((\pi_{bid} \sigma_{color='red'} B) \bowtie R) \bowtie S)$$



Another self join:
Close competitors

<u>sid</u>	sname	rating	age
28	yuppy	9	35.0
31	lubber	8	55.5
44	guppy	5	35.0
58	rusty	10	35.0

- Find pairs of sailors (sids) with ratings that differ by no more than one.

$\rho(S1, Sailors) \quad \rho(S2, Sailors)$

$\rho(TempJoin(1 \rightarrow sid1, 5 \rightarrow sid2),$
 $S1 \bowtie_{S1.rating \leq S2.rating + 1 \wedge S1.rating \geq S2.rating - 1} S2)$

$\pi_{sid1, sid2} TempJoin$

sid1	sid2
28	28
28	31
28	58
31	28
...	...

- We don't want a lot of these results...

Another self join

<u>sid</u>	sname	rating	age
28	yuppy	9	35.0
31	lubber	8	55.5
44	guppy	5	35.0
58	rusty	10	35.0

- Find pairs of *different* sailors (sids) with ratings that differ by no more than one, listing each unordered pair once.

$$\begin{aligned}
 & \rho(S1, Sailors) \quad \rho(S2, Sailors) \\
 & \rho(TempJoin(1 \rightarrow sid1, 5 \rightarrow sid2), \\
 & S1 \bowtie_{S1.rating \leq S2.rating + 1 \wedge S1.rating \geq S2.rating - 1 \wedge \underline{S1.sid < S2.sid}} S2)
 \end{aligned}$$

$$\pi_{sid1, sid2} TempJoin$$

sid1	sid2
28	31
28	58

- That's better!

Like Query 0

Sailors

<u>sid</u>	sname	rating	age
------------	-------	--------	-----

Boats

<u>bid</u>	name	color
------------	------	-------

Reserves

<u>sid</u>	<u>bid</u>	<u>day</u>
------------	------------	------------

- ▶ Find colors of boats with names starting with C

Detail of Boat

Another Boat detail (assume lowercase names)

$$\pi_{color}((\sigma_{name \geq 'c' \wedge name < 'd'} Boats))$$

$$\pi_{color} \sigma_{name \geq 'c' \wedge name < 'd'} Boats$$



Like Query 3

Sailors

<u>sid</u>	sname	rating	age
------------	-------	--------	-----

Boats

<u>bid</u>	name	color
------------	------	-------

Reserves

<u>sid</u>	<u>bid</u>	<u>day</u>
------------	------------	------------

- ▶ Find names of sailors who've reserved a red or a green boat. List names and the boat color (two rows if the sailor rented both color boats)

ρ (*Tempboats*, ($\sigma_{color='red' \vee color='green'}$ *Boats*))

$\pi_{sname,color}$ (*Tempboats* \bowtie *Reserves* \bowtie *Sailors*)

