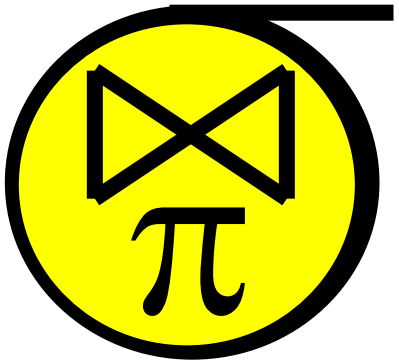
Information Storage and Management I

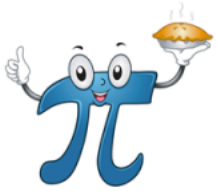
Dr. Alejandro Arbelaez



Relational Algebra

- Labs Today 3-4 PM – **1.10**

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 two relations.
  - ***Set-difference*** (-) Tuples in reln. 1, but not in reln. 2.
  - ***Union*** ($\cup$) Tuples in reln. 1 and in reln. 2.
  - ***Renaming*** (ρ) (for named perspective)
- Additional operations:
  - Intersection, ***join***, division, renaming: Not essential, but (very!) useful.
- Since each operation returns a relation, operations can be *composed*!

Projection

- Deletes attributes that are not in *projection list*.
- ***Schema*** of result contains exactly the fields in the projection list, with the same names that they had in the (only) input relation.
- Projection operator has to eliminate ***duplicates***! (Why??)
  - Note: real systems typically don't do duplicate elimination unless the user explicitly asks for it. (Why not?)

S2

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

$$\pi_{sname,rating}(S2)$$

sname	rating
yuppy	9
lubber	8
guppy	5
rusty	10

Selection

- Selects rows that satisfy ***selection condition***
- No duplicates in result! (Why?)
- ***Schema*** of result identical to schema of (only) input relation.
- *Result* relation can be the *input* for another relational algebra operation! (*Operator composition.*)

S2

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

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

sid	sname	rating	age
28	yuppy	9	35.0
58	rusty	10	35.0

Union

- All of these operations take two input relations, which must be ***compatible***:
  - Same number of fields.
  - "*Corresponding*" fields have the same type.
- What is the *schema* of result?

$S1 \cup S2$ → sid, sname, rating , age

S1

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

S2

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

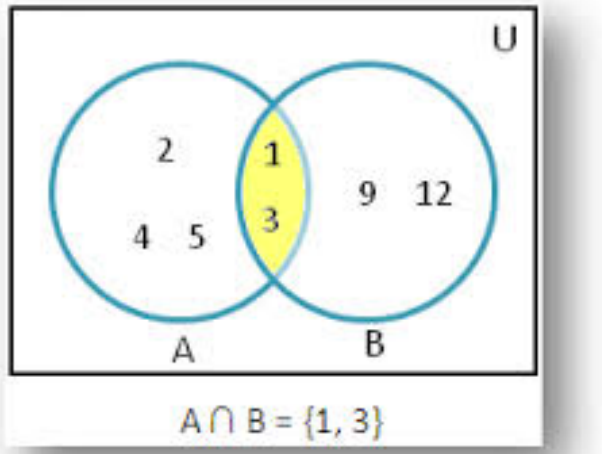
$S1 \cup S2$

sid	sname	rating	age
22	dustin	7	45.0
31	lubber	8	55.5
58	rusty	10	35.0
44	guppy	5	35.0
28	yuppy	9	35.0

Intersection

- Produces a resulting relation that contains a tuple for every tuple in BOTH of the two input relations
- The relations being combined must be **compatible** (type-compatible)

Set theory



Set-Difference (Minus)

- All of these operations take two input relations, which must be ***compatible***:
  - Same number of fields.
  - "Corresponding" fields have the same type.
- What is the *schema* of result?

$S1 - S2$ → sid, sname, rating , age

S1

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

S2

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

Cross-Product (Times)

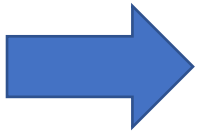
- Each row of S1 is paired with each row of R1.
- *Result schema* has one field per field of S1 and R1, with field names "*inherited*" if possible.
  - *Conflict*: Both S1 and R1 have a field called *sid*.

S1

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

R1

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



(sid)	sname	rating	age	(sid)	bid	day

Cross-Product (Times)

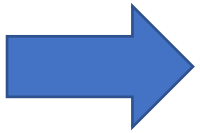
- Each row of S1 is paired with each row of R1.
- *Result schema* has one field per field of S1 and R1, with field names "*inherited*" if possible.
  - *Conflict*: Both S1 and R1 have a field called *sid*.

S1

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

R1

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



(sid)	sname	rating	age	(sid)	bid	day
22	dustin	7	45.0	22	101	10/10/96

Cross-Product (Times)

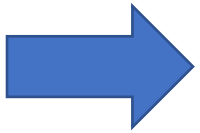
- Each row of S1 is paired with each row of R1.
- *Result schema* has one field per field of S1 and R1, with field names "*inherited*" if possible.
  - *Conflict*: Both S1 and R1 have a field called *sid*.

S1

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

R1

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



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

Cross-Product (Times)

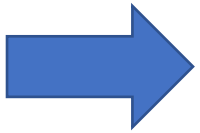
- Each row of S1 is paired with each row of R1.
- *Result schema* has one field per field of S1 and R1, with field names "*inherited*" if possible.
  - *Conflict*: Both S1 and R1 have a field called *sid*.

S1

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

R1

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



(sid)	sname	rating	age	(sid)	bid	day
22	dustin	7	45.0	22	101	10/10/96
22	dustin	7	45.0	58	103	11/12/96
31	lubber	8	55.5	22	101	10/10/96

Cross-Product (Times)

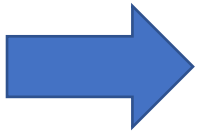
- Each row of S1 is paired with each row of R1.
- *Result schema* has one field per field of S1 and R1, with field names "*inherited*" if possible.
  - *Conflict*: Both S1 and R1 have a field called *sid*.

S1

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

R1

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



(sid)	sname	rating	age	(sid)	bid	day
22	dustin	7	45.0	22	101	10/10/96
22	dustin	7	45.0	58	103	11/12/96
31	lubber	8	55.5	22	101	10/10/96
31	lubber	8	55.5	58	103	11/12/96

Cross-Product (Times)

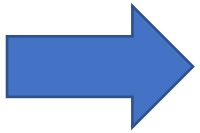
- Each row of S1 is paired with each row of R1.
- *Result schema* has one field per field of S1 and R1, with field names "*inherited*" if possible.
  - *Conflict*: Both S1 and R1 have a field called *sid*.

S1

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

R1

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



(sid)	sname	rating	age	(sid)	bid	day
22	dustin	7	45.0	22	101	10/10/96
22	dustin	7	45.0	58	103	11/12/96
31	lubber	8	55.5	22	101	10/10/96
31	lubber	8	55.5	58	103	11/12/96
58	rusty	10	35.0	22	101	10/10/96

Cross-Product (Times)

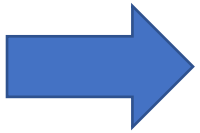
- Each row of S1 is paired with each row of R1.
- *Result schema* has one field per field of S1 and R1, with field names "*inherited*" if possible.
  - *Conflict*: Both S1 and R1 have a field called *sid*.

S1

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

R1

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



(sid)	sname	rating	age	(sid)	bid	day
22	dustin	7	45.0	22	101	10/10/96
22	dustin	7	45.0	58	103	11/12/96
31	lubber	8	55.5	22	101	10/10/96
31	lubber	8	55.5	58	103	11/12/96
58	rusty	10	35.0	22	101	10/10/96
58	rusty	10	35.0	58	103	11/12/96

Renaming operator: $\rho\,(C(1 \to sid1, 5 \to sid2),\ S1 \times R1)$

Joins

- ***Condition Join***: $R \bowtie_C S = \sigma_C (R \times S)$

Joins

- ***Condition Join***: $R \bowtie_C S = \sigma_C (R \times S)$

- ***Result schema*** same as that of cross-product.
- Fewer tuples than cross-product, might be able to compute more efficiently
- Sometimes called a ***theta-join***.

Joins

- ***Condition Join***: $R \bowtie_C S = \sigma_C (R \times S)$

	(sid)	sname	rating	age	(sid)	bid	day
X	22	dustin	7	45.0	22	101	10/10/96
	22	dustin	7	45.0	58	103	11/12/96
X	31	lubber	8	55.5	22	101	10/10/96
	31	lubber	8	55.5	58	103	11/12/96
X	58	rusty	10	35.0	22	101	10/10/96
X	58	rusty	10	35.0	58	103	11/12/96

(sid)	sname	rating	age	(sid)	bid	day
22	dustin	7	45.0	58	103	11/12/96
31	lubber	8	55.5	58	103	11/12/96

$$S1 \bowtie_{S1.sid < R1.sid} R1$$

- ***Result schema*** same as that of cross-product.
- Fewer tuples than cross-product, might be able to compute more efficiently
- Sometimes called a ***theta-join***.

Joins

- ***Equi-Join***: A special case of condition join where the condition *c* contains only ***equalities*** and ^.

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

$$S1 \bowtie_{sid} R1$$

- ***Result schema*** similar to cross-product, but only one copy of fields for which equality is specified.
- ***Natural Join***: Equijoin on *all* common fields.

Join vs. Cross-Product

- R ⋈ S is very common and thus must be carefully optimized
- R x S followed by a selection is inefficient because the cross-product is large because the cross-product is large

Nested Loops Joins

Tuple-based nested loop $R \bowtie S$

```
For each tuple r in R do
    For each tuple s in S do
        If r and s join then output (r, s)
```

Cost: T(R) * T(S)

T(R): Number of tuples in R
T(S) Number of tuples in S

Relational Algebra and Queries

Sailors-Boats-reservations

Popular BD example

DB Example

Sid	Sname	Rating	Age
22	Dustin	7	45.0
29	Brutus	1	33.0
31	Lubber	8	55.5
32	Andy	8	25.5
58	Rusty	10	35.0
64	Horatio	7	35.0
71	Zorba	10	16.0
74	Horatio	9	35.0
85	Art	3	25.5
95	Bob	3	63.5

An Instance S3 of Sailors

Sid	Bid	Day
22	101	10/10/98
22	102	10/10/98
22	103	10/8/98
22	104	10/7/98
31	102	11/10/98
31	103	11/6/98
31	104	11/12/98
64	101	9/5/98
64	102	9/8/98
74	103	9/8/98

An Instance R2 of Reserves

Bid	Bname	Color
101	Interlake	Blue
102	Interlake	Red
103	Clipper	Green
104	Marine	Red

An Instance B1 of Boats

Find names of sailors who've reserved boat #103

Sid	Sname	Rating	Age
22	Dustin	7	45.0
29	Brutus	1	33.0
31	Lubber	8	55.5
32	Andy	8	25.5
58	Rusty	10	35.0
64	Horatio	7	35.0
71	Zorba	10	16.0
74	Horatio	9	35.0
85	Art	3	25.5
95	Bob	3	63.5

An Instance S3 of Sailors

Sid	Bid	Day
22	101	10/10/98
22	102	10/10/98
22	103	10/8/98
22	104	10/7/98
31	102	11/10/98
31	103	11/6/98
31	104	11/12/98
64	101	9/5/98
64	102	9/8/98
74	103	9/8/98

An Instance R2 of Reserves

Bid	Bname	Color
101	Interlake	Blue
102	Interlake	Red
103	Clipper	Green
104	Marine	Red

An Instance B1 of Boats

Find names of sailors who've reserved boat #103

Sid	Sname	Rating	Age
22	Dustin	7	45.0
29	Brutus	1	33.0
31	Lubber	8	55.5
32	Andy	8	25.5
58	Rusty	10	35.0
64	Horatio	7	35.0
71	Zorba	10	16.0
74	Horatio	9	35.0
85	Art	3	25.5
95	Bob	3	63.5

An Instance S3 of Sailors

Sid	Bid	Day
22	101	10/10/98
22	102	10/10/98
22	103	10/8/98
22	104	10/7/98
31	102	11/10/98
31	103	11/6/98
31	104	11/12/98
64	101	9/5/98
64	102	9/8/98
74	103	9/8/98

An Instance R2 of Reserves

Bid	Bname	Color
101	Interlake	Blue
102	Interlake	Red
103	Clipper	Green
104	Marine	Red

An Instance B1 of Boats

$$\sigma_{bid=103} Reserves$$

Find names of sailors who've reserved boat #103

Sid	Sname	Rating	Age
22	Dustin	7	45.0
29	Brutus	1	33.0
31	Lubber	8	55.5
32	Andy	8	25.5
58	Rusty	10	35.0
64	Horatio	7	35.0
71	Zorba	10	16.0
74	Horatio	9	35.0
85	Art	3	25.5
95	Bob	3	63.5

An Instance S3 of Sailors

Sid	Bid	Day
22	101	10/10/98
22	102	10/10/98
22	103	10/8/98
22	104	10/7/98
31	102	11/10/98
31	103	11/6/98
31	104	11/12/98
64	101	9/5/98
64	102	9/8/98
74	103	9/8/98

An Instance R2 of Reserves

Bid	Bname	Color
101	Interlake	Blue
102	Interlake	Red
103	Clipper	Green
104	Marine	Red

An Instance B1 of Boats

$$\sigma_{bid=103} Reserves$$

Find names of sailors who've reserved boat #103

Sid	Sname	Rating	Age
22	Dustin	7	45.0
29	Brutus	1	33.0
31	Lubber	8	55.5
32	Andy	8	25.5
58	Rusty	10	35.0
64	Horatio	7	35.0
71	Zorba	10	16.0
74	Horatio	9	35.0
85	Art	3	25.5
95	Bob	3	63.5

An Instance S3 of Sailors

Sid	Bid	Day
22	101	10/10/98
22	102	10/10/98
22	103	10/8/98
22	104	10/7/98
31	102	11/10/98
31	103	11/6/98
31	104	11/12/98
64	101	9/5/98
64	102	9/8/98
74	103	9/8/98

An Instance R2 of Reserves

Bid	Bname	Color
101	Interlake	Blue
102	Interlake	Red
103	Clipper	Green
104	Marine	Red

An Instance B1 of Boats

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

Find names of sailors who've reserved boat #103

Solution 1:

$$\pi_{sname}((\sigma_{bid=103}\,\mathrm{Re}serves) \bowtie Sailors)$$

Find names of sailors who've reserved boat #103

Solution 1:

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

An Instance R2 of Reserves

Sid	Bid	Day
22	101	10/10/98
22	102	10/10/98
22	103	10/8/98
22	104	10/7/98
31	102	11/10/98
31	103	11/6/98
31	104	11/12/98
64	101	9/5/98
64	102	9/8/98
74	103	9/8/98

Find names of sailors who've reserved boat #103

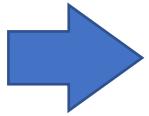
Solution 1:

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

Sid	Bid	Day
22	103	10/8/98
31	103	11/6/98
31	104	11/12/98
74	103	9/8/98

$\bowtie$

Sid	Sname	Rating	Age
22	Dustin	7	45.0
29	Brutus	1	33.0
31	Lubber	8	55.5
32	Andy	8	25.5
58	Rusty	10	35.0
64	Horatio	7	35.0
71	Zorba	10	16.0
74	Horatio	9	35.0
85	Art	3	25.5
95	Bob	3	63.5



Sid	Bib	Day	Sname	Rating	Age
22	103	10/8/98	Dustin	7	45
31	103	11/6/98	Lubber	8	55.5
74	103	9/8/98	Horatio	9	35.0

Find names of sailors who've reserved boat #103

Solution 1:

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

Sname
Dustin
Lubber
Horatio

Find names of sailors who've reserved boat #103

Solution 1:

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

Solution 2:

$$\rho\ (Temp1,\ \sigma_{bid=103} Reserves)$$

$$\rho\ (Temp2,\ Temp1 \bowtie Sailors)$$

$$\pi_{sname}\ (Temp2)$$

Find names of sailors who've reserved boat #103

Solution 3: $\pi_{sname}(\sigma_{bid=103}(Reserves \bowtie Sailors))$

Sid	Sname	Rating	Age
22	Dustin	7	45.0
29	Brutus	1	33.0
31	Lubber	8	55.5
32	Andy	8	25.5
58	Rusty	10	35.0
64	Horatio	7	35.0
71	Zorba	10	16.0
74	Horatio	9	35.0
85	Art	3	25.5
95	Bob	3	63.5

An Instance S3 of Sailors

Sid	Bid	Day
22	101	10/10/98
22	102	10/10/98
22	103	10/8/98
22	104	10/7/98
31	102	11/10/98
31	103	11/6/98
31	104	11/12/98
64	101	9/5/98
64	102	9/8/98
74	103	9/8/98

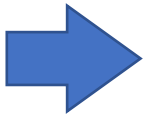
An Instance R2 of Reserves

Sid	Sname	Rating	Age	Sid	Bid	Day
22	Dustin	7	45.0	22	101	10/10/98
22	Dustin	7	45.0	22	102	10/10/98
22	Dustin	7	45.0	22	103	10/8/98
22	Dustin	7	45.0	22	104	10/7/98
31	Lubber	8	55.5	31	102	11/10/98
31	Lubber	8	55.5	31	103	11/6/98
…	…	…	…	…	…	…
64	Horatio	7	35.0	64	101	9/5/98
64	Horatio	7	35.0	64	102	9/8/98
74	Horatio	9	35.0	74	103	9/8/98

Find names of sailors who've reserved boat #103

Solution 3: $\pi_{sname}(\sigma_{bid=103}(Reserves \bowtie Sailors))$

Sid	Sname	Rating	Age	Sid	Bid	Day
22	Dustin	7	45.0	22	101	10/10/98
22	Dustin	7	45.0	22	102	10/10/98
22	Dustin	7	45.0	22	103	10/8/98
22	Dustin	7	45.0	22	104	10/7/98
31	Lubber	8	55.5	31	102	11/10/98
31	Lubber	8	55.5	31	103	11/6/98
…	…	…	…	…	…	…
64	Horatio	7	35.0	64	101	9/5/98
64	Horatio	7	35.0	64	102	9/8/98
74	Horatio	9	35.0	74	103	9/8/98

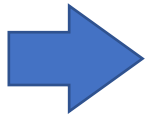


Sid	Sname	Rating	Age	Sid	Bid	Day
22	Dustin	7	45.0	22	103	10/8/98
31	Lubber	8	55.5	31	103	11/6/98
74	Horatio	9	35.0	74	103	9/8/98

Find names of sailors who've reserved boat #103

Solution 3: $\pi_{sname}(\sigma_{bid=103}(Reserves \bowtie Sailors))$

Sid	Sname	Rating	Age	Sid	Bid	Day
22	Dustin	7	45.0	22	103	10/8/98
31	Lubber	8	55.5	31	103	11/6/98
74	Horatio	9	35.0	74	103	9/8/98



Sname
Dustin
Lubber
Horatio

Find names of sailors who've reserved boat #103

Solution 1:

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

Solution 3:

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

Which one to choose?
and why?
Think about this!

Outer Join

In an outer join, along with tuples that satisfy the matching criteria, we also include some or all tuples that do not match the criteria

- Left Outer join
- Right Outer Join
- Full Outer Join

Left Outer Join

- In the left outer join, operation allows keeping all tuple in the left relation.
- If there is no matching tuple is found in right relation, then the attributes of right relation in the join result are filled with null values.

Left Outer Join ⟕

A

Num	Square
2	4
3	9
4	16

B

Num	Cube
2	8
3	18
5	75

A ⟕ B

Num	Square	Cube
2	4	8
3	9	18
4	16	NULL

- In the left outer join, operation allows keeping all tuple in the left relation.
- If there is no matching tuple is found in right relation, then the attributes of right relation in the join result are filled with null values.

Left Outer Join ⟕

A

Num	Square
2	4
3	9
4	16

B

Num	Cube
2	8
3	18
5	75

A ⟕ B

Num	Square	Cube
2	4	8
3	9	18
4	16	NULL

SELECT A.num, A.square, B.cube1 **FROM** A **LEFT JOIN** B **ON** A.num = B.num;

Right Outer Join

- In the right outer join, operation allows keeping all tuple in the right relation
- If there is no matching tuple is found in the left relation, then the attributes of the left relation in the join result are filled with null values

Right Outer Join ⟖

A

Num	Square
2	4
3	9
4	16

B

Num	Cube
2	8
3	18
5	75

A ⟖ B

Num	Cube	Square
2	8	4
3	18	9
5	75	NULL

- In the right outer join, operation allows keeping all tuple in the right relation
- If there is no matching tuple is found in the left relation, then the attributes of the left relation in the join result are filled with null values

Right Outer Join ⟖

A

Num	Square
2	4
3	9
4	16

B

Num	Cube
2	8
3	18
5	75

A ⟖ B

Num	Cube	Square
2	8	4
3	18	9
5	75	NULL

SELECT B.num, B.cube1, A.square **FROM** A **RIGHT JOIN** B **ON** A.num = B.num;

Full Outer Joins ⟗

A

Num	Square
2	4
3	9
4	16

B

Num	Cube
2	8
3	18
5	75

A ⟗ B

Num	Square	Cube
2	4	8
3	9	18
4	16	NULL
5	NULL	75

In a full outer join, all tuples from both relations are included in the result, irrespective of the matching condition.

As far as I understand MySQL does not support this operation ☹