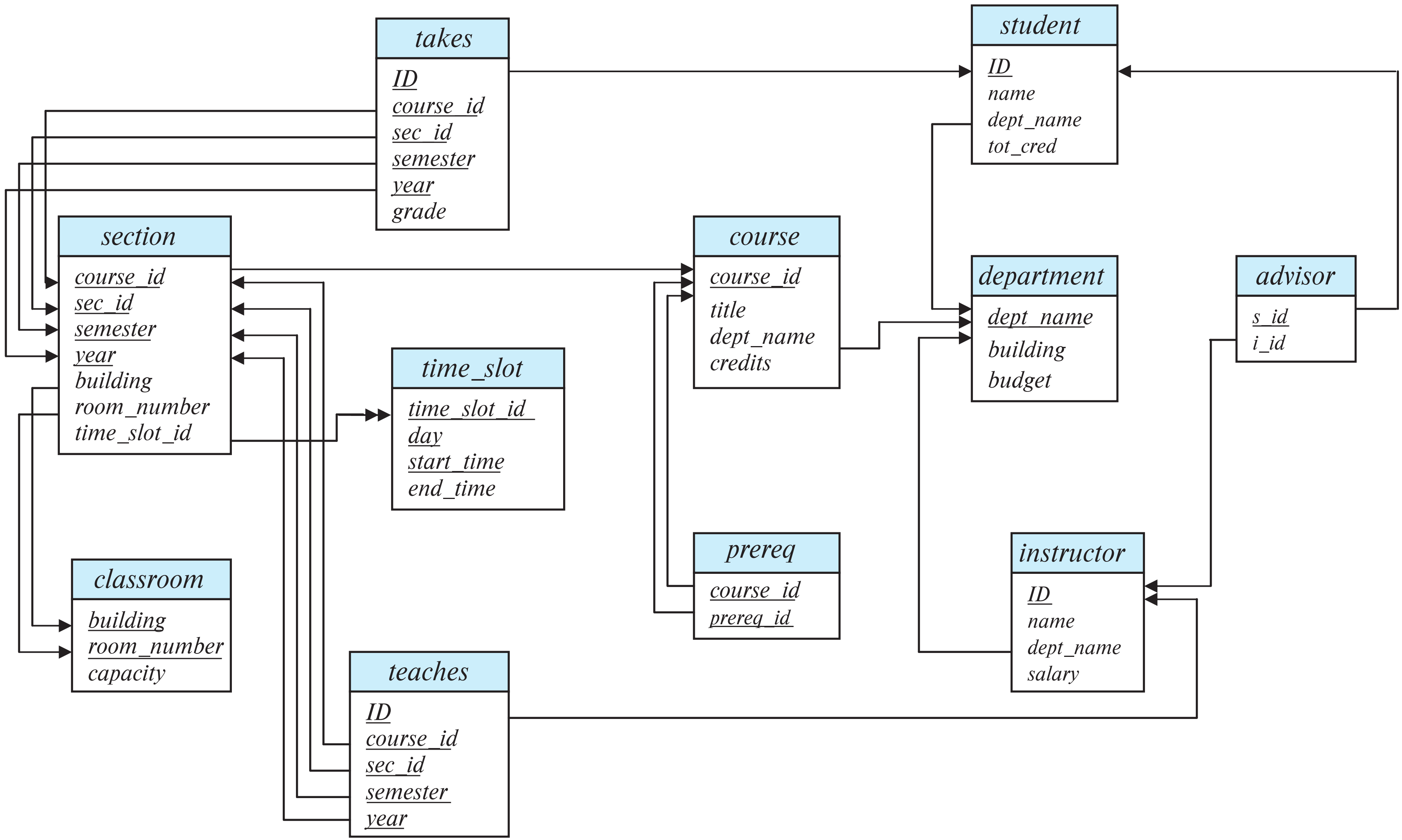
Relational Algebra

Miao Qiao
The University of Auckland

Recap: Schema for University Database



Relational Algebra

- Simplest query: relation name

instructor

ID	*name*	*dept_name*	*salary*
22222	Einstein	Physics	95000
12121	Wu	Finance	90000
32343	El Said	History	60000
45565	Katz	Comp. Sci.	75000
98345	Kim	Elec. Eng.	80000
76766	Crick	Biology	72000
10101	Srinivasan	Comp. Sci.	65000
58583	Califieri	History	62000
83821	Brandt	Comp. Sci.	92000
15151	Mozart	Music	40000
33456	Gold	Physics	87000
76543	Singh	Finance	80000

(a) The *instructor* table

- Use operators to filter, slice, combine

Select operator: pick certain rows

- Select *instructors* where the instructor is in the "Physics" department.

$$\sigma_{\textit{dept\_name}=\text{“Physics”}}(\textit{instructor})$$

ID	*name*	*dept_name*	*salary*
22222	Einstein	Physics	95000
33456	Gold	Physics	87000

- Select instructors in Physics depart with salary > 90000.

$$\sigma_{\textit{dept\_name}=\text{“Physics”} \wedge \textit{salary} > 90000}(\textit{instructor})$$

- $\sigma_{cond} Rel$
- Comparisons: =, ≠, >, ≥. <. ≤
- Logic connectives: ∧ (and), ∨ (or), ¬ (not)

ID	*name*	*dept_name*	*salary*
22222	Einstein	Physics	95000
12121	Wu	Finance	90000
32343	El Said	History	60000
45565	Katz	Comp. Sci.	75000
98345	Kim	Elec. Eng.	80000
76766	Crick	Biology	72000
10101	Srinivasan	Comp. Sci.	65000
58583	Califieri	History	62000
83821	Brandt	Comp. Sci.	92000
15151	Mozart	Music	40000
33456	Gold	Physics	87000
76543	Singh	Finance	80000

(a) The *instructor* table

Project operator: pick certain columns

- Pick ID, name and salary of *instructor*

$$\Pi_{ID,\ name,\ salary}\ (instructor)$$

- Result:

ID	*name*	*salary*
10101	Srinivasan	65000
12121	Wu	90000
15151	Mozart	40000
22222	Einstein	95000
32343	El Said	60000
33456	Gold	87000
45565	Katz	75000
58583	Califieri	62000
76543	Singh	80000
76766	Crick	72000
83821	Brandt	92000
98345	Kim	80000

- $$\prod_{A_1, A_2, A_3 \dots A_k} (r)$$

ID	*name*	*dept_name*	*salary*
22222	Einstein	Physics	95000
12121	Wu	Finance	90000
32343	El Said	History	60000
45565	Katz	Comp. Sci.	75000
98345	Kim	Elec. Eng.	80000
76766	Crick	Biology	72000
10101	Srinivasan	Comp. Sci.	65000
58583	Califieri	History	62000
83821	Brandt	Comp. Sci.	92000
15151	Mozart	Music	40000
33456	Gold	Physics	87000
76543	Singh	Finance	80000

(a) The *instructor* table

Compose operator: pick both rows and columns

- Find the names of all instructors in the Physics department.

$$\sigma_{\,dept\_name\ =\text{“}Physics\text{”}}\ (instructor)$$

$$\Pi_{name}(\sigma_{\,dept\_name\ =\text{“}Physics\text{”}}\ (instructor))$$

- $\Pi_{A1,\ A2}\ Expr$
- $\sigma_{cond}\ Expr$

ID	*name*	*dept_name*	*salary*
22222	Einstein	Physics	95000
12121	Wu	Finance	90000
32343	El Said	History	60000
45565	Katz	Comp. Sci.	75000
98345	Kim	Elec. Eng.	80000
76766	Crick	Biology	72000
10101	Srinivasan	Comp. Sci.	65000
58583	Califieri	History	62000
83821	Brandt	Comp. Sci.	92000
15151	Mozart	Music	40000
33456	Gold	Physics	87000
76543	Singh	Finance	80000

(a) The *instructor* table

Cross-product: combines two relations (a.k.a Cartesian-product)

instructor X *teaches*

instructor.ID

instructor
ID
name
dept_name
salary

teaches
ID
course_id
sec_id
semester
year

teaches.id

instructor.ID	*name*	*dept_name*	*salary*	*teaches.ID*	*course_id*	*sec_id*	*semester*	*year*
10101	Srinivasan	Comp. Sci.	65000	10101	CS-101	1	Fall	2017
10101	Srinivasan	Comp. Sci.	65000	10101	CS-315	1	Spring	2018
10101	Srinivasan	Comp. Sci.	65000	10101	CS-347	1	Fall	2017
10101	Srinivasan	Comp. Sci.	65000	12121	FIN-201	1	Spring	2018
10101	Srinivasan	Comp. Sci.	65000	15151	MU-199	1	Spring	2018
10101	Srinivasan	Comp. Sci.	65000	22222	PHY-101	1	Fall	2017
...	...	...	...	...	...	...	...	...
...	...	...	...	...	...	...	...	...
12121	Wu	Finance	90000	10101	CS-101	1	Fall	2017
12121	Wu	Finance	90000	10101	CS-315	1	Spring	2018
12121	Wu	Finance	90000	10101	CS-347	1	Fall	2017
12121	Wu	Finance	90000	12121	FIN-201	1	Spring	2018

Cross-product: combine two relations (a.k.a Cartesian-product)

instructor X *teaches*

- Find the instructors and the courses that they taught

instructor
ID
name
dept_name
salary

teaches
ID
course_id
sec_id
semester
year

$$\sigma_{\textit{instructor.id} = \textit{teaches.id}} (\textit{instructor} \times \textit{teaches})$$

Natural join

- Combine two relations, enforce equality on all attributes with same name
- Eliminate the copy of duplicated attributes

instructor
ID
name
dept_name
salary

$$instructor \bowtie teaches$$

teaches
ID
course_id
sec_id
semester
year

- Find all the names of instructors whose department building is 303 and who have taught a course in 2024.

department
dept_name
building
budget

$$\Pi_{name}\left(\sigma_{building\ =\ '303'\ \wedge\ year\ =\ 2024}\ instructor \bowtie teaches \bowtie \text{department}\right)$$

Theta join

- $Exp_1 \bowtie_\theta Exp_2$
  - θ denotes the selection condition
  - Equivalent to $\sigma_\theta(Exp_1 \bowtie Exp_2)$

- Basic operation implemented in DBMS

Wrap up

- Data definition language
- Basic steps in creating and using relational DB
- Data manipulation language
- Relational algebra
  - Simplest query: relation name
  - Use operators to filter, slice, combine
  - Operators so far: select, project, cross-product, natural join, theta join

Query Quest

employee (*ID*, *person_name*, *street*, *city*)

works (*ID*, *person_name*, *company_name*, *salary*)

company (*company_name*, *city*)

- Find the Query
  a. Find the ID and name of each employee who works for "BigBank".
  b. Find the ID, name, and city of residence of each employee who works for "BigBank".
  c. Find the ID, name, street address, and city of residence of each employee who works for "BigBank" and earns more than $10000 per year.

Set operation

- Union Operation
  - Find the names of instructors and the names of the departments

$$\Pi_{name} instructor \cup \Pi_{dept\_name} instructor$$

- Intersection operation
  - Find names that are both an instructor name and a department name

$$\Pi_{name} instructor \cap \Pi_{dept\_name} department$$

instructor
ID
name
dept_name
salary

department
dept_name
building
budget

- Set difference operation
  - Find the names of the departments who has no instructors

$$\Pi_{dept\_name} department - \Pi_{dept\_name} instructor$$

ID	*name*	*dept_name*	*salary*
22222	Einstein	Physics	95000
12121	Wu	Finance	90000
32343	El Said	History	60000
45565	Katz	Comp. Sci.	75000
98345	Kim	Elec. Eng.	80000
76766	Crick	Biology	72000
10101	Srinivasan	Comp. Sci.	65000
58583	Califieri	History	62000
83821	Brandt	Comp. Sci.	92000
15151	Mozart	Music	40000
33456	Gold	Physics	87000
76543	Singh	Finance	80000

(a) The *instructor* table

The Rename Operation

- Rename operator, ρ, name and format the results of an expression
- $\rho_{x(A1,A2,\ ..\ An)}\ (E)$ general form
  - $\rho_x\ (E)$
  - $\rho_{A1,A2,\ ..\ An}\ (E)$
- Functions:
  - To unify schemas for set operations
    - Find names that are both an instructor name and a department name

$$\Pi_{name} instructor \cap \Pi_{dept\_name}\ instructor$$

$$\rho_{x(A)}\ (\Pi_{name} instructor) \cap \rho_{x(A)}\ (\Pi_{dept\_name}\ instructor)$$

The Rename Operation

- Rename operator, ρ, name and format the results of an expression
- General form $\rho_{x(A1,A2,\ ..\ An)}\ (E)$
  - $\rho_x\ (E)$
  - $\rho_{A1,A2,\ ..\ An}\ (E)$
- Functions:
  - Unify schemas for set operations
  - For disambiguation in "self-joins"
    - Find pairs of instructors in the same dept
      - Instructor x instructor ?
      - $\rho_{r1(a1,b1,c1,d1)}$ Instructor x $\rho_{r2(a2,b2,c2,d2)}$ Instructor
      - $\sigma_{r1.c1\ =\ r2.c2}$
      - $\rho_{r1(a1,b1,c,d1)}$ Instructor $\bowtie$ $\rho_{r2(a2,b2,c,d2)}$ Instructor

ID	*name*	*dept_name*	*salary*
22222	Einstein	Physics	95000
12121	Wu	Finance	90000
32343	El Said	History	60000
45565	Katz	Comp. Sci.	75000
98345	Kim	Elec. Eng.	80000
76766	Crick	Biology	72000
10101	Srinivasan	Comp. Sci.	65000
58583	Califieri	History	62000
83821	Brandt	Comp. Sci.	92000
15151	Mozart	Music	40000
33456	Gold	Physics	87000
76543	Singh	Finance	80000

(a) The *instructor* table

Assignment Operation

- Break down relational algebra expressions to their parts
- Find all instructor in the “Physics” and Music department.

$$Physics \leftarrow \sigma_{dept\_name=\text{“Physics”}}(instructor)$$

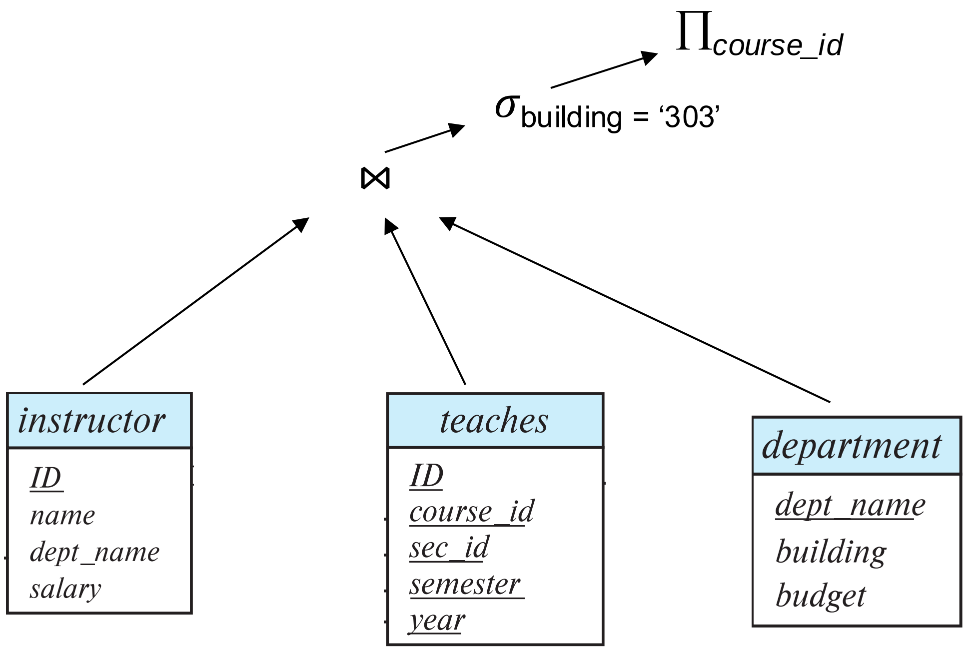
$$Music \leftarrow \sigma_{dept\_name=\text{“Music”}}(instructor)$$

$$Physics \cup Music$$

- The assignment operation is denoted by $\leftarrow$ and works like assignment in a programming language.

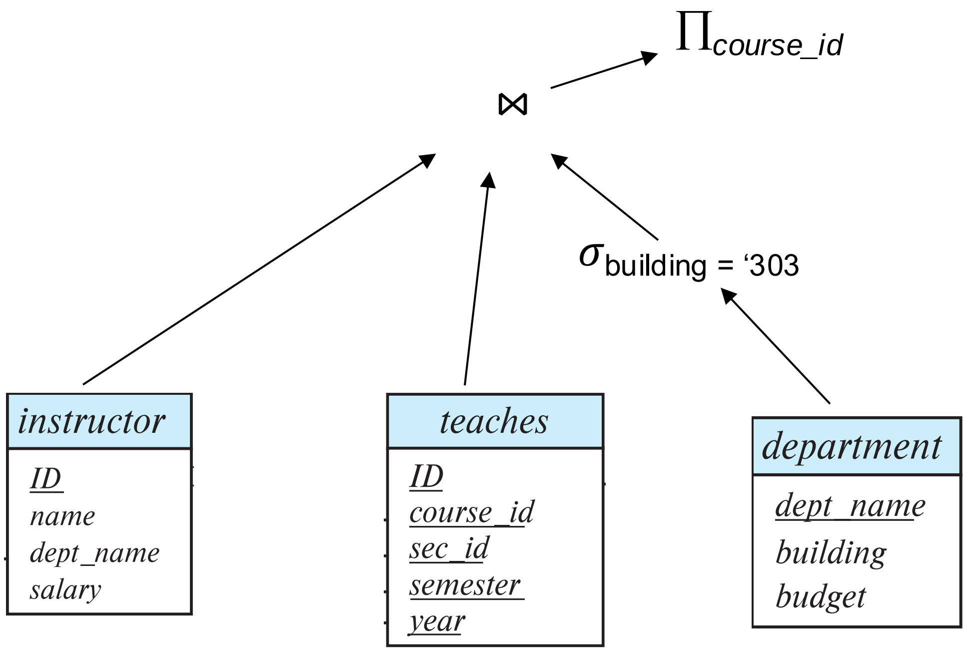
Expression tree

- Find the course IDs whose instructors are in building 303



Expression tree

- Find the course IDs whose instructors are in building 303



- The two queries are not identical; they are, however, equivalent -- they give the same result on any database.

Relational Algebra

- A query language of a set of operations that take one or two relations as input and produce a new relation as their result.
- Operators
  - select: σ
  - project: $\prod$
  - union: $\cup$
  - set difference: –
  - Cartesian product: x
    - Natural Join $\bowtie$
    - Theta join $\bowtie_\theta$
  - rename: ρ
  - assignment : $\leftarrow$
- Expression tree

Wrap up

- Relational algebra
  - Simplest query: relation name
  - Use operators to filter, slice, combine
  - Operators: select, project, cross-product, natural join, theta join
    - Set operation, assignment, rename
  - Expression tree

Query Quest

employee (*ID*, *person_name*, *street*, *city*)

works (*ID*, *person_name*, *company_name*, *salary*)

company (*company_name*, *city*)

- Find the Query
  - Find the ID and name of each employee in this database who lives in the same city as the company for which she or he works.

FIN

Any questions?