

CSE460/560 DATA MODELS AND QUERY LANGUAGES

Relational Algebra - 2

Cheng-En Chuang

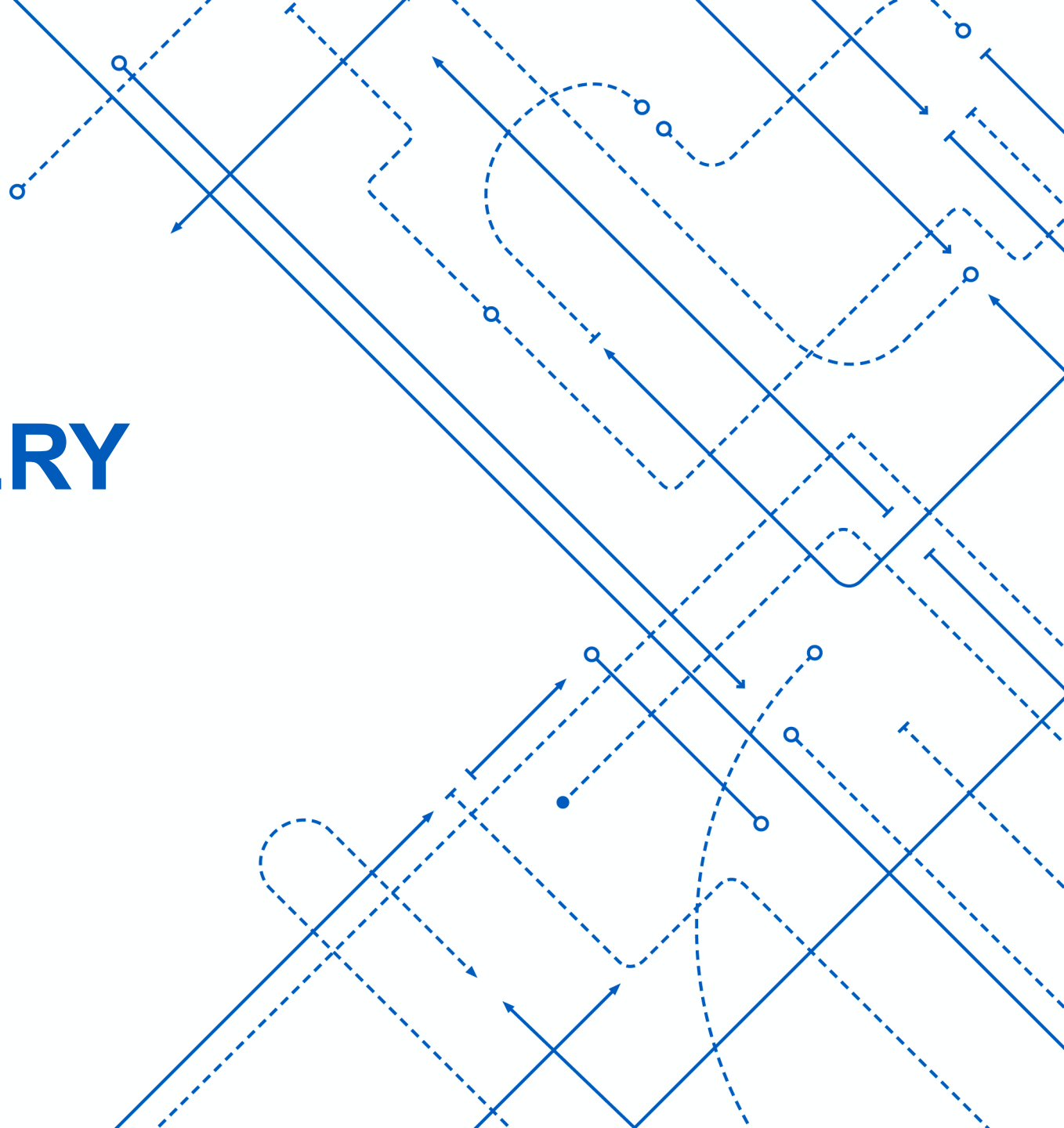
(Slides Adopted from Jan Chomicki and Ning Deng)



University at Buffalo

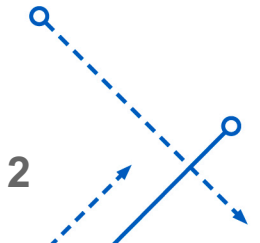
Department of Computer Science
and Engineering

School of Engineering and Applied Sciences



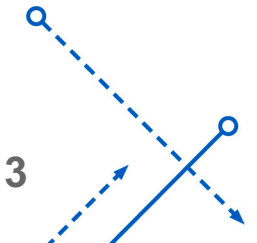
Outline

1. Projection π
2. Selection σ
3. Composability
4. Union \cup
5. Set Difference $(-)$
6. Cross Product (\times)
7. Renaming (ρ)



Outline

1. **Projection π**
2. Selection σ
3. Composability
4. Union \cup
5. Set Difference $(-)$
6. Cross Product (\times)
7. Renaming (ρ)



Example Instances

Student

<u>FirstName,</u>	GPA,	SID
[James,	3.9,	1701]
[Jean,	3.9,	1702]
[John,	3.0,	1703]
[Mary,	4.0,	1801]
[Mike,	4.0,	1805]

Club

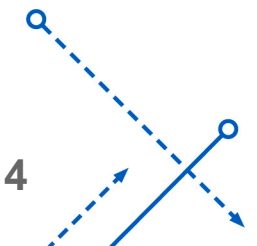
<u>President,</u>	Name
[James,	C1]
[Mary,	C2]
[Tom,	C3]

MajorsIn

<u>SID,</u>	MID
[1701,	01]
[1701,	02]
[1805,	03]
[1801,	04]

Major

<u>MID,</u>	Name
[01,	CS]
[02,	EE]
[03,	Math]
[04,	ME]

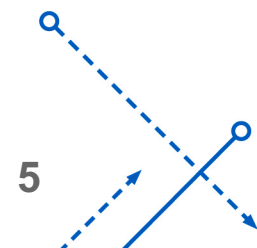


Projection π

- Pick columns A_1, \dots, A_k which are distinct attributes of relation R , from R
 - $\text{arity}(\pi_{A_1, \dots, A_k}(R)) = k$
 - A tuple $t \in \pi_{A_1, \dots, A_k}(R)$ iff for some $s \in R$
 - $t[A_1, \dots, A_k] = s[A_1, \dots, A_k]$
- Example

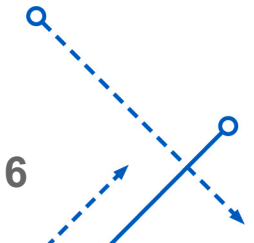
- $\pi_{\text{firstname}, \text{gpa}}(\text{Student})$
- How about $\pi_{\text{gpa}}(\text{Student})$?

<u>FirstName,</u>	<u>GPA</u>
[James,	3.9]
[Jean,	3.9]
[John,	3.0]
[Mary,	4.0]
[Mike,	4.0]



Outline

1. Projection π
- 2. Selection σ**
3. Composability
4. Union \cup
5. Set Difference $(-)$
6. Cross Product (\times)
7. Renaming (ρ)



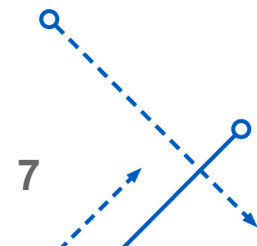
Selection σ

- Select tuples/rows that satisfy the selection condition c from relation R
 - c is a condition on attributes of R and c is built from
 - Comparisons between operands which can be constants or attribute names
 - Boolean operators: \wedge (*AND*), \vee (*OR*), \neg (*NOT*)
 - $arity(\sigma_c(R)) = arity(R)$
 - A tuple $t \in \sigma_c(R)$ iff $t \in R$ and t satisfies c
- Example

- $\sigma_{gpa < 3.5}(Student)$

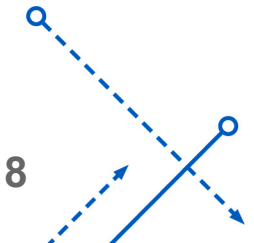
- $\sigma_{gpa < 3.5} \left(\begin{array}{c} \text{Student} \\ \hline \text{FirstName, GPA, SID} \\ \hline \{ \text{James, 3.9, 1701} \} \\ \{ \text{Jean, 3.9, 1702} \} \\ \{ \text{John, 3.0, 1703} \} \\ \{ \text{Mary, 4.0, 1801} \} \\ \{ \text{Mike, 4.0, 1805} \} \end{array} \right) = \begin{array}{c} \text{FirstName, GPA, SID} \\ \hline \{ \text{John, 3.0, 1703} \} \end{array}$

- When does selection need to eliminate duplicates?



Outline

1. Projection π
2. Selection σ
- 3. Composability**
4. Union \cup
5. Set Difference $(-)$
6. Cross Product (\times)
7. Renaming (ρ)



Composing σ and π

- $\pi_{firstname}(\sigma_{GPA>3.5}(Student))$

- $\pi_{firstname}(\sigma_{GPA>3.5}(\text{Student}))$

FirstName	GPA	SID
James	3.9	1701
Jean	3.9	1702
John	3.0	1703
Mary	4.0	1801
Mike	4.0	1805

- Then?

- The is the schema of result of this query?

FirstName

[James]

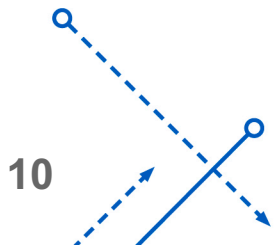
[Jane]

[Mary]

[Mike]

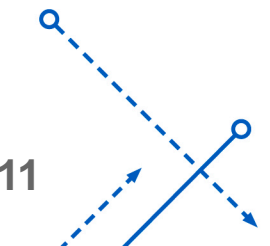
Outline

1. Projection π
2. Selection σ
3. Composability
4. **Union \cup**
5. Set Difference $(-)$
6. Cross Product (\times)
7. Renaming (ρ)



Union \cup

- Takes two **compatible** relations
 - Returns all tuples in either relation
- Property
 - $arity(R_1 \cup R_2) = arity(R_1) = arity(R_2)$
 - $t \in R_1 \cup R_2 \text{ iff } t \in R_1 \text{ or } t \in R_2$
- Compatibility
 - $arity(R_1) = arity(R_2)$
 - The corresponding attribute domain in R_1 and R_2 are the same
 - Thus compatibility of two relations can be determined solely on their schemas
 - Can we do $Student \cup Club$?
 - 📍 How about $\pi_{firstname}(\sigma_{gpa=4.0}(Student)) \cup \sigma_{gpa<3.5}(Student)$?
 - What is this query doing?



Outline

1. Projection π
2. Selection σ
3. Composability
4. Union \cup
- 5. Set Difference $-$**
6. Cross Product \times
7. Renaming ρ



Set Difference (−)

- Takes two compatible relations
 - Compute all tuples that in the first relation
 - But not in the second relation
- Property
 - $arity(R_1 - R_2) = arity(R_1) = arity(R_2)$
 - $t \in R_1 - R_2 \text{ iff } t \in R_1 \wedge t \notin R_2$
- Example

$$\pi_{firstname}(Student) - \pi_{president}(Club)$$

- 📍 What is this query doing?

Student		
<u>FirstName,</u>	GPA,	SID
[James,	3.9,	1701]
[Jean,	3.9,	1702]
[John,	3.0,	1703]
[Mary,	4.0,	1801]
[Mike,	4.0,	1805]

Club	
<u>President,</u>	Name
[James,	C1]
[Mary,	C2]
[Tom,	C3]

Outline

1. Projection π
2. Selection σ
3. Composability
4. Union \cup
5. Set Difference $(-)$
- 6. Cross Product (\times)**
7. Renaming(ρ)



Cross Product(\times)

- Takes two relations
 - Pair each tuple $t_1 \in R_1$ with each tuple $t_2 \in R_2$
- Property
 - Given $arity(R_1) = k_1, arity(R_2) = k_2$
 - $arity(R_1 \times R_2) = k_1 + k_2$
 - $t \in R_1 \times R_2$ iff :
 - The first k_1 components of t form a tuple in R_1
 - The next k_2 components of t form a tuple in R_2



Cross Product(\times)

- Example

- $Student \times MajorsIn$

Student		
FirstName,	GPA,	SID
[James,	3.9,	1701]
[Jean,	3.9,	1702]
[John,	3.0,	1703]
[Mary,	4.0,	1801]
[Mike,	4.0,	1805]

 \times

MajorsIn	
SID,	MID
[1701,	01]
[1701,	02]
[1805,	03]
[1801,	04]

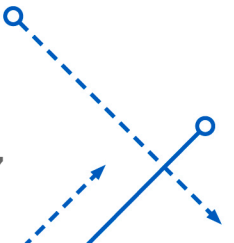
 $=$

FirstName,	GPA,	(SID),	(SID),	MID
[James,	3.9,	1701,	1701,	01]
[James,	3.9,	1701,	1701,	02]
[James,	3.9,	1701,	1805,	03]
[James,	3.9,	1701,	1801,	04]
[Jean,	3.9,	1702,	1701,	01]
[Jean,	3.9,	1702,	1701,	02]
[Jean,	3.9,	1702,	1805,	03]
[Jean,	3.9,	1702,	1801,	04]
[John,	3.0,	1703,	1701,	01]
[John,	3.0,	1703,	1701,	02]
[John,	3.0,	1703,	1805,	03]
[John,	3.0,	1703,	1801,	04]
[Mary,	4.0,	1801,	1701,	01]
[Mary,	4.0,	1801,	1701,	02]
[Mary,	4.0,	1801,	1805,	03]
[Mary,	4.0,	1801,	1801,	04]

... ..

Outline

1. Projection π
2. Selection σ
3. Composability
4. Union \cup
5. Set Difference $(-)$
6. Cross Product (\times)
7. **Renaming (ρ)**



Renaming (ρ)

- Gives a set of new attribute names as indicated in the list of B_1, \dots, B_n to R
- Property
 - Let A_1, \dots, A_n be the attributes of R before renaming
 - $arity(\rho_{B_1, \dots, B_n}(R)) = arity(R) = n$
 - $t \in \rho_{B_1, \dots, B_n}(R)$ iff for some $s \in R$
 - $t[B_1, \dots, B_n] = s[A_1, \dots, A_n]$
- Example
 - $\rho_{FirstName, GPA, Sid, Miid, Mid}(Student \times MajorsIn)$

FirstName, GPA, Sid, Miid, Mid

...

...



Set- and Bag- RA

Which operators behaviors differently in Set- and Bag- RA?

Select (σ)	No
Projection (π)	Yes
Cross-product (\times)	No
Set-difference ($-$)	No
Union (\cup)	Yes

Recommended Reading

Database Systems: The Complete Book
Chapter 2.4, 5.2