

CSE460/560 DATA MODELS AND QUERY LANGUAGES

Relational Algebra – Derived Operators

Cheng-En Chuang

(Slides Adopted from Jan Chomicki and Ning Deng)



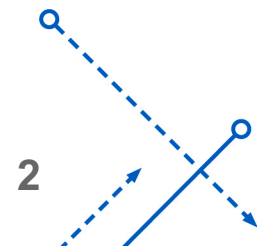
University at Buffalo

Department of Computer Science
and Engineering

School of Engineering and Applied Sciences

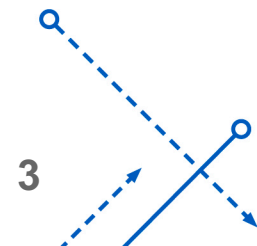
Outline

1. Intersection \cap
2. θ – Join $\bowtie_{A \theta B}$
3. Natural Join θ
4. Quotient $/$
5. Linear Notation



Outline

1. Intersection \cap
2. θ – Join $\bowtie_{A \theta B}$
3. Natural Join θ
4. Quotient $/$
5. Linear Notation



Intersection \cap

- Takes two compatible relations
 - Returns tuples in **both** relations
- Properties
 - $arity(R_1 \cap R_2) = arity(R_1) = arity(R_2)$
 - $t \in R_1 \cap R_2$ iff $t \in R_1 \wedge t \in R_2$
- Intersection $R_1 \cap R_2$ is a derived operation
 - How to define it in terms of essential operators?
- Example
 - $\rho_{FirstName}(\pi_{FirstName}(Student) \cap \pi_{president}(Club))$
 - What is this query asking for?

Student

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

Club

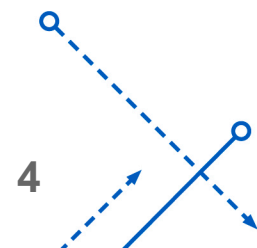
<u>President,</u>	<u>Name</u>
[James,	C1]
[Mary,	C2]
[Tom,	C3]

MajorsIn

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

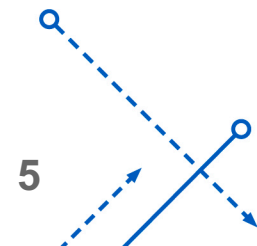
Major

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



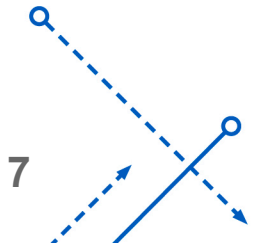
Outline

1. Intersection \cap
2. θ – Join $\bowtie_{A \theta B}$
3. Natural Join θ
4. Quotient $/$
5. Linear Notation



Outline

1. Intersection \cap
2. θ – Join $\bowtie_{A \theta B}$
3. **Natural Join** \bowtie
4. Quotient $/$
5. Linear Notation



Natural Join ⋈

- A useful join variant connects two relations
- A_1, \dots, A_n : all attributes of R_1
- B_1, \dots, B_k : all attributes of R_2
- m : the number of attributes common to R_1 and R_2
- $R_1 \bowtie R_2$
 - Select from $R_1 \times R_2$ the tuples that agree on all common attributes
 - Project duplicated columns out
- Properties
 - $arity(R_1 \bowtie R_2) = arity(R_1) + arity(R_2) - m$
- Example
 - $MajorsIn \bowtie Major$

Student

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

Club

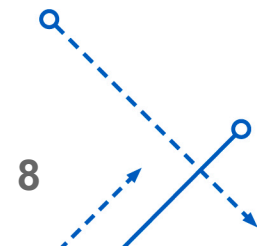
President,	Name
[James,	C1]
[Mary,	C2]
[Tom,	C3]

MajorsIn

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

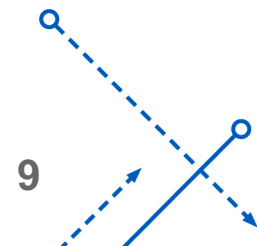
Major

MID,	Name
[01,	CS]
[02,	EE]
[03,	Math]
[04,	ME]



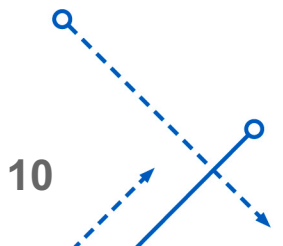
Outline

1. Intersection \cap
2. θ – Join $\bowtie_{A \theta B}$
3. Natural Join θ
4. **Quotient /**
5. Linear Notation



Quotient

- A_1, \dots, A_{n+k} : all attributes of R_1
- A_{n+1}, \dots, A_{n+k} : all attributes of R_2
- Quotient (Division) identifies all the attributes value in a relation
 - Found to be paired with all values from another relation
- Properties
 - $arity(R_1/R_2) = arity(R_1) - arity(R_2) = n$
 - $t \in R_1/R_2$ iff for all $s \in R_2$ there is a $w \in R_1$ such that
 - $w[A_1, \dots, A_n] = t[A_1, \dots, A_n]$ and
 - $w[A_{n+1}, \dots, A_{n+k}] = s[A_{n+1}, \dots, A_{n+k}]$



Quotient

- Express queries with “all”
 - Which are students taking all the provided courses?

<u>Name,</u>	<u>Course</u>
[John,	OS]
[James,	DB]
[James,	ML]
[James,	OS]
[James,	DS]
[Mary,	DB]
[Mary,	OS]

S

Course
[OS]

C1

Name
[John]
[James]
[Mary]

S/C1

Course
[OS]
[DB]

C2

Name
[James]
[Mary]

S/C2

Course
[OS]
[DB]
[ML]

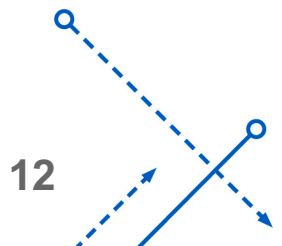
C3

Name
[James]

S/C3

Quotient

- And yes, Quotient can be expressed in terms of essential operators
- $R_1/R_2 = \pi_{A_1, \dots, A_n}(R_1) - \pi_{A_1, \dots, A_n}(\pi_{A_1, \dots, A_n}(R_1) \times R_2 - R_1)$
- All possible candidates – Unqualified Candidates
 - All possible candidates
 - $\pi_{A_1, \dots, A_n}(R_1)$
 - Unqualified Candidates
 - All possible qualifying sub-conditions – All qualifying sub-conditions
 - All possible qualifying sub-conditions
 - $\pi_{A_1, \dots, A_n}(R_1) \times R_2$
 - All qualifying sub-conditions
 - R_1



Outline

1. Intersection \cap
2. θ – Join $\bowtie_{A \theta B}$
3. Natural Join θ
4. Quotient $/$
5. **Linear Notation**



Linear Notation

- Invent new names for intermediate relations
 - Assign them values that are RA expressions
- Renaming of attributes in schema of new relation implicitly
- Example

$$R_1(FirstName) := \pi_{FirstName}(Student) \cap \pi_{president}(Club)$$

$$R_2(FirstName) := \pi_{FirstName}(\sigma_{GPA>3.5}(Student))$$

$$Ans(Name) := R_1 \cup R_2$$

- What is this query asking?
 - What if we change the final expression to

$$Ans(Name) := R_1 \cap R_2$$

Recommended Reading

Database Systems: The Complete Book
Chapter 2.4, 5.2