Relational Algebra 2

Lecture Handout

Dr Eugenia Ternovska

Simon Fraser University

In this part

Derived operations of Relational Algebra (RA):

- Intersection
- Natural Join
- Theta-Join
- Equijoin
- Outer Joins (full, left, right)
- Division

Derived operations

some common examples

$\cap$ can be expressed in terms of difference:

$$R \cap S = R - (R - S)$$

$\bowtie$ can be expressed in terms of π , σ , $\times$, ρ (next slide)

There are many more examples

Derived relational algebra operations are operations on relations that are expressible via an expression built from the basic operators.

Derived Operation: Natural Join

$R \bowtie S$ – pairing two tables on **common attributes**

• Pair only those tuples of R and S that agree in whatever attributes are common to the schemas of R and S

• A tuple r and a tuple s are matched if and only if r and s agree on value on each of attributes A1, A2, . . . , An that are **in both schemas**

• The resulting tuple is called joined tuple

Derived Operation: Natural Join

Example

R	A	B	$\bowtie$	S	B	C	$=$	$R \bowtie S$	A	B	C
	1	2			1	a			1	2	b
	4	2			2	b			4	2	b
	3	4			3	c					

Joined tuples: $(1, 2, b)$, $(4, 2, b)$, on attribute **B**, which we do not repeat

Dangling Tuple

- A tuple that fails to pair with any tuple of the other relation
- Has no effect on the result of a natural join

Dangling tuples: left:$(3, 4)$; right: $(1, a)$, $(3, c)$

Derived Operation: Natural Join

Express $\bowtie$ using basic operations of RA:

Example

Customer: **CustID**, Name, City, Address

Account: Number, Branch, **CustID**, Balance

Customer $\bowtie$ Account $=$

$$\pi_{X \cup Y}\left(\sigma_{\text{CustID}=\text{CustID}'}\left(\text{Customer} \times \rho_{\text{CustID} \to \text{CustID}'}(\text{Account})\right)\right)$$

where $X = \{$ all attributes of Customer $\}$
$Y = \{$ all attributes of Account $\}$

The schema of $\bowtie$ is the union of schemas, $X \cup Y$
Notice: no repeated attributes at all in the resulting schema

Other derived operations: Theta-join $\bowtie_\theta$

Example

R	A	B	C
	1	2	3
	6	7	8
	9	7	8

S	B	C	D
	2	3	4
	2	3	5
	7	8	10

$=$

$R \bowtie_{A<D} S$	A	R.B	R.C	S.B	S.C	D
	1	2	3	2	3	4
	1	2	3	2	3	5
	1	2	3	7	8	10
	6	7	8	7	8	10
	9	7	8	7	8	10

$R \bowtie_{A<D \text{ AND } R.B \neq S.B} S$	A	R.B	R.C	S.B	S.C	D
	1	2	3	7	8	10

Other derived operations: Theta-join $\bowtie_\theta$

- Take the product of R and S
- Select from the product only those tuples that satisfy the condition Θ
- Schema of $R \bowtie_\theta S$ is the union of the (modified) schemas of R and S, after renaming the repeated attributes, say B in R to R.B, and B in S to S.B, as in the previous slide

(This rule for schema is the same as for Product's schema)

More derived operations

Theta-join	$R \bowtie_\theta S = \sigma_\theta(R \times S)$
Equijoin	$\bowtie_\theta$ where θ is a **conjunction of equalities**
Semijoin	$R \ltimes_\theta S = \pi_X\big(R \bowtie_\theta S\big)$ where X is the set of attributes of R
Antijoin	$R \overline{\ltimes}_\theta S = R - (R \ltimes_\theta S)$

Why use derived operations?

- to write queries more succinctly
- they can be optimized independently, as separate operations

Joins

Inner Joins

No padding with NULLs

- Natural Join (match and do not repeat common attributes)
- Theta-Join (join on condition θ, rename and repeat common attributes)
- Equijoin (same, but θ is a conjunction of equalities)

Outer Joins

Include unmatched tuples from the participating relations and pad them with NULLs, rename and repeat common attributes

- Full outer join
- Left outer join
- Right outer join

(Semijoin and Antijoin are separate, not in these categories)

Outer Join: Example

Example

R	A	B
	1	2
	3	4

S	B	C	D
	2	3	4
	5	6	7

$R \bowtie^{\text{outer}}_{R.B=S.B} S$	A	R.B	S.B	C	D
	1	2	2	3	4
	3	4	NULL	NULL	NULL
	NULL	NULL	5	6	7

Recall that those tuples that are not joined are called dangling tuples

All dangling tuples are padded with NULLs

Outer Join: Summary

- The outer join (a.k.a.**full outer join**) $R \bowtie^{\text{outer}}_{\theta} S$
- Schema contains union of the (possibly renamed) attributes in R and S
- Tuples consist of:
  - The tuples like in the regular Natural Join, but with column repetition preserved (see example)
  - The tuples of R that do not join with any tuple in S, and padded with NULL on the right instead
  - The tuples of S that do not join with any tuple in R, padded with NULL on the left instead

Left Outer Join: Example

Example

R	A	B
	1	2
	3	4

S	B	C	D
	2	3	4
	5	6	7

$R \bowtie^{\text{left}}_{R.B=S.B} S$	A	R.B	S.B	C	D
	1	2	2	3	4
	3	4	NULL	NULL	NULL

Left dangling tuples are padded with NULLs on the right

Left Outer Join: Summary

- The left outer join $R \bowtie^{\text{left}}_{\theta} S$
- Schema contains union of the (possibly renamed) attributes in R and S
- Tuples consist of:
  - The tuples like in the regular Natural Join, but with column repetition preserved
  - The tuples of R that do not join with any tuple in S (left dangling tuples), padded with NULL on the right instead

Right Outer Join: Example

Example

R	A	B
	1	2
	3	4

S	B	C	D
	2	3	4
	5	6	7

$R \bowtie^{\text{right}}_{R.B=S.B} S$	A	R.B	S.B	C	D
	1	2	2	3	4
	NULL	NULL	5	6	7

Right dangling tuples are padded with NULLs on the left

Right Outer Join: Summary

- The right outer join $R \bowtie^{\text{right}}_{\theta} S$
- Schema contains union of the (possibly renamed) attributes in R and S
- Tuples consist of:
  - The tuples like in the regular Natural Join, but with column repetition preserved
  - The tuples of S that do not join with any tuple in R (right dangling tuples), padded with NULL on the left instead

Another Derived Operation: Division (Example)

Find the names of students who have taken exams in all CS courses

Exams

Student	Course
John	Databases
John	Chemistry
Mary	Programming
Mary	Math
Mary	Databases

CS

CourseName
Databases
Programming

Exams $\div$ CS $=$

Student
Mary

$$= \pi_{\textbf{Student}}(\text{Exams}) - \pi_{\textbf{Student}}\big((\pi_{\textbf{Student}}(\text{Exams}) \times \text{CS}) - \text{Exams}\big)$$

Division

R over set of attributes X

S over set of attributes $Y \subset X$

Let $Z = X - Y$

$$R \div S = \{\ \bar{r} \in \pi_Z(R) \mid \forall \bar{s} \in S\ (\bar{r}\bar{s} \in R\ )\}$$

$$= \{\ \bar{r} \in \pi_Z(R) \mid \{\bar{r}\} \times S \subseteq R\ \}$$

$$= \pi_Z(R) - \pi_Z\big((\pi_Z(R) \times S) - R\big)$$

Here, $\bar{r}$ and $\bar{s}$ denote tuples of data values.

Acknowledgements

[1] Database Systems: The Complete Book, 2nd EditionHector Garcia-Molina, Jeffrey D. Ullman, Jennifer WidomPrentice Hall, 2009

[2] Database System Concepts, Seventh EditionAvi Silberschatz, Henry F. Korth, S. SudarshanMcGraw-Hill, March 2019www.db-book.com

Additional references and resources used in preparation of this course are listed on the course webpage or mentioned in slides.