

# Chapter 3

## Relational Database Languages: Relational Algebra

We first consider only *query* languages.

**Relational Algebra:** Queries are expressions over operators and relation names.

**Relational Calculus:** Queries are special formulas of first-order logic with free variables.

**SQL:** Combination from algebra and calculus and additional constructs. Widely used DML for relational databases.

**QBE:** Graphical query language.

**Deductive Databases:** Queries are logical rules.

73

## RELATIONAL DATABASE LANGUAGES: COMPARISON AND OUTLOOK

### Remark:

- Relational Algebra and (safe) Relational Calculus have the same expressive power. For every expression of the algebra there is an equivalent expression in the calculus, and vice versa.
- A query language is called **relationally complete**, if it is (at least) as expressive as the relational algebra.
- These languages are compromises between efficiency and expressive power; they are not computationally complete (i.e., they cannot simulate a Turing Machine).
- They can be embedded into host languages (e.g. C++ or Java) or extended (PL/SQL), resulting in full computational completeness.
- Deductive Databases (Datalog) are more expressive than relational algebra and calculus.

74

## 3.1 Relational Algebra: Computations over Relations

### Operations on Tuples – Overview Slide

Let  $\mu \in \text{Tuple}(\bar{X})$  where  $\bar{X} = \{A_1, \dots, A_k\}$ .

(Formal definition of  $\mu$  see Slide 62)

- For  $\emptyset \subset \bar{Y} \subseteq \bar{X}$ , the expression  $\mu[\bar{Y}]$  denotes the **projection** of  $\mu$  to  $\bar{Y}$ .

Result:  $\mu[\bar{Y}] \in \text{Tuple}(\bar{Y})$  where  $\mu[\bar{Y}](A) = \mu(A), A \in \bar{Y}$ .

- A **selection condition**  $\alpha$  (wrt.  $\bar{X}$ ) is an expression of the form  $A \theta B$  or  $A \theta c$ , or  $c \theta A$  where  $A, B \in \bar{X}$ ,  $\text{dom}(A) = \text{dom}(B)$ ,  $c \in \text{dom}(A)$ , and  $\theta$  is a **comparison operator** on that domain like e.g.  $\{=, \neq, \leq, <, \geq, >\}$ .

A tuple  $\mu \in \text{Tuple}(\bar{X})$  **satisfies** a selection condition  $\alpha$ , if – according to  $\alpha$  –  $\mu(A) \theta \mu(B)$  or  $\mu(A) \theta c$ , or  $c \theta \mu(A)$  holds.

These (atomic) selection conditions can be combined to formulas by using  $\wedge, \vee, \neg$ , and  $(, )$ .

- For  $\bar{Y} = \{B_1, \dots, B_k\}$ , the expression  $\mu[A_1 \rightarrow B_1, \dots, A_k \rightarrow B_k]$  denotes the **renaming** of  $\mu$ .

Result:  $\mu[\dots, A_i \rightarrow B_i, \dots] \in \text{Tuple}(\bar{Y})$  where  $\mu[\dots, A_i \rightarrow B_i, \dots](B_i) = \mu(A_i)$  for  $1 \leq i \leq k$ .

75

Let  $\mu \in \text{Tuple}(\bar{X})$  where  $\bar{X} = \{A_1, \dots, A_k\}$ .

### Projection (Reduction to a subset of the attributes)

For  $\emptyset \subset \bar{Y} \subseteq \bar{X}$ , the expression  $\mu[\bar{Y}]$  denotes the **projection** of  $\mu$  to  $\bar{Y}$ .

Result:  $\mu[\bar{Y}] \in \text{Tuple}(\bar{Y})$  where  $\mu[\bar{Y}](A) = \mu(A), A \in \bar{Y}$ .

projection to a given set of attributes

#### Example 3.1

Consider the relation schema  $R(\bar{X}) = \text{Continent}(\text{name}, \text{area}): \bar{X} = [\text{name}, \text{area}]$

and the tuple  $\mu = \boxed{\text{name} \rightarrow \text{“Asia”}, \text{area} \rightarrow 4.50953\text{e}+07}$ .

formally:  $\mu(\text{name}) = \text{“Asia”}, \mu(\text{area}) = 4.5\text{E}7$

**projection attributes:** Let  $\bar{Y} = [\text{name}]$

Result:  $\mu[\text{name}] = \boxed{\text{name} \rightarrow \text{“Asia”}}$

□

76

Again,  $\mu \in \text{Tup}(\bar{X})$  where  $\bar{X} = \{A_1, \dots, A_k\}$ .

Selection (only those tuples that satisfy some condition)

A **selection condition**  $\alpha$  (wrt.  $\bar{X}$ ) is an expression of the form  $A \theta B$  or  $A \theta c$ , or  $c \theta A$  where  $A, B \in \bar{X}$ ,  $\text{dom}(A) = \text{dom}(B)$ ,  $c \in \text{dom}(A)$ , and  $\theta$  is a **comparison operator** on that domain like e.g.  $\{=, \neq, \leq, <, \geq, >\}$ .

A tuple  $\mu \in \text{Tup}(\bar{X})$  **satisfies** a selection condition  $\alpha$ , if – according to  $\alpha$  –  $\mu(A) \theta \mu(B)$  or  $\mu(A) \theta c$ , or  $c \theta \mu(A)$  holds.

yes/no-selection of tuples (without changing the tuple)

### Example 3.2

Consider again the relation schema  $R(\bar{X}) = \text{continent}(\text{name}, \text{area})$ :  $\bar{X} = [\text{name}, \text{area}]$ .

Selection condition:  $\text{area} > 10000000$ .

Consider again the tuple  $\mu = \boxed{\text{name} \rightarrow \text{“Asia”}, \text{area} \rightarrow 4.50953\text{e}+07}$ .

formally:  $\mu(\text{name}) = \text{“Asia”}$ ,  $\mu(\text{area}) = 4.5\text{E}7$

check:  $\mu(\text{area}) > 10000000$

Result: yes. □

These (atomic) selection conditions can be combined to formulas by using  $\wedge$ ,  $\vee$ ,  $\neg$ , and  $(, )$ .

77

Let  $\mu \in \text{Tup}(\bar{X})$  where  $\bar{X} = \{A_1, \dots, A_k\}$ .

Renaming (of attributes)

For  $\bar{Y} = \{B_1, \dots, B_k\}$ , the expression  $\mu[A_1 \rightarrow B_1, \dots, A_k \rightarrow B_k]$  denotes the **renaming** of  $\mu$ .

Result:  $\mu[\dots, A_i \rightarrow B_i, \dots] \in \text{Tup}(\bar{Y})$  where  $\mu[\dots, A_i \rightarrow B_i, \dots](B_i) = \mu(A_i)$  for  $1 \leq i \leq k$ .

renaming of attributes (without changing the tuple)

### Example 3.3

Consider (for a tuple of the table  $R(\bar{X}) = \text{encompasses}(\text{country}, \text{continent}, \text{percent})$ ):

$\bar{X} = [\text{country}, \text{continent}, \text{percent}]$ .

Consider the tuple  $\mu = \boxed{\text{country} \rightarrow \text{“R”}, \text{continent} \rightarrow \text{“Asia”}, \text{percent} \rightarrow 80}$ .

formally:  $\mu(\text{country}) = \text{“R”}$ ,  $\mu(\text{continent}) = \text{“Asia”}$ ,  $\mu(\text{percent}) = 80$

Renaming:  $\bar{Y} = [\text{code}, \text{name}, \text{percent}]$

Result: a new tuple  $\mu[\text{country} \rightarrow \text{code}, \text{continent} \rightarrow \text{name}, \text{percent} \rightarrow \text{percent}] =$

$\boxed{\text{code} \rightarrow \text{“R”}, \text{name} \rightarrow \text{“Asia”}, \text{percent} \rightarrow 80}$  that now fits into the schema  $\text{new\_encompasses}(\text{code}, \text{name}, \text{percent})$ . □

The usefulness of renaming will become clear later ...

78

## EXPRESSIONS IN THE RELATIONAL ALGEBRA

### What is an algebra?

- An algebra consists of a "domain" (i.e., a set of "things"), and a set of operators.
- Operators map elements of the domain to other elements of the domain.
- Each of the operators has a "semantics", that is, a definition how the result of applying it to some input should look like.
- **Algebra expressions** are built over basic constants and operators (inductive definition).

### Relational Algebra

- The "domain" consists of all relations (over arbitrary sets of attributes).
- The operators are then used for combining relations, and for describing computations - e.g., in SQL.
- **Relational algebra expressions** are defined inductively over relations and operators.
- Relational algebra expressions define queries against a relational database.

79

## INDUCTIVE DEFINITION OF EXPRESSIONS

### Atomic Expressions

- For an arbitrary attribute  $A$  and a constant  $a \in \text{dom}(A)$ , the **constant relation**  $A : \{a\}$  is an algebra expression.

Format:  $[A]$

Result relation:  $\{a\}$

<b>A:{a}</b>
<b>A</b>
<b>a</b>

- Given a database schema  $\mathbf{R} = \{R_1(\bar{X}_1), \dots, R_n(\bar{X}_n)\}$ , every relation name  $R_i$  is an algebra expression.

Format of  $R_i$ :  $\bar{X}_i$

Result relation (wrt. a given database state  $\mathcal{S}$ ): the relation  $\mathcal{S}(R_i)$  that is currently stored in the database.

80

## Structural Induction: Applying an Operator

- takes one or more input relations  $in_1, in_2, \dots$
- produces a result relation  $out$ :
  - $out$  has a **format**, depends on the formats of the input relations.
  - $out$  is a relation, i.e., it contains some tuples, depends on the content of the input relations.

81

## BASE OPERATORS

Let  $\bar{X}, \bar{Y}$  formats and  $r \in \text{Rel}(\bar{X})$  and  $s \in \text{Rel}(\bar{Y})$  relations over  $\bar{X}$  and  $\bar{Y}$ .

### Union

Assume  $r, s \in \text{Rel}(\bar{X})$ .

Result format of  $r \cup s$ :  $\bar{X}$

Result relation:  $r \cup s = \{\mu \in \text{Tup}(\bar{X}) \mid \mu \in r \text{ or } \mu \in s\}$ .

$$r = \begin{array}{ccc} \hline A & B & C \\ a & b & c \\ d & a & f \\ c & b & d \end{array}$$
$$s = \begin{array}{ccc} \hline A & B & C \\ b & g & a \\ d & a & f \end{array}$$
$$r \cup s = \begin{array}{ccc} \hline A & B & C \\ a & b & c \\ d & a & f \\ c & b & d \\ b & g & a \end{array}$$

82

## Set Difference

Assume  $r, s \in \text{Rel}(\bar{X})$ .

Result format of  $r \setminus s$ :  $\bar{X}$

Result relation:  $r \setminus s = \{\mu \in r \mid \mu \notin s\}$ .

	<u>A</u>	<u>B</u>	<u>C</u>
$r =$	a	b	c
	d	a	f
	c	b	d

	<u>A</u>	<u>B</u>	<u>C</u>
$s =$	b	g	a
	d	a	f

	<u>A</u>	<u>B</u>	<u>C</u>
$r \setminus s =$	a	b	c
	c	b	d

83

## Projection (Reduction to a subset of the attributes)

Assume  $r \in \text{Rel}(\bar{X})$  and  $\bar{Y} \subseteq \bar{X}$ .

Result format of  $\pi[\bar{Y}](r)$ :  $\bar{Y}$

Result relation:  $\pi[\bar{Y}](r) = \{\mu[\bar{Y}] \mid \mu \in r\}$ .

### Example 3.4

<b>Continent</b>	
<u>name</u>	<u>area</u>
Europe	9562489.6
Africa	3.02547e+07
Asia	4.50953e+07
America	3.9872e+07
Australia	8503474.56

Let  $\bar{Y} = [\text{name}]$

$\mu_1[\text{name}] = \text{name} \rightarrow \text{"Europe"}$

$\mu_2[\text{name}] = \text{name} \rightarrow \text{"Africa"}$

$\mu_3[\text{name}] = \text{name} \rightarrow \text{"Asia"}$

$\mu_4[\text{name}] = \text{name} \rightarrow \text{"America"}$

$\mu_5[\text{name}] = \text{name} \rightarrow \text{"Australia"}$

$\pi[\text{name}](\text{Continent})$
<u>name</u>
Europe
Africa
Asia
America
Australia

□

84

### Selection (Reduction of number of tuples by a condition)

Assume  $r \in \text{Rel}(\bar{X})$  and a selection condition  $\alpha$  over  $\bar{X}$ .

Result format of  $\sigma[\alpha](r)$ :  $\bar{X}$

Result relation:  $\sigma[\alpha](r) = \{\mu \in r \mid \mu \text{ satisfies } \alpha\}$ .

#### Example 3.5

<b>Continent</b>	
<u>name</u>	<u>area</u>
Europe	9562489.6
Africa	3.02547e+07
Asia	4.50953e+07
America	3.9872e+07
Australia	8503474.56

Let  $\alpha = \text{"area} > 10000000\text{"}$

$\mu_1(\text{area}) > 10000000$ ?– no

$\mu_2(\text{area}) > 10000000$ ?– yes

$\mu_3(\text{area}) > 10000000$ ?– yes

$\mu_4(\text{area}) > 10000000$ ?– yes

$\mu_5(\text{area}) > 10000000$ ?– no

$\sigma[\text{area} > 10E6](\text{Continent})$	
<u>name</u>	<u>area</u>
Africa	3.02547e+07
Asia	4.50953e+07
America	3.9872e+07

□

### Renaming (of attributes)

Assume  $r \in \text{Rel}(\bar{X})$  with  $\bar{X} = [A_1, \dots, A_k]$  and a renaming  $[A_1 \rightarrow B_1, \dots, A_k \rightarrow B_k]$ .

Result format of  $\rho[A_1 \rightarrow B_1, \dots, A_k \rightarrow B_k](r)$ :  $[B_1, \dots, B_k]$

Result relation:  $\rho[A_1 \rightarrow B_1, \dots, A_k \rightarrow B_k](r) = \{\mu[A_1 \rightarrow B_1, \dots, A_k \rightarrow B_k] \mid \mu \in r\}$ .

#### Example 3.6

Consider the renaming of the table *encompasses*(country, continent, percent):

$\bar{X} = [\text{country}, \text{continent}, \text{percent}]$

Renaming:  $\bar{Y} = [\text{code}, \text{name}, \text{percent}]$

$\rho[\text{country} \rightarrow \text{code}, \text{continent} \rightarrow \text{name}, \text{percent} \rightarrow \text{percent}](\text{encompasses})$		
<u>code</u>	<u>name</u>	<u>percent</u>
R	Europe	20
R	Asia	80
D	Europe	100
⋮	⋮	⋮

□

**(Natural) Join (Combining two relations via common attributes)**

Assume  $r \in \text{Rel}(\bar{X})$  and  $s \in \text{Rel}(\bar{Y})$  for arbitrary  $\bar{X}, \bar{Y}$ .

Convention: For  $\bar{X} \cup \bar{Y}$ , as a shorthand, write  $\overline{XY}$ .

for two tuples  $\mu_1 = [v_1, \dots, v_n]$  and  $\mu_2 = [w_1, \dots, w_m]$ ,  $\mu_1\mu_2 := [v_1, \dots, v_n, w_1, \dots, w_m]$ .

Result format of  $r \bowtie s$ :  $\overline{XY}$ .

Result relation:  $r \bowtie s = \{\mu \in \text{Dup}(\overline{XY}) \mid \mu[\bar{X}] \in r \text{ and } \mu[\bar{Y}] \in s\}$ .

**Motivation**

Simplest Case:  $\bar{X} \cap \bar{Y} = \emptyset \Rightarrow$  Cartesian Product  $r \bowtie s = r \times s$

$r \times s = \{\mu_1\mu_2 \in \text{Dup}(\overline{XY}) \mid \mu_1 \in r \text{ and } \mu_2 \in s\}$ .

$r =$	<table border="1" style="border-collapse: collapse;"> <tr><th>A</th><th>B</th></tr> <tr><td>1</td><td>2</td></tr> <tr><td>4</td><td>5</td></tr> </table>	A	B	1	2	4	5	$s =$	<table border="1" style="border-collapse: collapse;"> <tr><th>C</th><th>D</th></tr> <tr><td>a</td><td>b</td></tr> <tr><td>c</td><td>d</td></tr> <tr><td>e</td><td>f</td></tr> </table>	C	D	a	b	c	d	e	f	$r \bowtie s =$	<table border="1" style="border-collapse: collapse;"> <tr><th>A</th><th>B</th><th>C</th><th>D</th></tr> <tr><td>1</td><td>2</td><td>a</td><td>b</td></tr> <tr><td>1</td><td>2</td><td>c</td><td>d</td></tr> <tr><td>1</td><td>2</td><td>e</td><td>f</td></tr> <tr><td>4</td><td>5</td><td>a</td><td>b</td></tr> <tr><td>4</td><td>5</td><td>c</td><td>d</td></tr> <tr><td>4</td><td>5</td><td>e</td><td>f</td></tr> </table>	A	B	C	D	1	2	a	b	1	2	c	d	1	2	e	f	4	5	a	b	4	5	c	d	4	5	e	f
A	B																																														
1	2																																														
4	5																																														
C	D																																														
a	b																																														
c	d																																														
e	f																																														
A	B	C	D																																												
1	2	a	b																																												
1	2	c	d																																												
1	2	e	f																																												
4	5	a	b																																												
4	5	c	d																																												
4	5	e	f																																												

**Example 3.7 (Cartesian Product of Continent and Encompasses)**

The cartesian product combines everything with everything, not only “meaningful” combinations:

<b>Continent × encompasses</b>				
name	area	continent	country	percent
Europe	9562489.6	Europe	D	100
Europe	9562489.6	Europe	R	20
Europe	9562489.6	Asia	R	80
Europe	9562489.6	:	:	:
Africa	3.02547e+07	Europe	D	100
Africa	3.02547e+07	Europe	R	20
Africa	3.02547e+07	Asia	R	80
Africa	3.02547e+07	:	:	:
Asia	4.50953e+07	Europe	D	100
Asia	4.50953e+07	Europe	R	20
Asia	4.50953e+07	Asia	R	80
Asia	4.50953e+07	:	:	:
:	:	:	:	:



## Back to the Natural Join

General case  $\bar{X} \cap \bar{Y} \neq \emptyset$ : shared attribute names constrain the result relation.

Again the definition:  $r \bowtie s = \{\mu \in \text{Dup}(\overline{XY}) \mid \mu[\bar{X}] \in r \text{ and } \mu[\bar{Y}] \in s\}$ .

(Note: this implies that the tuples  $\mu_1 := \mu[\bar{X}] \in r$  and  $\mu_2 := \mu[\bar{Y}] \in s$  coincide in the shared attributes  $\bar{X} \cap \bar{Y}$ )

### Example 3.8

Consider *encompasses*(country,continent,percent) and *isMember*(organization,country,type):

<i>encompasses</i>			<i>isMember</i>		
country	continent	percent	organization	country	type
R	Europe	20	EU	D	member
R	Asia	80	UN	D	member
D	Europe	100	UN	R	member
:	:	:	:	:	:

$$\begin{aligned} \text{encompasses} \bowtie \text{isMember} = \{ \mu \in \text{Dup}(\text{country, cont, perc, org, type}) \mid \\ \mu[\text{country, cont, perc}] \in \text{encompasses} \text{ and } \mu[\text{org, country, type}] \in \text{isMember} \} \end{aligned}$$

□

### Example 3.8 (Continued)

$$\begin{aligned} \text{encompasses} \bowtie \text{isMember} = \{ \mu \in \text{Dup}(\text{country, cont, perc, org, type}) \mid \\ \mu[\text{country, cont, perc}] \in \text{encompasses} \text{ and } \mu[\text{org, country, type}] \in \text{isMember} \} \end{aligned}$$

start with  $(R, \text{Europe}, 20) \in \text{encompasses}$ .

check which tuples in *isMember* match:

$(UN, R, \text{member}) \in \text{isMember}$  matches:

result:  $(R, \text{Europe}, 20, UN, \text{member})$  belongs to the result.

(some more matches ...)

continue with  $(R, \text{Asia}, 80) \in \text{encompasses}$ .

$(UN, R, \text{member}) \in \text{isMember}$  matches:

result:  $(R, \text{Asia}, 80, UN, \text{member})$  belongs to the result.

(some more matches ...)

continue with  $(D, \text{Europe}, 100) \in \text{encompasses}$ .

$(EU, D, \text{member}) \in \text{isMember}$  matches:

result:  $(D, \text{Europe}, 100, EU, \text{member})$  belongs to the result.

$(UN, D, \text{member}) \in \text{isMember}$  matches:

result:  $(D, \text{Europe}, 100, UN, \text{member})$  belongs to the result.

(some more matches ...)

□

### Example 3.8 (Continued)

Result:

<i>encompasses</i> ⋈ <i>isMember</i>				
<i>country</i>	<i>continent</i>	<i>percent</i>	<i>organization</i>	<i>type</i>
<i>R</i>	<i>Europe</i>	<i>20</i>	<i>UN</i>	<i>member</i>
<i>R</i>	<i>Europe</i>	<i>20</i>	:	:
<i>R</i>	<i>Asia</i>	<i>80</i>	<i>UN</i>	<i>member</i>
<i>R</i>	<i>Asia</i>	<i>80</i>	:	:
<i>D</i>	<i>Europe</i>	<i>100</i>	<i>UN</i>	<i>member</i>
<i>D</i>	<i>Europe</i>	<i>100</i>	<i>EU</i>	<i>member</i>
<i>D</i>	<i>Europe</i>	<i>100</i>	:	:
:	:	:	:	:

□

### Example 3.9 (and Exercise)

Consider the expression

$Continent \bowtie \rho[country \rightarrow code, continent \rightarrow name, percent \rightarrow percent](encompasses)$

□

#### Functionalities of the Join

- Combining relations
- Selective functionality: only matching tuples survive  
(consider joining cities and organizations on headquarters)

## DERIVED OPERATORS

#### Intersection

Assume  $r, s \in \text{Rel}(\bar{X})$ .

Then,  $r \cap s = \{\mu \in \text{Tup}(\bar{X}) \mid \mu \in r \text{ and } \mu \in s\}$ .

#### Theorem 3.1

Intersection can be expressed by difference:  $r \cap s = r \setminus (r \setminus s)$ .

□

## $\theta$ -Join

Combination of **Cartesian Product** and **Selection**:

Assume  $r \in \text{Rel}(\bar{X})$ , and  $s \in \text{Rel}(\bar{Y})$ , such that  $\bar{X} \cap \bar{Y} = \emptyset$ , and  $A\theta B$  a selection condition.

$$r \bowtie_{A\theta B} s = \{\mu \in \text{Dup}(\overline{XY}) \mid \mu[\bar{X}] \in r, \mu[\bar{Y}] \in s \text{ and } \mu \text{ satisfies } A\theta B\} = \sigma[A\theta B](r \times s).$$

## Equi-Join

$\theta$ -join that uses the “=”-predicate.

### Example 3.10 (and Exercise)

Consider again Example 3.7:

$\text{Continent} \bowtie \text{encompasses} = \text{Continent} \times \text{encompasses}$  contained tuples that did not really make sense.

$(\text{Continent} \bowtie \text{encompasses})_{\text{continent=name}}$  would be more useful.

Furthermore, consider

$\pi[\text{continent, area, code, percent}]((\text{Continent} \bowtie \text{encompasses})_{\text{continent=name}})$ :

- removes the - now redundant - “name” column,
- is equivalent to the natural join  $(\rho[\text{name} \rightarrow \text{continent}](\text{continent})) \bowtie \text{encompasses}$ .  $\square$

## Semi-Join

- recall: joins combine, but are also selective
- semi-join acts like a selection on a relation  $r$ :  
selection condition not given as a boolean formula on the attributes of  $r$ , but by “looking into” another relation (a subquery)

Assume  $r \in \text{Rel}(\bar{X})$  and  $s \in \text{Rel}(\bar{Y})$  such that  $\bar{X} \cap \bar{Y} \neq \emptyset$ .

Result format of  $r \ltimes s$ :  $\bar{X}$

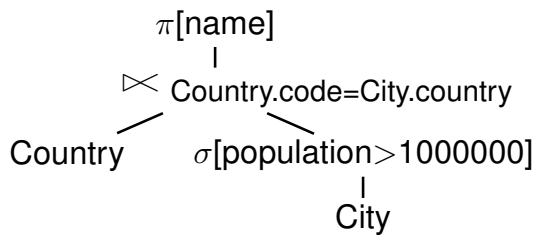
Result relation:  $r \ltimes s = \pi[\bar{X}](r \bowtie s)$

The semi-join  $r \ltimes s$  does *not* return the join, but checks which tuples of  $r$  “survive” the join with  $s$  (i.e., “which find a counterpart in  $s$  wrt. the shared attributes”):

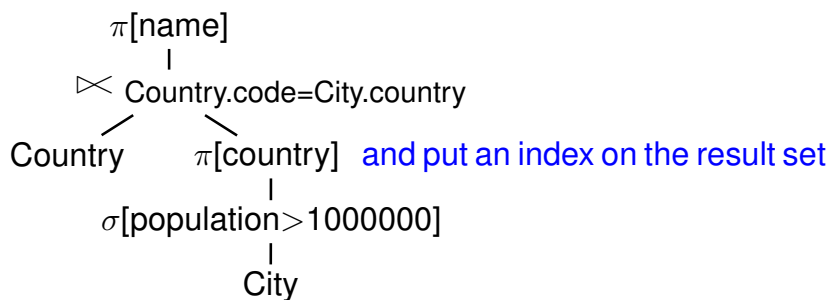
- Used with subqueries: (main query)  $\ltimes$  (subquery)
- $r \ltimes s \subseteq r$
- Used for optimizing the evaluation of joins (often in combination with indexes).

## Semi-Join: Example

Give the names of all countries where a city with at least 1000000 inhabitants is located:



- Have a short look “inside” the subquery, but don’t actually use it:
- look only if there is a big city in this country.
- “if the country code is in the set of country codes ...”:



95

## Outer Join

- The join is the operator for combining relations

### Example 3.11

- *Persons work in divisions of a company, tools are assigned to the divisions:*

<b>Works</b>	
<b>Person</b>	<b>Division</b>
John	Production
Bill	Production
John	Research
Mary	Research
Sue	Sales

<b>Tools</b>	
<b>Division</b>	<b>Tool</b>
Production	hammer
Research	pen
Research	computer
Admin.	typewriter

<b>Works <math>\bowtie</math> Tools</b>		
<b>Person</b>	<b>Division</b>	<b>Tool</b>
John	Production	hammer
Bill	Production	hammer
John	Research	pen
John	Research	computer
Mary	Research	pen
Mary	Research	computer

- *join contains no tuple that describes Sue,*
- *join contains no tuple that describes the administration or sales division,*
- *join contains no tuple that shows that there is a typewriter.*

□

96

## Outer Join

Assume  $r \in \text{Rel}(\bar{X})$  and  $s \in \text{Rel}(\bar{Y})$ .

Result format of  $r \bowtie s$ :  $\overline{XY}$

The outer join extends the “inner” join with all tuples that have no counterpart in the other relation (filled with null values):

### Example 3.12 (Outer Join)

Consider again Example 3.11

Works $\bowtie$ Tools		
Person	Division	Tool
John	Production	hammer
Bill	Production	hammer
John	Research	pen
John	Research	computer
Mary	Research	pen
Mary	Research	computer
Sue	Sales	NULL
NULL	Admin	typewriter

Works $\times$ Tools	
Person	Division
John	Production
Bill	Production
John	Research
Mary	Research

Works $\times$ Tools	
Division	Tool
Production	hammer
Research	pen
Research	computer

97

Formally, the result relation  $r \bowtie s$  is defined as follows:

$J = r \times s$  — take the (“inner”) join as base

$r_0 = r \setminus \pi[\bar{X}](J) = r \setminus (r \times s)$  —  $r$ -tuples that “are missing”

$s_0 = s \setminus \pi[\bar{Y}](J) = s \setminus (r \times s)$  —  $s$ -tuples that “are missing”

$\bar{Y}_0 = \bar{Y} \setminus \bar{X}$ ,  $\bar{X}_0 = \bar{X} \setminus \bar{Y}$

Let  $\mu_s \in \text{Dup}(\bar{Y}_0)$ ,  $\mu_r \in \text{Dup}(\bar{X}_0)$  such that  $\mu_1, \mu_2$  consist only of *null* values

$$r \bowtie s = J \cup (r_0 \times \{\mu_s\}) \cup (s_0 \times \{\mu_r\}).$$

### Example 3.12 (Continued)

For the above example,

$J = \text{Works} \times \text{Tools}$

$r_0 = [\text{“Sue”, “Sales”}]$ ,  $s_0 = [\text{“Admin”, “Typewriter”}]$

$\bar{Y}_0 = \text{Tool}$ ,  $\bar{X}_0 = \text{Person}$

$$\mu_1 = \begin{array}{|c|} \hline \text{Tool} \\ \hline \text{null} \\ \hline \end{array} \quad \mu_2 = \begin{array}{|c|} \hline \text{Person} \\ \hline \text{null} \\ \hline \end{array}$$

$$r_0 \times \{\mu_1\} = \begin{array}{|c|c|c|} \hline \text{Person} & \text{Division} & \text{Tool} \\ \hline \text{Sue} & \text{Sales} & \text{null} \\ \hline \end{array}$$

$$s_0 \times \{\mu_2\} = \begin{array}{|c|c|c|} \hline \text{Person} & \text{Division} & \text{Tool} \\ \hline \text{null} & \text{Admin} & \text{Typewriter} \\ \hline \end{array} \quad \square$$

98

## Left and Right Outer Join

Analogously to the (full) outer join:

- $r \bowtie s = J \cup (r_0 \times \{\mu_s\})$ .
- $r \bowtie s = J \cup (s_0 \times \{\mu_r\})$ .

## Generalized Natural Join

Assume  $r_i \subseteq \text{ Tup}(\bar{X}_i)$ .

Result format:  $\cup_{i=1}^n \bar{X}_i$

Result relation:  $\bowtie_{i=1}^n r_i = \{\mu \in \text{ Tup}(\cup_{i=1}^n \bar{X}_i) \mid \mu[\bar{X}_i] \in r_i\}$

### Exercise 3.1

Prove that the natural join is associative (which makes the generalized natural join well-defined):

$$\begin{aligned} \bowtie_{i=1}^n r_i &= ((\dots((r_1 \bowtie r_2) \bowtie r_3) \bowtie \dots) \bowtie r_n) \\ &= (r_1 \bowtie (r_2 \dots (r_{n-1} \bowtie r_n) \dots)) \end{aligned}$$

□

## Relational Division

Assume  $r \in \text{ Rel}(\bar{X})$  and  $s \in \text{ Rel}(\bar{Y})$  such that  $\bar{Y} \subsetneq \bar{X}$ .

Result format of  $r \div s$ :  $\bar{Z} = \bar{X} \setminus \bar{Y}$ .

The result relation  $r \div s$  is specified as “all  $\bar{Z}$ -values that occur in  $\pi[\bar{Z}](r)$ , with the additional condition that they occur in  $r$  together with each of the  $\bar{Y}$  values that occur in  $s$ ”.

Formally,

$$r \div s = \{\mu \in \text{ Tup}(\bar{Z}) \mid \{\mu\} \times s \subseteq r\} = \pi[\bar{Z}](r) \setminus \pi[\bar{Z}]((\pi[\bar{Z}](r) \times s) \setminus r).$$

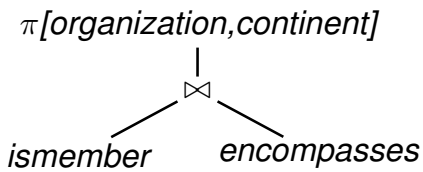
this implies that  $\mu \in \pi[\bar{Z}](r)$

- Simple observation:  $\pi[\bar{Z}](r) \supseteq r \div s$ .  
This constrains the set of possible results.
- Often,  $\bar{Z}$  and  $\bar{Y}$  correspond to the keys of relations that represent the instances of entity types.

### Example 3.13 (Relational Division)

Compute those organizations that have at least one member on each continent:

First step: which organizations have (some) member on which continents:

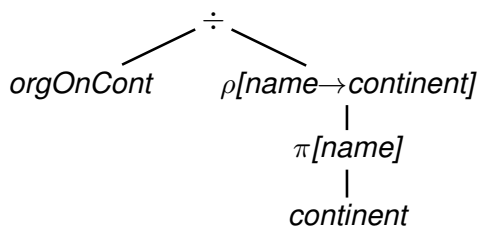


```
SELECT DISTINCT i.organization, e.continent
FROM ismember i, encompasses e
WHERE i.country=e.country
ORDER by 1
```

<b>orgOnCont</b>	
<b>organization</b>	<b>continent</b>
UN	Europe
UN	Asia
UN	America
UN	Africa
UN	Australia
NATO	Europe
NATO	America
NATO	Asia
:	:

101

### Example 3.13 (Cont'd)



<b>orgOnCont</b>	
<b>organization</b>	<b>continent</b>
UN	Europe
UN	Asia
UN	America
UN	Africa
UN	Australia
NATO	Europe
NATO	America
NATO	Asia
:	:

$\rho[\text{name} \rightarrow \text{continent}]$ $(\pi[\text{name}](\text{continent}))$
<b>continent</b>
Asia
Europe
Australia
America
Africa

$$r(\bar{X}), s(\bar{Y}), \bar{Z} := \bar{X} \setminus \bar{Y}$$

$$r \div s = \{\mu \in \text{Dup}(\bar{Z}) \mid \{\mu\} \times s \subseteq r\}$$

$$\bar{X} = [\text{organization, continent}]$$

$$\bar{Y} = [\text{continent}]$$

$$\text{Thus, } \bar{Z} = [\text{organization}].$$

- UN: occurs with each continent in orgOnCont  
⇒ belongs to the result.
- NATO: does not occur with each continent in orgOnCont  
⇒ does not belong to the result.

102

### Example 3.13 (Cont'd)

Consider again the formal algebraic characterization of Division:

$$r \div s = \{\mu \in \text{Dup}(\bar{Z}) \mid \{\mu\} \times s \subseteq r\} = \pi[\bar{Z}](r) \setminus \pi[\bar{Z}]((\pi[\bar{Z}](r) \times s) \setminus r).$$

1.  $r = \text{orgOnCont}$ ,  $s = \pi[\text{name}](\text{continent})$ ,  $Z = \text{Country}$ .
2.  $(\pi[\bar{Z}](r) \times s)$  contains all tuples of organizations with each of the continents, e.g.,  $(\text{NATO}, \text{Europe})$ ,  $(\text{NATO}, \text{Asia})$ ,  $(\text{NATO}, \text{America})$ ,  $(\text{NATO}, \text{Africa})$ ,  $(\text{NATO}, \text{Australia})$ .
3.  $((\pi[\bar{Z}](r) \times s) \setminus r)$  contains all such tuples which are not "valid", e.g.,  $(\text{NATO}, \text{Africa})$ .
4. projecting this to the organizations yields all those organizations where a non-valid tuple has been generated in (2), i.e., that have no member on some continent (e.g., NATO).
5.  $\pi[\bar{Z}](r)$  is the list of all organizations ...
6. ... subtracting those computed in (4) yields those that have a member on each continent. □

103

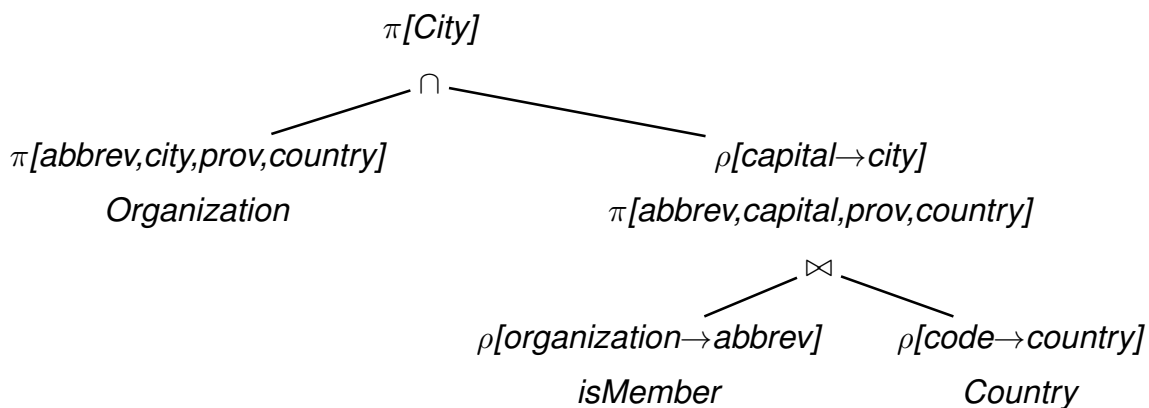
## EXPRESSIONS

- inductively defined: combining expressions by operators

### Example 3.14

The names of all cities where (i) headquarters of an organization are located, and (ii) that are capitals of a member country of this organization.

As a tree:



Note that there are many equivalent expressions. □

104



## EXPRESSIONS IN THE RELATIONAL ALGEBRA AS QUERIES

Let  $\mathbf{R} = \{R_1, \dots, R_k\}$  a set of relation schemata of the form  $R_i(\bar{X}_i)$ . As already described, an **database state** to  $\mathbf{R}$  is a **structure**  $\mathcal{S}$  that maps every relation name  $R_i$  in  $\mathbf{R}$  to a relation  $\mathcal{S}(R_i) \subseteq \text{Tuple}(\bar{X}_i)$

Every algebra expression  $Q$  defines a **query** against the state  $\mathcal{S}$  of the database:

- For given  $\mathbf{R}$ ,  $Q$  is assigned a **format**  $\Sigma_Q$  (the format of the answer).
- For every database state  $\mathcal{S}$ ,  $\mathcal{S}(Q) \subseteq \text{Tuple}(\Sigma_Q)$  is a relation over  $\Sigma_Q$ , called the **answer set** for  $Q$  wrt.  $\mathcal{S}$ .
- $\mathcal{S}(Q)$  can be computed according to the inductive definition, starting with the innermost (atomic) subexpressions.
- Thus, the relational algebra has a **functional semantics**.

105

## SUMMARY: INDUCTIVE DEFINITION OF EXPRESSIONS

### Atomic Expressions

- For an arbitrary attribute  $A$  and a constant  $a \in \text{dom}(A)$ , the **constant relation**  $A : \{a\}$  is an algebra expression.  
 $\Sigma_{A:\{a\}} = [A]$  and  $\mathcal{S}(A : \{a\}) = A : \{a\}$
- Every relation name  $R$  is an algebra expression.  
 $\Sigma_R = \bar{X}$  and  $\mathcal{S}(R) = \mathcal{S}(R)$ .

106

## SUMMARY (CONT'D)

### Compound Expressions

Assume algebra expressions  $Q_1, Q_2$  that define  $\Sigma_{Q_1}, \Sigma_{Q_2}, \mathcal{S}(Q_1)$ , and  $\mathcal{S}(Q_2)$ .

Compound algebraic expressions are now formed by the following rules (corresponding to the algebra operators):

### Union

If  $\Sigma_{Q_1} = \Sigma_{Q_2}$ , then  $Q = (Q_1 \cup Q_2)$  is the **union** of  $Q_1$  and  $Q_2$ .

$\Sigma_Q = \Sigma_{Q_1}$  and  $\mathcal{S}(Q) = \mathcal{S}(Q_1) \cup \mathcal{S}(Q_2)$ .

### Difference

If  $\Sigma_{Q_1} = \Sigma_{Q_2}$ , then  $Q = (Q_1 \setminus Q_2)$  is the **difference** of  $Q_1$  and  $Q_2$ .

$\Sigma_Q = \Sigma_{Q_1}$  and  $\mathcal{S}(Q) = \mathcal{S}(Q_1) \setminus \mathcal{S}(Q_2)$ .

### Projection

For  $\emptyset \neq \bar{Y} \subseteq \Sigma_{Q_1}$ ,  $Q = \pi[\bar{Y}](Q_1)$  is the **projection** of  $Q_1$  to the attributes in  $\bar{Y}$ .

$\Sigma_Q = \bar{Y}$  and  $\mathcal{S}(Q) = \pi[\bar{Y}](\mathcal{S}(Q_1))$ .

107

## INDUCTIVE DEFINITION OF EXPRESSIONS (CONT'D)

### Selection

For a selection condition  $\alpha$  over  $\Sigma_{Q_1}$ ,  $Q = \sigma[\alpha]Q_1$  is the **selection** from  $Q_1$  wrt.  $\alpha$ .

$\Sigma_Q = \Sigma_{Q_1}$  and  $\mathcal{S}(Q) = \sigma[\alpha](\mathcal{S}(Q_1))$ .

### Natural Join

$Q = (Q_1 \bowtie Q_2)$  is the **(natural) join** of  $Q_1$  and  $Q_2$ .

$\Sigma_Q = \Sigma_{Q_1} \cup \Sigma_{Q_2}$  and  $\mathcal{S}(Q) = \mathcal{S}(Q_1) \bowtie \mathcal{S}(Q_2)$ .

### Renaming

For  $\Sigma_{Q_1} = \{A_1, \dots, A_k\}$  and  $\{B_1, \dots, B_k\}$  a set of attributes,  $\rho[A_1 \rightarrow B_1, \dots, A_k \rightarrow B_k]Q_1$  is the **renaming** of  $Q_1$

$\Sigma_Q = \{B_1, \dots, B_k\}$  and  $\mathcal{S}(Q) = \{\mu[A_1 \rightarrow B_1, \dots, A_k \rightarrow B_k] \mid \mu \in \mathcal{S}(Q_1)\}$ .

108

## Example

### Example 3.15

*Professor*(PNr, Name, Office), *Course*(CNr, Credits, CName)

*teach*(PNr, CNr), *examine*(PNr, CNr)

- For each professor (name) determine the courses he gives (CName).

$$\pi [\text{Name, CName}] ((\text{Professor} \bowtie \text{teach}) \bowtie \text{Course})$$

- For each professor (name) determine the courses (CName) that he teaches, but that he does not examine.

$$\begin{aligned} & \pi [\text{Name, CName}] (( \\ & (\pi [\text{Name, CNr}] (\text{Professor} \bowtie \text{teach})) \\ & \setminus \\ & (\pi [\text{Name, CNr}] (\text{Professor} \bowtie \text{examine})) \\ & ) \bowtie \text{Course}) \end{aligned}$$

*Simpler expression:*

$$\pi [\text{Name, CName}] ((\text{Professor} \bowtie (\text{teach} \setminus \text{examine})) \bowtie \text{Course}) \quad \square$$

## EQUIVALENCE OF EXPRESSIONS

Algebra expressions  $Q, Q'$  are called **equivalent**,  $Q \equiv Q'$ , if and only if for all structures  $\mathcal{S}$ ,  $\mathcal{S}(Q) = \mathcal{S}(Q')$ .

Equivalence of expressions is the basis for **algebraic optimization**.

Let  $\text{attr}(\alpha)$  the set of attributes that occur in a selection condition  $\alpha$ , and  $Q, Q_1, Q_2, \dots$  expressions with formats  $X, X_1, \dots$

### Projections

- $\bar{Z}, \bar{Y} \subseteq \bar{X} \Rightarrow \pi[\bar{Z}](\pi[\bar{Y}](Q)) \equiv \pi[\bar{Z} \cap \bar{Y}](Q)$ .
- $\bar{Z} \subseteq \bar{Y} \subseteq \bar{X} \Rightarrow \pi[\bar{Z}](\pi[\bar{Y}](Q)) \equiv \pi[\bar{Z}](Q)$ .

### Selections

- $\sigma[\alpha_1](\sigma[\alpha_2](Q)) \equiv \sigma[\alpha_2](\sigma[\alpha_1](Q)) \equiv \sigma[\alpha_1 \wedge \alpha_2](Q)$ .
- $\text{attr}(\alpha) \subseteq \bar{Y} \subseteq \bar{X} \Rightarrow \pi[\bar{Y}](\sigma[\alpha](Q)) \equiv \sigma[\alpha](\pi[\bar{Y}](Q))$ .

### Joins

- $Q_1 \bowtie Q_2 \equiv Q_2 \bowtie Q_1$ .
- $(Q_1 \bowtie Q_2) \bowtie Q_3 \equiv Q_1 \bowtie (Q_2 \bowtie Q_3)$ .

## EQUIVALENCE OF EXPRESSIONS (CONT'D)

### Joins and other Operations

- $\text{attr}(\alpha) \subseteq \bar{X}_1 \cap \bar{X}_2 \Rightarrow \sigma[\alpha](Q_1 \bowtie Q_2) \equiv \sigma[\alpha](Q_1) \bowtie \sigma[\alpha](Q_2)$ .
- $\text{attr}(\alpha) \subseteq \bar{X}_1, \text{attr}(\alpha) \cap \bar{X}_2 = \emptyset \Rightarrow \sigma[\alpha](Q_1 \bowtie Q_2) \equiv \sigma[\alpha](Q_1) \bowtie Q_2$ .
- Assume  $\bar{V} \subseteq \overline{\bar{X}_1 \bar{X}_2}$  and let  $\bar{W} = \bar{X}_1 \cap \overline{\bar{V} \bar{X}_2}$ ,  $\bar{U} = \bar{X}_2 \cap \overline{\bar{V} \bar{X}_1}$ .  
Then,  $\pi[\bar{V}](Q_1 \bowtie Q_2) \equiv \pi[\bar{V}](\pi[\bar{W}](Q_1) \bowtie \pi[\bar{U}](Q_2))$ ;
- $\bar{X}_2 = \bar{X}_3 \Rightarrow Q_1 \bowtie (Q_2 \text{ op } Q_3) \equiv (Q_1 \bowtie Q_2) \text{ op } (Q_1 \bowtie Q_3)$  where  $\text{op} \in \{\cup, \setminus\}$ .

### Exercise 3.2

Prove some of the equalities (use the definitions given on the "Base Operators" slide). □

111

## EXPRESSIVE POWER OF THE ALGEBRA

### Transitive Closure

The transitive closure of a binary relation  $R$ , denoted by  $R^*$  is defined as follows:

$$\begin{aligned}
 R^1 &= R \\
 R^{n+1} &= \{(a, b) \mid \text{there is an } s \text{ s.t. } (a, s) \in R^n \text{ and } (s, b) \in R\} \\
 R^* &= \bigcup_{1.. \infty} R^n
 \end{aligned}$$

Examples:

- $\text{child}(x, y)$ :  $\text{child}^* = \text{descendant}$
- flight connections
- $\text{flows\_into}$  of rivers in MONDIAL

### Theorem 3.2

There is no expression of the relational algebra that computes the transitive closure of arbitrary binary relations  $r$ . □

112

## EXAMPLES

Time to play. Perhaps postpone examples after comparison with SQL (next subsections)

### Aspects

- join as “extending” operation (cartesian product – “all pairs of X and Y such that ...”)
- equijoin as “restricting” operation
- natural join/equijoin in many cases along key/foreign key relationships
- relational division (in case of queries of the style “return all X that are in a given relation with all Y such that ...”)