Relational Algebra

Prof Dr Melike Şah Direkoğlu

Slide credit: Juliana Freire, Yasemin Bay Ayzeren

DBMS Architecture

How does a SQL query engine work ?

- SQL query $\rightarrow$ relational algebra plan
- Relational algebra plan $\rightarrow$ Optimized plan
- Execute each operator of the plan

What is an ‘Algebra’

- Mathematical system consisting of:
  - **Operands;** variables or values from which new values can be constructed.
  - **Operators;** symbols denoting procedures that construct new values from given values.

Is this the Algebra you know?

Algebra -> operators and operands

Expressions -> applying operators to atomic operands and/or other expressions

Algebra of arithmetic expression: operands are variables (i.e. x, y) and constants (i.e. 2, 7, 3), and operators are the usual arithmetic operators (i.e. +, *, -)

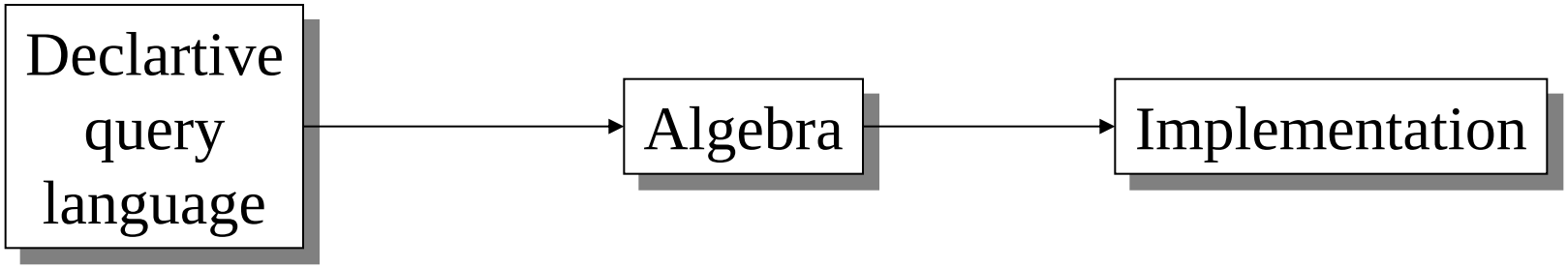
E.g., (x+y)*2 or ((x+7)/(y-3)) + x

What is Relational Algebra?

- An algebra whose **operands are variables that represent relations**.
- Operators include *union, intersection, selection, projection, Cartesian product*, *etc.* are designed to do the most common things that we need to do with relations in a database.
  - The result is an algebra that can be used as a **query language for relations**
  - E.g., (π ownerChecking-account) ∩ (π ownerSavings-account)

Relational Algebra

- Formalism for creating new relations from existing ones
- Its place in the big picture:



SQL,
relational calculus

Relational algebra

Relational Algebra

- Five operators:
  - Union: $\cup$
  - Difference: -
  - Selection: σ
  - Projection: Π
  - Cartesian Product: $\times$
- Derived or auxiliary operators:
  - Intersection, complement
  - Joins (**natural**, left join, right join etc.)
  - Renaming: ρ

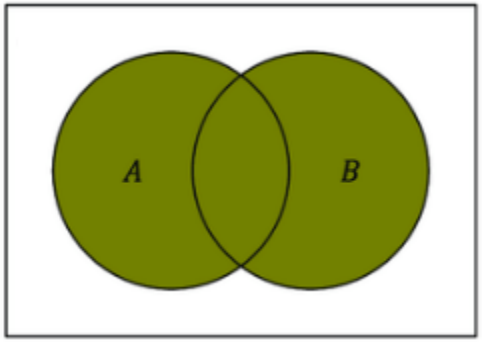
Core Relational Algebra

- Union, intersection, and difference: Usual set operations, but both operands must have the same relation schema.
- Selection: Picking certain rows.
- Projection: Picking certain columns.
- Products and joins: Compositions of relations.
- Renaming of relations and attributes.

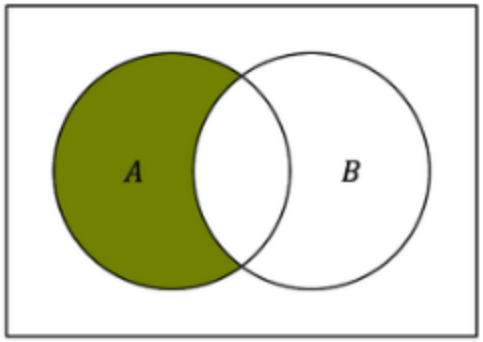
Relational Algebra Notations

- Five operators:
  - Union: $\cup$
  - Difference: -
  - Selection: σ
  - Projection: Π
  - Cartesian Product: $\times$
- Derived or auxiliary operators:
  - Intersection, complement
  - Joins (natural,equi-join, theta join, semi-join)
  - Renaming: ρ

1. Union and 2. Difference



- R1 ∪ R2 (remove duplicates)
- Example (AllEmployees):
  - ActiveEmployees ∪ RetiredEmployees



- R1 – R2
- Example (ActiveEmployees):
  - AllEmployees -- RetiredEmployees

Union (Distinct) Operation– Example

Relations *r, s:*

A	B
α	1
α	2
β	1

r

A	B
α	2
β	3

s

n $r \cup s$:

A	B
α	1
α	2
β	1
β	3

Union Example

Union-Distinct

Table1	
column1	column2
a	b
a	c
a	d

U

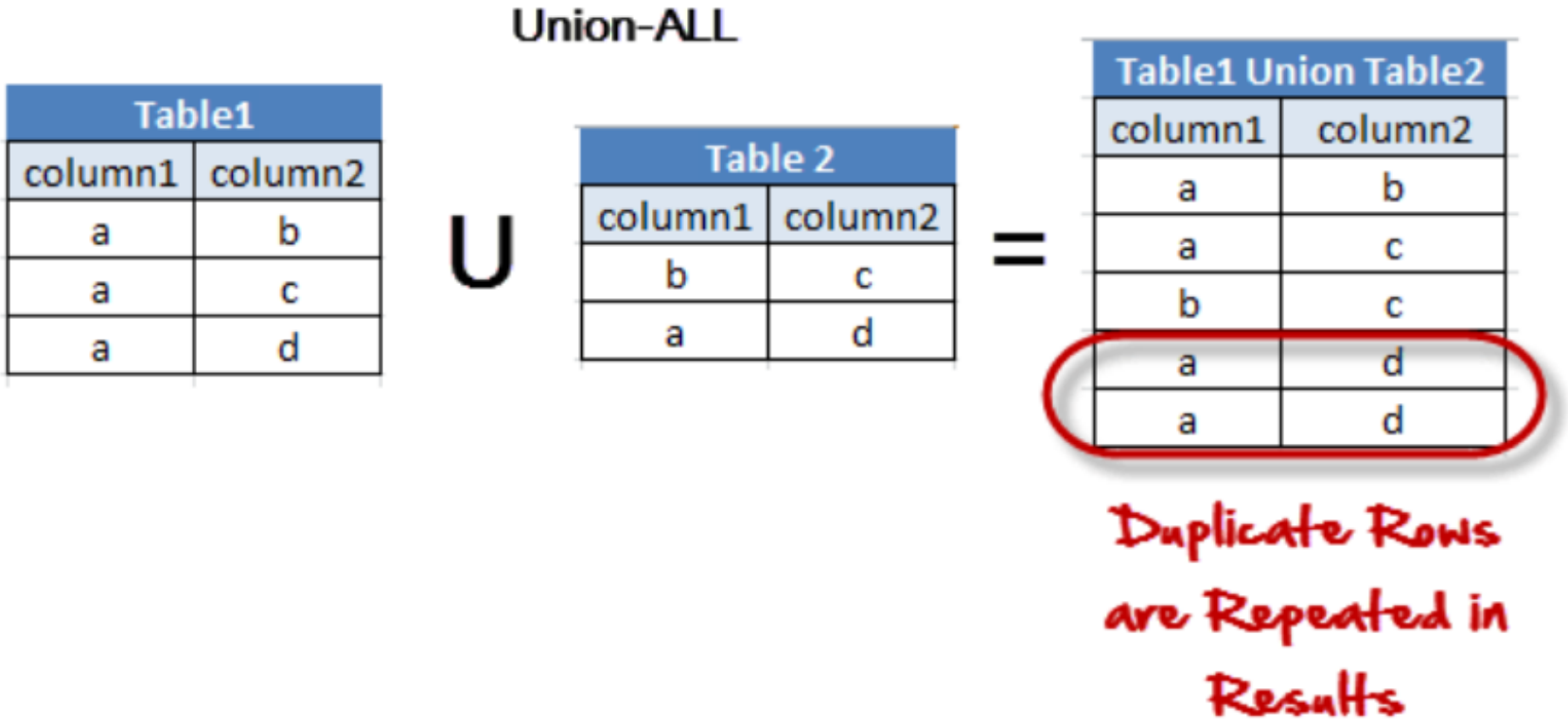
Table 2	
column1	column2
b	c
a	d

=

Table1 Union Table2	
column1	column2
a	b
a	c
a	d
b	c

Duplicate row not repeated in results

Union Example



Set Difference Operation – Example

- Relations r, s

A	B
α	1
α	2
β	1

r

A	B
α	2
β	3

s

- $r - s$:

A	B
α	1
β	1

Set Difference Operation – Example

R

A	B
1	2
3	4

S

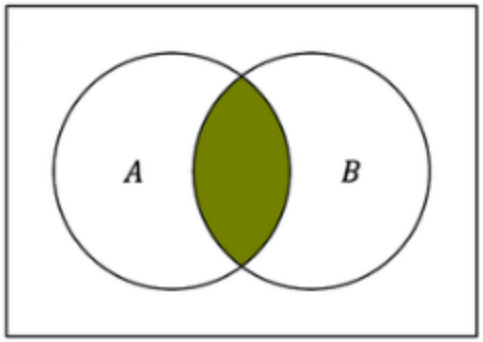
A	B
1	2
5	6

R – S

A	B
3	4

What about Intersection ?

- It is a derived operator, cannot be directly implemented. We can use the set difference:
- $R1 \cap R2 = R1 - (R1 - R2)$
- Also expressed as a join (will see later)
- Example
  - UnionizedEmployees $\cap$ RetiredEmployees



3. Selection

- Returns all tuples which satisfy a condition (Selects certain rows (tuples/records)
- Notation: $\sigma_c(R)$
- Examples
  - $\sigma_{Salary > 40000}$ (Employee)
  - $\sigma_{name = \text{“Smith”}}$ (Employee)
- The condition c can be connective operators $=, <, \leq, >, \geq, <>$
- Logical operators $\wedge$ (and), $\vee$ (or), $\neg$ (not)

SSN	Name	Salary
1234545	John	200000
5423341	Smith	600000
4352342	Fred	500000

$\sigma_{\text{Salary} > 400000}$ (Employee)

SSN	Name	Salary
5423341	Smith	600000
4352342	Fred	500000

SSN	Name	Salary
1234545	John	200000
5423341	Smith	600000
4352342	Fred	500000

$\sigma_{\text{Name} = \text{“Smith”}}$ (Employee)

SSN	Name	Salary
5423341	Smith	600000

SSN	Name	Salary
1234545	John	200000
5423341	Smith	600000
4352342	Fred	500000

$$\sigma_{\neg (\text{Name} = \text{“Smith”} \vee \text{Name} = \text{“Fred”})}(\text{Employee})$$

SSN	Name	Salary
1234545	John	200000

SSN	Name	Salary
1234545	John	200000
5423341	Smith	600000
4352342	Fred	500000

$$\sigma_{(\text{Name} = \text{“Smith”} \wedge \text{Salary} > 500000)}(\text{Employee})$$

SSN	Name	Salary
5423341	Smith	600000

4. Projection

- Eliminates columns, then removes duplicates (Selecting certain Attributes and removes duplicates)
- Notation: $\Pi_{A1,\ldots,An}(R)$
- Example: project social-security number and names:
  - $\Pi_{SSN,\ Name}$ (Employee)
  - Output schema: Answer(SSN, Name)

SSN	Name	Salary
1234545	John	200000
5423341	Alice	600000
4352342	Jane	250000

$\Pi_{\text{Name,Salary}}$ (Employee)

Name	Salary
John	200000
Alice	600000
Jane	250000

SSN	Name	Salary
1234545	John	200000
5423341	Alice	600000
4352342	Jane	250000

$\Pi_{\text{SSN, Salary}} (\sigma_{\text{Salary} < 300000)}) (\text{Employee})$

SSN	Salary
1234545	200000
4352342	250000

Comparing Select and Project Queries

EMPLOYEE

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dno
John	B	Smith	123456789	1965-01-09	731 Fondren, Houston, TX	M	30000	333445555	5
Franklin	T	Wong	333445555	1955-12-08	638 Voss, Houston, TX	M	40000	888665555	5
Alicia	J	Zelaya	999887777	1968-01-19	3321 Castle, Spring, TX	F	25000	987654321	4
Jennifer	S	Wallace	987654321	1941-06-20	291 Berry, Bellaire, TX	F	43000	888665555	4
Ramesh	K	Narayan	666884444	1962-09-15	975 Fire Oak, Humble, TX	M	38000	333445555	5
Joyce	A	English	453453453	1972-07-31	5631 Rice, Houston, TX	F	25000	333445555	5
Ahmad	V	Jabbar	987987987	1969-03-29	980 Dallas, Houston, TX	M	25000	987654321	4
James	E	Borg	888665555	1937-11-10	450 Stone, Houston, TX	M	55000	NULL	1

$\sigma_{(Dno=4\ \textbf{AND}\ Salary>25000)\ OR\ (Dno=5\ \textbf{AND}\ Salary>30000)}(EMPLOYEE)$.

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dno
Franklin	T	Wong	333445555	1955-12-08	638 Voss, Houston, TX	M	40000	888665555	5
Jennifer	S	Wallace	987654321	1941-06-20	291 Berry, Bellaire, TX	F	43000	888665555	4
Ramesh	K	Narayan	666884444	1962-09-15	975 Fire Oak, Humble, TX	M	38000	333445555	5

Comparing Select and Project Queries

EMPLOYEE

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dno
John	B	Smith	123456789	1965-01-09	731 Fondren, Houston, TX	M	30000	333445555	5
Franklin	T	Wong	333445555	1955-12-08	638 Voss, Houston, TX	M	40000	888665555	5
Alicia	J	Zelaya	999887777	1968-01-19	3321 Castle, Spring, TX	F	25000	987654321	4
Jennifer	S	Wallace	987654321	1941-06-20	291 Berry, Bellaire, TX	F	43000	888665555	4
Ramesh	K	Narayan	666884444	1962-09-15	975 Fire Oak, Humble, TX	M	38000	333445555	5
Joyce	A	English	453453453	1972-07-31	5631 Rice, Houston, TX	F	25000	333445555	5
Ahmad	V	Jabbar	987987987	1969-03-29	980 Dallas, Houston, TX	M	25000	987654321	4
James	E	Borg	888665555	1937-11-10	450 Stone, Houston, TX	M	55000	NULL	1

$$\pi_{\text{Lname, Fname, Salary}}(\text{EMPLOYEE}).$$

Lname	Fname	Salary
Smith	John	30000
Wong	Franklin	40000
Zelaya	Alicia	25000
Wallace	Jennifer	43000
Narayan	Ramesh	38000
English	Joyce	25000
Jabbar	Ahmad	25000
Borg	James	55000

5. Cartesian Product

- Combining tuples of two relations R1 and R2
- Each tuple in R1 with each tuple in R2
- Notation: R1 × R2
- Example:
  - Employee × Dependents
- Very rare in practice; mainly used to express joins

Cartesian-Product Operation – Example

- Relations *r*, *s*:

A	B
α	1
β	2

r

C	D	E
α	10	a
β	10	a
β	20	b
γ	10	b

s

- *r* x *s*:

A	B	C	D	E
α	1	α	10	a
α	1	β	10	a
α	1	β	20	b
α	1	γ	10	b
β	2	α	10	a
β	2	β	10	a
β	2	β	20	b
β	2	γ	10	b

Cartesian Product Example

Employee

Name	SSN
John	999999999
Tony	777777777

Dependents

EmployeeSSN	Dname
999999999	Emily
777777777	Joe

Employee x Dependents

Name	SSN	EmployeeSSN	Dname
John	999999999	999999999	Emily
John	999999999	777777777	Joe
Tony	777777777	999999999	Emily
Tony	777777777	777777777	Joe

Cartesian Product Example

Employee

emp_no	ename	salary	dept_no
001	Smith	2500	1
002	Adams	3000	1
003	Mary	2750	2
004	John	3000	1

Department

dept_no	dname
1	Accounting
2	Sales

Employee × Department

emp_no	ename	salary	dept_no	dept_no	dname
001	Smith	2500	1	1	Accounting
001	Smith	2500	1	2	Sales
002	Adams	3000	1	1	Accounting
002	Adams	3000	1	2	Sales
003	Mary	2750	2	1	Accounting
003	Mary	2750	2	2	Sales
004	John	3000	1	1	Accounting
004	John	3000	1	2	Sales

Cartesian Product Example (Cont.)

Employee × Department

Employee.dept_no **Department.dept_no**

emp_no	ename	salary	dept_no	dept_no	dname
001	Smith	2500	1	1	Accounting
001	Smith	2500	1	2	Sales
002	Adams	3000	1	1	Accounting
002	Adams	3000	1	2	Sales
003	Mary	2750	2	1	Accounting
003	Mary	2750	2	2	Sales
004	John	3000	1	1	Accounting
004	John	3000	1	2	Sales

$\sigma_{\text{Employee.dept\_no=Department.dept\_no}}$ (Employee × Department)

emp_no	ename	salary	dept_no	dept_no	dname
001	Smith	2500	1	1	Accounting
002	Adams	3000	1	1	Accounting
003	Mary	2750	2	2	Sales
004	John	3000	1	1	Accounting

Natural Join

- Notation: R1 |×| R2
- Meaning: $R1 \mid\times\mid R2 = \Pi_A(\sigma_C(R1 \times R2))$
- Where:
  - The selection σ_C checks equality of all common attributes
  - The projection eliminates the duplicate common attributes

Natural Join Example

Employee

Name	SSN
John	999999999
Tony	777777777

Dependents

SSN	Dname
999999999	Emily
777777777	Joe

Employee $\bowtie$ **Dependents =**

$\Pi_{\text{Name, SSN, Dname}}(\sigma_{\text{SSN=SSN}}(\text{Employee} \times \rho_{\text{SSN, Dname}}(\text{Dependents}))$

Name	SSN	Dname
John	999999999	Emily
Tony	777777777	Joe

Natural Join

- R=

A	B
X	Y
X	Z
Y	Z
Z	V

S=

B	C
Z	U
V	W
Z	V

- R |×| S=

A	B	C
X	Z	U
X	Z	V
Y	Z	U
Y	Z	V
Z	V	W

Join: Example

⋈ = join

Account ⋈ $_{\text{Number=Account}}$ Deposit

Account	Number	Owner	Balance	Type
	101	J. Smith	1000.00	checking
	102	W. Wei	2000.00	checking
	103	J. Smith	5000.00	savings
	104	M. Jones	1000.00	checking
	105	H. Martin	10,000.00	checking

Deposit	Account	Transaction-id	Date	Amount
	102	1	10/22/00	500.00
	102	2	10/29/00	200.00
	104	3	10/29/00	1000.00
	105	4	11/2/00	10,000.00

Number	Owner	Balance	Type	Account	Transaction-id	Date	Amount
102	W. Wei	2000.00	checking	102	1	10/22/00	500.00
102	W. Wei	2000.00	checking	102	2	10/29/00	200.00
104	M. Jones	1000.00	checking	104	3	10/29/00	1000.00
105	H. Martin	10,000.00	checking	105	4	11/2/00	10000.00

Join : Example

⋈ join Account $\bowtie_{\text{Number=Account}}$ Deposit

Note that when the join is based on equality, then we have two identical attributes (columns) in the answer.

Number	Owner	Balance	Type	Account	Trans-id	Date	Amount
102	W. Wei	2000.00	checking	102	1	10/22/00	500.00
102	W. Wei	2000.00	checking	102	2	10/29/00	200.00
104	M. Jones	1000.00	checking	104	3	10/29/00	1000.00
105	H. Martin	10,000.00	checking	105	4	11/2/00	10000.00

6. Renaming

- Changes the schema (attribute names), not the instance
- Notation: $\rho_{B1,\ldots,Bn}(R)$
- Example:
  - $\rho_{\text{LastName, SocSocNo}}$ (Employee)
  - Output schema:
    Answer(LastName, SocSocNo)

Renaming Example

Employee

Name	SSN
John	999999999
Tony	777777777

$\rho_{LastName,\ SocSocNo}$ **(Employee)**

LastName	SocSocNo
John	999999999
Tony	777777777

Renaming Example

Employee

emp_no	ename	salary	dept_no
001	Smith	2500	1
002	Adams	3000	1
003	Mary	2750	2
004	John	3000	1

Π emp_no, ename, P Annual Salary (salary*12), dept_no (employee)

emp_no	ename	Annual Salary	dept_no
001	Smith	30,000	1
002	Adams	36,000	1
003	Mary	33,000	2
004	John	36,000	1