Database Systems

Instructor: Bilal Khalid Dar

Relational Algebra

DBMS Architecture

How does a SQL engine work ?

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

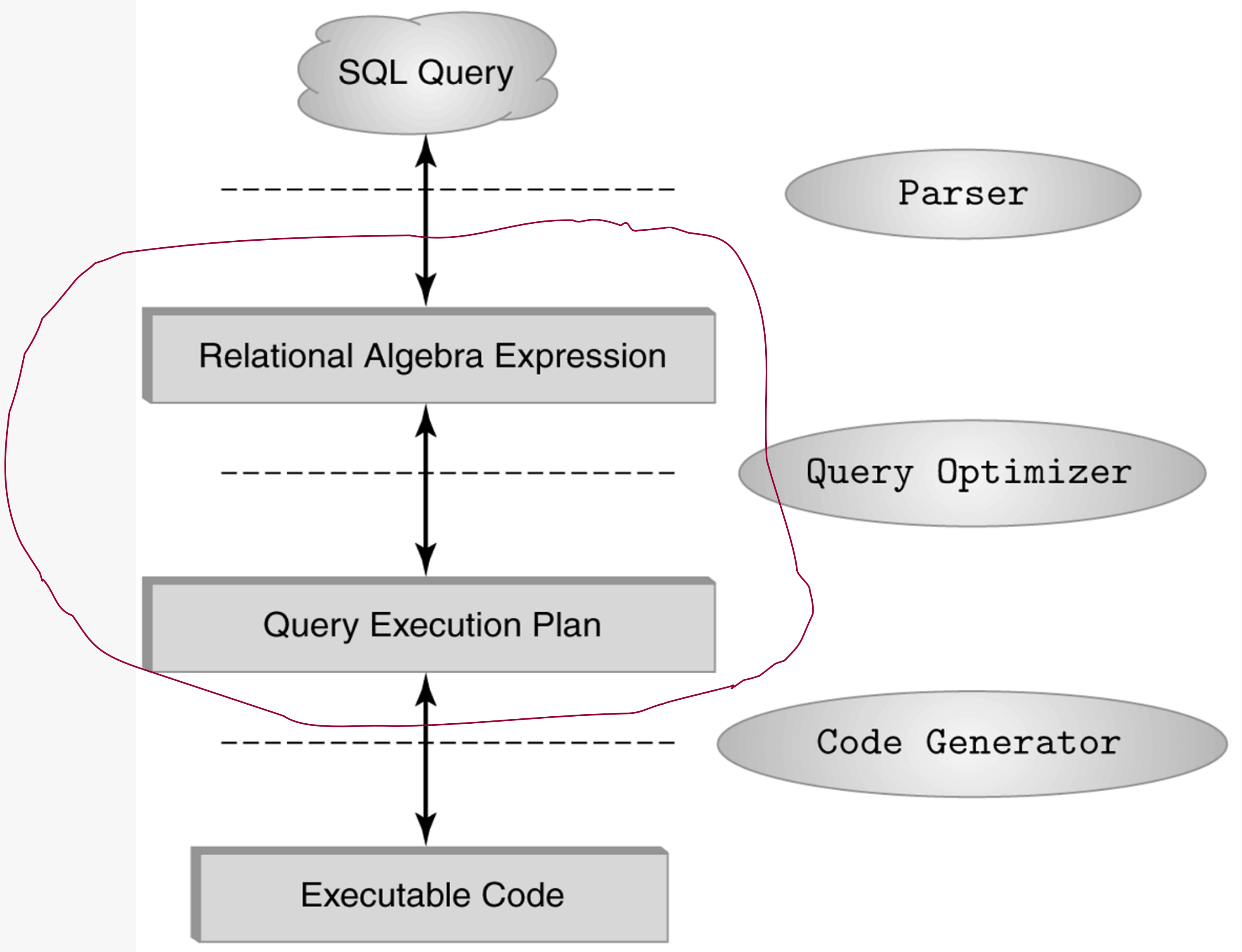
What is an Algebra?

- A language based on operators and a domain of values
- Operators map values taken from the domain into other domain values
- Hence, an expression involving operators and arguments produces a value in the domain
- When the domain is a set of all relations (and the operators are as described later), we get the *relational algebra*
- We refer to the expression as a *query* and the value produced as the *query result*

Relational Algebra

- 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

The Role of Relational Algebra in a DBMS



1. Union and 2. Difference

- R1 ∪ R2
- Example:
  - ActiveEmployees ∪ RetiredEmployees

- R1 – R2
- Example:
  - AllEmployees -- RetiredEmployees

What about Intersection ?

- It is a derived operator
- $R1 \cap R2 = R1 - (R1 - R2)$
- Also expressed as a join (will see later)
- Example
  - UnionizedEmployees $\cap$ RetiredEmployees

Union Compatible Relations

- Two relations are *union compatible* if
  - Both have same number of columns
  - Names of attributes are the same in both
  - Attributes with the same name in both relations have the same domain
- Union compatible relations can be combined using *union*, *intersection*, and *set difference*

Example

Tables:

Person (*SSN, Name, Address, Hobby*)

Professor (*Id, Name, Office, Phone*)

are <u>not</u> union compatible.

But

π_{Name} (Person) and π_{Name} (Professor)

<u>are</u> union compatible so

π_{Name} (Person) - π_{Name} (Professor)

makes sense.

3. Selection

- Returns all tuples which satisfy a condition
- Notation: $\sigma_c(R)$
- Examples
  - $\sigma_{\text{Salary > 40000}}$ (Employee)
  - $\sigma_{\text{name = “Smith”}}$ (Employee)
- The condition c can be =, <, $\leq$, >, $\geq$, <>

Select Operator

- Produce table containing subset of rows of argument table satisfying condition

$$\sigma_{condition}\,(relation)$$

- Example:

Person

Id	*Name*	*Address*	*Hobby*
1123	John	123 Main	stamps
1123	John	123 Main	coins
5556	Mary	7 Lake Dr	hiking
9876	Bart	5 Pine St	stamps

$\sigma_{Hobby=\text{'stamps'}}(\text{Person})$

Id	*Name*	*Address*	*Hobby*
1123	John	123 Main	stamps
9876	Bart	5 Pine St	stamps

Selection Condition

- Operators: $<, \leq, \geq, >, =, \neq$
- Simple selection condition:
  - *<attribute> operator <constant>*
  - *<attribute> operator <attribute>*
- *<condition>* AND *<condition>*
- *<condition>* OR *<condition>*
- NOT *<condition>*

Selection Condition - Examples

- $\sigma_{\,Id>3000\ \text{OR}\ Hobby=\text{'hiking'}}$ (Person)
- $\sigma_{\,Id>3000\ \text{AND}\ Id<3999}$ (Person)
- $\sigma_{\,\text{NOT}(Hobby=\text{'hiking'})}$ (Person)
- $\sigma_{\,Hobby\neq\text{'hiking'}}$ (Person)

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

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

SSN	Name	Salary
5423341	Smith	600000
4352342	Fred	500000

4. Projection

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

SSN	Name	Salary
1234545	John	200000
5423341	John	600000
4352342	John	200000

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

Name	Salary
John	20000
John	60000

Project Operator

- Produces table containing subset of columns of argument table

$$\pi_{\textit{attribute list}}(\textit{relation})$$

- Example:

Person

Id	Name	Address	Hobby
1123	John	123 Main	stamps
1123	John	123 Main	coins
5556	Mary	7 Lake Dr	hiking
9876	Bart	5 Pine St	stamps

$\pi_{Name,Hobby}$(Person)

Name	Hobby
John	stamps
John	coins
Mary	hiking
Bart	stamps

Project Operator

- Example:

Person

Id	*Name*	*Address*	*Hobby*
1123	John	123 Main	stamps
1123	John	123 Main	coins
5556	Mary	7 Lake Dr	hiking
9876	Bart	5 Pine St	stamps

$\pi_{Name,Address}$(Person)

Name	*Address*
John	123 Main
Mary	7 Lake Dr
Bart	5 Pine St

Result is a table (no duplicates); can have fewer tuples than the original

Expressions

$$\pi_{\textit{Id, Name}} (\sigma_{\textit{Hobby}=\text{'stamps'} \text{ OR } \textit{Hobby}=\text{'coins'}} (\text{Person}))$$

Id	*Name*	*Address*	*Hobby*
1123	John	123 Main	stamps
1123	John	123 Main	coins
5556	Mary	7 Lake Dr	hiking
9876	Bart	5 Pine St	stamps

Person

Id	*Name*
1123	John
9876	Bart

Result

5. Cartesian Product

- Each tuple in R1 with each tuple in R2
- Notation: $R1 \times R2$
- Example:
  - $Employee \times Dependents$
- Very rare in practice; mainly used to express joins

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

- If R and S are two relations, $R \times S$ is the set of all concatenated tuples <*x*,*y*>, where *x* is a tuple in R and *y* is a tuple in S
  - R and S need not be union compatible.
  - *But* R and S must have distinct attribute names. Why?
- $R \times S$ is expensive to compute. But why?

R:

A	*B*
x1	x2
x3	x4

S:

C	*D*
y1	y2
y3	y4

$R \times S$:

A	*B*	*C*	*D*
x1	x2	y1	y2
x1	x2	y3	y4
x3	x4	y1	y2
x3	x4	y3	y4

Relational Algebra

- 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

Renaming

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

Renaming Example

Employee	
Name	SSN
John	999999999
Tony	777777777

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

LastName	SocSocNo
John	999999999
Tony	777777777

Example – Second Method

Transcript (*StudId*, *CrsCode*, *Semester*, *Grade*)

Teaching (*ProfId*, *CrsCode*, *Semester*)

$\pi_{StudId,\ CrsCode}$ (Transcript)[*StudId*, *CrsCode1*]

$\times$ $\pi_{ProfId,\ CrsCode}$(Teaching) [*ProfId*, *CrsCode2*]

This is a relation with 4 attributes:

StudId, *CrsCode1*, *ProfId*, *CrsCode2*


Thank
You