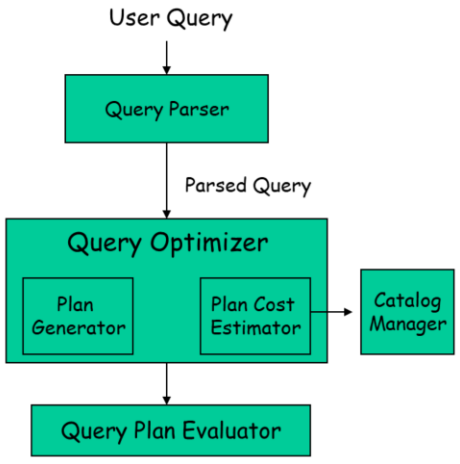
Overview of Query Evaluation

- A user query is expressed using SQL or other languages, ideally natural languages
- A parsed query is essentially treated as a relational algebra expression
  - Selection (б)
  - Join (⋈)
  - Project(Π)
  - Union, intersection and difference, cross product
- Query optimizer
  - Enumerates the possible plans to evaluate expression,
  - Select a small subset of these plans and estimate their cost



This lecture will introduce Relational Algebra

Relational Algebra

- What is it?
- Why we need it?

Think about algebra.

For example (a^2+b)*c/d.

Algebra has symbols defining for different operations.

These operations has different level of priority. So (a^2+b)*c/d is different from a^2+b*c/d

We can rewrite the same formula for faster calculation. (a^2+b)*c/d = a^2*c/d+b*c/d, but the left hand side have less operations, so faster to calculate

Union, Intersection and Difference Operators

- These are defined if r and s have the same schema
  - $r \cup s = \{x: x \in r \vee x \in s\}$; its schema is same as that of r or s
  - $r - s = \{x: x \in r \wedge x \notin s\}$; its schema is same as that of r or s
  - $r \cap s = \{x: x \in r \wedge x \in s\}$; its schema is same as that of r or s

- Example

r

A	B
1	2
3	4
5	6
7	8

s

A	B
1	3
3	4
5	6

$r \cup s$

A	B
1	2
3	4
5	6
7	8
1	3

$r - s$

A	B
1	2
7	8

$r \cap s$

A	B
3	4
5	6

Start with the basic

Relations are sets of tuples

So all set operators can be applied here (but there is a requirement: the schema of the two relations has to be the same)

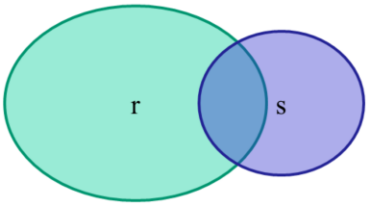
Union ∪

Intersection ∩

Difference −

Question

- Is $r \cap s = s \cap r$
- Is $r \cup s = s \cup r$
- Is $r - s = s - r$



Select Operator

- $\sigma_c(r)$ selects those tuples of r that satisfy condition c
- Examples of conditions
  - $A \leq B$,
  - $A > 2 \wedge A < B$
- $\sigma_c(r) = \{x: x \in r \wedge x \text{ satisfies condition } c\}$
- Example

r

A	B
1	2
3	4
5	6
7	6

$\sigma_{A > 2 \wedge A < B}(r)$

A	B
3	4
5	6

Sigma σ

Question

- Do you remember what we learned in the Chapter on Select?

Project Operator

- $\Pi_X(r)$ retains only X columns
  - In $\Pi_X(r)$, X must consist of some attributes of r
- The schema of $\Pi_X(r)$ is X
- $\Pi_X(r) = \{x[X]: x \in r\}$
  - Here x[X] includes only attributes in X, other attributes are excluded
- Example

r

A	B	C
1	2	4
5	4	2
5	6	2
7	8	3

$\Pi_{AC}(r)$

A	C
1	4
5	2
7	3

- $\Pi_{AC}(r)$ is a set from which a "duplicate tuple was removed"

Pi π

Row VS Column

- In relational algebra, “Selection” is on rows (i.e., which rows satisfy the conditions), while “Projection” is on columns.
- Don’t get confused with the Select statement in SQL.
  - Select name from Students where id=“001”
  - This sql is $\pi_{name}(\sigma_{id="001"}(Students))$

$\pi_{name}(\sigma_{id="001"}(Students))$ chooses the row first and then choose the column. Can we switch these two operations?

Natural Join Operator

- $r \bowtie s$ combines tuples of r and s agreeing on common attributes
- $r \bowtie s$ is always defined
- Suppose schema of r is R and schema of s is S, Then schema of $r \bowtie s$ is $R \cup S$
- $r \bowtie s = \{x : x[R] \in r \wedge x[S] \in s\}$
  - For a tuple to be in $r \bowtie s$, its R portion should be in r and S portion in S
  - Implicitly, R and S portions match on the common attributes $R \cap S$
- Example

r

A	B
1	2
3	4
5	6
9	4
10	5

s

B	C
2	3
4	4
4	6
9	4

$r \bowtie s$

A	B	C
1	2	3
3	4	4
3	4	6
9	4	4
9	4	6

Join $\bowtie$

Question

- Do you remember what we learned in Chapter on Join?

Renaming Operator

- Sometimes it becomes necessary to rename attributes or relations before or after an operator is applied
- Renaming does not alter the information content, i.e. tuples
- We use the syntactic form $\alpha \rightarrow \beta$ to express that the identifier α is changed to β.

- Example
  - Consider $r \cup (s: C \rightarrow B): \rightarrow q$
  - Here, attribute C in s is renamed to B
  - After computing the union the result is named as q
  - ":" has the lowest priority

r

A	B
1	2
3	4
5	6

s

A	C
1	3
3	4
5	6

q

A	B
1	2
3	4
5	6
1	3

Cross Product Operator

- $r \times s$ concatenates tuples of r and s
- Every common attribute in r and s renamed
- In renaming prefix "r." and "s." are used to make attributes unique across r and s
- $r \times s = \{x \circ y : x \in \text{renamed } r \wedge y \in \text{renamed } s\}$
  - Here, $\circ$ denotes concatenation of two tuples to form a larger tuple
  - The schema of $r \times s$ consists of all attributes of r and s after renaming
- Example

r

A	B
1	2
3	4
5	6

s

B	C
4	2
7	3

$r \times s$

A	r.B	s.B	C
1	2	4	2
1	2	7	3
3	4	4	2
3	4	7	3
5	6	4	2
5	6	7	3

Cross product (aka Cartesian product)

Additional Join Operators

- $r \bowtie_c s = \{x \circ y : x \in \text{renamed } r \wedge y \in \text{renamed } s \wedge x \circ y \text{ satisfies } c\}$
  - The schema of $r \bowtie_c s$ consists of all attributes of r and s after renaming
- Example

r

A	B
1	2
3	4
5	6

s

B	C
4	2
4	3

$r \bowtie_{r.B \neq s.B} s$

A	r.B	s.B	C
1	2	4	2
1	2	4	3
5	6	4	2
5	6	4	3

Question

- Can you link natural join with cross product?

- $s \times r$
- $s \bowtie r$

Relational Algebra as a Query Language

Emp

Name	DName	Salary
John	Toys	50K
Mary	Toys	60K
Leu	Shoes	55K

Dept

DName	MName
Toys	Mary

- *Give names and salaries of employees in Toys or Credit.*

$\Pi_{\text{Name, Salary}}(\sigma_{\text{DName = 'Toys'} \vee \text{DName = 'Credit'}} (\text{Emp}))$

Name	Salary
John	50K
Mary	60K

- Another algebraic expression for the query is as follows:

$$\Pi_{\text{Name, Salary}} (\sigma_{\text{DName = 'Toys'}} (\text{Emp})) \cup \Pi_{\text{Name, Salary}}(\sigma_{\text{DName='Credit'}} (\text{Emp}))$$

- Different users may express the query in different ways

Relational Algebra as a Query Language

Emp

Name	DName	Salary
John	Toys	50K
Mary	Toys	60K
Leu	Shoes	55K

Dept

DName	MName
Toys	Mary

- Who is John's manager?

  $\Pi_{MName}(\sigma_{emp.DName=Dept.DName \wedge Name='John'}(Emp \times Dept))$

  - Observe how operators are composed
  - Retrieves the following relation

MName
Mary

Quiz on Canvas

- Access code: sailors
- 2 attempts allowed
- Open book, open notes, no communication

Exercise

Sailors(sid, sname, rating, age)
Boats(bid, bname, color)
Reserve(sid, bid, day)

- Find the names of sailors who have reserved boat 103

$$\pi_{sname}\left(\sigma_{bid=103}(Reserves \bowtie Sailors)\right)$$

Exercise

Sailors(sid, sname, rating, age)
Boats(bid, bname, color)
Reserve(sid, bid, day)

- Find the names of sailors who have reserved a red boat

$$\pi_{sname}\left(\left(\sigma_{color="red"}(Reserve \bowtie Boats)\right) \bowtie sailors\right)$$

Exercise

Sailors(sid, sname, rating, age)
Boats(bid, bname, color)
Reserve(sid, bid, day)

- Find the colors of boats reserved by Lubber

$$\pi_{color}\left(\left(\sigma_{sanme="Lubber"}(Reserve \bowtie Sailors)\right) \bowtie Boats\right)$$

Sailors(<u>sid</u>, sname, rating, age)
Boats(<u>bid</u>, bname, color)
Reserve(<u>sid, bid, day</u>)

- Find the names of sailors who have reserved a red or a green boat

$$\pi_{sname}\left(\left(\sigma_{color=\text{"red"} \vee color=\text{"green"}}(Reserve \bowtie Boats)\right) \bowtie sailors\right)$$

$$\pi_{sname}\left(\left(\sigma_{color=\text{"red"}}(Reserve \bowtie Boats)\right) \bowtie sailors\right)$$

$$\cup\ \pi_{sname}\left(\left(\sigma_{color=\text{"red"}}(Reserve \bowtie Boats)\right) \bowtie sailors\right)$$

Sailors(sid, sname, rating, age)
Boats(bid, bname, color)
Reserve(sid, bid, day)

- Find the names of sailors who have reserved a red and a green boat

$\pi_{sname}\left(\left(\sigma_{color=\text{"red"} \wedge color=\text{"green"}}(Reserve \bowtie Boats)\right) \bowtie sailors\right)$ is wrong.

Because this means select the boats that are both red and green.

$\pi_{sname}\left(\left(\sigma_{color=\text{"red"}}(Reserve \bowtie Boats)\right) \bowtie sailors\right) \cap$

$\pi_{sname}\left(\left(\sigma_{color=\text{"red"}}(Reserve \bowtie Boats)\right) \bowtie sailors\right)$ can also run into problem, if two sailors have the same name.

It's better to use sid

$$\pi_{sname}\left(\pi_{sid}\left(\left(\sigma_{color=\text{"red"}}(Reserve \bowtie Boats)\right) \bowtie sailors\right)\right.$$

Sailors(sid, sname, rating, age)
Boats(bid, bname, color)
Reserve(sid, bid, day)

- Find the names of sailors who have reserved at least one boats

Sailors(sid, sname, rating, age)
Boats(bid, bname, color)
Reserve(sid, bid, day)

- Find the sids of sailors with age over 20 who have not reserved a red boat