Relational Algebra Operators

Relational Data Model

- Structure
  - TABLE or RELATION is the only element
- Value constraints
  - Unique or keys
  - NULLs
- Operations
  - Relational algebra or algebra for tables

TABLE Or Relation

- Schema or table header
  - Attributes or columns
  - Type or domain
    - Primitive: int, float, char[], string or varchar[]
    - Containers not allowed
- A table is seen as a collection (or multiset) of tuples
  - Cannot index in the table

Relational Algebra

- Set of operations or functions on tables
  - Input schema(s) → Output schema
  - Input tuples → Output tuples
- Single table operations
  - Select column, select tuple (row), aggregate, grouping
- Multiple table operations
  - Product and Join, Union, Intersection, Difference

Projection π

- Input table
  - T(A,B,C)
- **A B C**

  1 2 3

  3 4 6

  8 5 4

  7 4 3

- **$T' = \pi_{A,\ (A+B+C)\ AS\ S'}(T)$**
- Output table: T'
  - Schema
    - T'(A,S')
  - Same number of tuples as T
  - No duplicate elimination
- **A S'**

  1 6

  3 13

  8 17

  7 14

Selection σ

- Input table
  - T(A,B,C)
- **A B C**

  1 2 3

  3 4 6

  8 5 4

  7 4 3

- **T' = $\sigma_{A>1 \text{ AND } B+C>A}$(T)**
- Output table: T'
  - Schema
    - T'(A,B,C)
    - Same schema as T
  - Only tuples satisfying predicate
- **A B C**

  3 4 6

  8 5 4

Duplicate Elimination δ

- Input table

  T(A,B)

0	1
2	3
0	1
2	4
3	4

- **T' = δ(T)**
- Output table: T'
  - Schema
    - T'(A,B)
    - Same schema as T
  - Only distinct tuples
  - At most the same number of tuples from T

T'(A,B)

0	1
2	3
2	4
3	4

Sorting τ

- Input table
  T(A,B)

0	1
2	3
0	1
2	4
3	4

- **$T' = \tau_{B\ [DESC]}(T)$**
- Output table: T'
  - Schema
    - T'(A,B)
    - Same schema as T
  - Same tuples sorted

T'(A,B)

2	4
3	4
2	3
0	1
0	1

Aggregations
SUM, AVG, COUNT, MIN, MAX

- Input table

  T(A,B)

0	1
2	3
0	1
2	4
3	4

- **$T' = SUM_A(T)$**
- **$T'' = MAX_{A+B}(T)$**
- Output table: T'
  - Schema
    - T'(X)
  - Single tuple with aggregate result

T'(X)

7

T''(X)

7

GroupBy Aggregations γ

- Input table T(A,B)

0	1
2	3
0	1
2	4
3	4

- **$T' = \gamma_{A,\ MIN(B)\ AS\ MB}(T)$**
- Output table: T'
  - Schema
    - T'(A, MB)
    - Arguments of γ
  - Tuples have distinct values for A and group aggregate value for other attributes

T'(A,MB)

0	1
2	3
3	4

Set Operations U, ∩, -

- Input tables

R(A,B)		S(A,B)	
1	1	1	2
1	2	4	3
3	4		

- Schema of R, S, and result table T’ is the same (A,B)

- Union: T’ = R U S

1 1

1 2

3 4

4 3

- Difference: T’ = R - S

1 1

3 4

- Difference: T’ = S - R

4 3

- Intersection: T’ = R ∩ S

1 2

Cartesian Product x

- R(A) = {1,1,2,3}
- S(B) = {1,3,4}
- **T = R x S(A,B)** = {
  (1,1),(1,3),(1,4),
  (1,1),(1,3),(1,4),
  (2,1),(2,3),(2,4),
  (3,1),(3,3),(3,4)}
- The result consists of pairs of one element from R and one from S
- Every element from R is paired with every element from S
- The number of elements in R x S is |R|*|S|, i.e., the size of R multiplied by the size of S

- The schema of the result is the **union** of the R schema and the S schema
  - R(A)
  - S(B)
  - T(A,B) = A U B

Join $\bowtie$

- R(A) = {1,1,2,3}
- S(B) = {1,3,4}
- **T = R $\bowtie_{A=B}$ S** = {

  (1,1),~~(1,3),(1,4),~~

  (1,1),~~(1,3),(1,4),~~

  ~~(2,1),(2,3),(2,4),~~

  ~~(3,1)~~,**(3,3)**,~~(3,4)~~} = {(1,1),(1,1),(3,3)}
- Join condition between attributes from the two tables
- Only those tuples from the Cartesian product that satisfy the join condition are included in the result

- The schema of the result is the **union** of the R schema and the S schema
  - R(A)
  - S(B)
  - T(A,B) = A U B
- **R $\bowtie_{A=B}$ S = $\sigma_{A=B}$(RxS)**

Outer Joins

R(A,B)

0	1
2	3
0	1
2	4
3	4

S(B,C)

0	1
2	4
2	5
3	4
0	2
3	4

$R \bowtie S$ [natural join] (A,B,C)

2	3	4
2	3	4

$R \bowtie_o S$ [full outer join] (A,B,C)

2	3	4
2	3	4
0	1	-
0	1	-
2	4	-
3	4	-
-	0	1
-	2	4
-	2	5
-	0	2

Left (Right) Outer Joins

R(A,B)

0	1
2	3
0	1
2	4
3	4

S(B,C)

0	1
2	4
2	5
3	4
0	2
3	4

R $\bowtie_o$ S [full outer join]

(A,B,C)

2	3	4
2	3	4
0	1	-
0	1	-
2	4	-
3	4	-
-	0	1
-	2	4
-	2	5
-	0	2

R $\bowtie_L$ S

[left outer join]

(A,B,C)

2	3	4
2	3	4
0	1	-
0	1	-
2	4	-
3	4	-

R $\bowtie_R$ S

[right outer join]

(A,B,C)

2	3	4
2	3	4
-	0	1
-	2	4
-	2	5
-	0	2

Relational Algebra ↔ SQL

- SELECT ↔ Projection π
- FROM ↔ Input tables
- WHERE ↔ Selection σ, Join predicates
- DISTINCT ↔ Duplicate elimination δ
- ORDER BY ↔ Sorting τ
- GROUP BY ↔ GroupBy aggregations γ
- UNION, INTERSECT, EXCEPT ↔ Set operations $\cup$, $\cap$, -
- JOIN ↔ Join