

Database Management System

Relational Algebra in DBMS

RAMNA SATTAR



Relational Algebra in DBMS

What is Relational Algebra?

It collects instances of relations as input and gives occurrences of relations as output. It uses various operation to perform this action.

The output of these operations is a new relation, which might be formed from one or more input relations.

The relational algebra is a relation-at-a-time (or set) language where all tuples are controlled in one statement without the use of a loop. There are several variations of syntax for relational algebra commands, and you use a common symbolic notation for the commands.



Relational Algebra in DBMS

We can use Relational Algebra to fetch data from this Table(relation)

Select Name students with age less than 17

Output

Name
Ckon
Dkon

ID	Name	Age
1	Akon	17
2	Bkon	19
3	Ckon	15
4	Dkon	13

The output for query is also in form of a table(relation), with results in different columns



Relational Algebra in DBMS

Basic Relational Algebra Operations:

Relational Algebra divided in various groups

Unary Relational Operations

SELECT (symbol: σ)

PROJECT (symbol: π)

RENAME (symbol: ρ)

Relational Algebra Operations From Set Theory

UNION (\cup)

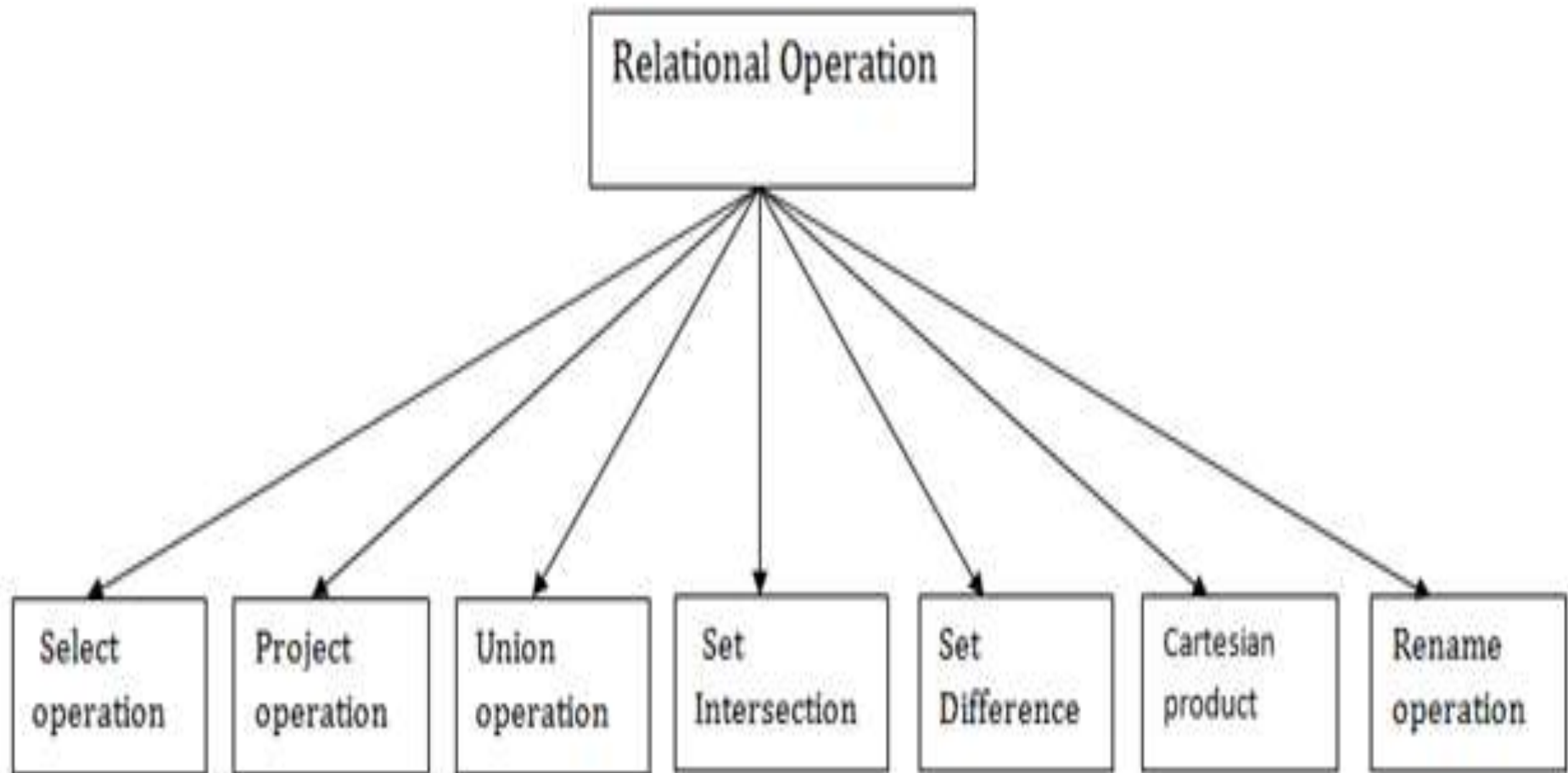
INTERSECTION (\cap)

DIFFERENCE ($-$)

CARTESIAN PRODUCT (\times)



Relational Algebra in DBMS



Relational Algebra in DBMS

SELECT (σ):

The SELECT operation is used for selecting a subset of the tuples according to a given selection condition. σ Symbol denotes it. It is used as an expression to choose tuples which meet the selection condition. Select operation selects tuples that satisfy a given predicate.

Syntax: $\sigma_p(r)$

σ is the predicate

r stands for relation which is the name of the table

p is propositional logic

Example 1

$\sigma_{\text{topic} = \text{"Database"}}(\text{Tutorials})$

Output - Selects tuples from Tutorials where topic = 'Database'.



Relational Algebra in DBMS

Projection(π):

The projection eliminates all attributes of the input relation but those mentioned in the projection list. The projection method defines a relation that contains a vertical subset of Relation. This helps to extract the values of specified attributes to eliminates duplicate values. (π) The symbol used to choose attributes from a relation. This operation helps you to keep specific columns from a relation and discards the other columns.

Syntax: $\pi_{A1, A2...}(r)$

where A1, A2 etc are attribute names(column names).

For example,

$\pi_{\text{Name, Age}}(\text{Student})$

Above statement will show us only the **Name** and **Age** columns for all the rows of data in **Student** table.



Relational Algebra in DBMS

Union Operation (U):

This operation is used to fetch data from two relations (tables) or temporary relation (result of another operation).

For this operation to work, the relations (tables) specified should have same number of attributes (columns) and same attribute domain. Also the duplicate tuples are automatically eliminated from the result.

Syntax: $A \cup B$

where A and B are relations.

For example, if we have two tables **RegularClass** and **ExtraClass**, both have a column **student** to save name of student, then,

$\Pi_{\text{Student}}(\text{RegularClass}) \cup \Pi_{\text{Student}}(\text{ExtraClass})$

Above operation will give us name of **Students** who are attending both regular classes and extra classes, eliminating repetition.



Relational Algebra in DBMS

Set Difference (-):

This operation is used to find data present in one relation and not present in the second relation. This operation is also applicable on two relations, just like Union operation.

Syntax: $A - B$

where A and B are relations.

For example, if we want to find name of students who attend the regular class but not the extra class, then, we can use the below operation:

$\Pi_{\text{Student}}(\text{RegularClass}) - \Pi_{\text{Student}}(\text{ExtraClass})$



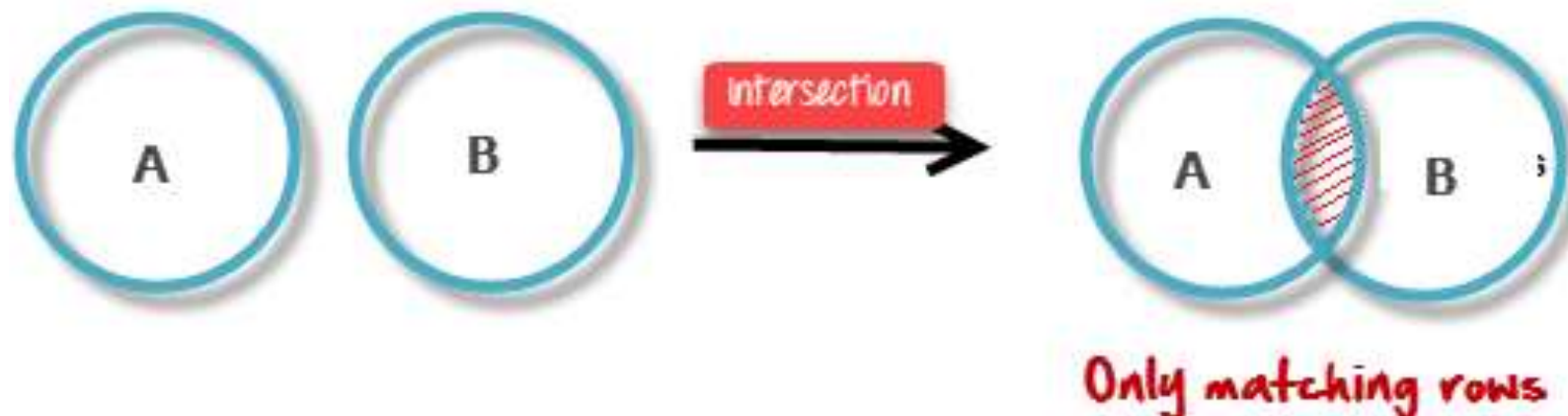
Relational Algebra in DBMS

Intersection:

An intersection is defined by the symbol \cap

$A \cap B$

Defines a relation consisting of a set of all tuple that are in both A and B.
However, A and B must be union-compatible.



Relational Algebra in DBMS

Cartesian product(X):

This type of operation is helpful to merge columns from two relations. It is also known as a cross product. Generally, a Cartesian product is never a meaningful operation when it performs alone. However, it becomes meaningful when it is followed by other operations.

Syntax: $A \times B$

Example – Cartesian product

$\sigma_{\text{column 2} = '1'} (A \times B)$

Output – The above example shows all rows from relation A and B whose column 2 has value 1



Relational Algebra in DBMS

Rename Operation (ρ):

This operation is used to rename the output relation for any query operation which returns result like Select, Project etc. Or to simply rename a relation(table)

Syntax: $\rho(\text{RelationNew}, \text{RelationOld})$



Relational Algebra in DBMS

Join Operations:

Join operation is essentially a cartesian product followed by a selection criteria.

Join operation denoted by \bowtie .

JOIN operation also allows joining variously related tuples from different relations.



Relational Algebra in DBMS

Types of JOIN:

Various forms of join operation are:

Inner Joins:

In an inner join, only those tuples that satisfy the matching criteria are included, while the rest are excluded.

- **Theta join**
- **EQUI join**
- **Natural join**

Outer Join:

In an outer join, along with tuples that satisfy the matching criteria, we also include some or all tuples that do not match the criteria.

- **Left Outer Join**
- **Right Outer Join**
- **Full Outer Join**



Relational Algebra in DBMS

Theta join(θ) :

They have tuples from different relations if and only if they satisfy the theta condition, here the comparison operators ($\leq, \geq, <, >, =, ^\top$) come into picture.

Let us consider simple example to understand in a much better way, suppose we want to buy a mobile and a laptop, based on our budget we have thought of buying both such that mobile price should be less than that of laptop.



Relational Algebra in DBMS

MOBILE

MODEL	PRICE
Asus	10k
Samsung	20k
<u>Iphone</u>	50k

LAPTOP

MODEL	PRICE
Acer	20k
HP	35k
Apple	80k

After JOIN



Asus	Acer
Asus	HP
Asus	Apple
Samsung	HP
Samsung	Apple
<u>Iphone</u>	Apple



Relational Algebra in DBMS

Equi Join:

It is also known as an inner join. It is the most common join. It is based on matched data as per the equality condition. The equi join uses the comparison operator(=). When a theta join uses only equivalence condition, it becomes a equi join.

For example:

$A \bowtie_{A.\text{column 2} = B.\text{column 2}} (B)$

$A \bowtie_{A.\text{column 2} = B.\text{column 2}} (B)$	
column 1	column 2
1	1



Relational Algebra in DBMS

Natural Join :

It does not utilize any of the comparison operator. Here the condition is that the attributes should have same name and domain. There has to be at least one common attribute between two relations. It forms the cartesian product of two arguments, performs selection forming equality on those attributes that appear in both relations and eliminates the duplicate attributes.

Consider the example, where two tables namely employment table and department table have been shown.



Relational Algebra in DBMS

EMPLOYMENT

NAME	EMPID	DPT_NAME
A	11	Sales
B	12	Finance
C	13	Finance

DEPARTMENT

DPT_NAME	MANAGER
Finance	M1
Sales	M2

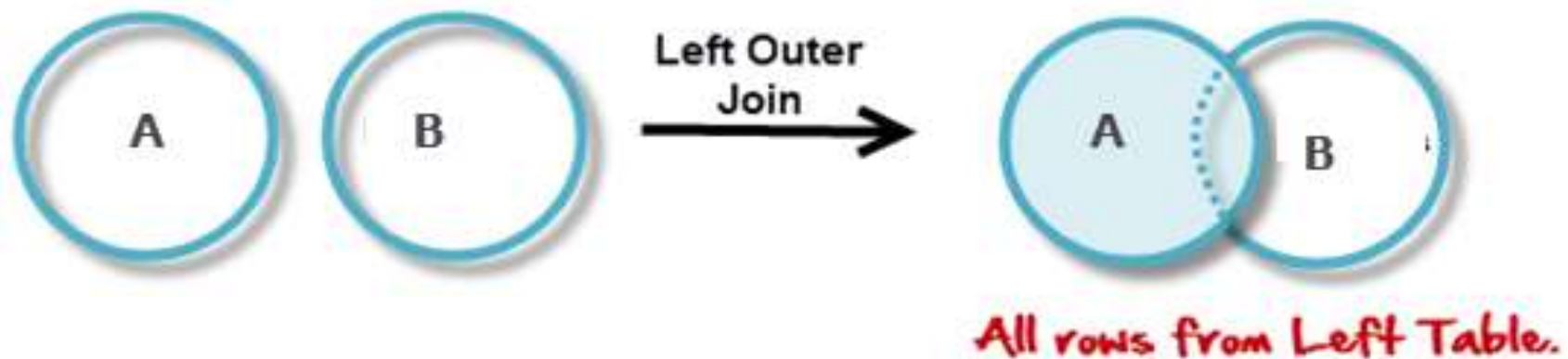
After JOIN 

Name	EMPID	DPT_NAME	MANAGER
A	11	Sales	M2
B	12	Finance	M1
C	13	Finance	M1

Relational Algebra in DBMS

Left Outer Join(A B):

In the left outer join, operation allows keeping all tuple in the left relation. However, if there is no matching tuple is found in right relation, then the attributes of right relation in the join result are filled with null values.



Relational Algebra in DBMS

Right Outer Join: (A B):

In the right outer join, operation allows keeping all tuple in the right relation. However, if there is no matching tuple is found in the left relation, then the attributes of the left relation in the join result are filled with null values.

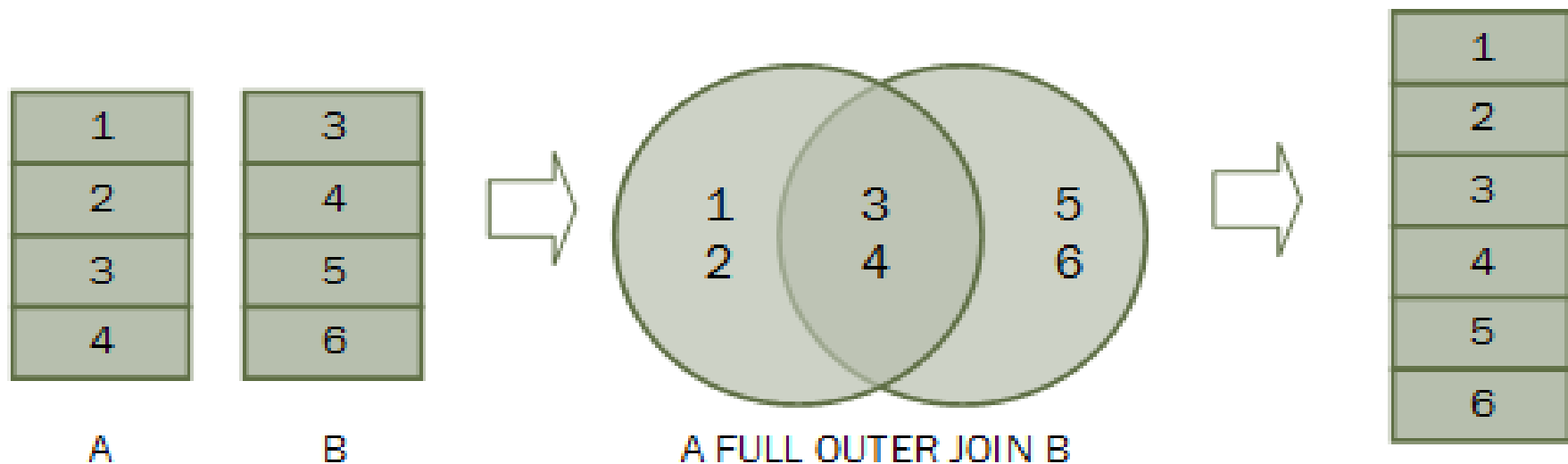


Relational Algebra in DBMS

Full Outer Join: (A B):

In a full outer join, all tuples from both relations are included in the result, irrespective of the matching condition.

A B



Relational Algebra in DBMS

Consider below example of two tables – country table that has 3 records and state table that has 4 records.

COUNTRY

COUNTRY_ID	COUNT_NME
1	India
2	Pakistan
4	Nepal

STATE

STATE_ID	COUNTRY_ID	STATE_NME
1	1	Karnataka
2	1	Tamil Nadu
3	2	Islamabad
4	NULL	Bangladesh



Relational Algebra in DBMS

LEFT OUTER JOIN

COUNTRY_ID	COUNT_NME	STATE_ID	COUNTY_ID	STATE_NME
1	India	1	1	Karnataka
1	India	2	1	Tamilnadu
2	Pakistan	3	2	Islamabad
4	Nepal	NULL	NULL	NULL



Relational Algebra in DBMS

RIGHT OUTER JOIN

COUNTRY_ID	COUNT_NME	STATE_ID	COUNTY_ID	STATE_NME
1	India	1	1	Karnataka
1	India	2	1	Tamil nadu
2	Pakistan	3	2	Islamabad
NULL	NULL	4	NULL	Bangladesh



Relational Algebra in DBMS

FULL OUTER JOIN

COUNTRY_ID	COUNT_NME	STATE_ID	COUNTY_ID	STATE_NME
1	India	1	1	Karnataka
1	India	2	1	Tamil nadu
2	Pakistan	3	2	Islamabad
4	Nepal	NULL	NULL	NULL
NULL	NULL	4	NULL	Bangladesh



THANK YOU

ANY QUERY???

