Introduction to Operating Systems and SQL for Data Science

Lecture 8 – Relational/bag algebra

Previous lecture

- Data models
- Relational data model
- Keys
- Data structures in RDBMS

Relational Algebra

- The relational algebra is a precise mathematical notation and set of rules for manipulating “relations”.
- Relational algebra is a collection of operations on relations. Each operation takes one or two relations as its operand(s) and produces another relation as its result.
- Relational algebra is the basis for some high-level languages and was initially defined by Codd.
- SQL is a more human-readable form of relational algebra.

“Core” Relational Algebra

- Relational algebra operators may be classified into two groups:

1. Traditional Set Operators:
   a. Union
   b. Difference
   c. Intersection
   d. Cartesian Product
2. Special Relational Operators:
   a. Selection
   b. Projection
   c. Join

Reichman University

Simple Special Relational Operators

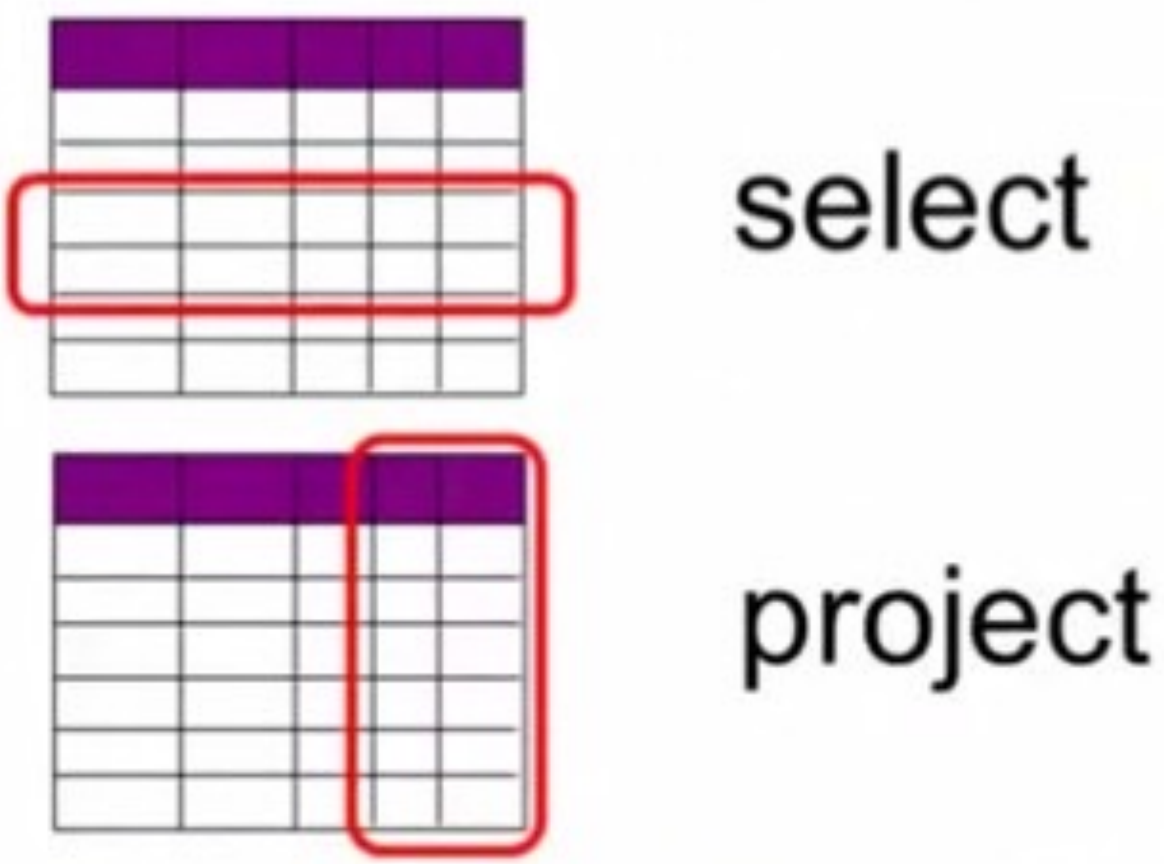
“Core” Relational Algebra

- Relational algebra operators may be classified into two groups:

1. Traditional Set Operators:
   a. Union
   b. Difference
   c. Intersection
   d. Cartesian Product
2. Special Relational Operators:
   a. Selection
   b. Projection
   c. Join

Reichman University

Selection & Projection



https://www.ibm.com/docs/en/informix-servers/14.10?topic=concepts-selection-projection

Reichman University

Selection

- $\sigma_{predicate}(R)$

  - Works on a single relation R and defines a relation that contains only those tuples (rows) of R that satisfy the specified condition (<u>predicate</u>).

Selection – an example

- List all staff with a salary greater than $10,000.
- $\sigma_{salary>10000}(Staff)$

staffNo	fName	lName	Position	Gender	salary
SL21	John	White	Assistant	M	12,000
SG37	Ann	Beech	Manager	F	30,000
SG14	David	Ford	Supervisor	M	18,000

Projection

- $\pi_{col1,\ldots,coln}(R)$

    - Works on a single relation R and defines a relation that contains a vertical subset of R, extracting the values of specified attributes and eliminating duplicates.

Reichman University

Projection – an example

- Produce a list of salaries for all staff, showing only staffNo, fName, lName, and salary details.
- $\pi_{staffNo,fName,lName,salary}(Staff)$

staffNo	fName	lName	salary
SL21	John	White	12,000
SG37	Ann	Beech	30,000
SG14	David	Ford	18,000

Set Operators

“Core” Relational Algebra

- Relational algebra operators may be classified into two groups:

1. Traditional Set Operators:
   a. Union
   b. Difference
   c. Intersection
   d. Cartesian Product
2. Special Relational Operators:
   a. Selection
   b. Projection
   c. Join

Union

- $R \cup S$
  - Union of two relations R and S defines a relation that contains all the tuples of R, or S, (or both) R, duplicate tuples being eliminated.
  - If R and S, have I and J tuples, respectively, union is obtained by concatenating them into one relation with a maximum of (I + J) tuples.

Union – an example

- List all cities where there is either a branch office or a property for rent.

$$\pi_{city}(Branch) \cup \pi_{city}(PropertyForRent)$$

City
London
Tel Aviv
New York

Reichman University

Set Difference

- $R - S$
  - Defines a relation consisting of the tuples that are in relation R, but not in S.

  - R and S must be union-compatible.

Set Difference – an example

- List all cities where there is a branch office but not a properties for rent.

$$\pi_{city}(Branch) - \pi_{city}(PropertyForRent)$$

City
London

Reichman University

Intersection

- $R \cap S$
  - Defines a relation consisting of the set of all tuples that are in both R and S.
  - R and S must be union-compatible.
  - Expressed using basic operations:

$$R \cap S = R - (R - S)$$

Intersection – an example

- List all cities where there is both a branch office and at least one property for rent.

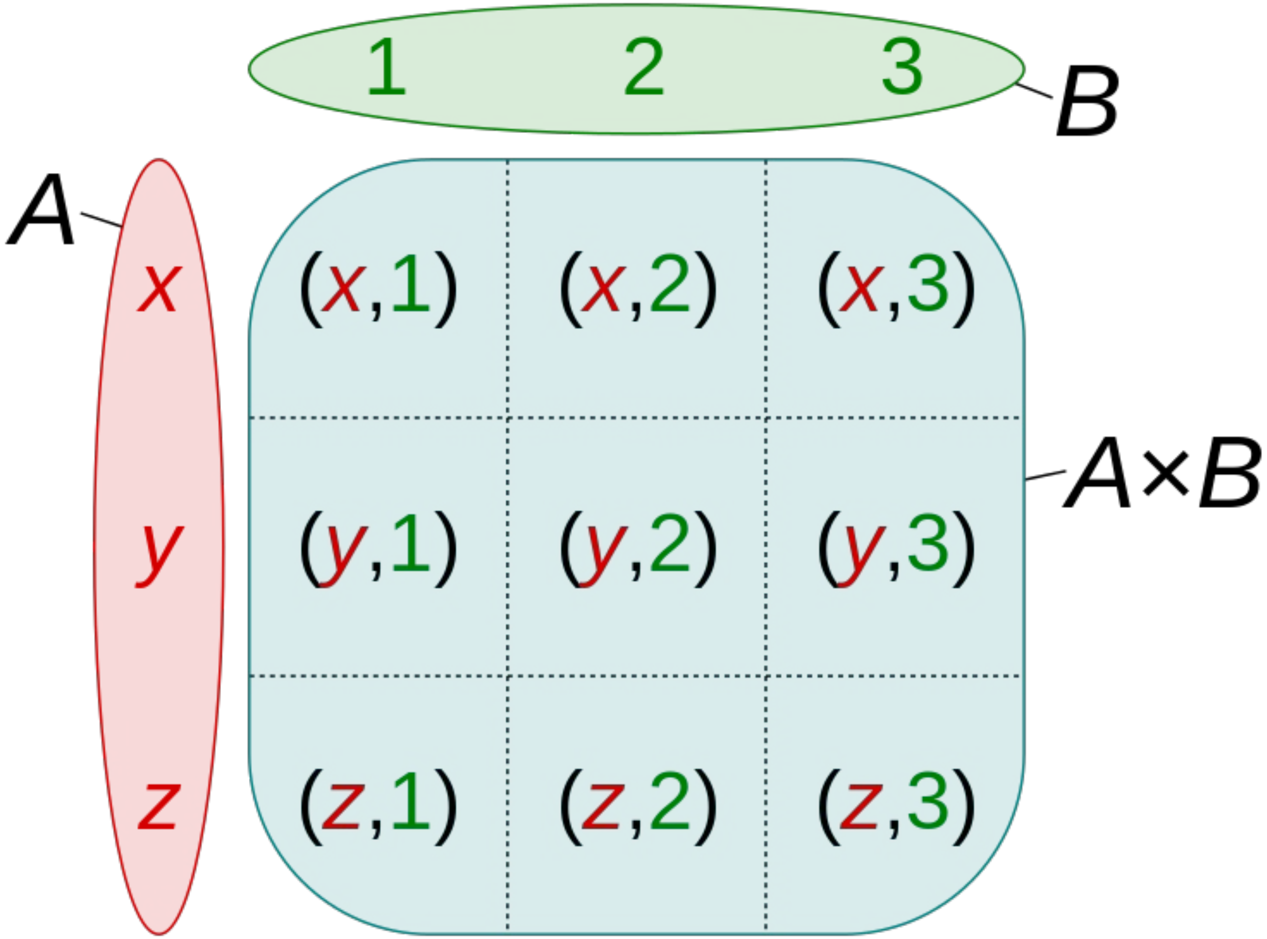
$$\pi_{city}(Branch) \cap \pi_{city}\ (PropertyForRent)$$

City
Tel Aviv
New York

Cartesian Product

- $R\ X\ S$
  - Defines a relation that is the concatenation of every tuple of relation R with every tuple of relation S.

Cartesian Product – an example



Reichman University

Cartesian Product – a fun example

We can create a standard 52-card deck using a cartesian product:

We define the following sets:

- Ranks - {A, K, Q, J, 10, 9, 8, 7, 6, 5, 4, 3, 2}.
- Suits – {♠, ♥, ♦, ♣}

Ranks × *Suits* returns a set of the form {(A, ♠), (A, ♥), (A, ♦), (A, ♣), (K, ♠), …, (3, ♣), (2, ♠), (2, ♥), (2, ♦), (2, ♣)}.

Reichman University

Intersection – an example

- List the names and comments of all clients who have viewed a property for rent.

$$\pi_{clientNo,fName,lName}(Client)\ X\ \pi_{clientNo,PropertyNo,comment}(Viewing)$$

Client.clientNo	fName	lName	Viewing.clientNo	propertyNo	comment
CR76	John	Kay	CR56	PA14	Too small
CR76	John	Kay	CR76	PA14	Too remote
C56	Aline	Stewart	CR56	PA14	Too small
C56	Aline	Stewart	CR76	PA14	Too remote

Cartesian Product and Selection – an example

- Use selection operation to extract those tuples where Client.clientNo = Viewing.clientNo.

$$\sigma_{Client.clientNo=Viewing.clientNo}(\pi_{clientNo,fName,lName}(Client)\ X\ \pi_{clientNo,PropertyNo,comment}(Viewing))$$

Client.clientNo	fName	lName	Viewing.clientNo	propertyNo	comment
CR76	John	Kay	CR76	PA14	Too remote
C56	Aline	Stewart	CR56	PA14	Too small

- Cartesian product and Selection can be reduced to a single operation called a **Join**.

Reichman University

Special Relational Operators

"Core" Relational Algebra

- Relational algebra operators may be classified into two groups:

1. Traditional Set Operators:
    a. Union
    b. Difference
    c. Intersection
    d. Cartesian Product
2. Special Relational Operators:
    a. Selection
    b. Projection
    c. Join

Reichman University

Join operations

- Join is a derivative of Cartesian product.
- One of the most difficult operations to implement efficiently in an RDBMS and one reason why RDBMSs have intrinsic performance problems.
- One of the most common operations for Data Scientists.

Join operations

- Various forms of join operation:
  - Theta join (and Equijoin)
  - Natural join
  - Outer join (left, right and full)

Theta Join (θ-join)

- Can rewrite Theta join using basic Selection and Cartesian product operations.

$$R \bowtie_F S = \sigma_F(RXS)$$

- Degree of a Theta join is sum of degrees of the operand relations R and S. If predicate F contains only equality (=), the term Equijoin is used.

Equijoin – an Example

- List the names and comments of all clients who have viewed a property for rent.
- $\pi_{clientNo,fName,lName}(Client)$

$\bowtie_{Clients.clientNo=Viewing.clientNo}$

$\pi_{clientNo,propertyNo,comment}(Viewing)$

Client.clientNo	fName	lName	Viewing.clientNo	propertyNo	comment
CR76	John	Kay	CR76	PA14	Too small
CR56	Aline	Stewart	CR56	PA14	Too remote
CR56	Aline	Stewart	CR56	PG4	
CR56	Aline	Stewart	CR56	PG36	

Reichman
niversity

Natural Join

- $R \bowtie S$

- The natural join is defined by:

Cartesian Product, Selection, Projection

- Another Definition - An Equijoin of the two relations R and S over all common attributes x. One occurrence of each common attribute is eliminated from the result

Natural Join – an example

- List the names and comments of all clients who have viewed a property for rent.

$$\pi_{clientNo,fName,lName}(Client) \bowtie \pi_{clientNo,propertyNo,comment}(Viewing)$$

You don't need to specify the operand

ClientNo	fName	lName	propertyNo	comment
CR76	John	Kay	PA14	Too small
CR56	Aline	Stewart	PA14	Too remote
CR56	Aline	Stewart	PG4	
CR56	Aline	Stewart	PG36	

Reichman University

Outer Join

- To display rows in the results that do not have matching values in the join column, use outer join.

- R ⟕ S

(Left) outer join in which tuples from R that do not have matching values in common columns of S are also included in result relation.

Outer Join – an example

- Produce a status report on property viewings.
- $\pi_{propertyNo,city}(PropertyForRent) ⟕ \pi_{propertyNo,clientNo,comment}(Viewing)$

propertyNo	city	clientNo	comment
PA14	London	CR76	Too small
PA14	London	CR76	No dining room
PL94	Manchester	null	null
PL93	Liverpool	null	null

Reichman University

Independent operands

- From all the operands we saw only $\sigma, \pi, \cup, -, X$ are independent, which means that we can’t express them by combination of other operands.

- Other operands are dependent, which means we can express them by the above operands.

- Examples:
  - $R \cap S = R - (R - S)$
  - $R \bowtie_c S = \sigma_c(R \; X \; S)$

Aggregate functions & grouping

Aggregate functions

- Aggregate functions refer ti group of values in the tables. For example, compute the salary of all employees, or average salary, or min, or max, or count the number of employees. (Sum, average, max, min, count)

- We can group certain record by a specific value and than apply an aggregate function. For example, calculate the average salary of the R&D department.

Reichman University

Aggregate functions

- The general form of these functions are denoted using f:

- <grouping attributes> f <function list> (R)

- Where <grouping attributes> are the list of attributes of the relation R, and <function list> is the list of the following tuples: <function><attribute>.

Aggregate functions – an example

- Show the number of the department and for each department the number of employees and the average salary.

- $\text{DepartmentNo}\ f\ \underbrace{\text{Count}_{\text{SSN}}, \text{average}_{\text{SALARY}}}_{\text{Function list}}\ (\text{Employee})$

Grouping attribute

$\text{Count}_{\text{SSN}}$

Function

attribute

Reichman University