Relational Algebra & MapReduce

Overview

- Introduction to simplified relational algebra in MapReduce

- Useful introductory to Apache Pig & HBase

Relational Algebra

- In traditional DBMS, queries involve retrieval of small amount of data

- Review of the terminology
  - a relation is a table
  - Attributes are the column headers of the table
  - A set of attributes of a relation is called schema
  - Example: R (A1, A2, A3) indicates a relation called R whose attributes are A1, A2, and A3

Example

StudentId	Name	CourseId
S1	Anne	C1

- Attribute names are StudentId, Name, CourseId

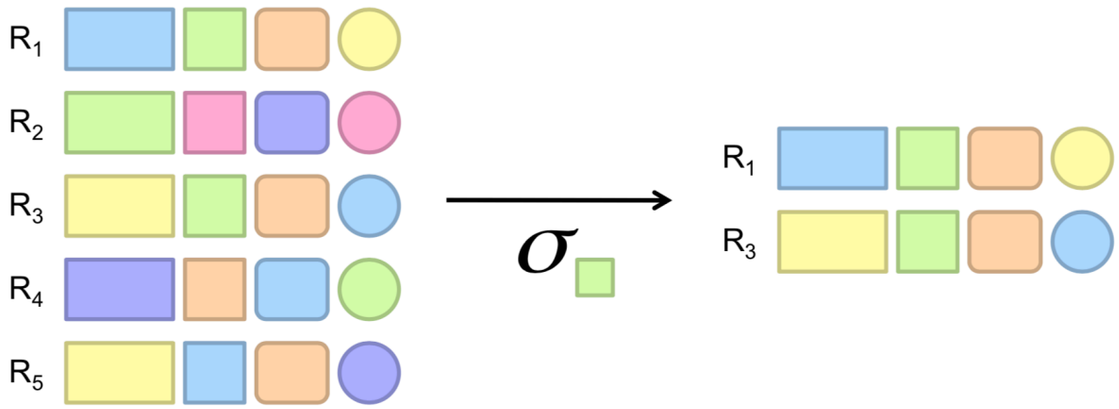
- Attribute values: S1, Anne, C1 (called tuple or row)

Relational Algebra Operators

- Relational Algebra
  - is a procedural query language
    - that takes relation as input, and generate relation as output
- Fundamental operations are:
  - select, projection, union, different, cartesian product, and rename

Select Operation

- Selects tuples that satisfies given condition (predicate)
  - picking certain rows
- Notation $\sigma_p(r)$
  - where σ stands for the selection
  - p stands for the logic formula.
    - it might use logic connectors; and, or, not
    - and relation operators; =, ≠, ≥, < , >, ≤.
  - r stands for the relation

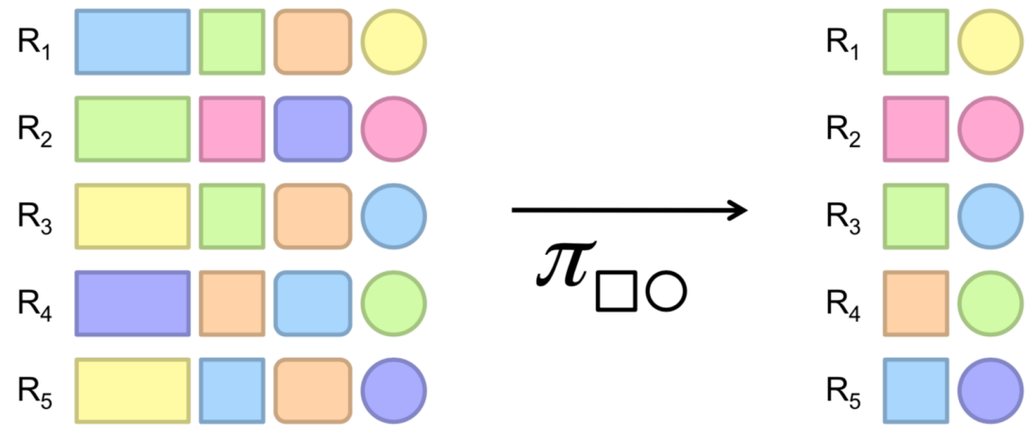


Selection in MapReduce

- In mapper:
    - process each tuple
        - emit (output) tuple that satisfies the conditions as key-value (tuple, tuple)
            - output is not a relation
- No need for reducers (only if to collect or do further processing)

Projection Operation

- Projects columns that satisfy given predicates
  - picking certain columns
- Notation: $\prod_{A1, A2}$ (r)
  - where:
    - A1, A2, are the attribute names of relation r



Projection in MapReduce

- In mapper:
  - read tuples
  - just emit given attributes
- Reducer are not needed
  - unless we want to collect results from mappers in fewer files
    - Instead of having a lot of files (output from mappers), we can have one reducer which reads all mappers output and put them in one file
      - output the same input

Group by - Aggregation

- Aggregation functions:
  - AVG, MAX, MIN, SUM, COUNT
- Example:
  - table visits contains URLs visited by users and time spent per url
  - we want to know average time spent per url
  - SQL query:

SELECT url,AVG(time) from visits GROUP by url

visits relation

url	user	time
www.ptuk.edu.ps	Ali	30

Group by in MapReduce

- Mappers:
  - read tuples
  - emit time, keyed by url
- grouping values by keys is handled automatically by the framework
- Reducer(s)
  - compute the average, min, max – depends on what you want to do
  - In case of average: Iterate over the values (time), sum and count, compute average

Relational Join