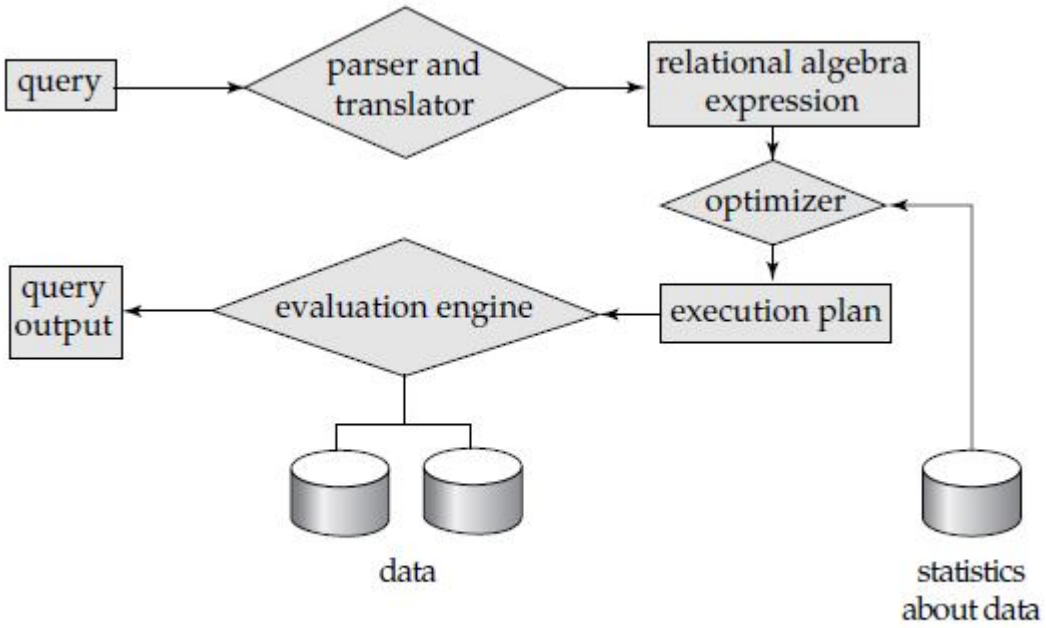
# Lecture Slot - 17th Oct

Parag Agrawal 09010130

Anurag Anshu 09010110

# Query Evaluation




Steps in query processing.

## Operations considered

σ , π , Join , ×

Let we have two tables R and S with number of pages M and N, respectively.

## Selection operator

Unary Operator

Given: a relation R

- **No index and data not sorted**
  Scan the entire table to get qualifying tuples
  Cost = O(M)
- **No index and data sorted**

  Binary search for retrieving the first qualifying tuple and then scan sequentially.

  Cost = $O(\log_2 M)$
- **$B^+$ tree index**

  Same as equality search

  Cost = $\log_F(1.5M)$ + no. of matching pages (if clustered index)

  Cost = $\log_F(0.15M)$ + no. of matching records (if unclustered index)
- **Hash index**

  Same as equality search

  Cost = constant I/O's

## Projection operator

Unary Operator

Given: a relation R

- **Projection based on Sorting**
  - Scan R and produce tuples having only the specified attributes (in the projection query).
  - Sort this set of tuples using all the above attributes as the key.
  - Scan these sorted tuples eliminating the duplicates.

  Cost = O(M logM)

- **Projection based on Hashing**

  Two phases

  1. Partitioning
  2. Duplicate Elimination

  Cost = O(M)

## Join operator

Binary Operator

Given: two relations R and S

- **No Memory Utilized**

  For each page of R we scan each page of S and get the qualifying tuples.

  Cost = O(M + M.N)

- **Memory utilized**

  1. We have **enough memory to hold one of the relations** say R

     - Move R into the memory.
     - For each tuple of S scan R and get the matching tuples.

     Cost = O(M + N)

  2. **Not enough memory to store any of the relations**

     - Break R into blocks that can fit in the available memory pages.
     - Scan all of S for each block of R.

       Let B = Number of available memory pages

     Cost = M + GIF(M/B)*N

     where GIF stands for Greatest integer Function

     Two sub cases

a) **Number of Memory Pages Constant**

Cost = O(M.N) (in above formula of 'Cost' set B = constant)

b) **Number of Memory Pages Fraction of M (or N)**

Cost = O(M + N) (in above formula of 'Cost' set B = fraction of M)

**3. Sort-Merge Join**

- Sort both the relations on the join attributes
- Merge the two sorted relations to get the matching tuples

(Note: Enumerating the cross product is avoided here)

Two sub cases

a) **Number of Memory Pages Constant**

Cost = O(MlogM) + O(NlogN) + O(M + N)
(for sorting) (for merging)

b) **Number of Memory Pages Fraction of M (or N)**

Here *sorting is done in just two passes* for each of R and S
Cost = O(M + N)

## Cross product

Binary operator

R × S = R Join S (with no Join condition)

Analysis similar to the analysis of Join