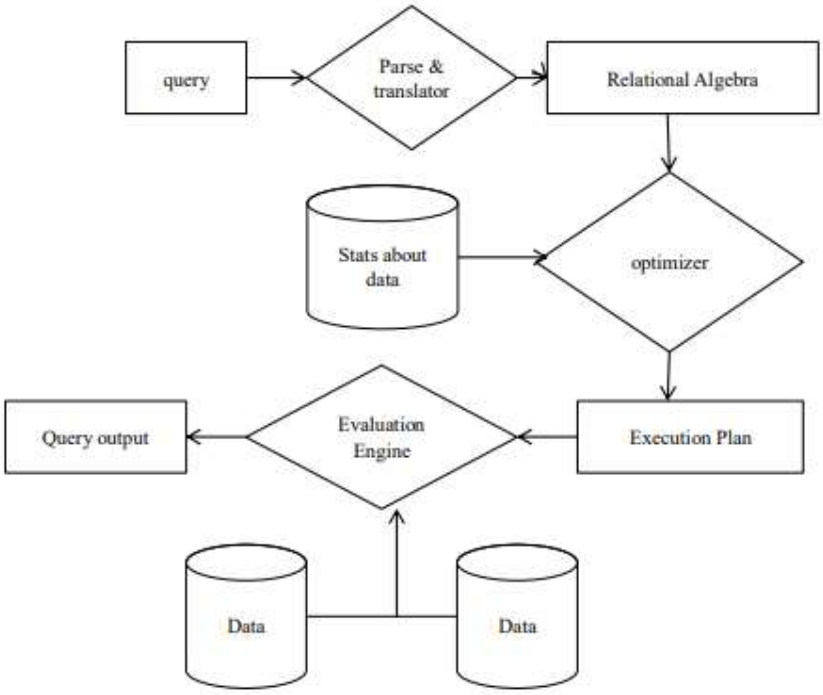
Query Optimization

Query Processing: Query processing refers to activities including translation of high level languages (HLL) queries into operations at physical file level, query optimization transformations, and actual evaluation of queries.

Steps in Query Processing:



Function of query parser is parsing and translation of HLL query into its immediate form relation algebraic expression.

A parse tree of the query is constructed and then translated in to relational algebraic expression.

Example: consider the following SQL query:

SELECT s.name

FROM reserves R, sailors S

Where R.sid=S.sid

AND R.bid=100 And S.rating>5

Its corresponding relational algebraic expression

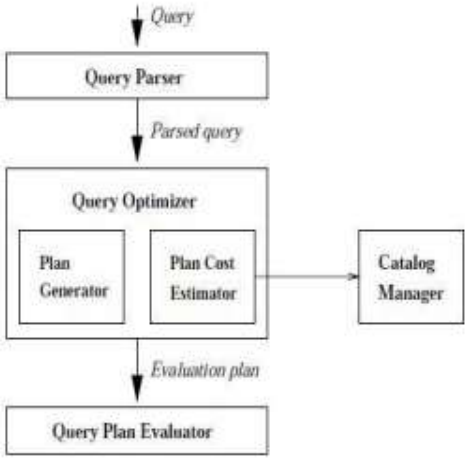
$$\pi_{sname}(\sigma_{bid=100 \wedge rating>5}(Reserves \bowtie_{sid=sid} Sailors))$$

Query Optimization :

Query optimization is the process of selecting an efficient execution plan for evaluating the query.

After parsing of the query, parsed query is passed to query optimizer, which generates different execution plans to evaluate parsed query and select the plan with least estimated cost.

Catalog manager helps optimizer to choose best plan to execute query generating cost of each plan.



Query optimization is used for accessing the database in an efficient manner. It is an art of obtaining desired information in a predictable, reliable and timely manner. Formally defines query optimization as a process of transforming a query into an equivalent form which can be evaluated more efficiently. The essence of query optimization is to find an execution plan that minimizes time needed to evaluate a query. To achieve this optimization goal, we need to accomplish two main tasks. First one is to find out the best plan and the second one is to reduce the time involved in executing the query plan.

Three different phases during the query processing in DBMS which are as follows: • Parsing and translation • Optimization • Evaluation. Usually, user queries are submitted to DBMS as SQL queries. During the parsing and translation phase, the given query is translated into its internal form. In generating the internal form of the query, the parser checks the syntax of the user's query, verifies that the relation names appearing in the query are names of the relations in the database and so on. The system constructs a parse tree representation of the query, which it then translates into a relational algebra expression.

For example let us consider the following SQL query:-

Select Sno from Student where Sno='101'

This query is then translated into either of the following relational algebra expressions as follows:-

$\sigma_{Sno = '101'}(\pi_{Sno}(\text{Student}))$

$\pi_{Sno}(\sigma_{Sno='101'}(\text{Student}))$

After parsing and translation into relational algebra expression, the query is then transformed into a form which is usually query tree or graph that can be handled by the optimization engine. Query representation During the optimization phase, the optimization engine performs various analyses on the query data. It applies various rules to the internal data structures of the query to transform these structures into equivalent and efficient representation. It then generates valid evaluation plans based upon the rules applied. From the generated evaluation plans, the best evaluation plan to be executed is determined and passed onto the query execution engine. The final phase in processing a query is the evaluation phase. During the evaluation phase, the best evaluation plan generated by the optimization engine is selected and then executed.

The next step is an optimization step that transforms the initial algebraic query using relational algebra transformation into other algebraic queries until the best one is found. A query execution plan is then founded which represented as a query tree includes information about the access method available for each relation as well as the algorithms used in computing the relational operations in the tree. The next step is called code generator, where we generate code for the selected query execution plan. This code is then executed by the run time database processor to produce the query result. The run time database processor has the task of running the query code, whether in compiled or interpreted mode, to produce the query result. If a run time error results, an error message is generated by the run time database processor.

Distributed database system vs. Centralized database system

Distributed database system:

Distributed database management system is basically a set of multiple and logical interrelated database which is distributed over the network. It includes single database which is further divided into sub fragments. Each fragment is integrated with each other and is controlled by individual database. It provides a mechanism that helps the users in distributing the data transparently. Distributed database management system is mostly used in warehouse to access and process the database of the clients at single time.

Centralized database management system:

Centralized data base is another type of database system which is located, maintained and stored in a single location such as mainframe computer. Data stored in the centralized DBMS is distributed across the network computers. It includes set of records which can easily be accessed from any location by using internet connection such as WAN and LAN. Centralized database system is commonly used in the organizations such as banks, schools, colleges etc to manage all their data in an appropriate manner.

Difference:

Centralized DBMS	***Distributed DBMS***
Data is stored only on one site	*Data is stored on different sites*
Data stored in single computer can be used by multiple users	*Data is stored over different sites which are connected with each other.*
If centralized system fails, then the entire system is halted.	*If one of the system fails, then user can access the data from other sites.*
Centralized DBMS is less reliable and reactive	*Distributed DBMS is more reliable and reactive*
Centralized DBMS is less sophisticated	*Distributed DBMS is more sophisticated*

Advantages and disadvantages of Centralized DBMS:

Advantages:

- With the help of centralized database management system, organizations an easily communicate with each other in less time. This approach basically allows the team members of an organization to work on cross-functional projects. It becomes easy for the team members to analyse the data and complete the tasks with good quality.
- By using centralized database system, individuals and teams can easily share their ideas with each other. It becomes easy for the organization to co-ordinate their work with the team members and achieve their business goals.
- Centralized database system also provides high level of security.
- Most of the organizations prefer to use centralized database to reduce the conflicts within the organization. Sharing the information with each other leads to a happier working environment.

Disadvantages of Centralized database:

- Organizations may face issues while using centralized database due to heavy workload requirements.
- While using centralized database system, organizations may have to spend more money to manage and store the data.

Advantages and disadvantages of Distributed database system:

Advantages:

- Distributed database system reflects the organizational structure in an appropriate manner. It becomes easy to access the organizational data in an effective manner.

- By using one particular site, users can access the data stored at different sites easily and effectively.
- Distributed database system also improves the availability, reliability and performance of the organization. Failure of one site allow the users to access the information from other sites easily.
- Distributed database system helps the organizations to handle their growth expansion. Increase in the database size can easily be managed by using distributed database system.

Disadvantage:

- Distributed database system increases the complexity and cost of the organization. It becomes difficult for the organization to maintain and manage the local database management system due to which organizations may face difficulty to establish a network between the sites.
- In distributed database system, it becomes difficult for the organizations to control the replicate data.
- While using distributed database system, organizations may face difficulty in maintaining the database integrity. Organizations have to spend more communication and processing cost to enforce the integrity constraints.
- While using distributed database system, organizations cannot use static SQL.