Assignment (Video- 28 to 30): Database System Impl. (COP6726)

Name: Vikas Chaubey**, UFID:** 3511 5826**, Email:** vikas.chaubey@ufl.edu

1) **Linear Scans:** In the linear scan mode, the database is read linearly that means each row of database is accessed and then read sequentially in serial order, while iterating through the records the columns are validated against the condition given in the query. This is also called as full table scan mode. Linear scans are the slowest way of reading the database because it involves heavy number of input and output disk operations, these operations are costly because they involve costly disk to memory transfers of data. In a database table linear scans take place when the queries are not inexed.in that case even though very few records have to be selected, all the records are iterated and searched for the matching results hence linear scans are very costly operations. Linear scan has some pros as well since the records are read sequentially it's very easy to predict the cost as every time database system needs to scan full table row by row.

2) **Column vs Vectorized Processing:** C_Store is a database management system which stores data in columns instead of rows as in traditional database management systems. This database system is optimized for reading rather than writing. In various cases when large queries are operated on data in the result set only few columns are required. When the query is executed in a row bases relational database system then a large number of rows or records are retrieved and then required columns are extracted from the result set, in case of column based database the records themselves are saved in columns of rows not rows , hence when these complex queries are executed in C_Store database models the query execution is really fast and efficient. When we have a large dataset, with many rows and columns, only retrieving the columns that we need to execute our analytics query is substantially more efficient than retrieving all the records in the database. Vectorized code makes efficient utilization of CPU cache. For example, in a row of data with 10 columns and a query plan that needs to operate only on a single column. In a row-oriented query processing model, nine columns would occupy cache unnecessarily, limiting the number of values that can fit into cache. With column-oriented processing, only the values from particular column of interest would be read into cache, allowing for many more values to be processed together and efficient usage of CPU-memory bandwidth. A column-based database is always preferred for analytics-based applications. This database system is optimized for reading. In case of analytics most of the queries are complex and heavy, in case of column-based database the records themselves are saved in columns not rows, hence when these complex queries are executed in column-based database models the query execution is really fast and efficient. When we have a large dataset, with many rows and columns, only retrieving the columns that we need to execute our analytics query is substantially more efficient than retrieving all the records in the database.

3) **Concurrent Join:** Concurrent join, joins the beginning of the result obtained by one thread to the end of results obtained by the other thread which is running concurrently in database as part of query. The goal of concurrency is to perform a task fast and make the best efficient use of computing resources. when multiple threads are working to perform a task or execute a query then concurrent join is used to join the results obtained by multiple executing threads to prepare query output.

4) **User Defined Aggregates:** User defined aggregate functions are aggregate functions which are defined by user. These aggregate functions are user programmable routines which operate on a set of rows or records and return an aggregate value. Generally, Query languages provide predefined aggregate functions such as sum, max, min etc. which act on number of records and return aggregate value, but if the user has specific requirement to aggregate selected data in specific way then in that case user defined aggregation could be used. There are certain classes which are required for creation and registration of user defined aggregate functions and routines. User programs and creates those routines and then these routines have to registered in order to be used in the programming languages.

5) **TOP-K GLA:** In recent years in various industry use cases required to develop effective techniques to perform adhoc search and retrieval in relational databases. The main popular method to handle the adhoc retrieval problem is to use TOP-K querying model. This model queries and retrieves the relevant information from the databases and then the results are ranked and then TOP -K results are returned. Various algorithms are used to implement TOP-K model. Such as basic top – K algorithms and FA and TA algorithms for Top K querying.