

## CS631 ASSIGNMENT – DAVID GOMEZ CAMARGO

### 1. BASIC TERM DEFINITIONS:

- **Data:** known facts that can be recorded and have an implicit meaning.
- **Database:** a collection of related data.
- **Database Management System (DBMS):** a software package/system to facilitate the creation and maintenance of a computerized database.
- **Database System:** the DBMS software together with the data itself. Sometimes, the applications are also included.
- **Database Catalog:** the system catalog consists of tables and views that describe the structure of the database.
- **Program-Data Independence:** refers to the capability of leaving data intact and accessible regardless of modifications to the database that contains the data.
- **User View:** external schema or subschema with part or all the contents of a database specified to facilitate a particular purpose or user activity.
- **Database Administrator (DBA):** performs all activities related to designing, maintaining, creating, and running efficient and cost-effective databases and environments.
- **End User:** those people whose jobs require access to the database for querying, updating, and generating reports.
- **Canned Transaction:** standard types of queries and updates which are frequently used by end-users to constantly query and update database. These are the transactions that are carefully programmed and tested in advance.

### 2. DIFFERENCE BETWEEN DATABASE SCHEMA AND DATABASE STATE:

A database schema defines the structure of the database, how the data will be logically represented and be stored, whereas database state is the representation of currently stored data.

The schema of the database provides a list of data types which will be associated with each field whereas the state of the database can be affected on account of inserting or updating information in the database. Which means that the schema of the database is not supposed to be changing but the state of the database can.

### 3. MAIN COMPONENTS OF DBMS:

- **Software:** the computer system consists of a set of programs, functions, and processes which are collectively called as software. The DBMS, which stands in between the user and the physical database, gives us an intuitive interface for storing, accessing, and updating data and is known as DBMS software.
- **Hardware:** the physical part of the computer which consists of a hard disk and other I/O devices.

- **Data:** it is the most important component in the DBMS. Data processing is the main task of DBMS. Data can be inserted, updated, deleted, and retrieved from the DBMS.
- **Data Access Language:** database language that is used to perform multiple operations in DBMS. Data definition language, data, data manipulation language, data control language, and transaction control language are database access languages.
- **Procedure:** instructions for utilizing a DBMS. This comprises steps for installing a DBMS, logging in and out of DBMS software, managing databases, reports, and creating backups.
- **User:** users can operate a database in various ways by controlling and managing the database. There are many types of users such as native users, sophisticated users, application users and specialized users.

#### 4. TYPES OF FACILITIES PROVIDED IN A MULTI-USER DBMS:

- **Data storage service:** data should be reliably stored in a multiuser database management system. In order to provide reliable facility to the users that are interrelated in that multi-user DBMS so that whenever needed the resources from the Client or the server then data can easily fetch intensely in multiuser DBMS.
- **Update service and retrieval service:** updating and retrieval service must be efficiently provided for the user and the server so that whenever needed any update or retrieval of any data from that multi-user DBMS can easily access those files.
- **User accessible catalogue support service for data communication:** Support Service Management makes user-initiated requests in a very friendly manner. The service of user catalogue is accessible in the portal for the end user and should provide various support services that are faced by many users.
- **Transaction service:** transaction is basically a program that includes various database operations interlinking with many databases with its retrieval, update, delete and insert operation. Only data retrieval must be used in the transaction facility as transaction should be only read only transaction.
- **Authorization service:** authorization service must be implemented so that only authorized users can access that database that are interrelated with each other.
- **Concurrently control service:** this service important as it allows to coordinate intensely various transactions that also provide Data integrity and control multi-user access of multi-user databases. When there is any operation implemented then it should not lose any data during operation manipulation in this service.
- **Data Independence service:** Data Independence service allows the user or the provider to change database schema at one level without requiring or needing that change at another higher level of that database schema. This service allows data to keep data separated from each other. In many systems it is very essential function when data has to be represented for computation and representation.
- **Recovery service from deadlock situations:** Recovery Process service must be applied in multi-user DBMS as to recover all data that has been lost in any deadlock situation and how to recover.

- **Utility service:** Utility service allow to perform operations that are associated with a particular operating system like copying the file and writing any script file.

## 5. DIFFERENCE BETWEEN LOGICAL AND PHYSICAL DATA INDEPENDENCE:

Logical data independence is the ability to modify the conceptual schema without having alteration in external schemas or application programs. Alterations in the conceptual schema may include addition or deletion of fresh entities, attributes or relationships and should be possible without having alteration to existing external schemas or having to rewrite application programs.

Physical data independence is the ability to modify the inner schema without having alteration to the conceptual schemas or application programs. Alteration in the internal schema might include using new storage devices, using different data structures, switching from one access method to another, using different file organizations or storage structures and modifying indexes.

## 6. DIFFERENCE BETWEEN EXTERNAL, INTERNAL, AND CONCEPTUAL SCHEMAS. RELATIONSHIP TO THE CONCEPT OF LOGICAL AND PHYSICAL DATA INDEPENDENCE:

The internal schema describes the physical storage structure of the database. It uses a physical data model and describes the complete details of data storage and access paths for the database.

The conceptual schema describes the structure of the whole database for a community of users. It doesn't include the details of physical storage structures and concentrates on describing entities, data types, relationships, user operations, and constraints.

The external schema describes the part of the database that a particular user group is interested in and hides the rest of the database from that user group.

Logical and physical data independences:

- External schemas provide logical data independence.
- Conceptual schemas provide physical data independence.
- The logical data independence is the capacity to change the conceptual schema without having to change external schemas or application programs.
- The physical data independence is the capacity to change the internal schema without having to change the conceptual schema. Hence, the external schemas need not be changed.
- Generally, physical data independence exists in most databases and file environment in which the exact location of data on disk, splitting, compression and so on are hidden from the user.

- The logical data independence is very hard to come by because it allows structural and constraint changes without affecting application programs.

## 7. DIFFERENCE BETWEEN THE TWO-TIER AND THREE-TIER CLIENT/SERVER ARCHITECTURES:

- **Two-Tier Database Architecture:** in two-tier, the application logic is either buried inside the User Interface on the client or within the database on the server (or both). With two-tier client/server architectures, the user system interface is usually located in the user's desktop environment and the database management services are usually in a server that is a more powerful machine that services many clients.
- **Three-Tier Database Architecture:** in three-tier, the application logic or process lives in the middle-tier, it is separated from the data and the user interface. Three-tier systems are more scalable, robust and flexible. In addition, they can integrate data from multiple sources. In the three-tier architecture, a middle tier was added between the user system interface client environment and the database management server environment. There are a variety of ways of implementing this middle tier, such as transaction processing monitors, message servers, or application servers.

## 8. JOB REQUIREMENTS:

- **DBA:** strong SQL and Data Analysis skills, working knowledge of SQL best practices (e.g., schema management), solid knowledge on reporting tools like PowerBI and prior experience towards working / leading a Data Operations engagement will be an added advantage.
- **Database developer:** Bachelor's Degree in computer science or equivalent experience required, working knowledge of ETL development, 2-5 years of MariaDB, MySQL, Oracle with development of database procedures and functions, working knowledge of Elasticsearch is a plus and good working knowledge of Linux bash scripting.
- **Database designer:** extensive experience with Oracle, complex SQL, and PL/SQL; excellent Relational database design experience and skills; proficient with data modeling tools, e.g., Power Designer or ERWIN; excellent analytical and problem-solving skills; prefer to have Azure knowledge and experience.
- **Data Analyst:** Bachelor's degree in computer science, statistics or mathematics; intermediate knowledge of Python (mandatory), SQL and R programming (optional); experience working in data collection, analysis and quality control. Strong understanding of Text Processing & Analytics; Ability to understand how to parse & query text using tools like Elasticsearch; Working knowledge of coding simple scripts with Python specific experience handling HTTP request/response, JSON and converting and extracting data from unstructured documents; Experience writing complex SQL queries for MySQL database, like aggregate and pivot over multiple fields; Experience creating Tableau dashboards and visualizations; intermediate excel skills, ability to perform SUM, IF, VLOOKUP, DATE and TEXT functions.

- **Business Analyst:** BA or BS in Business, Accounting or related field required; MBA preferred; A minimum of 5 years of Business Analyst experience; exceptional communication skills and professionalism; Strong analytical skills and problem-solving ability; Experience in technical writing (business cases, project charters, etc.); Experience in business process improvement efforts; Experience with SQL, Tableau, or related systems is a plus; Strong business acumen.