Question Paper Solutions
UNIT-I

1) Define the following terms: 20 Marks (Jun/July 2013)

i) Data base
Ans: A database is a collection of related data, where data means recorded facts. Computer- based repositories for data. It is logically coherent data to which some meaning is attached.
ii) Canned transaction:
Ans: These are the transactions that are carefully programmed and tested in advance. Ex bank teller, check account balances post withdrawals/deposits.
iii) Data model:
Ans: It is a collection of concepts that can be used to describe the concepts/logical structure of a database which provides the necessary means to achieve their abstraction.
iv)Meta data:
Ans: It is the data about data which is stored in system catalog which contains description of the structure of each file the type and the storage format of each file and the various constraints on the data.
v) Data base designer:
Ans: They are responsible for identifying the data to be stored and for choosing an appropriate way to organize it. They also define views for different categories of users.

2) Explain the characteristics of data base approach. 4 Marks (Jun /July2014/June/July 2015)

1. ***Self-Describing Nature of a Database System*** - it has a complete definition or description of the database structure and constraints. This definition is stored in the system **catalog**, which contains information such as the structure of each file, the type and storage format of each data item, and various constraints on the data. This information stored in the system catalog is called, **Meta-data** and it describes the structure of the primary database. This allows the DBMS software to work with different databases.
2. ***Insulation between Programs and Data and Data Abstraction*** - Called **program-data independence**. Allows changing data storage structures and operations without having to change the DBMS access programs. The structure of data files is stored in the DBMS catalog separately from the access programs.
3. ***Data Abstraction***: A data model is used to hide storage details and present the users with a conceptual view of the database.
4. ***Support of Multiple Views of the Data*** - Each users may see a different view of the database, which describes only the data of interest to that user.
5. ***sharing of Data and Multi-user*** *Transaction Processing* - the DBMS must include *concurrency control* software to ensure that the result of multi-user access is correct.

3) What are the responsibilities of a data base administrators? 4 Marks (Dec/Jan 2014, Jun / July2013)
Ans: A Data base administrator oversees and manages the data base system. Duties include authorizing users to access the data base coordinating/monitoring its use, acquiring H/W and S/w

for upgrade. DBA must have supporting staff. The DBA is responsible for problems such as breach of security or poor system response time.

4) Explain the typical components of a DBMS with a neat diagram. 8 Marks (Dec /Jan 2013, Dec/Jan 2014, Dec2014/Jan2015)

DBMS Component Modules

• A higher-level stored data manager module of the DBMS controls access to DBMS information that is stored on disk.

• The DDL compiler processes schema definitions, specified in the DDL, and stores descriptions of the schemas (meta-data) in the DBMS catalog.

• The run-time database processor handles database accesses at run time. The query compiler handles high-level queries that are entered interactively.

• The pre-compiler extracts DML commands from an application program written in a host programming language. These commands are sent to the DML compiler for compilation into object code for database access.

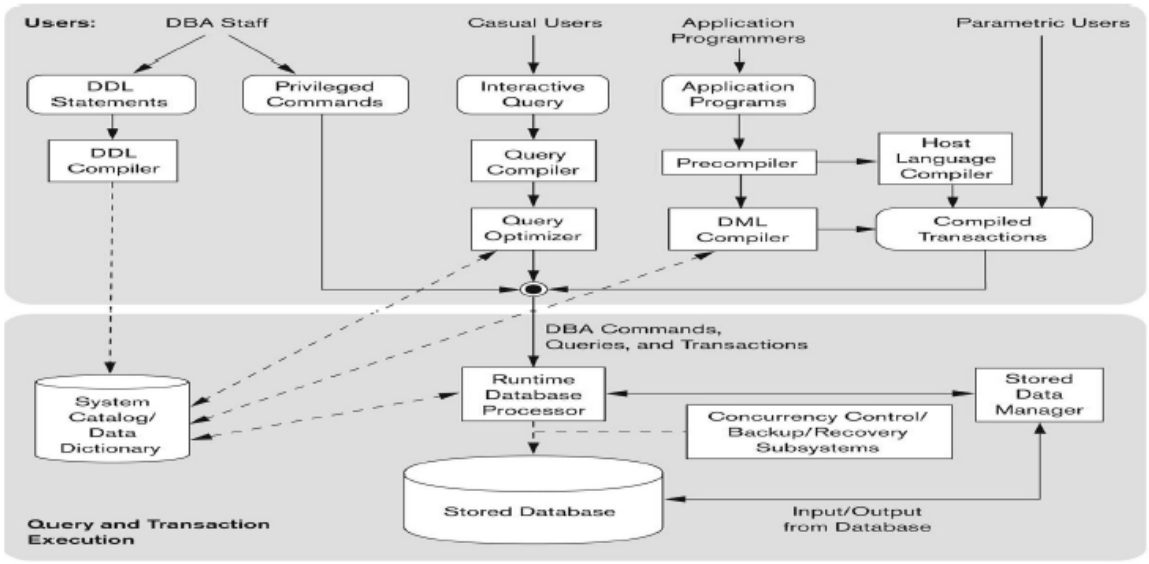


Figure 2.3
Component modules of a DBMS and their interactions.

5) Define and explain the following terms with an example for each. 8 Marks (Dec /Jan 2013, Jun / July2014)

i) Snapshot: The data in the database at a particular moment in time is called a database *state or snapshot.* A database state (also called *instances*) changes every time data is inserted, deleted, or modified

ii) Intension: The distinction between database schema and database state is that when we define a new database, we specify the database schema only to the DBMS The schema is sometimes called the intension.

iii) Extension: The distinction between database schema and database state is that when we define a new database, we specify the database schema only to the DBMS (the current state of the database is the *empty state* with no data). The database state is called an **extension** of the schema

6) What is meant by "persistent storage for program objects"? Explain. 4 Marks (Dec/Jan 2013)

Providing Persistent Storage for Program Objects and Data Structures - objects survive the termination of program execution and can later be retrieved the persistent storage of programs objects and data structures is an important function of database systems.

7) Briefly discuss the advantages of using the DBMS 8 Marks (Dec/Jan 2014)

1. **Controlling Redundancy:** Data redundancy (such as tends to occur in the "file processing" approach) leads to **wasted storage space**, **duplication of effort** (when multiple copies of a datum need to be updated), and a higher likelihood of the introduction of **inconsistency**. On the other hand, redundancy can be used to improve performance of queries. Indexes, for example, are entirely redundant, but help the DBMS in processing queries more quickly.
2. **Restricting Unauthorized Access:** A DBMS should provide a **security and authorization subsystem**, which is used for specifying restrictions on user accounts. Common kinds of restrictions are to allow read-only access (no updating), or access only to a subset of the data (e.g., recall the Bursar's and Registrar's office examples from above).
3. **Providing Persistent Storage for Program Objects:** Object-oriented database systems make it easier for complex runtime objects (e.g., lists, trees) to be saved in secondary storage so as to survive beyond program termination and to be retrievable at a later time.
4. **Providing Storage Structures for Efficient Query Processing:** The DBMS maintains indexes (typically in the form of trees and/or hash tables) that are utilized to improve the execution time of queries and updates. (The choice of which indexes to create and maintain is part of *physical database design and tuning* and is the responsibility of the DBA).The **query processing and optimization** module is responsible for choosing an efficient query execution plan for each query submitted to the system.
5. **Providing Backup and Recovery:** The subsystem having this responsibility ensures that recovery is possible in the case of a system crash during execution of one or more transactions.
6. **Providing Multiple User Interfaces:** For example, query languages for casual users, programming language interfaces for application programmers, forms and/or command codes for parametric users, menu-driven interfaces for stand-alone users.
7. **Representing Complex Relationships among Data:** A DBMS should have the capability to represent such relationships and to retrieve related data quickly.
8. **Enforcing Integrity Constraints:** Most database applications are such that the semantics (i.e., meaning) of the data require that it satisfy certain restrictions in order to make sense. Perhaps the most fundamental constraint on a data item is its data type, which specifies the universe of values from which its value may be drawn. (E.g., a Grade field could be defined to be of type Grade_Type, which, say, we have defined as including precisely the values in the set { "A", "A-", "B+", ..., "F" }.

Another kind of constraint is *referential integrity*, which says that if the database includes an entity that refers to another one, the latter entity must exist in the database. For example, if (R56547,

CIL102) is a tuple in the Enrolled_In relation, indicating that a student with ID R56547 is taking a course with ID CIL102, there *must be* a tuple in the Student relation corresponding to a student with that ID.

9. **Permitting Inferencing and Actions Via Rules:** In a **deductive** database system, one may specify *declarative* rules that allow the database to infer new data! E.g., Figure out which students are on academic probation. Such capabilities would take the place of application programs that would be used to ascertain such information otherwise. **Active** database systems go one step further by allowing "active rules" that can be used to initiate actions automatically.

8) Discuss the main Characteristics of the database approach. How does it differ from Traditional file systems? ` 8 Marks (Jun / July 2013, Jun / July2014)

Characteristics of the Database Approach are *Self-Describing Nature of a Database System* - it has a complete definition or description of the database structure and constraints. This definition is stored in the system **catalog**, which contains information such as the structure of each file, the type and storage format of each data item, and various constraints on the data. This information stored in the system catalog is called, **Metadata** and it describes the structure of the primary database. This allows the DBMS software to work with different databases.

Insulation between Programs and Data and Data Abstraction - Called **program-data independence**. Allows changing data storage structures and operations without having to change the DBMS access programs. The structure of data files is stored in the DBMS catalog separately from the access programs.

Data Abstraction: A data model is used to hide storage details and present the users with a conceptual view of the database.

Support of Multiple Views of the Data - Each users may see a different view of the database, which describes only the data of interest to that user.

Sharing of Data and Multiuser *Transaction Processing* - the DBMS must include *concurrency control* software to ensure that the result of multiuser access is correct.

9) Explain the difference between Logical and physical data Independence? 8 Marks (Jun / July 2013)

One important characteristic of the database approach was the insulation of programs and data. We can define two types of data independence:

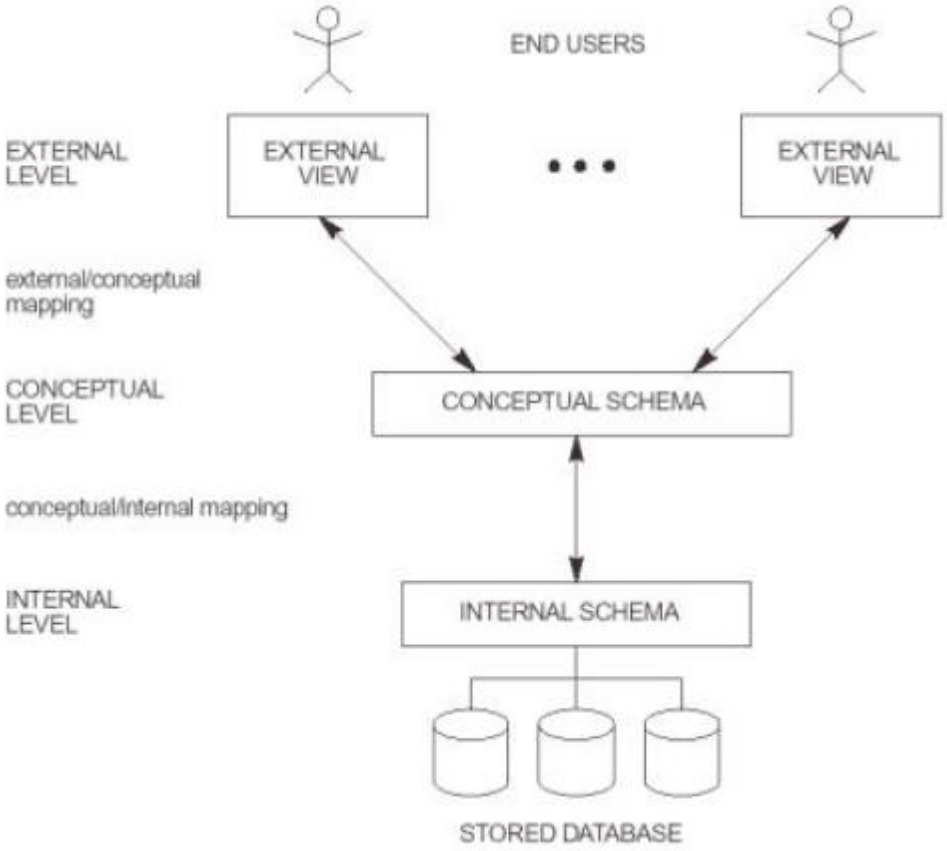
Logical data independence - the capacity to change the conceptual scheme without having to change external schemas or application programs.

Physical data independence - the capacity to change the internal scheme without having to change the conceptual (or external) schemas. Changes to the internal schema may be needed because some physical files had to be reorganized.

10. **Explain the three-schema architecture. What is the logical data independence and Physical data independence? 8 marks (June/July 2015)**

One important characteristic of the database approach was the insulation of programs and data (see Chapter 1). We can define two types of data independence:

Logical data independence - the capacity to change the conceptual scheme without having to change external schemas or application programs.



Physical data independence - the capacity to change the internal scheme without having to change the conceptual (or external) schemas. Changes to the internal schema may be needed because some physical files had to be reorganized.

11) Define the database and briefly explain the implicit properties of the database?
4 Marks (June/July 2015)

The main implicit properties of a database are listed below:

1. A database represents some aspect of the real world, sometimes called the miniworld or universe of discourse (UoD). Changes to the miniworld are reflected in the database.
2. A database is a logically coherent collection of data with some inherent meaning.
3. A database is designed, built, and populated with data for a specific purpose.

12) Define the following with examples: 10 marks (Dec 14/Jan 15)

i) Value set it consists of set of values

ii) Complex attributes, we refer to an attribute that involves some combination of multivalued *and* compositeness as a **complex** attribute.

iii) Data model a collection of concepts that can be used to describe the conceptual/logical structure of a database--- provides the necessary means to achieve this abstraction.

iv) Schema constructs which is specified during design and is not expected to change often.

v) Meta data (i.e., data about data) is stored in the so-called **system catalog**,