Relational Database Management Systems – Oct 2010

I. Section-A: 5 X 4 =20 Marks

1. Define (a) Database (b) DBMS

Database: Database is a collection of inter-related data items that can be processed by one or more application systems

DBMS: Database Management System is a collection of interrelated data and set of programs to access those data. The DBMS is a general purpose software system that facilitates the process of defining constructing and manipulating databases for various applications.

2. Draw and explain E-R diagram with an example.

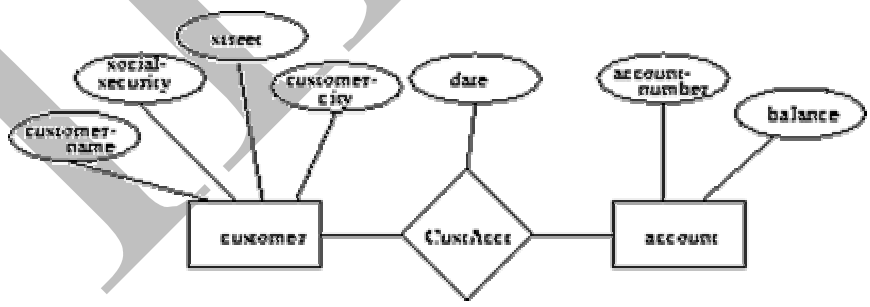
E-R Diagram: An entity-relationship diagram (ERD) is a data modeling technique that creates a graphical representation of the entities, and the relationships between entities, within an information system. Any ER diagram has an equivalent relational table, and any relational table has an equivalent ER diagram.

Entity: The entity is a person, object, place or event for which data is collected. It is equivalent to a database table. An entity can be defined by means of its properties, called attributes. For example, the CUSTOMER entity may have attributes for such things as name, address and telephone number.

Relationship: The relationship is the interaction between the entities.

E-R diagram components are:

- Rectangles representing entity sets.
- Ellipses representing attributes.
- Diamonds representing relationship sets.
- Lines' linking attributes to entity sets and entity sets to relationship sets.



3. Explain the difference between table and view.

Table: Table are defined in three steps.

1. The name of the table is given.
2. Each column is defined, possibly including column constraints.
3. Table constraints are defined.

Create table Tablename(column-name1, data type(number of characters))
(column-name2, data type(number of characters))
……...

Prasanth Kumar K
(Head-Dept of Computers, IIMC)

(column-name, data type(number of characters))

View: A definition of a restricted portion of the database. Views are useful for maintaining confidentiality and restricts access to selected parts of the database and for simplifying frequently used
query types.

The formats of create view command is

Create view V As select statement
As = query specification
V= view name

Create view stud-view as
(Select stud-no, stud-name
From student
Where percentage<=35)

A view is a database object that represents one or more database tables. It doesn't occupy any table Space.

4. Explain Update and Insert statement.

Update: The update statement is used for changing records.
The general syntax is:
UPDATE <table-name>
SET <column> = <value>
[where condition]

E.g. Update emp
set sal=sal+1000;

The above statement increases the salary of all employees by Rs.1000.

e.g. Update emp
set sal=sal+1500
where job='clerk';

The above statement increases the salary of clerical employees by Rs.1500.

Insert: To get data into a database, we need to use the 'insert' statement. The general syntax is:
INSERT INTO <table-name> (<column1>,<column2>,<column3>,...)
VALUES (<column-value1>,<column-value2>,<column-value3>);

Eg. Insert into emp(empno,ename,sal) values(200,'abc',10000);

5. Write note on DBA Goals.

A database must be protected from accidents, such as input or programming errors, from malicious use of the database, and from hardware of software failures that corrupt data. Protection from accidents that cause data inaccuracies is part of the goal of maintaining data integrity. These accidents include failures during transaction processing. Logical errors that violate the assumption those transactions preserve database consistency constraints, and anomalies due to concurrent access to the database (concurrent processing).

Protecting the database from unauthorized or malicious use is termed data security. Although the dividing line between data integrity and data security is not precise, a working definition is as follows:

1. Integrity is concerned with making certain that operations performed by users are correct and maintain database consistency.
2. Security is concerned with limiting users to performing only those operations that are allowed.

The possibility of hardware or software failure requires that database recovery procedures be implemented as well. That is, means must be provided to restore databases that have been corrupted by system malfunctions to a consistent state.

6. Explain the advantages of distributed query processing?

Some database systems support relational databases whose parts are physically separated. Different relations might reside at different sites, multiple copies of a single relation can be distributed among several sites, or one relation might be partitioned into subrelations and these subrelations distributed. In order to evaluate a query posed at a given site, it may be necessary to transfer data between various sites. The key consideration here is that the time required to process such a query will largely be comprised of the time spent transmitting data between sites rather than the time spent on retrieval from secondary storage or computation.

Semijoin: - Suppose the relations R and S shown in Figure. Is stored at sites 1 and 2, respectively. If we wish to respond to a query at site 1 which requires the computation:

JOIN (R, S),

We could transmit all of S from site 2 to site 1 and compute the join at site 1. This would involve the transmission of all 24 values of S.

Site 1		*Site 2*		
R		S		
A1	*A2*	A2	A3	A4
1	3	3	13	16
1	4	3	14	16
1	6	7	13	17
2	3	10	14	16
2	6	10	15	17
3	7	11	15	16
3	8	11	15	16
3	9	12	15	16

7. Explain Distributed Two Phase Locking.

Two-phase locking (2PL) synchronizes reads and writes by explicitly detecting and preventing conflicts between concurrent operations. Before reading date item x, a transaction must have a read lock on x. Before writing into x, it must have a write lock on x. The ownership of locks is generally governed by two rules.

1. Different transactions cannot simultaneously own conflicting locks,

2. Once a transaction surrenders ownership of a lock, it may never obtain additional locks.

8. Explain the following disk performance factors: (a) Data Transfer Rate (b) Data Transfer Time.

Data Transfer Rate: - The rate at which data can be read from the disk from the main, memory, or equivalently, the rate at which data are written from main memory to disk. Data Transfer Rate refers to the amount of time required to transfer data from the disk to primary memory. It is a function of rotational speed and the density recorded data. Data Transfer Time is usually expressed in thousands of bytes per second.

Data Transfer Time: - The expected time (T) to access a disk address and transfer a block of data is estimated as

$T = A+R/2+L/2$

Where A is the Access motion time, R is the Rotational delay, L is the length of the block in bytes, and D is the Data Transfer Rate.

II. Section-B: 5 X 10 =50 Marks

9 (a). Explain the components of DBMS.

A complete database system in an organization consists of four components.

(i). Hardware: The hardware is the set of physical devices on which a database resides. It consists of one or more computers, disk drives, CRT terminals, printers, tape drivers, connecting cables, etc.

The computers used for processing the data in the database may be mainframe, mini computers or personal computers. Mainframe and mini computers have traditionally been used on a stand-alone basis to support multiple users accessing a common database. Personal computers are often used with stand-alone databases controlled and accessed by a single user.

Disk drivers are the main storage mechanism for databases. Desktop computers, CRT terminals and printers are used for entering and retrieving information from the database.

The success of the database system has been heavily dependent on advances in hardware technology. A very large amount of main memory and disk storage is required to maintain and control the huge quantity of data stored in a database.

(ii). Software: A database system includes two types of software:

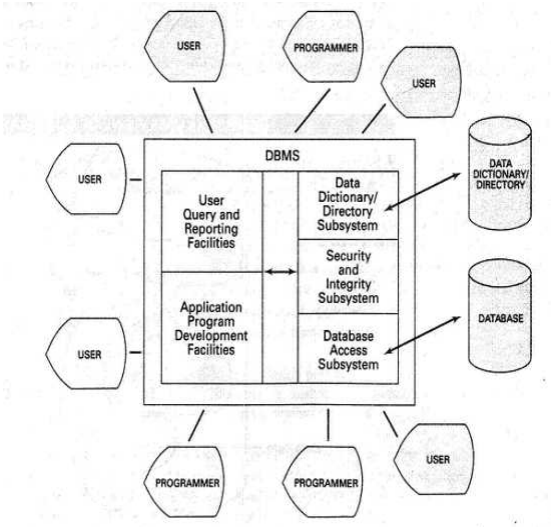
a. General Purpose database management software usually called the database management system (DBMS).
b. Application software that uses DBMS facilities to manipulate the database to achieve a specific business functions.

Application software is generally written by programmers to solve a specific company problem. It may be written in languages like COBOL or C or it may be written in a language supplied by DBMS like SQL. Application software uses the facilities of the DBMS to access and manipulate data in the database providing reports or documents needed for the information and processing needs of the company.

The DBMS is system software similar to an operating system. It provides a number of services to end users and programmers.

DBMS typically provides most of the following services.

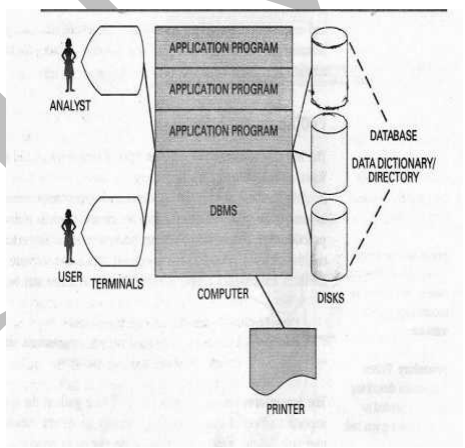
1. A central data definition and data control facility known as a data dictionary/directory or catalog.
2. Data security and integrity mechanisms.
3. Concurrent data access for multiple users.
4. User-oriented data query, manipulation and reporting capabilities.
5. Programmer-oriented application system development capabilities.



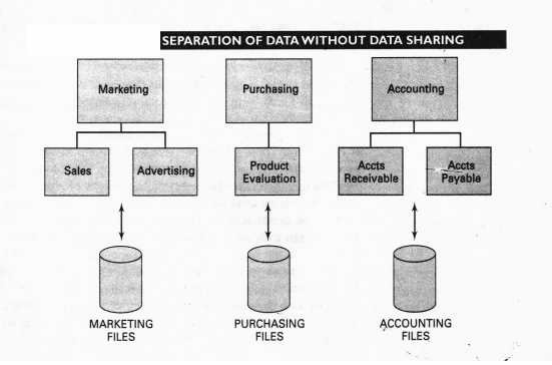
(iii). Data: No database system can exist without data. Data can be collected and entered into the database according to the defined structure.

(iv). People: Two different types of people concerned with the database. They are:

1. Users: Executives, Managers, Staff, Clerical personnel.
2. Practitioners: Database Administrators, Programmers.



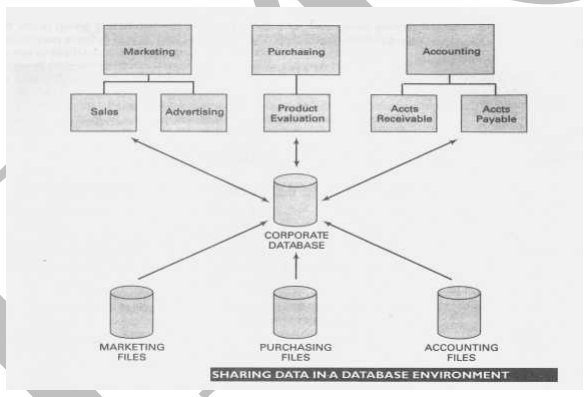
9(b). Explain data sharing in detail.



The most significant difference between a file-based system and a database system is that data are shared.

There are three types of data sharing:

(i). Sharing between Functional Units: The data sharing suggests that people in different functional areas use common pool of data, each of their own applications. Without data sharing, the marketing group may have their data files, the purchasing group theirs, the accounting group theirs and so on. Each group benefits only from its own data. The combined data are more valuable than the sum of the data in separate files. Not only does each group continue to have access to its own data but, within reasonable limits of control, they have access to other data as well. The concept of combining data for common use is called data integration.



(ii). Sharing data between Different Levels of Users: Different levels of users need to share data. Three different levels of users are normally distinguished: operations, middle management and executive. These levels correspond to the three different types of automated business systems that have evolved during the past three decades:

a. Electronic Data Processing (EDP): EDP was first applied to the lower operational levels of the organization to automate the paperwork. Its basic characteristics include:

- A focus on data, storage, processing and flows at the operational level.
- Efficient transaction processing.
- Summary reports for management.

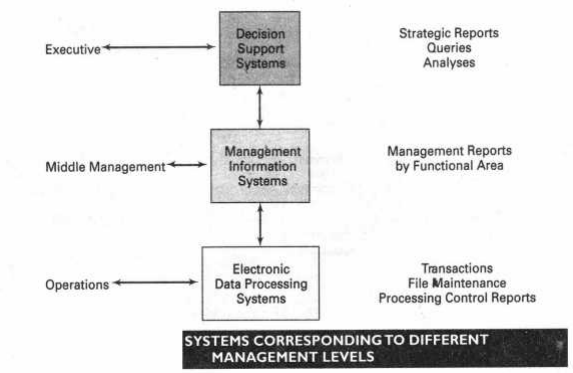
b. Management Information System (MIS): The MIS approach elevated the focus on information systems activities with additional emphasis on integration and planning of the information systems function. This includes:

- An information focus aimed at the middle mangers.
- An integration of EDP jobs by business function such as production MIS, marketing MIS, personnel MIS, etc.
- Inquiry and report generation usually with a database.

Prasanth Kumar K
(Head-Dept of Computers, IIMC)

c. Decision Support System: DSS is focused still higher in the organization with an emphasis on the following characteristics:

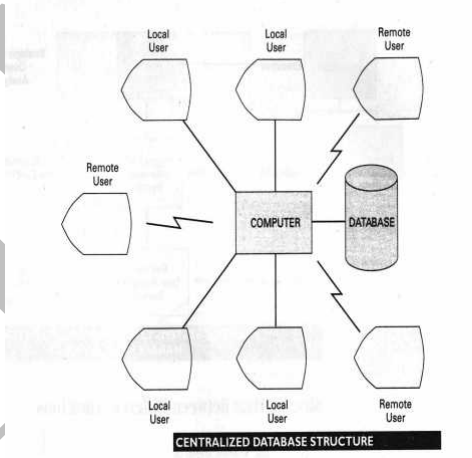
- Decision focused, aimed at top managers and executive decision makers.
- Emphasis on flexibility, adaptability and quick response.
- Support for the personnel decision-making styles of individual mangers.



SYSTEMS CORRESPONDING TO DIFFERENT MANAGEMENT LEVELS

(iii). Sharing data between Different Locations: A company with several locations has important data distributed over a wide geographical area. Sharing these data is a significant problem.

A centralized database is physically confined to a single location, controlled by a single computer. Most functions for the databases are created are accomplished more easily if the database is centralized. That is, it is easier to update, back up, query and control access to a database if we know exactly where it is and what software controls it.



CENTRALIZED DATABASE STRUCTURE

10(a). What is Normalization? Explain any three normal forms.

Normalization is the process of converting a relation into standard form.

It is of different types:

1st Normal Form: A table is in first normal form if all the key attributes have been defined and it contains no repeating groups

2nd Normal Form: A table is in second normal form (2NF) if and only if it is in 1NF and every non key attribute is fully functionally dependent on the whole of the primary key (i.e. there are no partial dependencies).

3rd Normal Form: A table is in third normal form (3NF) if and only if it is in 2NF and every non key attribute is non transitively dependent on the primary key (i.e. there are no transitive dependencies)

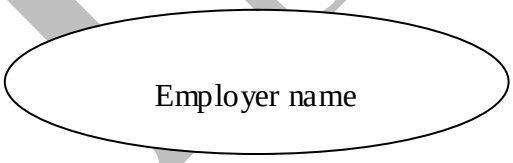
1. Anomalies can occur when a relation contains one or more transitive dependencies.
2. A relation is in 3NF when it is in 2NF and has no transitive dependencies.
3. A relation is in 3NF when 'All non-key attributes are dependent on the key, the whole key and nothing but the key'.

Boyce-Codd Normal Form: A table is in Boyce-Codd normal form (BCNF) if and only if it is in 3NF and every determinant is a candidate key.

10 (b). Explain the following terms:

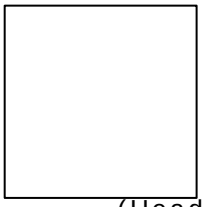
(a). Entity: An entity is a thing or object in the real world. An entity set is a set of entities of same properties of attributes. An entity is represented in the shape of rectangle. The letters in the entity should be capital letters only. Ex-EMPLOYER

(b). Attribute: They are descriptive properties possessed by each member of an entity set. An attribute is represented in the shape of ellipse. Ex- employer salary, employer designation, employer name etc.



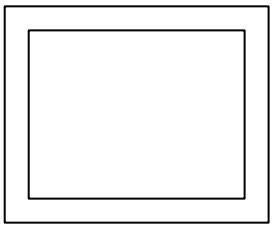
(c). Relationship: The association between two or more entities is called relationship. They are represented in the form of rhombus or diamond shape.

(d). Specilization: An object set that is a subset of another object set.
(e). Generalization: An object set that is a superset of another object set.
(f). Strong Entity: A strong relationship also known, as identifying relationship, which is not dependent upon others.



Prasanth Kumar K
(Head-Dept of Computers, IIMC)

(g). Weak Entity: It is otherwise called as non-identifying relationship, which is dependent upon others.



(h). Simple Attribute: The attributes are simple i.e. they are not divided. For example, age, sex, marital status would be classified as simple attributes.
(i). Composite attributes: They can be divided into sub-parts. For example, address can be sub-divided into street, city, state, etc.
(j). Multi valued attribute: Any attributes that have one or more number of dependents. For example, a person may have several college degrees and a household may have several phones with different numbers.

11(a). Explain Schema, Table, View. How to create Schema, Table and View in SQL?

Schema: The logical design of the database is called Schema. The concept of relation schema corresponds to the programming language notion of type definition.

Create Schema Sch_name

Authorization Auth_name

Domain definition

Table definition

View definition

Table: Table is defined in three steps.

1. The name of the table is given.
2. Each column is defined, possibly including column constraints.
3. Table constraints are defined.

```
Create table Tablename(column-name1, data type(number of characters))
                      (column-name2, data type(number of characters))
                      ……...
                      (column-name, data type(number of characters))
```

View: A definition of a restricted portion of the database. Views are useful for maintaining confidentiality and restricts access to selected parts of the database and for simplifying frequently used
query types.

The formats of create view command is

```
Create view V As select statement
        As = query specification
        V= view name
```

```
Create view stud-view as
(Select stud-no, stud-name
 From student
Where percentage<=35)
```

A view is a database object that represents one or more database tables. It doesn't occupy any table Space.

11(b). Explain the DDL commands in SQL with examples.

1. CREATE 2. ALTER 3. DROP 4. TRUNCATE 5.RENAME

12(a). Explain file organization in detail in DBMS.

File organization is the method in which the way records are stored in the file.

There are three basic ways of physically organizing files on storage devices. Sequential organization, indexed-sequential organization and direct organization. This is not an entire set of all organization options available but those that are omitted are modifications of these basic organization types. Therefore it is not necessary to be exhaustive in order to cover the essential concepts.

In discussing the topic at hand, the terms organization and access are often used loosely if not interchangeably. The reason is that the way in which data are stored is closely intertwined with the method access.

Sequential File Organization: - Sequential file organizational means that records are stored adjacent to one another according to a key such as employee number, account number, and so forth. A conventional implementation arranges the records in acesending order of key values. This is efficient method of organizing records when an application. Such as a payroll program, will be updating a significant number of the stored records.

If a sequential file is maintained on magnetic tape, its records can only be accessed in a sequential manner. That is, if access to the tenth record in sequence is desired generally the preceding nine records must be read. Direct access of a particular record is impossible. Consequently magnetic tapes are not well suited for database operations and are usually relegated log files and recording archival information.

Indexed- Sequential File Organization: - When files are sequentially organized on a disk pack, however, direct access of records is possible. Indexed-sequential file organization provides facilities for accessing records both sequentially and directly. Records are stored in the usual physical sequence by primary key. In addition an index of record locations is stored on the disk. This allows records to be accessed sequentially for applications requiring the updating of large numbers of records, as well as providing the ability to access records directly in response to user queries.

Direct File Organization: - We have studied two forms of file organization: Sequential and indexed sequential. We have concurrently outlined the two associated methods of file access: sequential access and direct access. Records in a simple sequential file organization can be accessed directly, as well as sequentially. We now turn to a discussion of a third type of file organization called direct or hashed. Only direct access methods are applicable to this type of file organization.

12(b). Explain how Security is given in database?

Database integrity problems can be challenging, but they are generally easier to cope with than malicious access to the database, which includes the following:

2. Theft of information
3. Unauthorized modification of data
4. Unauthorized destruction of data

Thus, database security methods focus on preventing unauthorized users from accessing the database. Because DBMS features that make the database easy to access and manipulate also open doors to intruders, most DBMS include security features that allow only authorized persons or processing that can be accompanied once access is made.

Authentication: - Database access usually requires user authentication and authorization. For user authentication, the first level of security establishes that the person seeking system the user knows, such as log-on number and password, (2) something the user possesses, such as plastic ID card, or (3) a physical representation of the user, such as fingerprint or voiceprint.

Authorization and views: - A view is a means of providing a user with a personalized model of the database. It is also a useful way of limiting a user's access to various positions of the database: Data a user does not need to se are simply hidden from view. This simplifies system usage while promoting security. Executing selects, projections, and joins on existing relations can represent views. The user might also be restricted from seeing any part of the existing relation or from executing joins on certain relations.

Types of Views: - Different types of access authorization may be allowed for a particular view, such as the following:

1. Read authorization: allows reading, but not modification of data.
2. Insert authorization: allows insertion of new data, but no modification of existing data.
3. Update authorization: allows modification of data, but not deletion.
4. Delete authorization: allows deletion of data.

Views and security in SQL: -
CREATE VIEW viewname As (select statement)

Encryption: - The various authentication and authorization measures that are standard for protection access to database may not be adequate for highly sensitive data. In such instances, it may be desirable to encrypt the data. Encrypted data cannot be read by an intruder unless that party knows the method of encryption. Considerable research has been devoted to developing encryption methods.

13(a). What is Distributed Database Design? Explain Distributed Database Design.

Distributed Database Design: A database that is distributed among a network of geographically separated locations. A distributed database is not entirely stored in one central location but is distributed among a network of locations that are geographically separated and connected by communication links. Each location has its own database and it also able to access data maintained at other locations.

The design of a distributed database system can be a complex task. Careful consideration must be given to the objectives and strategies to be served by the design and parallel decisions must be made as to how the data are to be distributed among the various network sites.

Strategies and Objectives: - Some of the strategies and objectives that are common to most implementation of distributed database systems are:

1) Location Transparency: - Location transparency enables a user to access data without knowing, or being concerned with, the site at which the data reside. The location of the data is hidden from the user.

2) Replication Transparency: - Replication transparency means that when more than one copy of the data exists, one copy must be chosen when retrieving data, and all copies must be updated when changes are made. Choosing one copy of the data for retrieval and always ensuring that all copies of the data are updated can be a burden on users. A DBMS should handle all such requirements, thereby freeing the user to concentrate on information needs.

3) Configuration Independence: - Configuration independence enables the organization to add or replace hardware without changing the existing software components of the DBMS. Configuration independence results in a system that is expandable when its current hardware is saturated.

4) No homogeneous DBMSs: - It is sometimes desirable to integrate database maintained by different DBMS on different computers. Often the DBMS are supplied by different vendors and may support different data models. One approach to integrating this database is to provide a single user interface that can be used to access the data maintained by the non-homogenous DBMS. The different data models supported by the non-homogenous DDBMS are hidden from the user by this single, system wide interface.

5) Data Replication: - Data replication occurs if the system maintains several identical copies of a relation, R, with each copy being stored at a different site. Typically replication is introduced to increase the availability of the system: When a copy is unviable due to site failure(s), it should be possible to access another copy.

13(b). Explain the functions and capabilities of DBMS

A) The Data Dictionary/Directory: - An effective database system will allow growth and modification in the database without comprising the integrity of its data. The data dictionary/directory (DD/D) aids the accomplishment of this objective by allowing the definitions of data to be maintained separately from the data itself. This allows changes to be made to the data definitions with no effect on the stored data. For example, the subschema used by a particular program could be modified without in any way affecting the stored data. Other benefits provided by the DD/D include these:

- Physical storage structures can be changed without affecting the programs that use the data.
- Passwords and other security measures can be stored in the DD/D to facilitate control over data access.
- Centralized data definition enables easy reporting on the status of the database: Why is responsible for the various data items.

To yield these benefits, the DD/D usually includes the following features:

- A language for defining entries in the DD/D.
- A manipulation language for adding, deleting, and modifying entries in the DD/D
- Methods for validating entries in the DD/D
- Means for producing reports concerning the data contained in the DD/D.

Data Security and Integrity: -

1) Access Controls: - Access control is an important factor because they are a means of preventing unauthorized access to data. In the data-sharing database environment, good access controls are essential.

2) Concurrency controls: - Concurrency controls are a means of manipulating data integrity in the multi-user environment. Suppose user A and user B both access a given record at (essentially) the same time in order to process a transaction against the record. The DBMS must somehow limit access by one of the users until the others transaction has been completed. Without this type of facility, the accuracy and consistency of the database can rapidly erode.

3) View Controls: - It provides an automated means of limiting what a user is allowed to access from a given relation. This is a powerful feature that is commonly provided by relational DBMS. The ease of creating views and the capability of the view facility can be a useful distinguishing factor among DBMSs. The DBMS purchaser may also be interested in whether views can be updated and what limitations may apply.

4) Encryption: - It facilitates may be important to institutions whose databases contain very sensitive data. Encryption can also be important for the maintenance of a secure password directory.

5) Backup and Recovery controls: -Effective Backup and recovery controls are absolutely essential to efficient operation of the database system. The ease of use of backup and recovery controls, and their completeness, and their reliability should be major factors in the DBMS selection decision.

Query, Data Manipulation, and Reporting Capabilities: -

The DBMS's ability to support reporting requirements, along with users' query and data manipulation needs, is the cornerstone of today's management information systems. A sound DBMS is going to provide the capability to generate structured reports in a variety of formats. In addition, the DBMS will provide a query language that is powerful, yet easy to learn and use. The language should be able to support both planned and unplanned query requirements with short response times.

Support of Specialized Programming Requirements: -

Developing specialized programs to interface with the DBMS requires facilities for supporting program development and program testing. A worthy DBMS will provide a host language for expressing standard procedural program structures or will provide an interface capability for quick prototyping of applications.

Physical Data Organization Options: -

The firm acquiring a DBMS may not wish to involve itself in the details of physical data organization. Instead, it may gauge the efficiency of a DBMS's physical organization by running sample applications.

For those who are interested, however, exploring the physical organization features may be of value. For example, it is known that the inverted list is most efficient in supporting multikey retrieval, whereas the chain list is superior for file updating since there is no need for updating a separate file. Information on other architectural features may be elicited in the process of considering the DBMS's capability to support the types of applications common to the firm.