ST. XAVIER’S COLLEGE

**(Affiliated to Tribhuvan University)**

**Maitighar, Kathmandu**

****

**Database Management System**

**Theory Assignment (#2)**

**SUBMITTED BY**

**Pratik Gautam**

**013BSCCSIT029**

**4th sem/ 2nd year**

**SUBMITTED TO**

**Er. Sanjay K Yadav**

**Lecturer**

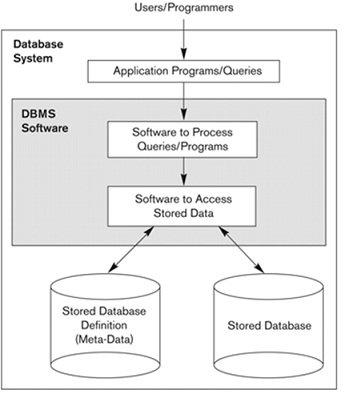
**Department of Computer Science**

**St. Xavier’s College**

**Date of submission: 2072/ 04/11**

1. **Simplified database system organization**

A DBMS (database management system) is a collection of programs that enables users to create and maintain database. The DBMS is a common purpose software system that facilitates the process of constructing, defining, manipulating and sharing databases among various users as well as applications. Defining a database state the database involves specifying the constraints, data types and structures of the data to be stored in the database. The descriptive information is as well stored in the database in the form database catalogue or dictionary- it is called meta-data. Manipulating the data comprises the querying the database to retrieve the specific data.



1. **Approaches to management of data**
2. **Database approach**

The database approach is a way in which data is stored within a computer. It is organized into various charts that are accessed by a variety of computer applications from different locations. Databases are composed of a variety of information that is pertinent and relevant to the organization that is using the database. Databases are often composed of two different pieces of information. The data and the schema are related to each other, but are not the same thing. The schema is the way in which the data will be organized within a table or chart. It can be complex, but is often as simple as the rules that the data must obey in order to create the database. Essentially, it organizes the data. The data is the actual information that is put into the database. It can be composed of numbers, letters or other characters and is representative of some physical entities. Certain rules can be set in place by the schema that will prevent different types of data from being put into the database. For example, some employer related information may prevent an employer from having a date of birth that shows that are less than 18 years old. This can help human resources catch any errors that exist with employee birthdates.

1. **File system approach**

A file system is the method an operating system uses to name files and assign them locations for efficient storage and retrieval. File systems specify conventions for naming files, including the maximum number of [characters](http://whatis.techtarget.com/definition/character) in a name, which characters can be used and, in some systems, how long the file name [suffix](http://searchcio-midmarket.techtarget.com/definition/suffix) can be. A file system also includes a format for specifying the [path](http://searchnetworking.techtarget.com/definition/path) to a file through the structure of [directories](http://searchwindowsserver.techtarget.com/definition/directory).

1. **Database vs. file system approach**
2. **Data abstraction**

Data abstraction is the reduction of a particular body of data to a simplified representation of the whole. Abstraction, in general, is the process of taking away or removing characteristics from something in order to reduce it to a set of essential characteristics. As in abstract art, the representation is likely to be one potential abstraction of a number of possibilities. A database abstraction layer, for example, is one of a number of such possibilities. Data abstraction is usually the first step in database design. A complete database is much too complex a system to be developed without first creating a simplified framework. Data abstraction makes it possible for the developer to start from essential elements -- data abstractions -- and incrementally add data detail to create the final system.

1. **Reliability**

The Process Equipment Reliability Database is your source of industry-reported equipment reliability data. The purpose of the PERD database is to provide high quality, valid, and useful data pertaining to the hydrocarbon and chemical process industries.  These data can support equipment availability analysis, reliability and design improvements, maintenance strategies, quantitative risk analysis, and life cycle cost determinations.

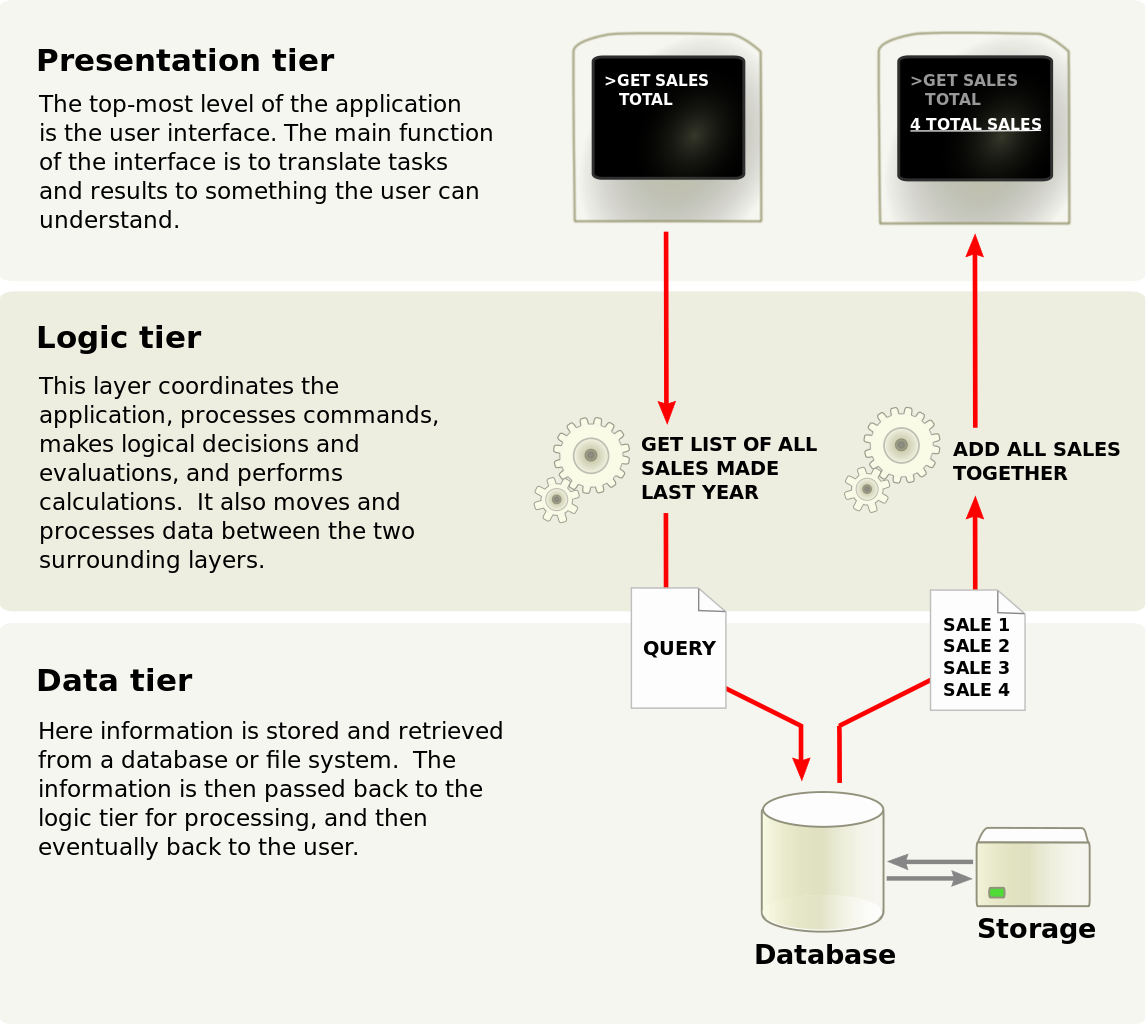
1. **Efficiency / performance**
2. **3 layer architecture (ANCI / APARC architecture)**

Three-tier architecture is a client-server architecture in which the functional process logic, data access, computer data storage and user interface are developed and maintained as independent modules on separate platforms. Three-tier architecture is a software design pattern and well-established software architecture

Three-tier architecture allows any one of the three tiers to be upgraded or replaced independently. The user interface is implemented on a desktop PC and uses a standard graphical user interface with different modules running on the application server. The relational database management system on the database server contains the computer data storage logic. The middle tiers are usually multitier.

The three tiers in three-tier architecture are:

1. **Presentation Tier:** Occupies the top level and displays information related to services available on a website. This tier communicates with other tiers by sending results to the browser and other tiers in the network.
2. **Application Tier:** Also called the middle tier, logic tier, business logic or logic tier, this tier is pulled from the presentation tier. It controls application functionality by performing detailed processing.
3. **Data Tier:** Houses database servers where information is stored and retrieved. Data in this tier is kept independent of application servers or business logic.



1. **Advantage and disadvantage of DBMS**

* 1. **Controlling Redundancy**: In file system, each application has its own private files, which cannot be shared between multiple applications. 1:his can often lead to considerable redundancy in the stored data, which results in wastage of storage space. By having centralized database most of this can be avoided. It is not possible that all redundancy should be eliminated. Sometimes there are sound business and technical reasons for· maintaining multiple copies of the same data. In a database system, however this redundancy can be controlled.
  2. **Integrity can be enforced**: Integrity of data means that data in database is always accurate, such that incorrect information cannot be stored in database. In order to maintain the integrity of data, some integrity constraints are enforced on the database. A DBMS should provide capabilities for defining and enforcing the constraints.
  3. **Inconsistency can be avoided** : When the same data is duplicated and changes are made at one site, which is not propagated to the other site, it gives rise to inconsistency and the two entries regarding the same data will not agree. At such times the data is said to be inconsistent. So, if the redundancy is removed chances of having inconsistent data is also removed.
  4. **Data can be shared:** As explained earlier, the data about Name, Class, Father \_\_name etc. of General\_Office is shared by multiple applications in centralized DBMS as compared to file system so now applications can be developed to operate against the same stored data. The applications may be developed without having to create any new stored files.

e)**Standards can be enforced** : Since DBMS is a central system, so standard can be enforced easily may be at Company level, Department level, National level or International level. The standardized data is very helpful during migration or interchanging of data. The file system is an independent system so standard cannot be easily enforced on multiple independent applications.

## Disadvantages of DBMS

a)  **Complexity :** The provision of the functionality that is expected of a good DBMS makes the DBMS an extremely complex piece of software. Database designers, developers, database administrators and end-users must understand this functionality to take full advantage of it. Failure to understand the system can lead to bad design decisions, which can have serious consequences for an organization.

b)  **Size :** The complexity and breadth of functionality makes the DBMS an extremely large piece of software, occupying many megabytes of disk space and requiring substantial amounts of [memory](http://ecomputernotes.com/fundamental/input-output-and-memory/what-are-the-different-types-of-ram-explain-in-detail) to run efficiently.

c)  **Performance:** Typically, a File Based system is written for a specific application, such as invoicing. As result, performance is generally very good. However, the DBMS is written to be more general, to cater for many applications rather than just one. The effect is that some applications may not run as fast as they used to.

**d)  Higher impact of a failure:** The centralization of resources increases the vulnerability of the system. Since all users and applications rely on the ~vailabi1ity of the DBMS, the failure of any component can bring operations to a halt.

e**) Cost of DBMS:** The cost of DBMS varies significantly, depending on the environment and functionality provided. There is also the recurrent annual maintenance cost.

**f) Additional Hardware costs:** The disk storage requirements for the DBMS and the database may necessitate the purchase of additional storage space. Furthermore, to achieve the required performance it may be necessary to purchase a larger machine, perhaps even a machine dedicated to running the DBMS. The procurement of additional hardware results in further expenditure.

1. **Drawback of using a file system to store data**

**Data Redundancy-**

The files are created in the file system as and when required by an enterprise over its growth path. So in that case the repetition of information about an entity cannot be avoided. Eg. The addresses of customers will be present in the file maintaining information about customers holding savings account and also the address of the customers will be present in file maintaining the current account. Even when same customers have a saving account and current account his address will be present at two places.

**Data Inconsistency:**

Data redundancy leads to greater problem than just wasting the storage ? i.e. it may lead to inconsistent data. Same data which has been repeated at several places may not match after it has been updated at some places. For example: Suppose the customer requests to change the address for his account in the Bank and the Program is executed to update the saving bank account file only but his current bank account file is not updated. Afterwards the addresses of the same customer present in saving bank account file and current bank account file will not match. Moreover there will be no way to find out which address is latest out of these two.

**Difficulty in Accessing Data:**

For generating ad hoc reports the programs will not already ? be present and only options present will to write a new program to generate requested report or to work manually. This is going to take impractical time and will be more expensive. For example: Suppose all of sudden the administrator gets a request to generate a list of all the customers holding the saving banks account who lives in particular locality of the city. Administrator will not have any program already written to generate that list but say he has a program which can generate a list of all the customers holding the savings account. Then he can either provide the information by going thru the list manually to select the customers living in the particular locality or he can write a new program to generate the new list. Both of these ways will take large time which would generally be impractical.

**Data Isolation:**

Since the data files are created at different times and supposedly by ? different people the structures of different files generally will not match. The data will be scattered in different files for a particular entity. So it will be difficult to obtain appropriate data. For example: Suppose the Address in Saving Account file have fields: Add line1, Add line2, City, State, Pin while the fields in address of Current account are: House No., Street No., CSE 321: Database System Md. Rashedul Islam 2 Locality, City, State, Pin. Administrator is asked to provide the list of customers living in a particular locality. Providing consolidated list of all the customers will require looking in both files headache for the administrator.

**Security and access control:**

Database should be protected from unauthorized users. Every ? user should not be allowed to access every data. Since application programs are added to the system For example: The Payroll Personnel in a bank should not be allowed to access accounts information of the customers. Concurrency Problems: When more than one user are allowed to process the database. If ? in that environment two or more users try to update a shared data element at about the same time then it may result into inconsistent data.