# ST.XAVIER’S COLLEGE

# MAITIGHAR, KATHMANDU

****

**ASSIGNMENT #2**

**Database Management System**

**Submitted By:**

Pratibha Panta

012BSCCSIT028

4th Sem, 2nd Year

**Submitted To:**

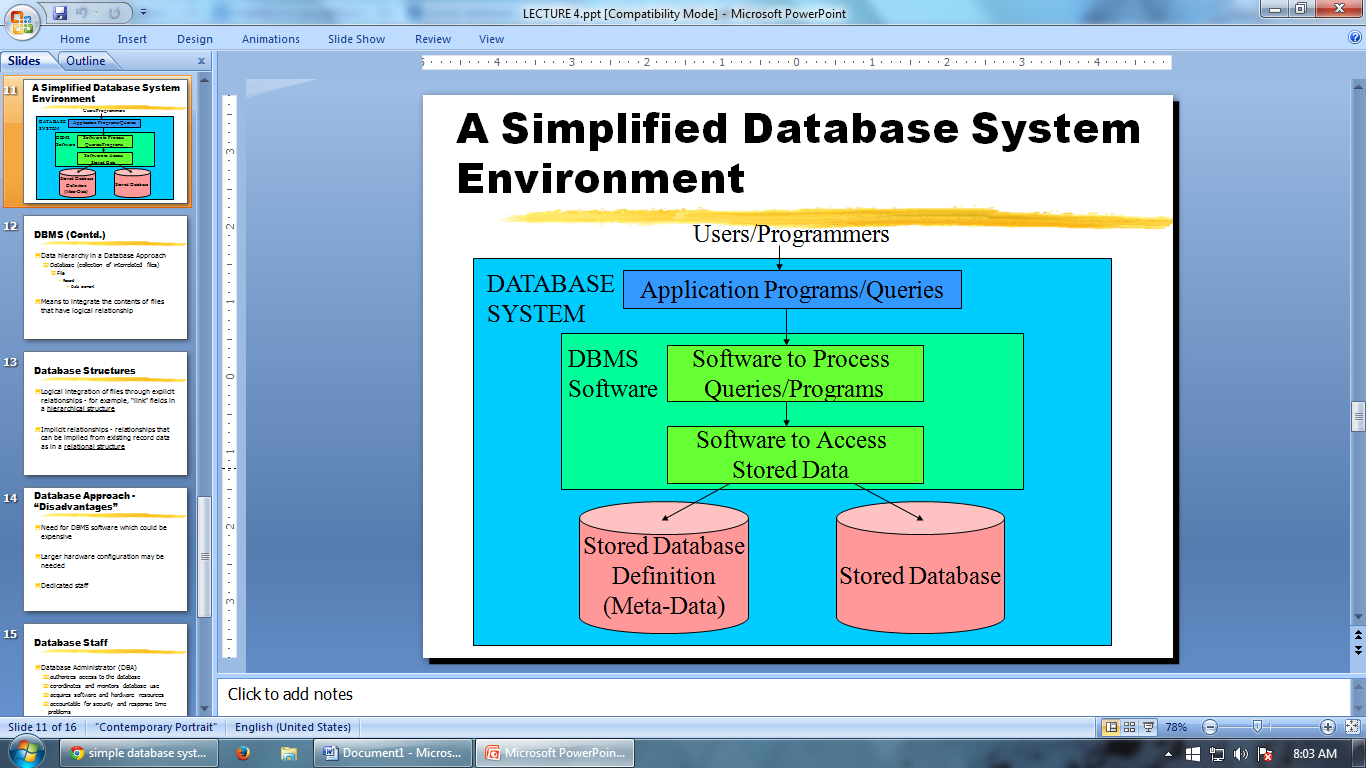
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Er. Sanjay Kumar Yadav

Department of Computer Science

Lecturer

**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.

An application program accesses the database through transferring the quarries or requests for data to DBMS.

The significant function provided by the DBMS includes protecting the database and maintain the database.

It is logical integration of files through explicit relationships - for example, “link” fields in a hierarchical structure

Implicit relationships - relationships that can be implied from existing record data as in a relational structure

**Approach to management of data**

Data management is the development and execution of architectures, policies, practices and procedures in order to manage the information lifecycle needs of an enterprise in an effective manner.

**Traditional Approach**

Programming with files or file-processing

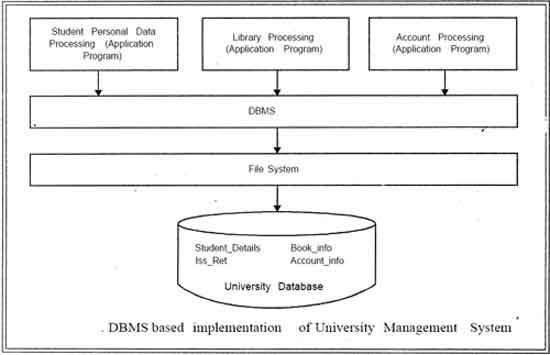
Files linked to a specific application

Structure of data files is embedded in the application programs

**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.

In order to remove all limitations of the File Based Approach, a new approach was required that must be more effective known as Database approach.



database implies separation of physical storage from use of the data by an application program to achieve program/data independence. Using a database system, the user or programmer or application specialist need not know the details of how the data are stored and such details are "transparent to the user". Changes (or updating) can be made to data without affecting other components of the system. These changes include, for example, change of data format or file structure or relocation from one device to another.

**Characteristics of database**

The data in a database should have the following features:

**Organized/Related**. It should be well organized and related.

**Shared.** Data in a database are shared among different users and applications.

**Permanent or Persistence**. Data in a database exist permanently in the sense the data can live beyond the scope of the process that created it.

**Validity/integrity/Correctness**. Data should be correct with respect to the real world entity that they represent.

**Security.** Data should be protected from unauthorized access.

**Consistency**. Whenever more than one data element in a database represents related real world values, the values should be consistent with respect to the relationship.

**Non-redundancy**: No two data items in a database should represent the same real world entity.

**Independence**. Data at different levels should be independent of each other so that the changes in one level should not affect the other levels.

**Easily Accessible**. It should be available when and where it is needed i.e. it should be easily accessible.

**Recoverable**. It should be recoverable in case of damage.

**Flexible to change**. It should be flexible to change.

To create, manage and manipulate data in databases, a management system known as database management system was developed.

**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.

In a computer, a file system (sometimes written filesystem) is the way in which files are named and where they are placed logically for storage and retrieval.

For example, DOS, Windows, OS/2, Macintosh and Unix-based operating systems (OSes) all have file systems in which files are placed somewhere in a hierarchical (tree) structure. A file is placed in a directory (folder in Windows) or subdirectory at the desired place in the tree structure.

File systems specify conventions for naming files, including the maximum number of characters in a name, which characters can be used and, in some systems, how long the file name suffix can be. A file system also includes a format for specifying the path to a file through the structure of directories.

**Drawbacks of using a file system to store data**

File structure changes usually result in program changes

~Data redundancy or duplication

->Wasted space

->Naming problems

->Inconsistency because of lack of synchronization in updating the duplicated data

**Database vs file system approach**

A database is generally used for storing related, structured data, with well defined data formats, in an efficient manner for insert, update and/or retrieval (depending on application).

On the other hand, a file system is a more unstructured data store for storing arbitrary, probably unrelated data. The file system is more general, and databases are built on top of the general data storage services provided by file systems.

There are also differences in the expected level of service provided by file systems and databases. While databases must be self consistent at any instant in time (think about banks tracking money!), provide isolated transactions and durable writes, a file system provides much looser guarantees about consistency, isolation and durability. The database uses sophisticated algorithms and protocols to implement reliable storage on top of potentially unreliable file systems. It is these algorithms that make database storage more expensive in terms of processing and storage costs that make general file systems an attractive option for data that does not require the extra guarantees provided by a database.

As technology moves forward, though, the lines are blurring, as some file systems pick up features previously the domain of databases (transactions, advanced queries) and some databases relax the traditional constraints of consistency, isolation and durability. ZFS and BTRFS might be considered examples of the former, MongoDB and CouchDB examples of the latter.

**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

**Reliability**

A reliable DDBMS is one that can continue to process user requests even when the

underlying system is unreliable, i.e., failures occur

• Failures

– Transaction failures

– System (site) failures, e.g., system crash, power supply failure

– Media failures, e.g., hard disk failures

– Communication failures, e.g., lost/undeliverable messages

• Reliability is closely related to the problem of how to maintain the atomicity and

durability properties of transactions.

**Efficiency/performance**

~Focus on relational model

~Any column in a relational database can be searched for values.

~To improve efficiency indexes using storage structures such as BTrees and Hashing are used.

But many useful functions are not indexable and require complete sacns of the database.

**3 layer architecture(ANSI/APARC architecture)**

An early proposal for a standard terminology and general architecture database a system was produced in 1971 by the DBTG (Data Base Task Group) appointed by the Conference on data Systems and Languages. The DBTG recognized the need for a two level approach with a system view called the schema and user view called subschema. The American National Standard Institute terminology and architecture in 1975.ANSI-SPARC recognized the need for a three level approach with a system catalog.

There are following three levels or layers of DBMS architecture:

1. Extenal Level

2. Conceptual Level

3. Internal Level

1. **External Level:** - External Level is described by a schema i.e. it consists of definition of logical records and relationship in the external view. It also contains the method of deriving the objects in the external view from the objects in the conceptual view.

2**. Conceptual Level**: - Conceptual Level represents the entire database. Conceptual schema describes the records and relationship included in the Conceptual view. It also contains the method of deriving the objects in the conceptual view from the objects in the internal view.

3. **Internal Level**: - Internal level indicates hoe the data will be stored and described the data structures and access method to be used by the database. It contains the definition of stored record and method of representing the data fields and access aid used.

A mapping between external and conceptual views gives the correspondence among the records and relation ship of the conceptual and external view. The external view is the abstraction of conceptual view which in turns is the abstraction of internal view. It describes the contents of the database as perceived by the user or application program of that view.

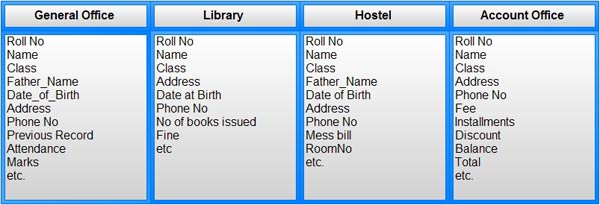
A mapping between conceptual records from the physical database.

**Advantages and disadvantages of DBMS**

The database management system has promising potential advantages, which are explained below:

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.

For example: In case of college database, there may be the number of applications like General Office, Library, Account Office, Hostel etc. Each of these applications may maintain the following information into own private file applications:



It is clear in the above database that Rollno, Name, Class, Father\_Name, Address,

Phone\_No, Date\_of\_birth which are stored repeatedly in file system in each application, need not be stored repeatedly in case of database, because every other application can access this information by joining of relations on the basis of common column i.e. Rollno. Suppose any user of Library system need the Name, Address of any particular student and by joining of Library and General Office relations on the basis of column Rollno he/she can easily retrieve this information.

Thus, we can say that centralized system of DBMS reduces the redundancy of data to great extent but cannot eliminate the redundancy because RollNo is still repeated in all the relations.

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.

In case of DBMS, this integrity constraint is applied only once on the class field of the

General Office (because class field appears only once in the whole database), and all other applications will get the class information about the student from the General Office table so the integrity constraint is applied to the whole database. So, we can conclude that integrity constraint can be easily enforced in centralized DBMS system as compared to file system.

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.

An inconsistent database is capable of supplying incorrect or conflicting information. So there should be no inconsistency in database. It can be clearly shown that inconsistency can be avoided in centralized system very well as compared to file system ..

In case of DBMS, Roll number and address occurs together only single time in General\_Office table. So, it needs single updation and then an other application retrieve the address information from General\_Office which is updated so, all application will get the current and latest information by providing single update operation and this single update operation is propagated to the whole database or all other application automatically, this property is called as Propagation of Update.

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.

5. **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.

6. **Restricting unauthorized access:** When multiple users share a database, it is likely that some users will not be authorized to access all information in the database. For example, account office data is often considered confidential, and hence only authorized persons are allowed to access such data. In addition, some users may be permitted only to retrieve data, whereas other are allowed both to retrieve and to update. Hence, the type of access operation retrieval or update must also be controlled. Typically, users or user groups are given account numbers protected by passwords, which they can use to gain access to the database. A DBMS should provide a security and authorization subsystem, which the DBA uses to create accounts and to specify account restrictions. The DBMS should then enforce these restrictions automatically.

7. **Solving Enterprise Requirement than Individual Requirement**: Since many types of users with varying level of technical knowledge use a database, a DBMS should provide a variety of user interface. The overall requirements of the enterprise are more important than the individual user requirements. So, the DBA can structure the database system to provide an overall service that is "best for the enterprise".

8**. Providing Backup and Recovery**: A DBMS must provide facilities for recovering from hardware or software failures. The backup and recovery subsystem of the DBMS is responsible for recovery. For example, if the computer system fails in the middle of a complex update program, the recovery subsystem is responsible for making sure that the .database is restored to the state it was in before the program started executing.

9. **Cost of developing and maintaining system is lower**: It is much easier to respond to unanticipated requests when data is centralized in a database than when it is stored in a conventional file system. Although the initial cost of setting up of a database can be large, but the cost of developing and maintaining application programs to be far lower than for similar service using conventional systems. The productivity of programmers can be higher in using non-procedural languages that have been developed with DBMS than using procedural languages.

10. **Data Model can be developed** : The centralized system is able to represent the complex data and interfile relationships, which results better data modeling properties. The data madding properties of relational model is based on Entity and their Relationship, which is discussed in detail in chapter 4 of the book.

11**. Concurrency Control** : DBMS systems provide mechanisms to provide concurrent access of data to multiple users.

**Disadvantages of DBMS**

The disadvantages of the database approach are summarized as follows:

1. **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.

2. **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 to run efficiently.

3. **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.

4. **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.

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

6. **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.