**ST. XAVIER’S COLLEGE**

**(Affiliated to Tribhuvan University)**

Maitighar, Kathmandu



**Database Management System Assignment #2**

**Submitted by:**

Hemanchal Joshi  
013BSCCSIT021

**Submitted to:**

|  |  |
| --- | --- |
| **Er. Sanjay Kumar Yadav**  Lecturer, St. Xavier’s College  Dept. of Computer science |  |

Date of Submission:27’th July, 2015

1. **Simplified database system organization**

A database management system (DBMS) can be an extremely complex set of software programs that controls the organization, storage and retrieval of data (fields, records and files) in a database. It also controls the security and integrity of the database. The DBMS accepts requests for data from the application program and instructs the operating system to transfer the appropriate data. When a DBMS is used, information systems can be changed much more easily as the organization’s information requirements change. New categories of data can be added to the database without disruption to the existing system.

Data security prevents unauthorized users from viewing or updating the database. Using passwords, users are allowed access to the entire database or subsets of the database, called subschema’s (pronounced "sub-skeema"). For example, an employee database can contain all the data about an individual employee, but one group of users may be authorized to view only payroll data, while others are allowed access to only work history and medical data.DBMS can maintain the integrity of the database by not allowing more than one user to update the same record at the same time. The DBMS can keep duplicate records out of the database; for example, no two customers with the same customer numbers (key fields) can be entered into the database.

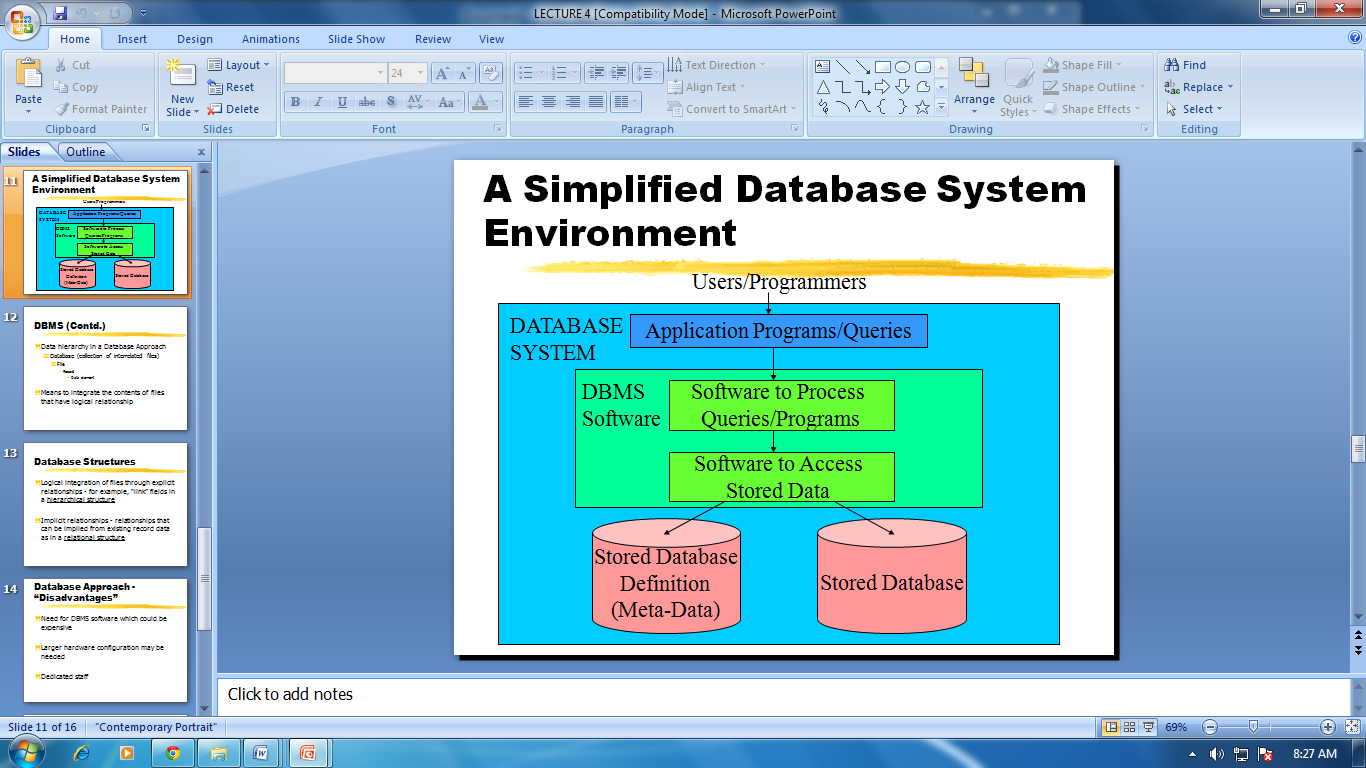


Fig: A simplified database system environment

* **Hardware Components in a Database System Environment**

Hardware identifies all the system's physical devices. It includes computers, computer peripherals, network components etc.

* **Software Components in a Database System Environment**

**Operating System**- The operating System manages all the hardware components and makes it possible for all other software to run on the computers. UNIX, LINUX, Microsoft Windows etc are the popular operating systems used in database environment.

**DBMS Software-** DBMS software manages the database with in the database system. Oracle Corporation's ORACLE, IBM's DB2, Sun's MYSQL, Microsoft's MS Access and SQL Server etc are the popular DBMS (RDBMS) software used in the database environment.

**Application Programs and Utilities**- Application programs and utilities software are used to access and manipulate the data in the database and to manage the operating environment of the database.

* **People in a Database System Environment**

People component includes all users associated with the database system. On the basis of primary job function we can identify five types of users in a database system: System Administrators, Database Administrators, Data Modelers, System Analysts and Programmers and End Users.

**System Administrators** oversees the database system's general operations. Database Administrator (DBA) physically implements the database according to the logical design. The DBA performs the physical implementation and maintenance of a database system.

**System Analysts and** programmers design and implement the application programs. They create the input screens, reports, and procedures through which end users access and manipulate the database.

**End Users** are the people who use the application. For example in case of a banking system, the employees, customer using ATM or online banking facility are end users.

* **Procedures in a Database Environment**

Procedures are the instructions and business rules that govern the design and use of the database system.

* **Data in the Database**

Data are the very important basic entity in a database. It is the collection of facts stored in the database.

1. **Approaches to management of data**

* **Database approach**

It is comprised of a single repository of data which is accessed by various users. Database system contains the database and a complete description of the database called meta-data. It consists of ability to change data structure without making changes to the programs. User is not constrained by the physical organization of data.

* **File system approach**

Files linked to a specific application. Structure of data files is embedded in the application programs in the file system approach. It is also called the traditional approach. File structure changes usually result in program changes. Data redundancy or duplication is the major disadvantage of this approach.  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.

1. **Database vs. File system Approaches**

* **Database approach**
* It is comprised of a single repository of data which is accessed by various users.
* Database system contains the database and a complete description of the database called meta-data.
* It consists of ability to change data structure without making changes to the programs.
* User is not constrained by the physical organization of data.
* **File system approach**
* Files linked to a specific application.
* Structure of data files is embedded in the application programs in the file system approach.
* It is also called the traditional approach.
* File structure changes usually result in program changes.
* Data redundancy or duplication is the major disadvantage of this approach.
* 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.

1. **Three Layer Architecture**

**1-tier architecture**

In 1-tier architecture, the DBMS is the only entity where the user directly sits on the DBMS and uses it. Any changes done here will directly be done on the DBMS itself. It does not provide handy tools for end-users. Database designers and programmers normally prefer to use single-tier architecture.

**2-tier architecture**

If the architecture of DBMS is 2-tier, then it must have an application through which the DBMS can be accessed. Programmers use 2-tier architecture where they access the DBMS by means of an application. Here the application tier is entirely independent of the database in terms of operation, design, and programming.

**3-tier Architecture (Three layer architecture)**

A 3-tier architecture separates its tiers from each other based on the complexity of the users and how they use the data present in the database. It is the most widely used architecture to design a DBMS.



* **Database (Data) Tier** − At this tier, the database resides along with its query processing languages. We also have the relations that define the data and their constraints at this level.
* **Application (Middle) Tier** − At this tier reside the application server and the programs that access the database. For a user, this application tier presents an abstracted view of the database. End-users are unaware of any existence of the database beyond the application. At the other end, the database tier is not aware of any other user beyond the application tier. Hence, the application layer sits in the middle and acts as a mediator between the end-user and the database.
* **User (Presentation) Tier** − End-users operate on this tier and they know nothing about any existence of the database beyond this layer. At this layer, multiple views of the database can be provided by the application. All views are generated by applications that reside in the application tier.

Multiple-tier database architecture is highly modifiable, as almost all its components are independent and can be changed independently.

1. **Advantages and disadvantages of DBMS**

**Advantages 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.

**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.

1. **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.

We can say the redundancy of data greatly affect the consistency of data. If redundancy is less, it is easy to implement consistency of data. Thus, DBMS system can avoid inconsistency to great extent.

**4. Data can be shared :** 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](http://ecomputernotes.com/fundamental/input-output-and-memory/what-are-the-different-types-of-ram-explain-in-detail) 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.

**7. Cost of Conversion:** In some situations, the cost oftlle DBMS and extra hardware may be insignificant compared with the cost of converting existing applications to run on the new DBMS and hardware. This cost also includes the cost of training staff to use these new systems and possibly the employment of specialist staff to help with conversion and running of the system. This cost is one of the main reasons why some organizations feel tied to their current systems and cannot switch to modern database technology.

**8. The overhead costs of using a DBMS are due to the following:**

* High initial investment in hardware, software, and training.
* Generality that a DBMS provides for defining and processing data.
* Overhead for providing security, concurrency control, recovery, and integrity functions.
* Additional problems may arise, if the database designers and DBA do not properly design the database or if the database systems applications are not implemented properly.
* Hence, it may be more desirable to use regular files under the following circumstances:
* The database and applications are simple, well defined and not expected to change.
* There are tight real-time requirements for some programs that may not be met because of DBMS overhead.
* Multiple user access to data is not required.
* An application may need to manipulate the data in a way not supported by the query language.