ST. XAVIER’S COLLEGE

**(Affiliated to Tribhuvan University)**

**Maitighar, Kathmandu**

****

**Database Management System**

**Theory Lab Assignment #3**

**SUBMITTED BY:**

**Siddhant Rimal**

**013BSCCSIT039**

**SUBMITTED TO**

|  |  |
| --- | --- |
| **Mr. Sanjay Kr. Yadav**  **( Lecturer )** |  |
| **Department of Computer Science** | |

Submission Date: August 5th 2015

1. **Additional advantages of Database Approach**
   1. **Expandability / Flexibility**

Database flexibility is a crucial criterion for database applicability. If stored in a flexible format, the same data may be useful in many different contexts. A database is very flexible as data can easily be extracted and inserted into it without knowing how the database works internally. The values of database entities can be changed according to the requirements. The database can also be designed in such a way that if a change it required, the physical structure of the database need not be changed. Also, a database will automatically expand according to the data stored in it.

* 1. **Reduce application development time**

A database makes it easy to store and retrieve data. For any application, data is a crucial component. Every application needs somewhere to store data and later retrieve it. If a database is used with an application for the purpose of storing data, the application will have an easy interface to work with data. The programmer need not worry about minor details of storing data as the database will do it for him. There is no need to open and close file streams and work with pointers because the database will return the required data with the means of a simple query.

* 1. **Economy of Scale**

A properly implemented database can give rise to an economy of scale. A database is used to store data and this database can be accessed by multiple users at a time. This gives rise to the ability to use the same database for multiple users. This reduces cost of the system as only one database is required for a large number of users. Thus, a database increases the number of users while reducing the cost of the system.

* 1. **Centralized control by the DBA**

A database administrator is a person who has access control of the system. A database administrator is responsible for designing the database and ensuring periodic maintenance of the system. A database administrator has centralized control of the database as in he can control all aspects of the database. He can add, edit, remove components and scale the database as required.

1. **Database System Components**
   1. **Data**

Data are raw facts and figures. It is a very important component of the database system. Most of the organizations generate, store and process 1arge amount of data. The data acts a bridge between the machine parts i.e. hardware and software and the users which directly access it or access it through some application programs. Data may be of different types:

* + 1. **User Data**

It consists of a table(s) of data called Relation(s) where Column(s) are called fields of attributes and rows are called Records for tables. A Relation must be structured properly.

* + 1. **Metadata**

A description of the structure of the database is known as Metadata. It basically means "data about data". System Tables store the Metadata which includes.

* Number of Tables and Table Names
* Number of fields and field Names
* Primary Key Fields
  1. **Hardware**

The hardware consists of the secondary storage devices such as magnetic disks (hard disk, zip disk, floppy disks), optical disks (CD-ROM), magnetic tapes etc. on which data is stored together with the Input/Output devices (mouse, keyboard, printers), processors, main memory etc. which are used for storing and retrieving the data in a fast and efficient manner. Since database can range from those of a single user with a desktop computer to those on mainframe computers with thousands of users, therefore proper care should be taken for choosing appropriate hardware devices for a required database.

* 1. **Software**

The Software part consists of DBMS which acts as a bridge between the user and the database or in other words, software that interacts with the users, application programs, and database and files system of a particular storage media (hard disk, magnetic tapes etc.) to insert, update, delete and retrieve data. For performing these operations such as insertion, deletion and updating we can either use the Query Languages like SQL, QUEL, Gupta SQL or application software such as Visual Basic, Developer etc.

* 1. **Users**

Users are those persons who need the information from the database to carry out their primary business responsibilities i.e. Personnel, Staff, Clerical, Managers, and Executives etc. On the basis of the job and requirements made by them they are provided access to the database totally or partially. The different types of users are:

* + 1. **Naïve Users**

They are those users who use only one part of the database application

* + 1. **Application Programmers**

They are developers who create application programs related with the database

* + 1. **Sophisticated Users**

They are those users according to whose requirements the database has been created

* + 1. **Specialized Users**

They are those users who solve problems like hardware problems that may arise outside of the database

1. **Data Communication Manager**

A **database/data-communications system** (DB/DC system) is a combination of a DC manager and a DBMS, in which the DBMS looks after the database and the DC manager handles all messages to and from the DBMS (or, more accurately, to and from applications that use the DBMS).The **data communications manager** (DC manager) is a software component that manages all message transmissions between the user and the DBMS (more accurately, between the user and some application running on top of the DBMS)

1. **Database System Utilities**
   1. **Loading**

A loading utility is used to load existing data files-such as text files or sequential files-into the database. Usually, the current (source) format of the data file and the desired (target) database file structure are specified to the utility, which then automatically reformats the data and stores it in the database. With the proliferation of DBMSs, transferring data from one DBMS to another is becoming common in many organizations. Some vendors are offering products that generate the appropriate loading programs, given the existing source and target database storage descriptions (internal schemas). Such tools are also called conversion tools

* 1. **Backup**

A backup utility creates a backup copy of the database, usually by dumping the entire database onto tape. The backup copy can be used to restore the database in case of catastrophic failure. Incremental backups are also often used, where only changes since the previous backup are recorded. Incremental backup is more complex but saves space

* 1. **File Reorganization**

This utility can be used to reorganize a database file into a different file organization to improve performance

* 1. **Performance monitoring**

Such a utility monitors database usage and provides statistics to the DBA. The DBA uses the statistics in making decisions such as whether or not to reorganize files to improve performance

1. **DATABASE SYSTEM LIFECYCLE**

The database life cycle (DBLC) defines the stages involved in getting any type of database off the drawing board and up and running. In fact, the DBLC never ends because database monitoring, modification, and maintenance are part of the life cycle, and these activities continue long after a database has been implemented. Put simply, the DBLC encompasses the lifetime of the database.

The five stages in the database life cycle are:

* 1. **Requirements analysis**

Requirements Analysis is the first and most important stage in the Database Life Cycle. It is the most labor-intensive for the database designer. This stage involves assessing the informational needs of an organization so that a database can be designed to meet those needs.

* 1. **Logical design**

During the first part of Logical Design, a conceptual model is created based on the needs assessment performed in stage one. A conceptual model is typically an entity-relationship (ER) diagram that shows the tables, fields, and primary keys of the database, and how tables are related (linked) to one another.

The tables sketched in the ER diagram are then normalized. The normalization process resolves any problems associated with the database design, so that data can be accessed quickly and efficiently.

* + 1. **conceptual model:** A description of the structure of a database.
    2. **entity-relationship (ER) diagram:** A diagram used during the design phase of database development to illustrate the organization of and relationships between data during database design.
    3. **normalization:** The process of applying increasingly stringent rules to a relational database to correct any problems associated with poor design.
  1. **Physical design**

The Physical Design stage has only one purpose: to maximize database efficiency. This means finding ways to speed up the performance of the RDBMS. Manipulating certain database design elements can speed up the two slowest operations in an RDBMS: retrieving data from and writing data to a database

* 1. **Implementation**

During the implementation stage of the DBLC, the tables developed in the ER diagram (and subsequently normalized) are converted into SQL statements and “fed” into the RDBMS to create a database. By this stage in the DBLC, the System Administrator has installed and configured an RDBMS.

* 1. **Monitoring, modification, and maintenance**

A successfully implemented database must be carefully monitored to ensure that it’s functioning properly and that it’s secure from unauthorized access. The RDBMS usually provides utilities to help monitor database functionality and security. Database modification involves adding and deleting records, importing data from other systems (as needed), and creating additional tables, user views, and other objects and tools.

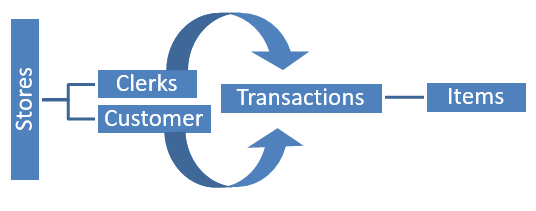
1. **CLASSIFICATION OF DBMS**
   1. **Hierarchical databases**

In the Hierarchical Database Model we have to learn about the databases. It is very fast and simple. In a hierarchical database, records contain information about there groups of parent/child relationships, just like as  a tree structure. The structure implies that a record can have also a repeating information. In this structure Data follows a series of records, It is a set of field values attached to it. It collects all  records together as a record type. These record types are the equivalent of tables in the relational model, and with the individual records being the equivalent of rows. To create links between these record types, the hierarchical model uses these type Relationships.

* 1. **Network databases**

A network databases are mainly used on a large digital computers. It more connections can be made between different types of data, network databases are considered more efficiency It contains limitations must be considered when we have to use this kind of database. It is Similar to the hierarchical databases, network databases .Network databases are similar to hierarchical databases by also having a hierarchical structure. A network database looks more like a cobweb or interconnected network of records.

In network databases, children are called members and parents are called occupier. The difference between each child or member can have more than one parent.



* 1. **Relational databases**

In relational databases, the relationship between data files is relational. Hierarchical and network databases require the user to pass  a hierarchy in order to access needed data. These databases connect to the data in different files by using common data numbers or a key field. Data in relational databases is stored in different access control tables, each having a key field that mainly identifies each row. In the relational databases are more reliable than either the hierarchical or network database structures. In relational databases, tables or files filled up with data are called relations (tuples) designates a row or record, and columns are referred to as attributes or fields.

Relational databases work on each table has a key field that uniquely indicates each row, and that these key fields can be used to connect one table of data to another.

* 1. **Object-oriented database**

In this Model we have to discuss the functionality of the object oriented Programming. It takes more than storage of programming language objects. Object DBMS's increase the semantics of the C++ and Java .It provides full-featured database programming capability, while containing native language compatibility. It adds the database functionality to object programming languages.This approach is the analogical of the application and database development into a constant data model and language environment. Applications require less code, use more natural data modeling, and code bases are easier to maintain. Object developers can write complete database applications with a decent amount of additional effort.

The object-oriented database derivation is the integrity of object-oriented programming language systems and consistent systems. The power of the  object-oriented databases comes from the cyclical treatment of both consistent data, as found in databases, and transient data, as found in executing programs.