**ST. XAVIER’S COLLEGE**

**(Affiliated to Tribhuvan University)**

Maitighar, Kathmandu



**DATABASE MANAGEMENT SYSTEM**

**LAB ASSIGNMENT#1**

**Submitted by:**

Pratibha Panta

013BSCCSIT028

**Submitted to:**

|  |  |
| --- | --- |
| **Er. Sanjay Kumar Yadav**  **Lecturer**  **Department of Computer Science**  **St Xavier’s College** |  |

Date of submission: 24th July, 2015

**Introduction**

A **database** is an organized collection of data. It is the collection of schemes, tables, queries, reports, views and other objects. The data is typically organized to model aspects of reality in a way that supports processes requiring information, such as modeling the availability of rooms in hotels in a way that supports finding a hotel with vacancies.

**Database management systems (DBMS)** are computer software applications that interact with the user, other applications, and the database itself to capture and analyze data. A general-purpose DBMS is designed to allow the definition, creation, querying, update, and administration of databases. Well-known DBMSs include MySQL, PostgreSQL, Microsoft SQL Server, Oracle, Sybase and IBM DB2. A database is not generally portable across different DBMSs, but different DBMS can interoperate by using standards such as SQL and ODBC or JDBC to allow a single application to work with more than one DBMS. Database management systems are often classified according to the database model that they support; the most popular database systems since the 1980s have all supported the relational model as represented by the SQL language. Sometimes a DBMS is loosely referred to as a 'database'.

**Overview**

Formally, a "database" refers to a set of related data and the way it is organized. Access to this data is usually provided by a "database management system" (DBMS) consisting of an integrated set of computer software that allows users to interact with one or more databases and provides access to all of the data contained in the database (although restrictions may exist that limit access to particular data). The DBMS provides various functions that allow entry, storage and retrieval of large quantities of information as well as provides ways to manage how that information is organized.

Because of the close relationship between them, the term "database" is often used casually to refer to both a database and the DBMS used to manipulate it.

Outside the world of professional information technology, the term *database* is often used to refer to any collection of related data (such as spreadsheet or a card index). This article is concerned only with databases where the size and usage requirements necessitate use of a database management system.[[2]](https://en.wikipedia.org/wiki/Database#cite_note-Ullman-2)

Existing DBMSs provide various functions that allow management of a database and its data which can be classified into four main functional groups:

* **Data definition** – Creation, modification and removal of definitions that define the organization of the data.
* **Update** – Insertion, modification, and deletion of the actual data.[[3]](https://en.wikipedia.org/wiki/Database#cite_note-3)
* **Retrieval** – Providing information in a form directly usable or for further processing by other applications. The retrieved data may be made available in a form basically the same as it is stored in the database or in a new form obtained by altering or combining existing data from the database.[[4]](https://en.wikipedia.org/wiki/Database#cite_note-4)
* **Administration** – Registering and monitoring users, enforcing data security, monitoring performance, maintaining data integrity, dealing with concurrency control, and recovering information that has been corrupted by some event such as an unexpected system failure

**Purpose of Database Management Systems**

Organizations use large amounts of data. A database management system (DBMS) is a software tool that makes it possible to organize data in a database.

The standard acronym for database management system is DBMS, so you will often see this instead of the full name. The ultimate purpose of a database management system is to store and transform data into information to support making decisions.

A DBMS consists of the following three elements:

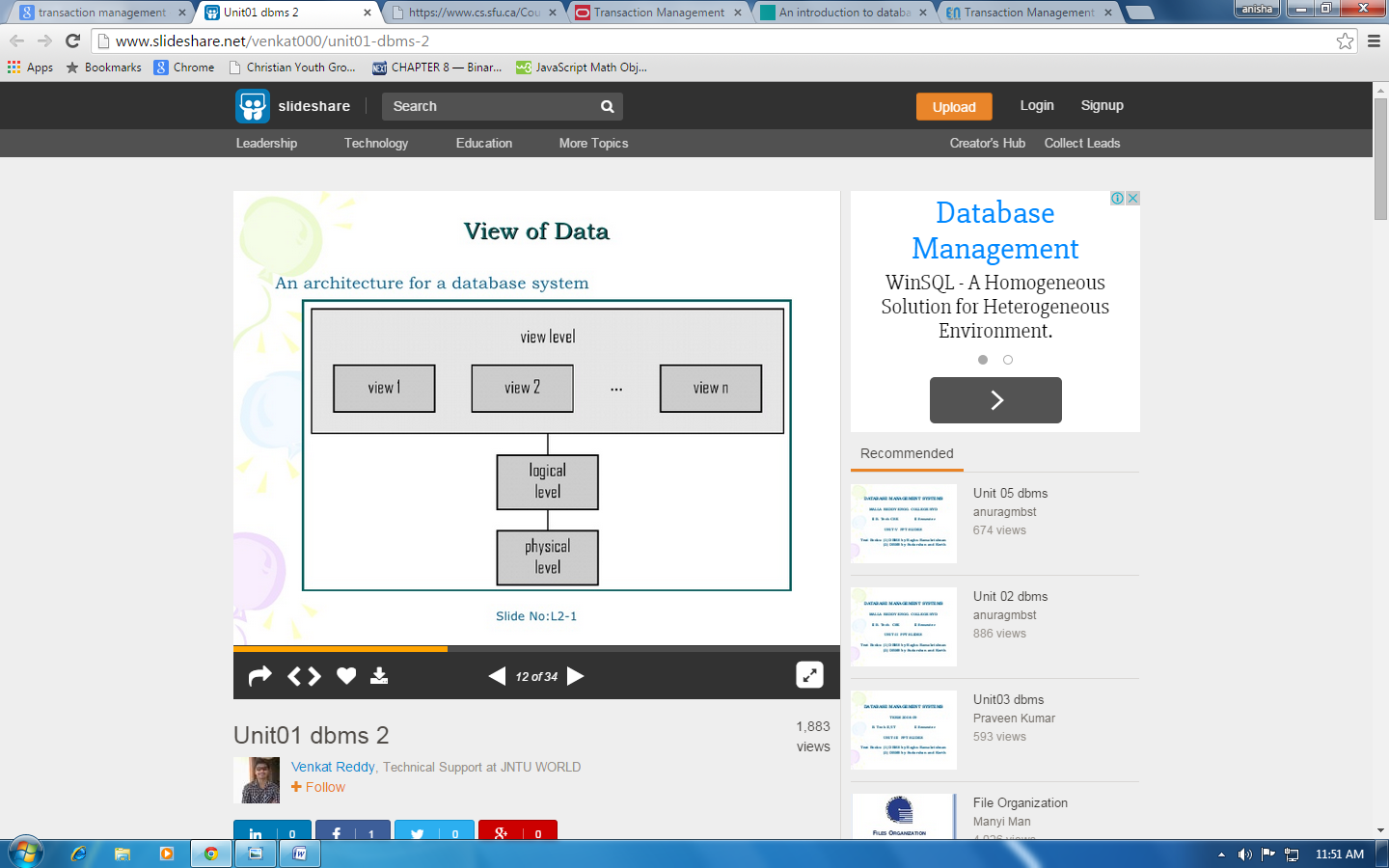
**The physical database**: the collection of files that contain the data

**The database engine**: the software that makes it possible to access and modify the contents of the database

**The database scheme**: the specification of the logical structure of the data stored in the database

**View of Data**

In a database management system, a view is a way of portraying information in the database. This can be done by arranging the data items in a specific order, by highlighting certain items, or by showing only certain items. For any database, there are a number of possible views that may be specified. Databases with many items tend to have more possible views than databases with few items. Often thought of as a virtual table, the view doesn't actually store information itself, but just pulls it out of one or more existing tables. Although impermanent, a view may be accessed repeatedly by storing its criteria in a query.



The best view for a particular purpose depends on the information the user needs. For example, in a telephone directory, a user might want to look up the name associated with a number, without concern for the street address.

**Database languages**

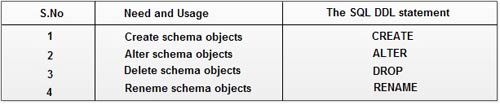
A DBMS must provide appropriate languages and interfaces for each category of users to express database queries and updates. Database Languages are used to create and maintain database on computer. There are large numbers of database languages like Oracle, MySQL, MS Access, dBase, FoxPro etc. SQL statements commonly used in Oracle and MS Access can be categorized as data definition language (DDL), data control language (DCL) and data manipulation language (DML).

1. **Data Definition Language (DDL)**

It is a language that allows the users to define data and their relationship to other types of data. It is mainly used to create files, databases, data dictionary and tables within databases.

It is also used to specify the structure of each table, set of associated values with each attribute, integrity constraints, security and authorization information for each table and physical storage structure of each table on disk.

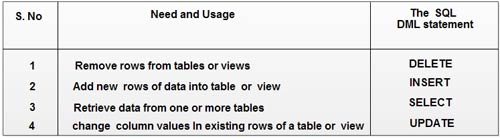
The following table gives an overview about usage of DDL statements in SQL:



1. **Data Manipulation Language (DML)**

It is a language that provides a set of operations to support the basic data manipulation operations on the data held in the databases. It allows users to insert, update, delete and retrieve data from the database. The part of DML that involves data retrieval is called a query language.

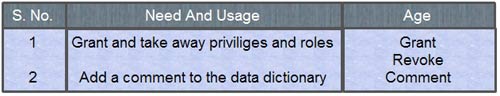
The following table gives an overview about the usage of DML statements in SQL:



1. **Data Control Language (DCL)**

DCL statements control access to data and the database using statements such as GRANT and REVOKE. A privilege can either be granted to a User with the help of GRANT statement. The privileges assigned can be SELECT, ALTER, DELETE, EXECUTE, INSERT, INDEX etc. In addition to granting of privileges, you can also revoke (taken back) it by using REVOKE command.

The following table gives an overview about the usage of DCL statements in SQL:



**Relational Database**

Computer database in which all data is stored in Relations which (to the user) are tables with rows and columns are relational database. Each table is composed of records (called Tuples) and each record is identified by a field (attribute) containing a unique value. Every table shares at least one field with another table in 'one to one,' 'one to many,' or 'many to many' relationships. These relationships allow the database user to access the data in almost an unlimited number of ways, and to combine the tables as building blocks to create complex and very large databases.

Relational databases are created using a special computer language, structured query language (SQL), which is the standard for database interoperability. SQL is the foundation for all of the popular database applications available today, from Access to Oracle.

**Database designs**

Database design is the process of producing a detailed data model of a database. This logical data model contains all the needed logical and physical design choices and physical storage parameters needed to generate a design in a data definition language, which can then be used to create a database. A fully attributed data model contains detailed attributes for each entity.

The term database design can be used to describe many different parts of the design of an overall database system. Principally, and most correctly, it can be thought of as the logical design of the base data structures used to store the data. In the relational model these are the tables and view. In an object database the entities and relationships map directly to object classes and named relationships. However, the term database design could also be used to apply to the overall process of designing, not just the base data structures, but also the forms and queries used as part of the overall database application within the database management system (DBMS).

The process of doing database design generally consists of a number of steps which will be carried out by the database designer. Usually, the designer must:

Determine the relationships between the different data elements.

Superimpose a logical structure upon the data on the basis of these relationships.

**Object based and semi structured database**

**Object DATABASE OR Object oriented database management system** is a database in which the information is represented in form of object as used in object-oriented programming. It is different from rational database. This type of database is used when there is complex data or/and multiple data relationships. It have a many-to-many object relationship. It should not be used when there are few join tables and there are large volume of simple transaction data.

It works well with the following application:

--> Multimedia Application.

--> CAS Application

**Features of Object Oriented Database:**

1. It support transactions.
2. It supply querying in bulk data.
3. Concurrent Access
4. Security

In **Semi-Structured Database** the data are in the form of structured data that edoes not conform with the formal structure of data models associated with rational databases or other form of data. Therefore, it is also known as self-describing structure.

**Types of Semi-Structured Database:**

XML semi-structured database

JSON (JavaScript Object Notation)semi-structured database

**Advantages of Semi-Structured Database**

It can show the information of data source that is not constrained by schema.

It is used to view structured data as semi-structured data.

The data transfer format may be portable.

Data storage and querying

1. Storage management
2. Query processing
3. Transaction processing

**Storage management**

Storage manager is a program module that provides the interface between the low level data stored in the database and the application programs and queries submitted to the system.

The storage manager is responsible to the following tasks:

~interaction with the file manager

~efficient storing, retrieving and updating of data

**Issue**

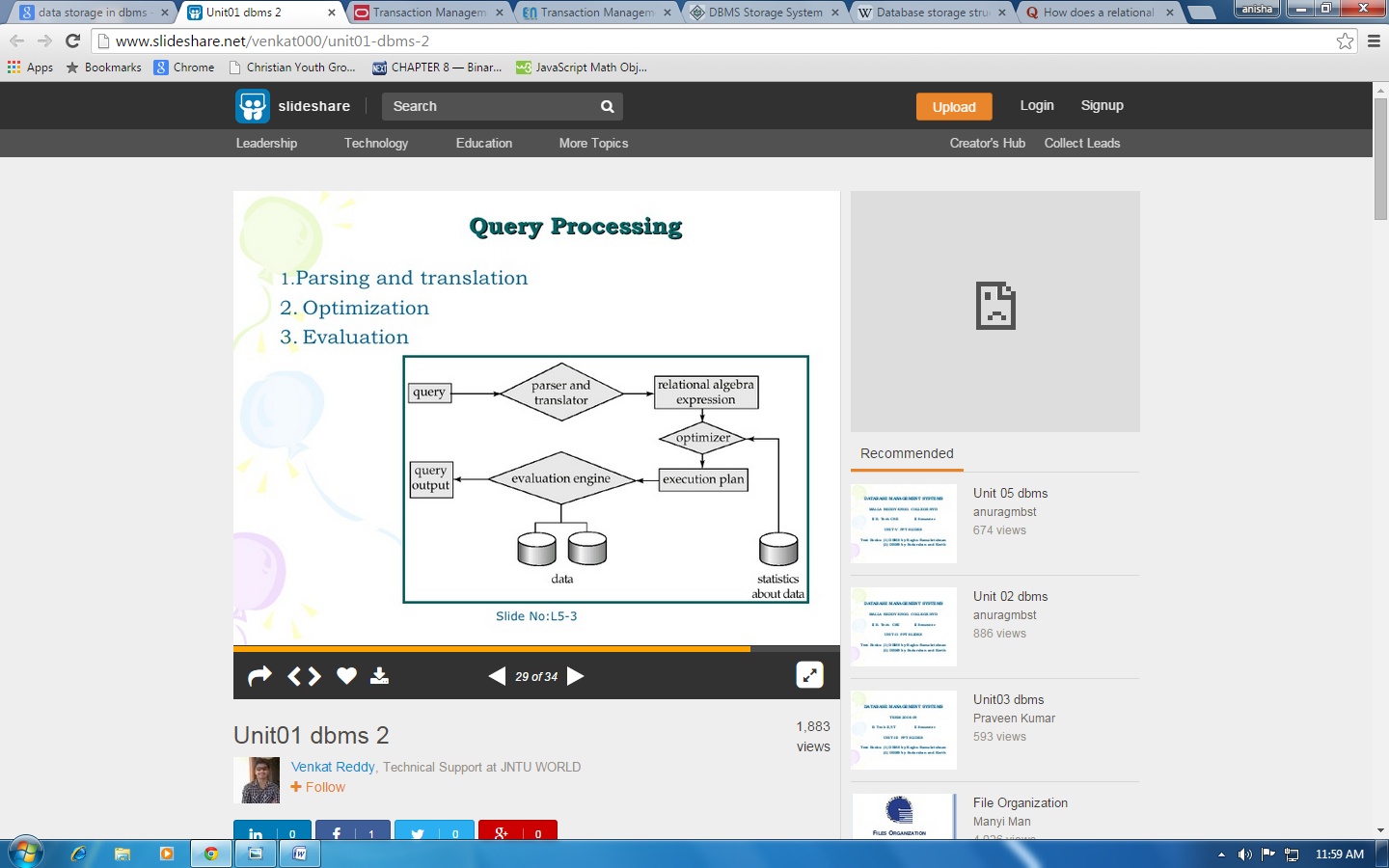
~storage access

~file organization

~indexing and hashing

**Query Processing**

1. Parsing and translation
2. Optimization
3. Evaluation



Alternative ways of evaluating a given query

~equivalent expressions

~different algorithms for each operation

Cost difference between a good and a bad way of evaluating a query can be enormous

Need to estimate the cost of operations

~depends critically on statistical information about relations which the database must maintain

~need to estimate statistics for intermediate results to compute cost of complex expressions

**Transaction Management**

A transaction is a logical unit of work that contains one or more SQL statements. A transaction is an atomic unit. The effects of all the SQL statements in a transaction can be either all committed (applied to the database) or all rolled back (undone from the database).

A transaction begins with the first executable SQL statement. A transaction ends when it is committed or rolled back, either explicitly with a COMMIT or ROLLBACK statement or implicitly when a DDL statement is issued.

To illustrate the concept of a transaction, consider a banking database. When a bank customer transfers money from a savings account to a checking account, the transaction can consist of three separate operations:

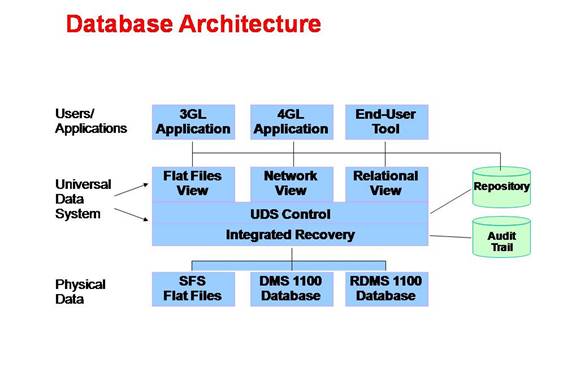
~Decrement the savings account

~Increment the checking account

~Record the transaction in the transaction journal

**Database Architecture**

Database architecture focuses on the design, development, implementation and maintenance of computer programs that store and organize information for businesses, agencies and institutions. A database architect develops and implements software to meet the needs of users. Several types of databases, including relational or multimedia, may be created. Additionally, database architects may use one of several languages to create databases, such as structured query language (SQL). You might be familiar with Oracle and Microsoft Access, which are two common database programs.



Database users and administrators

**A database administrator (DBA**) directs or performs all activities related to maintaining a successful database environment. Responsibilities include designing, implementing, and maintaining the database system; establishing policies and procedures pertaining to the management, security, maintenance, and use of the database management system; and training employees in database management and use. A DBA is expected to stay abreast of emerging technologies and new design approaches. Typically, a DBA has either a degree in Computer Science and some on-the-job training with a particular database product or more extensive experience with a range of database products. A DBA is usually expected to have experience with one or more of the major database management products, such as Structured Query Language, SAP, and Oracle-based database management software.

The **database user** is the identity of the login when it is connected to a database. The database user can use the same name as the login, but that is not required. This topic assumes that a login already exists in SQL Server. For information about how to create a login, see Create a Login.

**Structure Of Dbms**

DBMS (Database Management System) acts as an interface between the user and the database. The user requests the DBMS to perform various operations (insert, delete, update and retrieval) on the database. The components of DBMS perform these requested operations on the database and provide necessary data to the users. The various components of DBMS are shown below: -

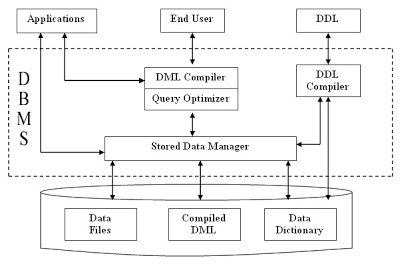


Fig. 2.1 Structure Of DBMS

1. **DDL Compiler** - Data Description Language compiler processes schema definitions specified in the DDL. It includes metadata information such as the name of the files, data items, storage details of each file, mapping information and constraints etc.

2. **DML Compiler and Query optimizer** - The DML commands such as insert, update, delete, retrieve from the application program are sent to the DML compiler for compilation into object code for database access. The object code is then optimized in the best way to execute a query by the query optimizer and then send to the data manager.

3**. Data Manager** - The Data Manager is the central software component of the DBMS also knows as Database Control System.

**History of Database systems**

There are both technical and business aspects of database technology to discuss here. The birth and the needs of a separate database system had both a business and technical reasons. Among the first official software standards, beyond programming languages, were different database standards of the 1960’s. The 1970’s saw the emergence of the relational model, and the mathematical and theoretical background here is both interesting and if not unique so at least uncommon.

The development of SQL databases during the late 1970’s and early 1980 changed the world of database systems completely, and together with other technology changes of those days, such as the emergence of standardized operation systems (most prominently Unix) and the Personal Computer, would transform the climate for competing completely.

As the storage capacities of even small computers grew, the potential and uses for databases grew. The PC oriented database systems, simple and working well on small PCs, largely disapperad as PCs became more powerful and above all networked.

The 1990 saw the emergence of the Internet, and again the means and needs for accessing data changed, as so did the database systems. Those years also saw data warehousing grow to something everyone could potentially use and benefit from.

Over many of these periods, SQL has managed to stay put longer than most other technologies. Something that grew out of mainframes 40 years ago is still with us. This is a fact that should also be debated.

**Reference**

<http://en.oreilly.com/mysql2010/public/schedule/detail/13607>

<https://www.google.com.np/?gws_rd=cr,ssl&ei=QM2xVYvLN6GOmwWP0a-ADg#q=history+of+database+systems>

<https://en.wikipedia.org/wiki/Database>

<https://www.google.com.np/?gws_rd=cr&ei=4NuxVa-wOIXH0ATDxpuICQ>