Database Management Systems
(DBMS)

(Database Model)

For

BCA Part-III (Session 2017-20) Students

BY

ANANT KUMAR
MCA, M. Phil, M. Tech.
Faculty Member
Department of Computer Science
J. D. Women's College, Patna

Database Management Systems

Role of the Database Administrator

The Database Administrator (DBA) who is like the super-user of the system. The role of the DBA is very important and is defined by the following functions.

- **Defining the Schema**

  The DBA defines the schema which contains the structure of the data in the application. The DBA determines what data needs to be present in the system ad how this data has to be represented and organized.

- **Liaising with Users**

  The DBA needs to interact continuously with the users to understand the data in the system and its use.

- **Defining Security & Integrity Checks**

  The DBA finds about the access restrictions to be defined and defines security checks accordingly. Data Integrity checks are also defined by the DBA.

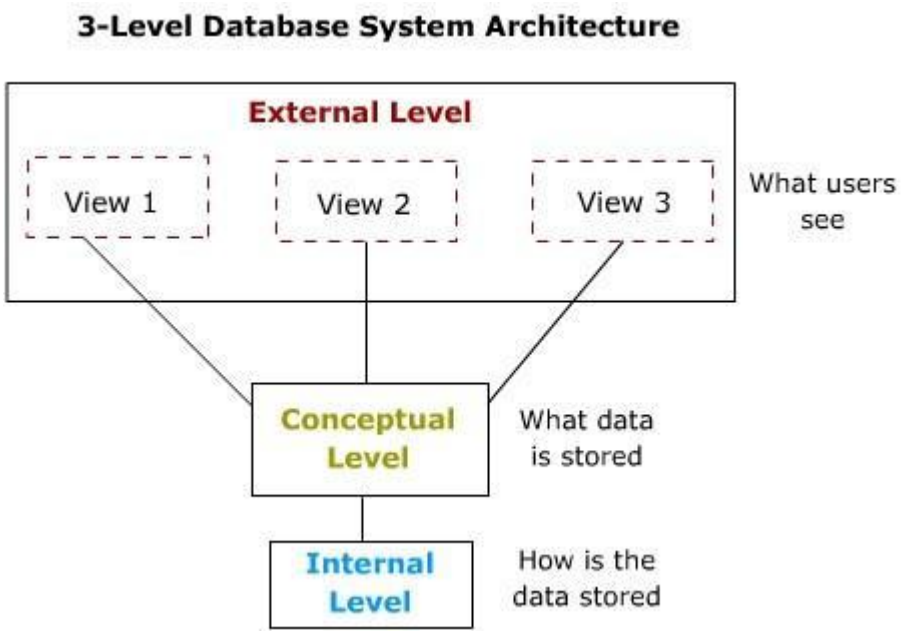
- **Defining Backup / Recovery Procedures**

  The DBA also defines procedures for backup and recovery. Defining backup procedures includes specifying what data is to backed up, the periodicity of taking backups and also the medium and storage place for the backup data.

- **Monitoring Performance**

  The DBA has to continuously monitor the performance of the queries and take measures to optimize all the queries in the application.

3-Level Database System Architecture



External Level
The External Level represents the collection of views available to different end-users.

Conceptual level
The conceptual level to describe the structure and constraints for the *whole* database for a community of users. Uses a *conceptual* or an *implementation* data model. (describes data stored in database, and the relationships among the data – Logical level)

Internal schema at the internal level to describe physical storage structures and access paths. Typically uses a *physical* data model.(how a record (e.g., customer) is stored-Physical level)

Schema

Schema – describes contents of the database

e.g., what information about a set of customers and accounts and the relationship between them)

Physical schema: how data is stored at physical level (how)
Logical schema: data contained at the logical level (what)

Data Independence

When a schema at a lower level is changed, only the **mappings** between this schema and higher-level schemas need to be changed in a DBMS that fully supports data independence. The higher-level schemas themselves are *unchanged*. Hence, the application programs need not be changed since they refer to the external schemas

Physical Data Independence – the ability to modify the physical schema without changing the logical schema

- Applications depend on the logical schema
- In general, the interfaces between the various levels and components should be well defined so that changes in some parts do not seriously influence others.

Logical Data Independence – the ability to modify conceptual schema without changing the external Schema or application programs.

Database model

A Database model defines the logical design and structure of a database and defines how data will be stored, accessed and updated in a database management system. Database model are:

- Hierarchical Model
- Network Model
- Entity-relationship Model
- Relational Model
- Object-oriented Data Model

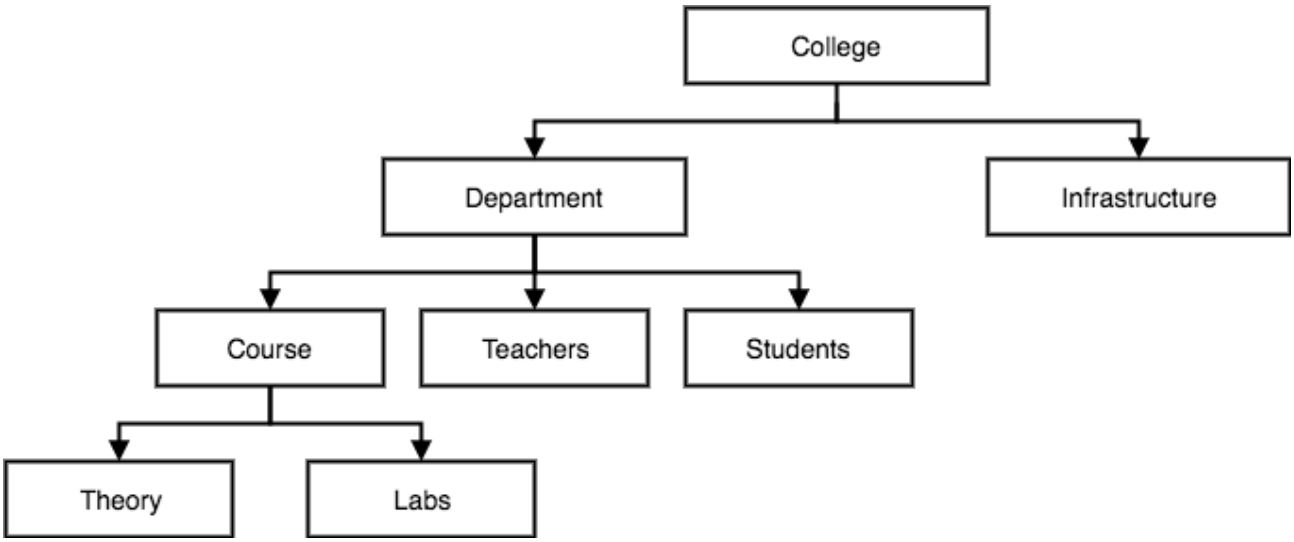
Hierarchical Model

This database model organises data into a tree-like-structure, with a single root, to which all the other data is linked. The heirarchy starts from the **Root** data, and expands like a tree, adding child nodes to the parent nodes.

In this model, a child node will only have a single parent node.

This model efficiently describes many real-world relationships like index of a book, recipes etc.

In hierarchical model, data is organised into tree-like structure with one one-to-many relationship between two different types of data, for example, one department can have many courses, many professors and of-course many students.

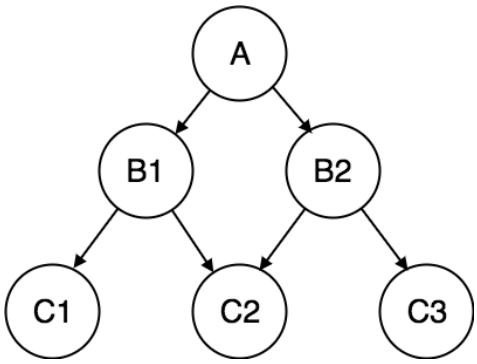


Network Model

This is an extension of the Hierarchical model. In this model data is organised more like a graph, and are allowed to have more than one parent node.

In this database model data is more related as more relationships are established in this database model. Also, as the data is more related, hence accessing the data is also easier and fast. This database model was used to map many-to-many data relationships.

This was the most widely used database model, before Relational Model was introduced.



Entity-relationship Model

In this database model, relationships are created by dividing object of interest into entity and its characteristics into attributes.

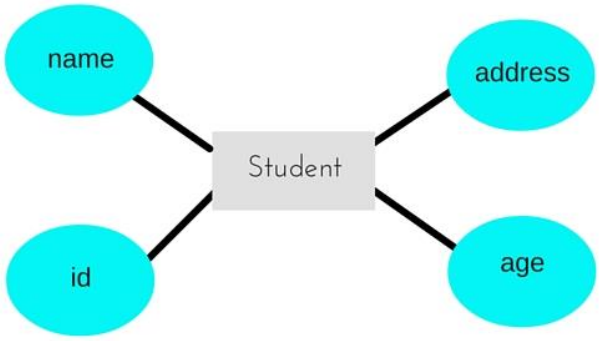
Different entities are related using relationships.

E-R Models are defined to represent the relationships into pictorial form to make it easier for different stakeholders to understand.

This model is good to design a database, which can then be turned into tables in relational model(explained below).

Let's take an example, If we have to design a School Database, then **Student** will be an **entity** with **attributes** name, age, address etc. As **Address** is generally complex, it can be another **entity** with **attributes** street name, pincode, city etc, and there will be a relationship between them.

Relationships can also be of different types.



Relational Model

In this model, data is organised in two-dimensional **tables** and the relationship is maintained by storing a common field.

This model was introduced by E.F Codd in 1970, and since then it has been the most widely used database model, infact, we can say the only database model used around the world.

The basic structure of data in the relational model is tables. All the information related to a particular type is stored in rows of that table.

Hence, tables are also known as **relations** in relational model.

In the coming tutorials we will learn how to design tables, normalize them to reduce data redundancy and how to use Structured Query language to access data from tables.

student_id	name	age
1	Akon	17
2	Bkon	18
3	Ckon	17
4	Dkon	18

subject_id	name	teacher
1	Java	Mr. J
2	C++	Miss C
3	C#	Mr. C Hash
4	Php	Mr. P H P

student_id	subject_id	marks
1	1	98
1	2	78
2	1	76
3	2	88

Object oriented data model

Object oriented data model is based upon real world situations. These situations are represented as objects, with different attributes. All these object have multiple relationships between them.

Elements of Object oriented data model:

Objects

The real world entities and situations are represented as objects in the Object oriented database model.

Attributes and Method

Every object has certain characteristics. These are represented using Attributes. The behaviour of the objects is represented using Methods.

Class

Similar attributes and methods are grouped together using a class. An object can be called as an instance of the class.

Inheritance

A new class can be derived from the original class. The derived class contains attributes and methods of the original class as well as its own.