



**Al-Mustaqbal University**

**College of Sciences**

**Cyber Security Department**



جامعة المستقبل  
AL MUSTAQBAL UNIVERSITY

كلية العلوم  
قسم الامن السيبراني

## Lecture: (4)

**Subject: Database Systems**

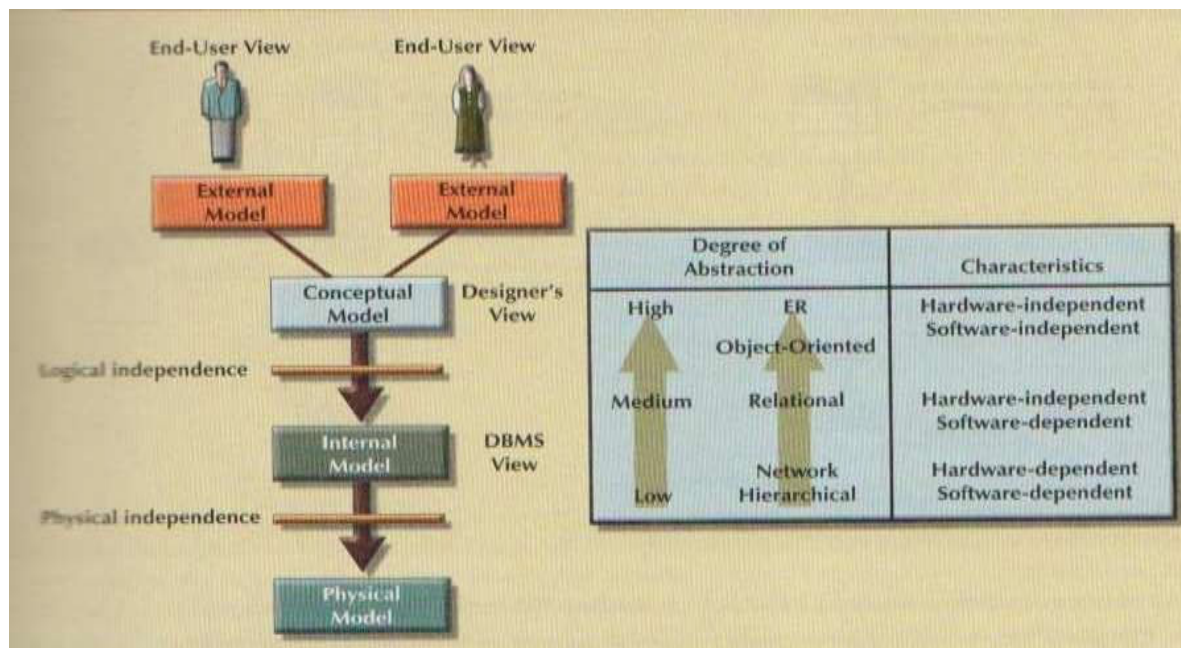
**Level: Second**

**Lecturer: Asst. Lecturer Qusai AL-Durrah**

## Degrees of Data Abstraction

If you ask ten data base designers what is a data model, you will end up with ten different answers. Depending on the degree of data abstraction.

In 1970s the American National standards Institute (ANSI) defined a framework for data modeling based on degrees of data abstraction.



## The External Model

It is the end users view of the data environment. the term end users refers to people who use the application programs to manipulate the data and generate information.

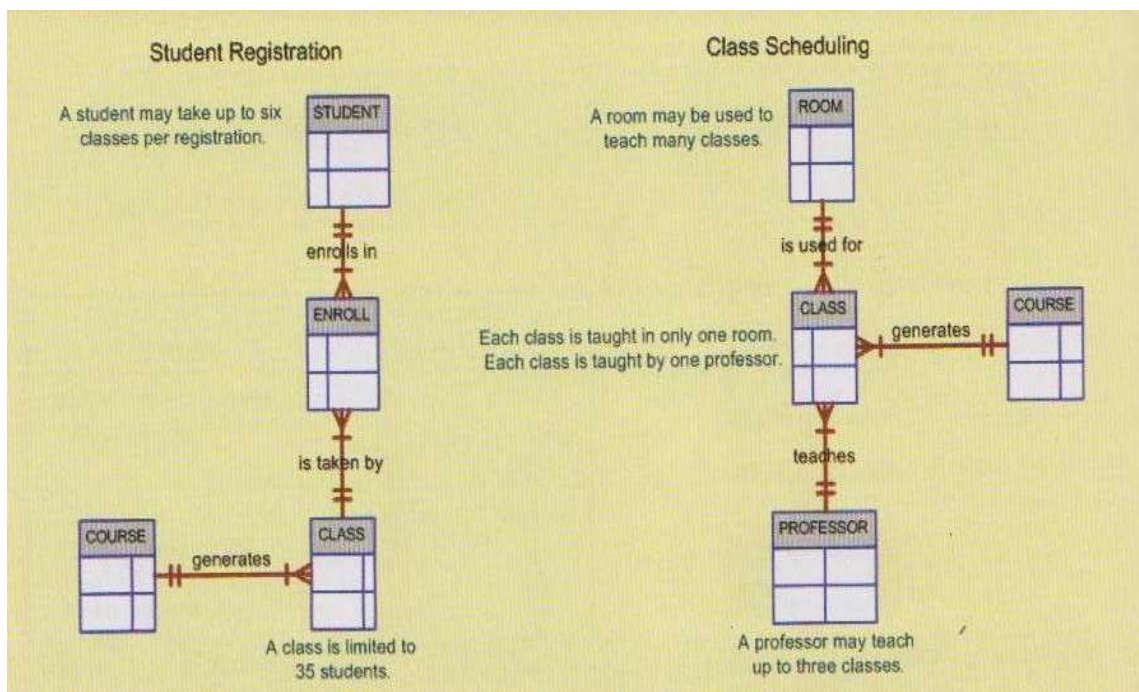
Because data is being modeled, ER or OO diagrams will be used to represent the external views. A specific representation of an external view is known as external schema.

The use of external views representing subsets of the database have some important advantages:

- It makes it easy to identify specific data required to support each business unit's operations.



- It makes the designer's job easy by providing feedback about the model. Specifically, the model can be checked to ensure that it supports all processes as defined by their external models, as well as all operational requirements and constraints.
- It helps to ensure security constraints in the data base design. Damaging an entire data base is more difficult when each business unit works with only a subset of data.
- It makes applications program development much simpler.



## The conceptual model

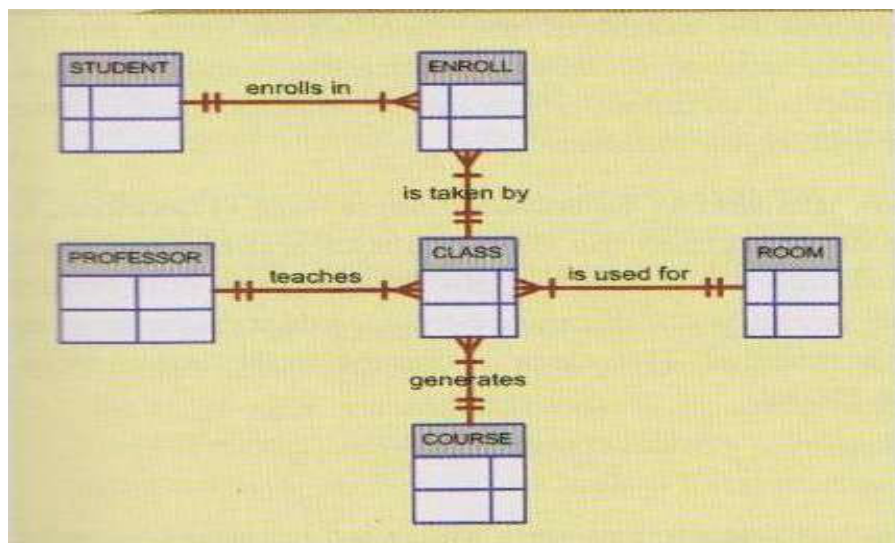
It represents a global view of the entire database as viewed by entire organization. That is, the conceptual model integrates all external views (entities, relationships, constraints, and processes) into a single global view of the entire data in the enterprise. Also known as a conceptual schema, it is the basis for identification and high – level description of the main data objects (avoiding any database model-specific details). The most widely used conceptual model is the ER model.

The advantage is:



- It provides a relatively easily understood view of the data environment.
- It is independent of both software and hardware. software independence means that the model does not depend on the DBMS software used to implement the model. hardware independence means that the model does not depend on the hardware used in the implementation of the model. therefore, changes in either the hardware or the DBMS software will have no effect on the db designer at the conceptual level.

Logical design is used to refer to the task of creating conceptual data model.



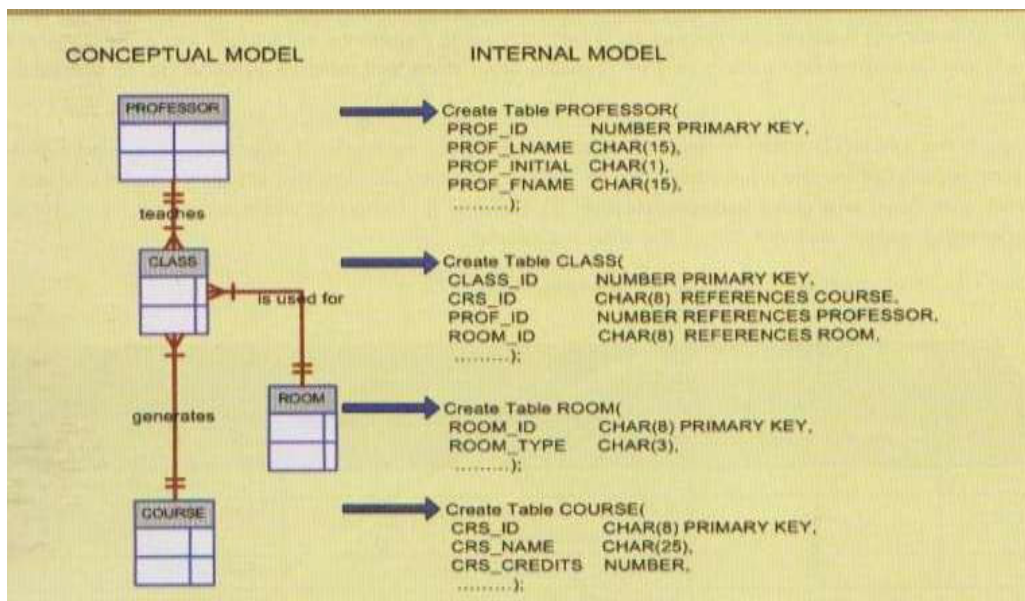
## The Internal Model

It is mapping the conceptual model to the DBMS. The internal model is the representation of the database as "seen" by the DBMS also it called the internal schema.

The relational model is used to implement internal model. the internal schema should map the conceptual model to the relational model constructs, the entities in the conceptual model are mapped to tables in the relational model.

Because the internal model depends on specific database software, it is said to be software-dependent.

It is logical independence, because you can change the internal model without affecting the conceptual model, it is hardware-independent because it is unaffected by the choice of computer on which the software is installed.



## The physical model

It is operating at the lowest level of abstraction, describing the way data are saved on storage media such as disks or tapes. The physical model requires the definition of both the physical storage devices and the access methods required to reach the data within those storages devices, making it both software-and hardware-dependent.

This model is physical independence, any change in storage devices or methods and even a change in operating system with not affect the internal model.

The responsibility regarding physical data model creation usually lies with database administrators and developers. Information systems and software applications heavily rely on interactions with physical databases. Physical data models need to be designed and implemented correctly. It is challenging to





modify physical data models once data from the existing application has been inserted into databases.

Below is a comparison between the logical and physical views:

