Data Models

Data Modeling and Data Models

- A data model is a relatively simple representation, usually graphical, of more complex real-world data structures.
- A model is an abstraction of a more complex real-world object or event.
- In a database environment, a data model represents data structures and their characteristics, relations, constraints, transformations, and other constructs with the purpose of supporting a specific problem domain.
- Data modeling, refers to the process of creating a specific data model for a determined problem domain - the first step in designing a database
- Data modeling is an iterative, progressive process.

Basic Building Blocks of Data Model

- The basic building blocks of all data models are *entities, attributes, relationships, and constraints.*
- An ***entity*** is anything (a person, a place, a thing, or an event) about which data are to be collected and stored.
  - May be physical objects, such as customers or products, but entities may also be abstractions, such as flight routes or musical concerts.
- An ***attribute*** is a characteristic of an entity.
- A ***relationship*** describes an association among entities.

- Three types of relationships:
  - *One-to-many* (1:M or 1..*) – one instance of an entity is allowed to be associated with several instances of the other entity.
  - *Many-to-many* (M:N or *..*) – several instances of an entity is allowed to be associated with several instances of the other entity. Though allowed but discouraged.
  - *One-to-one* (1:1 or 1..1) – one instance of an entity is allowed to be associated with one and only one instance of the other entity.

- Relationships are bidirectional, that is, interpretation can be from left to right or right to left.
- Example:
  - Many STUDENTs can enroll many CLASSes.
  - Many CLASSes are enrolled by many STUDENTs.

- ***Constraint*** is a restriction imposed on data.
  - Ensures data integrity
  - Normally expressed as rules.
  - Example: the quantity on hand in the inventory must be a natural number.

Business Rules

- Business rules describe the operations, definitions and constraints that apply to an organization.
- Sources of business rules:
  - Company executives
  - Policy makers
  - Department managers
  - Written documentation – operating procedures, standards and operations manuals.
- Business rules might be conflicting – needs to be reconciled

- The database designer must:
  - Reconcile the differences
  - Verify the results of the reconciliation to ensure that the business rules are appropriate and accurate.
- Note: only business rules that constraints the transaction that may be processed against the database are considered.

Characteristics of a Good Business Rule

Characteristics	Explanation
Declarative	It is a statement of policy, not how the policy is enforced or conducted.
Precise	It must have only on interpretation among all the interested people and its meaning must be clear.
Atomic	One statement only; no part of the rule can stand on its own as a rule.
Expressible	Must be stated in a natural language but in a structured manner.
Distinct	Must not be redundant; can refer to other rules.
Consistent	Should not contain conflicting statement and should not contradict with other rules
Business-oriented	Stated in terms business people can understand; only business people can modify or invalidate – owned by them.

- Importance of business rules in the design of databases:
  - They help standardize the company's view of data.
  - They can be a communications tool between users and designers.
  - They allow the designer to understand the nature, role, and scope of the data.
  - They allow the designer to understand business processes.
  - They allow the designer to develop appropriate relationship participation rules and constraints and to create an accurate data model.
- Note: not all business rules can be modeled

Translating Business Rules into Data Model Components

- A noun in a business rule will translate into an entity in the model, and
- A verb (active or passive) associating nouns will translate into a relationship among the entities.
- Example: "Many students can enroll many classes."

The Evolution of Data Model

Assignment No 1

- Research on the evolution of data models (from the first generation up to the fourth generation).
- Submit a paper on this topic.
  - Must be on a standard short size bond paper.
  - Observe proper margins and include a cover page.
  - DO NOT COPY AND PASTE the text.
  - Include references at the end of your paper.
  - Due date is 6 days from the date this assignment is given.

Degrees of Data Abstraction

- Degrees of data abstraction dictates how detail the model is.
- The American National Standards institute (ANSI) Standards and Planning Committee defined the framework for the data modeling based on the degrees of data abstraction.
  - External
  - Conceptual
  - Internal
- See fig 1. Levels of abstraction

External Model

- The data model based on the end users' view of the data; usually focused on the business unit he/she belongs.
- Important advantages:
  - It makes it easy to identify specific data required to support each business unit's operation.
  - It makes the designer's job easy by providing feedback about the models' adequacy.
  - It helps to ensure security constraints in the database design.
  - It makes application program development easier.
- See figure 2. Example of an external model

Conceptual Model

- Done after the external views had been identified.
- All external views are integrated into a single view – enterprise wide.
- It is also known as a conceptual schema – basis for the identification and high-level description of the main data objects.
- The most widely used conceptual model is the ER model.
- It is both hardware and software independent.
- Logical design refers to the creation of a conceptual model that software independent.
- See fig 3. conceptual model

Internal Model

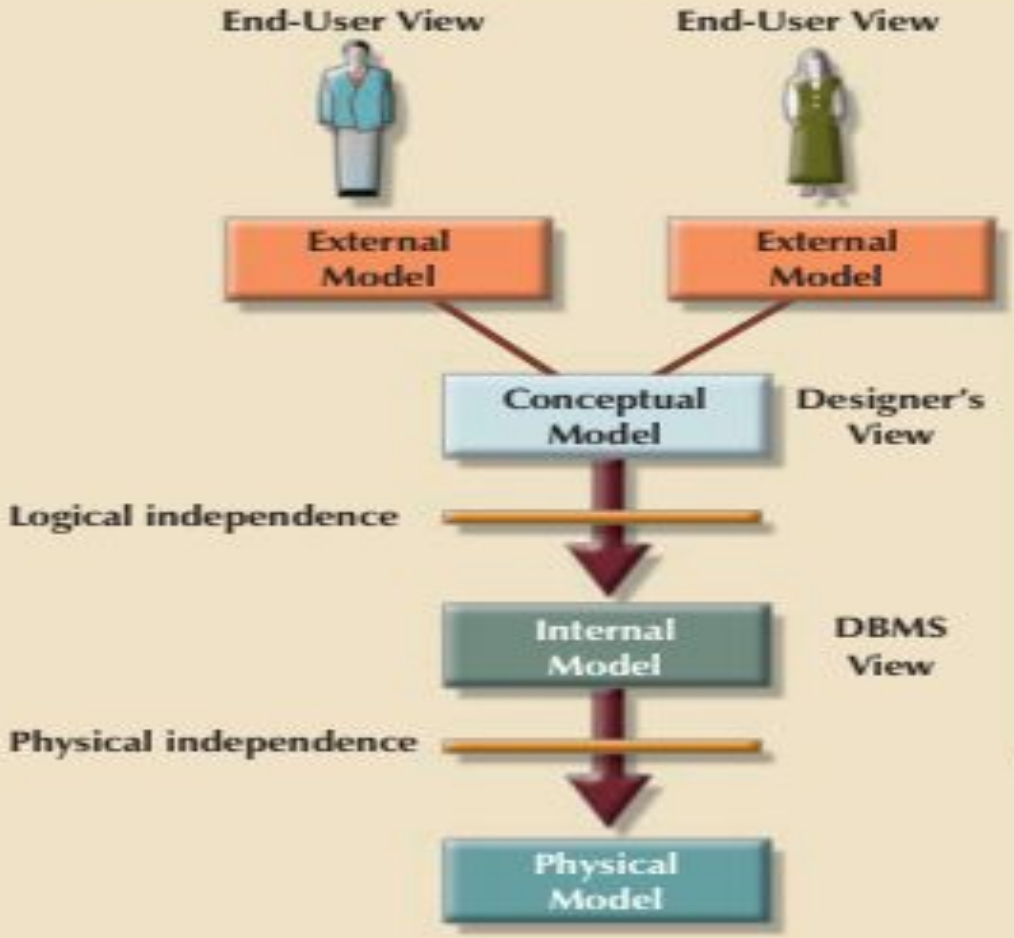
- Done once A DBMS is selected.
- Representation of the database as "seen" by the DBMS
- The conceptual model's characteristics and constraints are matched to those of the implementation model.
- The internal schema should map the conceptual model to the relational model constructs – the schema is expressed using SQL.
- See fig 4. An internal model

Physical Model

- The lowest level of abstraction, describing the way data are saved on storage media such as disks or tapes.
- Required for the early data models – however for the relational model, this level is no longer required.
- The physical model is dependent on the DBMS, methods of accessing files, and types of hardware storage devices supported by the operating system.

End of Presentation

Fig. 1: Data Abstraction Levels



Degree of Abstraction		Characteristics
High	ER	Hardware-independent Software-independent
	Object-Oriented	
Medium	Relational	Hardware-independent Software-dependent
Low	Network Hierarchical	Hardware-dependent Software-dependent

Fig 2: External Model

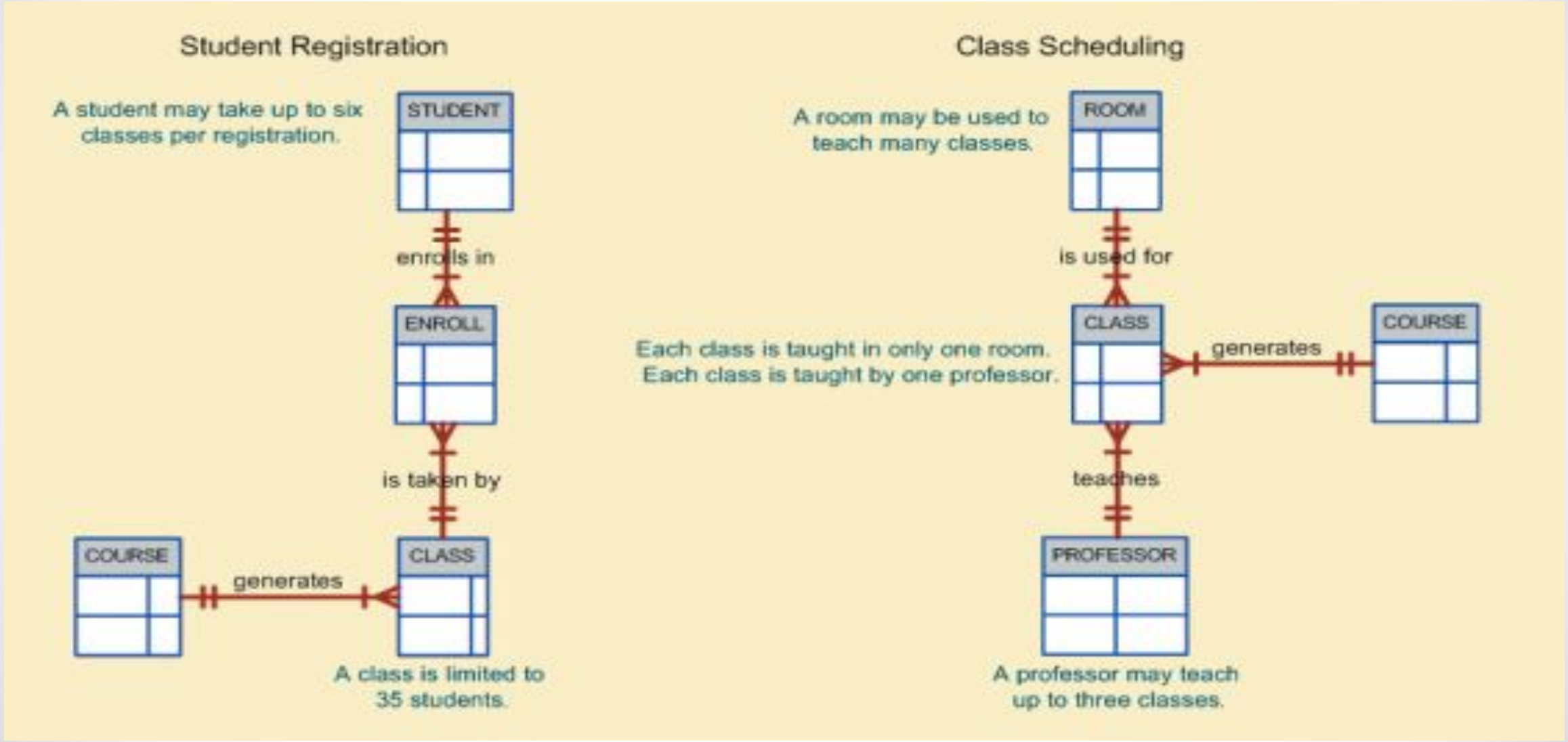


Fig 3. Conceptual model

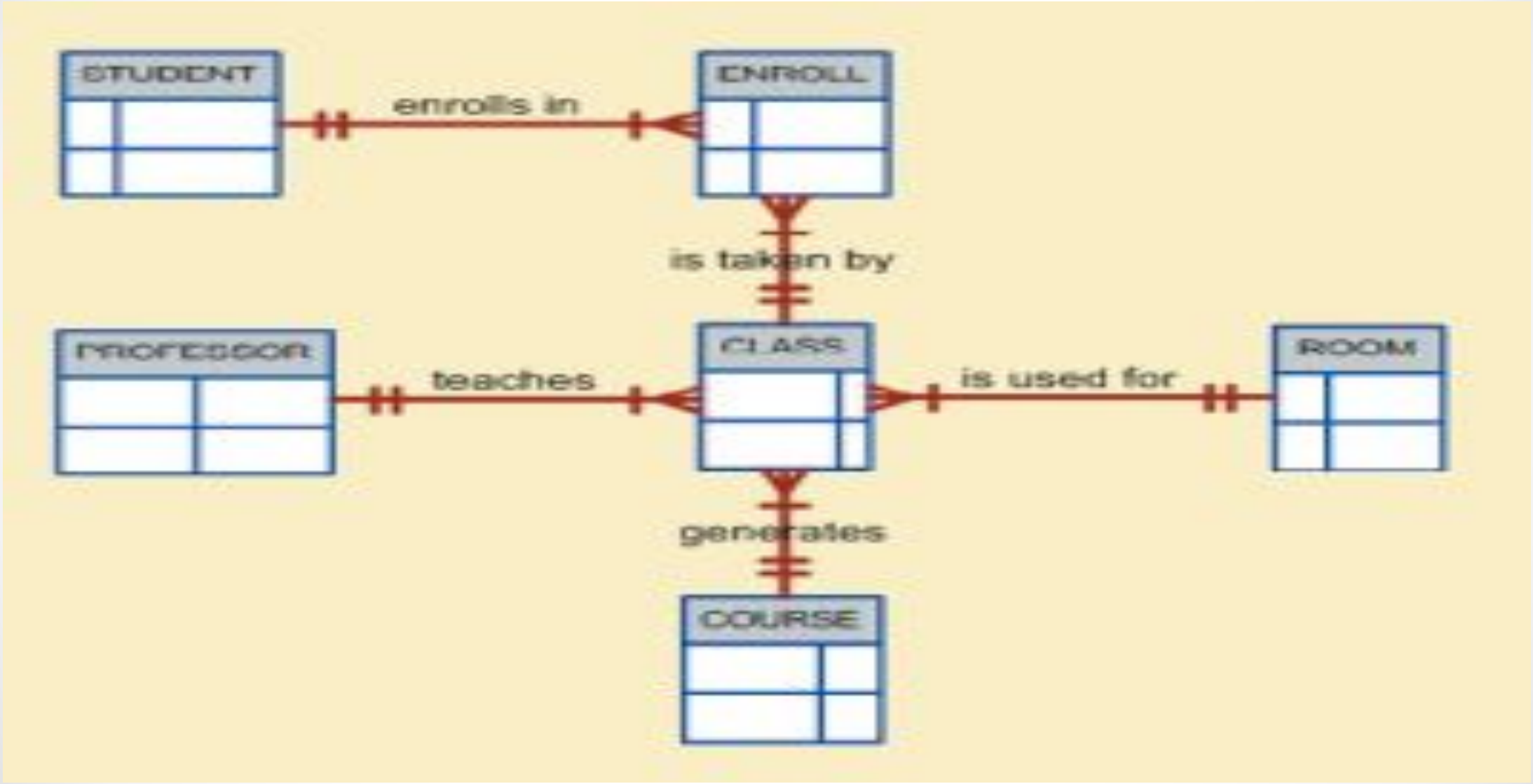
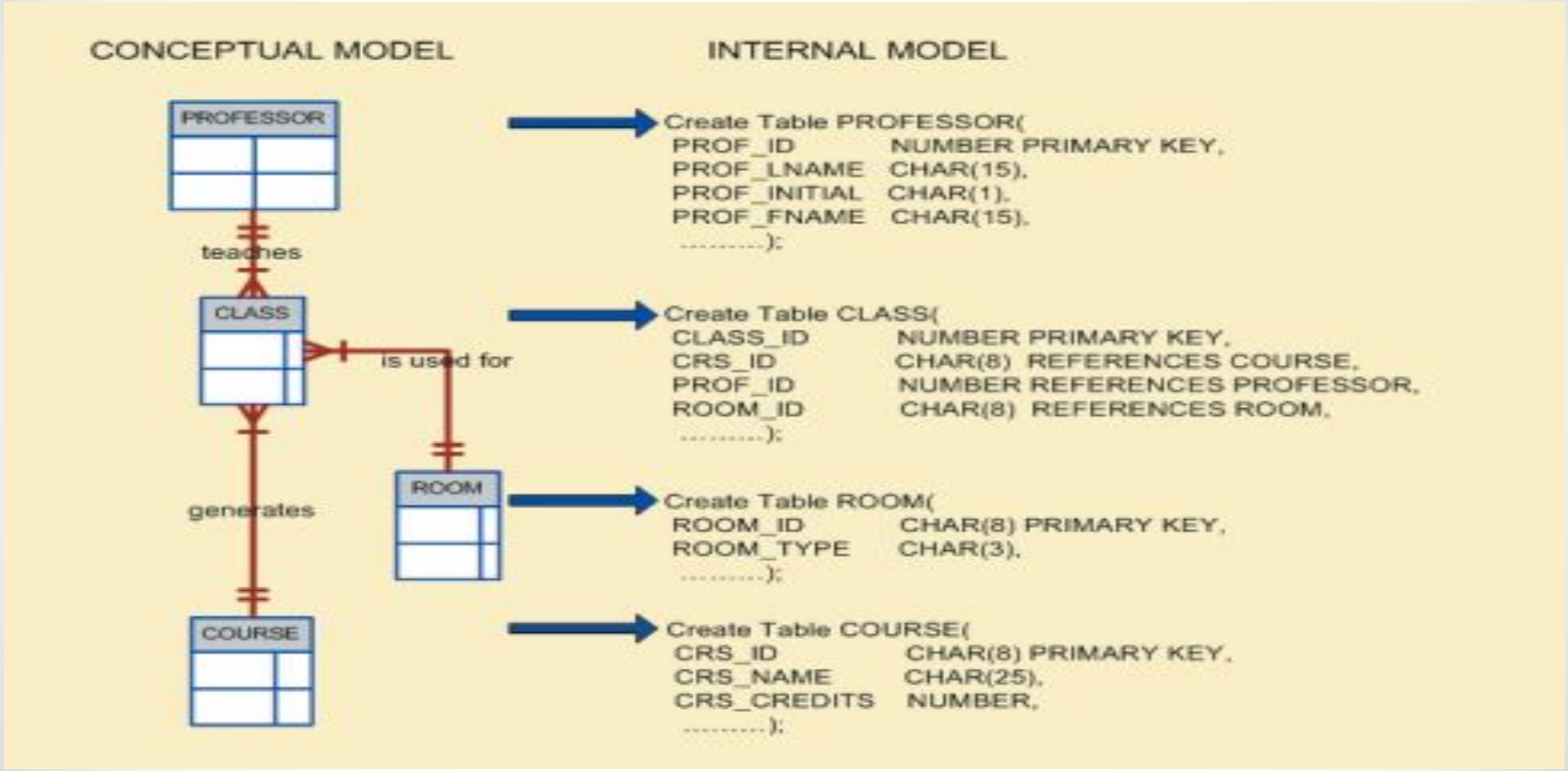


Fig 4: An Internal Model



Source:

Rob, Peter and Coronel, Carlos. *Database Systems : Design, Implementation and Management*, 7th Edition. Course Technology, Thomson Learning Inc. ©2007