WEEK 6: THE ENTIRY RELATIONSHIP MODEL (ERM)

The *ERM* forms the basis of an *Entity Relational Diagram (ERD)*.

- The ERD represents the conceptual database as viewed by the end user.
- ERDs depict the database's main components: entities, attributes, and relationships

Because an entity represents a real-world object, the words *entity* and *object* are often used interchangeably.

Thus, the entities (objects) of the ABC College database design include *students*, *classes*, *teachers*, and *classrooms*. The order in which the ERD components are covered is dictated by the way the modeling tools are used to develop ERDs that can form the basis for successful database design and implementation.

DEVELOPING AN ERD DIAGRAM

The process of database design is an *iterative* rather than a *linear* or *sequential* process. The verb *iterates* means "to do again or repeatedly."
An *iterative process* is, thus, one based on repetition of processes and procedures. Building an ERD usually involves the following activities:-

- Create a detailed narrative of the organization's description of operations.
- Identify the business rules based on the description of operations.
- Identify the main entities and relationships from the business rules.
- Develop the initial ERD.
- Identify the attributes and primary keys that adequately describe the entities.
- Revise and review the ERD.

During the review process, it is likely that additional objects, attributes, and relationships will be uncovered. Therefore, the basic ERM will be modified to incorporate the newly discovered ER components.

Subsequently, another round of reviews might yield additional components or clarification of the existing diagram. The process is repeated until the end users and designers agree that the *ERD* is a fair representation of the organization's activities and functions.

During the design process, the database designer does not depend simply on interviewing users to help define entities, attributes, and relationships. A surprising amount of information can be gathered by examining the business forms and reports that an organization uses in its daily operations.

THE PROCESS OF DEVELOPING AN ERD

To illustrate the use of the iterative process that ultimately yields a workable *ERD*; we can use the following school example:-

1. **ABC College (ABC) is divided into several schools:-**
    - A school of business
    - A school of arts and sciences
    - A school of education,
    - A school of applied sciences.

Each school is administered by a *dean* who is a *professor*.

Each professor can be the dean of only *one school*, and a professor doesn't have to be the dean of any school.

Therefore, a 1:1 relationship exists between *PROFESSOR* and *SCHOOL*. Note that the cardinality can be expressed by writing (1, 1) next to the entity PROFESSOR and (0, 1) next to the Entity SCHOOL.

2. **Each school comprises several departments.**

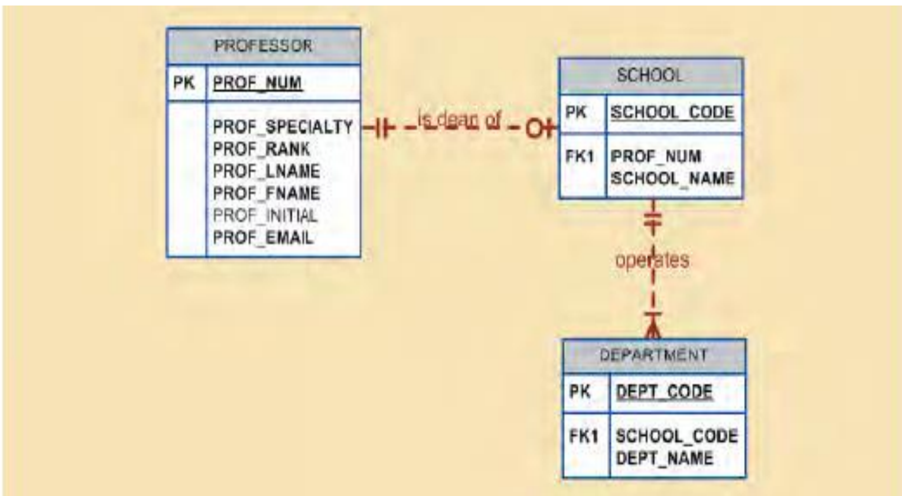
For example, the school of business has:-

- accounting department,
- management/marketing department,
- economics/finance department,

- Computer information systems department.

Note again the cardinality rules: The smallest number of departments operated by a school is *one*, and the largest number of departments is *undefined* (N). On the other hand, each department belongs to only a single school; thus, the cardinality is expressed by (1, 1). That is, the minimum number of schools that a department belongs to is one, as is the maximum number.

The figure below illustrates these *first two business rules*:-



NOTE

It is again appropriate to evaluate the reason for maintaining the 1:1 relationship between PROFESSOR and SCHOOL in the PROFESSOR is dean of SCHOOL relationship. It is worth repeating that the existence of 1:1 relationships often indicates a misidentification of attributes as entities. In this case, the 1:1 relationship could easily be eliminated by storing the dean's attributes in the SCHOOL entity.

This solution would also make it easier to answer the queries, "Who is the dean?" and "What are that dean's credentials?"

3. **Each department may offer courses.**

For example,

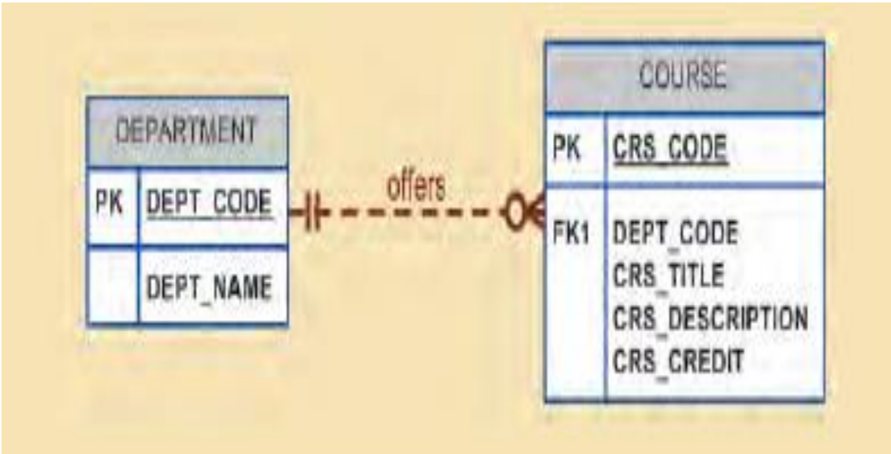
The management/marketing department offers courses such as:-

- Introduction to Management,
- Principles of Marketing,

– Production Management.

The **ERD** segment for this condition is shown in figure below:-

Note that this relationship is based on the way ABC College operates.

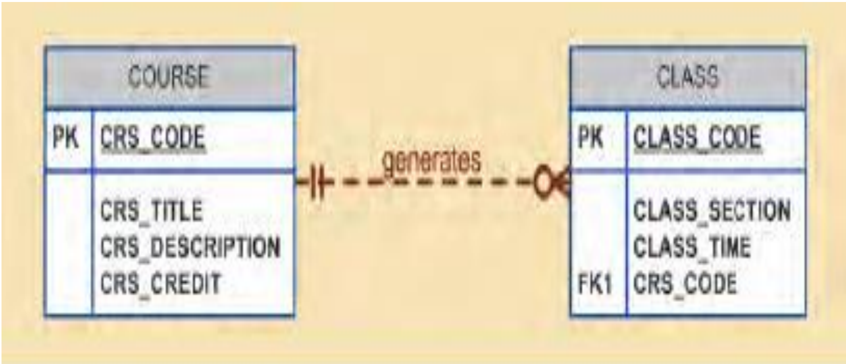


If, for example, **ABC** College had some departments that were classified as "research only," those departments would not offer courses; therefore, the COURSE entity would be optional to the DEPARTMENT entity.

4. **The relationship between COURSE and CLASS is illustrated above.**

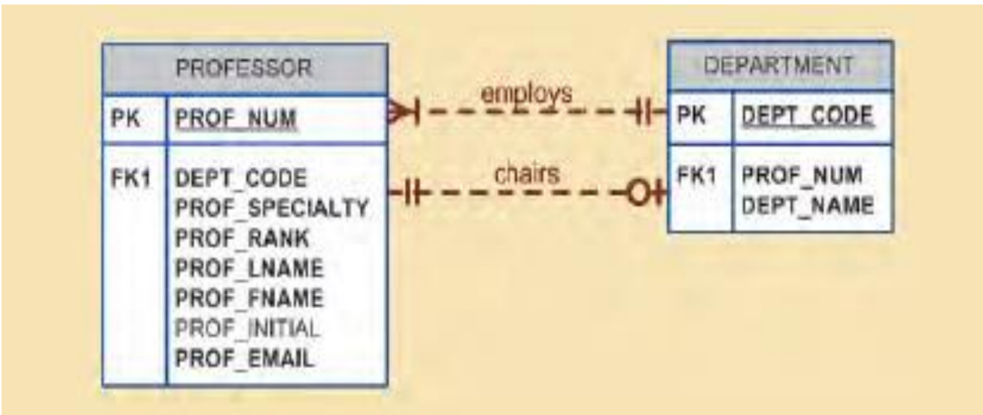
Nevertheless, it is worth repeating that a *CLASS* is a section of a *COURSE*. That is, a department may offer several sections (*classes*) of the same database course. Each of those *classes* is taught by a *professor* at a given time in a given place.

In short, a 1: M relationship exists between COURSE and CLASS. However, because a course may exist in ABC College's course catalog even when it is not offered as a class in a current class schedule, CLASS is optional to COURSE. Therefore, the relationship between COURSE and CLASS is as shown in the figure below



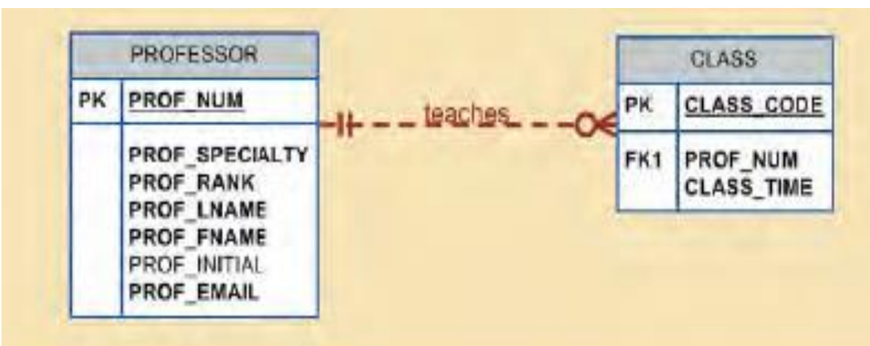
5. **Each department should have one or more professors assigned to it.**

One and only one of those professors *chairs* the department, and no professor is *mandated* to accept the chair position. Therefore, DEPARTMENT is optional to PROFESSOR in the "*chairs*" relationship. These relationships are as shown below:-



6. **Each professor may teach up to four classes; each class is a section of a course.**

A professor may also be on a research contract and teach no classes at all. The ERD below depicts these conditions.



7. **A student may enroll in several classes but takes each class only once during any given enrollment period.**

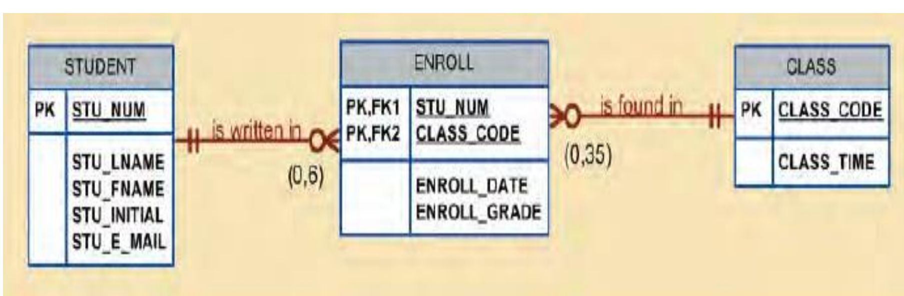
For example, during the current enrollment period, a student may decide to take five classes:-

- Statistics
- Accounting
- English
- Database
- History

Each student may enroll in up to six classes, and each class may have up to 35 students, thus creating an M: N relationship between STUDENT and CLASS. Because a CLASS can initially exist (at the start of the enrollment period) even though no students have enrolled in it, STUDENT is optional to CLASS in the M: N relationship.

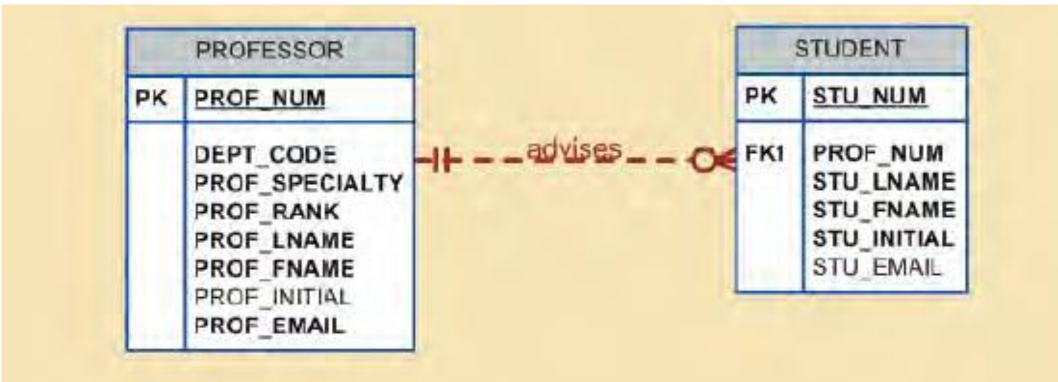
This M: N relationship must be divided into two 1: M relationships through the use of the ENROLL entity, shown in the ERD segment in the figure below.

But note that the optional symbol is shown next to ENROLL. If a class exists but has no students enrolled in it, that class doesn't occur in the ENROLL table. Note also that the ENROLL entity is weak: it is existence-dependent, and its (composite) PK is composed of the PKs of the STUDENT and CLASS entities.



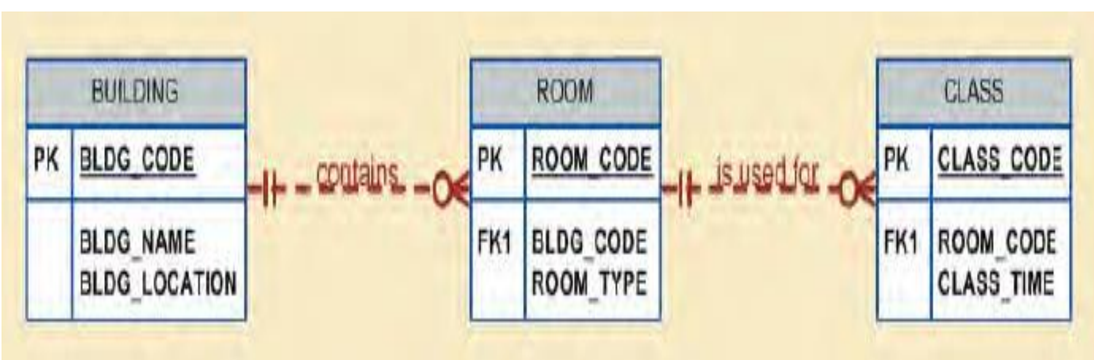
8. **Each student has an advisor in his or her department; each advisor counsels several students.**

An *advisor* is also a *professor*, but not all *professors* advise students. Therefore, STUDENT is optional to PROFESSOR in the "PROFESSOR advises STUDENT" relationship.



9. As you can see in Figure 4.34, the CLASS entity contains a ROOM_CODE attribute. Given the naming conventions, it is clear that ROOM_CODE is an FK to another entity. Clearly, because a *class* is taught in a *room*, it is reasonable to assume that the ROOM_CODE in CLASS is the FK to an entity named ROOM.

In turn, each room is located in a building. So the last ABC College ERD is created by observing that a BUILDING can contain many ROOMs, but each ROOM is found in a single BUILDING. In this ERD segment, it is clear that some buildings do not contain (class) rooms. For example, a storage building might not contain any named rooms at all.



Using the preceding summary, you can identify the following entities:

SCHOOL, COURSE, DEPARTMENT, CLASS, PROFESSOR, STUDENT

BUILDING, ROOM

ENROLL (the associative entity between STUDENT and CLASS)

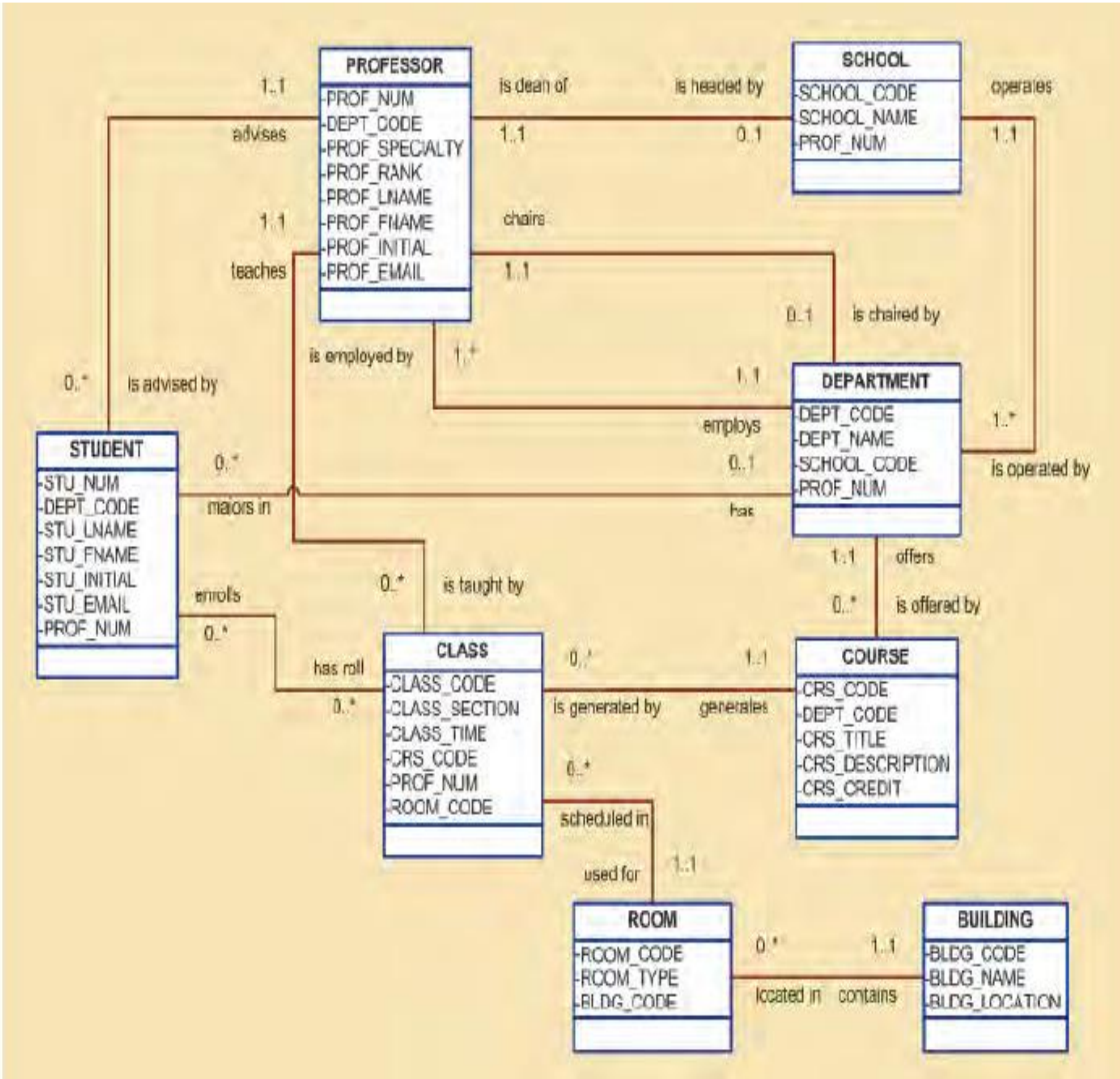
SUMMARY- STEPS FOLLOWED

- Discover the relevant entities,
- Define the initial set of relationships among them.
- Describe the entity attributes.
- Identifying the attributes of the entities helps you to better understand the relationships among entities.
- Define the connectivity and cardinality for the just-discovered relations based on the business rules.

The table below summarizes the ERM's components, and names the entities and their relations.

ENTITY	RELATIONSHIP	CONNECTIVITY	ENTITY
SCHOOL	operates	1:M	DEPARTMENT
DEPARTMENT	has	1:M	STUDENT
DEPARTMENT	employs	1:M	PROFESSOR
DEPARTMENT	offers	1:M	COURSE
COURSE	generates	1:M	CLASS
PROFESSOR	is dean of	1:1	SCHOOL
PROFESSOR	chairs	1:1	DEPARTMENT
PROFESSOR	teaches	1:M	CLASS
PROFESSOR	advises	1:M	STUDENT
STUDENT	enrolls in	M:N	CLASS
BUILDING	contains	1:M	ROOM
ROOM	is used for	1:M	CLASS
Note: ENROLL is the composite entity that implements the M:N relationship "STUDENT enrolls in CLASS."			

The Figure below shows the ERD for ABC College. Note that this is an implementation-ready model.



DATABASE DESIGN CHALLENGES: CONFLICTING GOALS

Database designers often must make design compromises that are triggered by conflicting goals, such as:-

1. Adherence to Design standards.

The database design must conform to design standards. Such standards have guided you in developing logical structures that minimize data redundancies, thereby minimizing the likelihood that destructive data anomalies will occur.

You have also learned how standards prescribe avoiding nulls to the greatest extent possible.

In short, design standards allow you to work with well-defined components and to evaluate the interaction of those components with some precision. Without design standards, it is nearly impossible to formulate a proper design process, to evaluate an existing design, or to trace the likely logical impact of changes in design.

2. Processing speed.

In many organizations, particularly those generating large numbers of transactions, high processing speeds are often a top priority in database design. High processing speed means minimal access time, which may be achieved by minimizing the number and complexity of logically desirable relationships

3. Information requirements.

The quest for timely information might be the focus of database design. Complex information requirements may dictate data transformations, and they may expand the number of entities and attributes within the design. Therefore, the database may have to sacrifice some of its “clean” design structures and/or some of its high transaction speed to ensure maximum information generation.