- .small-text {font-size: 0.50rem;}
- Chapter4: Entity Relationship (ER) Modeling
- Entity Relationship Model
- Entity
- Attributes
- Identifier and Composite identifier
- Entity's Notation
- Implementing Multi-valued Attributes
- Derived Attributes
- Relationship
- Relationship's Notation
- Existence Dependence
- Weak Entity
- Strong Entity
- Example of Strong and Weak Entities
- Illustrate Relationship Between Weak & Strong Entity
- Relationship Strength
- Illustration of Relationship Strength
- Implementation Strong / Weak Relationship in DBMS
- The Order to Load Tables Under 1:M Relationship
- Relationship Participation
- Illustration of Relationship Participation
- Relationship Degree
- Recursive Relationship
- Associative (Composite) Entities
- Illustration of Associative Entities
- Developing an ER Diagram
- Tiny College (TC) (1,2/10)
- Tiny College (TC) (3/10)
- Tiny College (TC) (4/10)
- Tiny College (TC) (5/10)
- Tiny College (TC) (6/10)
- Tiny College (TC) (7/10)
- Tiny College (TC) (8/10)
- Tiny College (TC) (9/10)
- Tiny College (TC) (10/10)
- Tiny College (TC) (Summary: Entities)

---

marp: true theme: default class: invert size: 16:9 paginate: true footer: 國立陽明交通大學電子與光子學士學位學程 headingDivider: 1 style: | section::after { content: attr(data-marpit-pagination) '/' attr(data-marpit-pagination-total); }

.middle-grid { display: grid; grid-template-columns: repeat(2, minmax(0, 1fr)); gap: 1rem; } .middle-grid img { width: 75%; } .grid { display: grid; grid-template-columns: 1fr 1fr; gap: 10px; } .grid img { width: 100%; } .red-text { color: red; } .brown-text { color: brown; } .blue-text { color: lightskyblue;
}

.small-text { font-size: 0.50rem; }

Chapter4: Entity Relationship (ER) Modeling

- Data modeling is the first step in database design, as a bridge between real-world objects and the database model implemented in the computer.
- It is important to illustrate data-modeling details graphically through entity relationship diagrams (ERDs) to facilitate communication.

Entity Relationship Model

- The entity relationship model (ERM) generate ERD (ER diagram)
- The ERD represents the external model as viewed by end users
- The ERD represents the conceptual model as viewed by database designer
- ERDs describe the database's main components:
  - Entities

    - Attributes
    - Relationships

Entity

- An entity is an object of interest to the end user
- An entity in the ERM is a table (relation)
- In Crow's Foot notation, an entity is represented by a rectangle
    - contains entity name
    - entity name is a noun usually written in all capital letters. It would depend on your naming convention

Attributes

- **Attributes** are characteristics of entities
- **Required attribute** (not null) and **optional attribute** (allow null)
- Attributes must have a **domain**, the set of possible values for a given attribute
- **Identifier** and **composite identifier** is one or more attributes that uniquely identify each row.
- **Simple attribute** (age, sex) and **composite attribute** (address, phone_number)
- **Single-valued attribute** (emp_id) and **multi-valued attributes** (car_color, emp_habit)
- **Derived attribute** whose value is calculated from other attributes (working_years)

Identifier and Composite identifier

- CLASS_CODE is a identifier
- (CRS_CODE, CLASS_SECTION) is a composite identifier

Entity's Notation

Required attribute: bold font PK: in a separated cell with bold and underline font

Implementing Multi-valued Attributes

- If necessary, replace multi-value attribute by creating several new attributes
- If necessary, replace multi-value attribute by creating an new entity

Q: What is the pros and cons between the two replacement approaches

Derived Attributes

Q: What attribute is proper to use derived attributed, working_year or total_amount?

Relationship

- The entities that participate in a relationship are also known as participants
- A relationship is identified by a name that describes the relationship
- The relationship name is an active or passive verb
- Connectivity describes the relationship classification: 1:1, 1:M, and M:N
- Cardinality expresses the minimum and maximum number of entity occurrences associated with one occurrence of the related entity

Relationship's Notation

- (1, 4): one professor teach at least one and no more than four classes
- (1, 1): each class is taught by one and only one professor

Existence Dependence

- Entity can be strong or weak depending on whether the entity can exist independently or not.

- A strong entity can exist apart from all of its related entities, it is existence-independent
- A weak entity is existence-dependent on another related entity occurrence
- Relationship 'EMPLOYEE claims DEPENDENT', the DEPENDENT entity is existence dependent on the EMPLOYEE entity. That is, DEPENDENT has a mandatory (NOT NULL) foreign key, EMP_NUM to link with EMPLOYEE.

Weak Entity

- A weak entity is existence-dependent on a strong entity with a strong (identifying) relationship - requires a non-null FK from the related strong entity and form a composite PK.
    - DEPENDENT(**EMP_NUM, DEP_SID**, DEP_NAME, DEP_DOB), EMP_NUM is FK,
- A weak entity always has a **mandatory participation** to a strong entity (every row of the weak entity must be associated with one row of a strong entity because of non-null FK).

Strong Entity

- A strong entity has a PK that uniquely identifies each record without depending on other entity.
- EMPLOYEE(**EMP_NUM**, EMP_LNAME, EMP_FNAME, EMP_INITIAL, EMP_DOB, EMP_HIREDATE)

Example of Strong and Weak Entities

Considering two entities: EMPLOYEE (strong) and DEPENDENT (weak)

- DEPENDENT is weak because it has no sufficient PK by itself at the beginning
    - DEPENDENT(**DEP_SID**, DEP_NAME, DEP_DOB, EMP_NUM), EMP_NUM is FK, but when two employee is couple, their children will be duplicated
    - DEP_SID alone cannot uniquely identify a dependent.
- Need expand PK of DEPENDENT by combining EMP_NUM
    - DEPENDENT(**EMP_NUM, DEP_SID**, DEP_NAME, DEP_DOB) to build a strong (identifying) relationship with EMPLOYEE to uniquely identify each dependent.

- DEP_NUM is better than DEP_SID in terms of privacy
    - DEPENDENT(**EMP_NUM, DEP_NUM**, DEP_NAME, DEP_DOB), EMP_NUM is non-null FK

Illustrate Relationship Between Weak & Strong Entity

Relationship Strength

- Relationship strength can be strong or weak based on how to define PK of a related entity.
- To implement a relationship, the PK of one entity (parent entity, normally on the “one” side of 1:M relationship) appears as a FK in the related entity (child entity, mostly the entity on the “many” side of 1:M relationship) to link two entities.
    - Non-identifying(weak) Relationships : if the PK of the "M side" entity does NOT contain a PK of the '1 side' entity
    - Identifying (strong) Relationships: when the PK of the "M side" entity contains the PK of the "1 side" entity

Illustration of Relationship Strength

- dotted line shows weak relationships; solid line shows strong relationships

Implementation Strong / Weak Relationship in DBMS

- Use a strong relationship when: • The M-side entity is conceptually a part of the 1-side. • The M-side object should be destroyed when the 1-side is destroyed (e.g., an employee’s dependant).
- Use a weak relationship when: • The M-site entity can exist independently of the 1-side. • The M-side object should not be deleted if the 1-side is deleted (e.g., an

employee and a department).

The Order to Load Tables Under 1:M Relationship

- Keep in mind that the order in which the tables are created and loaded is very important.
- In the “COURSE generates CLASS” relationship, the COURSE table must be created before the CLASS table. After all, it would not be acceptable to have the CLASS table’s foreign key refer to a COURSE table that did not yet exist.
- Load the data of the “1” side first in a 1:M relationship to avoid the possibility of referential integrity errors.

Relationship Participation

- Relationship participation is either optional or mandatory.
- Because of the bidirectional nature of relationships, it is necessary to determine the connectivity as well as max and min cardinalities of the relationship from COURSE to CLASS and from CLASS to COURSE.
- **Optional participation** means that some rows may not participate into the relationship
- **Mandatory participation** means that each row must participate into the relationship

Illustration of Relationship Participation

Relationship Degree

Recursive Relationship

Associative (Composite) Entities

- The ER model uses the associative entity to represent an M:N relationship between two or more entities
- It is also called a composite or bridge entity and is a 1:M relationship with the parent entities
- It is composed of the primary key attributes of each parent entity
- The composite entity may also contain additional attributes that play no role in connective process

Illustration of Associative Entities

STUDENT has CLASS is a M:N relationship

Developing an ER Diagram

Building an ERD usually involves the following activities as a iterative process:

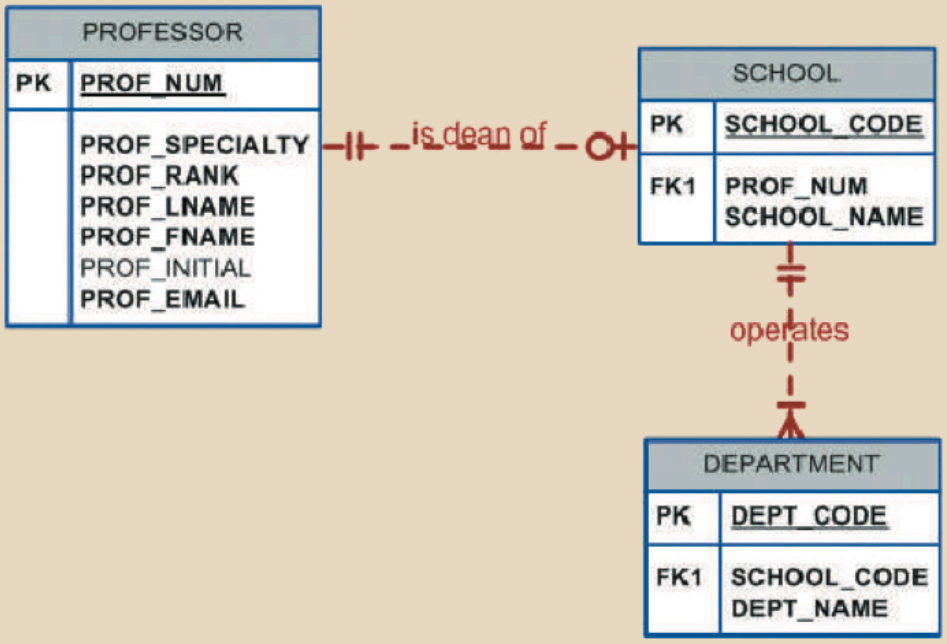
- Create a detailed description of the organization's operations
    - Interview users
    - Investigate SOPs, Forms, Reports
- Identify business rules based on the description of operations
- Identify main entities and relationships from the business rules
- Develop the initial ERD
- Identify the attributes and primary keys that adequately describe entities
- Revise and review the ERD

Tiny College (TC) (1,2/10)

- Tiny College is divided into several schools.
    - A school is managed by a professor.
    - Each professor can be the dean of only one school, or none of any school.

- Each school has several departments.
  - The number of departments operated by a school is at least one to many
  - Each department belongs to only a single school

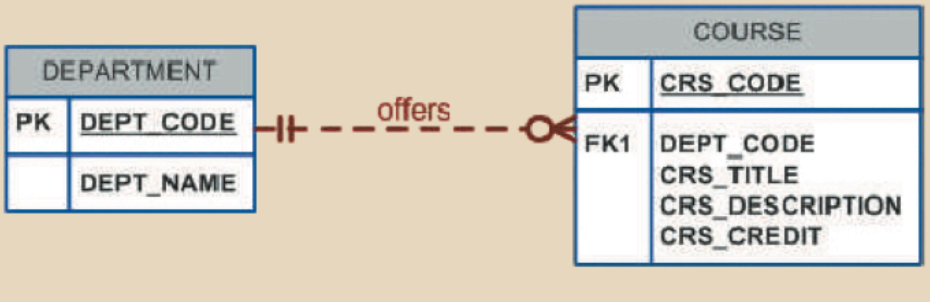
FIGURE 4.26 THE FIRST TINY COLLEGE ERD SEGMENT



Tiny College (TC) (3/10)

- Each department may offer courses.
  - Some departments that were classified as "research only," they would not offer courses; therefore, the COURSE entity would be optional to the DEPARTMENT entity.

FIGURE 4.27 THE SECOND TINY COLLEGE ERD SEGMENT

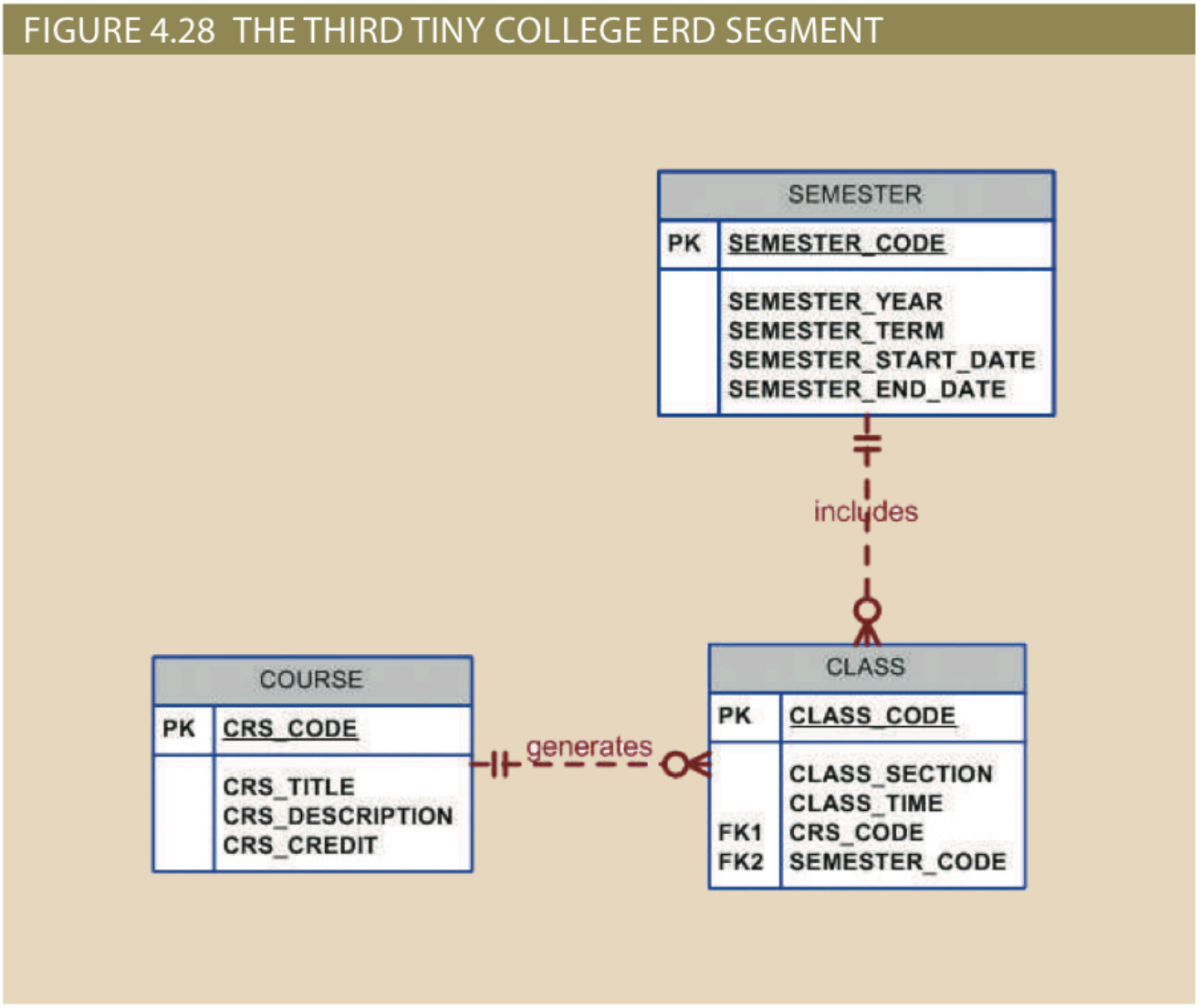


Tiny College (TC) (4/10)

- A course can be taught in several classes.
- A course may not be taught in some semester

- A class is offered during a given semester. SEMESTER defines the year and the term that the class will be offered.
- CLASS is optional to SEMESTER.
- CLASS is optional to COURSE.

FIGURE 4.28 THE THIRD TINY COLLEGE ERD SEGMENT

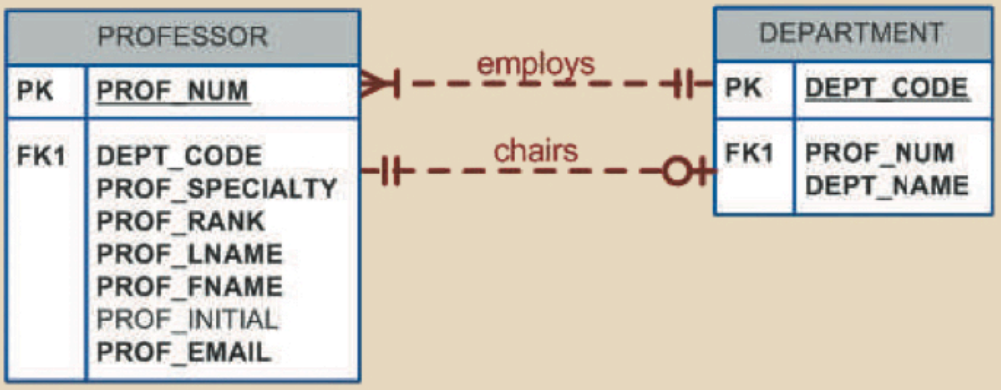


Tiny College (TC) (5/10)

- Each department should have one or more professors assigned to it.
- One and only one of those professors chairs the department
- Not all professors are required to chair a department.

- DEPARTMENT is optional to PROFESSOR in the "chairs" relationship.

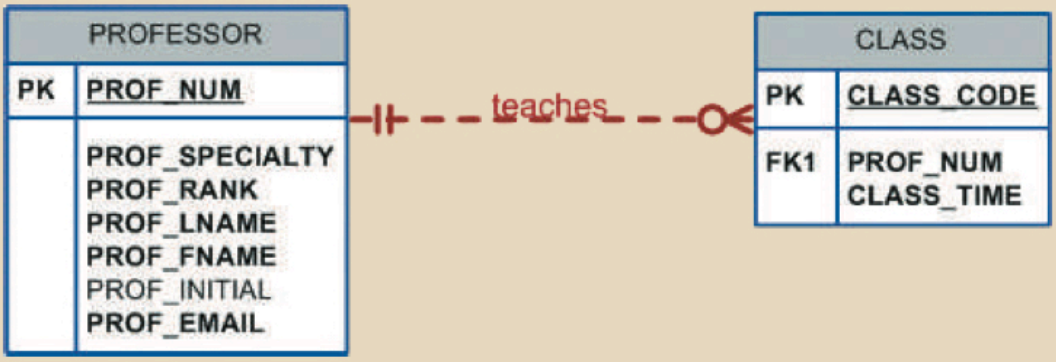
FIGURE 4.29 THE FOURTH TINY COLLEGE ERD SEGMENT



Tiny College (TC) (6/10)

- Each professor may teach up to four classes; each class is belong to a course.
- A professor may also be on a research contract and teach no classes at all.

FIGURE 4.30 THE FIFTH TINY COLLEGE ERD SEGMENT

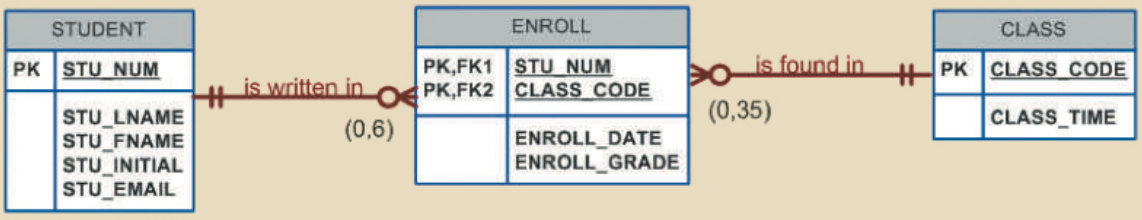


Tiny College (TC) (7/10)

- A student may enroll in several classes but take each class only once.
- Each student may enroll in up to six classes, and each class may have up to 35 students, (STUDENT and CLASS is M:N relationship).

- This M:N relationship must be divided into two 1:M relationships by ENROLL entity

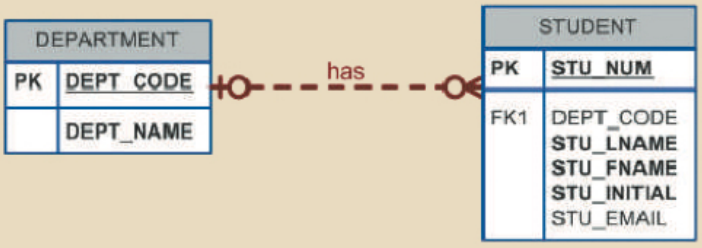
FIGURE 4.31 THE SIXTH TINY COLLEGE ERD SEGMENT



Tiny College (TC) (8/10)

- Each department has several students whose major is offered by that department. (VAGUE!!)
- Each student has only a single major associated with a single department.
- It is possible for a student not to declare a major field of study.

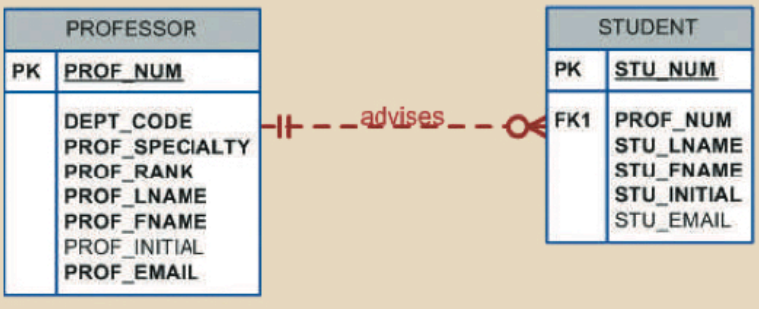
FIGURE 4.32 THE SEVENTH TINY COLLEGE ERD SEGMENT



Tiny College (TC) (9/10)

- 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.

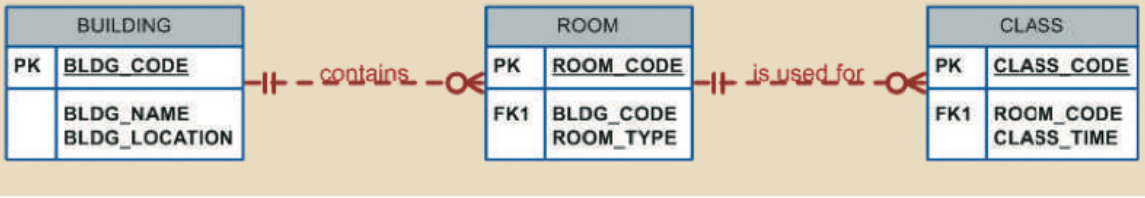
FIGURE 4.33 THE EIGHTH TINY COLLEGE ERD SEGMENT



Tiny College (TC) (10/10)

- A class is taught in a room.
- Each room is located in a building.
- A building can contain many rooms.
- Some buildings do not contain (class) rooms.

FIGURE 4.34 THE NINTH TINY COLLEGE ERD SEGMENT



Tiny College (TC) (Summary: Entities)

PROFESSOR COURSE STUDENT SCHOOL CLASS BUILDING DEPARTMENT SEMESTER ROOM ENROLL (the associative entity between STUDENT and CLASS)

Summary: Components of ERM

TABLE 4.4

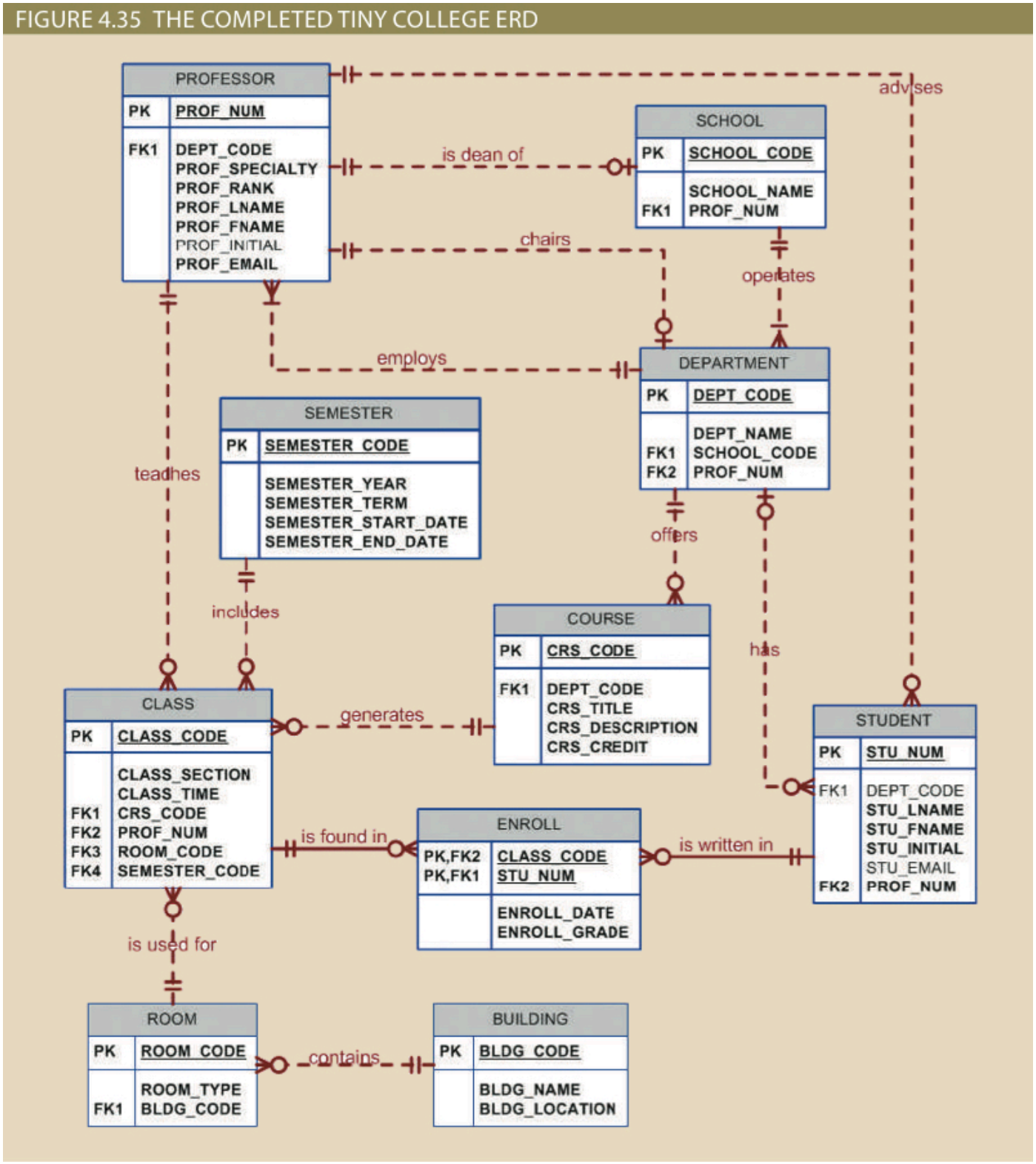
COMPONENTS OF THE ERM

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
SEMESTER	includes	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."

Summary: Completed ERD

FIGURE 4.35 THE COMPLETED TINY COLLEGE ERD



Database Design Challenges: Conflicting Goals

- Database designers must often make design compromises that are triggered by conflicting GOALS
    - Database design must conform to design standards
    - High processing speed may limit the number and complexity of logically desirable relationships

- However, a design that meets all requirements and design conventions are the most important goals

Review Questions

- What is the difference between weak entity and strong entity?
- What is the difference between weak (non-identifying) and identifying (strong) relationship?
- How to translate M:N relationship in ERM?

Homework #B

資料庫課程作業(B)

Present Final Project Progress

- Date: 04/15 (專班) and 04/16 (資財)
- Duration: 5 minutes per team
- Approach: oral presentation and one word page only
- Agenda:
    - Project briefing and teaming
    - Expected deliverables
    - Current progress
    - Support needed (if any)