Databases and Information Systems

Designing an ER Diagram from Textual Requirements

Prof. Ing. Lelio Campanile, PhD

Data Analytics Bachelor
Universita degli Studi della Campania Luigi Vanvitelli

Guided classroom exercise

Learning goals

- Understand how to move from a **textual description** to an **ER schema**
- Review the main concepts of the **Entity-Relationship model**
- Practice a **step-by-step design method**
- Recognize **entities, attributes, relationships, and cardinalities**

Quick review

Entity

Definition

An **entity** is a real-world object we want to represent in the database.

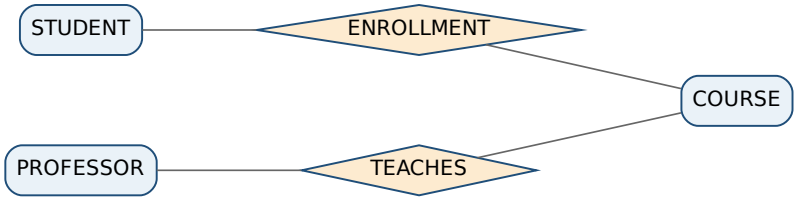
- Typical examples: **Student**, **Professor**, **Course**
- In **Chen notation**, entities are represented by **rectangles**
- Each entity will later have its own set of attributes

Relationship

Definition

A **relationship** expresses an association between two or more entities.

- Example: a **student enrolls** in a **course**
- Example: a **professor teaches** a **course**
- In Chen notation, relationships are represented by **diamonds**

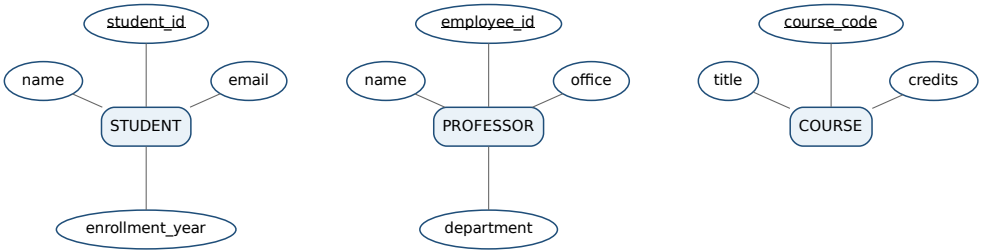


Attribute

Definition

An **attribute** is a property that describes an entity or, in some cases, a relationship.

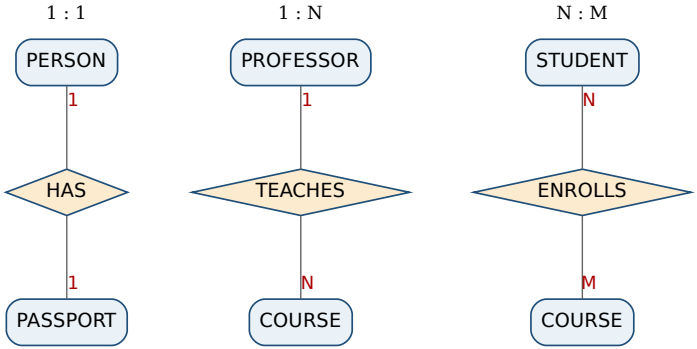
- Examples for **Student**: student_id, name, email, enrollment_year
- A **key attribute** uniquely identifies each entity instance
- In Chen notation, attributes are represented by **ellipses**



Cardinality

Definition

Cardinality tells us how many instances of one entity can be associated with instances of another entity through a relationship.



Guided example

Problem description

A university wants to manage information about **students**, **professors**, and **courses**.

- Each **student** has a student ID, name, email, and year of enrollment
- Each **professor** has an employee ID, name, office, and department
- Each **course** has a course code, title, and number of credits
- Students can enroll in courses
- Professors teach courses
- For each enrollment we store the **final grade**

Step 1 - Identify the entities

Question

Which **main objects** of the domain do we need to store?

Think for a moment before looking at the solution.

Step 1 - Identify the entities

Answer

From the description, the main entities are:

- **Student**
- **Professor**
- **Course**

Step 2 - Add the attributes

Question

Which properties should we store for each entity?

List the attributes for Student, Professor, and Course.

Step 2 - Add the attributes

Student

- student_id
- name
- email
- enrollment_year

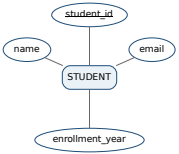
Professor

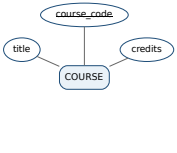
- employee_id
- name
- office
- department

Course

- course_code
- title
- credits

- Underlined attributes are used here as **keys**
- Keys distinguish one entity instance from another





Step 3 - Identify the relationships

Question

How are the entities connected?

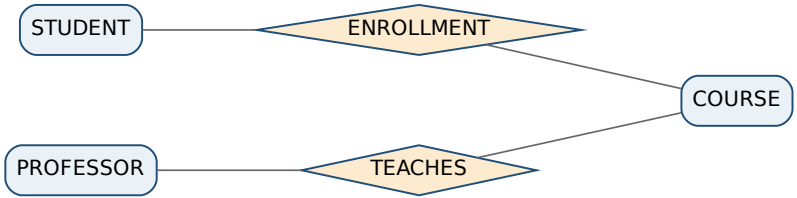
Look for the verbs in the textual description.

Step 3 - Identify the relationships

Answer

The text gives us two relationships:

- Students **enroll** in courses
- Professors **teach** courses



Step 4 - Determine the cardinalities

Question

For each relationship, how many instances can participate on each side?

- Can one student enroll in more than one course?
- Can one course have more than one student?
- Can one professor teach more than one course?
- Can one course be taught by more than one professor?

Step 4 - Determine the cardinalities

ENROLLMENT

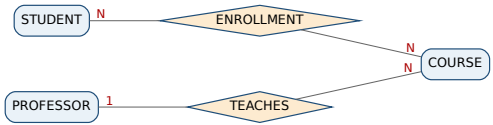
- A student can enroll in **many** courses
- A course can have **many** students

Therefore, the relationship is **N:M**.

TEACHES

- One professor can teach **many** courses
- Each course is taught by **one** professor

Therefore, the relationship is **1:N**.



Step 5 - Add attributes of relationships

Question

Where should we store the **grade**?

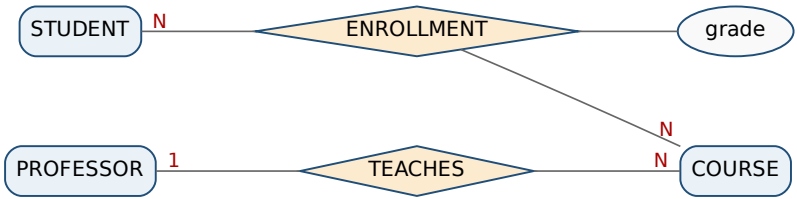
- Is grade an attribute of **Student**?
- Is grade an attribute of **Course**?
- Or does it depend on the specific *Student-Course* pair?

Step 5 - Add attributes of relationships

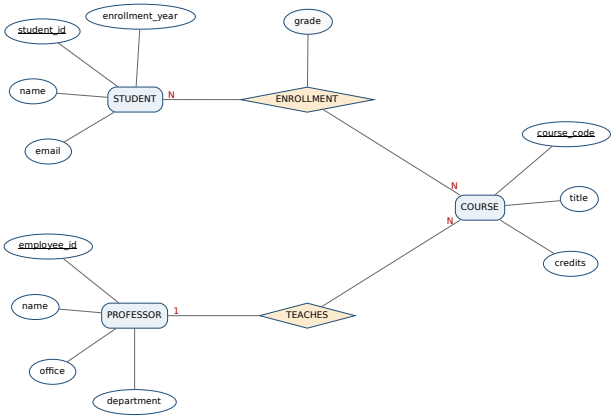
Answer

Some information does **not** belong to a single entity. It belongs to a **relationship**.

- The **grade** is not a property of Student alone
- The **grade** is not a property of Course alone
- The **grade** depends on the specific pair *(Student, Course)*



Final ER diagram



How to reason during design

1. Read the text and highlight the **main nouns** - possible entities
2. Search for **properties** - possible attributes
3. Search for **verbs** - possible relationships
4. For each relationship, ask: *how many?* - cardinality
5. Check whether some attributes belong to a **relationship** instead of an entity

Questions for the class

- Can a student exist in the database even if the student is not enrolled in any course?
- What would change if a course could be taught by **multiple professors**?
- Why is **grade** an attribute of **ENROLLMENT** and not of **STUDENT**?
- If we wanted to store the semester of the exam, where should we place it?

In-class exercise

Design an ER diagram for a **library system**.

The library stores information about:

- **Book** (ISBN, title, year)
- **Author** (author_id, name)
- **Member** (member_id, name, email)

Rules:

- A book can have multiple authors
- An author can write multiple books
- Members can borrow books
- For each borrowing, store borrow_date and return_date

Summary

- We started from a **textual description**
- We identified **entities**, **attributes**, and **relationships**
- We added **cardinalities**
- We recognized an **attribute of a relationship**
- We obtained a complete ER schema in **Chen notation**

Questions?