Databases and Information Systems

Designing an ER Diagram from Textual Requirements – Library System

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.

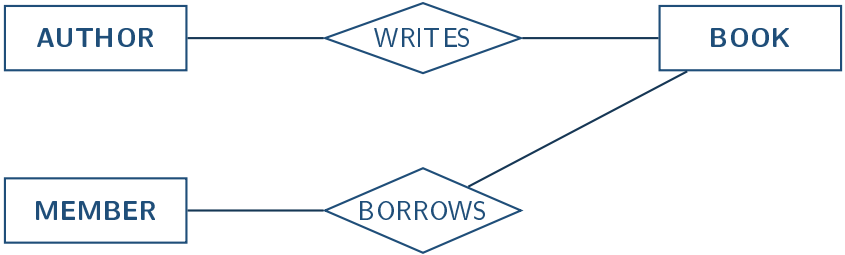
- Typical examples: **Book**, **Author**, **Member**
- In **BCN notation**, entities are represented by **rectangles**
- Each entity will later have its own set of attributes

Relationship

Definition

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

- Example: an **author writes** a **book**
- Example: a **member borrows** a **book**
- In BCN notation, relationships are represented by **diamonds**

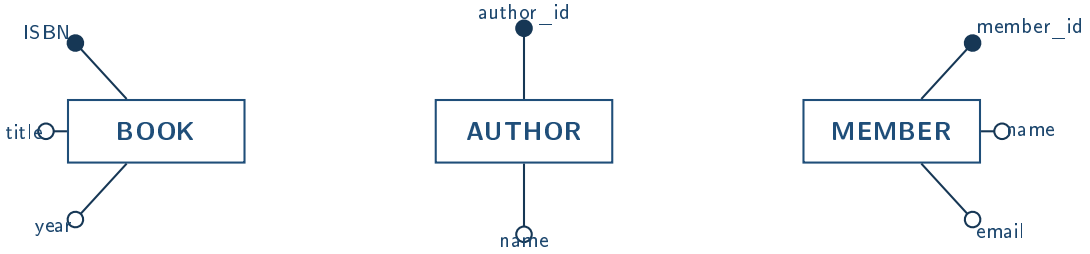


Attribute

Definition

An **attribute** is a property that describes an entity or a relationship.

- In BCN notation: a line ending in a small **circle**; name written **beside** the circle
- **Filled circle** = internal identifier; **empty circle** = regular attribute



Cardinality

Definition

Cardinality tells us how many instances of one entity can be associated with instances of another through a relationship. In BCN notation the pair **(min,Max)** is written on each arc.

$1:1$

$1:N$

$N:M$

Guided example

Problem description

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**

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:

- **Book** **Author** **Member**

Step 2 - Add the attributes

Question

Which properties should we store for each entity?

List the attributes for Book, Author, and Member.

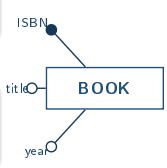
Step 2 - Add the attributes

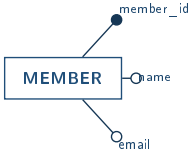
Book
- ISBN title year

Author
- author_id name

Member
- member_id
- name email

- Underlined = **identifier** (filled circle on arc)





Step 3 - Identify the relationships

Question

How are the entities connected?

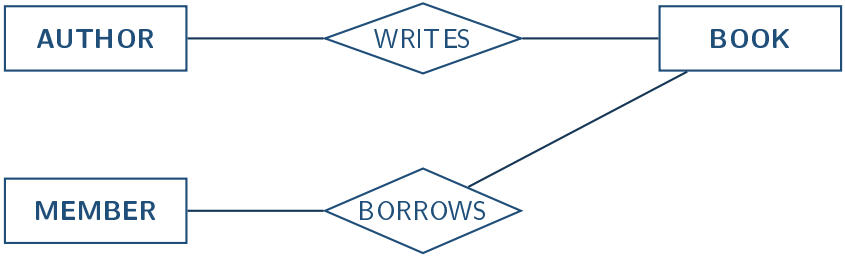
Look for the verbs in the textual description.

Step 3 - Identify the relationships

Answer

The text gives us two relationships:

- Authors **write** books
- Members **borrow** books



Step 4 - Determine the cardinalities

Question

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

- Can one book have more than one author?
- Can one author write more than one book?
- Can one member borrow more than one book?
- Can one book be borrowed by different members over time?

Step 4 - Determine the cardinalities

WRITES

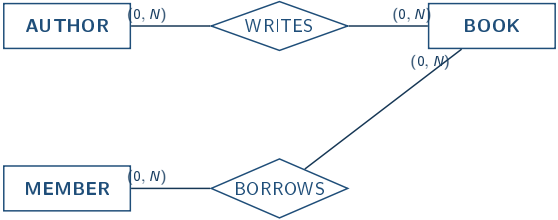
- A book can have **many** authors
- An author can write **many** books

Therefore: **N:M**.

BORROWS

- A member can borrow **many** books
- A book can be borrowed by **many** members over time

Therefore: **N:M**.



Step 5 - Add attributes of relationships

Question

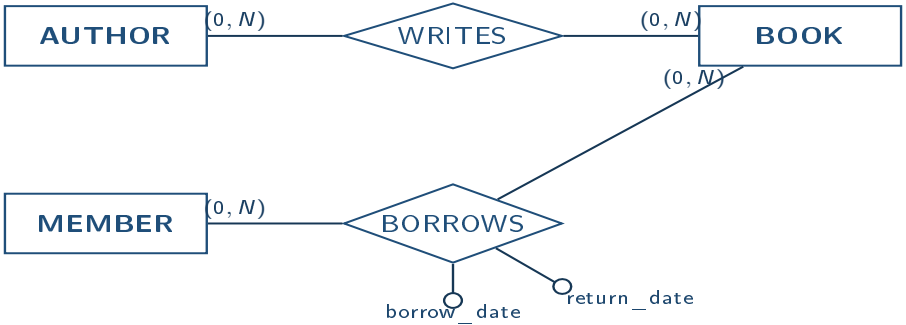
Where should we store **borrow_date** and **return_date**?

- Are they attributes of **Member**?
- Are they attributes of **Book**?
- Or do they depend on the specific *Member-Book* borrowing?

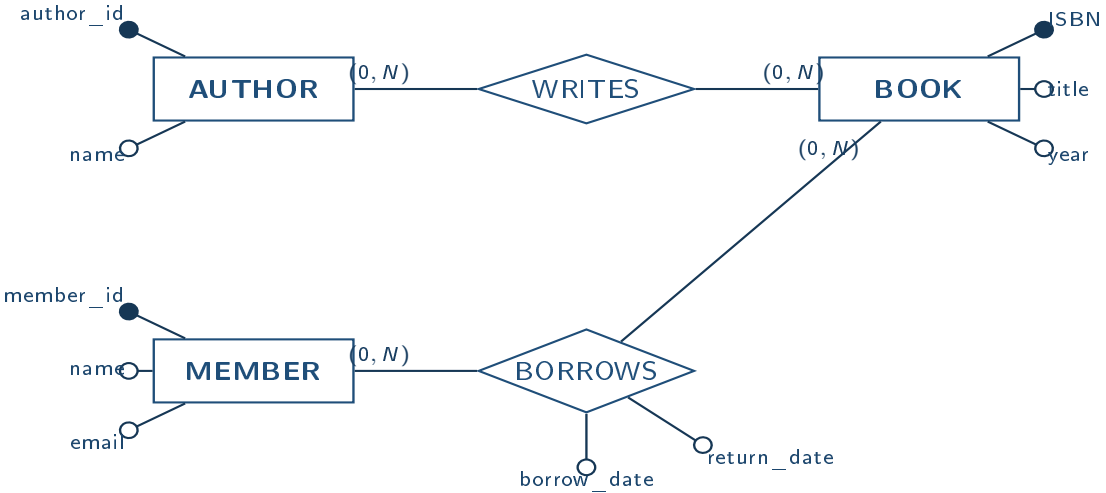
Step 5 - Add attributes of relationships

Answer

Some information does **not** belong to a single entity — it belongs to a **relationship**. Both dates depend on the specific pair *(Member, Book)*.



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 (min, Max)
5. Check whether some attributes belong to a **relationship** instead of an entity

Questions for the class

- Why is **WRITES** a many-to-many relationship?
- Why is **BORROWS** modeled as many-to-many?
- Why are **borrow_date** and **return_date** attributes of **BORROWS**?
- If we wanted to store the genre of a book, where should we place it?

In-class extension

How would you extend this model if the library also wanted to store:

- the **publisher** of each book,
- a **fine** for late returns,
- the **copy number** when the library owns multiple copies of the same book?

Summary

- We started from a **textual description**
- We identified **entities**, **attributes**, and **relationships**
- We added **cardinalities** in (min, Max) notation
- We recognized **attributes of a relationship**
- We obtained a complete ER schema in **BCN notation**

Questions?