Entity-Relationship (ER) Model

Problem → ER Design → Tables

- It's easier to model the real world with entities and relationships than using tables
- An ER design can be easily converted to relational tables
- ER Diagram is the graphical representation of an ER design

Example: Problem Description

- Student
  - id, name, address
- Department
  - name
- Classes
  - code, name, quarter, section number
- Class offerings and enrollment

Example: ER Diagram

Entity Set and Attributes

- Entity Set is a collection of entities
  - E.g. students, departments, classes
- Attributes are the common properties of the entities in an entity set
  - E.g. attributes of students: id, name, address
  - Must have simple values like numbers or strings (because they will be converted to columns in database tables)
**Keys**

- A key is an attribute or a combination of several attributes that uniquely identifies an entity in an entity set.
  - E.g. student ID
- Each entity set must have a key.
- An entity set may have multiple attributes that could be used as keys. Simply choose one of them as the key.

**Relationship**

- Students \(\rightarrow\) Take \(\rightarrow\) Classes
- Classes \(\rightarrow\) OfferedBy \(\rightarrow\) Departments

**Types of Relationships**

- Many-to-Many
- Many-to-One / One-to-Many
- One-to-One

**Many-to-Many Relationship**

- Take is a many-to-many relationship
  - Each student can take many classes
  - Each class can have many students
Many-to-One Relationship

- OfferedBy is a many-to-one relationship
  - Each class is offered by one department
  - Each department can offer many classes
- An arrow is used to indicate the "one" side

One-to-One Relationship

- Bestseller is a one-to-one relationship
  - Each manufacturer only has one bestselling product
  - Each product can only be the bestseller of one manufacturer
- Arrows on both sides

Relationship Examples

- Teachers teach Classes
- Customers place Orders
- Orders include Products

Attributes of Relationships

- Sometimes it's useful to attach an attribute to a relationship.
**Roles**

- An entity set may appear in the same relationship more than once.
- Label the edges with names called **Roles**

**ER Design**

1. Identify entity sets and attributes
   - Usually the *nouns* in the requirement description
2. Determine the keys
3. Identify relationships
   - Usually the *verbs* in the requirement description
4. Determine relationship types

---

**Design the Store Database**

Design a database to keep track of information about products, customers, and orders.

- Each product has an ID, a category, a description, and a price.
- Each customer has an ID, a name, and an address.
- Each order is placed by one customer, and an order includes one or more products.

**Basic Rules of ER to Relational Conversion**

- A entity set is converted to a table
- A many-to-many relationship is also converted to a table, including
  - Its own attributes
  - Key attributes from the associated entity sets
- A many-to-one relationship is merged into the “many” side with a foreign key to the “one” side

---

**Converting Entity Sets ...**

- **Students** (id, name, addr)
- **Departments** (name)

---

**... Converting Entity Sets**

- **Classes** (code, quarter, section)

Classes (code, name, quarter, section)
Converting Relationships ...

Takes( student_id, code, quarter, section )

... Converting Relationships

Classes ( code, name, quarter, section, department_name )

Relational Schema After Conversion

Students( id, name, address )
Departments ( name )
Classes ( code, name, quarter, section, department_name )
Takes( student_id, code, quarter, section )

More Conversion Examples

Exercise

Convert the ER design of the Store database to relational schema