Fundamentals of Database Systems

Chapter 3:
Data Modeling using the Enhanced Entity Relationship Model

eSPIN University

Overview

We learned that:

1. A DMBS will allow us to create and manage databases.
2. We can query the database (retrieve data from tables in the database).
3. We can update the database (change, add, or delete data from tables in the database).
4. We can complete a conceptual design of the database using the ER model.

This chapter

- **Enhanced Entity Relationship Model**

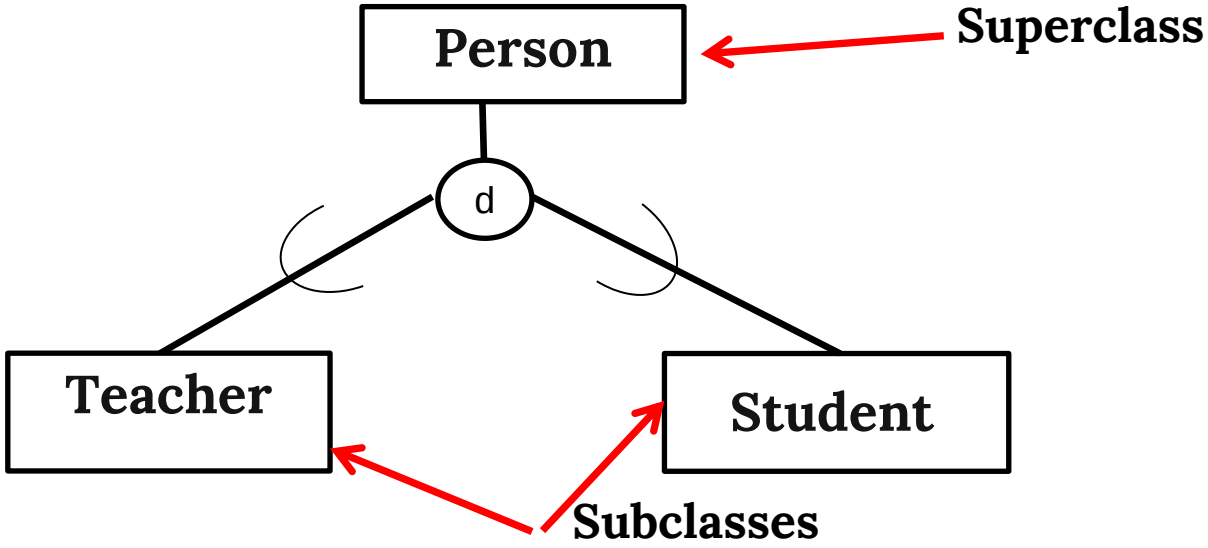
- **EER model includes all modeling concepts of the ER model**
- **In addition, EER includes:**

1. Subclasses and superclasses
2. Specialization and generalization
3. Attribute and relationship inheritance
4. Category or union type (*not covered here*)

- *The EER is not as popular as the ER model*

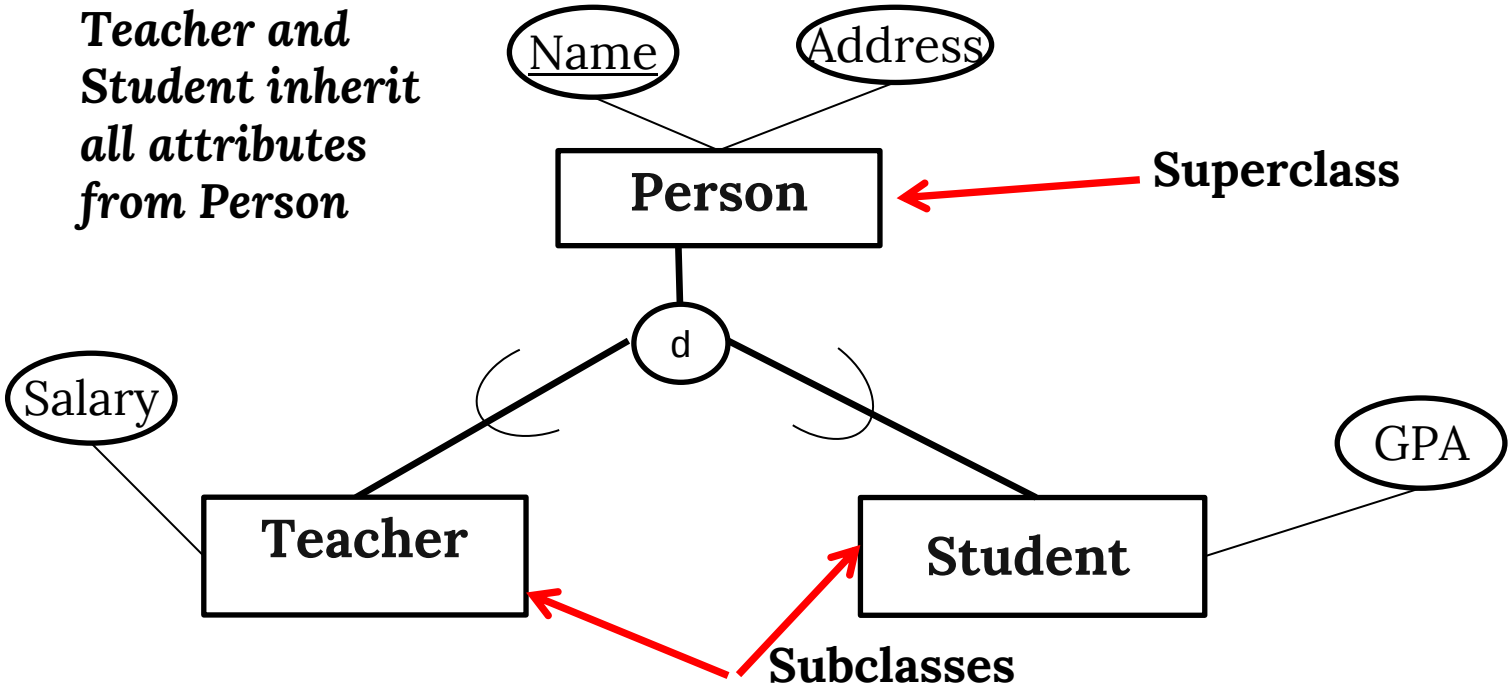
Subclasses and Superclasses

- Subclass entity inherits all attributes and relationships of superclass
- Example:



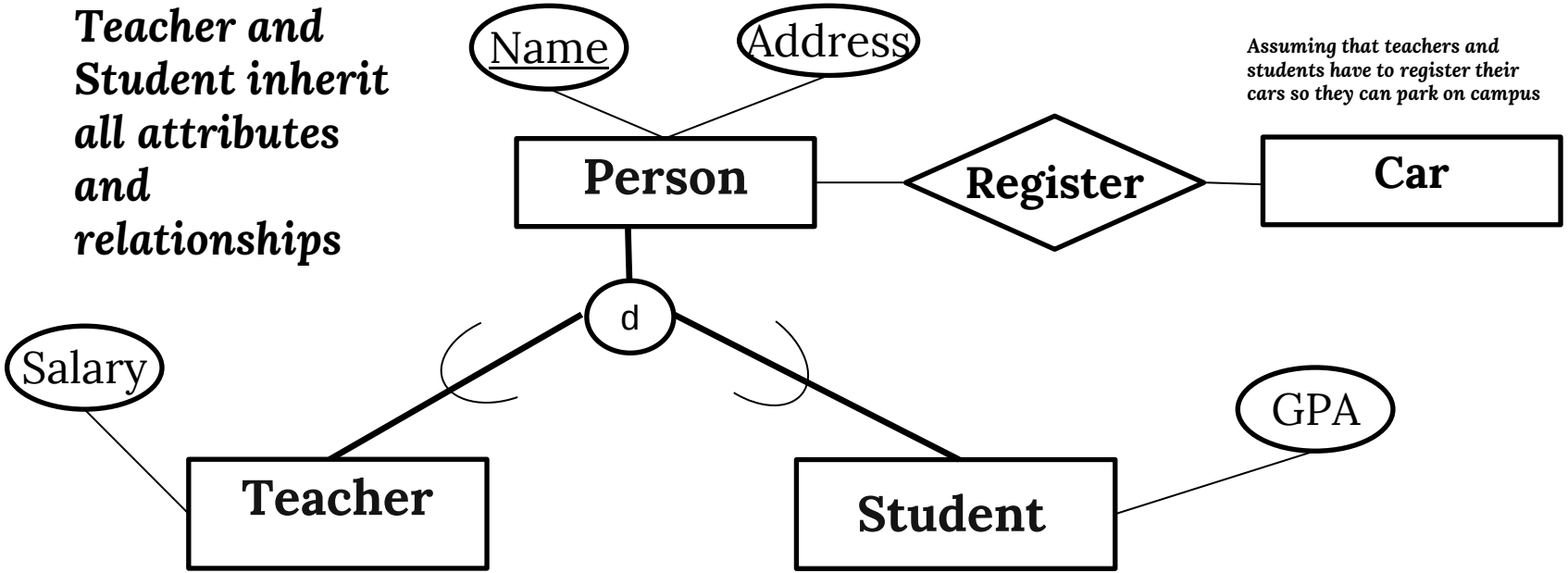
1. Subclasses and Superclasses

Teacher and Student inherit all attributes from Person



1. Subclasses and Superclasses

Teacher and Student inherit all attributes and relationships



1. Subclasses and Superclasses

- **Specialization:** Process of defining a set of subclasses of an entity type.
- Generate subclasses from one entity type
- **Ex #1:** Employee -> Programmer, developer, manager..
- **Ex #2:** TV Show -> Drama series, talk show, reality show..
- Certain attributes may apply to some but not all entities of the superclass

Subclasses and Superclasses

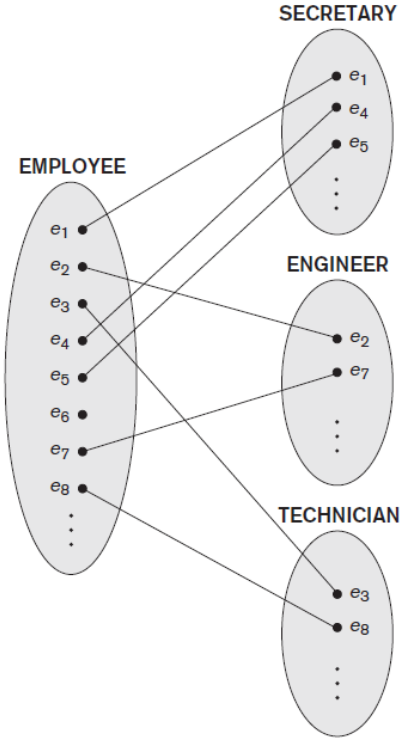


Figure 8.2
Instances of a specialization.

Subclasses and Superclasses

- **Generalization:** Process of defining a generalized entity type from the given entity types
- Generate a **superclass** from multiple entity types
- **Ex #1:** Books, video games -> Products
- **Ex #2:** Trucks, cars -> Vehicles

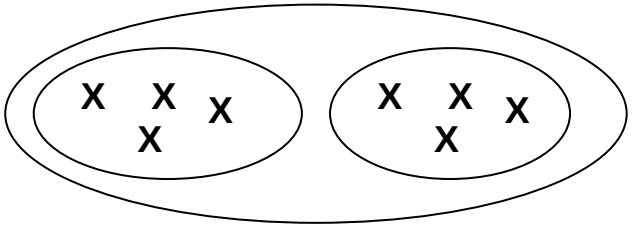
2. Constraints on Specialization and Generalization

- **Disjointness constraint:** Specify if the subclasses of the specialization must be disjointed.

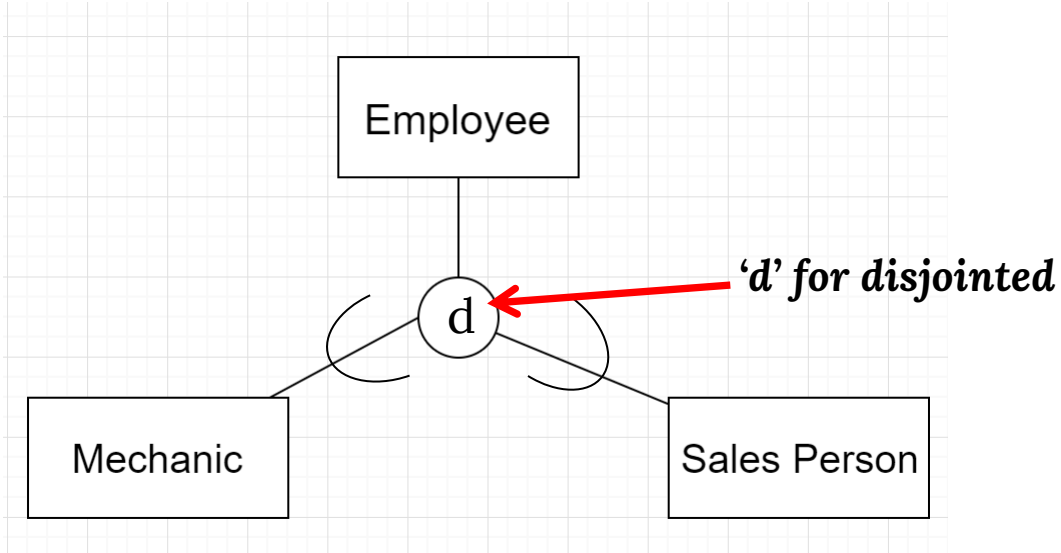
Two options:

- ***Disjointness*:** Can be either an entity of the subclass ‘A’ **or** the subclass ‘B’.
- ***Overlapping*:** Can be both an entity of the subclass ‘A’ **and** the subclass ‘B’

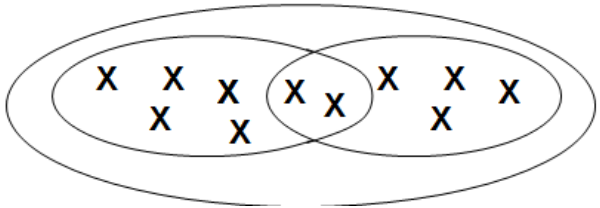
Disjointness constraint -> Disjointness

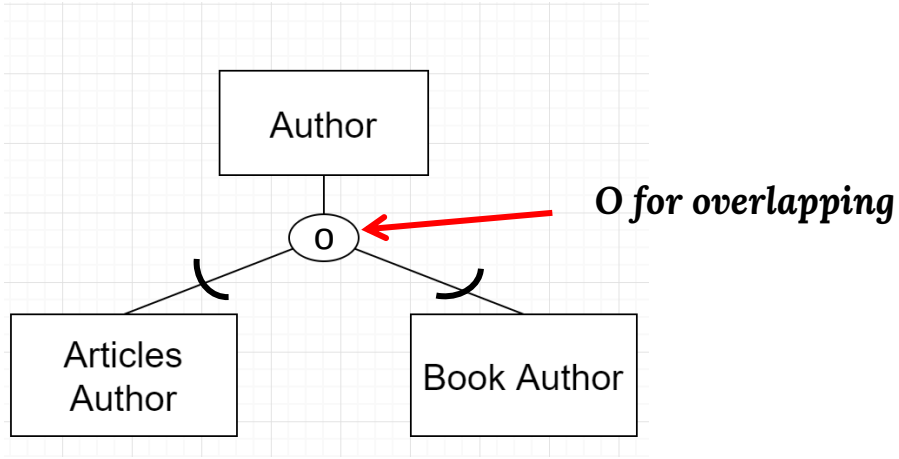


An employee can either be a mechanic or a sales person but not both



Disjointness constraint -> Overlapping





An Author can be both an article author and a book author

2. Constraints on Specialization and Generalization

- **Completeness constraint:** Total or partial

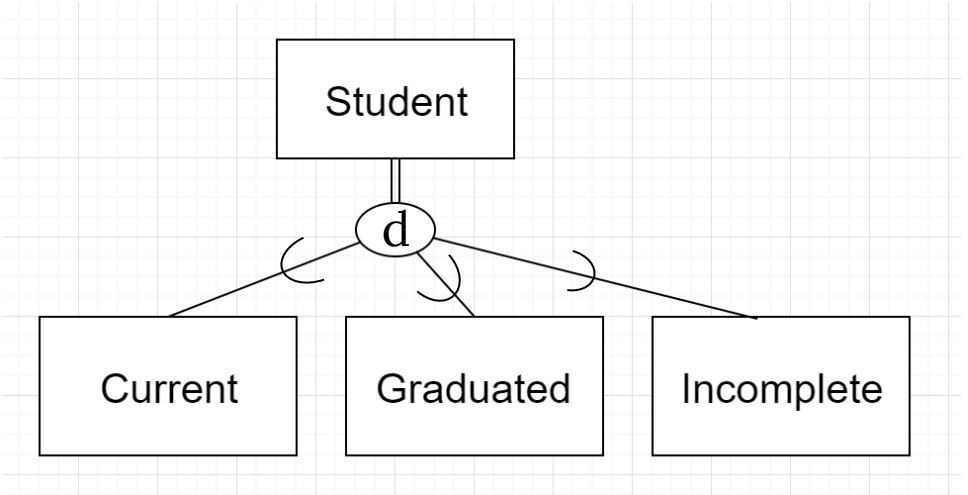
Two options:

- ***Total*:** Each entity (instance) of the superclass entity type has to be an entity of at least one of the subclasses
- ***Partial*:** Entities of the superclass entity type do not have to be an entity of at least one of the subclasses

Completeness constraint -> Total participation

A student ***has to*** be current, graduated, or incomplete.

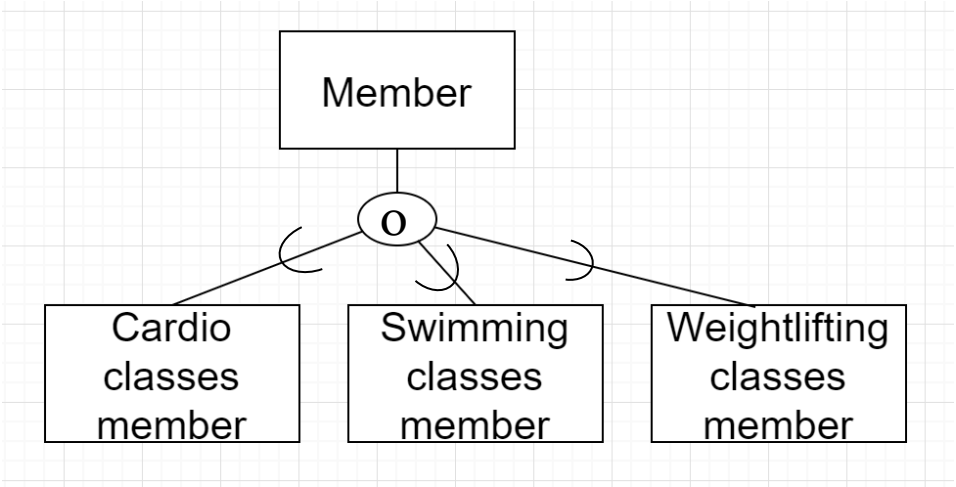
Why d?



Completeness constraint -> Partial participation

A gym member ***may*** register for cardio, swimming or weightlifting classes but doesn't have to.

Why o?



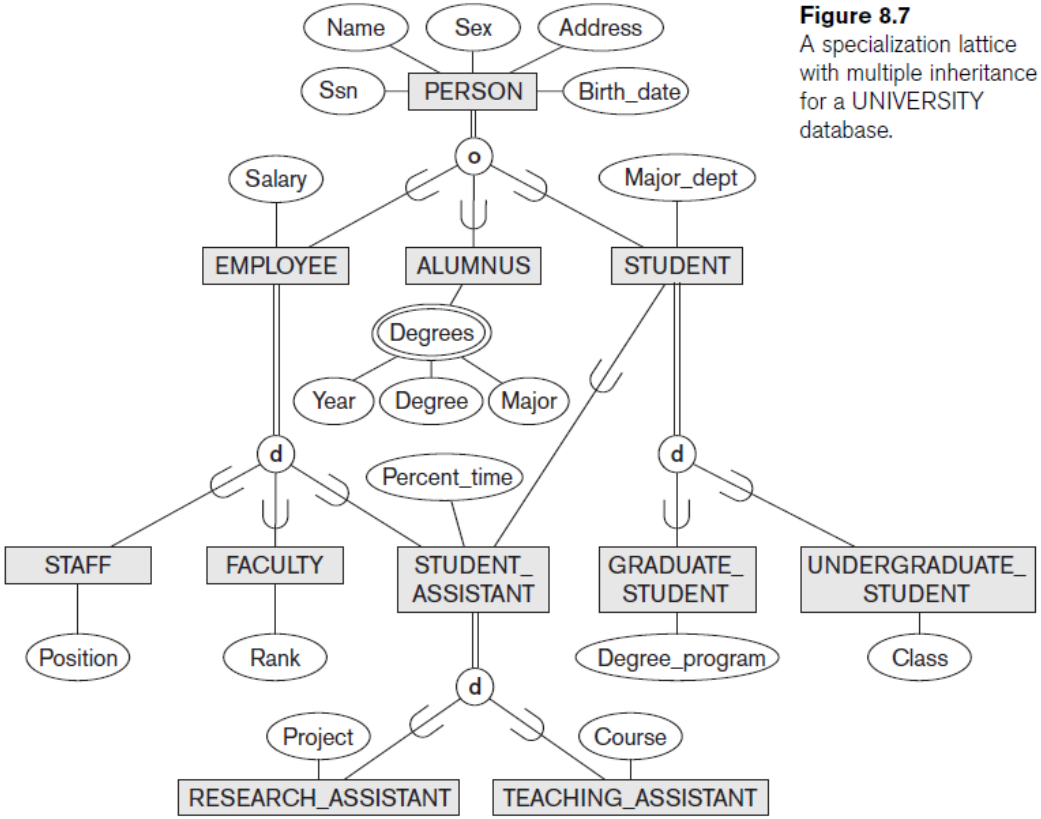


Figure 8.7
A specialization lattice with multiple inheritance for a UNIVERSITY database.

Summary on constraints

- **Four types of constraints:**

1. **Disjoint Total**
2. **Disjoint Partial**
3. **Overlap Total**
4. **Overlap Partial**

Summary on EER

- EER has features proposed to improve the ER
- Other features exist (not only the ones explained here)
- Not as popular or as well known as the ER