Relational Database Systems I

Topics:

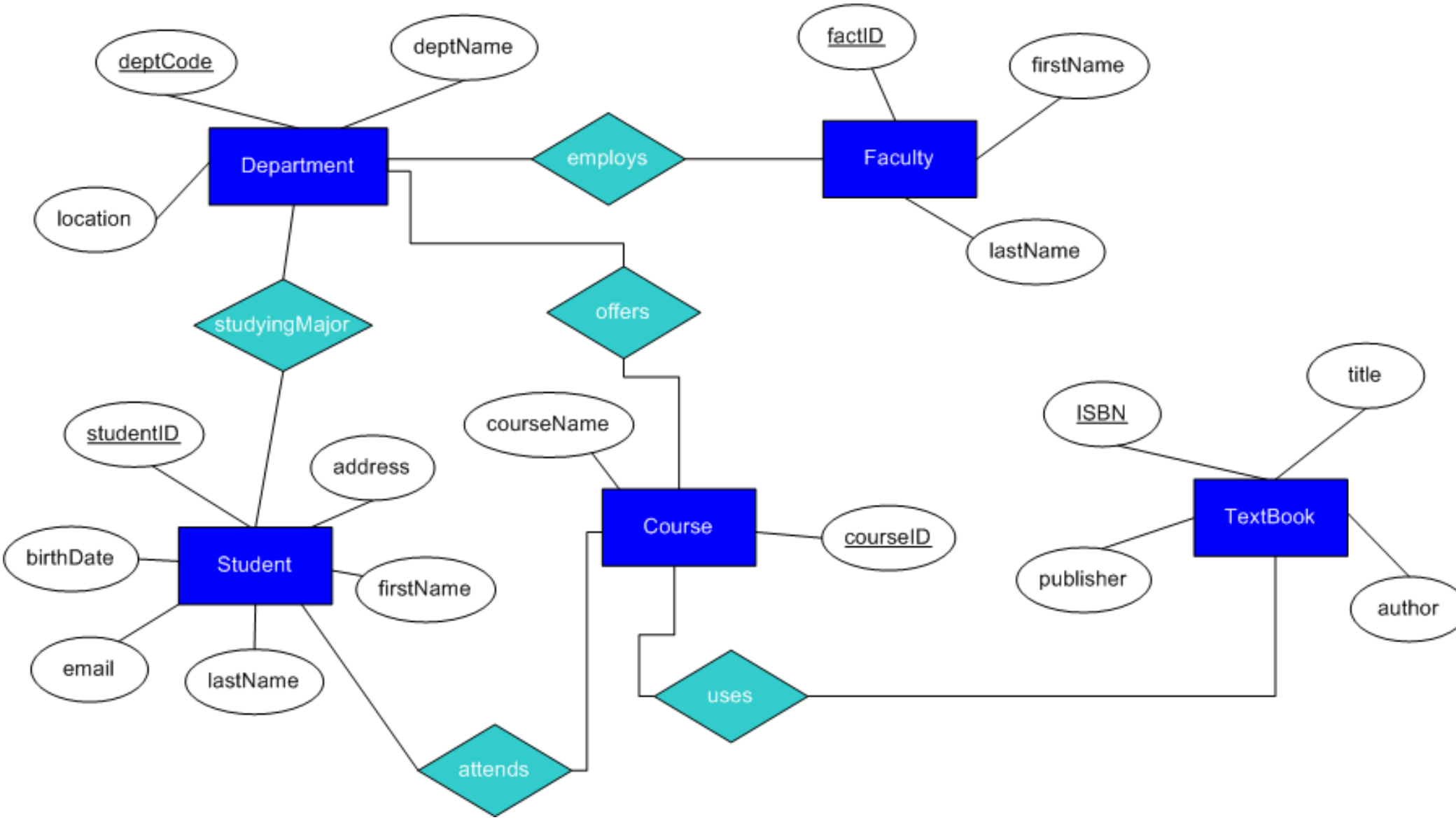
The Enhanced Entity-Relationship Model

Dr.Eng. RamezAlkhatib
ramezalkhatib@hotmail.com

Design Steps

Ideas $\Rightarrow$ E/R Design $\Rightarrow$ Relational Schema $\Rightarrow$ Relational DBMS

ER-Diagram for a University Database

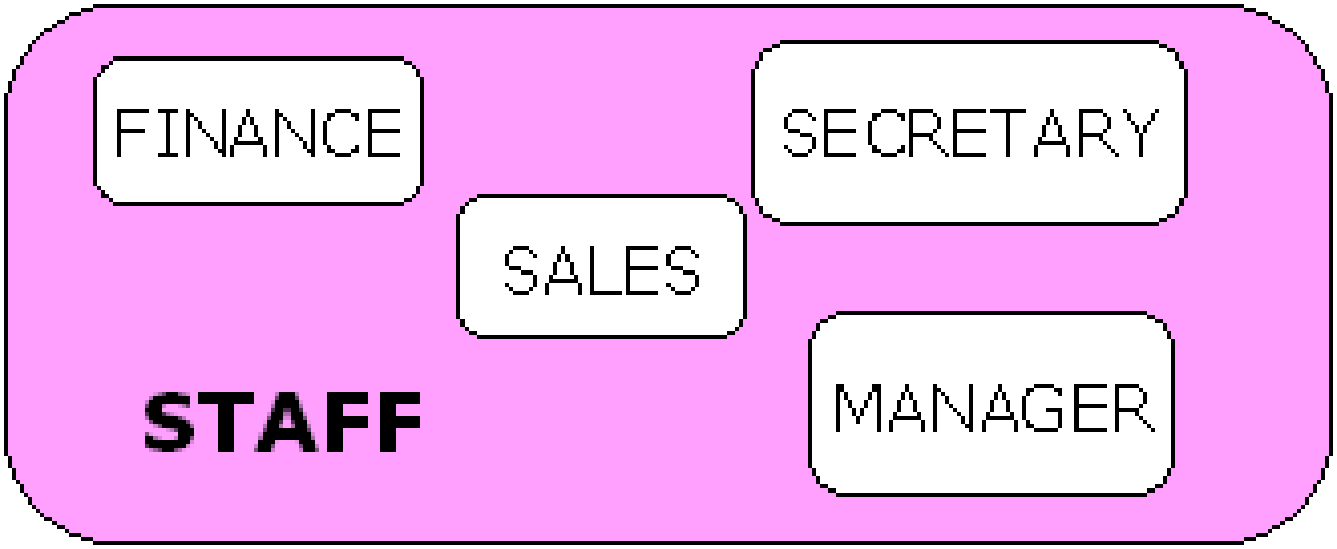


Enhanced Entity-Relationship (EER) Model

- The basic concepts of ER modelling is not powerful enough for some complex applications.
- We require some additional modelling concepts to represent **more semantic information** than the standard ER model.
- The **Enhanced Entity-Relationship (EER)** model extends the ER model to include various types of **abstraction** and to express **constraints** more clearly.
- Additional concepts in EER-Model:
  - **Specialization**
  - **Generalization**
  - **Categorization**
  - **Inheritance**
- Some of the concepts correspond to concepts in **object-oriented design**.

Superclass and Subclass

- **Example:** Staff may be grouped into Manager, Sales personnel, Finance Personnel, and Secretary.



  - Each of these groupings is a **subset** of `Staff` entities.
  - Each of these groupings is called **subclass** of `Staff`.
  - `Staff` is the **superclass** for each of these subclasses.

- **Superclass**: an entity type that includes distinct subclasses that require to be represented in a data model.

- **Subclass**: an entity type that has a distinct role and is also a member of a superclass.

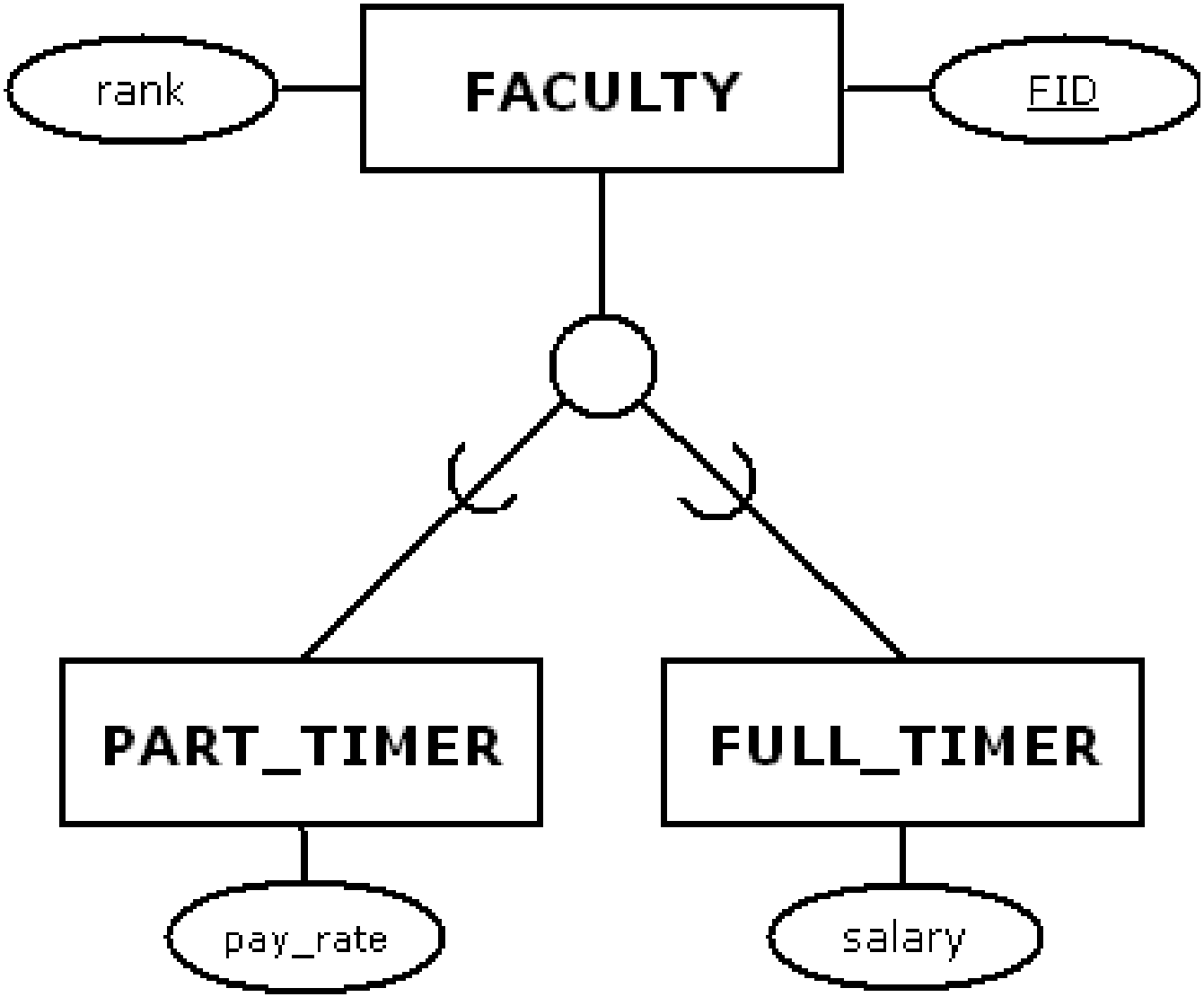
Inheritance

- An entity that is a member of a subclass **inherits** all **attributes** of the entity as a member of the super class.
- A subclass may optionally have distinct **local attributes**.
- A subclass also **inherits** all **relationships**.
- Sometimes the relationship superclass and subclass is referred to as an **isa relationship**.

Specialization

- **Specialization** is the process of defining a set of subclasses of a superclass.
- The set of subclasses is based upon some **distinguishing characteristics** of the entities of the superclass.
- Specialization is represented in EER-diagrams by
  - A **specialization circle** connected by a line to a superclass.
  - Each subclass is connected to the circle by a line having an **inheritance symbol**,i.e. a subset symbol or cup, with the open side facing the super class.

Example of Specialization



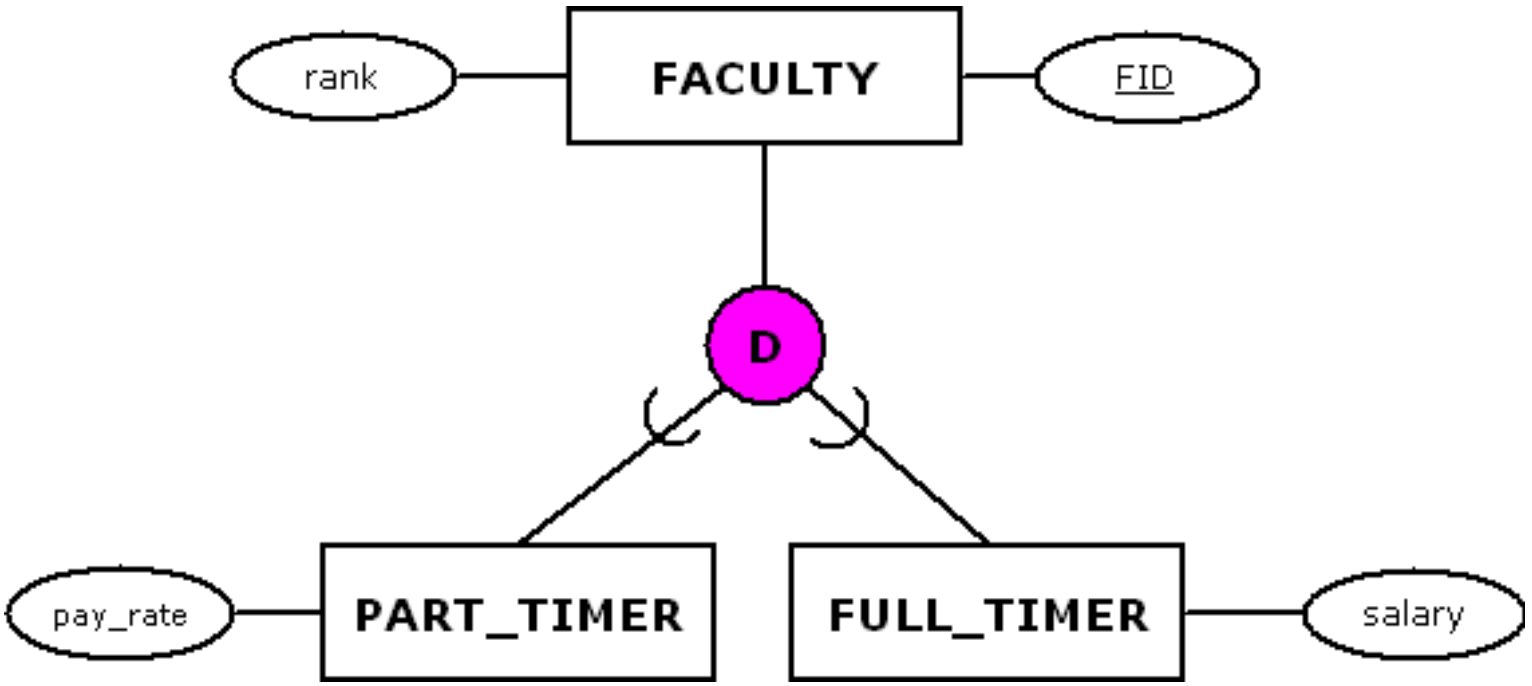
Generalization

- **Generalization** is the **reverse** of the specialization process.
- Several classes with **common features** are generalized into a superclass.
- **Example:**
  - Students and faculty both have IDs and names as attributes.
  - The two entities can be generalized into a new superclass `Person` having the common attributes.
  - The subclasses `Student` and `Faculty` would each retain their special attributes.
- Using generalization, we develop diagrams from **the bottom up**.
- Using specialization, we develop diagrams from the **top down**.

Constraints on Specialization and Generalization: Disjointness

- **Disjointness Constraint**: specifies that the subclasses of the specialization must be disjointed, i.e. an entity can be a member of at most one of the subclasses.
- The disjointness constraint is specified by `D` or `O` in the EER-diagram:
  - If the constraint is satisfied, then we write a `D` in the specialization circle.
  - If the constraint is not satisfied, then we have **overlapping** subclasses. We write an `O` in the circle.

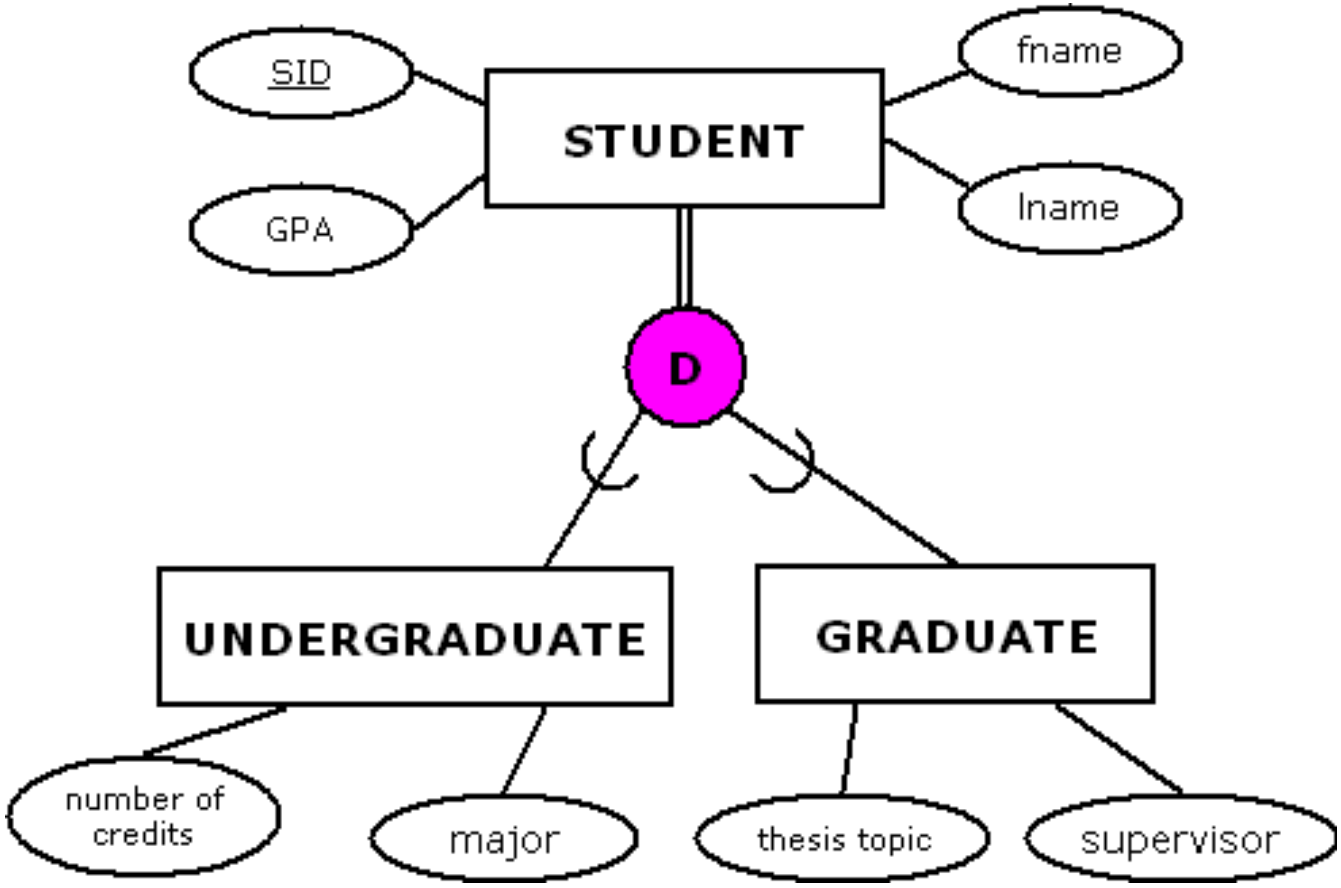
Example of Disjointness



Constraints on Specialization and Generalization: Completeness

- A specialization has also a **completeness constraint**, which shows whether every member of the entity set must participate in it.
- If every member of the superclass must belong to some subclass, we have **total** specialization.
- If some superclass members can be permitted not to belong to any subclass the specialization is **partial**.
- The completeness constraint is specified in the EER-diagram:
  - Total specialization: double line connecting the superclass to the specialization circle
  - Partial specialization: single line connecting the superclass to the the specialization circle

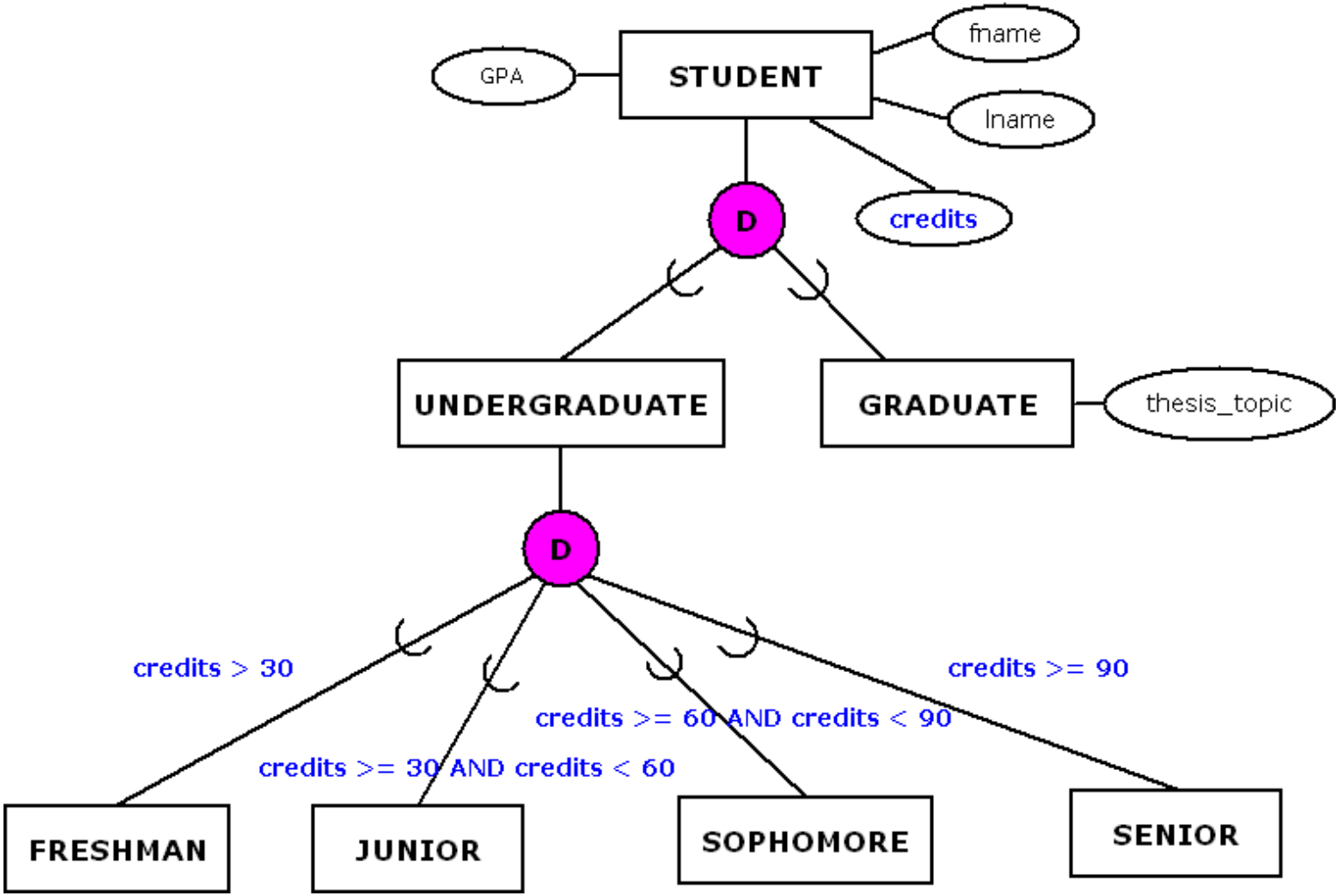
Example of Completeness



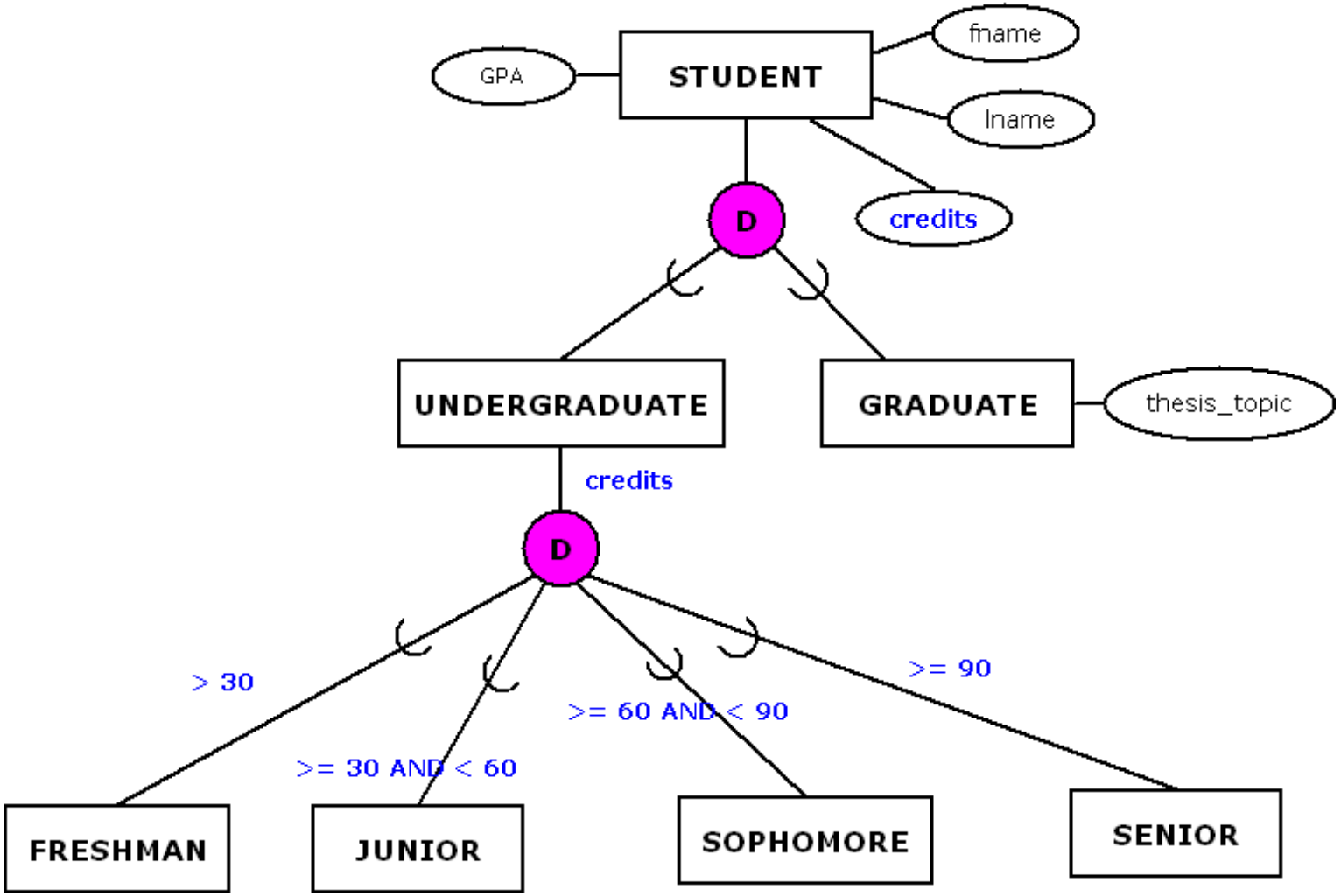
Constraints on Specialization: Predicate-defined

- In specialization hierarchies, it is possible to identify the subclass that an entity belongs to by checking a specific condition or predicate for each subclass.
- The specialization is called **predicate-defined specialization.**
- For the predicate-defined specialization: On an EER-diagram, we can write the defining predicate on the line from the specialization circle to the subclass.
- If the predicate uses the same attribute for all subclasses, we call the specialization an **attribute-defined specialization.**
- For the attribute-defined specialization: On an EER-diagram, we indicate the defining attribute on the line from the circle to the superclass, and the distinguishing value for each subclass on the line from the subclass to the circle.
- Specialization that are not predicate-defined are called **user-defined specialization.**

Example of Predicate-defined Specialization



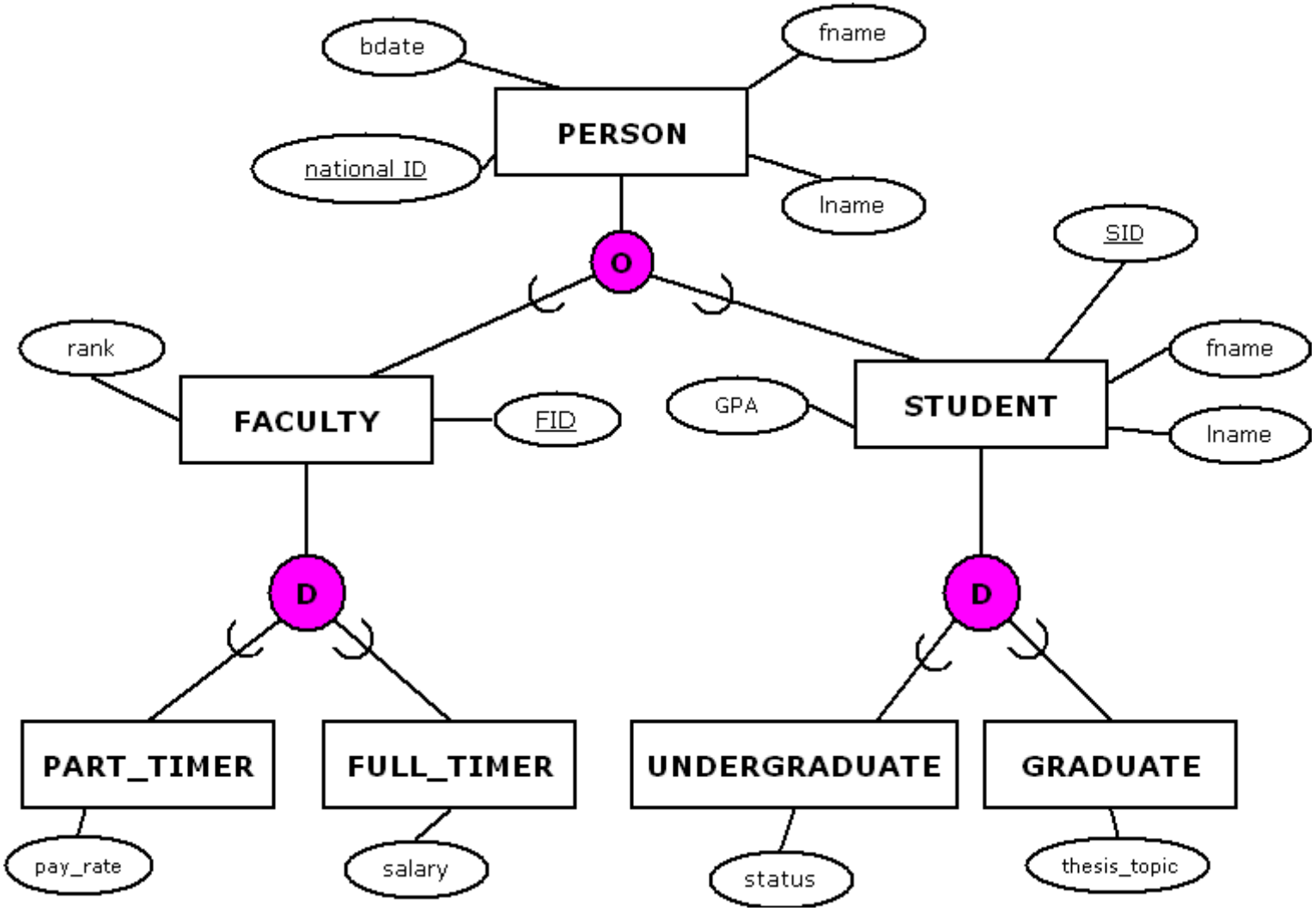
Example of Attribute-defined Specialization



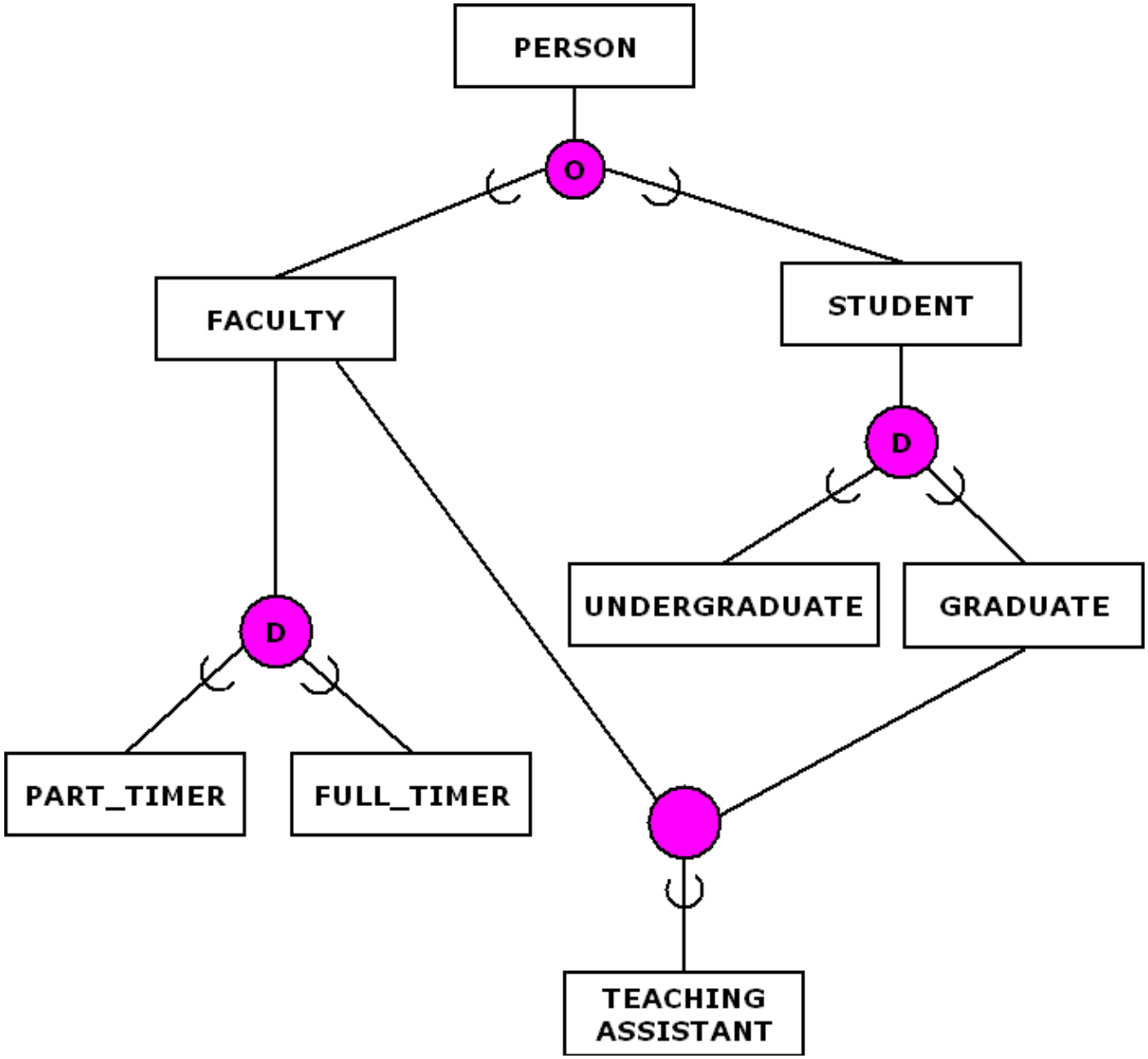
Hierarchies and Lattices

- A subclass may itself have further subclasses. This forms a **hierarchy** or a **lattice.**
- Hierarchy has a constraint that every subclass has only one superclass: **Single inheritance**.
- In a lattice, a subclass can have more than one superclass: **Multiple inheritance**.
- A subclass with more than one superclass is called **shared subclass**.
- In a lattice or hierarchy, a subclass inherits attributes not only of its direct subclass, but also of all its predecessor superclasses.

Example of Hierarchy



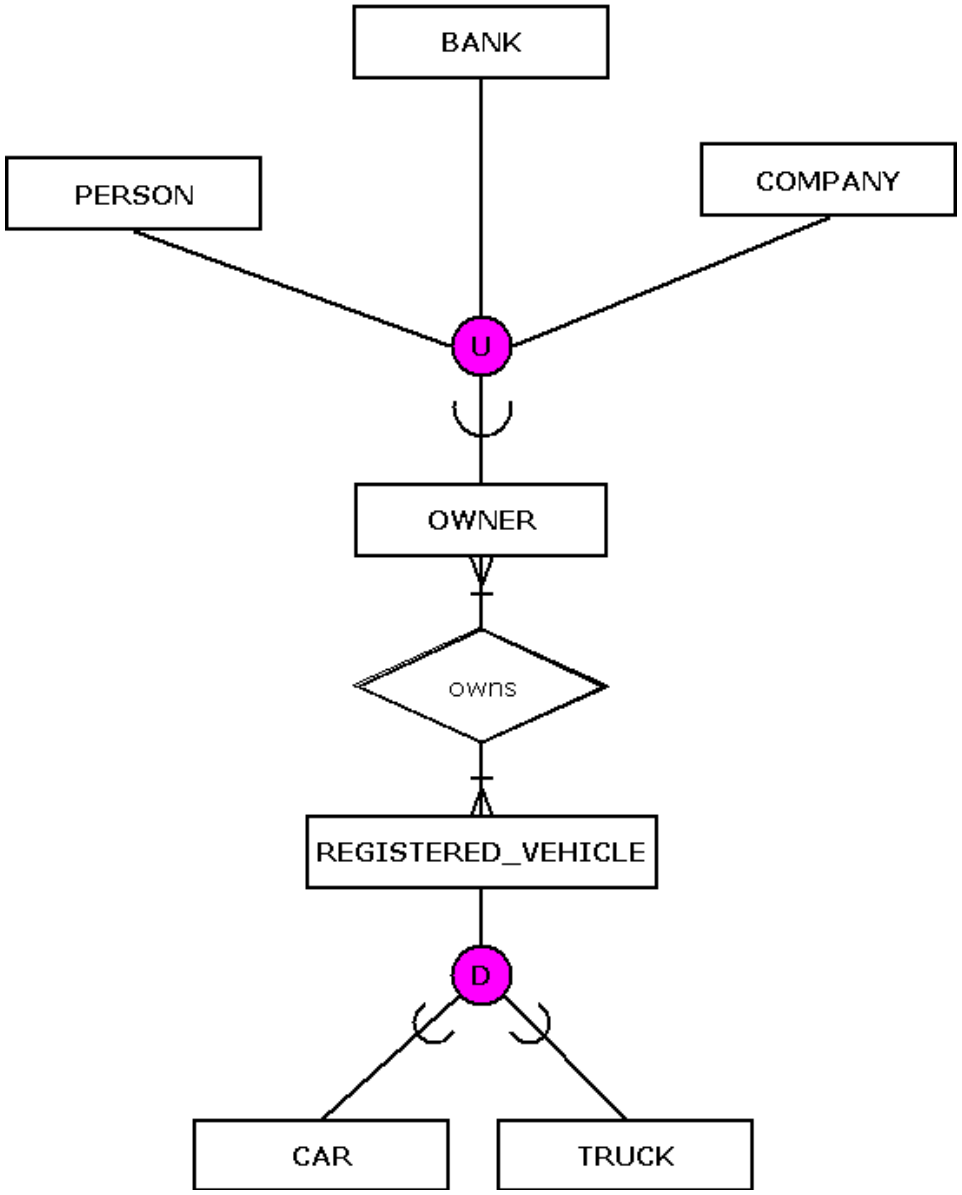
Example of Lattice



Categories: Union

- A shared subclass is a subclass having more than one superclass.
- A shared subclass inherits all attributes from all the superclasses.
- In some cases, we need to model a single superclass/subclass relationship with more than one superclass.
- In other words, the subclass should be related to one of the collection, called a **union** or **category**.
- An instance of the subclass inherits the attributes of only one of the superclass depending on which member of the union that it belongs to.
- **Example:** A database for vehicle registration: vehicle owner can be a person, a bank, or a company.
  - Category (subclass): `Owner` is a subset of the union of the three superclasses `Company`, `Bank`, and `Person`.
  - A category member must exist in a least one of its superclasses.
- In an EER-diagram: we represent a category by a **union circle** connecting the subclass with its superclasses.

Example of Union



Summary – What you should remember!

- Why the ER model has been extended to EE-R model
- The meaning of generalization and specification
- How to represent a generalization hierarchy on EE-R diagram
- How to represent generalization constraints
- The meaning of union