Lecture 5

DATABASE MANAGEMENT SYSTEM

Extended Entity Relationship Model (EER) Concepts

- Includes all modeling concepts of basic ER
- Additional concepts:
  - subclasses/superclasses
  - specialization/generalization
  - Categories
  - attribute inheritance
- The resulting model is called the **enhanced-ER or Extended ER (E2R or EER) model**
- It includes some object-oriented concepts, such as inheritance

Subclass and Superclass – 1/4

- An entity type may have additional meaningful subgroupings of its entities
  - *Example*: EMPLOYEE may be further grouped into SECRETARY, ENGINEER, MANAGER,TECHNICIAN,SALARIED_EMPLOYEE, HOURLY_EMPLOYEE,…
- Each of these groupings is a subset of EMPLOYEE entities
- **Each is called a subclass of EMPLOYEE**

Subclass and Superclass – 2/4

- **EMPLOYEE is the superclass for each of these subclasses**
- These are called **superclass/subclass or class/subclass relationships**.

- Example:
  - EMPLOYEE/SECRETARY
  - EMPLOYEE/TECHNICIAN

Subclass and Superclass – 3/4

- These are also called **IS-A relationships** (SECRETARY IS-A EMPLOYEE, TECHNICIAN IS-A EMPLOYEE, …)
- **The Subclass member is the same entity in a distinct specific role**
- **An entity cannot exist in the database merely by being a member of a subclass; it must also be a member of the superclass**

Subclass and Superclass – 4/4

- **A member of the superclass can be optionally included as a member of any number of its subclasses**

- Example:
  - A salaried employee who is also an engineer belongs to the two subclasses ENGINEER and SALARIED_EMPLOYEE

Attribute Inheritance

- An entity that is member of a subclass ***inherits all*** attributes of the entity as a member of the superclass
- It also inherits all relationships

Specialization – 1/3

- It 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 in the superclass**

- Example:
  - {SECRETARY, ENGINEER, TECHNICIAN} is a specialization of EMPLOYEE based upon ***job type***

Specialization – 2/3

- May have several specializations of the same superclass

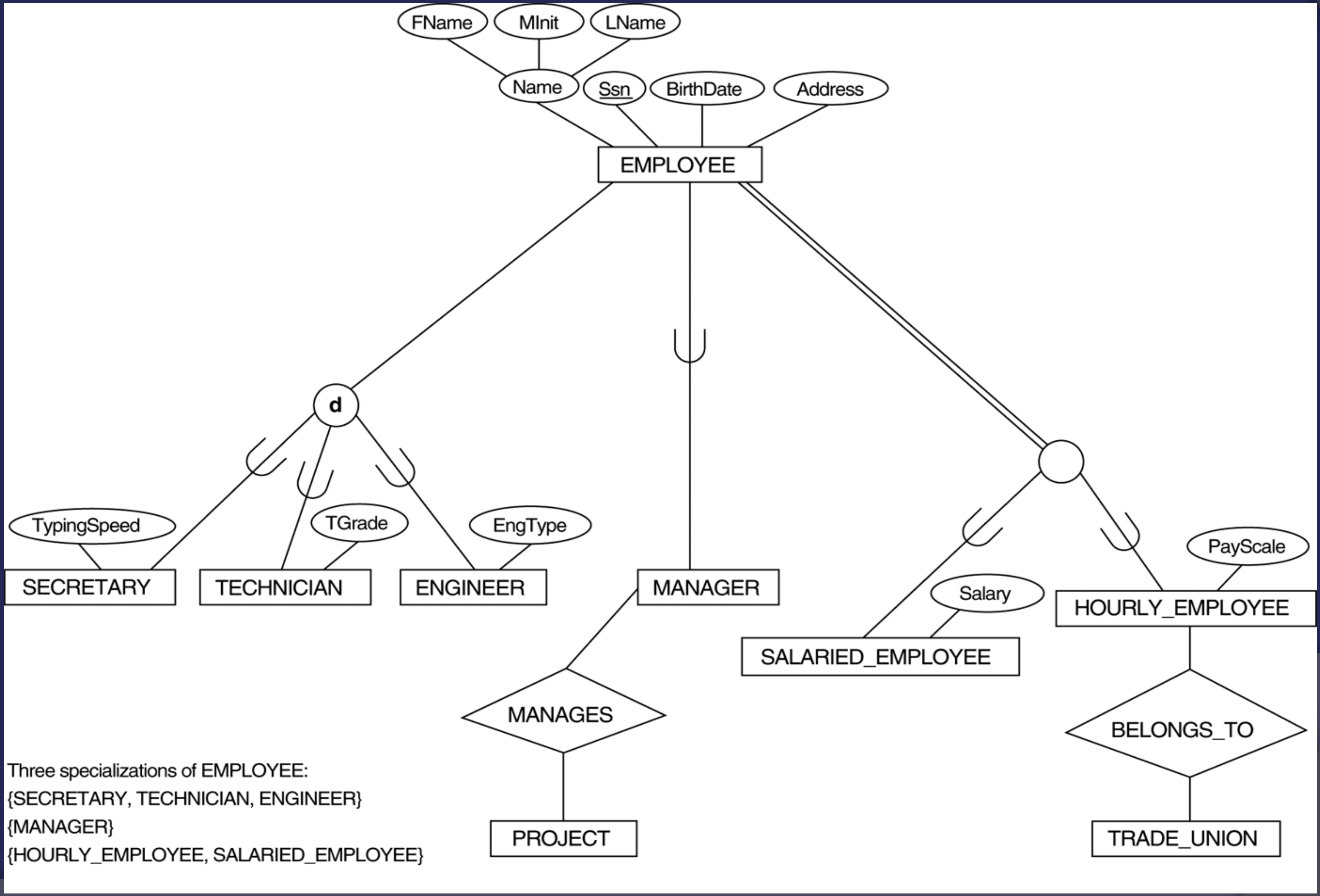
- Example:
  - Another specialization of EMPLOYEE based in ***method of pay*** *is {SALARIED_EMPLOYEE, HOURLY_EMPLOYEE}.*

Specialization – 3/3

- **Attributes of a subclass are called specific (or local) attributes**
  - For example, ***TypingSpeed*** of SECRETARY

- The subclass can participate in specific relationship types
  - For example, BELONGS_TO of HOURLY_EMPLOYEE

Example of Specialization



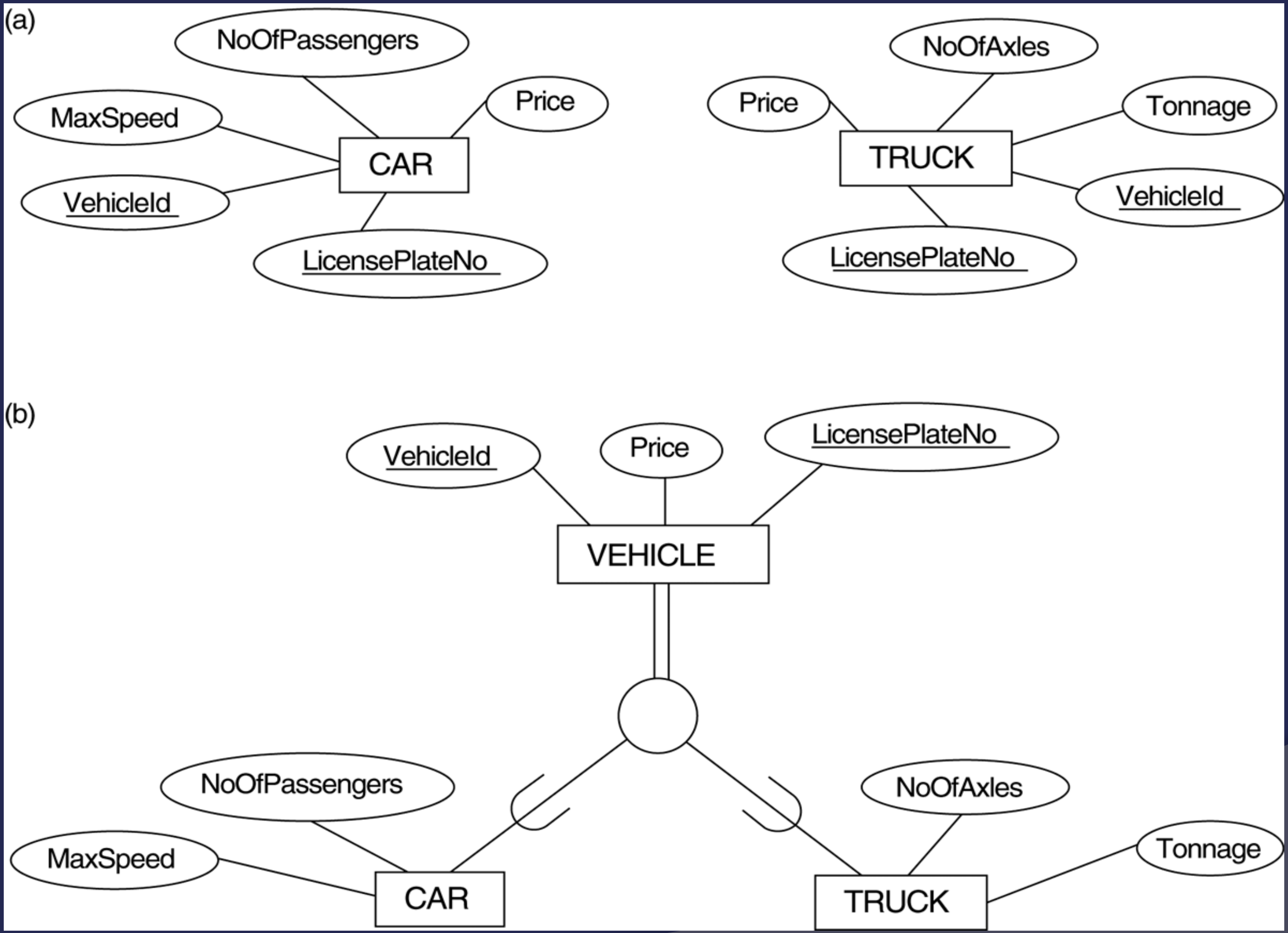
Three specializations of EMPLOYEE:
{SECRETARY, TECHNICIAN, ENGINEER}
{MANAGER}
{HOURLY_EMPLOYEE, SALARIED_EMPLOYEE}

Generalization

- **The reverse of the specialization process**
- Several classes **with common features** are generalized into a superclass; original classes become its subclasses
  - *Example*: CAR, TRUCK generalized into VEHICLE; both CAR, TRUCK become subclasses of the superclass VEHICLE.

- We can view {CAR, TRUCK} as a specialization of VEHICLE
- Alternatively, we can view VEHICLE as a generalization of CAR and TRUCK

Example of Generalization



Types of specializations – 1/2

1. If we can determine exactly those entities that will become members of each subclass by a condition, the subclasses are called ***predicate-defined (or* condition-defined) subclasses**

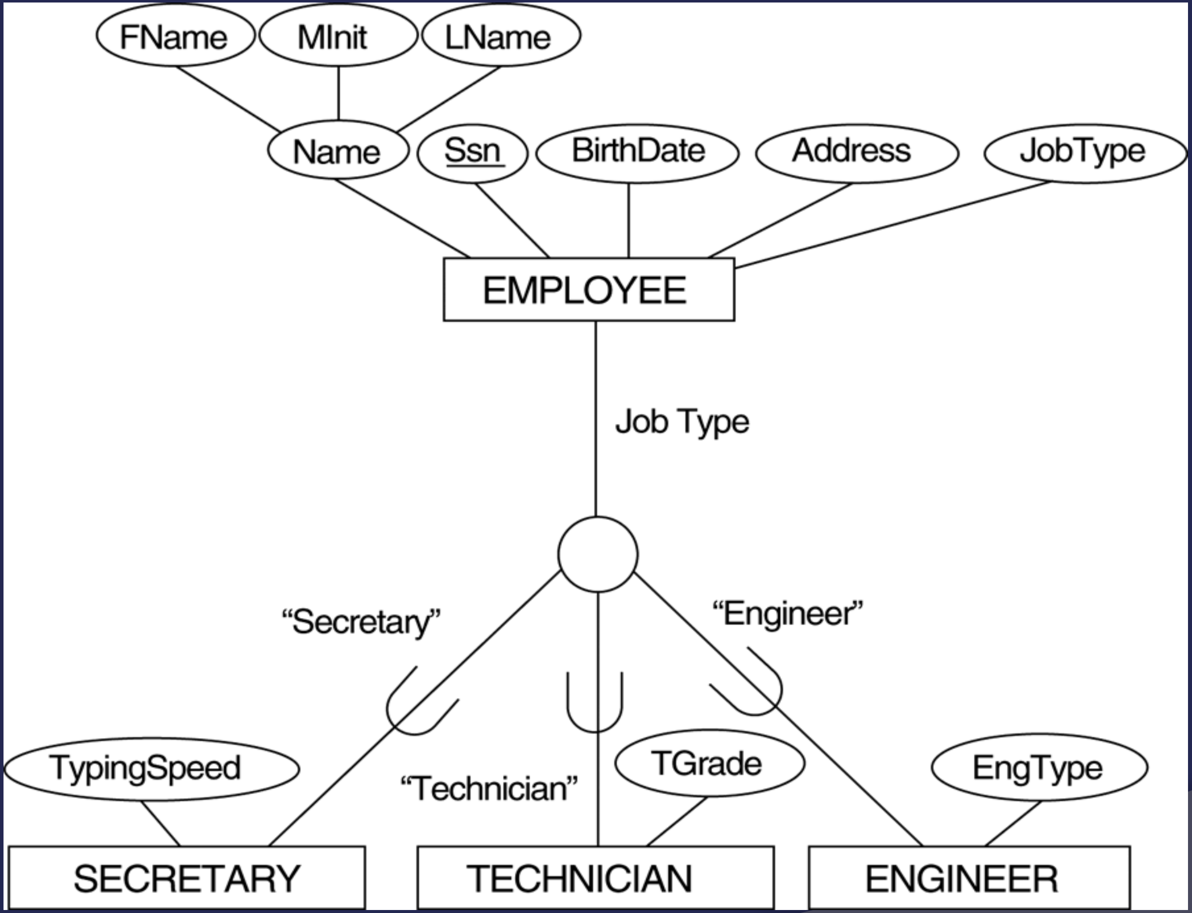
   - Condition is a **constraint** that determines subclass members
   - Display a predicate-defined subclass by writing the predicate condition next to the line attaching the subclass to its superclass

Types of specializations – 2/2

2. If all subclasses in a specialization have membership condition on same attribute of the superclass, specialization is called an ***attribute defined specialization***
   - Attribute is called the defining attribute of the specialization
   - *Example*: **JobType** is the defining attribute of the specialization {SECRETARY, TECHNICIAN, ENGINEER} of EMPLOYEE

3. **If no condition determines membership, the subclass is called *user-defined***

Attribute Defined Specialization



Constraints in Specialization/Generalization – 1/2

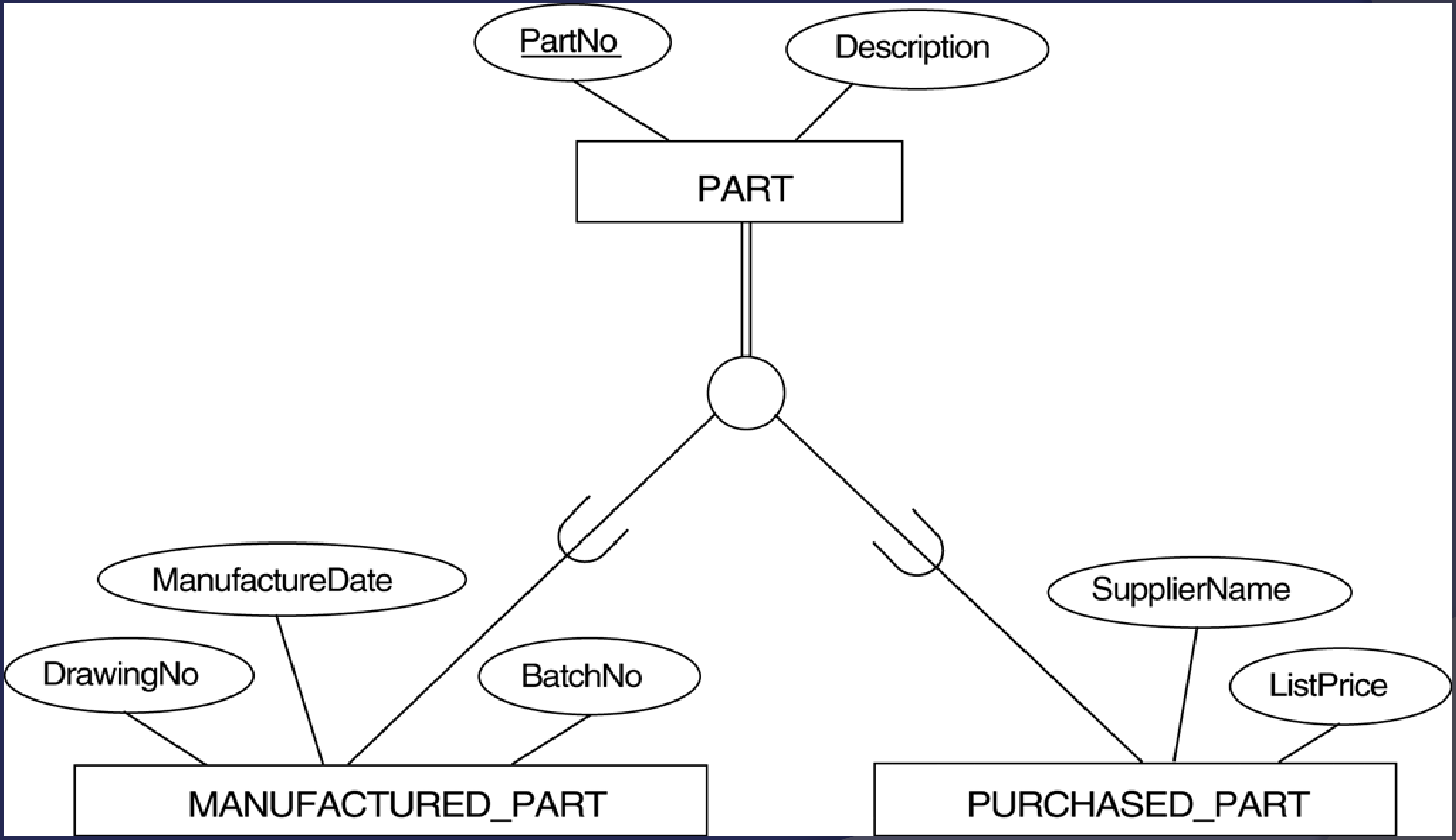
- **Disjointness Constraint:**
  - Specifies that the subclasses of the specialization must be **disjointed (an entity can be a member of <u>at most one</u> of the subclasses of the specialization)**
  - **Specified by d in EER diagram**

  - If not disjointed, **overlap; that is the same entity may be a member of <u>more than one</u> subclass of the specialization**
  - **Specified by O in EER diagram**

Constraints in Specialization/Generalization – 2/2

- **Completeness Constraint:**
  - Total specialization constraint
    - Every entity in the superclass must be a member of some subclass in the specialization/ generalization
    - Shown in EER diagrams by a double line
  - Partial specialization constraint
    - An entity may not belong to any of the subclasses
    - Shown in EER diagrams by a single line

Example - Overlapping specialization


PartNo
Description
PART
ManufactureDate
DrawingNo
BatchNo
MANUFACTURED_PART
SupplierName
ListPrice
PURCHASED_PART