Enhanced Entity-Relationship (EER) Modeling

Session Objectives

- ERR Diagram
- Superclass/Subclass
- Specialization /Generalization
- Disjoint/Overlap
- Total/ Partial

Session Outcomes

- At the end of this session, participants will be able to
  - Understand EER model concepts
  - Understand Superclass/Subclass
  - Understand Generalization/Specialization
  - Understand Disjoint/Overlap

Enhanced Entity Relational Model

Dr. P.Mirunalini
SSNCE

April 06, 2022

EER MODEL

- EER stands for **Enhanced ER or Extended ER**
- Current applications such as CAD/CAM,telecommunications, GIS,... have more complex requirements
- Led to the development of semantic data modelling concepts
- ER model can enhanced to include **semantic data model** leading to Enhanced ER [EER] model

EER MODEL

- EER Model Concepts
  - Includes all modeling concepts of basic ER
  - Additional concepts:
    - Subclasses/superclasses
    - Specialization/generalization
    - Attribute and relationship inheritance
- The additional EER concepts are used to model applications more completely and more accurately
- EER includes some object-oriented concepts, such inheritance
- The diagrammatic technique – EER diagrams

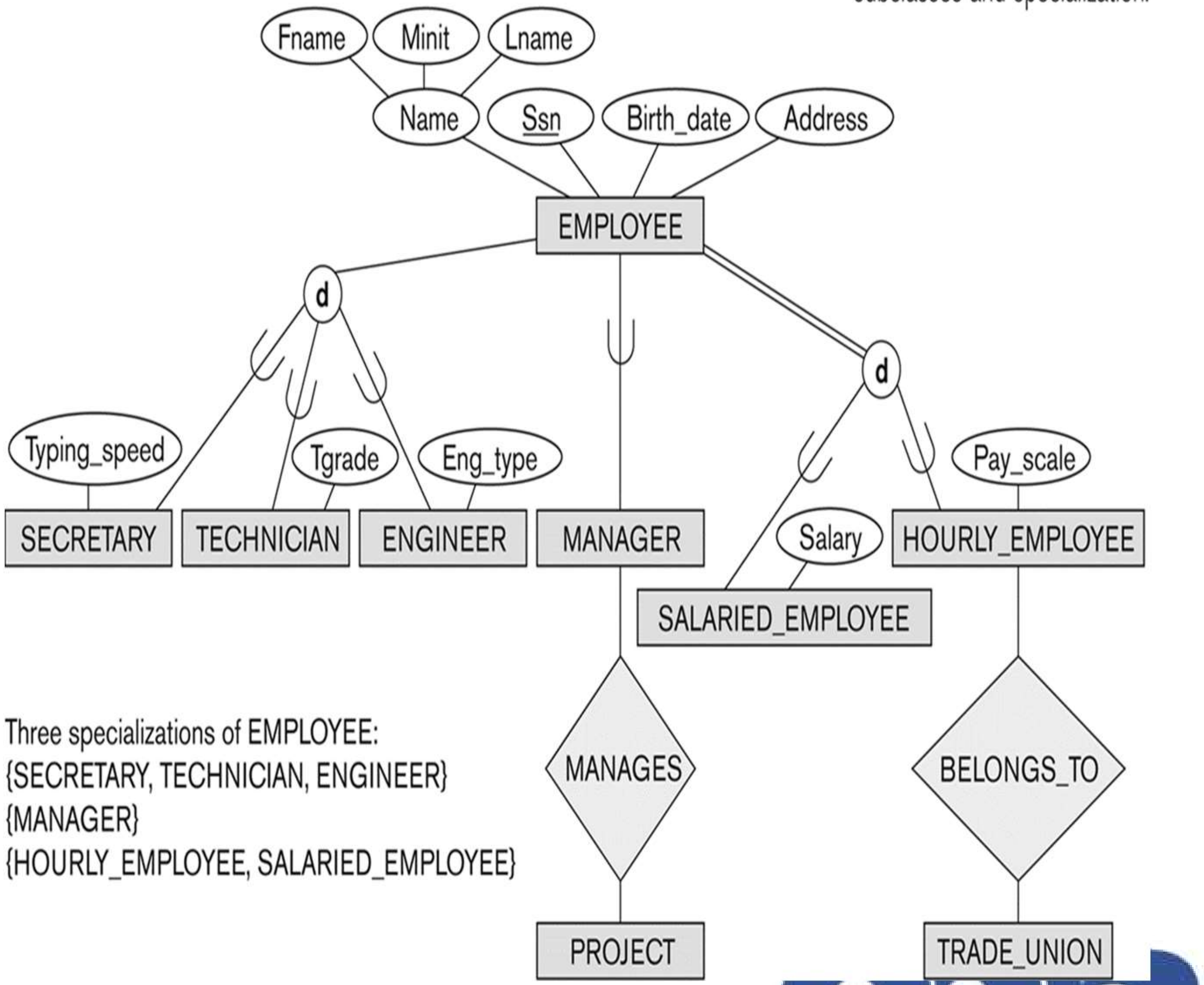
Superclass and Subclass

- An entity type may have additional meaningful subgroupings of its entities
- Need to represent explicitly because of their significance to the database applications
  - Example: EMPLOYEE may be further grouped into:
    - SECRETARY, ENGINEER, TECHNICIAN, …
      - Based on the EMPLOYEE's Job
    - MANAGER
      - EMPLOYEEs who are managers
    - SALARIED_EMPLOYEE, HOURLY_EMPLOYEE
      - Based on the EMPLOYEE's method of pay
  - The set or collection of entities in each of the groupings is the subset of entities that belong to the employee entity set
  - Every entity that is a member of one of these subgroupings is also an employee
  - Each of the subgroupings is called subclass or subtype

Superclass and Subclass

- EER diagrams extend ER diagrams to represent these additional subgroupings, called *subclasses* or *subtypes*
- The subclass that define a specialization are attached by lines to the circle which is connected to the superclass
- An entity that is member of a subclass inherits all attributes of the entity as a member of the superclass
- All relationships of the entity as a member of the superclass

Figure 4.1
EER diagram notation to represent subclasses and specialization.



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

Subclasses and Superclasses

- Each of these subgroupings is a subset of EMPLOYEE entities
- Each is called a **subclass** of EMPLOYEE and EMPLOYEE is the **superclass** for each of these subclasses
- The relationship between a superclass and any of the subclasses as superclass/subclass or class/subclass relationship.
- Example superclass/subclass relationships :
  - **EMPLOYEE/SECRETARY**
  - **EMPLOYEE/TECHNICIAN**
  - **EMPLOYEE/MANAGER**
- Important Concept associated with subclasses is the type of inheritance.
- An entity in the subclass represents the same real world entity from the superclass posses values for the specific attributes as well values of its attributes of the superclass

Subclasses and Superclasses

These are also called IS-A relationships

- SECRETARY IS-A EMPLOYEE, TECHNICIAN IS-A EMPLOYEE, ….

Note: An entity that is member of a subclass represents the same real-world entity as some member of the superclass:

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

Subclasses and Superclasses

- May have several specializations of the same superclass
- A salaried employee who is also an engineer belongs to the two subclasses:

  ENGINEER,

  SALARIED_EMPLOYEE
- A salaried employee who is also an engineering manager belongs to the three subclasses:

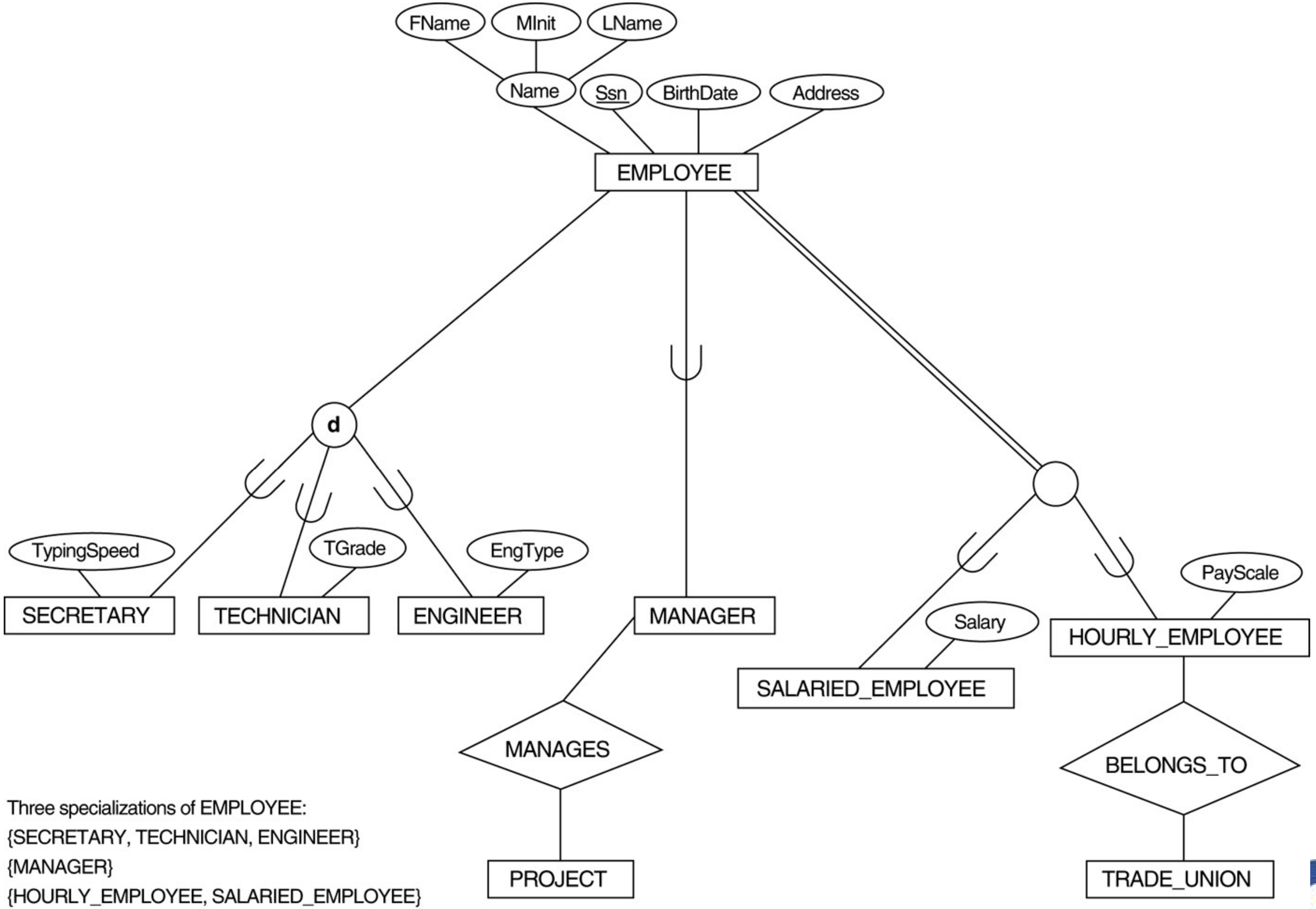
  MANAGER,

  ENGINEER, and

  SALARIED_EMPLOYEE
- SECRETARY (as well as TECHNICIAN and ENGINEER) inherit the attributes Name, SSN, …, from EMPLOYEE

It is not necessary that every entity in a superclass be a member of some subclass

Subclasses and Superclasses - Example



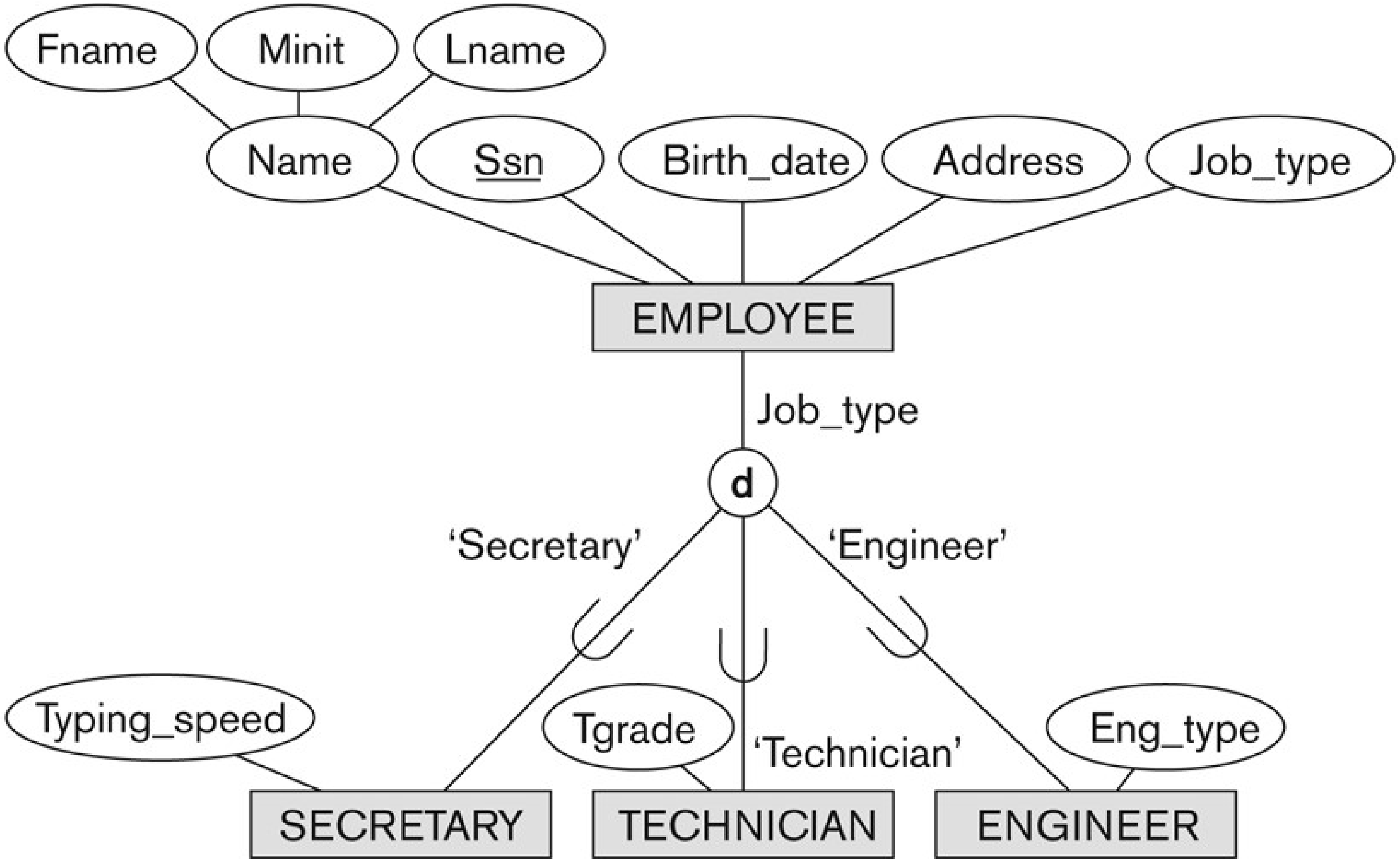
Three specializations of EMPLOYEE:
{SECRETARY, TECHNICIAN, ENGINEER}
{MANAGER}
{HOURLY_EMPLOYEE, SALARIED_EMPLOYEE}

Specialization

- Specialization is the process of defining a set of subclasses of a superclass
  - Ex: Subclasses {Secretary, Engineer and Technician) is a specialization of the superclass employee based on the job type
  - based upon some **distinguishing characteristics** of the entities in the superclass
  - attributes of a subclass are called specific or local attributes
  - the subclass can also participate in specific relationship types

Representing Specialization in EER Diagrams

Figure 4.4
EER diagram notation for an attribute-defined specialization on Job_type.



Several specializations of the same entity type based on the distinguishing characteristics.

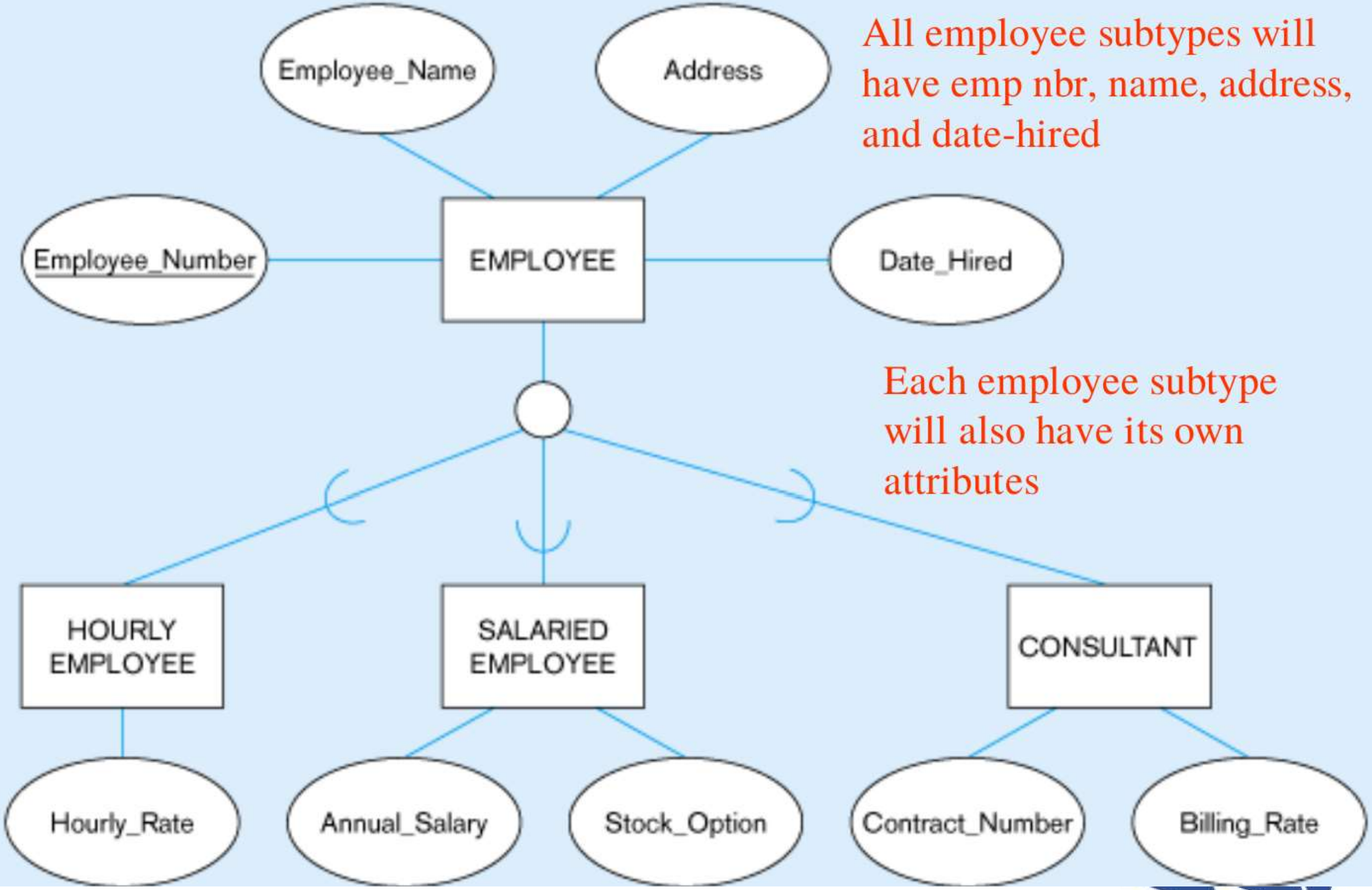
Specialization

- 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*.
- Example: Another specialization of EMPLOYEE based on *method of pay* is {SALARIED_EMPLOYEE,HOURLY_EMPLOYEE}.
- Attributes that apply to entities of particular subclass are called specific attributes. TypingSpeed of SECRETARY.
- The subclass can participate in specific relationship types.
- Superclass/subclass relationship resembles a 1:1 relationship at the instance level.

Specialization

- Two main reasons for including class/subclass relationships and specializations:
  - Certain attributes may apply to some but not all entities of the superclass entity type.
  - A subclass is defined in order to group the entities to which the attributes apply.
  - Members of the subclass may share the majority of their attributes with other members of the superclass.
  - Eg: secretary subclass has specific attribute typing_speed but inherted attributes from employee entity
  - Some realtion types may be participated in only by entities that are members of the subclass

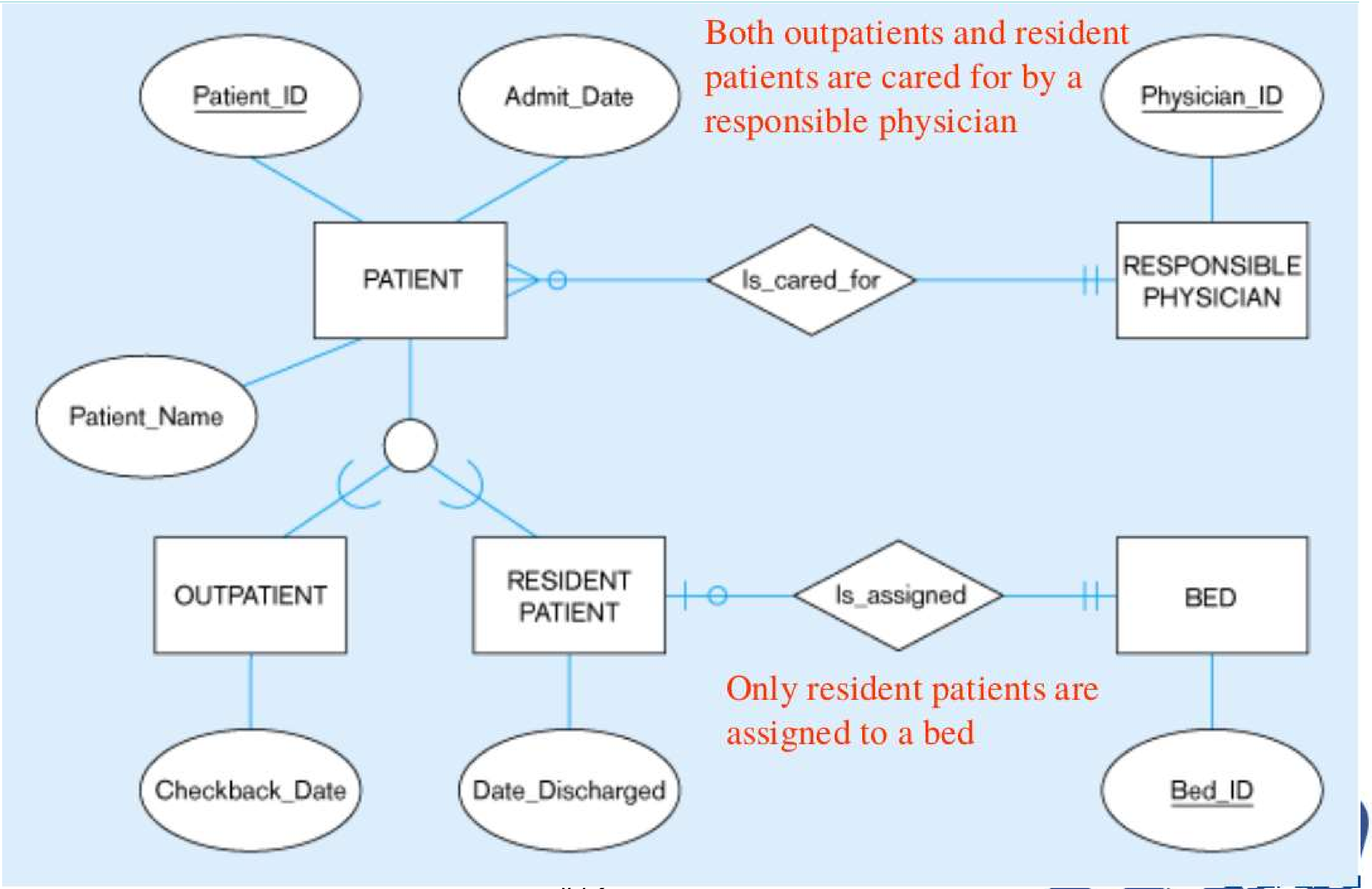
Specialization



All employee subtypes will have emp nbr, name, address, and date-hired

Each employee subtype will also have its own attributes

Specialization - Example



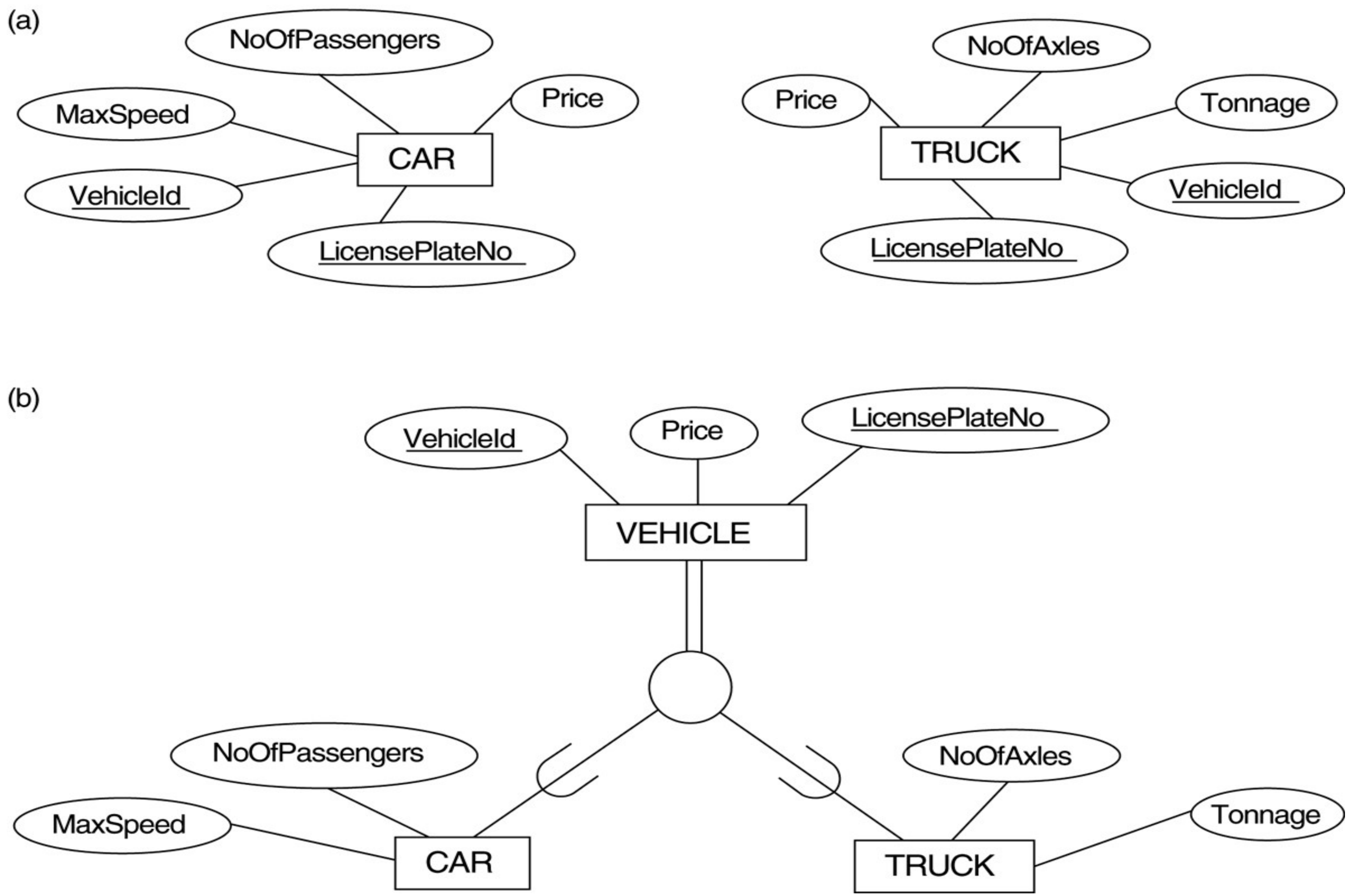
Both outpatients and resident patients are cared for by a responsible physician

Only resident patients are assigned to a bed

Generalization

- Generalization is the reverse of the specialization process
- Several classes with common features are generalized into a superclass
- Example: CAR, TRUCK generalized into VEHICLE;
- Both CAR, TRUCK become subclasses of the superclass VEHICLE.
- Diagrammatic notation are sometimes used to distinguish between generalization and specialization – but it is subjective

Generalization. (a) Two entity types, CAR and TRUCK. (b) Generalizing CAR and TRUCK into the superclass VEHICLE.



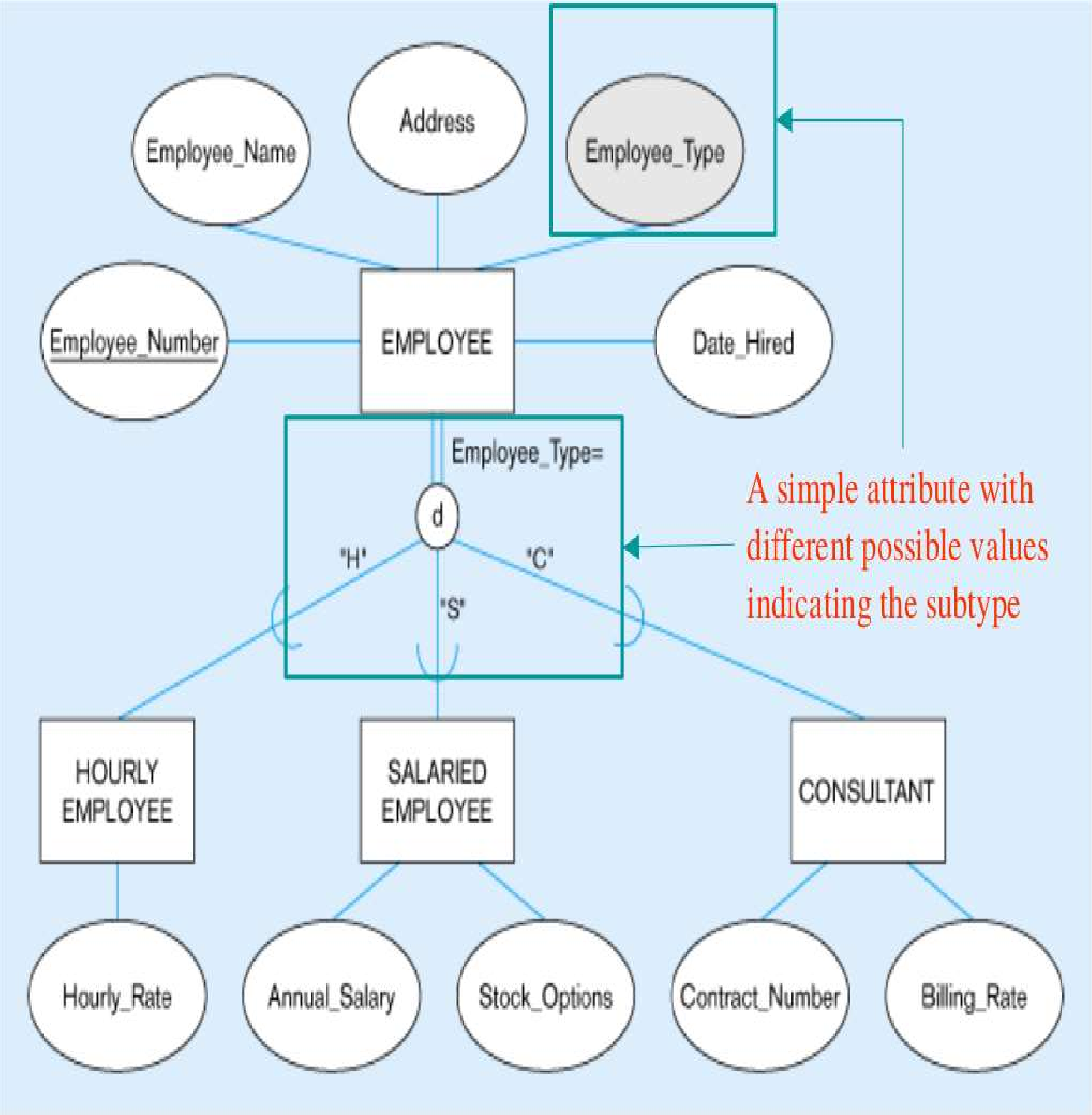
Constraints on Specialization/Generalization

Two basic constraints can apply to a Specialization/generalization:

– Disjointness Constraint:

– Completeness Constraint:

Predicate Defined Subclasses

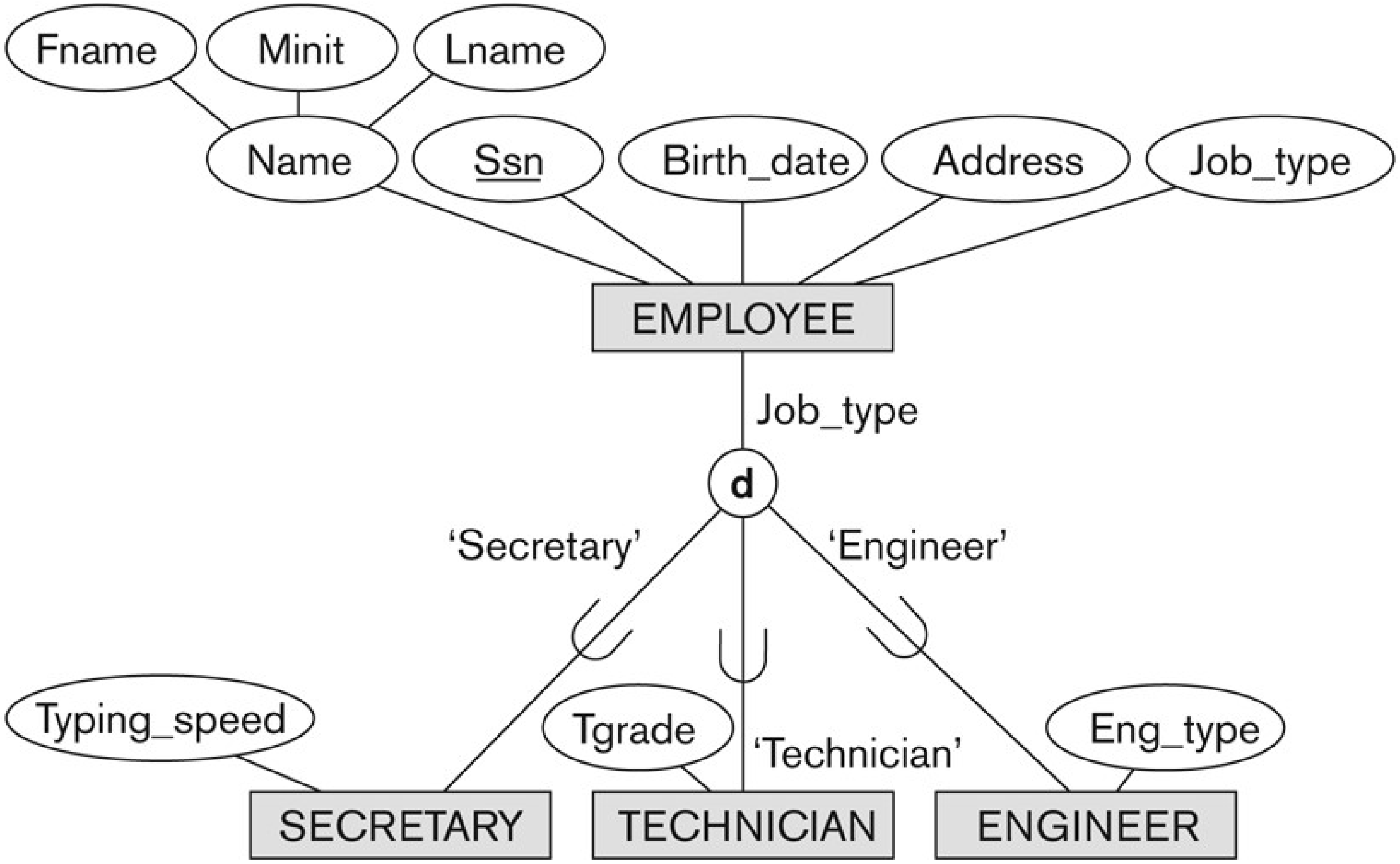


A simple attribute with different possible values indicating the subtype

Specializations that become a member of each of the subclass by placing a **condition** on the value of the attributes . Such subclasses are called as **predicate-defined subclasses.**

Attribute Defined subclasses

Figure 4.4
EER diagram notation for an attribute-defined specialization on Job_type.



Specializations have the membership condition on the same attribute of the superclass , the specialization is **attribute defined**

Constraints on Specialization/Generalization

Disjointness Constraint:

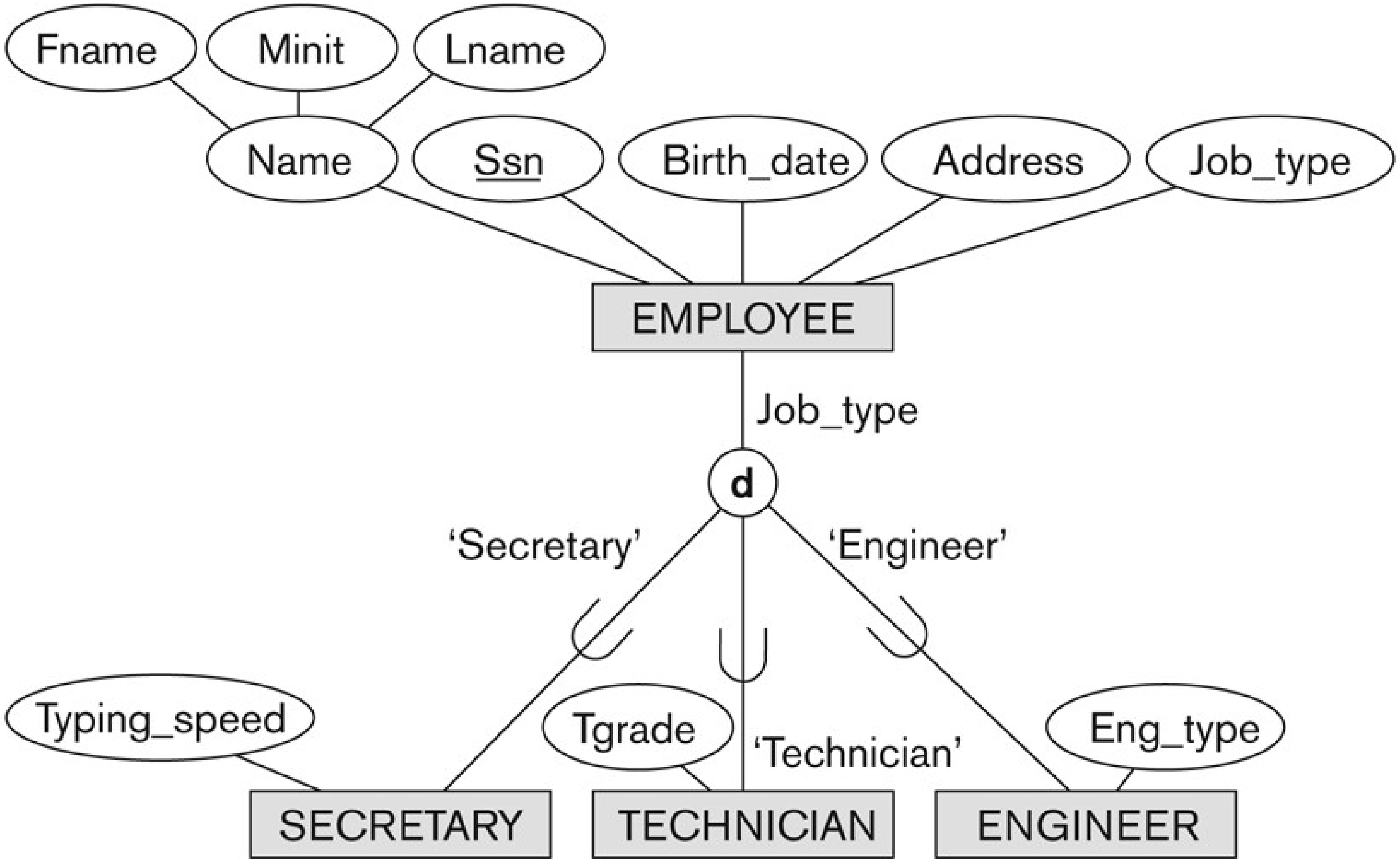
– Specifies that the subclasses of the specialization must be disjoint set:

- an entity can be a member of at most one of the subclasses of the specialization
- Entity can be a member of atmost one of the subclasses of the specialization.
- Specified by d in EER diagram
- The specilaization that is attribute defined implies the disjointness constraint

–

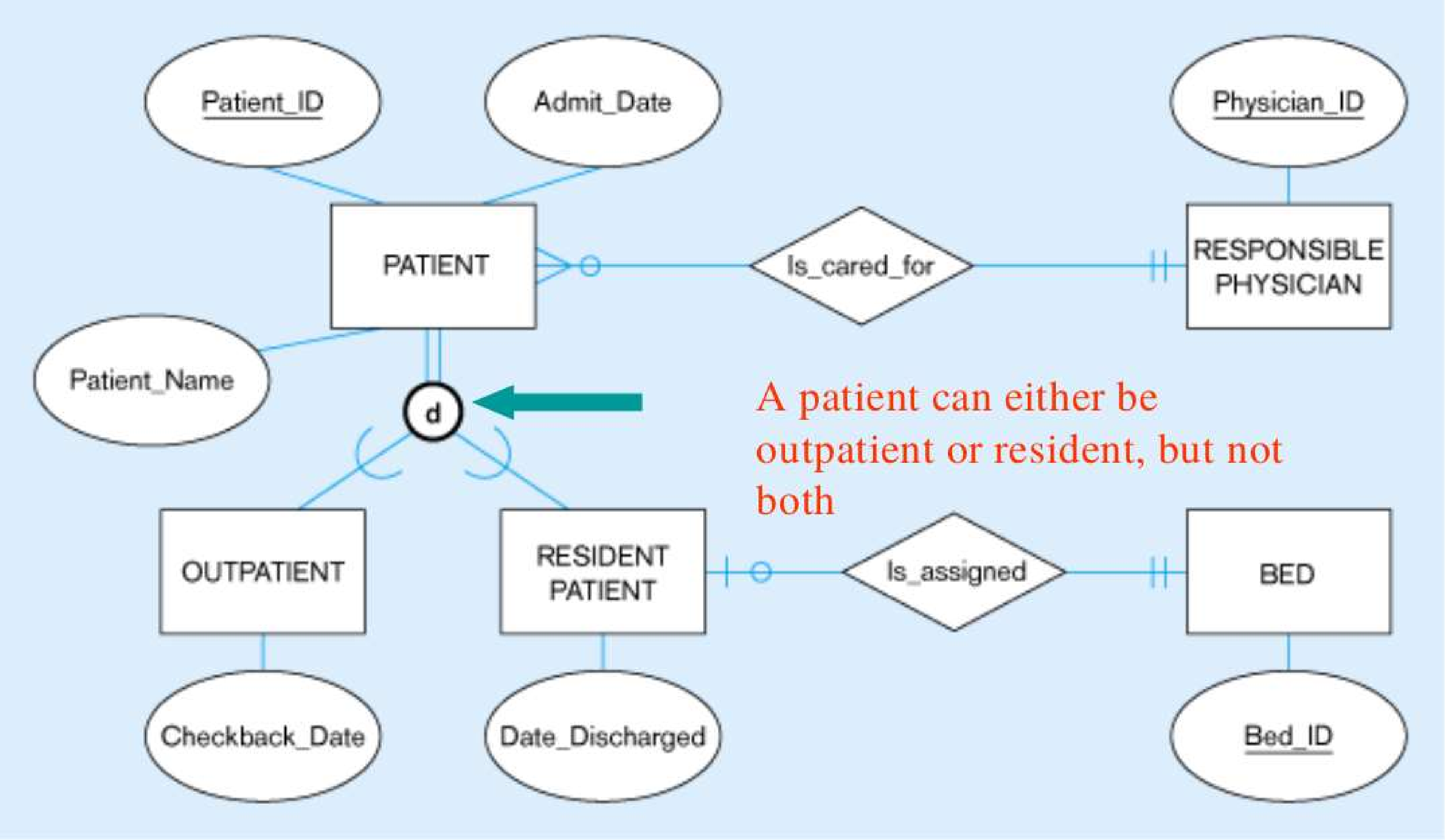
Attribute Defined subclasses

Figure 4.4
EER diagram notation for an attribute-defined specialization on Job_type.



Specializations have the membership condition on the same attribute of the superclass , the specialization is **attribute defined**

Disjoint



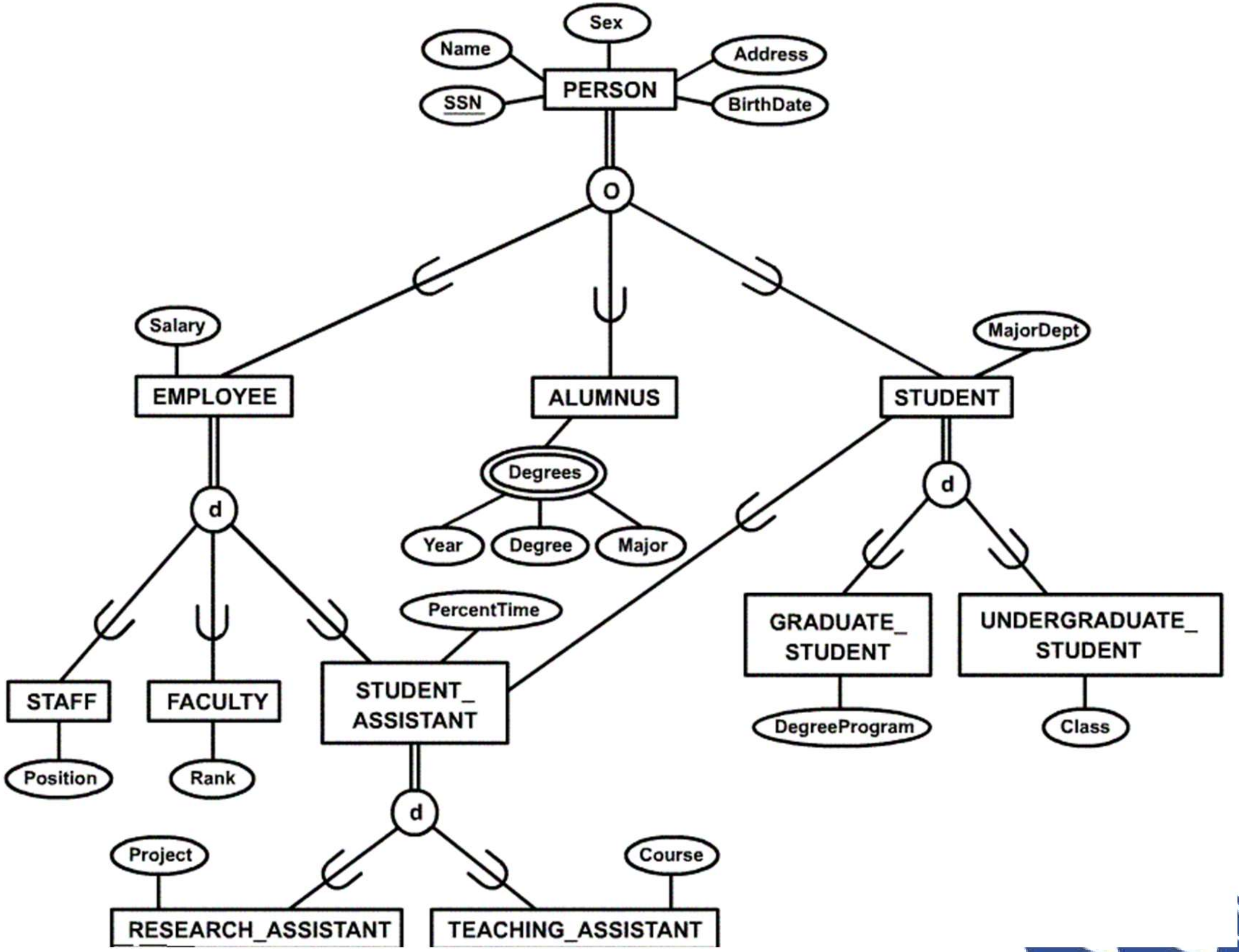
A patient can either be outpatient or resident, but not both

Constraints on Specialization/Generalization

– If not disjoint, specialization is overlapping:

- that is the same entity may be a member of more than one subclass of the specialization
- Same entity may be the member of more than one subclass of the specialization.
- Specified by o in EER diagram

Overlap


Sex
Name
Address
PERSON
SSN
BirthDate
o
Salary
MajorDept
EMPLOYEE
ALUMNUS
STUDENT
Degrees
d
d
Year
Degree
Major
PercentTime
GRADUATE_ STUDENT
UNDERGRADUATE_ STUDENT
STAFF
FACULTY
STUDENT_ ASSISTANT
DegreeProgram
Class
Position
Rank
d
Project
Course
RESEARCH_ASSISTANT
TEACHING_ASSISTANT

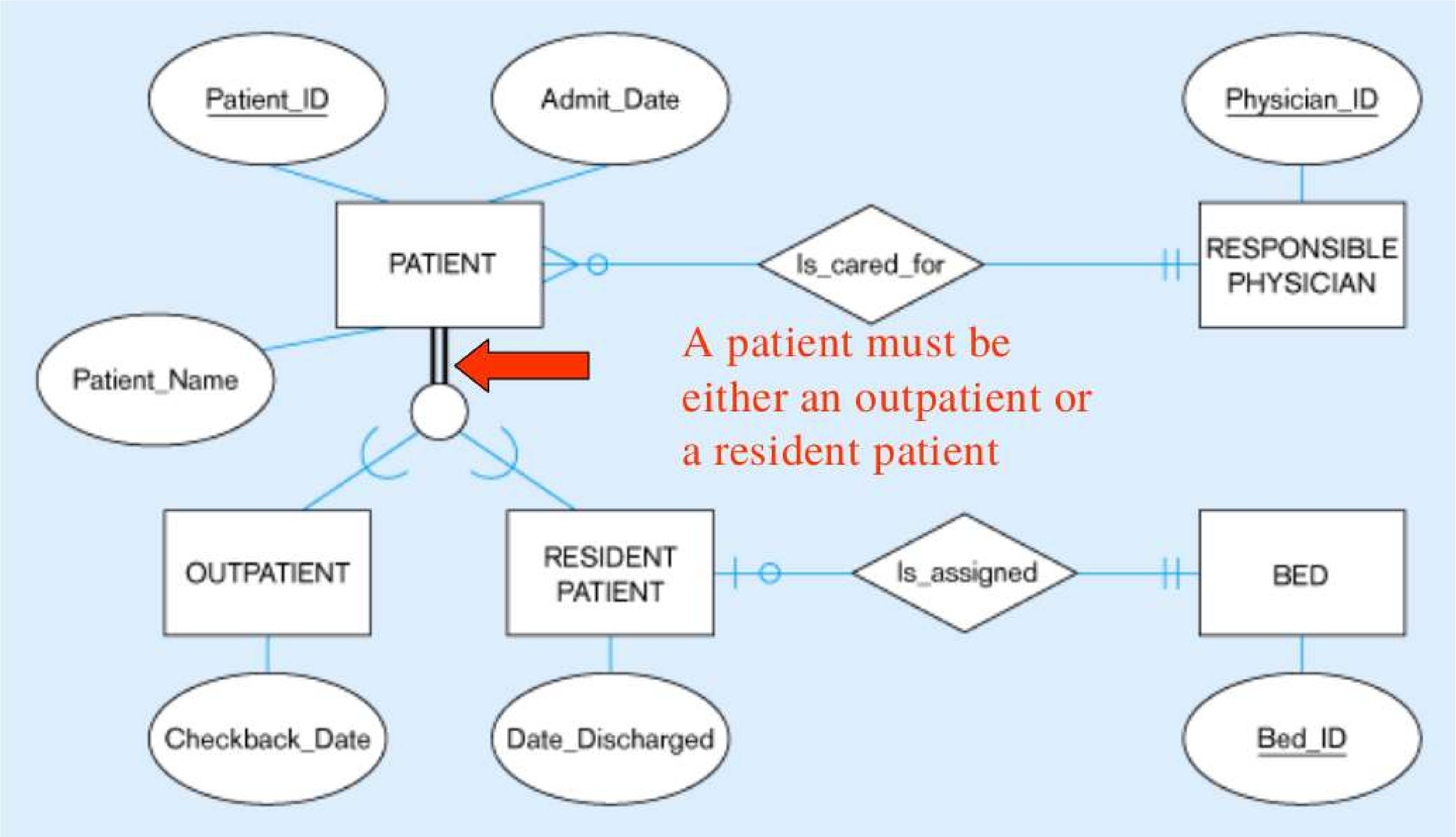
Constraints on Specialization/Generalization

Completeness:

- It might be total or partial constraint.
- A total specialization constraint specifies that every entity in the superclass must be a member of atleast one subclass in the specialization.
- Every employee must be either an hourly_employee or salaried employee
- Shown in EER diagram using a double line to connect the superclass to the circle.
- A single line used to display a partial specialization which allows an entity not to belong to any of the subclasses.

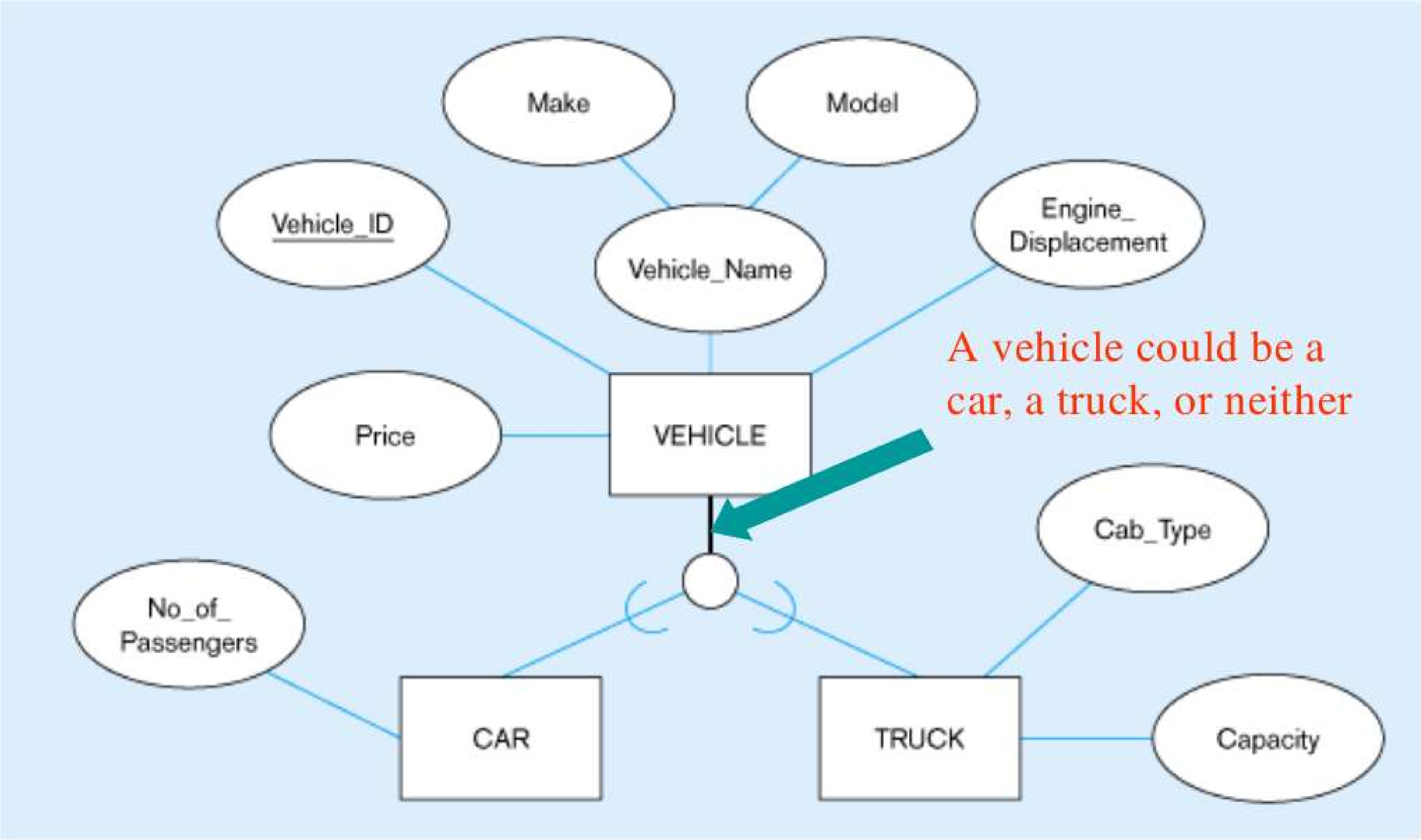
–

Completeness Constraint – Total



A patient must be either an outpatient or a resident patient

Completeness Constraint – Partial



A vehicle could be a car, a truck, or neither

Disjointness and Completeness

The disjointness and completeness constraints are *independent*.

we have the following four possible constraints on a specialization:

- Disjoint, total
- Disjoint, partial
- Overlapping, total
- Overlapping, partial

The correct constraint is determined from the real-world meaning that applies to each specialization.

Rules for Insertion and Deletion

Certain insertion and deletion rules apply to specialization (and generalization) as a consequence of the constraints specified earlier.

- Deleting an entity from a superclass implies that it is automatically deleted from all the subclasses to which it belongs.
- Inserting an entity in a superclass implies that the entity is mandatorily inserted in all *predicate-defined* (or *attribute-defined*) subclasses for which the entity satisfies the defining predicate.
- Inserting an entity in a superclass of a *total specialization* implies that the entity is mandatorily inserted in at least one of the subclasses of the specialization.

Summary

- ERR Diagram
- Superclass/Subclass
- Specialization /Generalization
- Disjoint/Overlap
- Total/ Partial