

Database Systems

Instructor
Bilal Khalid Dar



The Enhanced E-R Model

Objectives

- Define terms
- Understand use of supertype/subtype relationships
- Understand use of specialization and generalization techniques
- Specify completeness and disjointness constraints
- Develop supertype/subtype hierarchies for realistic business situations

Supertypes and Subtypes

- Enhanced ER model: extends original ER model with new modeling constructs
- **Subtype:** A subgrouping of the entities in an entity type that has attributes distinct from those in other subgroupings
- **Supertype:** A generic entity type that has a relationship with one or more subtypes
- **Attribute Inheritance:**
 - Subtype entities inherit values of all attributes of the supertype
 - An instance of a subtype is also an instance of the supertype

Figure 3-1 Basic notation for supertype/subtype notation

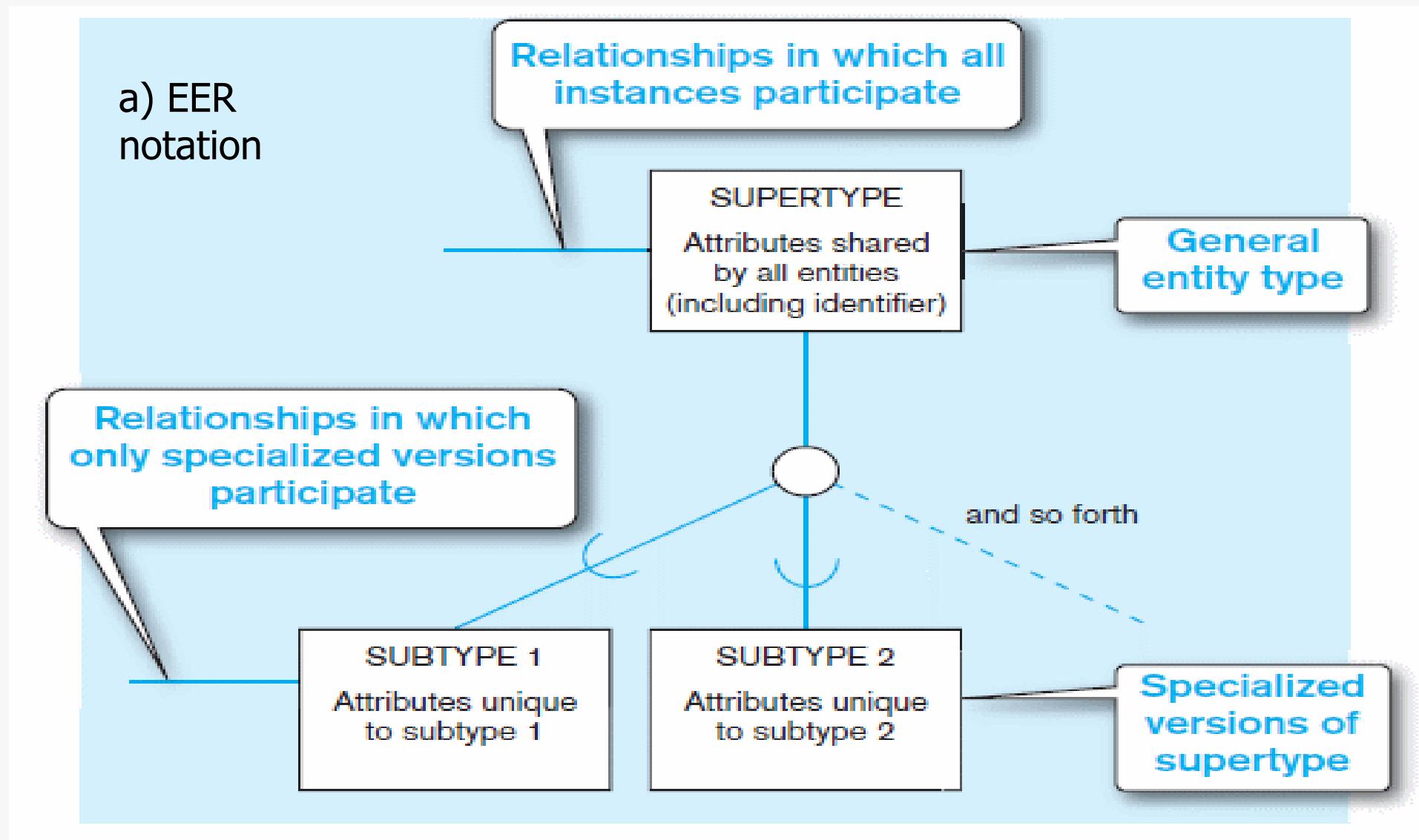
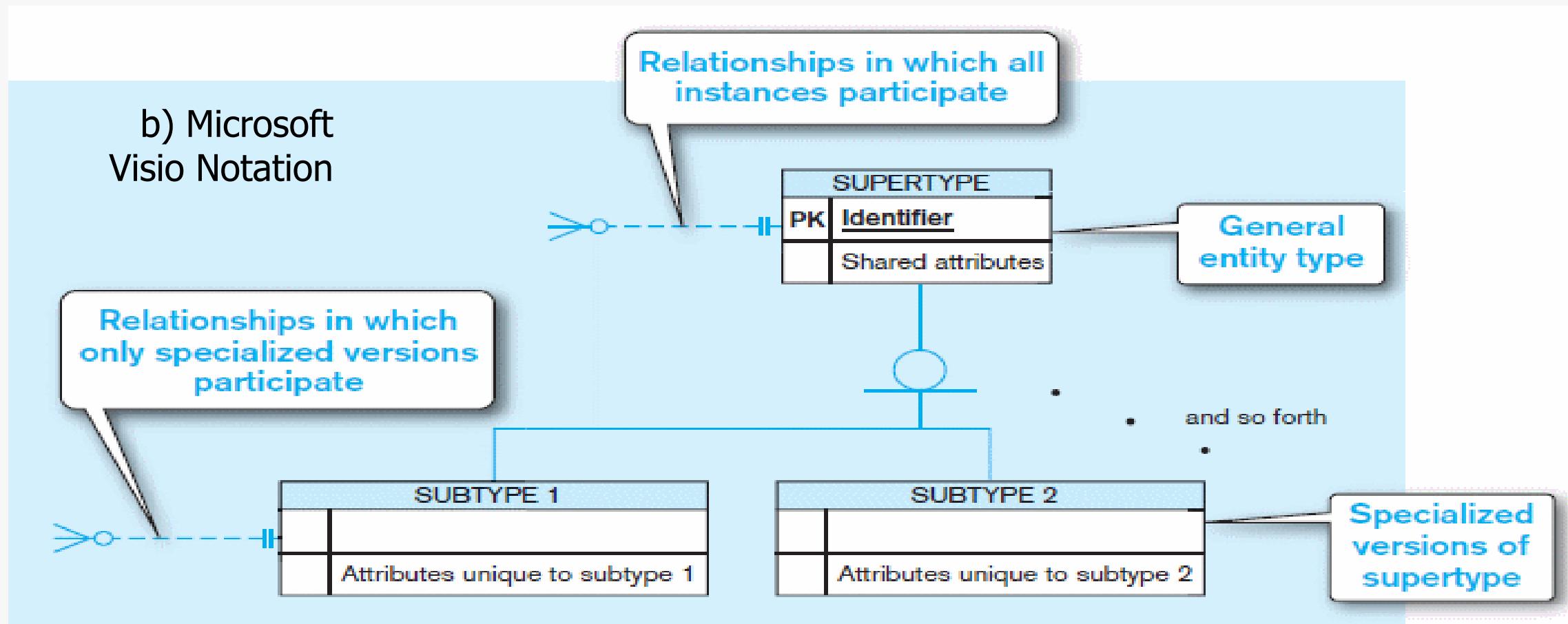
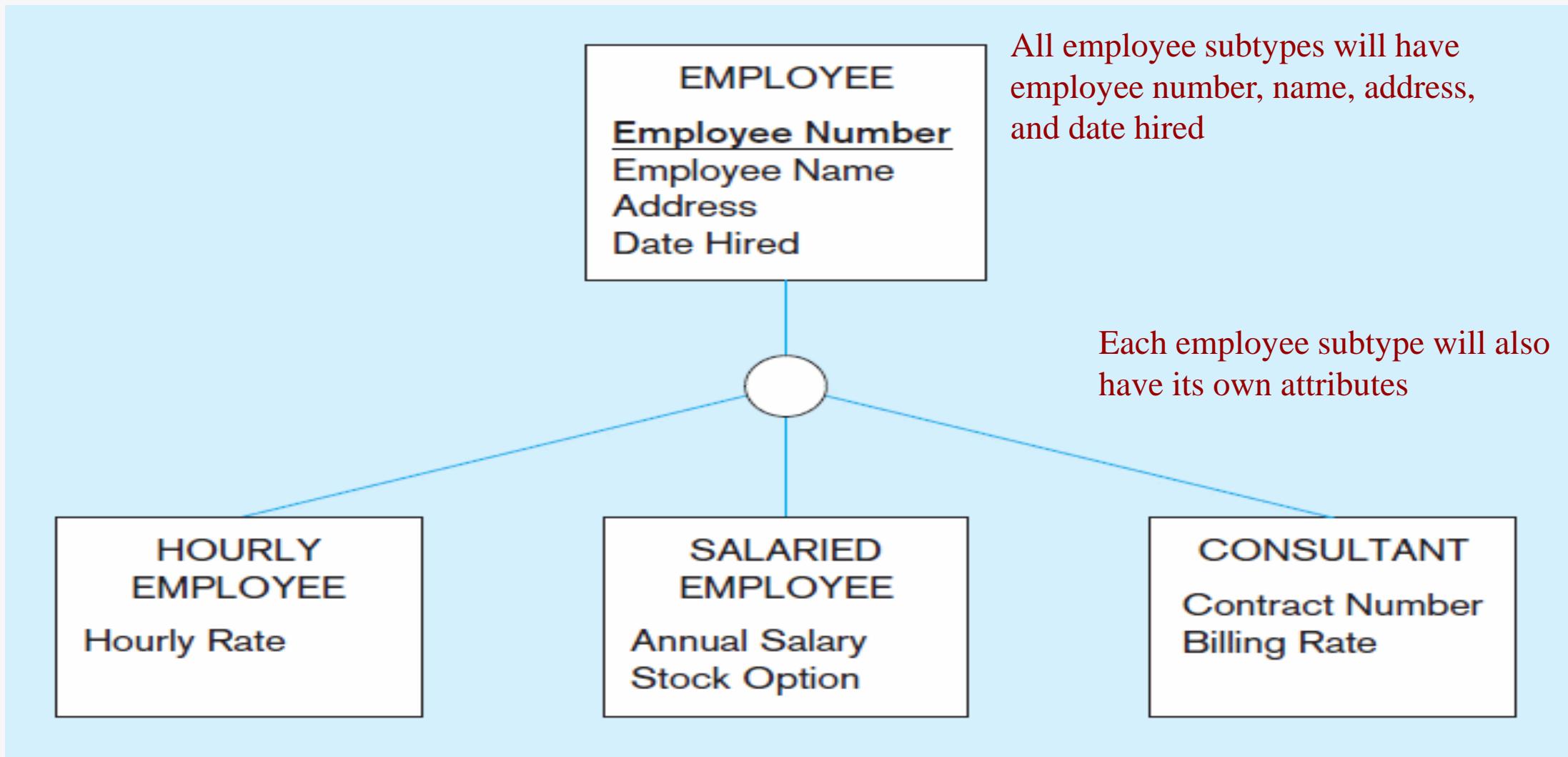


Figure 3-1 Basic notation for supertype/subtype notation (cont.)



Different modeling tools may have different notation for the same modeling constructs

Figure 3-2 Employee supertype with three subtypes



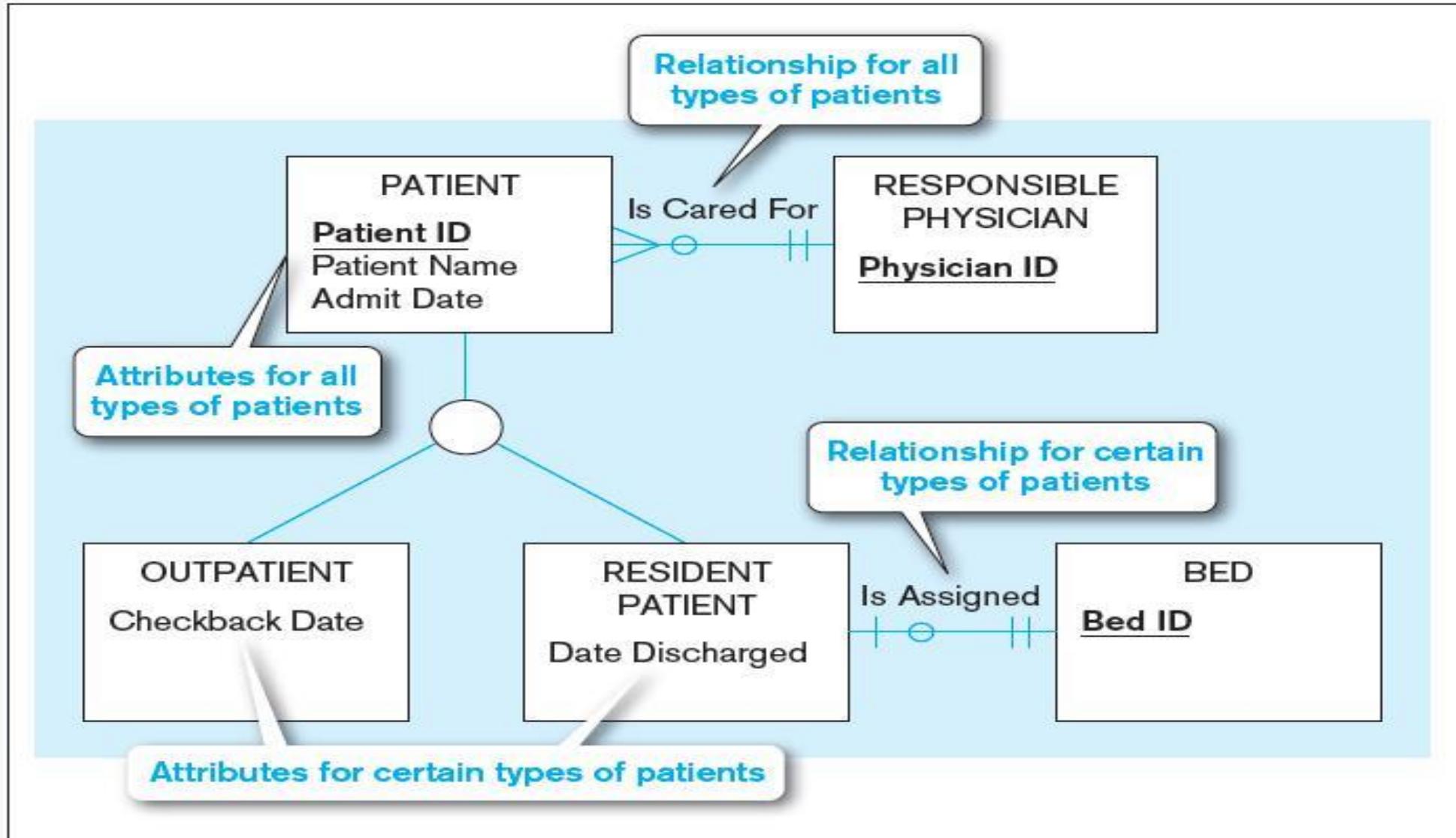
Subclasses and Superclasses (3)

- 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

Relationships and Subtypes

- Relationships at the *supertype* level indicate that all subtypes will participate in the relationship
- The instances of a *subtype* may participate in a relationship unique to that subtype. In this situation, the relationship is shown at the subtype level

Figure 3-3 Supertype/subtype relationships in a hospital



Generalization and Specialization

- ***Generalization:*** The process of defining a more general entity type from a set of more specialized entity types. BOTTOM-UP
- ***Specialization:*** The process of defining one or more subtypes of the supertype and forming supertype/subtype relationships. TOP-DOWN

Figure 3-4 Example of generalization

a) Three entity types: CAR, TRUCK, and MOTORCYCLE

CAR

Vehicle ID
Price
Engine Displacement
Vehicle Name
(Make, Model)
No Of Passengers

TRUCK

Vehicle ID
Price
Engine Displacement
Vehicle Name
(Make, Model)
Capacity
Cab Type

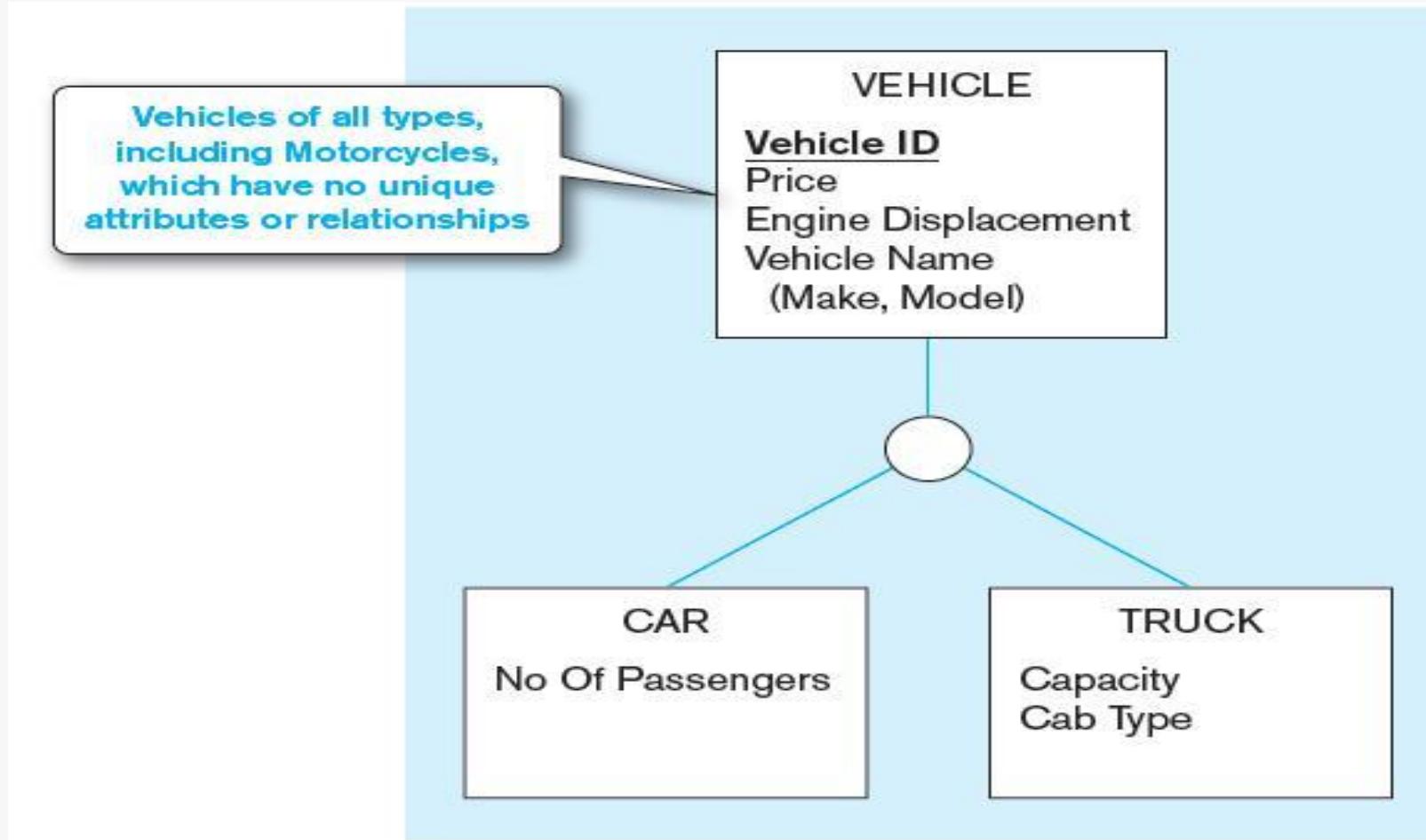
MOTORCYCLE

Vehicle ID
Price
Engine Displacement
Vehicle Name
(Make, Model)

All these types of vehicles have common attributes

Figure 3-4 Example of generalization (cont.)

b) Generalization to VEHICLE supertype



So we put the shared attributes in a supertype

Note: no subtype for motorcycle, since it has no unique attributes

Figure 3-5 Example of specialization

a) Entity type PART

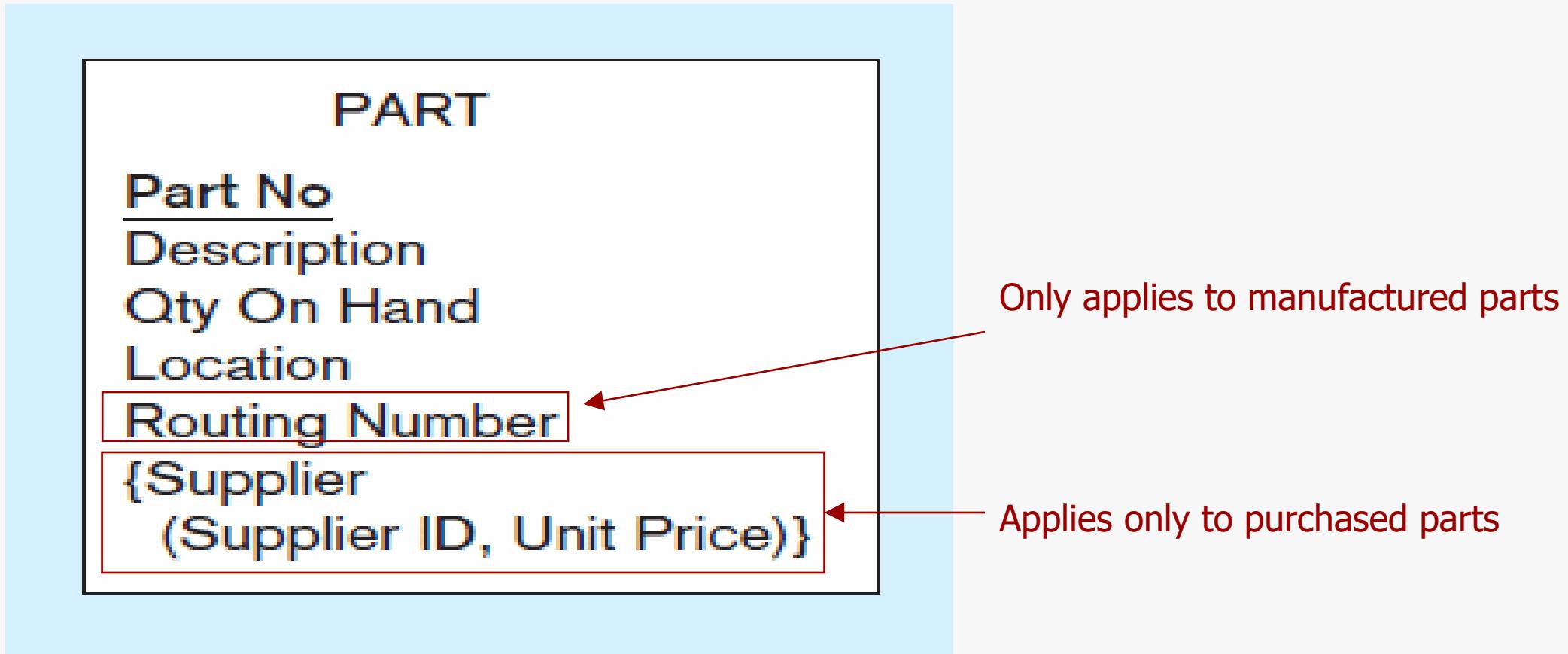
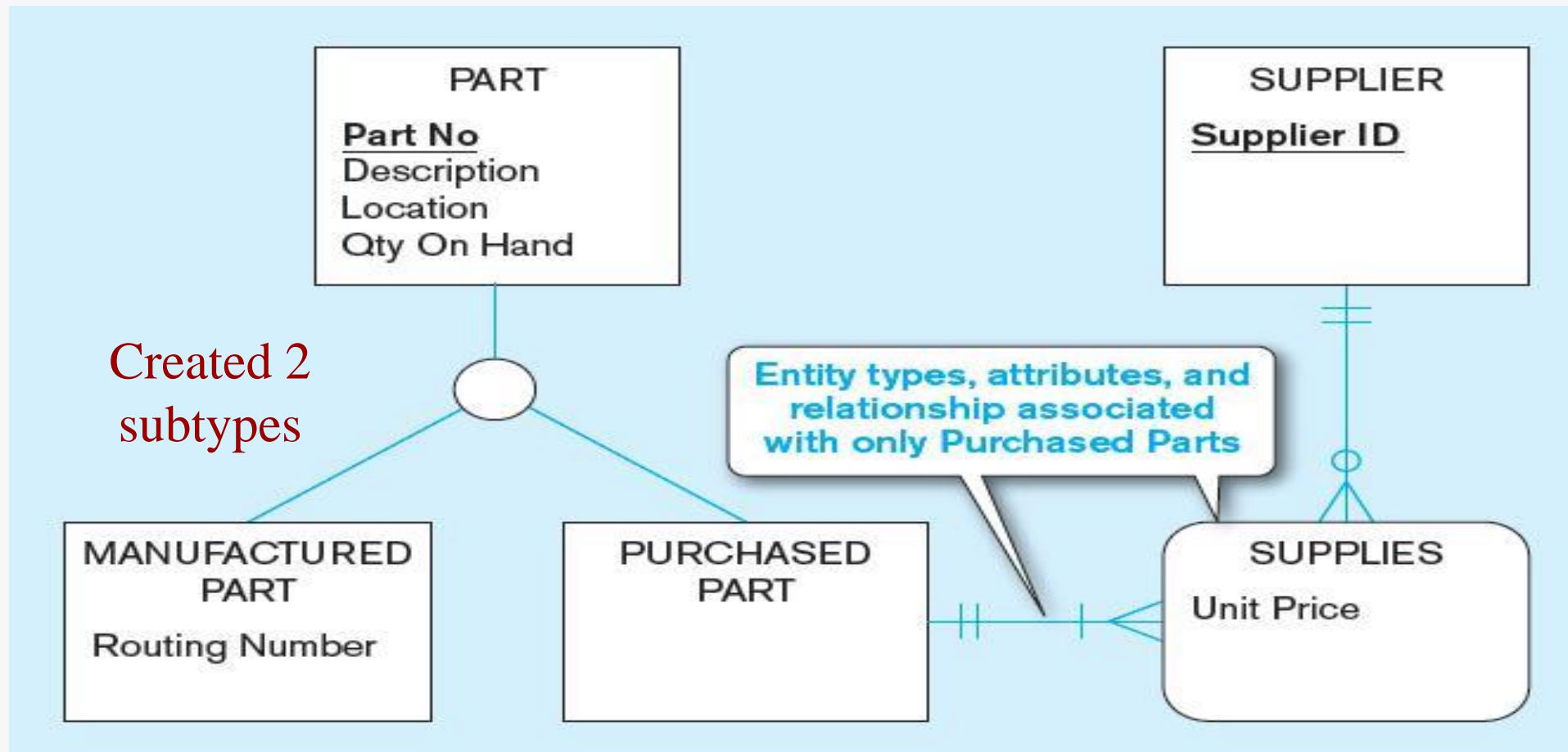


Figure 3-5 Example of specialization (cont.)

b) Specialization to MANUFACTURED PART and PURCHASED PART



Note: multivalued composite attribute was replaced by an associative entity relationship to another entity

Constraints on Specialization and Generalization (3)

- Two basic constraints can apply to a specialization/generalization:
 - Disjointness Constraint:
 - Completeness Constraint:

Constraints in Supertype/SUBTYPE RELATIONSHIPS

- ***Completeness Constraints:*** Whether an instance of a supertype ***must*** also be a member of at least one subtype
 - Total Specialization Rule: Yes (double line)
 - Partial Specialization Rule: No (single line)

Figure 3-6 Examples of completeness constraints

a) Total specialization rule

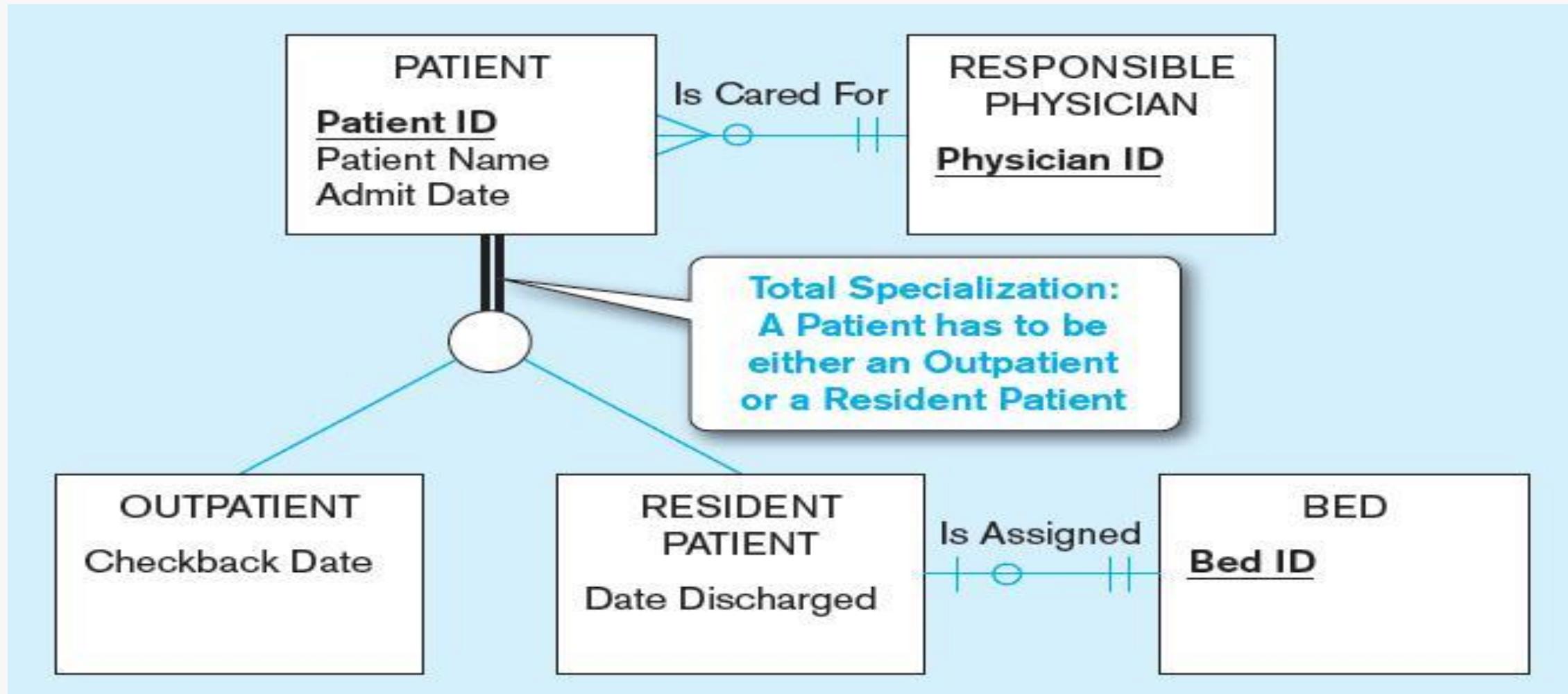
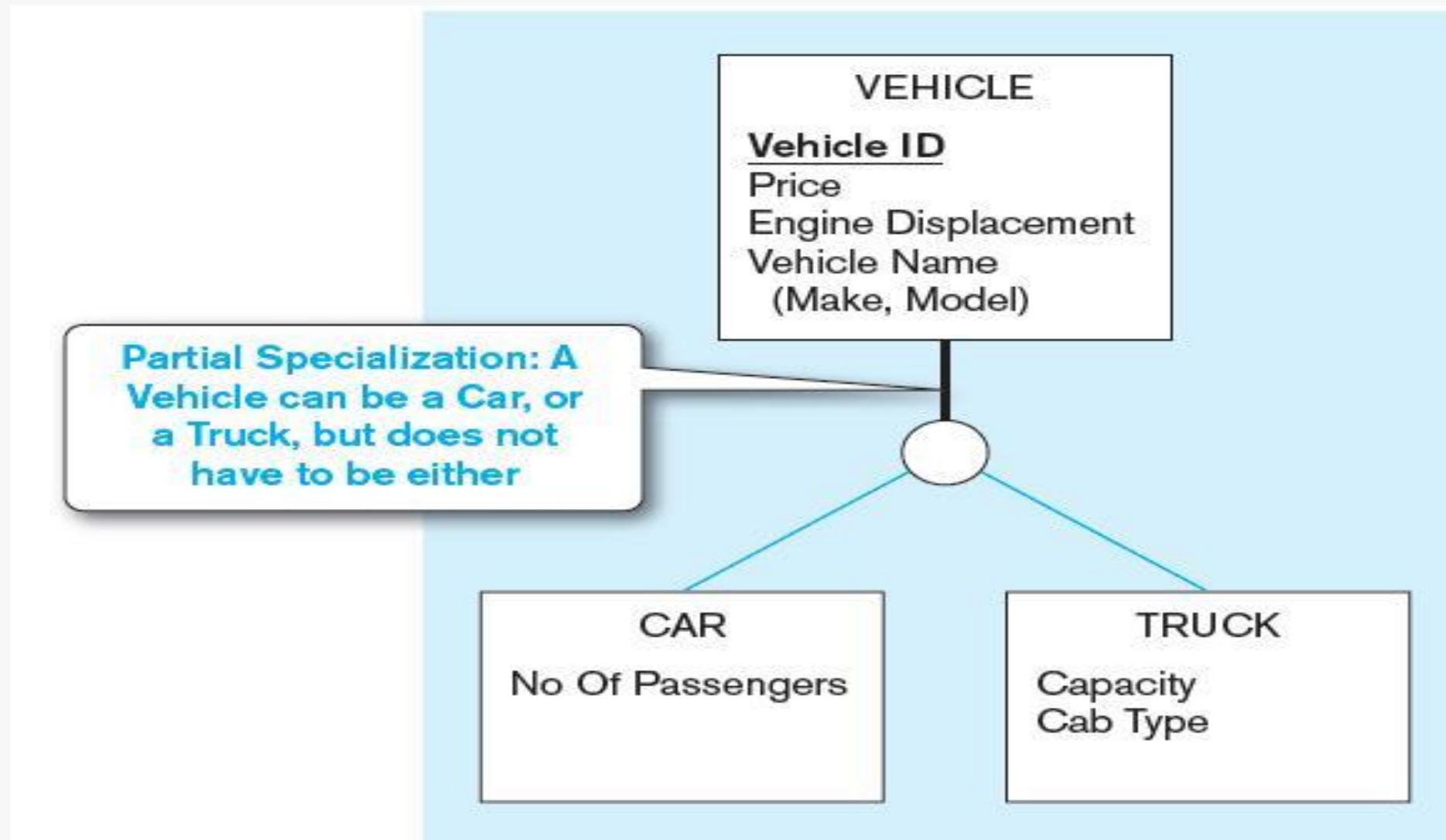


Figure 3-6 Examples of completeness constraints (cont.)

b) Partial specialization rule



Constraints in Supertype/SUBTYPE RELATIONSHIPS

- ***Disjointness Constraints:*** Whether an instance of a supertype may *simultaneously* be a member of two (or more) subtypes
 - Disjoint Rule: An instance of the supertype can be only ONE of the subtypes
 - Overlap Rule: An instance of the supertype could be more than one of the subtypes

Figure 3-7 Examples of disjointness constraints

a) Disjoint rule

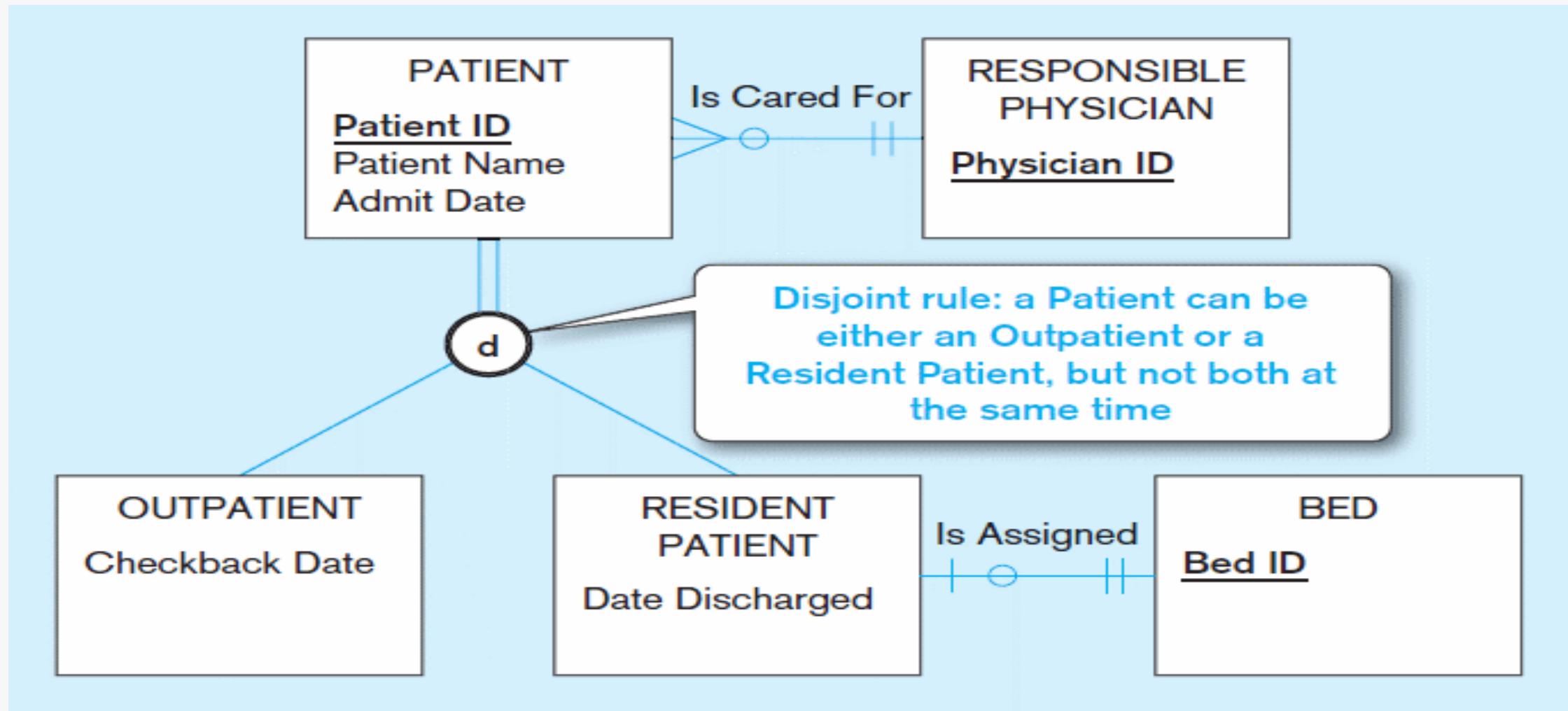
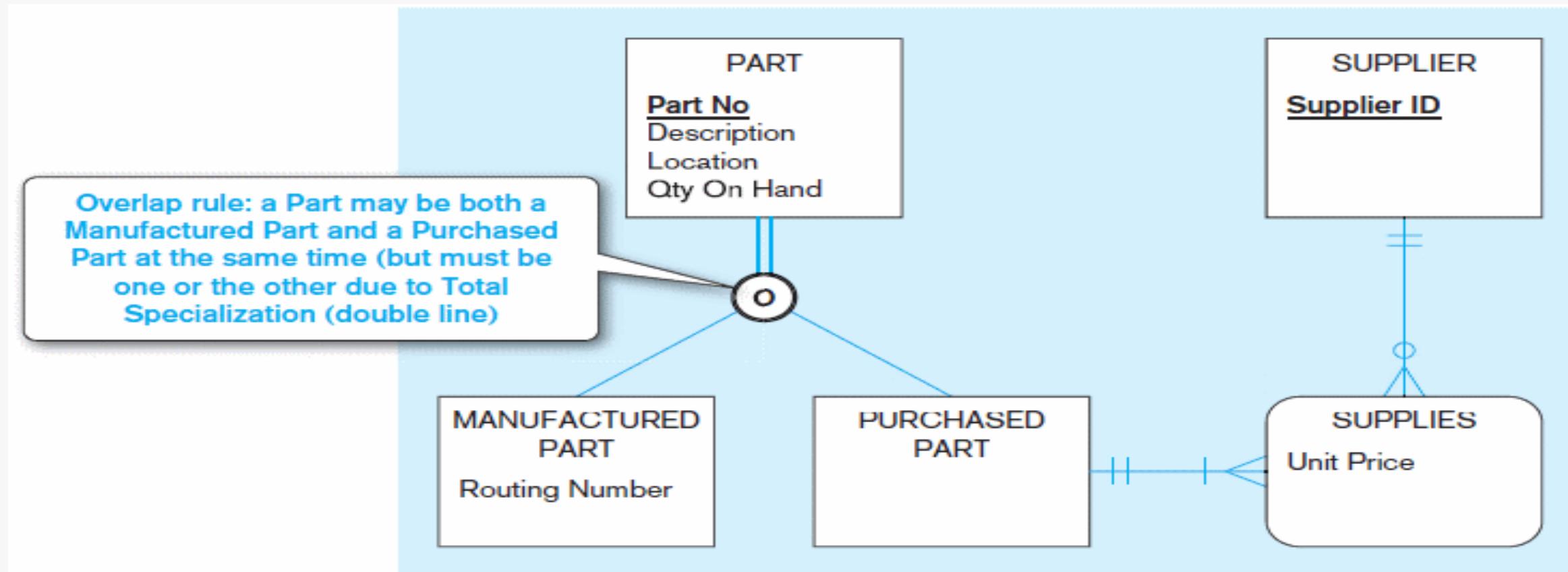


Figure 3-7 Examples of disjointness constraints (cont.)

b) Overlap rule



Constraints in Supertype/SUBTYPE RELATIONSHIPS

- ***Subtype Discriminator***: An attribute of the supertype whose values determine the target subtype(s)
 - **Disjoint** – a *simple* attribute with alternative values to indicate the possible subtypes
 - **Overlapping** – a *composite* attribute whose subparts pertain to different subtypes. Each subpart contains a Boolean value to indicate whether or not the instance belongs to the associated subtype

Figure 3-8 Introducing a subtype discriminator (*disjoint* rule)

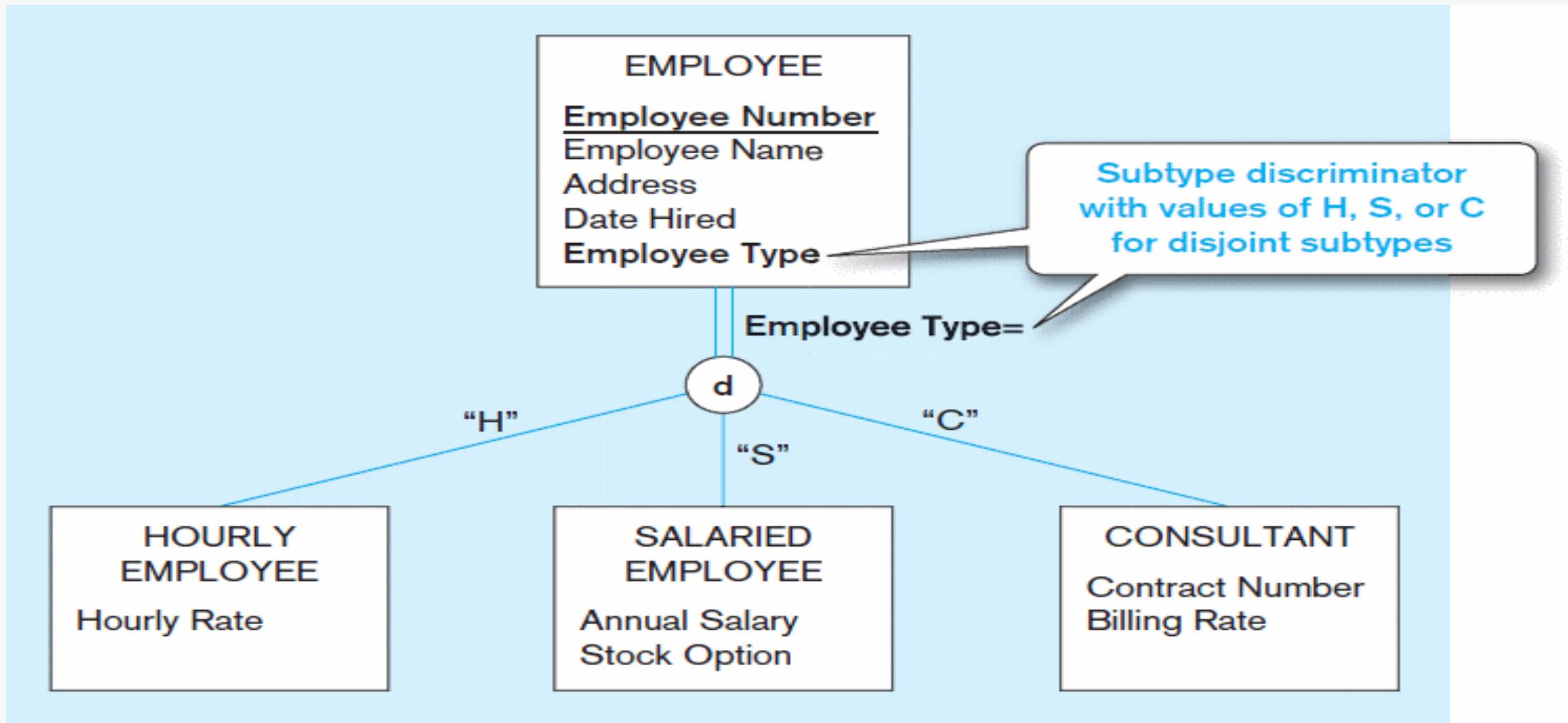


Figure 3-9 Subtype discriminator (**overlap** rule)

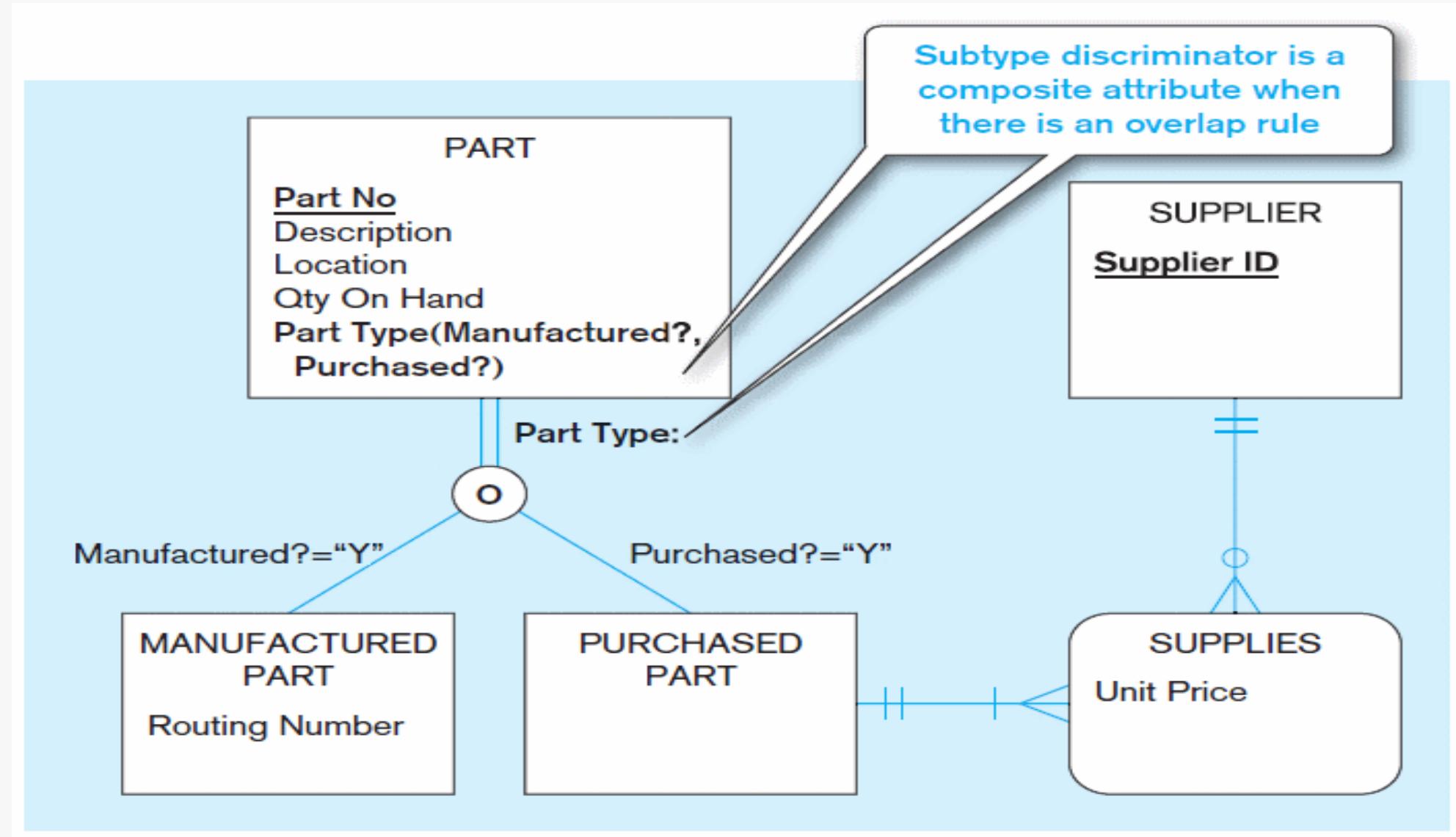
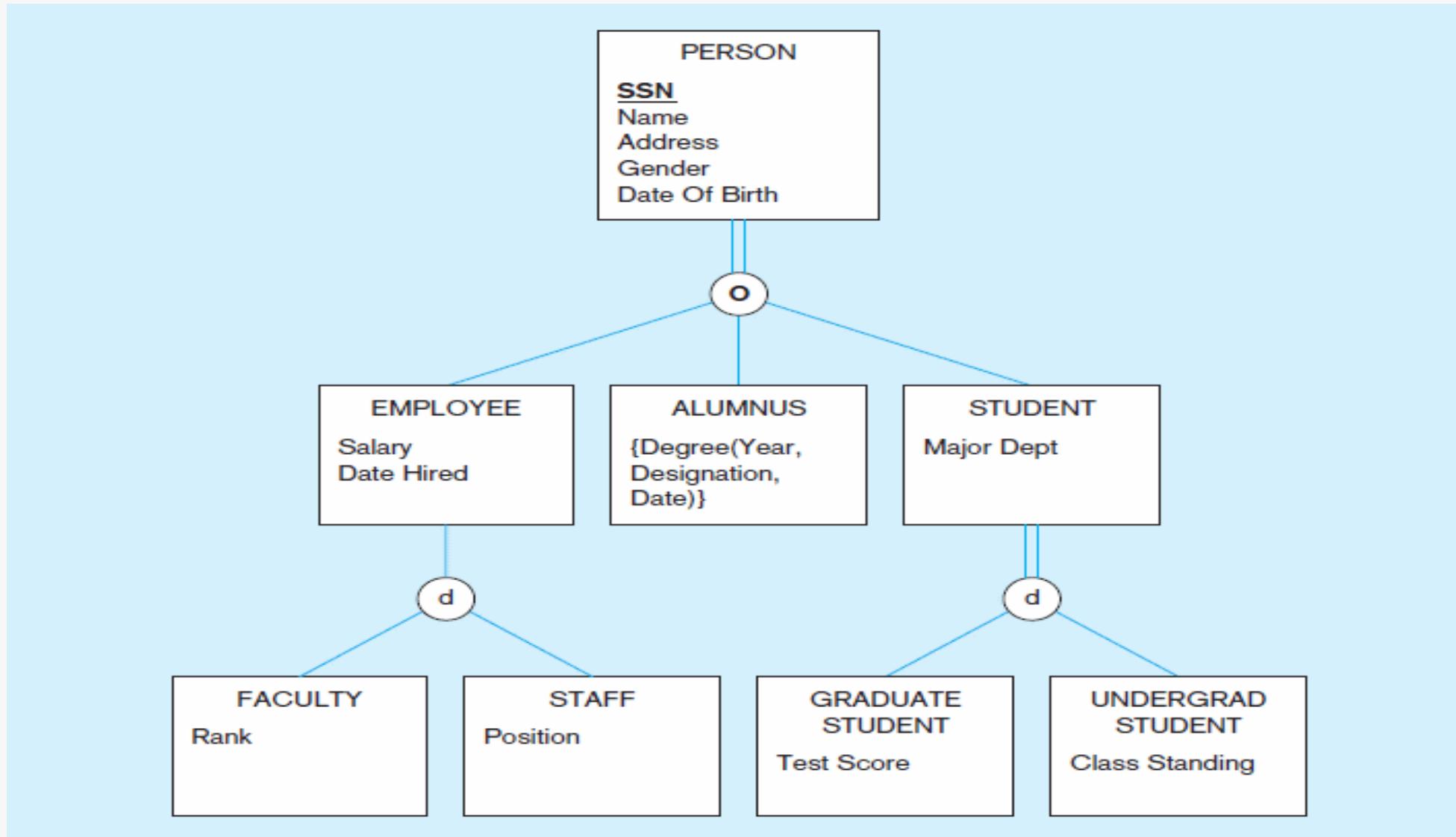


Figure 3-10 Example of supertype/subtype hierarchy



thank
you