(ADVANCED) DATABASE SYSTEMS (DATABASE MANAGEMENTS)

PROF. DR. HASAN HÜSEYİN BALIK
(3RD WEEK)

2. OUTLINE

2. Database Analysis

  2.1 Modeling Data in the Organization

 2.2 The Enhanced E-R Model

2.2 THE ENHANCED E-R MODEL

OBJECTIVES

- Define terms
- Understand use of supertype/subtype relationships
- Use specialization and generalization techniques
- Specify completeness and disjointness constraints
- Develop supertype/subtype hierarchies for realistic business situations
- Develop entity clusters
- Explain universal (packaged) data model
- Describe special features of data modeling project using packaged data model

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

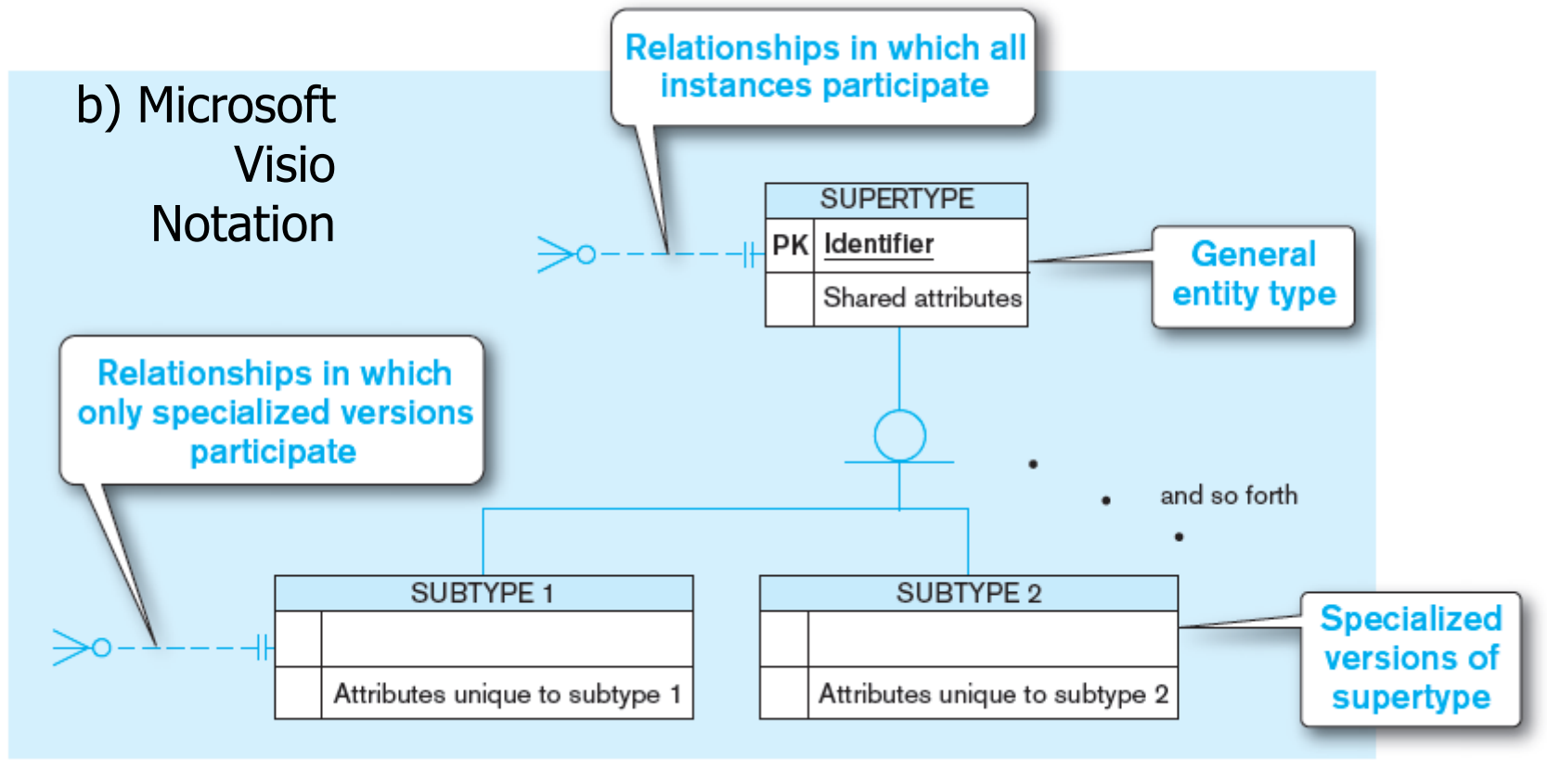 + An instance of a subtype is also an instance of the supertype

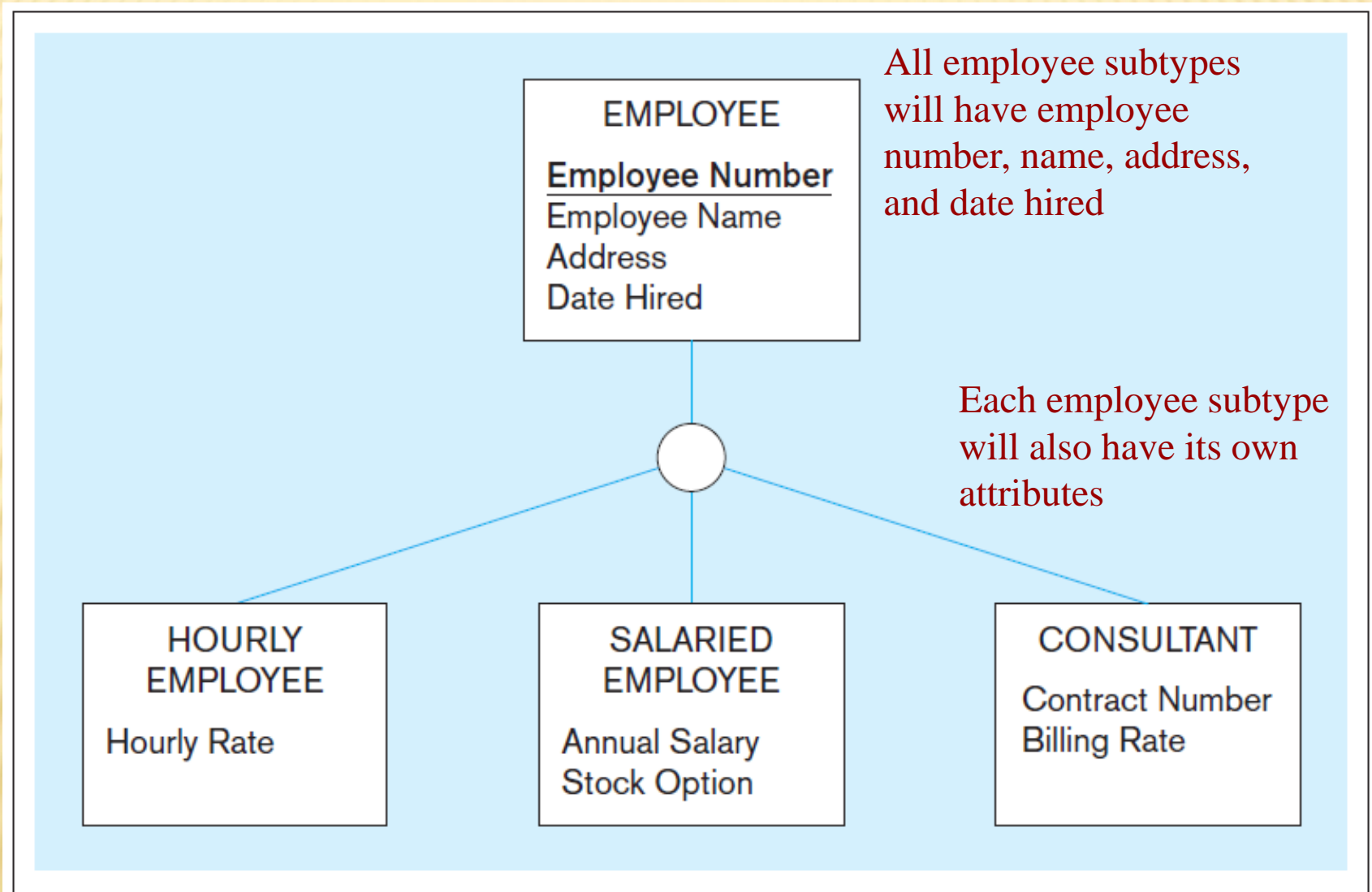# Basic notation for supertype/subtype notation

a) EER notation
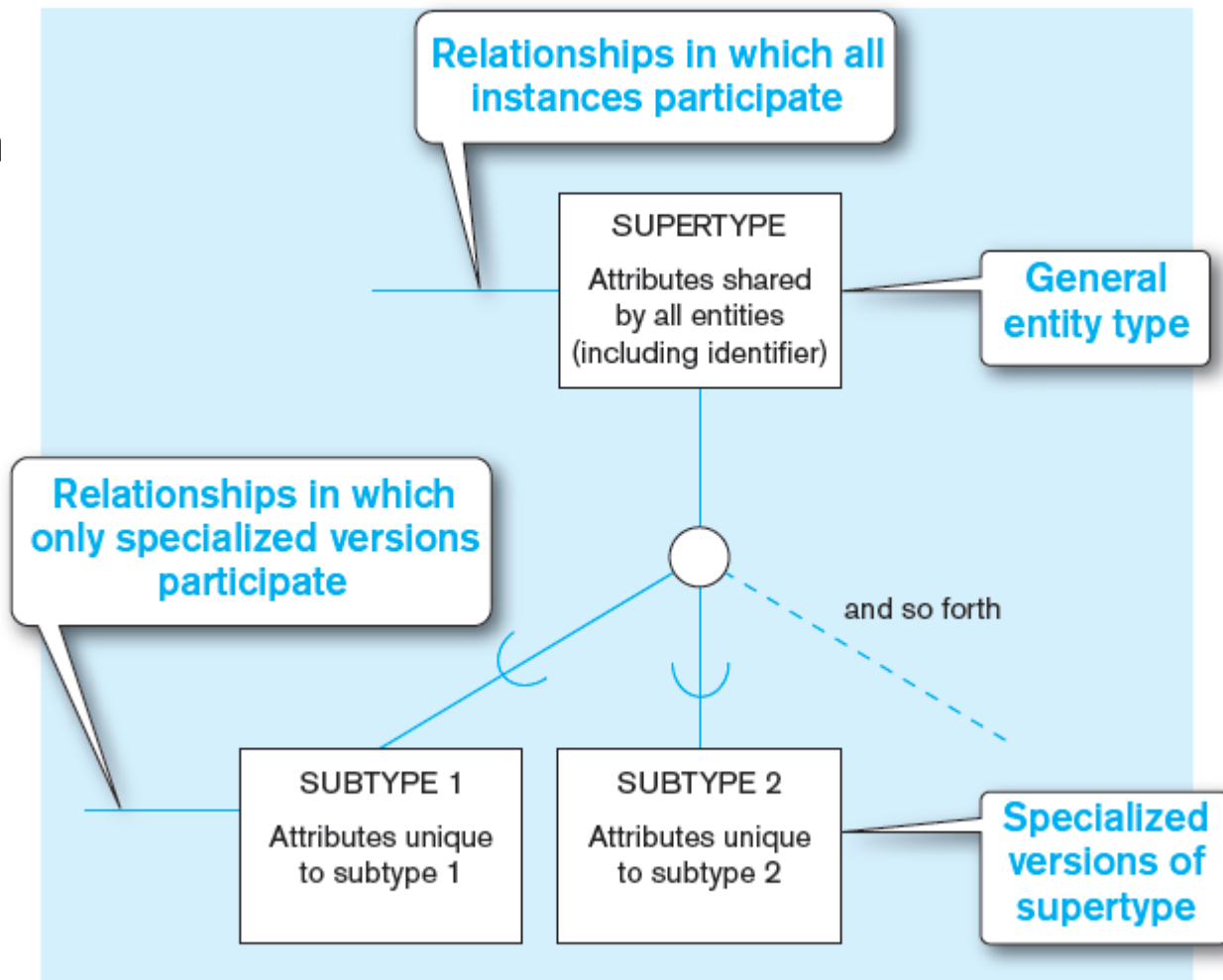
Basic notation for supertype/subtype notation (cont.)

b) Microsoft Visio Notation

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

Employee supertype with three subtypes

EMPLOYEE

Employee Number
Employee Name
Address
Date Hired

All employee subtypes will have employee number, name, address, and date hired

Each employee subtype will also have its own attributes

HOURLY EMPLOYEE

Hourly Rate

SALARIED EMPLOYEE

Annual Salary
Stock Option

CONSULTANT

Contract Number
Billing Rate

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
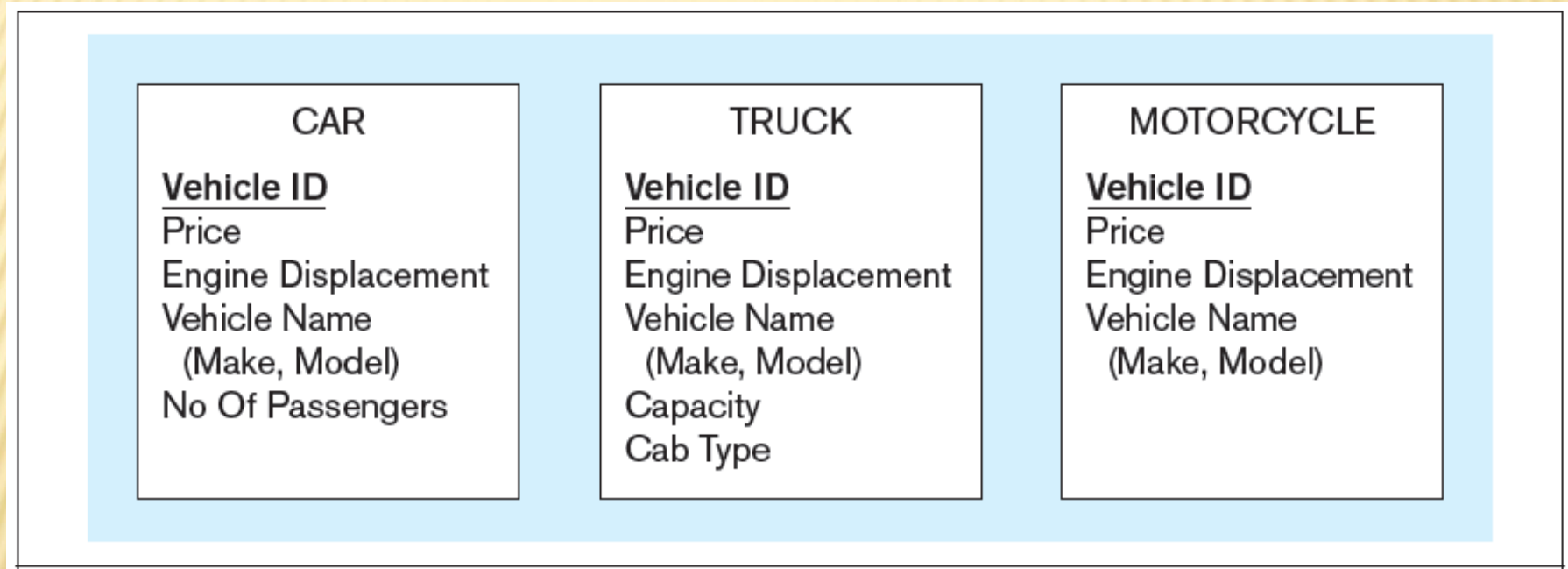
Supertype/subtype relationships in a hospital
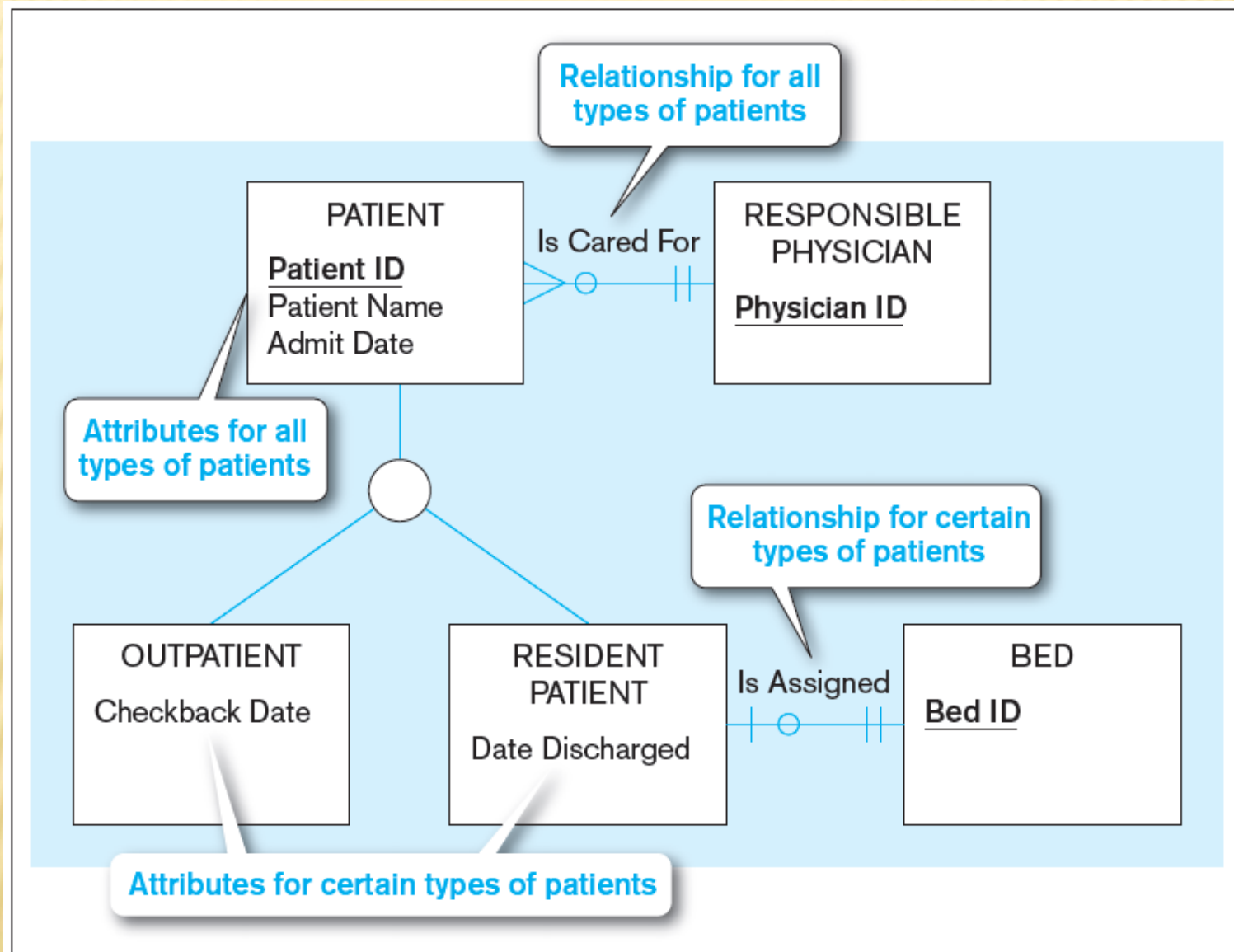
GENERALIZATION AND SPECIALIZATION

- ✖ *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
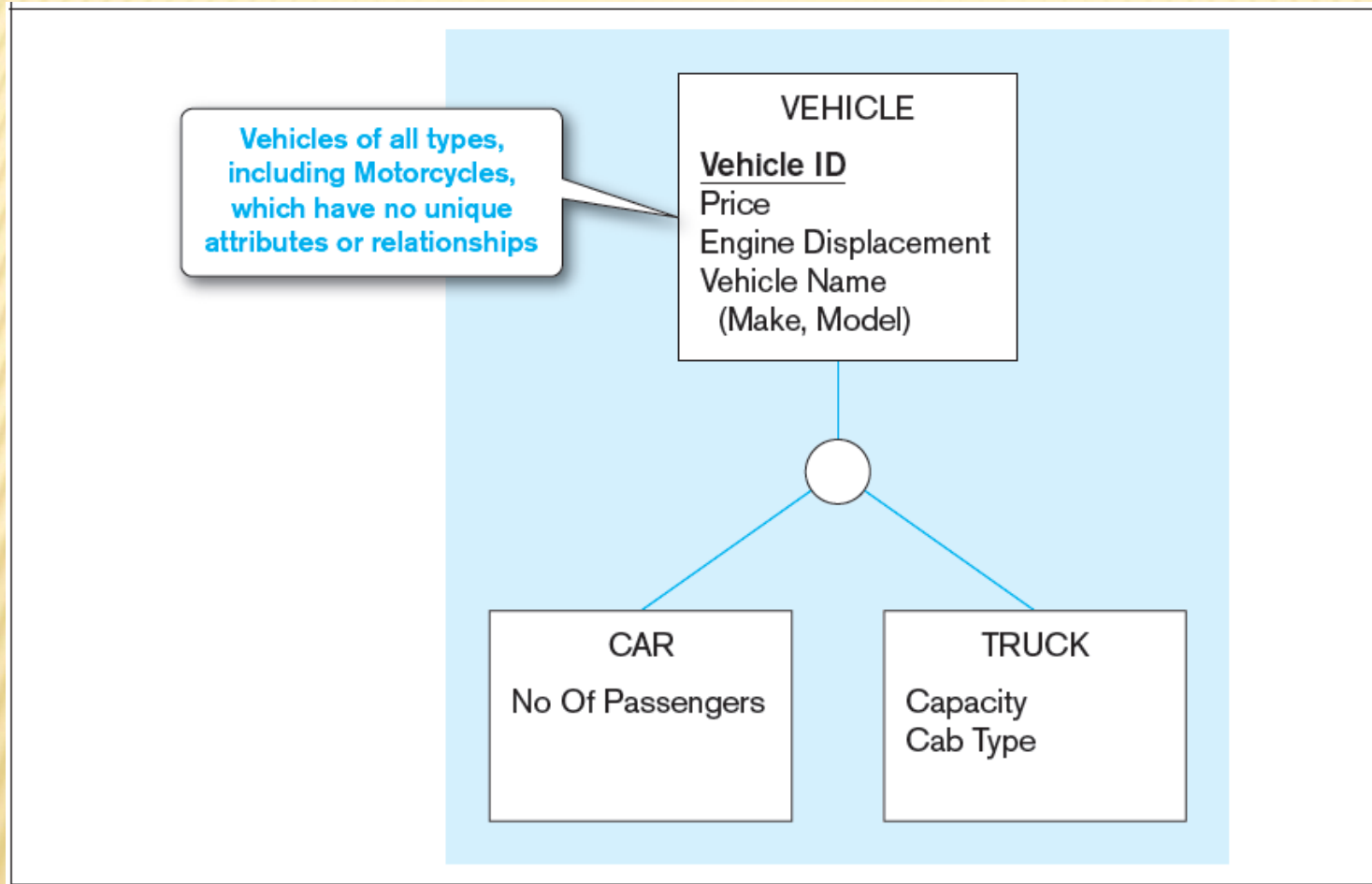
Example of generalization

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

Example of generalization (cont.)

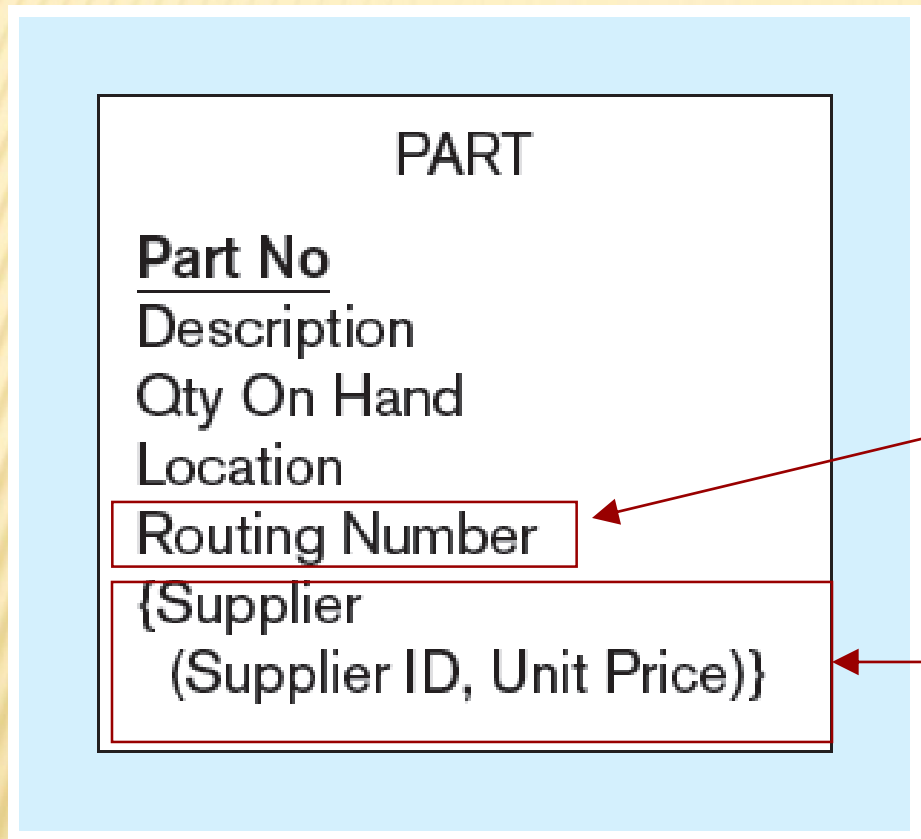## b) Generalization to VEHICLE supertype

So we put the shared attributes in a supertype

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

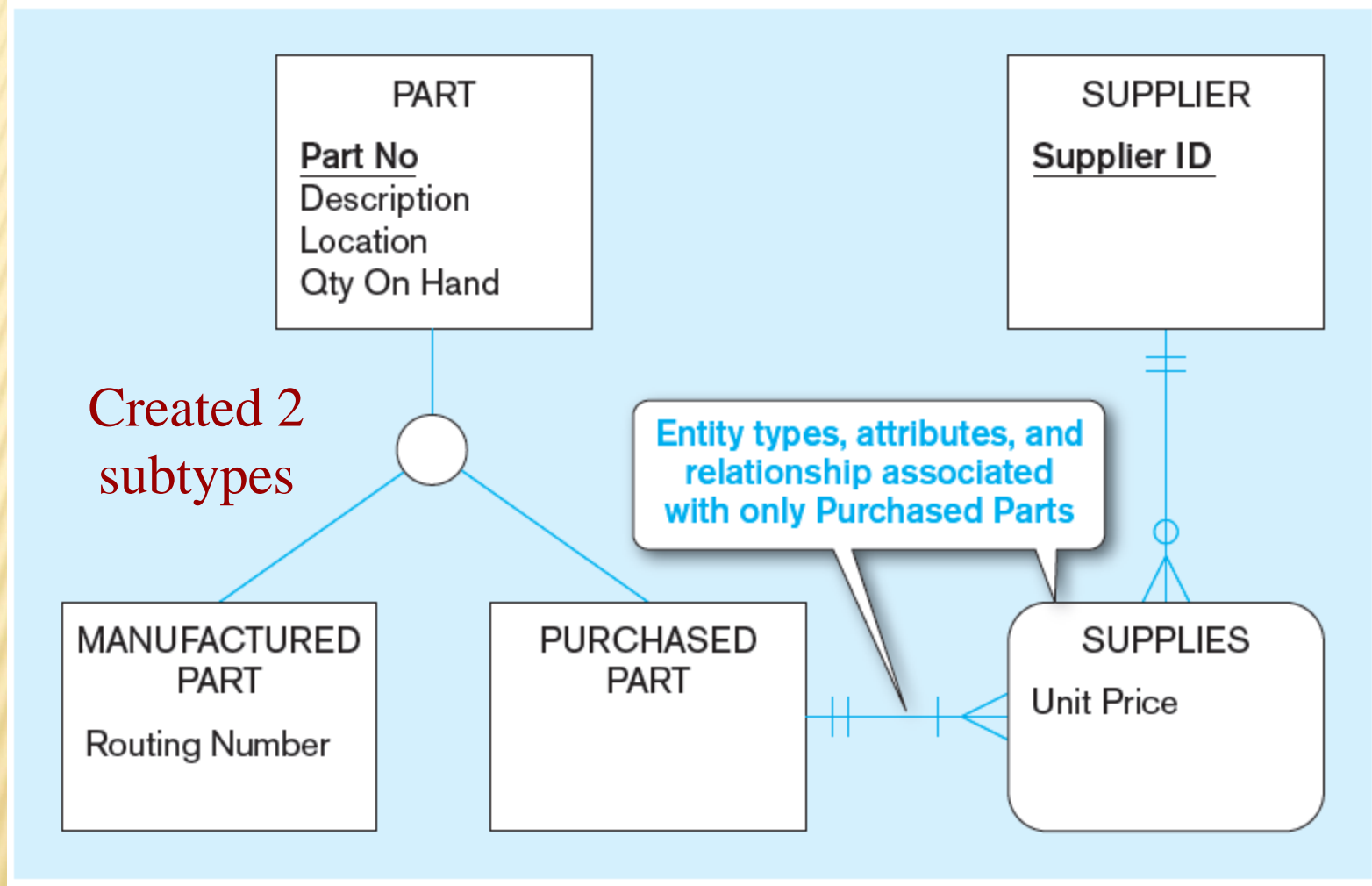# Example of specialization
a) Entity type PART

PART

Part No
Description
Qty On Hand
Location
Routing Number
{Supplier
  (Supplier ID, Unit Price)}

Only applies to manufactured parts

Applies only to purchased parts

Example of specialization (cont.)

b) Specialization to MANUFACTURED PART and PURCHASED PART

Created 2 subtypes

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

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)
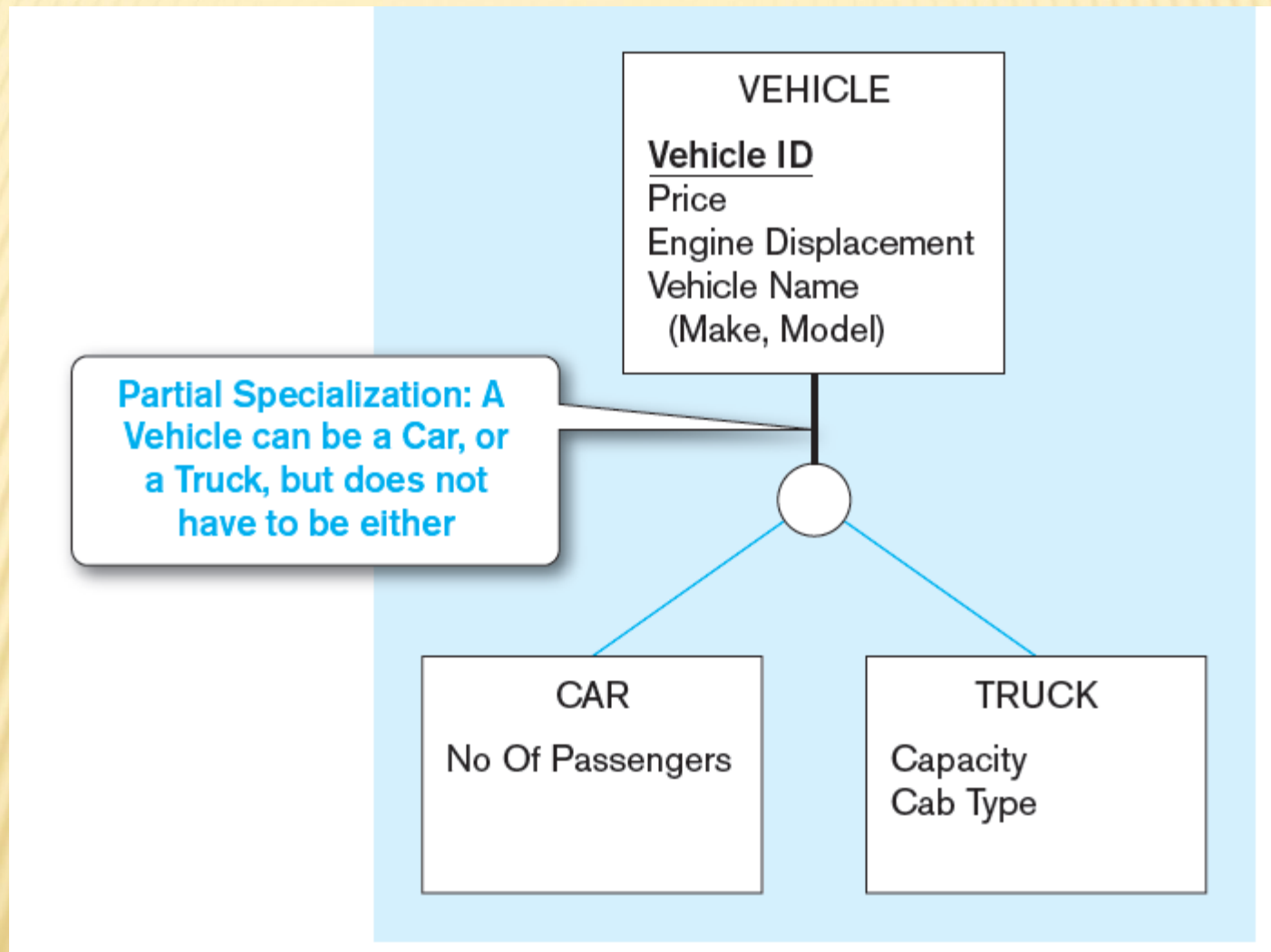
Examples of completeness constraints
a) Total specialization rule
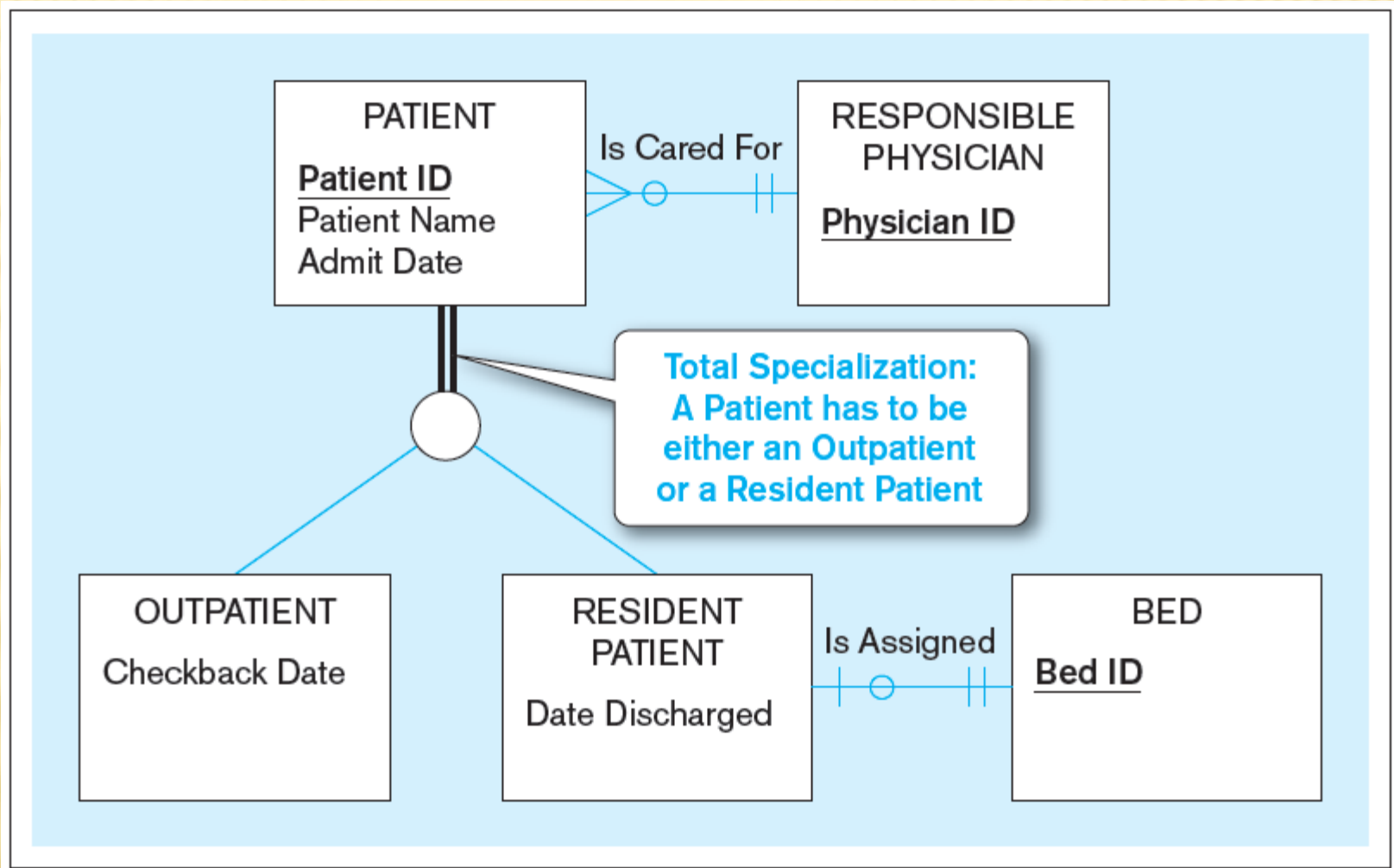
Examples of completeness constraints (cont.)
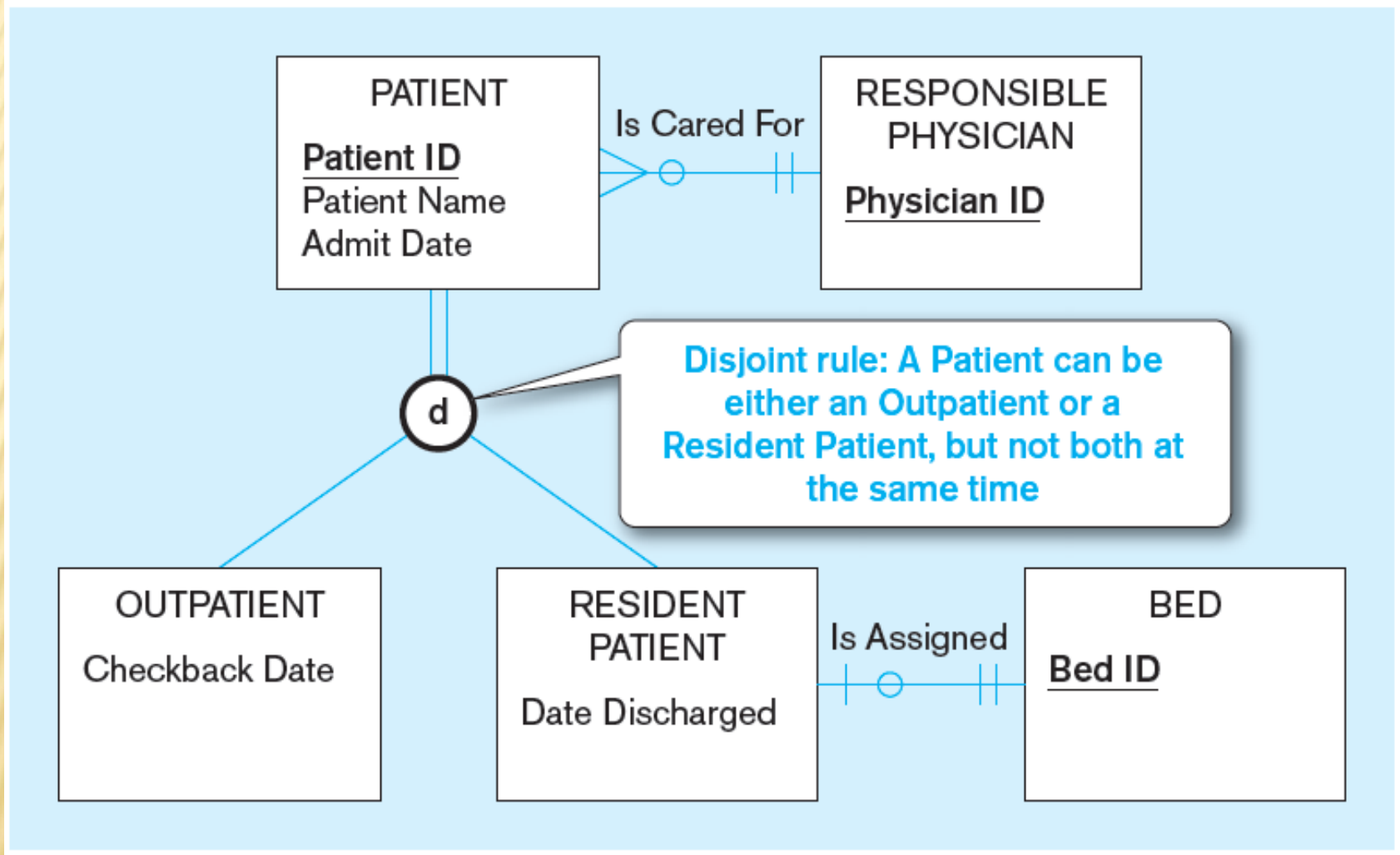## b) Partial specialization rule

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

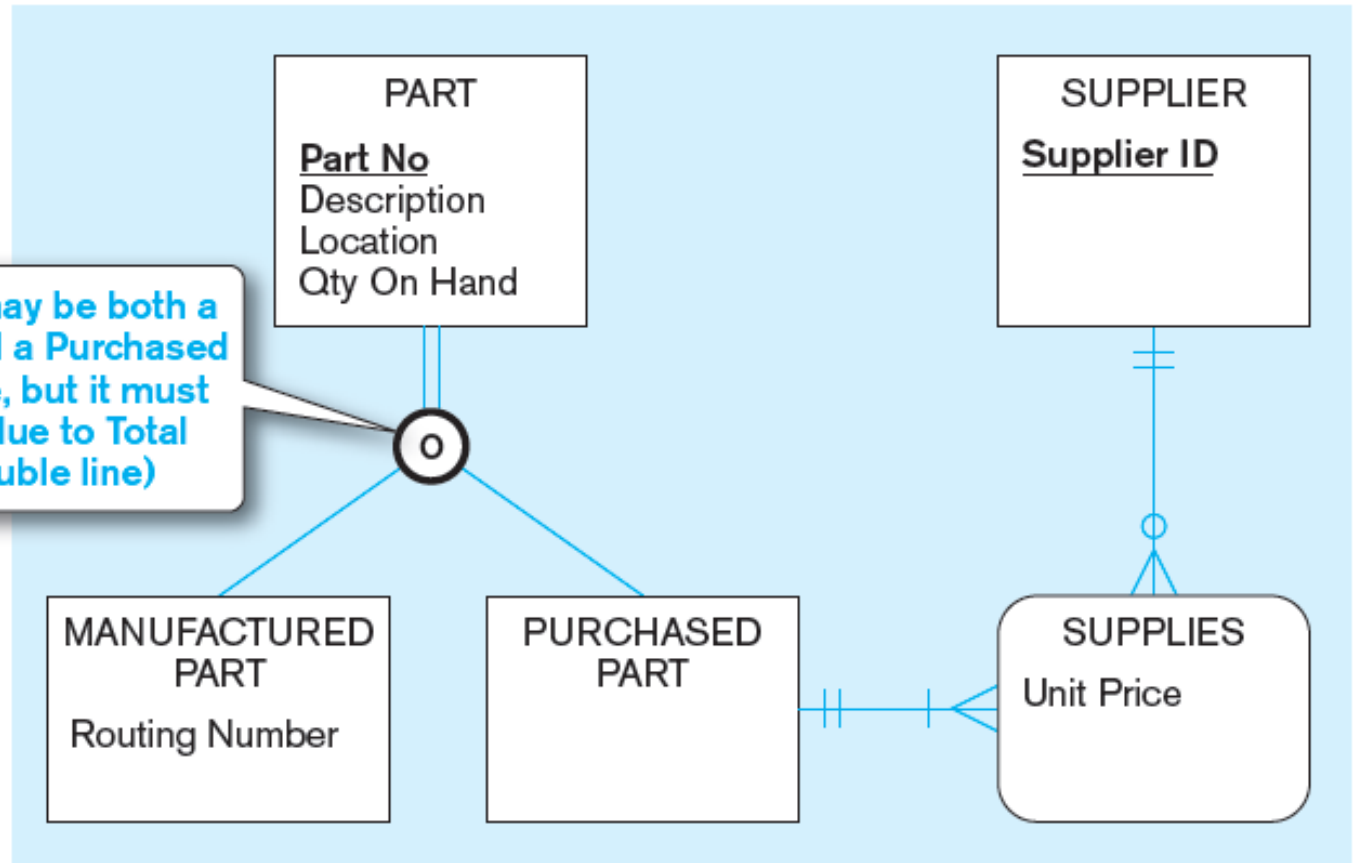# Examples of disjointness constraints
a) Disjoint rule

PATIENT
Patient ID
Patient Name
Admit Date

Is Cared For

RESPONSIBLE PHYSICIAN
Physician ID

d

Disjoint rule: A Patient can be either an Outpatient or a Resident Patient, but not both at the same time

OUTPATIENT
Checkback Date

RESIDENT PATIENT
Date Discharged

Is Assigned

BED
Bed ID

Overlap rule: A Part may be both a Manufactured Part and a Purchased Part at the same time, but it must be one or the other due to Total Specialization (double line)

PART

Part No
Description
Location
Qty On Hand

SUPPLIER

Supplier ID

MANUFACTURED PART

Routing Number

PURCHASED PART

SUPPLIES

Unit Price

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
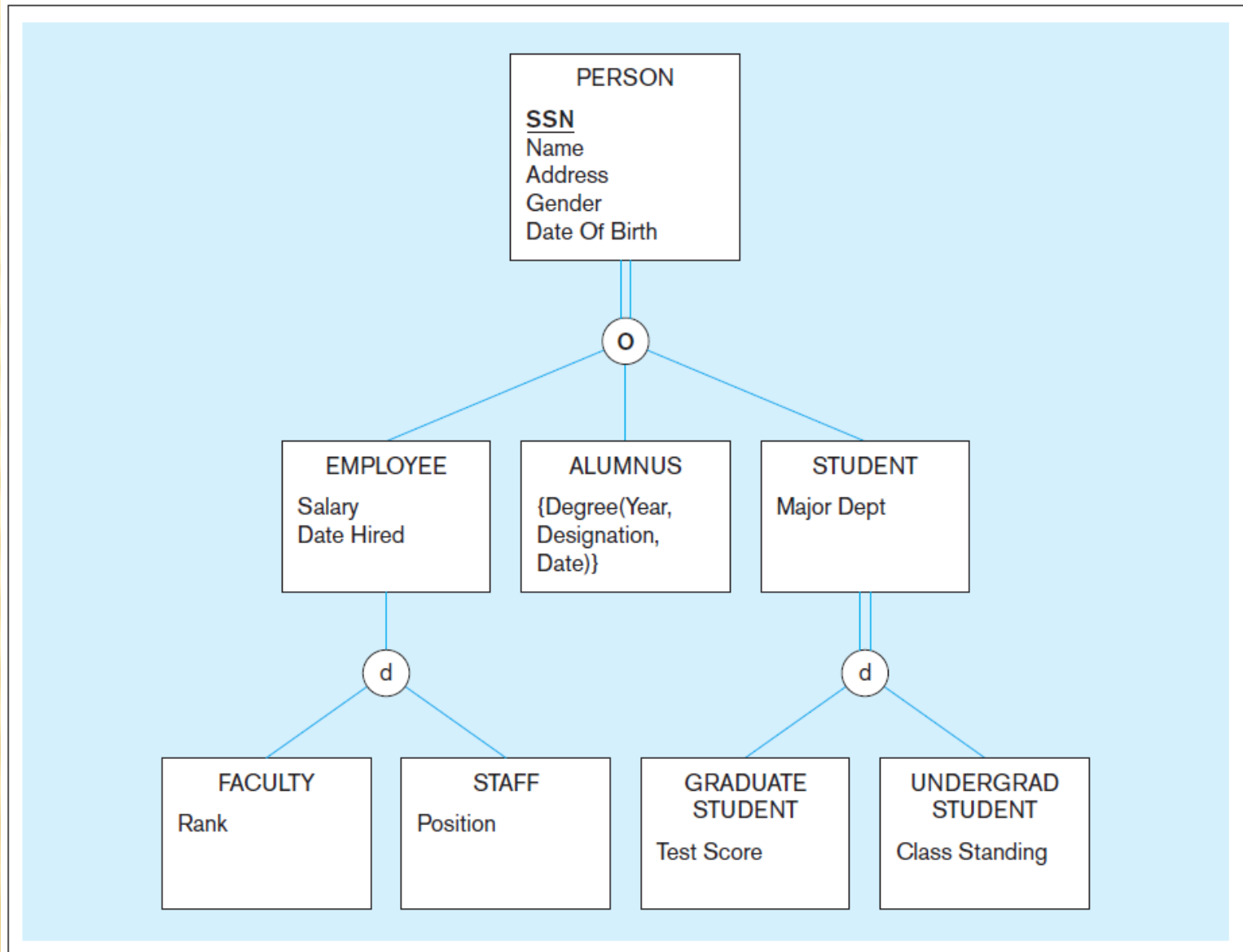
Introducing a subtype discriminator (*disjoint* rule)
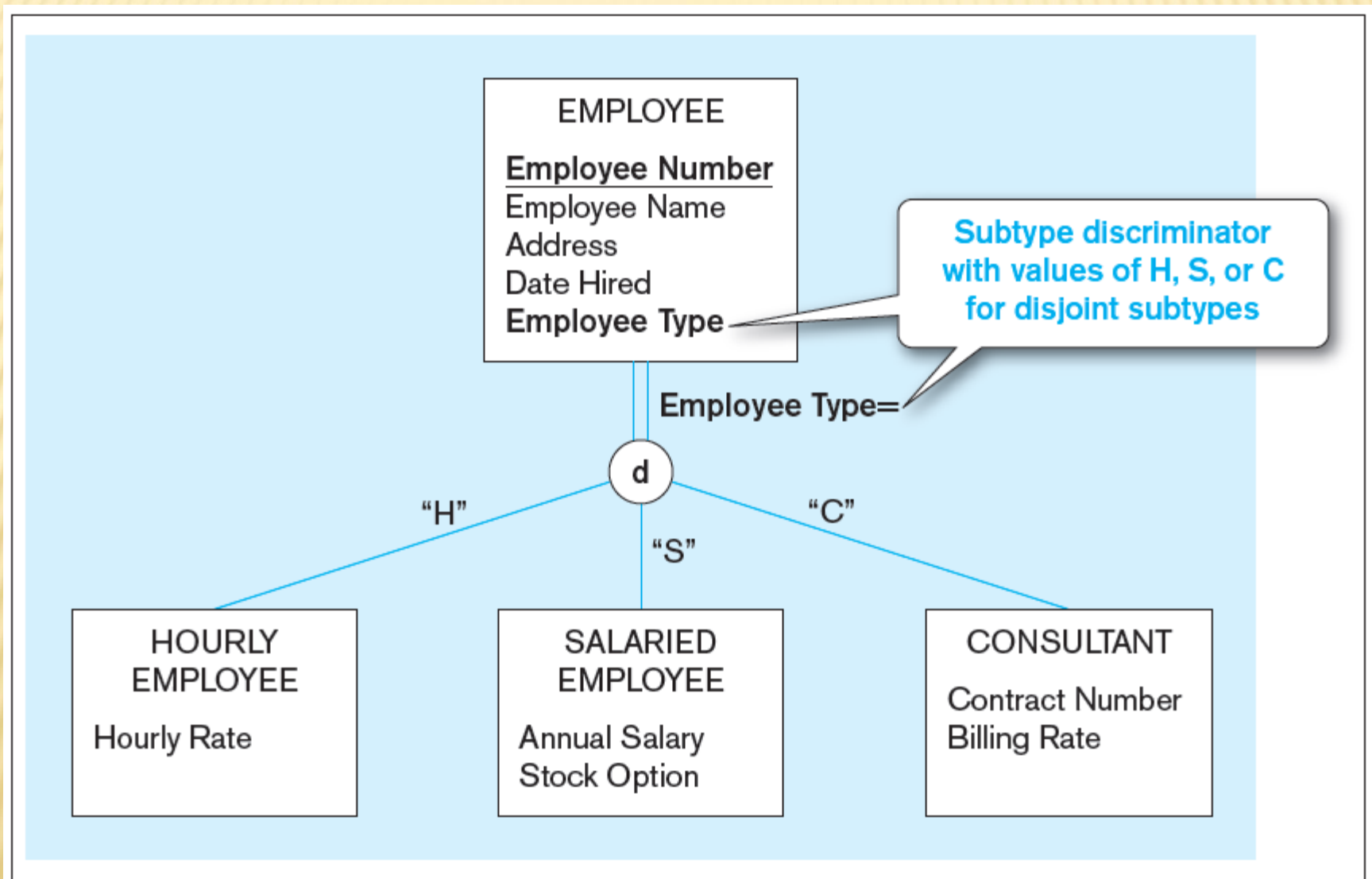
Subtype discriminator (*overlap* rule)
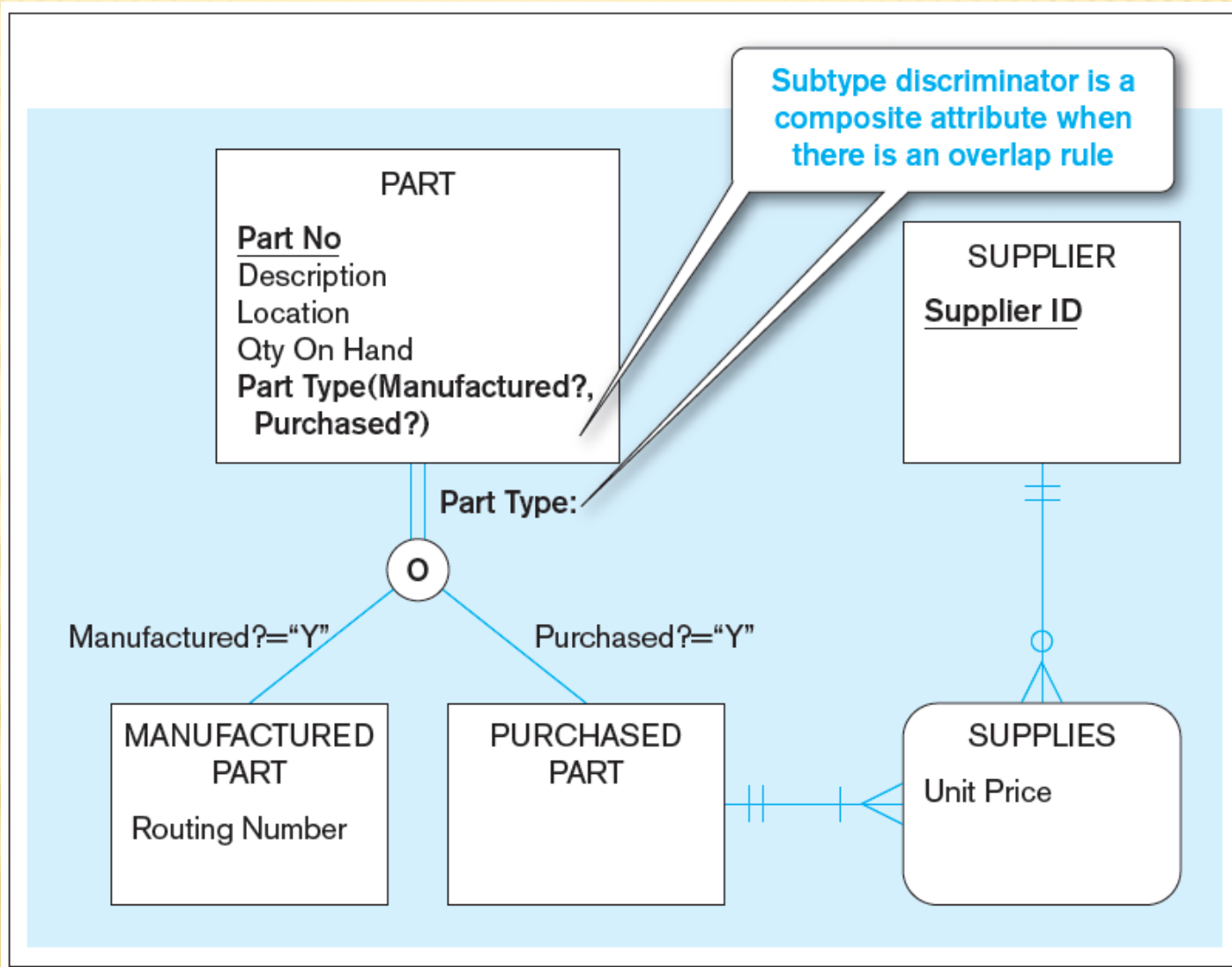
Example of supertype/subtype hierarchy

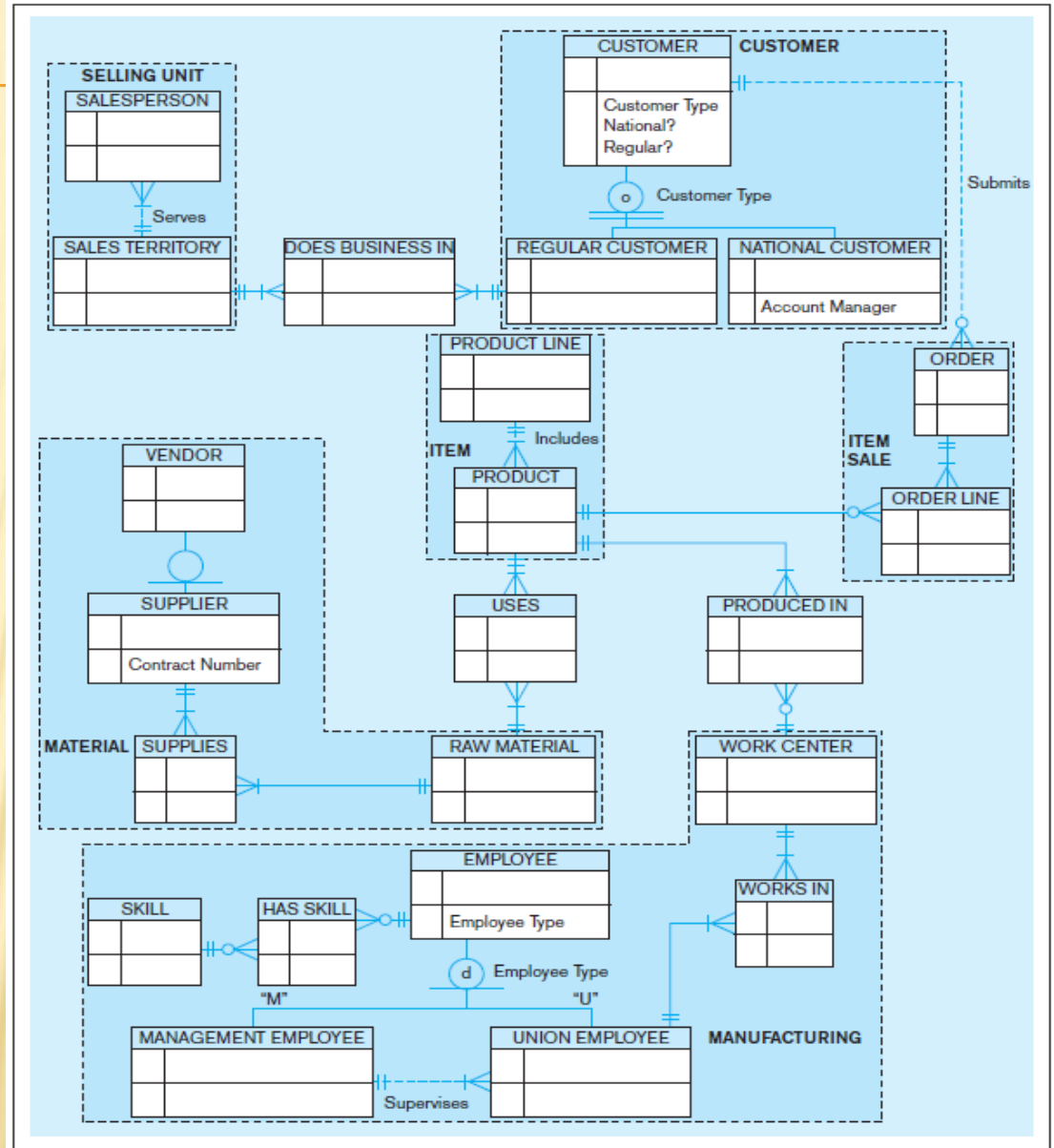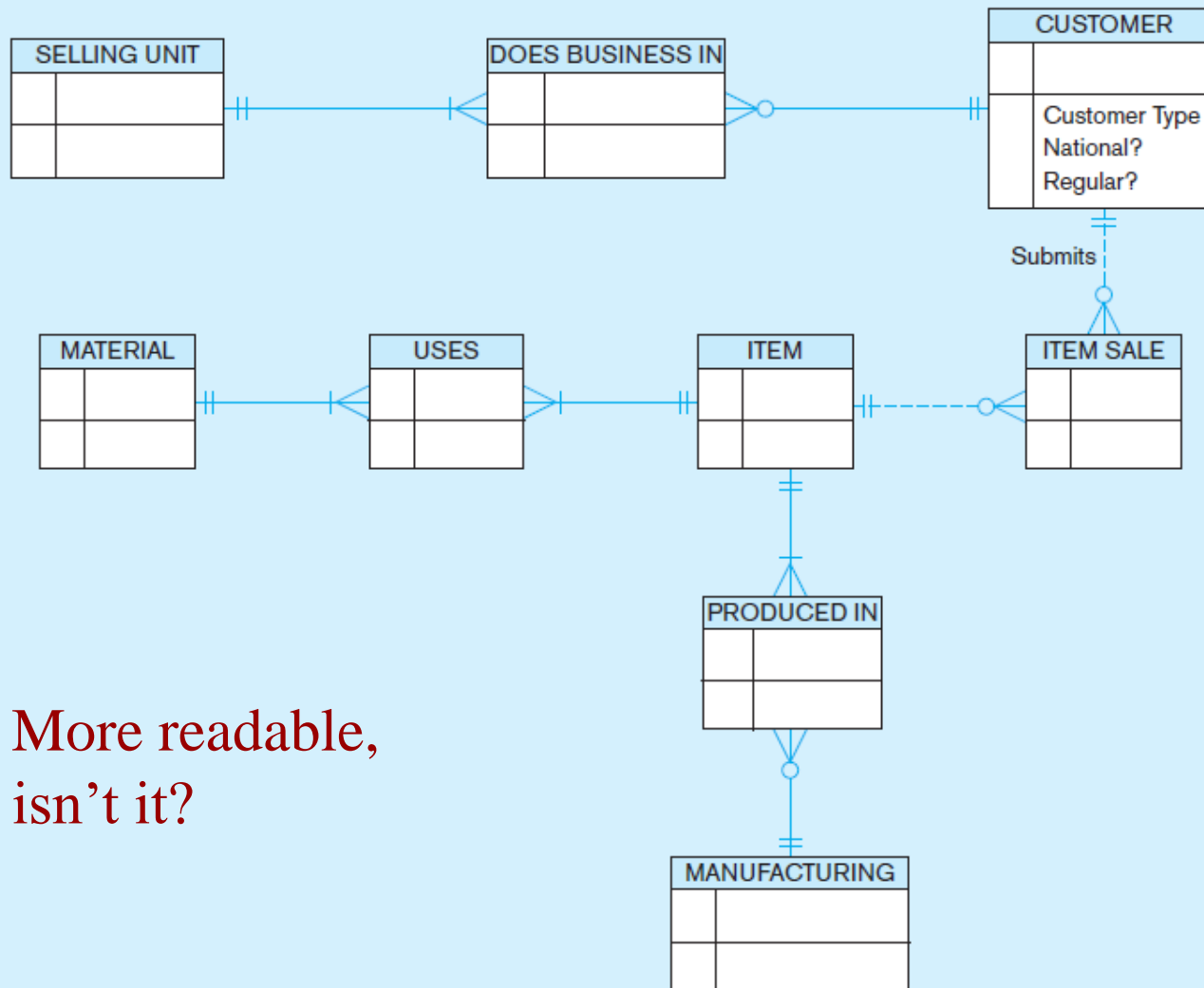# ENTITY CLUSTERS

- EER diagrams are difficult to read when there are too many entities and relationships.

- Solution: Group entities and relationships into *entity clusters.*

- **Entity cluster:** Set of one or more entity types and associated relationships grouped into a single abstract entity type

Possible entity clusters for Pine Valley Furniture in Microsoft Visio

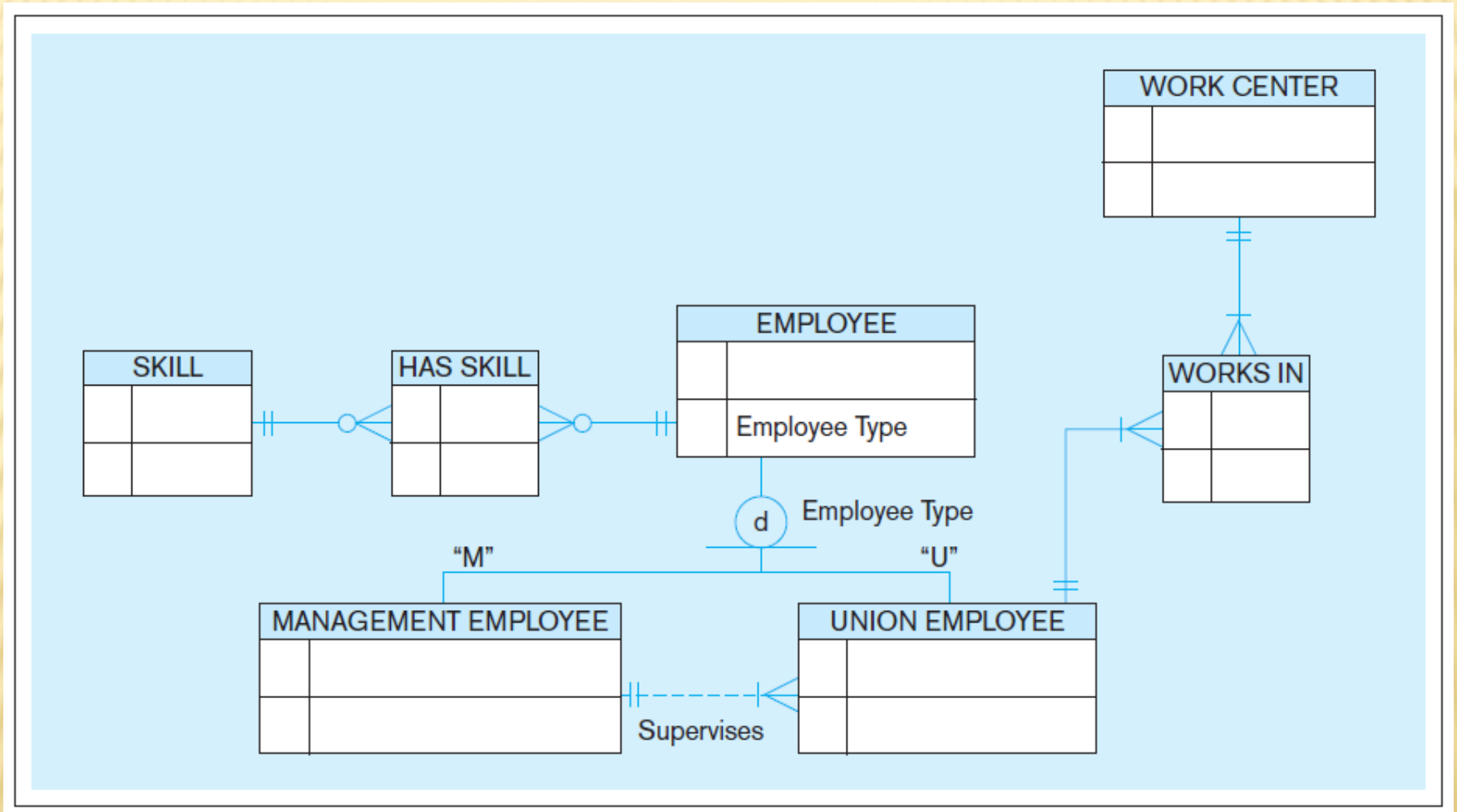Related groups of entities could become clusters

EER diagram of PVF entity clusters

More readable, isn't it?

Manufacturing entity cluster


Detail for a single cluster

PACKAGED DATA MODELS

- Predefined data models
- Could be universal or industry-specific
- Universal data model = a generic or template data model that can be reused as a starting point for a data modeling project (also called a "pattern")

ADVANTAGES OF PACKAGED DATA MODELS

- ✖ Use proven model components
- ✖ Save time and cost
- ✖ Less likelihood of data model errors
- ✖ Easier to evolve and modify over time
- ✖ Aid in requirements determination
- ✖ Easier to read
- ✖ Supertype/subtype hierarchies promote reuse
- ✖ Many-to-many relationships enhance model flexibility
- ✖ Vendor-supplied data model fosters integration with vendor's applications
- ✖ Universal models support inter-organizational systems

PARTY, PARTY ROLE, and ROLE TYPE in a universal data model

(a) Basic PARTY universal data model
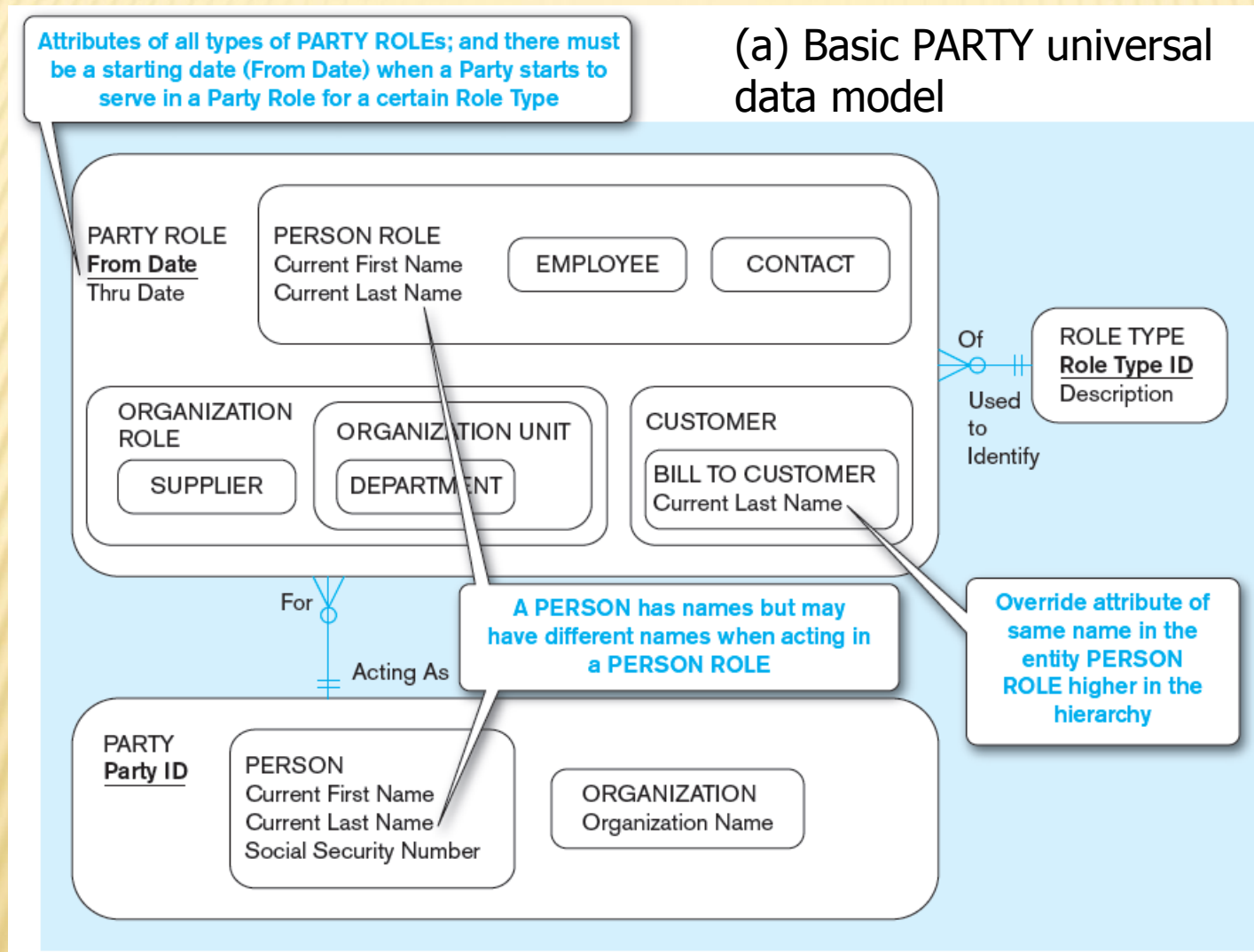
Packaged data models are generic models that can be customized for a particular organization's business rules.

PARTY, PARTY ROLE, and ROLE TYPE in a universal data model

(b) PARTY supertype/subtype hierarchy