Lesson 4:

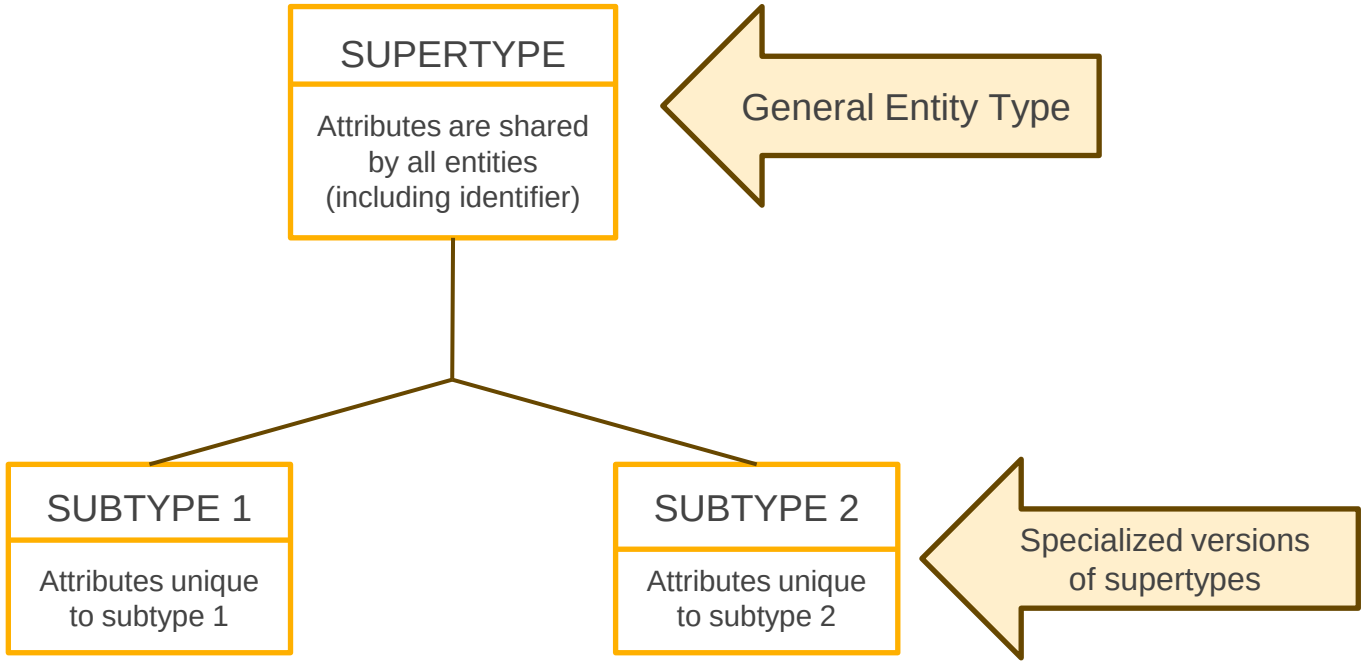
Advanced Data Modeling

Supertype & Subtype

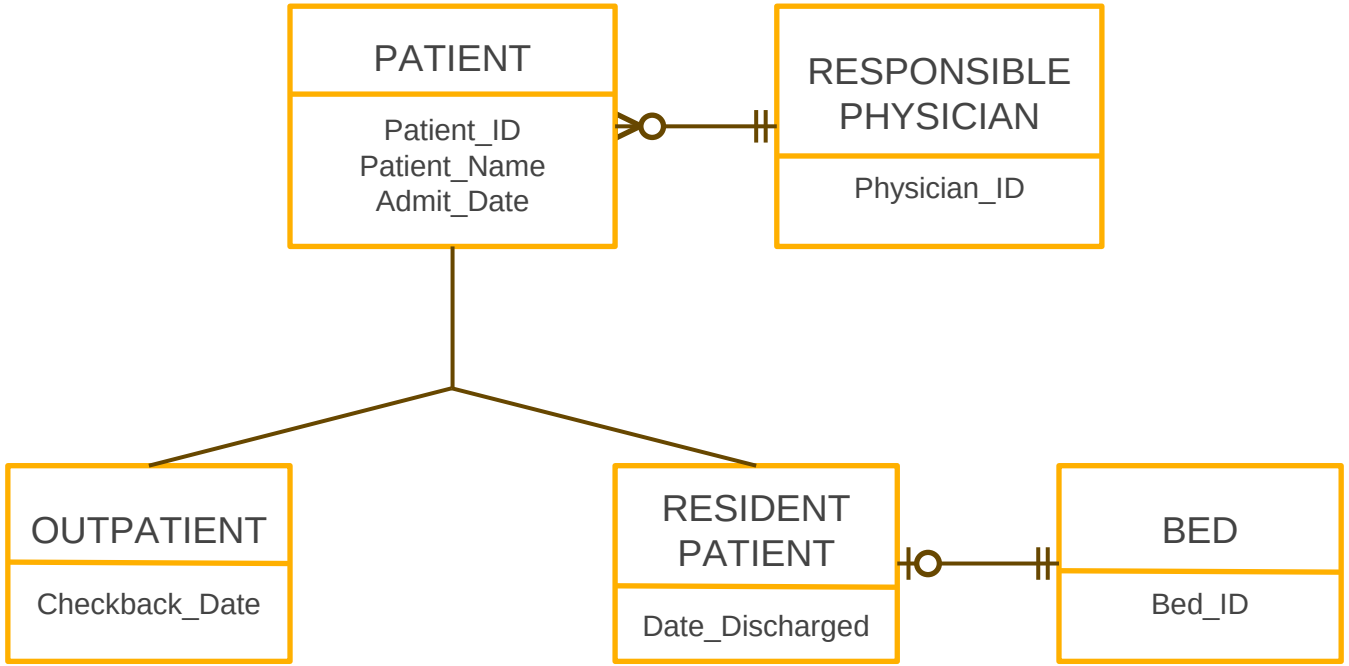
Supertype & Subtype

- An entity can be divided into several subgroups or subtypes
- An entity that can be divided into subtypes is called super type
- **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



Supertype/subtype relationships in a hospital

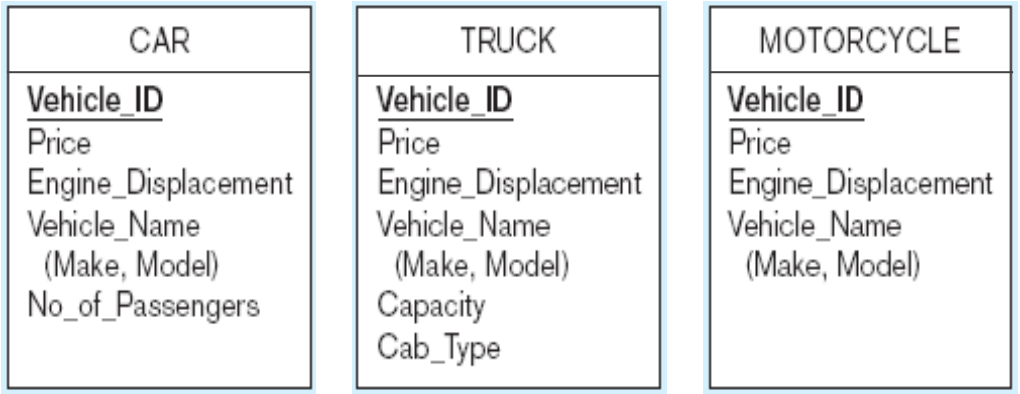


Generalization and Specialization

- **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**

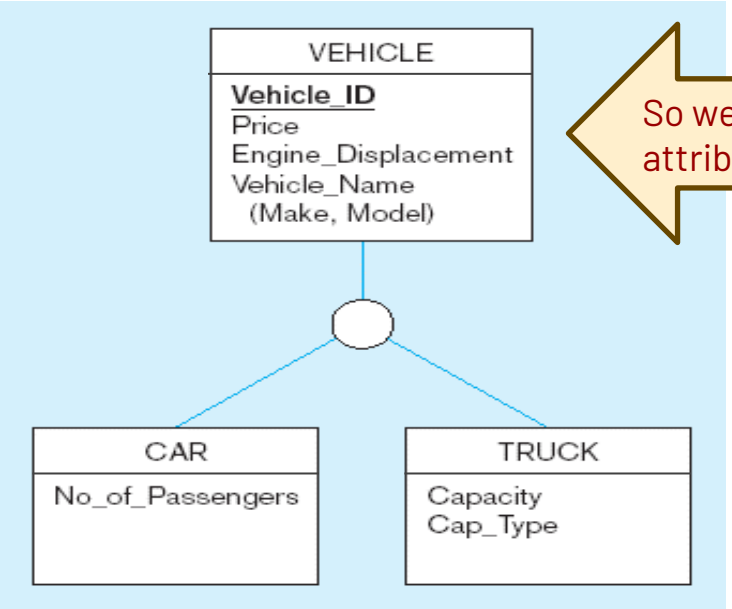
Generalization

- Three entity types: CAR, TRUCK, and MOTORCYCLE



Generalization

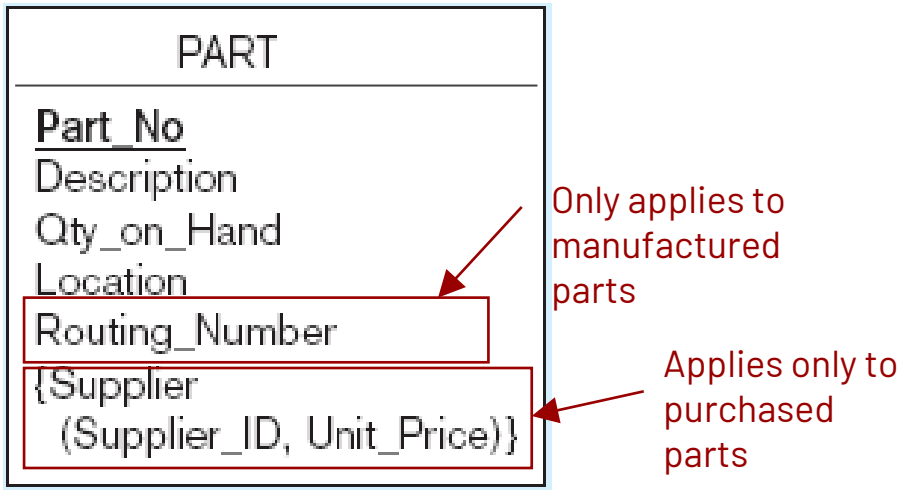
- Generalization to VEHICLE supertype



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

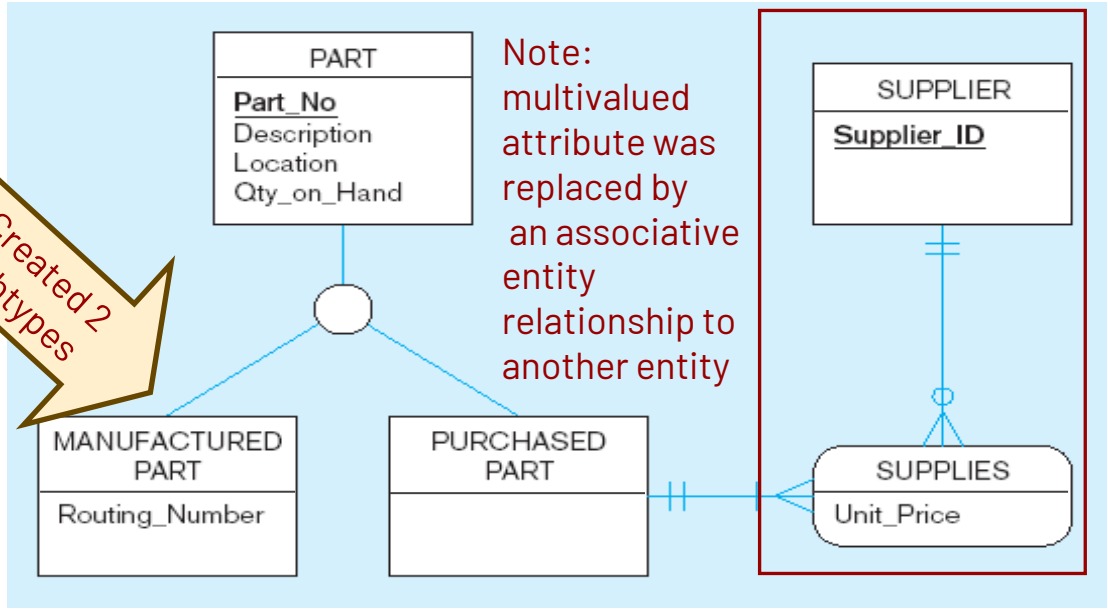
Specialization

- Entity type PART



Specialization

- Specialization to MANUFACTURED PART and PURCHASED PART

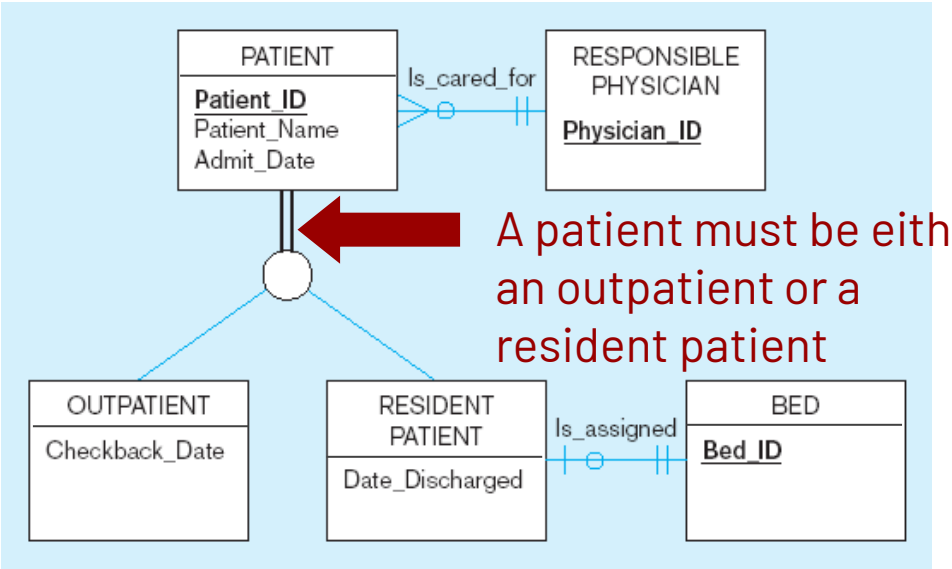


Constraints in Supertype/ Completeness Constraint

- **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)

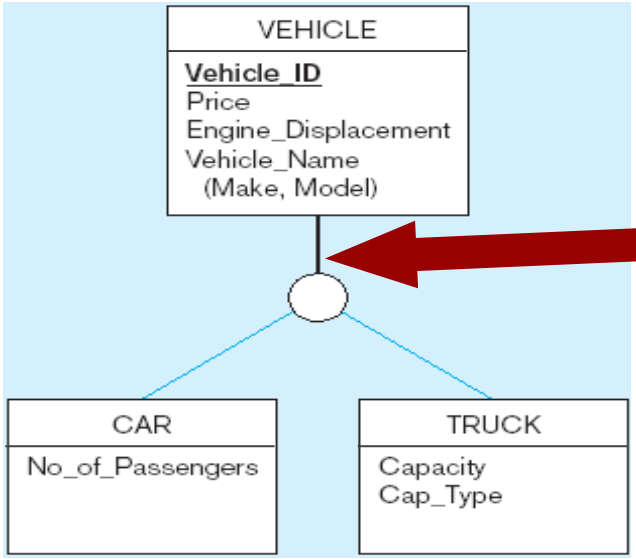
Constraints in Supertype/ Completeness Constraint

- Examples of completeness constraints
  - Total specialization rule



Constraints in Supertype/ Completeness Constraint

- Examples of completeness constraints
  - Partial specialization rule



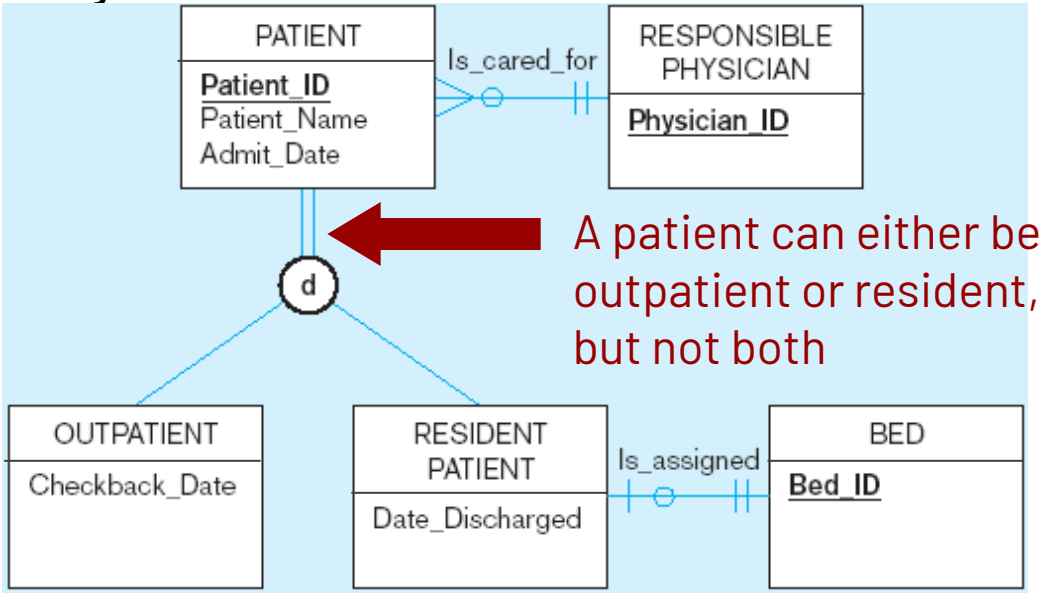
A vehicle could be a car, a truck, or neither

Constraints in Supertype/ Disjointness constraint

- **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

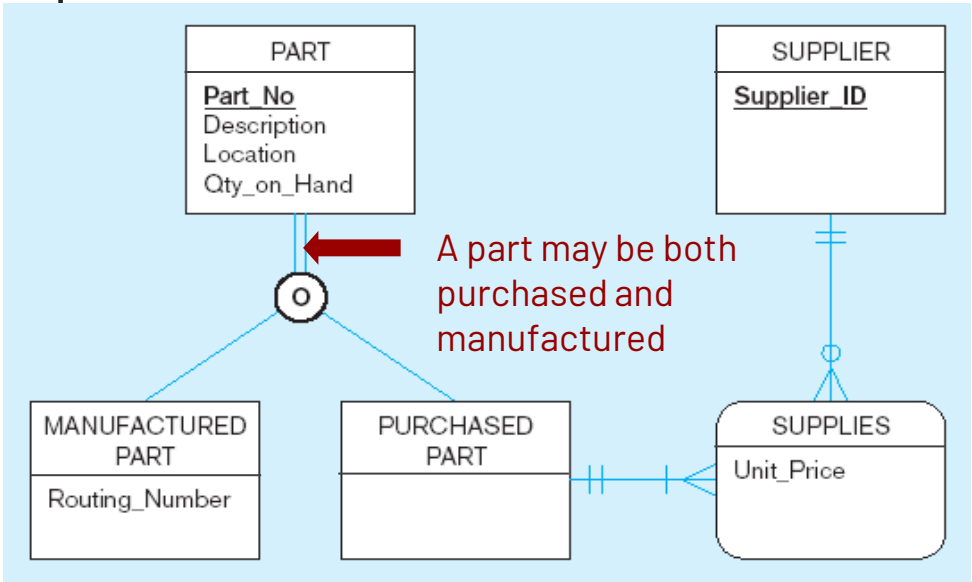
Constraints in Supertype/ Disjointness constraint

- Examples of disjointness constraints
  - Disjoint Rule



Constraints in Supertype/ Disjointness constraint

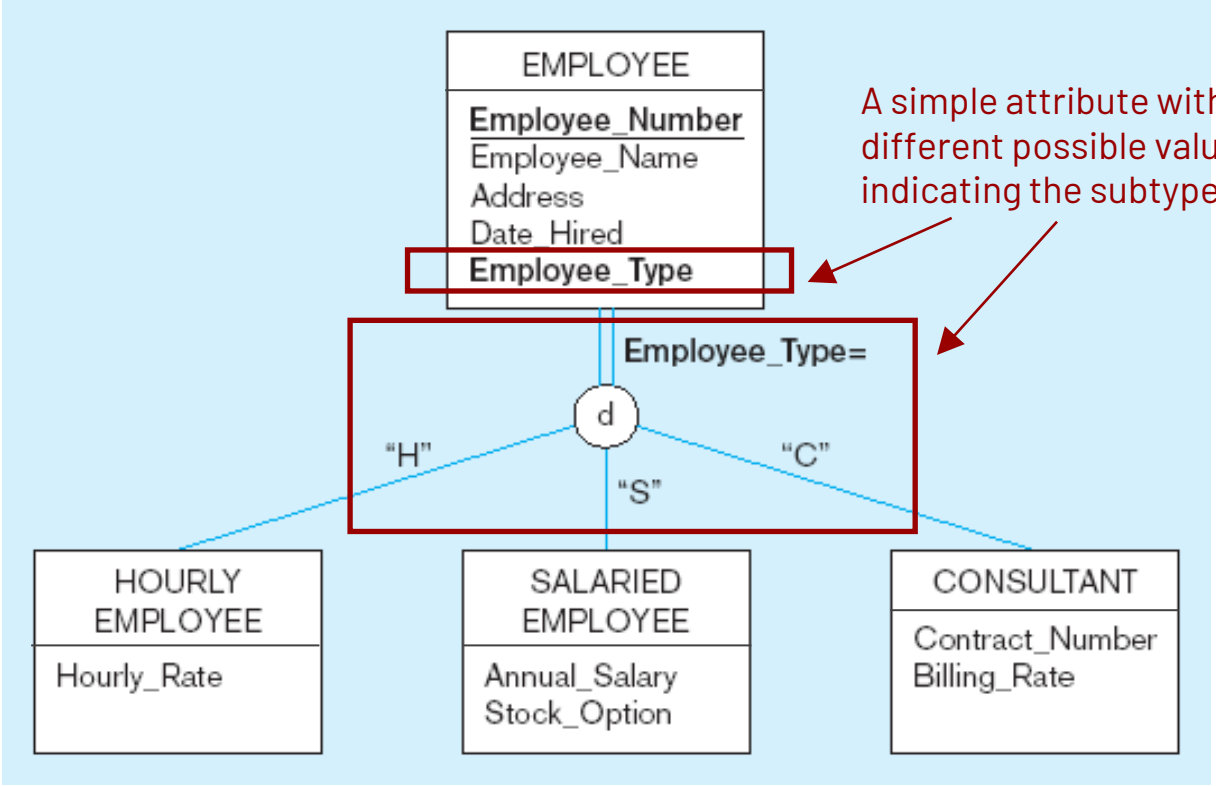
- Examples of disjointness constraints
  - Overlap Rule



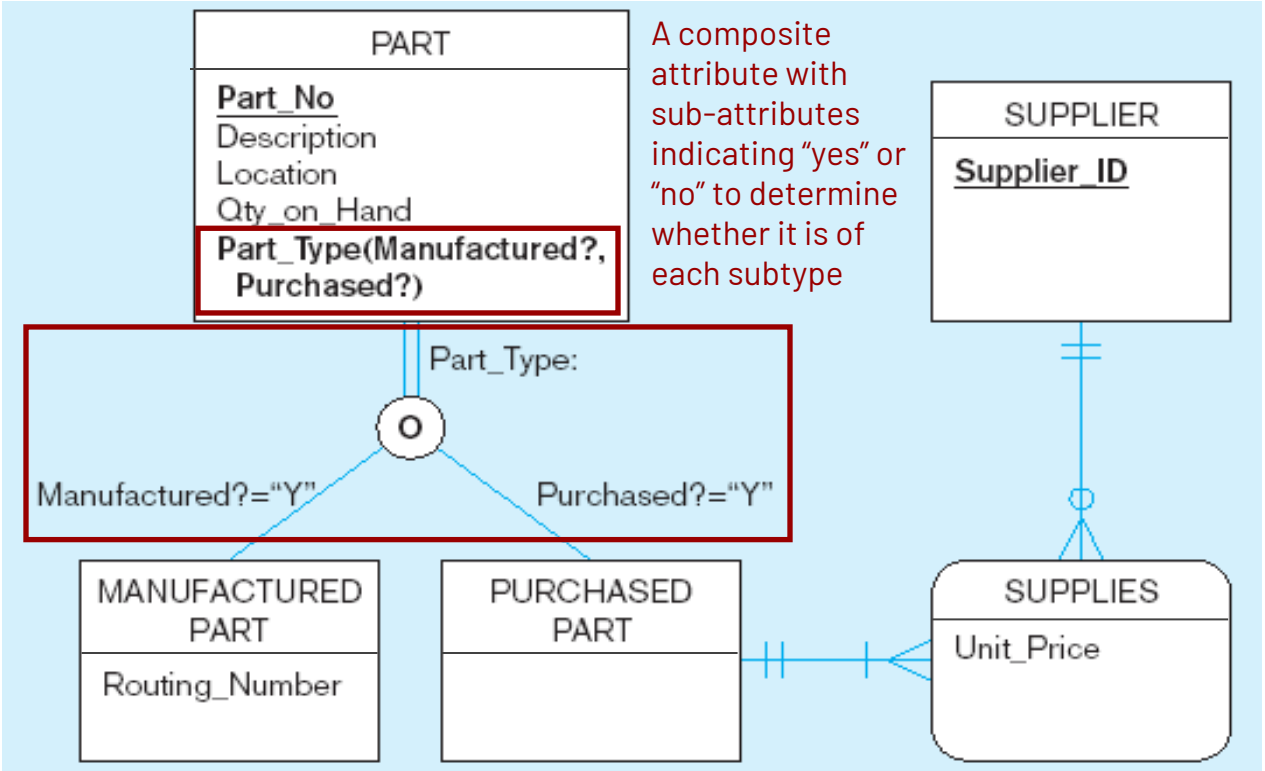
Constraints in Supertype/ Subtype Discriminators

- **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

Constraints in Supertype/ Subtype Discriminators (DISJOINT RULE)



Constraints in Supertype/ Subtype Discriminators (OVERLAP RULE)



Example of supertype/subtype hierarchy