

# Chapter 4 - The Enhanced Entity-Relationship (EER) Model

2018-09-04

## 4.1 - Subclasses, superclasses, and inheritance

↳ EER model includes all the modeling concepts of ER model, with the addition of subclass and superclass, and the related concepts of specialization and generalization.

↳ Another concept introduced is that of category or union type, which represents a collection of objects that is a union of objects of different entity types.

Superclass / Subclass: A subclass is a entity belonging (with potential other entities) to a superclass, or parent group.

↳ Also called supertype / subtype relationships

↳ Subclass is same real-world entity as the superclass

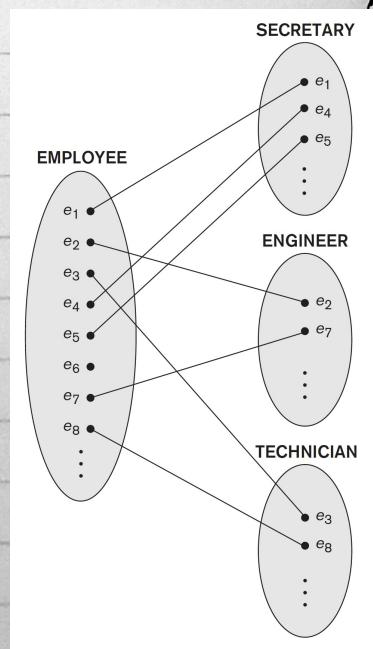
Type inheritance: Because a subclass has a superclass, it inherits all the attributes of the superclass.

## 4.2 - Specialization and Generalization

Specialization: process of defining a set of subclasses of entity type

Specific / local attributes: attributes that only apply to entities of a subclass

↳ Subclass can also participate in specific relationship types



→ Entity SECRETARY, ENGINEER, TECH belong to the same real-world entity.

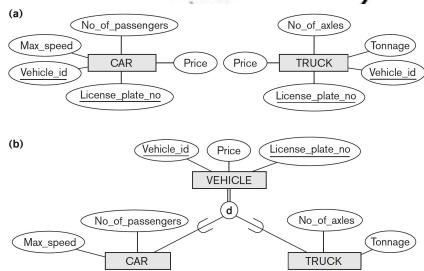
→ For example, e<sub>1</sub> is shown twice; as an employee and a secretary

→ Different from 1:1 relationship because 1:1 relationship are between distinct entity types

Summary: Subclasses / Superclasses all represent the same real-world entity-type. These subclasses and superclasses also can have specific and local attributes. They can also participate in specific relationship types.

## Chapter 4 - The EER Model

4.2.2 - Generalization: The reverse of specialization; suppression of the differences between several entity types, identifying their common features, and generalizing them into a single superclass.

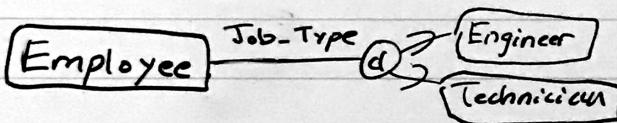


→ Example: CAR and TRUCK have several things in common; they can be generalized into a superclass called VEHICLE.

### 4.3.1 - Constraints on Specialization / Generalization

→ Some subclasses are called predicate-defined, which means they are categorized based on some attribute of the superclass.

↳ Example: "Job-type" as a predicate for subclass of employee



↳ If all subclasses have their membership condition on the same attribute of the superclass, the specialization is called attribute-defined specialization.

↳ When no condition for determining membership in subclass exists, the subclass is called user-defined.

\* First constraint group  
↳ Two other constraints apply: disjointness and overlapping.

↳ Disjoint: → entity can be a member of at most one subclass

↳ Overlap: → entity may be a member of more than one subclass

\* Second constraint group  
↳ Completeness constraint: total or partial

↳ Total specialization constraint specifies that every entity in superclass must be a member of at least one subclass (Double-line)!

↳ Partial specialization allows entity not to belong to any of the subclasses (single line!)

Summary: Generalization is the opposite of specialization, which happens when entity types share similar attributes. Special/general also have constraints, some are predicate-defined, or user-defined.

Disjointness/overlap and completeness are other constraints.

# Chapter 4 - The EER Model

2018-09-04

→ Disjointness and completeness constraints are independent. Four possibilities: (1) disjoint, total (2) disjoint, partial (3) overlapping, total (4) overlapping, partial

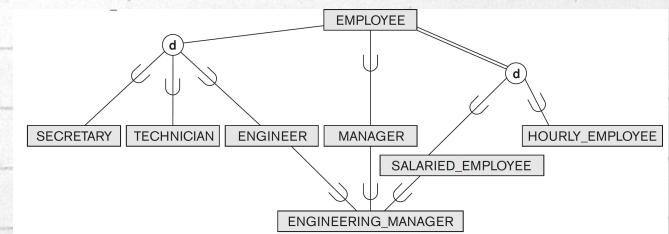
→ Certain insertion / deletion rules apply as a consequence of previous constraints:

↳ Deleting entity from superclass implies it is automatically deleted from all subclasses.

↳ Inserting entity in superclass implies also inserted into predicate / attribute defined subclasses.

↳ Inserting entity in superclass of total specialization implies entity is inserted in at least one of the subclasses of the specialization.

## 4.3.2 - Specialization / Generalization Hierarchies



→ Specialization hierarchy  
Every subclass participates  
as a subclass in only one  
class / subclass relationship

→ Example above : Every engineering manager is required to be an engineer, salaried employee, and manager.

Leaf node: Class that has no subclasses of its own.

Shared subclass: Subclass with more than one Superclass.

Multiple inheritance: Subclass inherits attributes and relationships from multiple superclasses.

## 4.3.3 - Special / General in Refining Conceptual Schemas

Top-down conceptual refinement: Starting with an entity type, then defining subclasses of type by successive specialization

Bottom-up conceptual synthesis: Starting with specialized types, and generalizing your way upward.

Summary: Superclasses have certain constraints when inserting / deleting data. Specialization / generalization involves hierarchies, and single / multiple inheritance.