**Enhanced Entity Relationship Model or Extended Entity Relationship Model**

The ER modeling concepts are sufficient for representing many database schemas for traditional database applications, which include many data-processing applications in business and industry.



However, designers of database applications have tried to design more accurate database schemas that reflect the data properties and constraints more precisely.

This was particularly important for newer applications of database technology, such as databases for engineering design and manufacturing (CAD/CAM)( Computer-Aided Design/Computer-Aided Manufacturing), telecommunications, complex software systems, and geographic information systems (GISs).

These types of databases have requirements that are more complex than traditional applications. These concepts were also developed independently in related areas of computer science, such as the knowledge representation area of artificial intelligence and the object modeling area in software engineering.

The enhanced entity–relationship (EER) model (or extended entity–relationship model) is a high-level or conceptual data model incorporating extensions to the original entity–relationship (ER) model, used in the design of databases.

Subclasses, Superclasses, and Inheritance

The EER model includes all the modeling concepts of the ER model. In addition, it includes the concepts of subclass and superclass and the related concepts of specialization and generalization.

Inheritance is the capability of one class of things to inherit(acquire) capabilities or properties from another class.

Example: Cars

The class ‘car’ inherits some of its properties from the class ‘Automobiles’ which inherits some of its properties from another class ‘vehicles’.

The object oriented language express this inheritance relationship by allowing one class to inherit from another.

The principle behind this sort of division is that each sub class shares common characteristics with the class from which it is derived.

Vehicles



Automobiles Pulled Vehicles

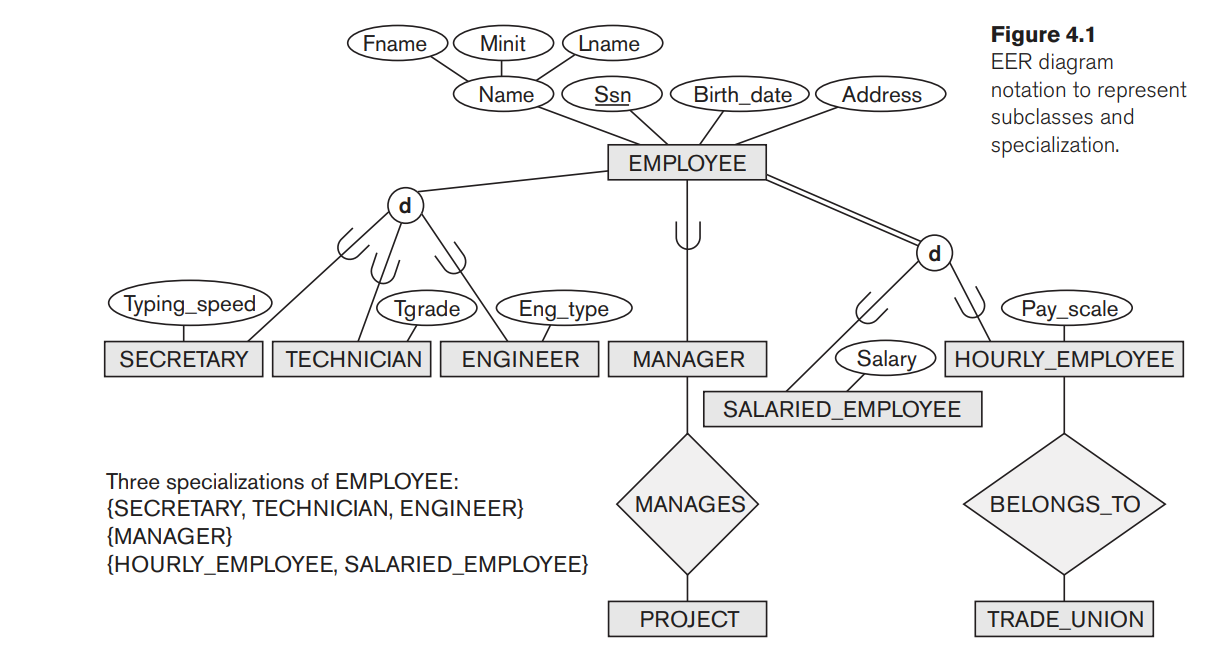
Car Bus Cart Rickshaw

The first enhanced ER (EER) model concept we take up is that of a subtype or subclass of an entity type. The name of an entity type is used to represent both a type of entity and the entity set or collection of entities of that type that exist in the database.

For example, the entities that are members of the EMPLOYEE entity type may be distinguished further into SECRETARY, ENGINEER, MANAGER, TECHNICIAN, SALARIED\_EMPLOYEE, HOURLY\_EMPLOYEE, and so on.

The set or collection of entities in each of the latter groupings is a subset of the entities that belong to the EMPLOYEE entity set, meaning that every entity that is a member of one of these subgroupings is also an employee. We call each of these subgroupings a subclass or subtype of the EMPLOYEE entity type, and the EMPLOYEE entity type is called the superclass or supertype for each of these subclasses.

Figure 4.1 shows how to represent these concepts diagramatically in EER diagrams.



**Specialization**

Specialization is the process of defining a set of subclasses of an entity type.This entity type is called the superclass of the specialization.

The set of subclasses that forms a specialization is defined on the basis of some distinguishing characteristic of the entities in the superclass.

For example, the set of subclasses {SECRETARY,ENGINEER, TECHNICIAN} is a specialization of the superclass EMPLOYEE that distinguishes among employee entities based on the job type of each employee.

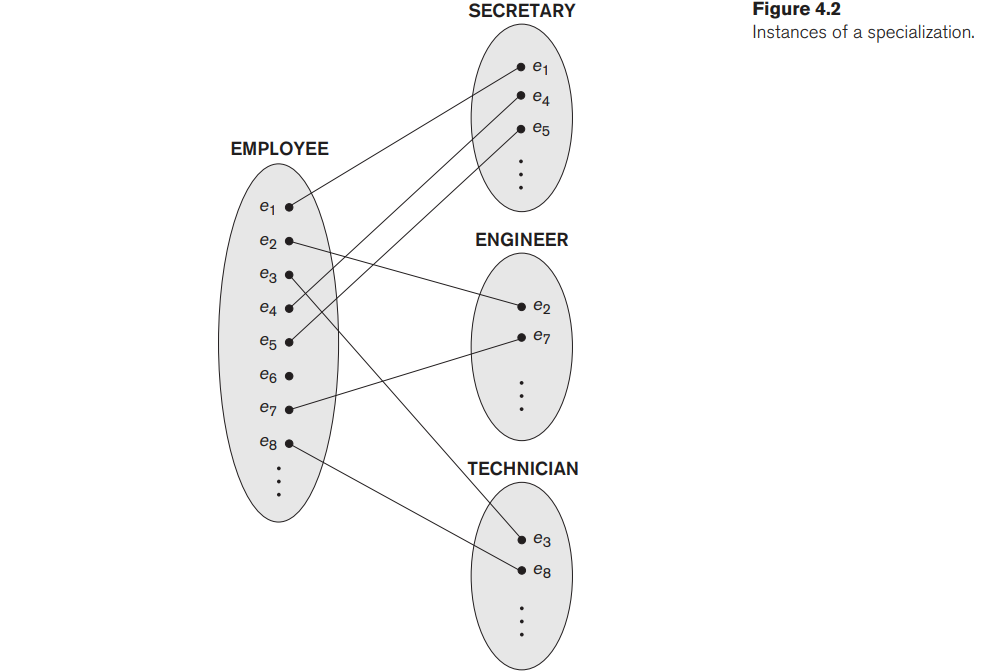
We may have several specializations of the same entity type based on different distinguishing characteristics. For example, another specialization of the EMPLOYEE entity type may yield the set of subclasses {SALARIED\_EMPLOYEE,HOURLY\_EMPLOYEE}. This specialization distinguishes among employees based on the method of pay.

Figure 4.1 shows how we represent a specialization diagrammatically in an EER diagram. The subclasses that define a specialization are attached by lines to a circle that represents the specialization, which is connected in turn to the superclass.

The subset symbol on each line connecting a subclass to the circle indicates the direction of the superclass/subclass relationship. Attributes that apply only to entities of a particular subclass—such as TypingSpeed of SECRETARY—are attached to the rectangle representing that subclass. These are called specific (or local) attributes of the subclass. Similarly, a subclass can participate in specific relationship types, such as the HOURLY\_EMPLOYEE subclass participating in the BELONGS\_TO relationship in Figure 4.1.

Figure 4.2 shows a few entity instances that belong to subclasses of the {SECRETARY,

ENGINEER, TECHNICIAN} specialization. In a superclass/subclass relationship the entity in the subclass is the same real-world entity as the entity in the superclass but is playing a specialized role—for example, an EMPLOYEE specialized in the role of SECRETARY, or an EMPLOYEE specialized in the role of TECHNICIAN.

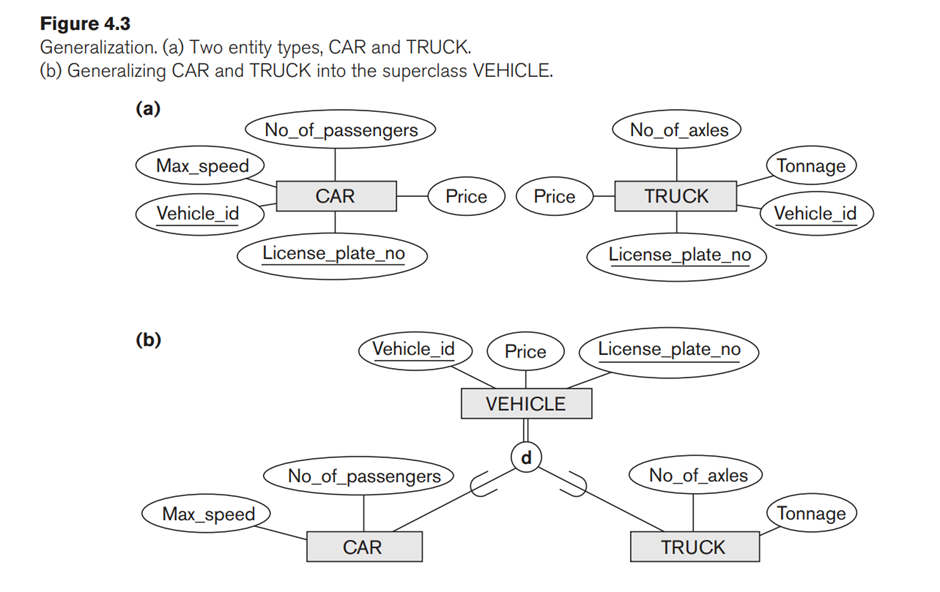


**Generalization**

Generalization is the reverse process of specialization. In generalization, the sub entities are combined together into a super entity set on the basis of some common features in such a way that the new entity thus formed contains all the features of the sub entities.

Generalization is a Bottom-to-Up approach.

For example, consider the entity types CAR and TRUCK shown in Figure 4.3(a). Because they have several common attributes, they can be generalized into the entity type VEHICLE, as shown in Figure 4.3(b). Both CAR and TRUCK are now subclasses of the generalized superclass VEHICLE.



A diagrammatic notation to distinguish between generalization and specialization is used in some design methodologies. An arrow pointing to the generalized superclass represents a generalization process, whereas arrows pointing to the specialized subclasses represent a specialization process.

In general, a superclass or subclass represents a collection of entities of the same type.