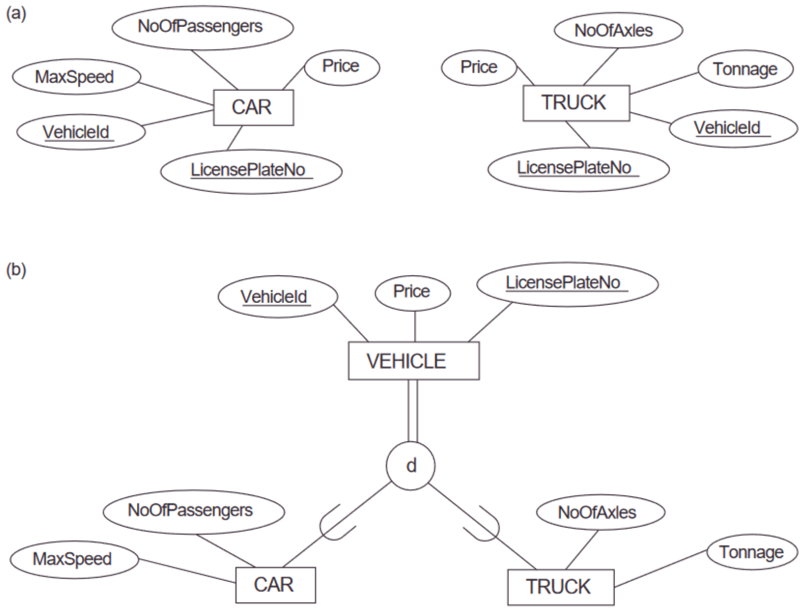
EER: Extended (or Enhanced) ER Model

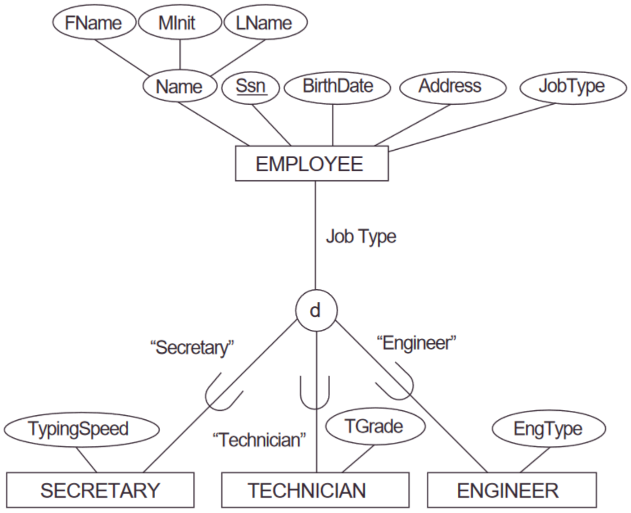
- *Generalization* is the result of computing the union of two or more entity sets (*subclasses*) to produce a higher-level entity set (*superclass*). It represents the containment relationship that exists between the higher-level entity set and one or more lower level entity sets.

  Examples of generalization. *(a) Two entity types CAR and TRUCK.*
  (b) Generalizing car and TRUCK into VEHICLE.



- *Specialization* constructs the lower level entity sets that are a subset of higher-level entity set. Specialization is the reverse of generalization (for the remainder, we focus on specialization without loss of generality).
- There might exist many specialization of the same entity set based on different distinguishing characteristics. Hence, an entity can be a member of a number of subclasses.
- An entity cannot merely exist by being a member of a subclass but no superclass. However, it is not essential that every entity in a superclass be a member of some subclass.
- Why specialization:
  1. Define a set of subclasses of an entity set.
  2. Associate additional specific attributes with each subclass.
  3. Establish additional specific relationship sets between each subclass and other entity sets.

- Example: An attribute-defined specialization on JobType attribute of EMPLOYEE.



Different types of specialization

- **Predicate-defined** (or condition-defined): Determine subclass membership by examining the value of a specific attribute (termed, **defining attribute**).
- **User-defined**: The user specifies subclass membership individually for each entity.

Constraints

- **Disjoint**: An entity can be a member of *at most one* of the subclasses.
- **Overlap**: When the subclasses are not disjoint.
- **Total:** Every entity in the superclass must be a member of some subclass.
- **Partial:** An entity might belong to no subclass.