

Fundamentals of Database Systems

Chapter 6: Mapping EER

Converting EER Diagram to Database Schema





Overview

☉ We learned that:

1. A DBMS will allow us to create and manage databases.
2. We can query the database (retrieve data from tables in the database).
3. We can update the database (change, add, or delete data from tables in the database).
4. We can complete a conceptual design of the database using the ER and the EER model.
5. We know how to convert an ERD to a database schema.



This chapter

- **Learning how to convert the EER diagram to the database schema.**



Mapping EER diagrams to relational/database schema

◎ **To map specialization or generalization, four options:**

- a. Multiple relations-Superclass and subclasses.
- b. Multiple relations-Subclass relations only
- c. Single relation with one type attribute (with disjoint constraint)
- d. Single relation with multiple type attribute (with overlapping constraint)



Option a. Multiple relations-Superclass and subclasses

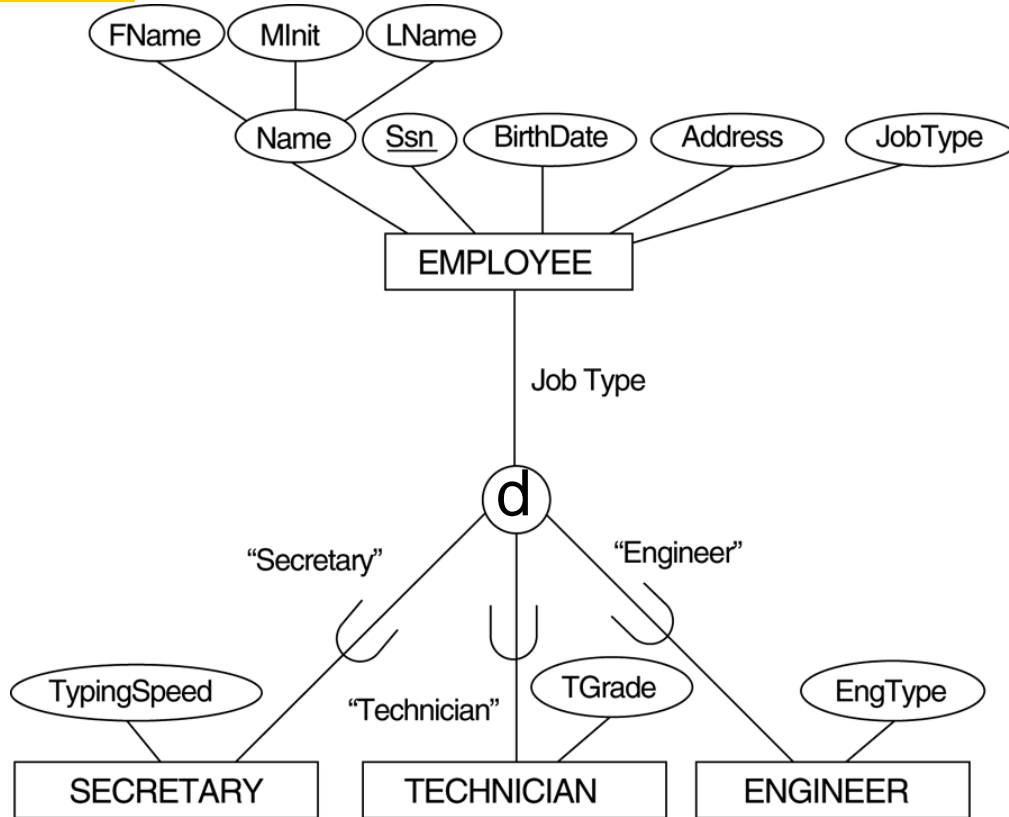
● How to convert:

- Create a relation for a superclass and its attributes
- Create a relation for each subclass includes local/specific attributes
- The primary key of the superclass which is propagated to subclass relation and becomes a primary key also, it also becomes foreign key to superclass relation.

Note: works for all constraints in the specialization

Option a. Multiple relations-Superclass and subclasses

Example of an EER



Option a. Multiple relations-Superclass and subclasses

Results based on option a

(a) EMPLOYEE

| | | | | | | |
|------------|-------|-------|-------|-----------|---------|---------|
| <u>SSN</u> | FName | MInit | LName | BirthDate | Address | JobType |
|------------|-------|-------|-------|-----------|---------|---------|

SECRETARY

| | |
|------------|-------------|
| <u>SSN</u> | TypingSpeed |
|------------|-------------|

TECHNICIAN

| | |
|------------|--------|
| <u>SSN</u> | TGrade |
|------------|--------|

ENGINEER

| | |
|------------|---------|
| <u>SSN</u> | EngType |
|------------|---------|



Option b. Multiple relations-Subclass relations only

- **How to convert:**

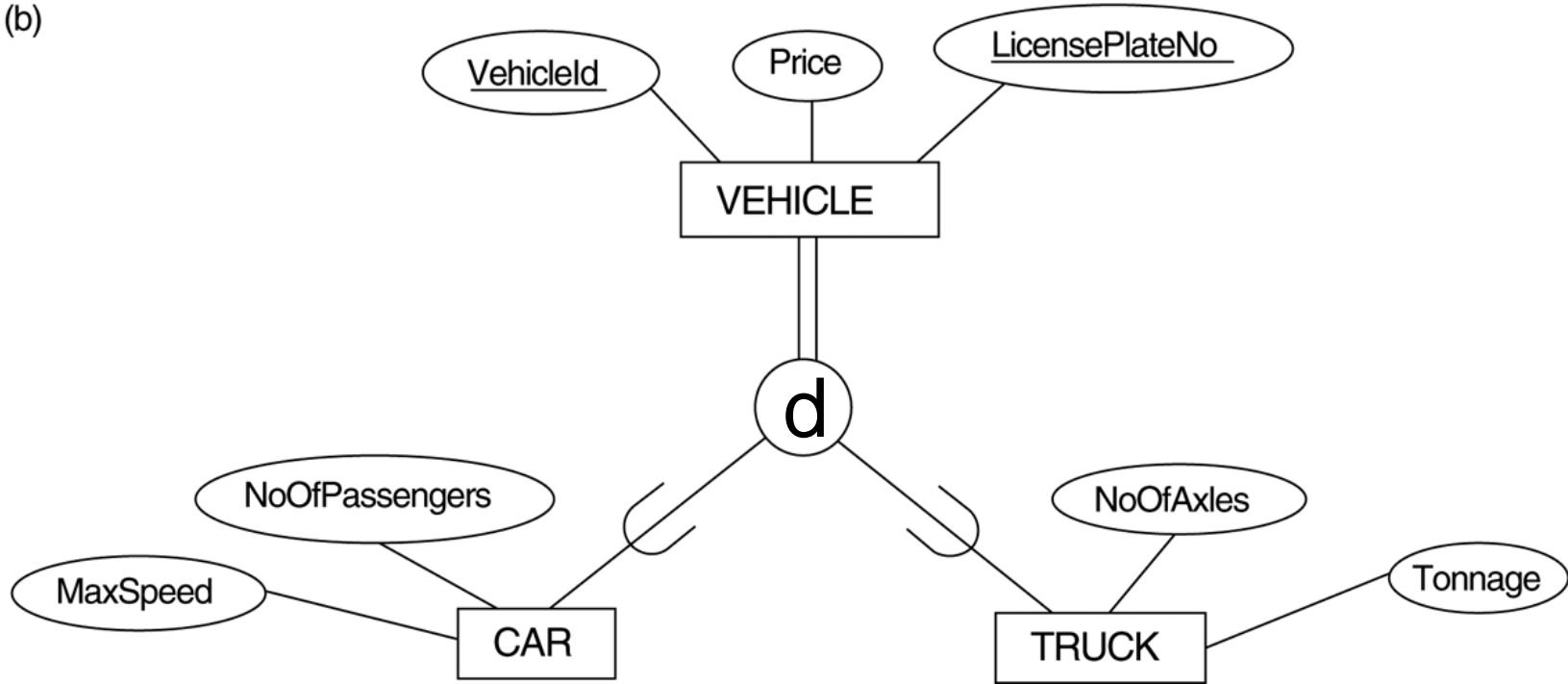
- Create a relation for each subclass includes the local attributes and all attributes for superclass.
- The primary key of the superclass is propagated to subclass relation and becomes its primary key.

Note: works only when both disjoint and total constraint

Option b. Multiple relations-Subclasses relations only

Example of an EER

(b)



Option b. Multiple relations-Subclasses relations only
Results based on option b

(b) CAR

| | | | | |
|------------------|----------------|-------|----------|----------------|
| <u>VehicleId</u> | LicensePlateNo | Price | MaxSpeed | NoOfPassengers |
|------------------|----------------|-------|----------|----------------|

TRUCK

| | | | | |
|------------------|----------------|-------|-----------|---------|
| <u>VehicleId</u> | LicensePlateNo | Price | NoOfAxles | Tonnage |
|------------------|----------------|-------|-----------|---------|



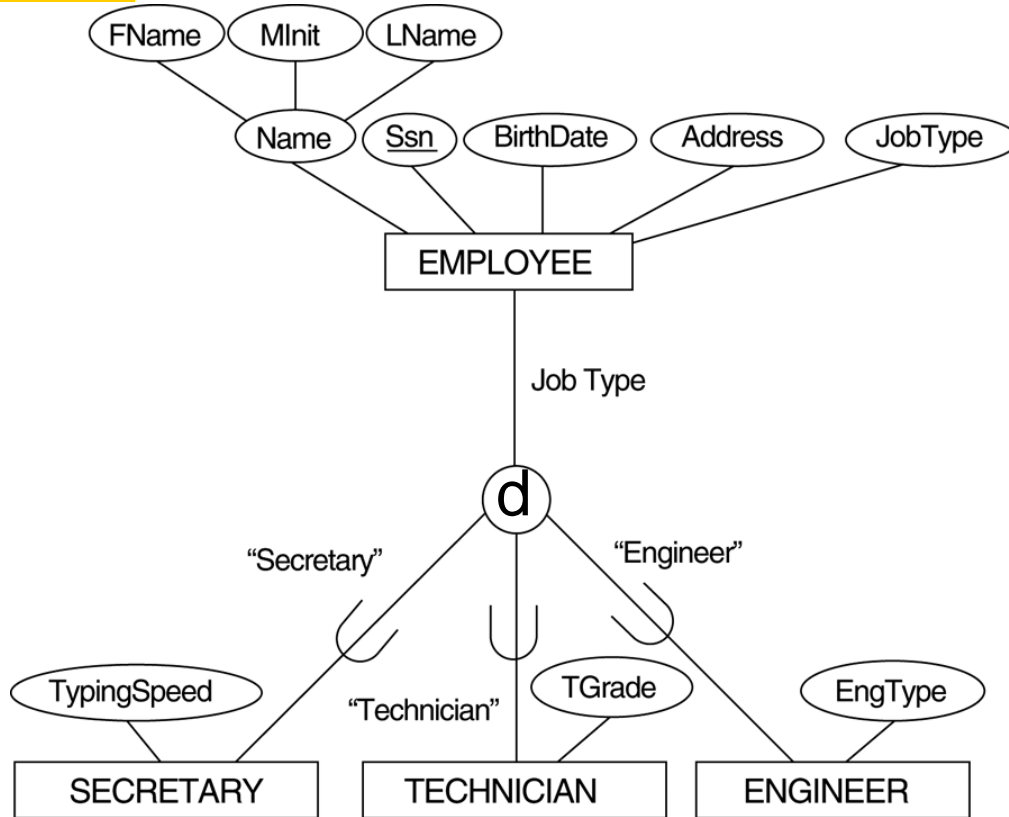
Option c. Single relation with one type attribute

- **How to convert:**

- Create only one relation and add a “type” attribute.
- This type attribute can be used to specify the subclass.
- If the specialization is partial, the “type” attribute can have null values (why?)

Option c. Single relation with one type attribute

Example of an EER



Option c. Single relation with one type attribute
Results based on option c

(c) EMPLOYEE

| <u>SSN</u> | FName | MInit | LName | BirthDate | Address | JobType | TypingSpeed | TGrade | |
|------------|-------|-------|-------|-----------|---------|---------|-------------|--------|--|
|------------|-------|-------|-------|-----------|---------|---------|-------------|--------|--|



Secretary or engineer or technician or
Null



Option d. Single relation with multiple type attribute

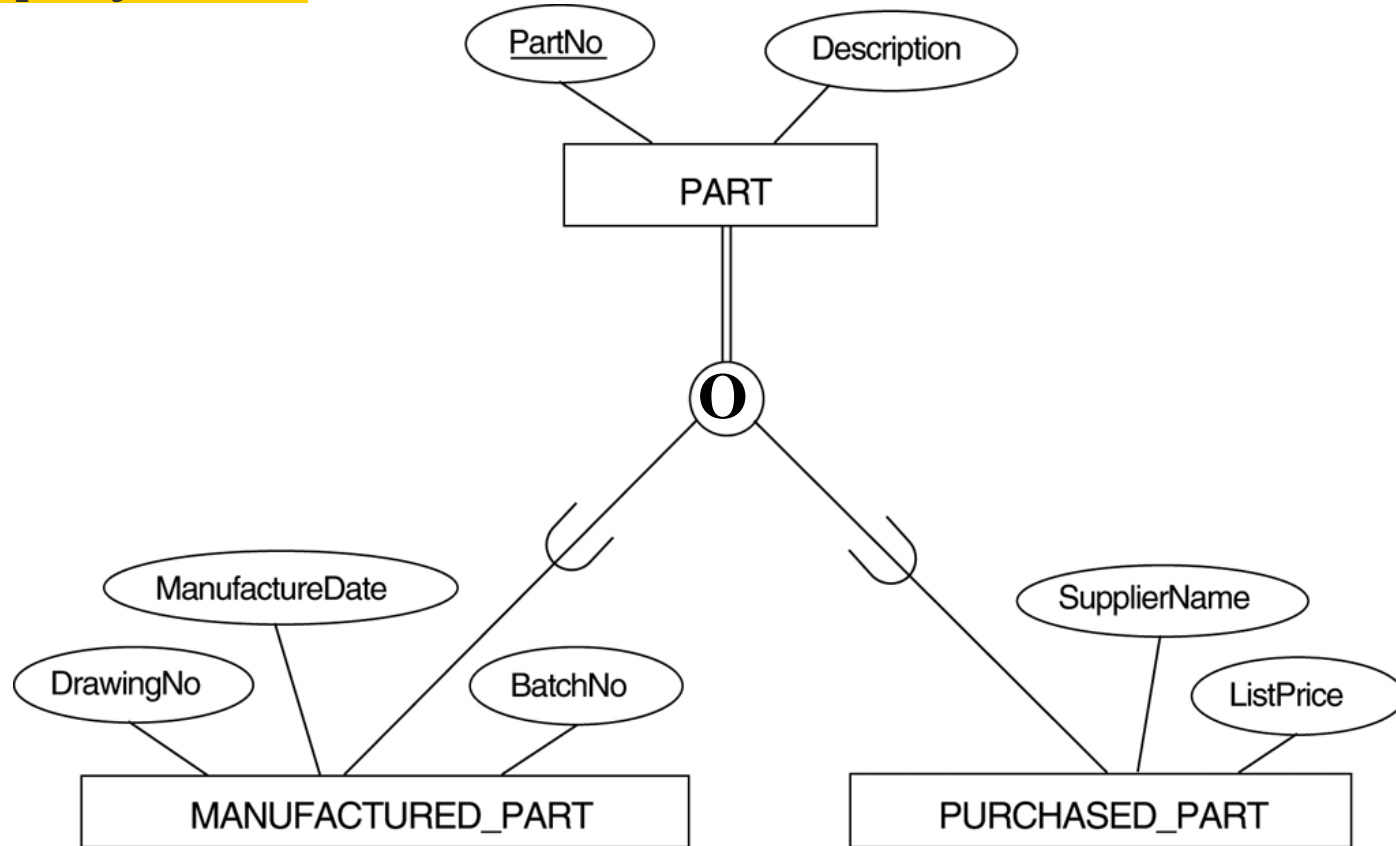
- **How to convert:**

- Create only one relation and add multiple Boolean “type” attribute.
- Each type field can have a domain {Yes, No} where a value of yes indicates that the tuple is a member of the subclass
- So, One type attribute for each subclass.

Note: Used to handle overlapping subclasses

Option d. Single relation with multiple attribute

Example of an EER



Option d. Single relation with multiple type attribute
Results based on option c

(d) PART

| <u>PartNo</u> | Description | MFlag | DrawingNo | ManufactureDate | BatchNo | PFlag | SupplierName | ListPrice |
|---------------|-------------|-------|-----------|-----------------|---------|-------|--------------|-----------|
|---------------|-------------|-------|-----------|-----------------|---------|-------|--------------|-----------|

Boolean type: Yes or No



Boolean type: Yes or No

