

Logical Database Design - Mapping EER Model to Relational

Mapping EER Model Concepts to Relations

Map Superclass/Subclass Relationships

- The relational model does not directly support Superclass/Subclass Relationships.
- There are various strategies that database designers can use to represent these relationships with the relational data model.

Map Superclass/Subclass Relationships

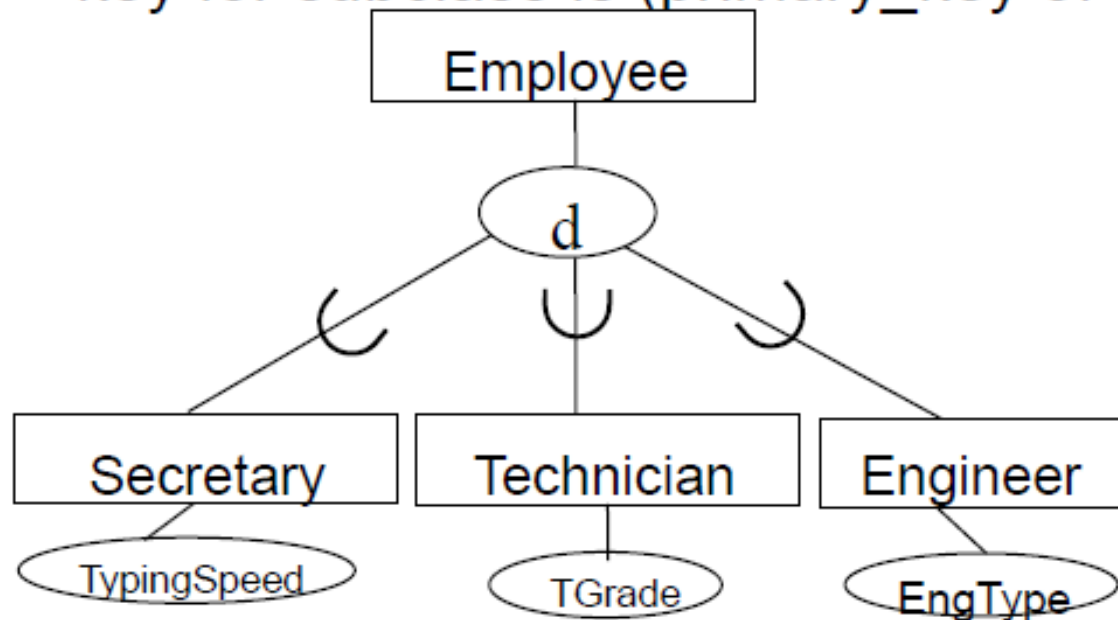
- Identify superclass as parent entity and subclass entity as child entity.
- There are various options on how to represent such a relationship as one or more relations.
- Most appropriate option is dependent on number of factors such as:
 - Disjointness and participation constraints on the superclass/subclass relationship,
 - Whether subclasses are involved in distinct relationships,
 - Number of participants in superclass/subclass relationship
- **In this course unit; discuss guidelines for the representation of a superclass/subclass relationship based only on the participation and disjoint constraints**

Map Superclass/Subclass Relationships

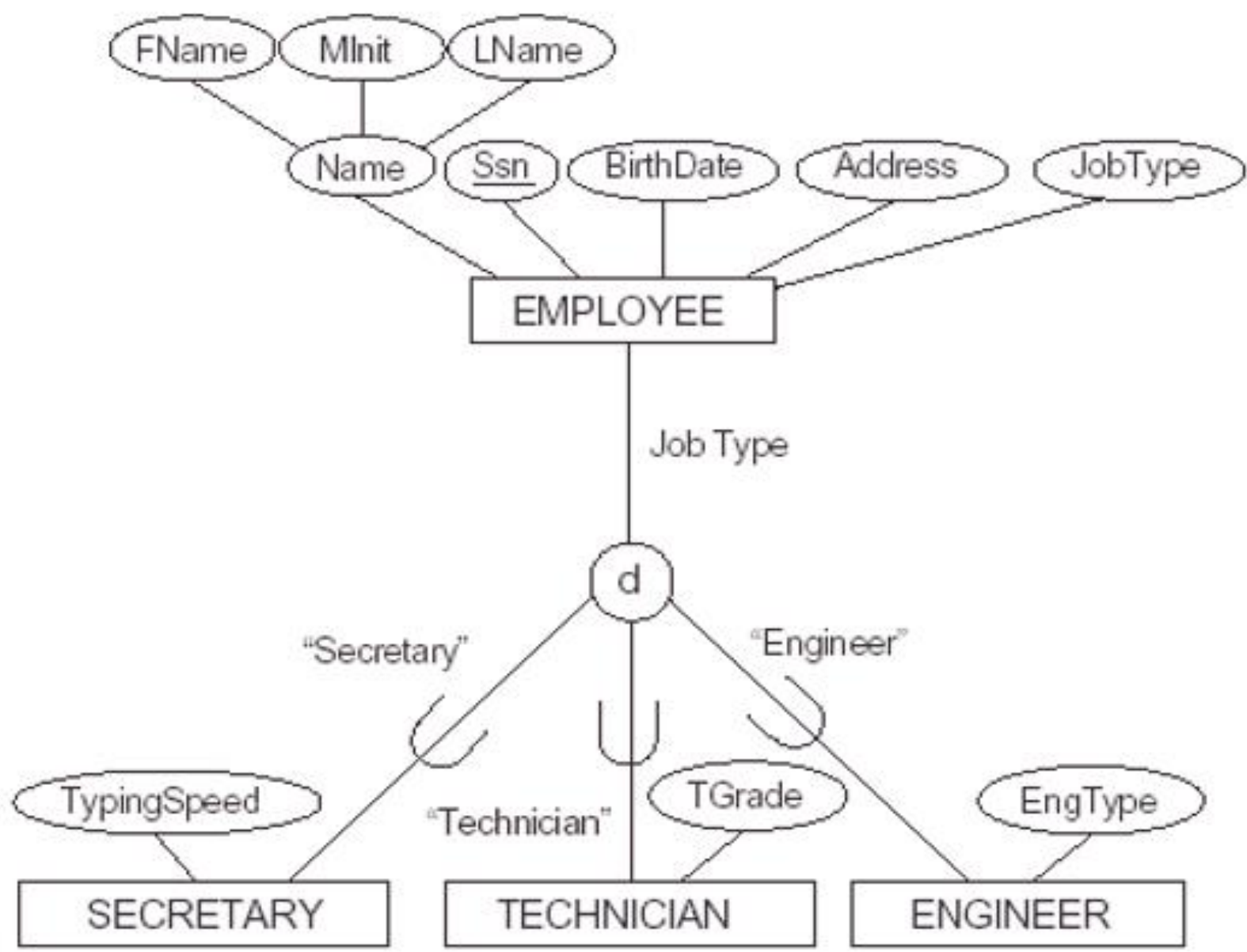
Participation Constraint – Partial

Disjoint Constraint - Disjoint

1. Create a relation for superclass
2. Create a relation for each subclass such that:
 $\{\text{primary_key of superclass}\} \cup \{\text{attributes of subclass}\}$
key for subclass is (primary_key of superclass)



Example - Partial, Disjoint



Partial, Disjoint

Employee

<u>Ssn</u>	FName	MInit	LName	B'date	Address	JobType
------------	-------	-------	-------	--------	---------	---------

Secretary

<u>Ssn</u>	TypingSpeed
------------	-------------

Technician

<u>Ssn</u>	TGrade
------------	--------

Engineer

<u>Ssn</u>	EngType
------------	---------

Map Superclass/Subclass Relationships

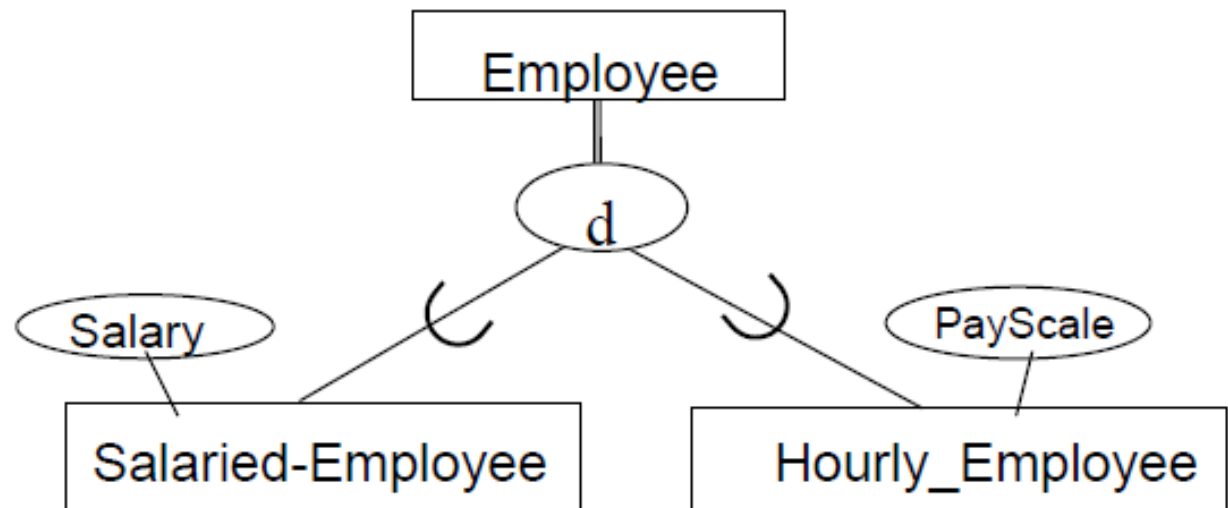
Participation Constraint – Total

Disjoint Constraint - Disjoint

Create a relation for each subclass such that:

$\{\text{primary_key of superclass}\} \cup \{\text{attributes of superclass}\}$
 $\cup \{\text{attributes of subclass}\}$

key for each relation is (primary_key of superclass)



Total, Disjoint

Salaried_Employee

<u>Ssn</u>	FName	MInit	LName	B'date	Address	JobType	Salary
------------	-------	-------	-------	--------	---------	---------	--------

Hourly_Employee

<u>Ssn</u>	FName	MInit	LName	B'date	Addr.	JobType	PayScale
------------	-------	-------	-------	--------	-------	---------	----------

Map Superclass/Subclass Relationships

Participation Constraint – Partial

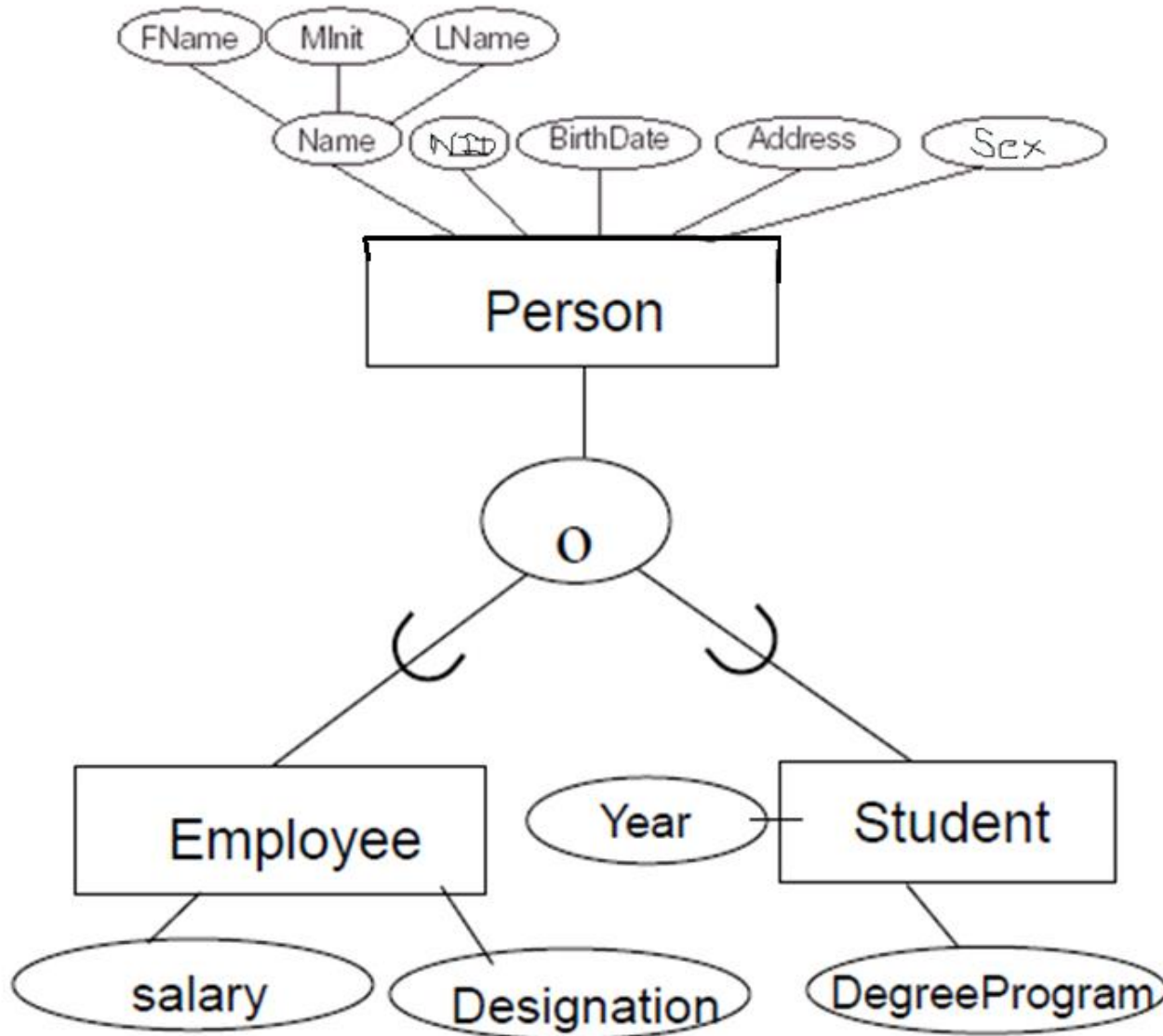
Disjoint Constraint - Overlap

- Create a relation for superclass
- Create a relation for all subclasses such that:

$\{\text{primary_key of superclass}\} \cup \{\text{attributes of all subclasses}\}$
 $\cup \{\text{with one or more discriminators to distinguish the subclass type}\}$

key for each relation is (primary_key of superclass)

Partial, Overlap



Partial, Overlap

Person

<u>NID</u>	FName	MInit	LName	B'date	Sex	Address
------------	-------	-------	-------	--------	-----	---------

Emp_Stud

<u>NID</u>	Salary	Designation	DegreeProgram	Year	Emp.Flag	Student Flag
------------	--------	-------------	---------------	------	----------	-----------------

Map Superclass/Subclass Relationships

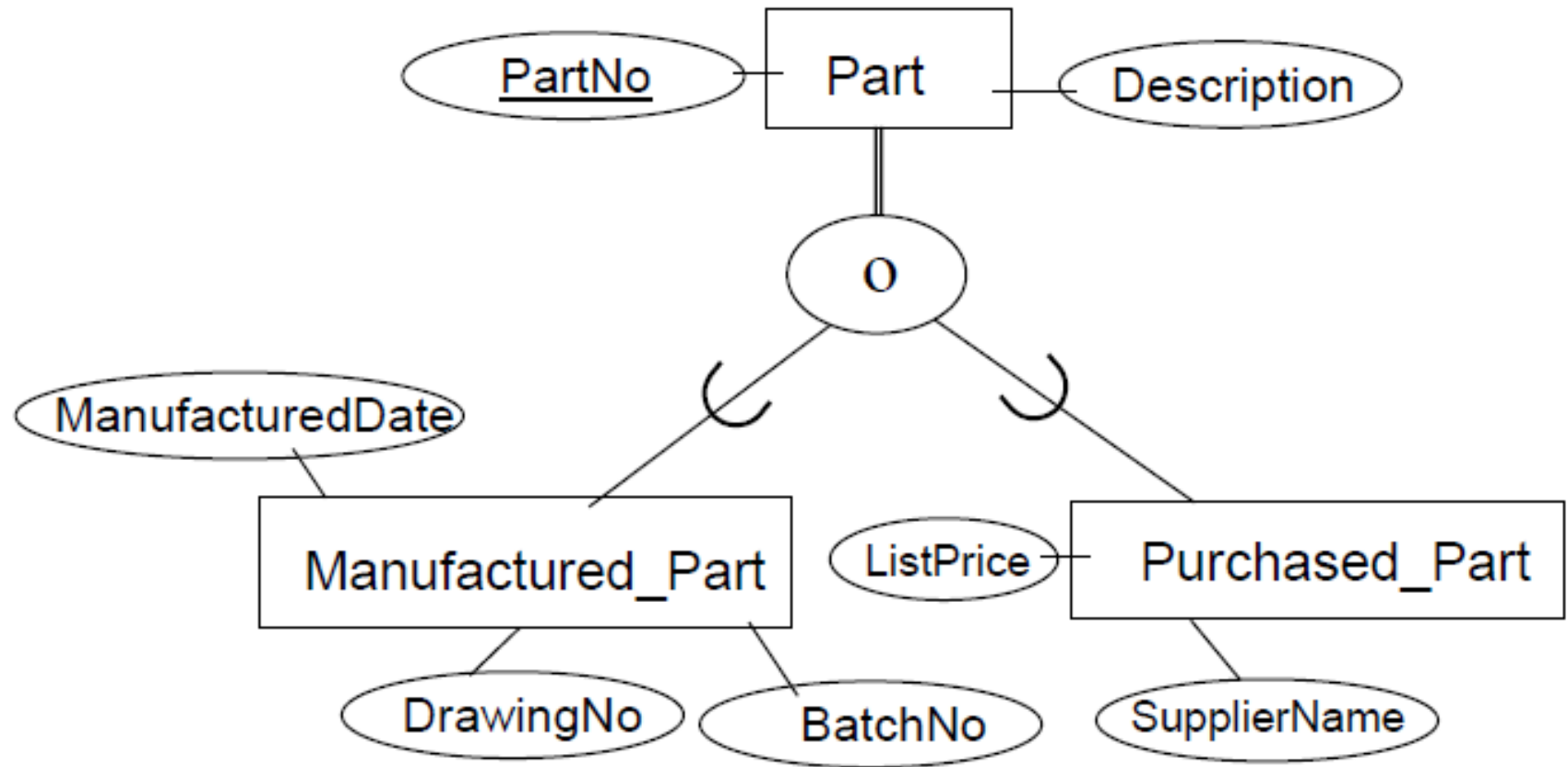
Participation Constraint – Total

Disjoint Constraint - Overlap

Create one relation such that:

$\{\text{primary_key of superclass attributes}\} \cup \{\text{attributes of superclass}\} \cup \{\text{attributes of all subclasses}\} \cup \{\text{type attribute}\}$

Total, Overlap



Total, Overlap

Part

{PartNo,Description,ManufacturedDate,DrawingNo,BatchNo,
SupplierName,ListPrice,ManufacFlag,PurchFlag}

Guidelines for Representation of Superclass/Subclass Relationships

Guidelines for the representation of a superclass/subclass relationship based on the participation and disjoint constraints.

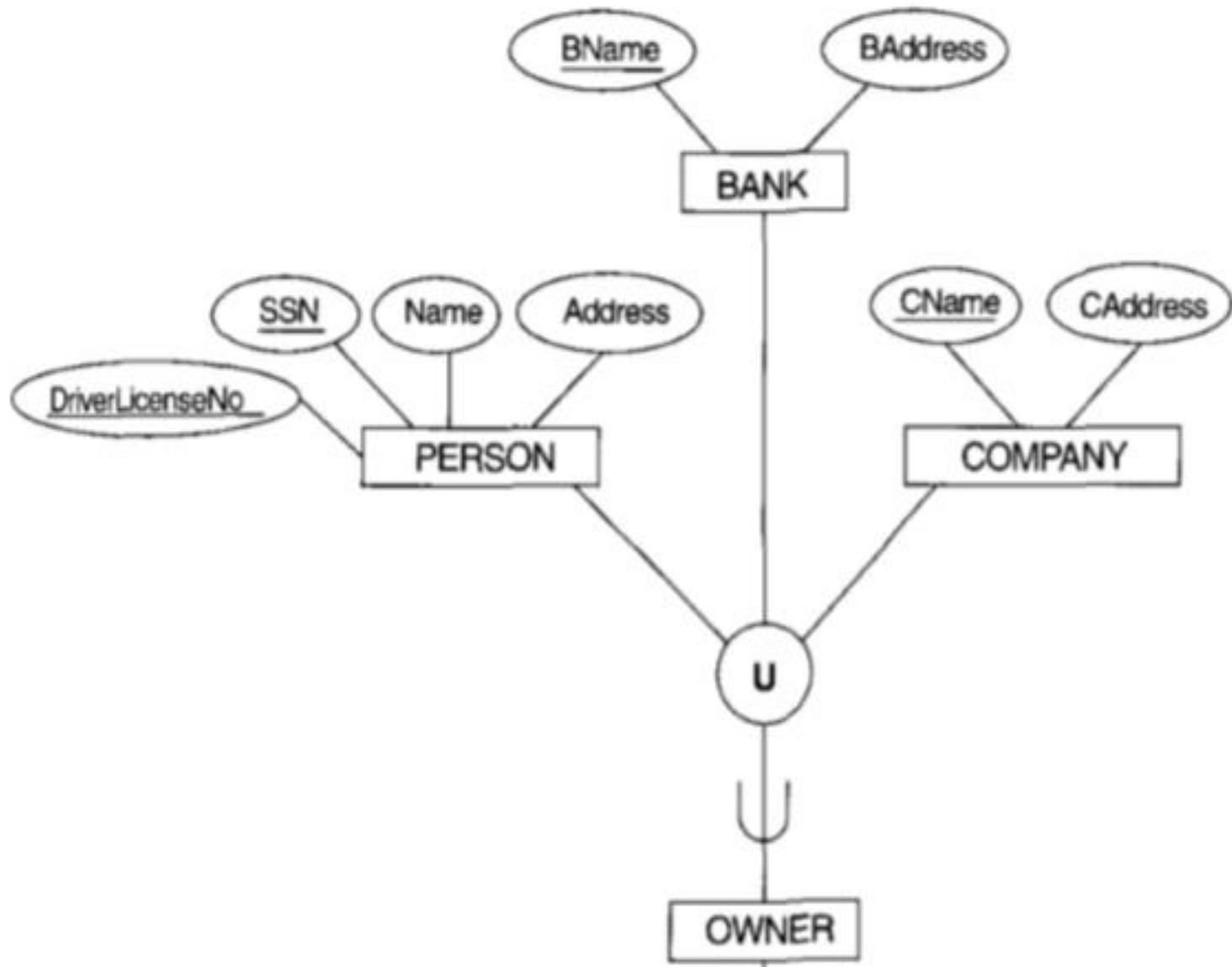
Participation constraint	Disjoint constraint	Relations required
Mandatory	Nondisjoint { And }	Single relation (with one or more discriminators to distinguish the type of each tuple)
Optional	Nondisjoint { And }	Two relations: one relation for superclass and one relation for all subclasses (with one or more discriminators to distinguish the type of each tuple)
Mandatory	Disjoint { Or }	Many relations: one relation for each combined superclass/subclass
Optional	Disjoint { Or }	Many relations: one relation for superclass and one for each subclass

Mandatory(Total); Optional (Partial); Nondisjoint (overlap/AND); Disjoint (OR)

Mapping of Categories

- A **category** is a **subclass** of the **Union** of **two or more superclasses** that can have different keys because they can be of different entity types.
- If the superclasses have different keys we have to define a new key attribute called a “surrogate key”, when *creating a relation* to correspond to the category.
- Create a relation to correspond to the category, include any attributes of the category. The primary key is the surrogate key.
- Each superclass is also mapped into a relation with its own primary key, the surrogate key becomes a foreign key for this.

Mapping of Categories



Mapping of Categories

PERSON

<u>SSN</u>	DriverLicenseNo	Name	Address	OwnerId
------------	-----------------	------	---------	---------

BANK

<u>BName</u>	BAddress	OwnerId
--------------	----------	---------

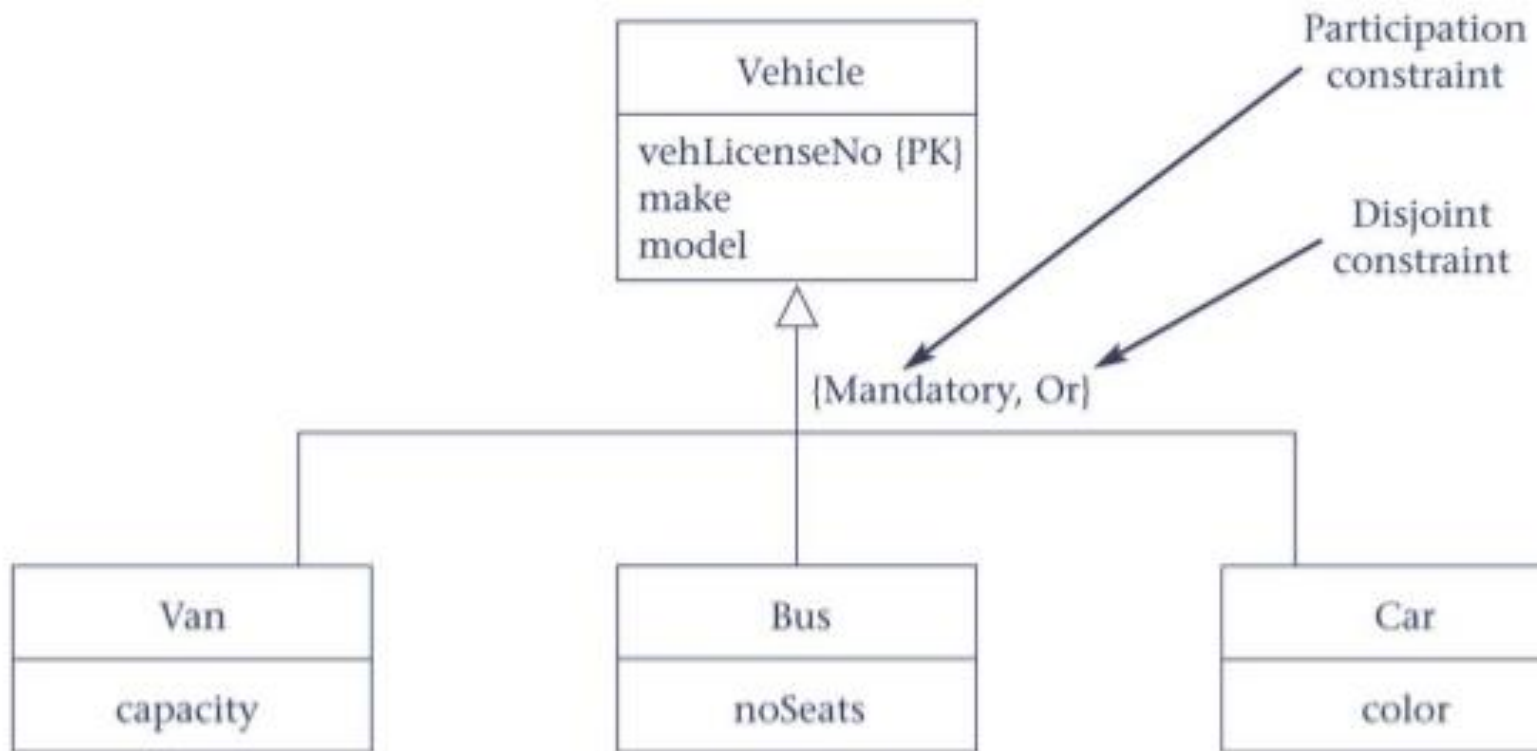
COMPANY

<u>CName</u>	CAddress	OwnerId
--------------	----------	---------

OWNER

<u>OwnerId</u>

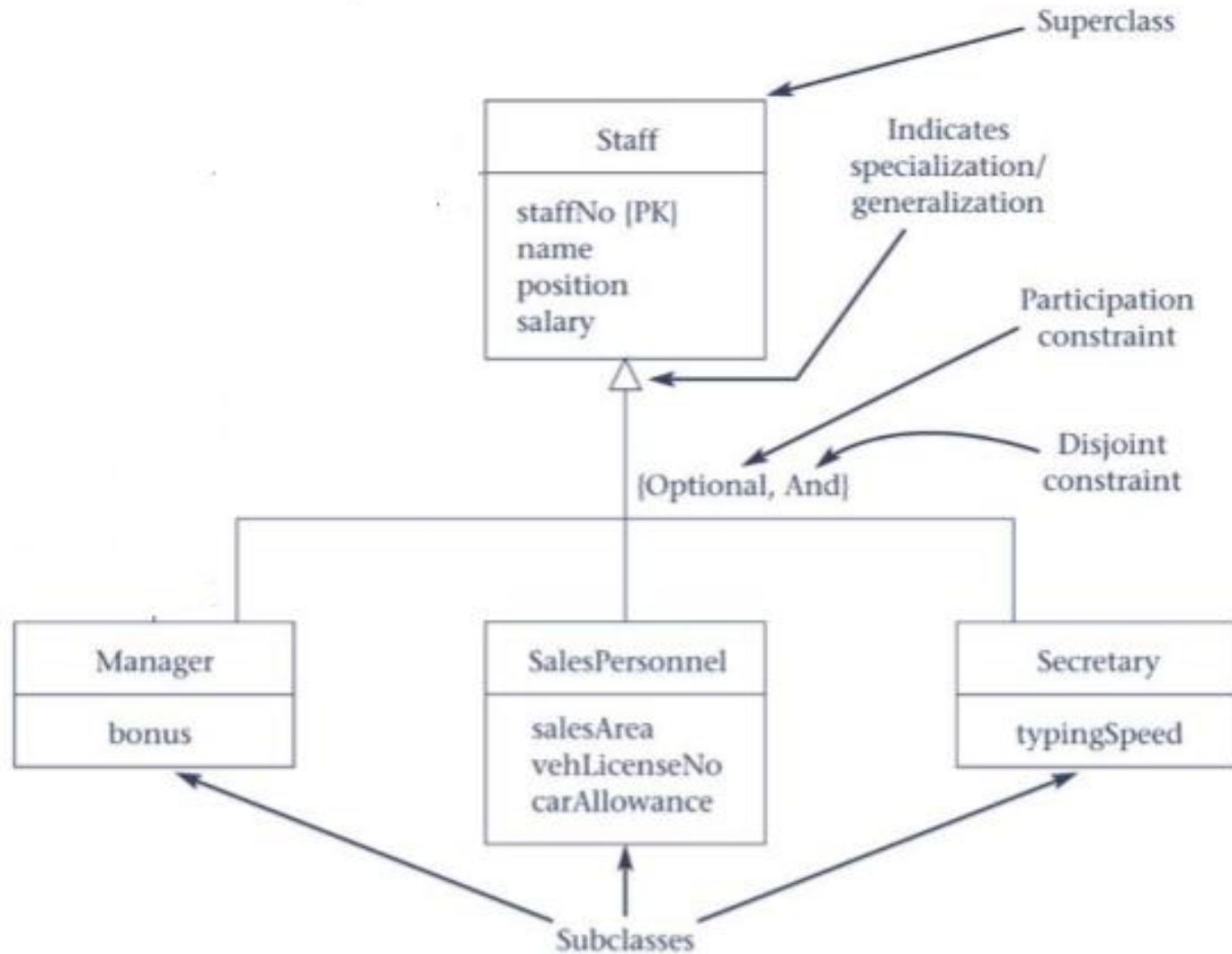
Example



Answer

- Van(vehLicenseNo, make, model, capacity)
- Car (vehLicenseNo, make, model, color)
- Bus (vehLicenseNo, make, model, noSeats)

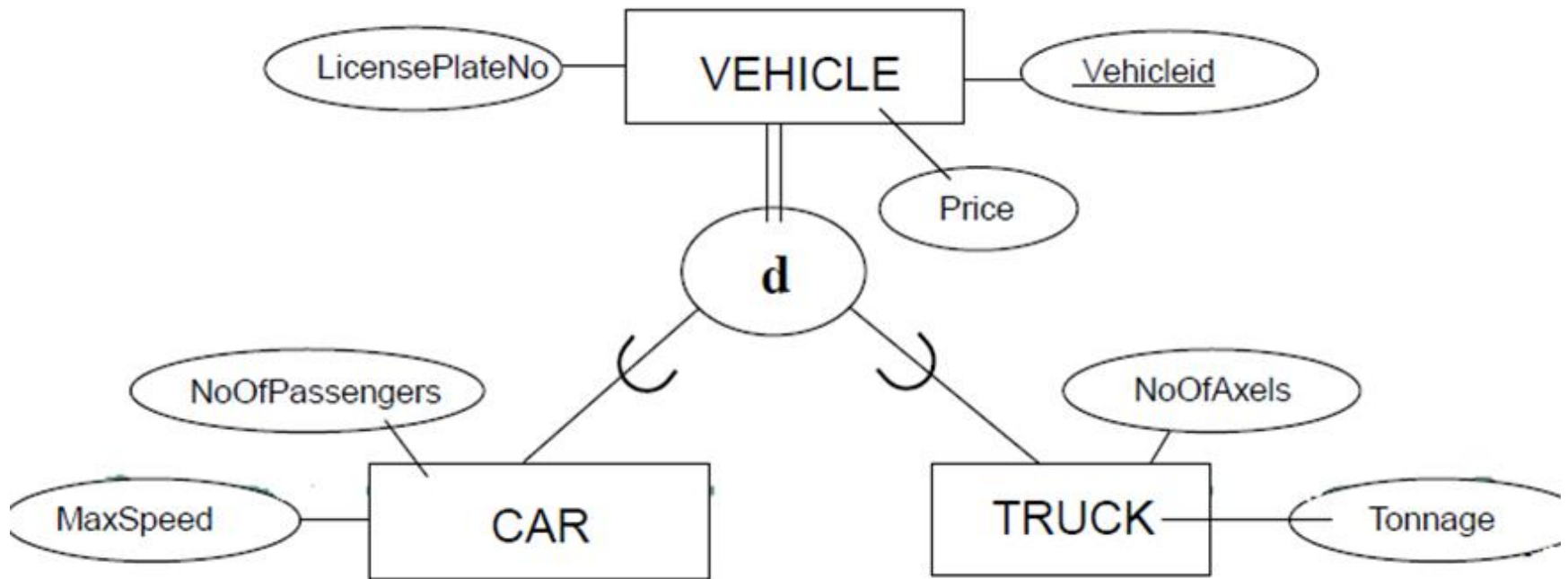
Example



Answer

- Staff (staffNo, name, position, salary)
- AllStaffSubClass(staffNo, bonus, salesArea, vehLicenseNo, carAllowance, typingSpeed, Fag)

Example

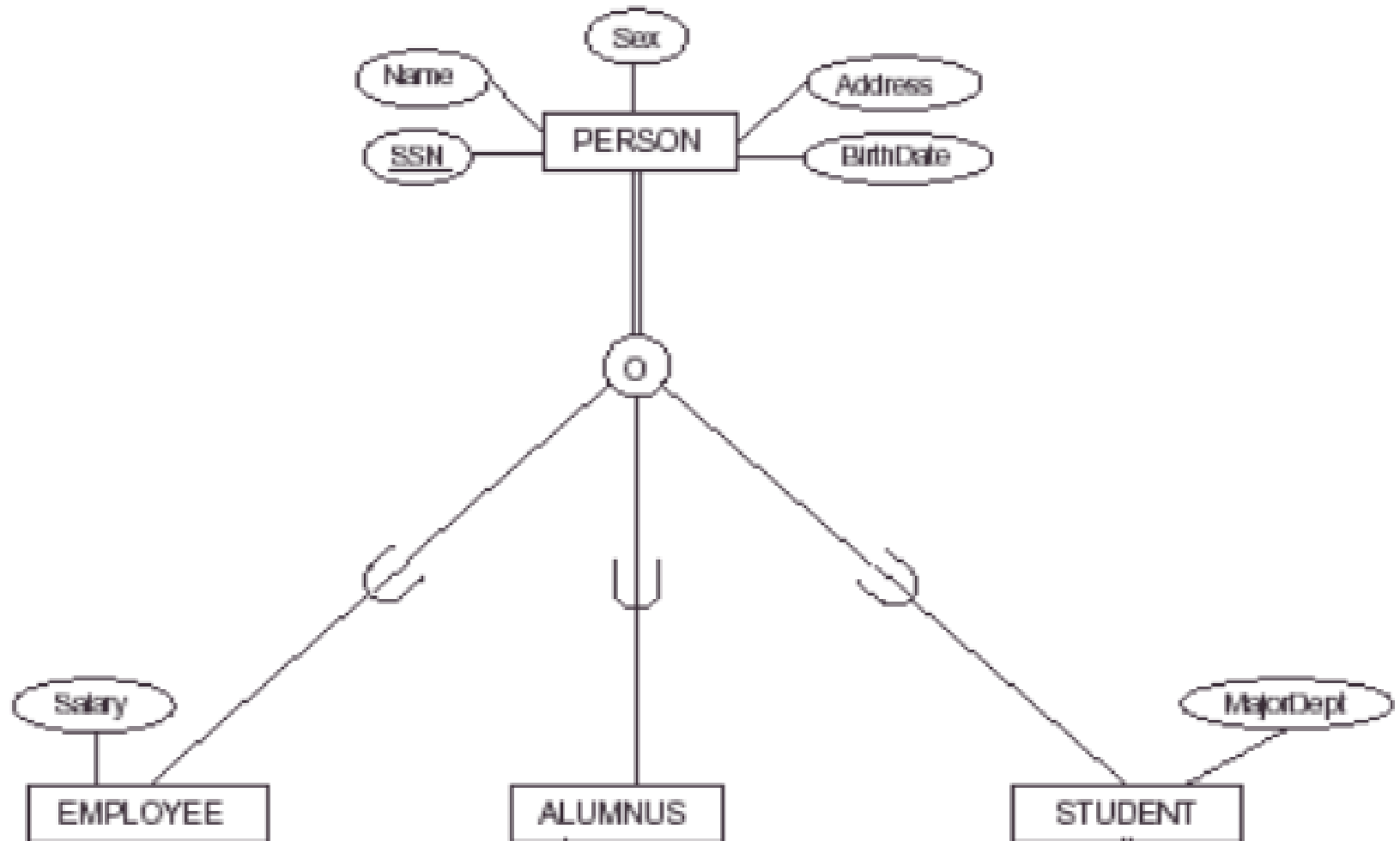


Answer

- CAR (VehicleId, LicensePlateNo, Price, MaxSpeed, NoOfPassengers)
- TRUCK (VehicleId, LicensePlateNo, Price, NoOfAxles, Tonnage)

Example

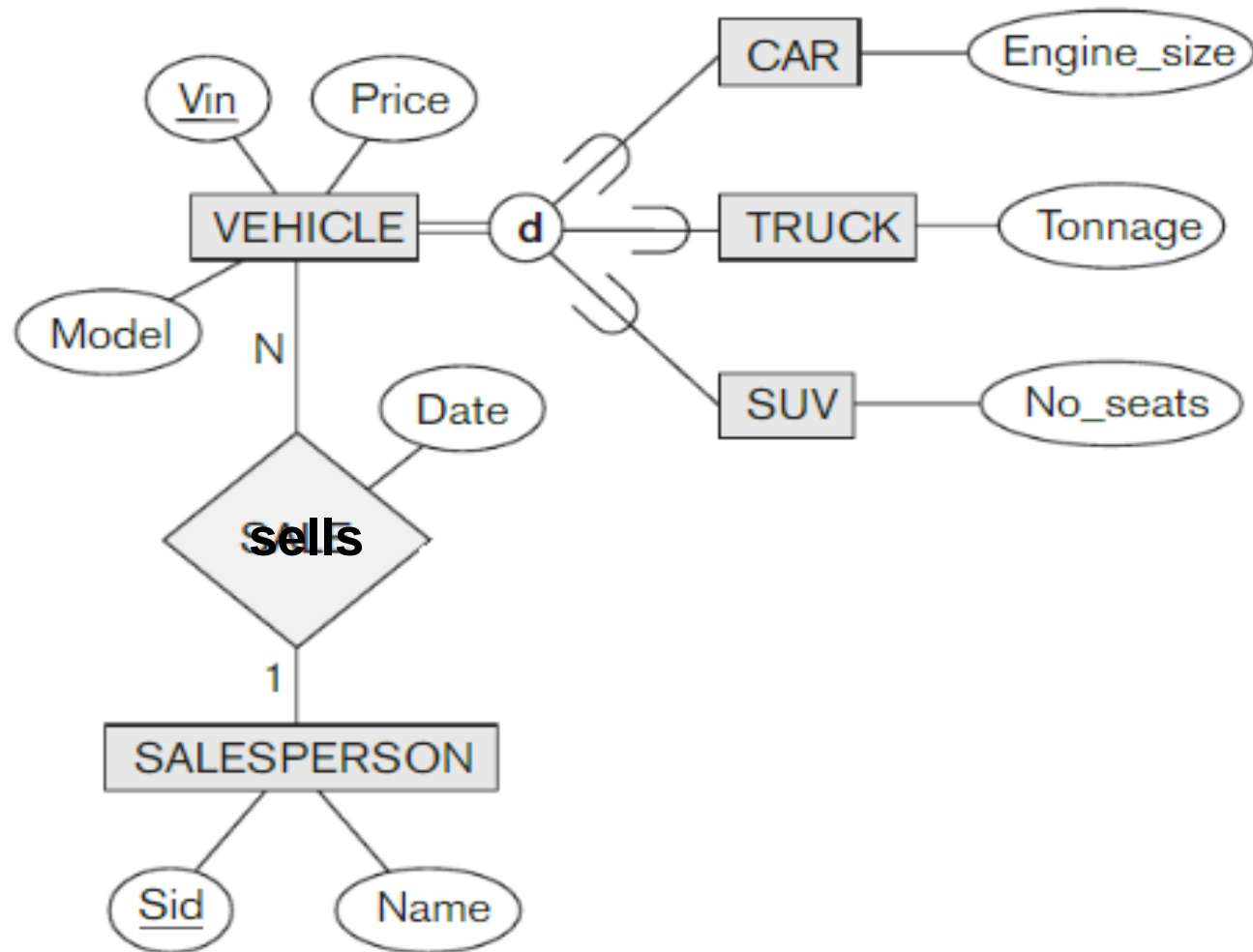
Map the following EER into relations



Answer

- PERSON(SSN, Name, Sex, Address, BirthDate, EFlag, Salary, AFlag, Sflag, MajorDept)

Example - Mapping the EER into relations.



Answer

- SALESPERSON(Sid, Name)
- CAR(Engine_size, Vin, Price, Model, Date, Sid)
- TRUCK(Tonnage, Vin, Price, Model, Date, Sid)
- SUV(No_seats, Vin, Price, Model, Date, Sid)

Homework

Mapping the EER into relations.

