CPSC 6127

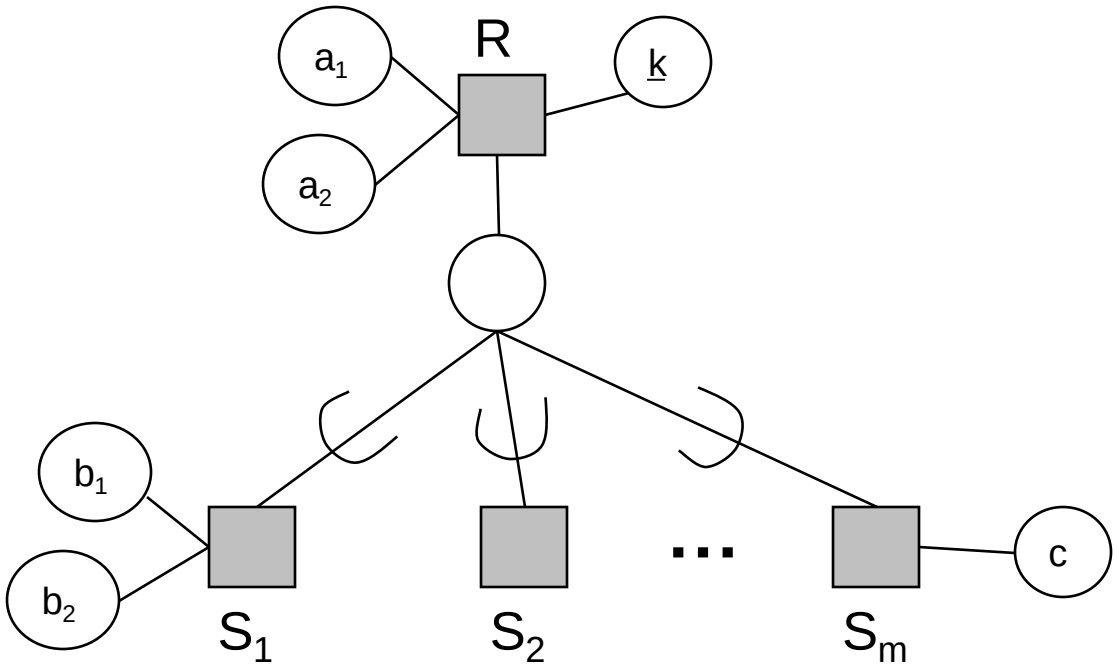
EER to Relational Mapping

EER to Relation Mapping

- Mapping of superclass/subclass relationships
- Mapping of shared subclasses
- Mapping of union types (categories)

Mapping of Superclass/Subclasses

- Let R be the superclass
- $\{S_1, S_2, \ldots, S_m\}$ are the subclasses
  - Each subclass can have its own local attributes



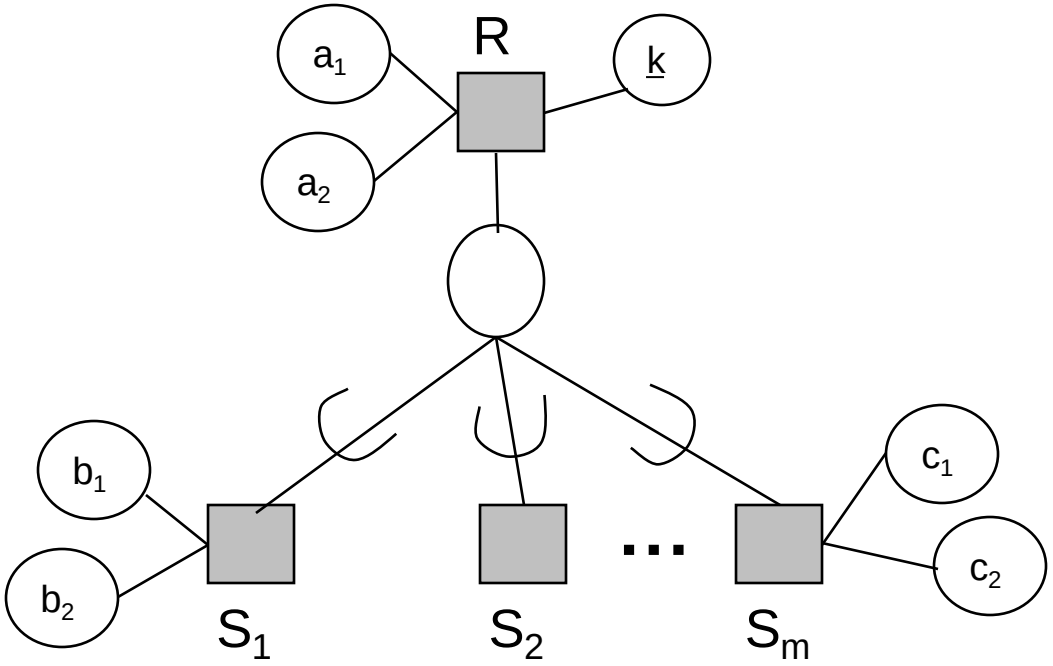
EER to Relation Mapping

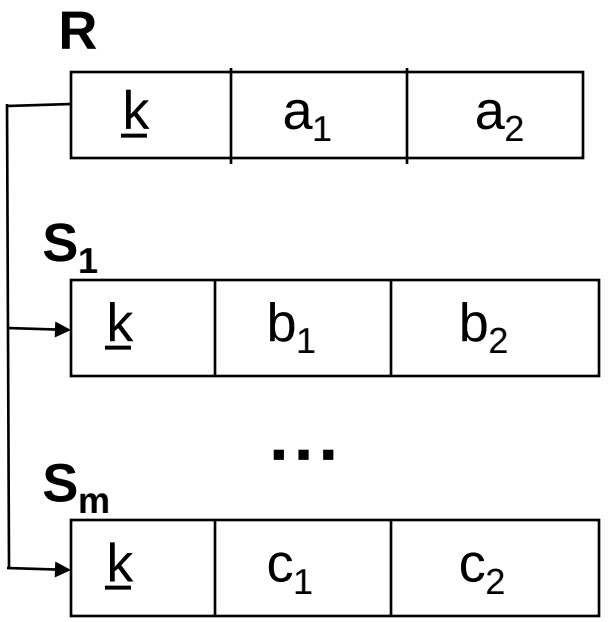
Step 7: 4 possible approaches

- Option 7A: Multiple relations – Superclass and subclasses
- Option 7B: Multiple relations – Subclass relations only
- Option 7C: Single relation with one type attribute
- Option 7D: Single relation with multiple type attributes

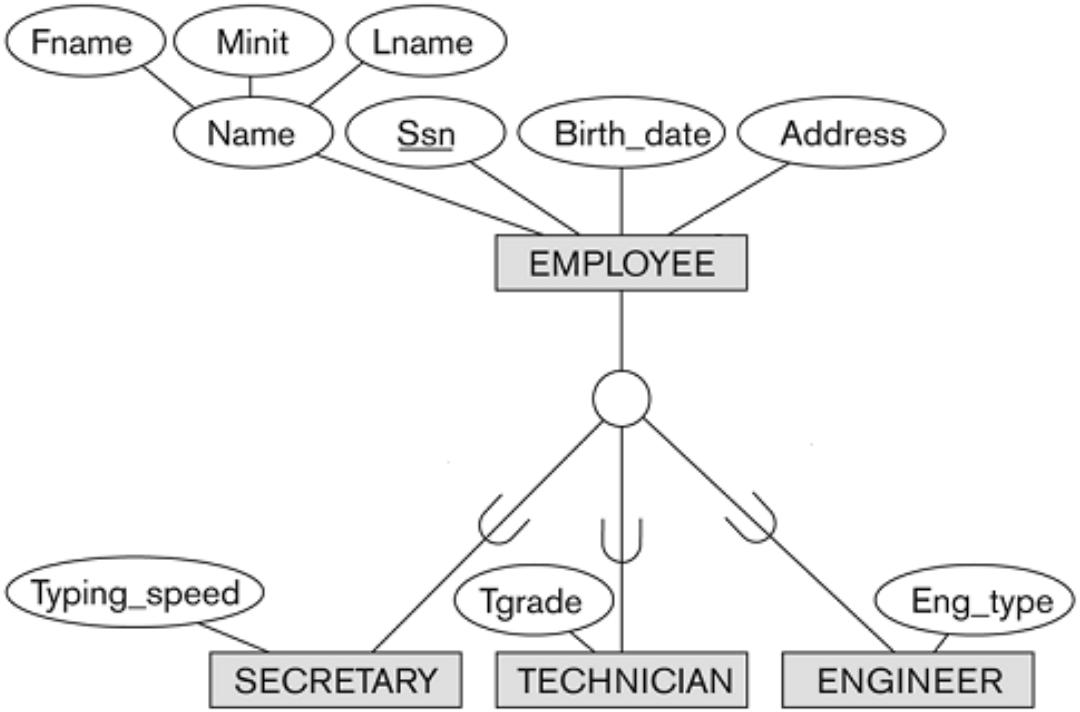
Mapping EER Constructs to Relations

- Option 7A: Multiple relations - superclass and subclass
  - Create superclass relation with attributes {k,a_1,a_2} and primary key (PK) = k
  - Create a relation **S_i** for each subclass S_i, with attributes {k} U {attributes of S_i} with PK = k.
  - This option works for any constraints: disjoint or overlapping; total or partial.





Example



EMPLOYEE

SSN	FName	MInit	LName	BirthDate	Address

SECRETARY

SSN	TypingSpeed

TECHNICIAN

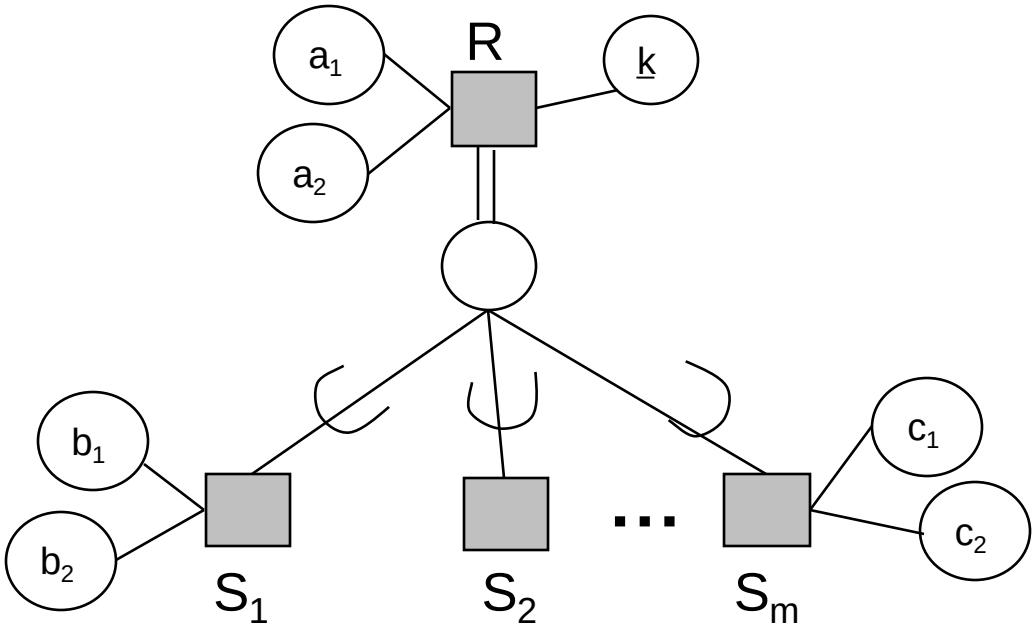
SSN	TGrade

ENGINEER

SSN	EngType

Mapping EER Constructs to Relations

- Option 7B: Multiple relations –subclass relations only
  - Create a relation for each subclass S_i, with the attributes of S_i and the superclass
  - Works for **total specialization** (i.e., every entity in the superclass must belong to (at least) one of the subclasses)
  - If not total, entity not belonging to any sub-class is lost.



S_1

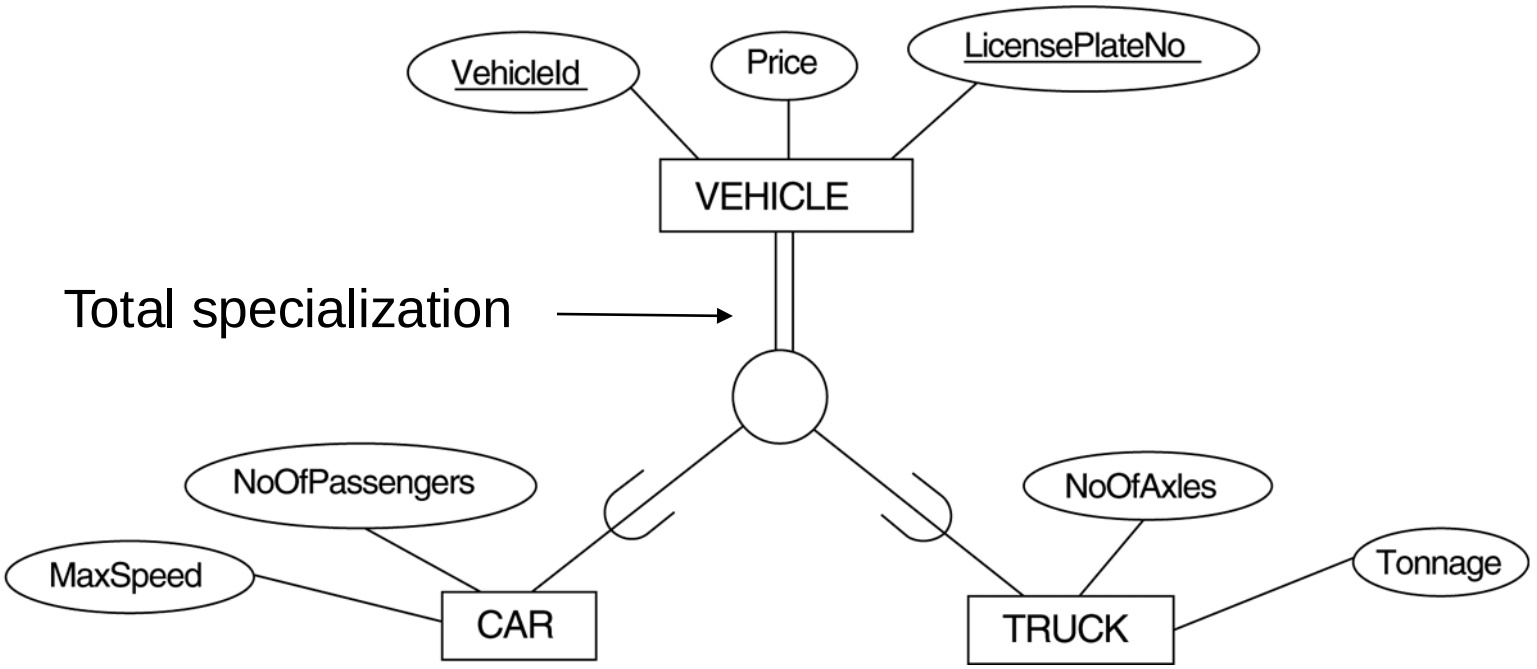
k	b_1	b_2	a_1	a_2

...

S_m

k	c_1	c_2	a_1	a_2

Example



CAR

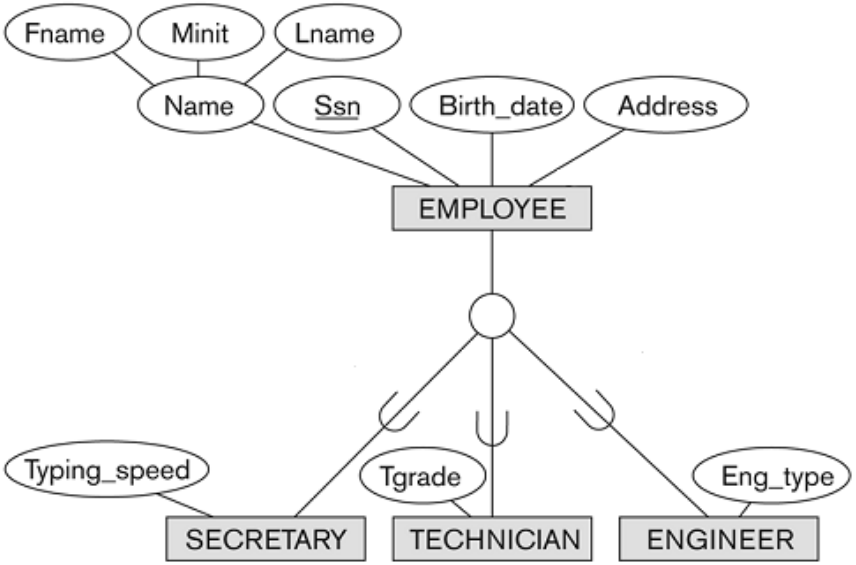
VehicleId	LicensePlateNo	Price	MaxSpeed	NoOfPassengers

TRUCK

VehicleId	LicensePlateNo	Price	NoOfAxles	**Tonnage**

Limitation of previous two approaches

- Query: What does John Doe do as an employee?



To answer this query, we need to scan all three subclass relations, which is inefficient!

EMPLOYEE

SSN	FName	MInit	LName	BirthDate	Address

SECRETARY

SSN	TypingSpeed

TECHNICIAN

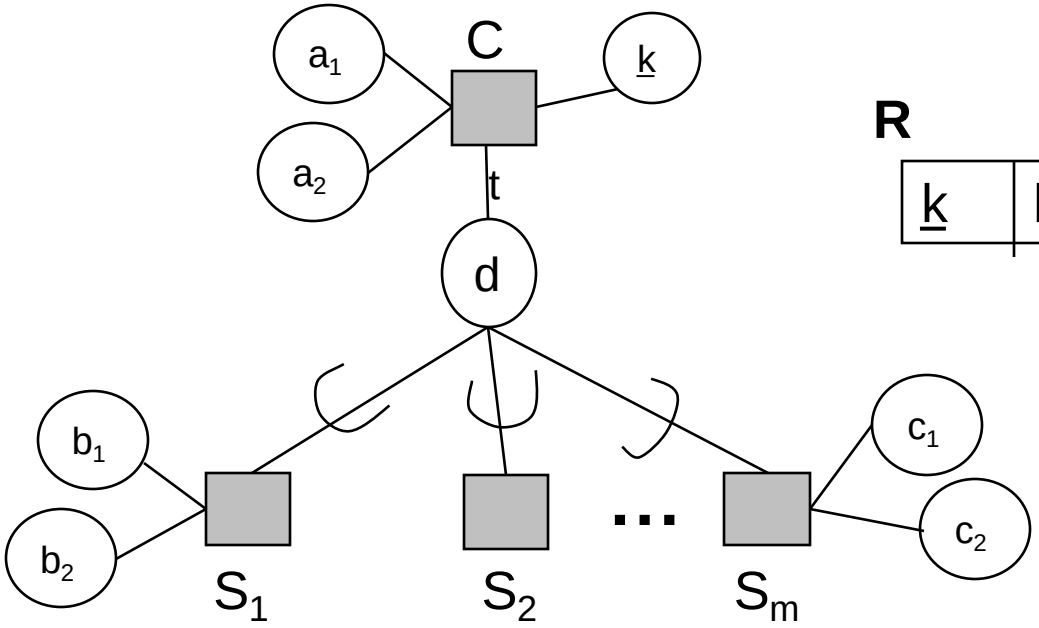
SSN	TGrade

ENGINEER

SSN	EngType

EER to Relations Mapping

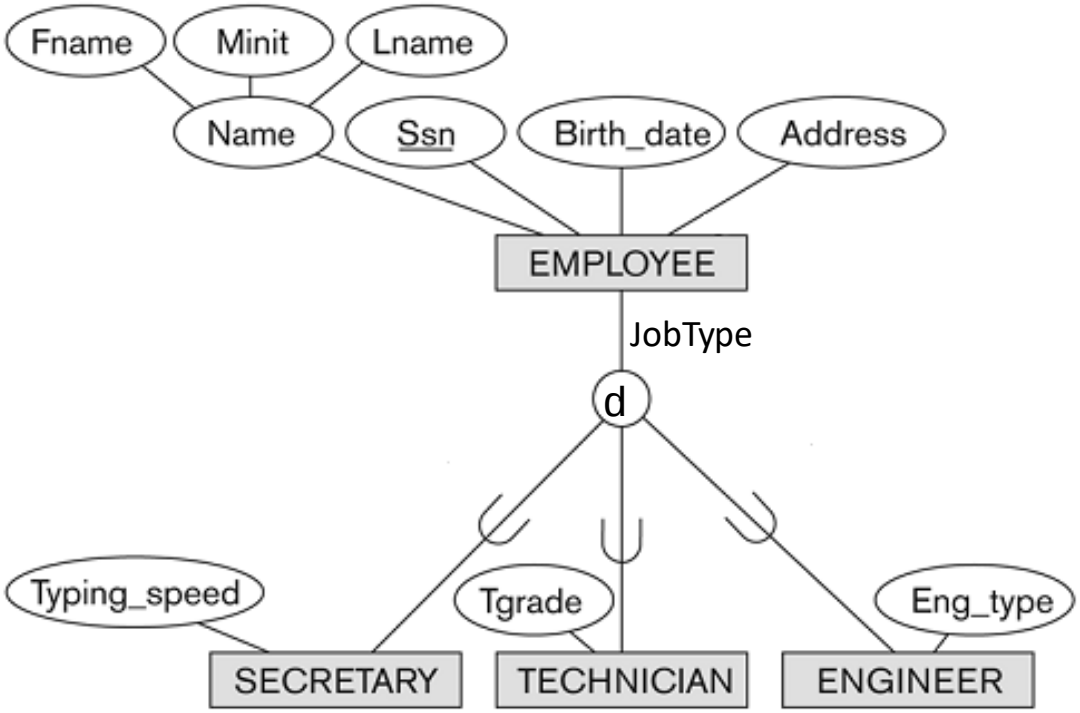
- Option 7C: Single relation – with one type attribute (t)
  - Create a single relation with attributes {k,a_1,…a_n} U {attributes of S_1} U…U {attributes of S_m} U {t} with primary key, PK = k
  - The attribute t is called a type (or discriminating) attribute
  - This option works for **DISJOINT specialization**
  - This option may generate a large number of null values.



R

k	b_1	b_2	c_1	c_2	…	a_1	a_2	t

Example

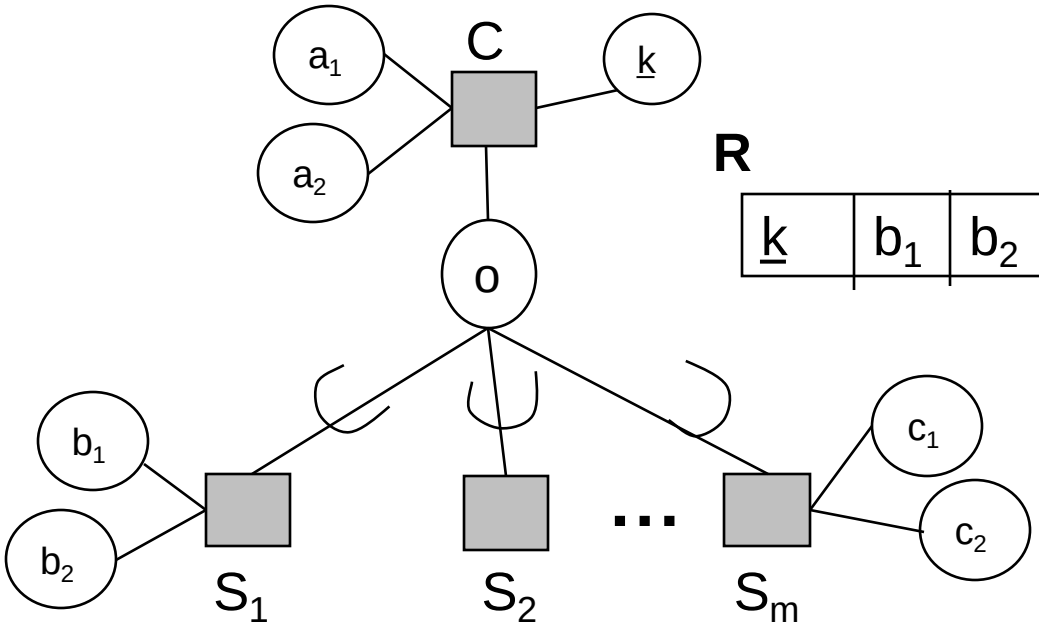


EMPLOYEE

SSN	FName	MInit	LName	BirthDate	Address	JobType	TypingSpeed	TGrade	**EngType**

EER to Relations Mapping

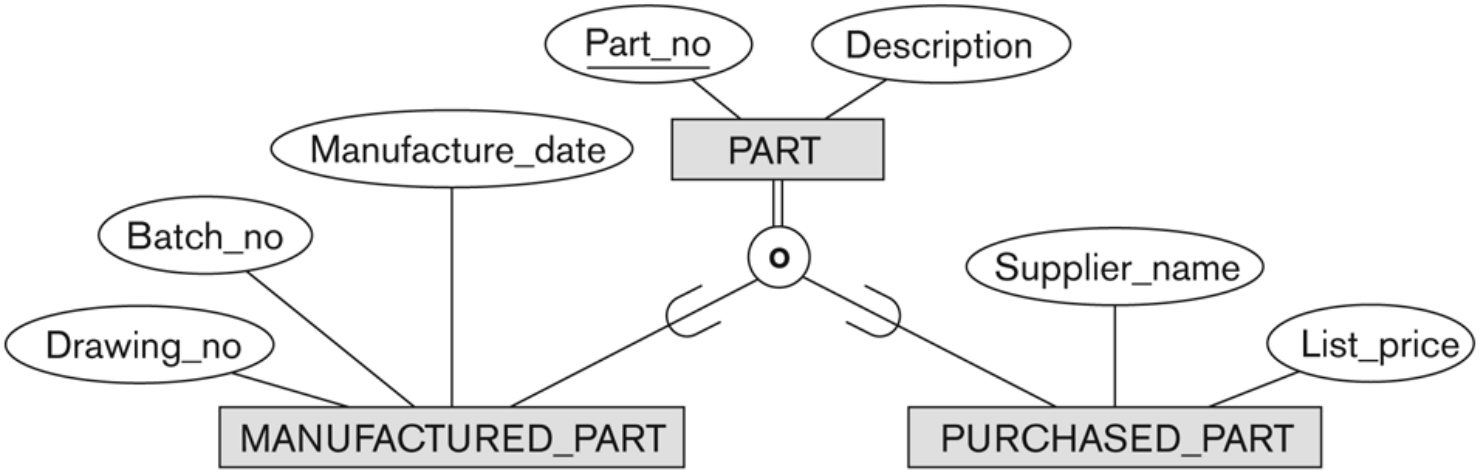
- Option 7D: Single relation – with multiple type attributes
  - Create a single relation with attributes {k,a_1,…a_n} U {attributes of S_1} U…U {attributes of S_m} U {t_1, t_2,…,t_m} and PK = k
  - t_i is a Boolean attribute indicating whether a tuple belongs to S_i.
  - This option works for **overlapping specialization**
  - This option is for specialization whose subclasses are overlapping



R

k	b_1	b_2	c_1	c_2	…	a_1	a_2	t_1	…	t_m

Example



PART

PartNo	Description	MFlag	DrawingNo	ManufactureDate	BatchNo	PFlag	SupplierName	ListPrice

Mapping Union Types/Categories

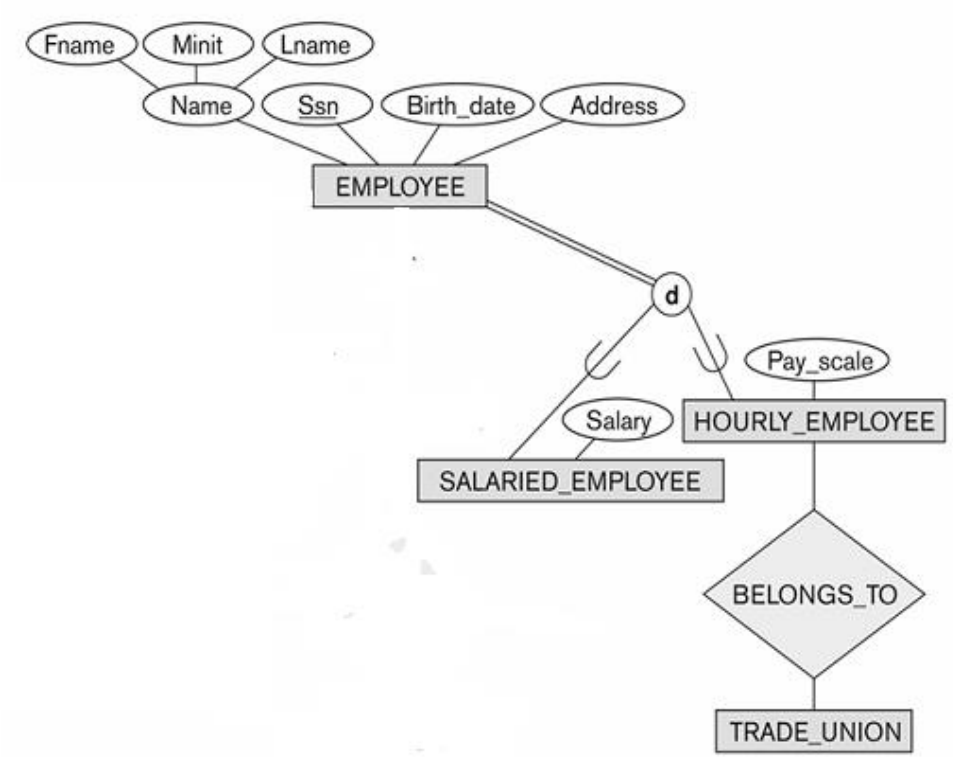
Steps:

For mapping a category whose defining superclass have different keys, it is customary to specify a new key attribute, called a surrogate key, when creating a relation to correspond to the category.

Let C_1, C_2, ..., C_m be the entity types participating in the union and S be the union type. Create a relation for S and *surrogate key* k_s, so that PK(S)=k_s, and also add k_s to each Attr(C_i) as a foreign key into S. If all the C_is have the same primary key type, use that as PK(S) instead.

So which option is better?

- Depends on applications; must consider tradeoff:
  - Too many relations (options 7A and 7B)
    - Inefficient query processing
  - Single relation (options 7C and 7D)
    - Lots of nulls; may “lose” some meaningful relationships

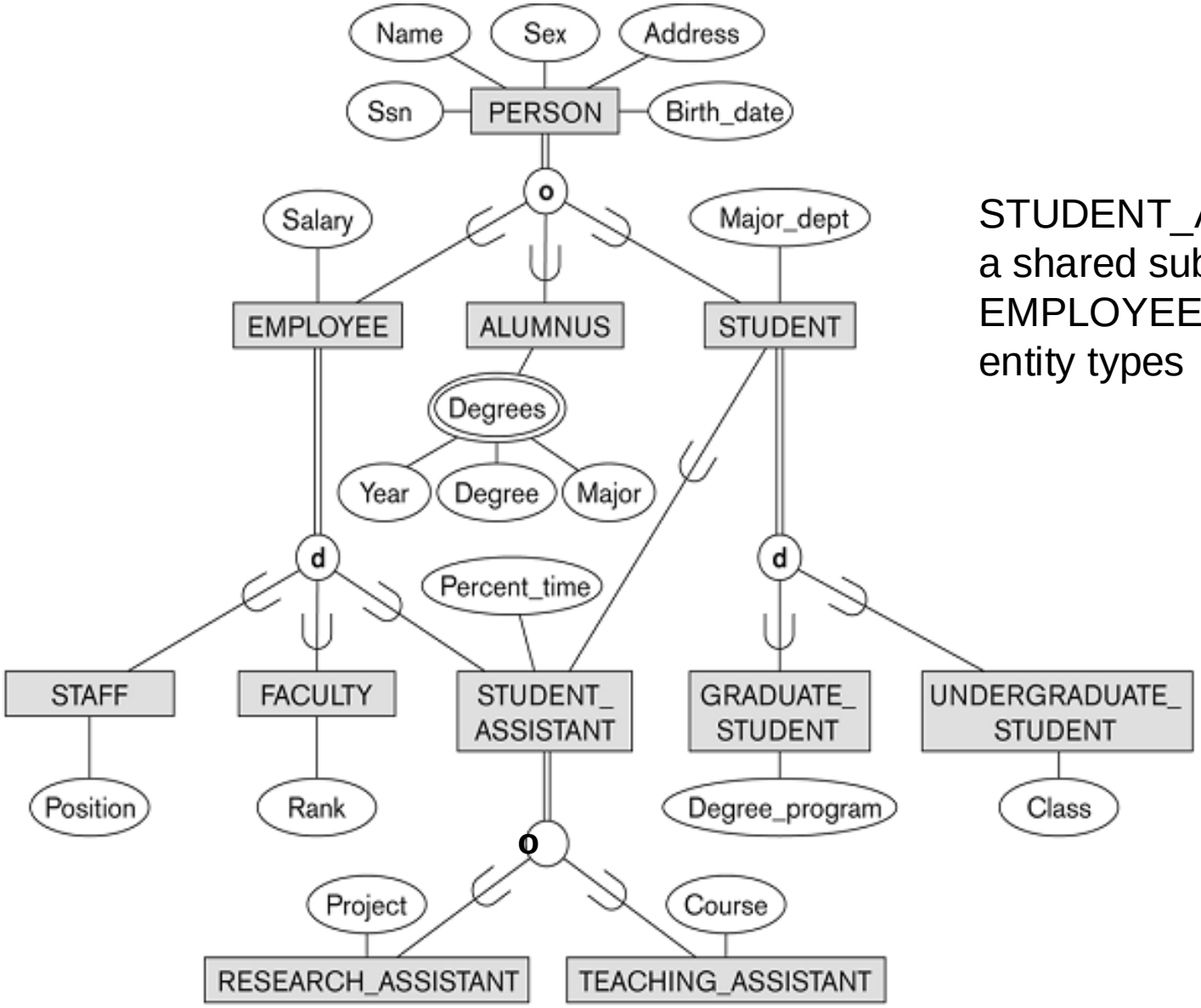


If everything is mapped to the Employee relation, we lose the relationship between hourly employee and trade union

EER to Relations Mapping

- Mapping of Shared Subclasses (Multiple Inheritance)
  - A shared subclass is a subclass of several superclasses, indicating multiple inheritance.
    - These superclasses must have the same key attribute
  - Can apply any of the four options (subject to their restrictions – total/partial, overlapping/disjoint)

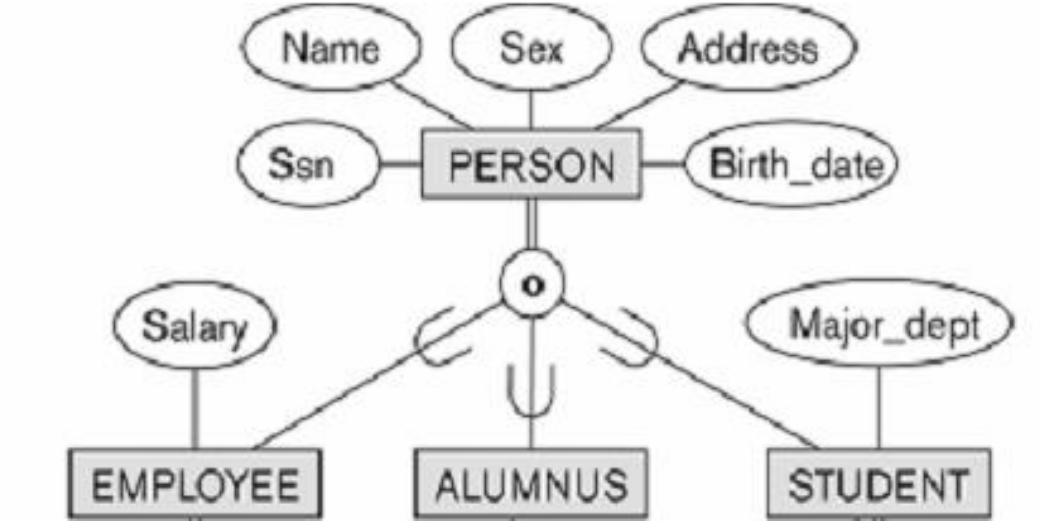
Example



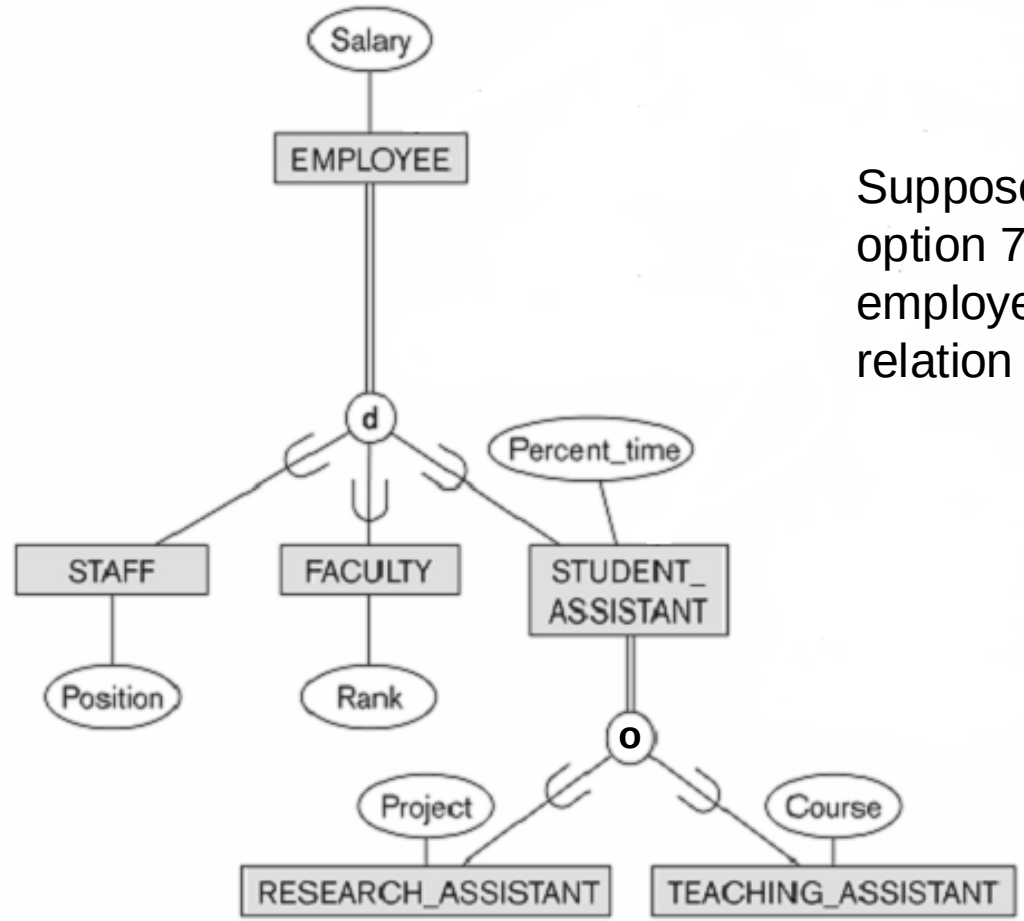
STUDENT_ASSISTANT is a shared subclass of the EMPLOYEE and STUDENT entity types

Example

- Since there are usually separate queries for employees, alumni, and students, we can use options 7A or 7B
  - Relations for option 7A: Person, Employee, Alumnus, Student
  - Relations for option 7B: Employee, Alumnus, Student



Example

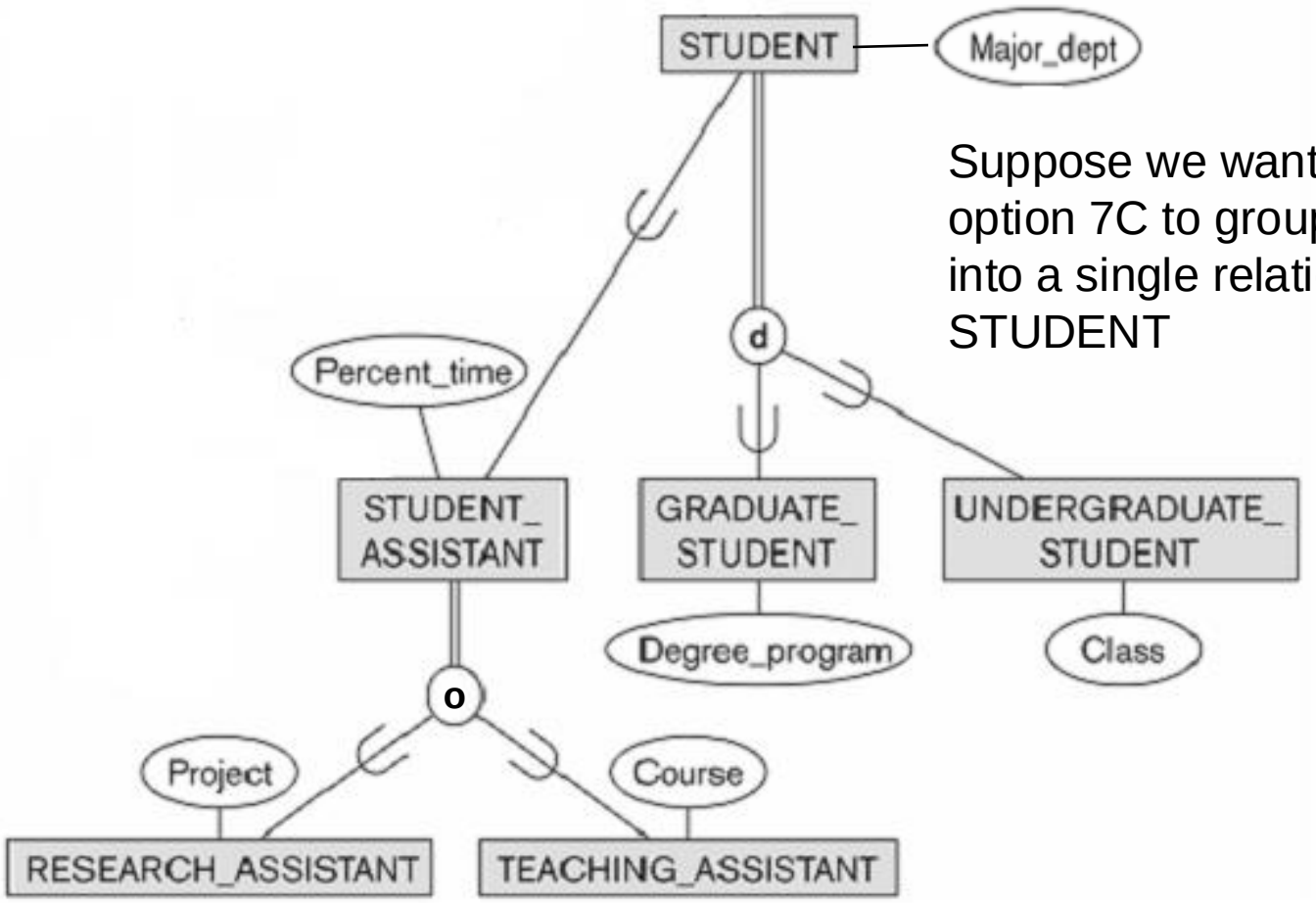


Suppose we want to use option 7C and 7D to group all employees into a single relation called EMPLOYEE

EMPLOYEE

Ssn	Salary	Employee_type	Position	Rank	Percent_time	Ra_flag	Ta_flag	Project	Course

Example

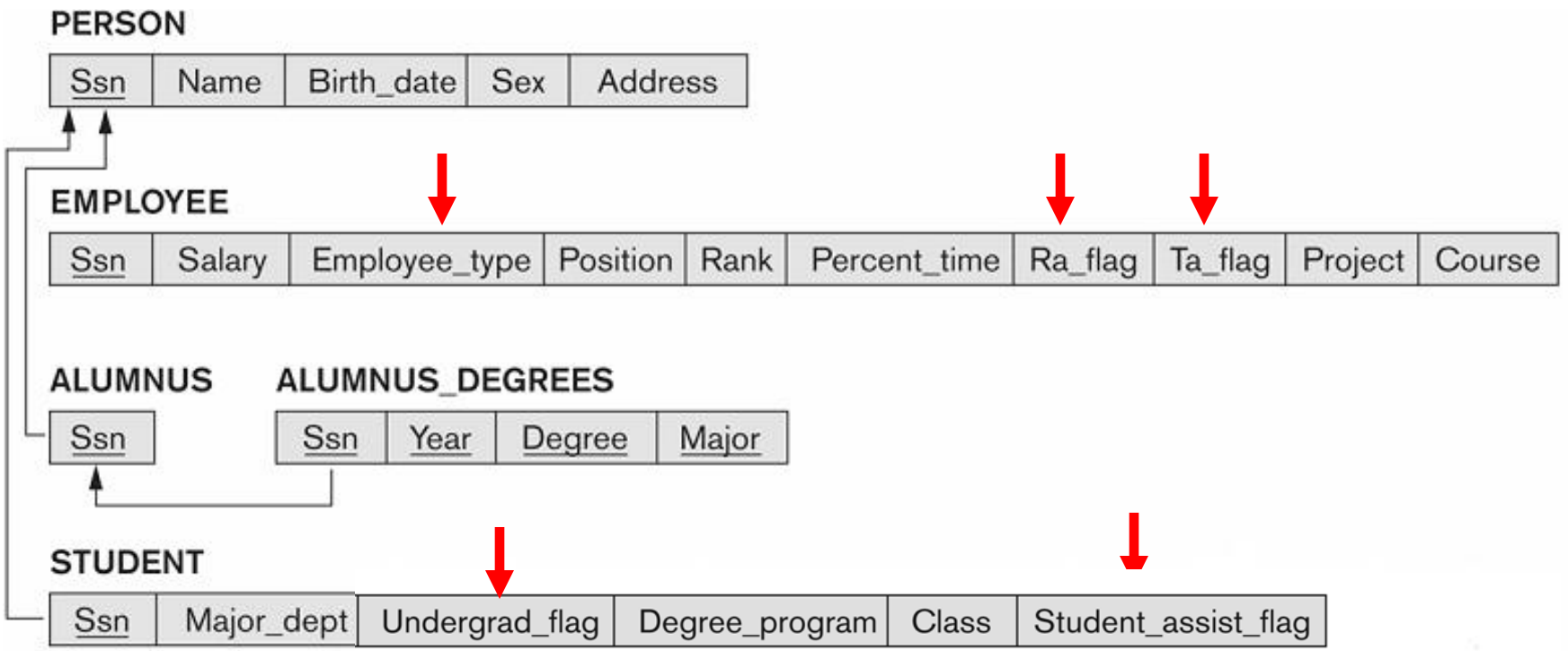


Suppose we want to use option 7C to group all students into a single relation called STUDENT

STUDENT

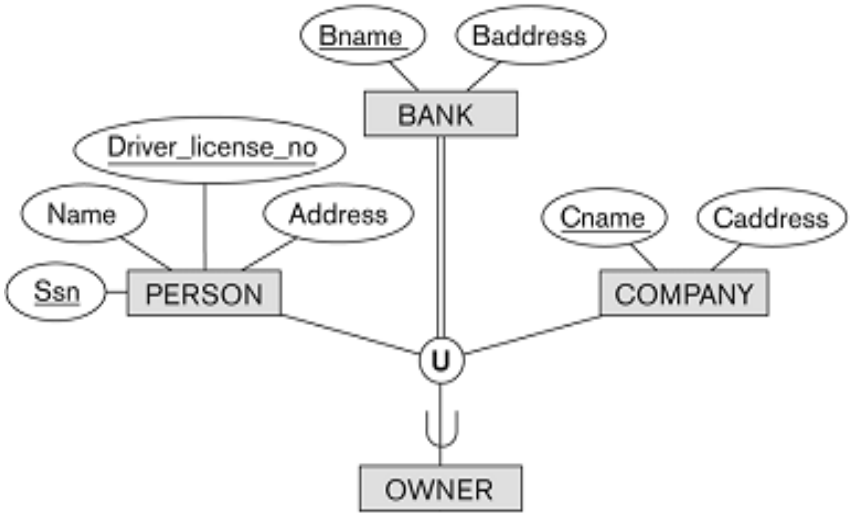
Ssn	Major_dept	Undergrad_flag	Degree_program	Class	Student_assist_flag

Putting it all together…



Union Types (Categories)

- Superclass can have several keys



- Add owner_type

How to deal with OWNER?

PERSON

Ssn	Driver_license_no	Name	Address

BANK

Bname	Baddress

COMPANY

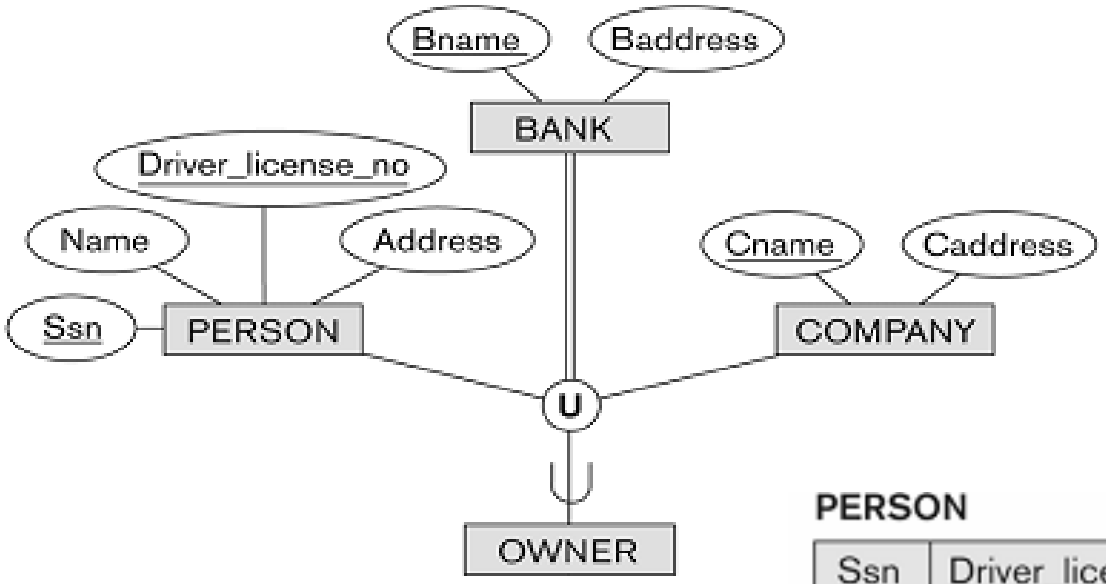
Cname	Caddress

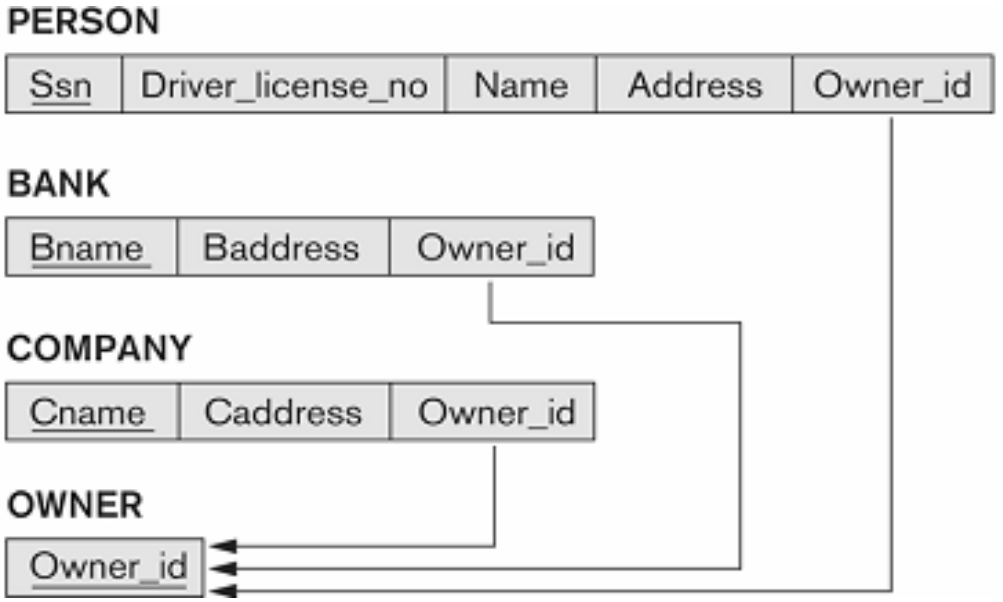
Mapping of Union Types (Categories)

- If all the superclasses have the same key
  - Include the key as an attribute of the category

- Otherwise
  - Create a new key attribute, called a **surrogate key**, as primary key of the category
  - Add surrogate key as foreign key for each superclass relation of the category
  - Add an attribute type to the category identifying particular entity type of the superclasses (PERSON, BANK, COMPANY)

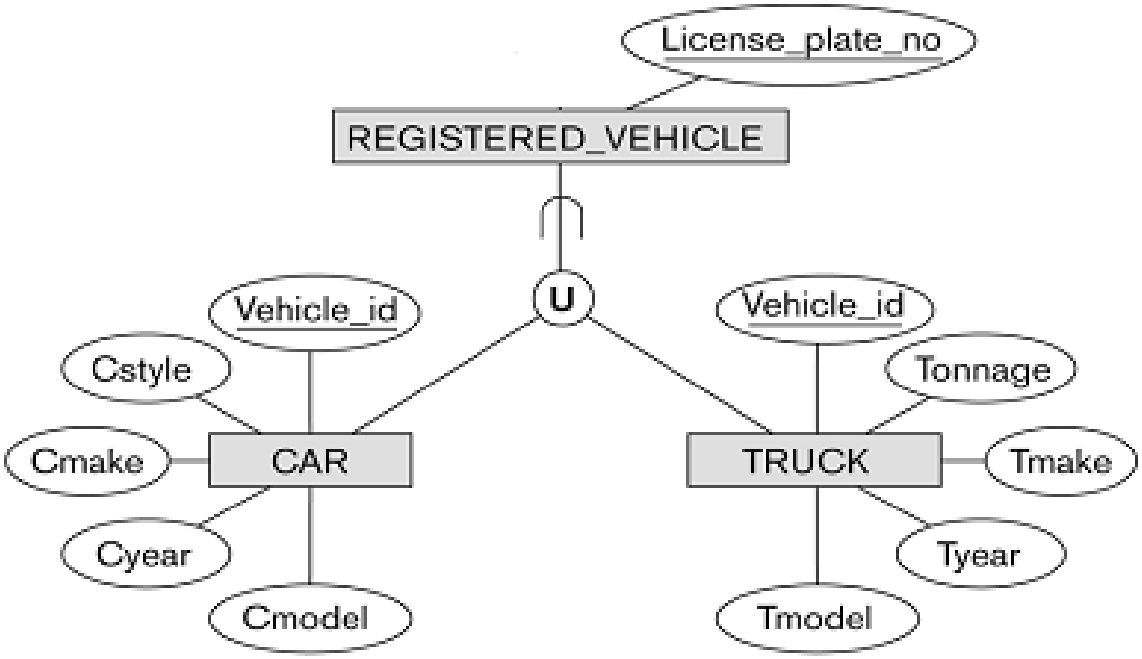
Example (Owner Entity Type)

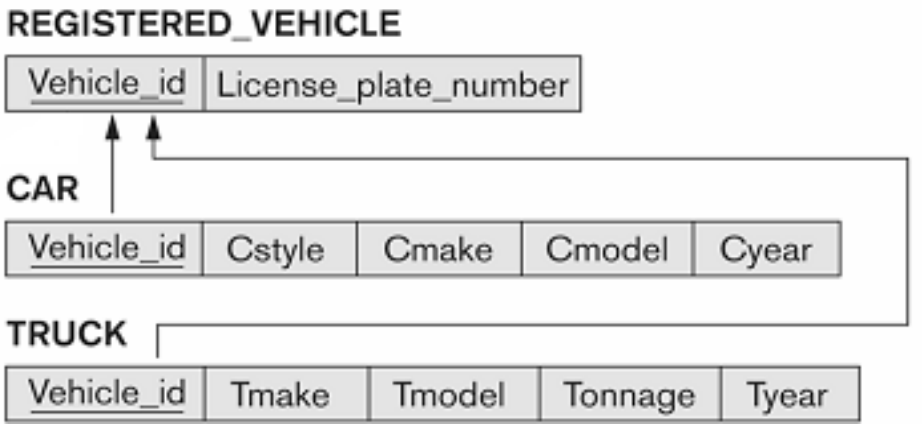




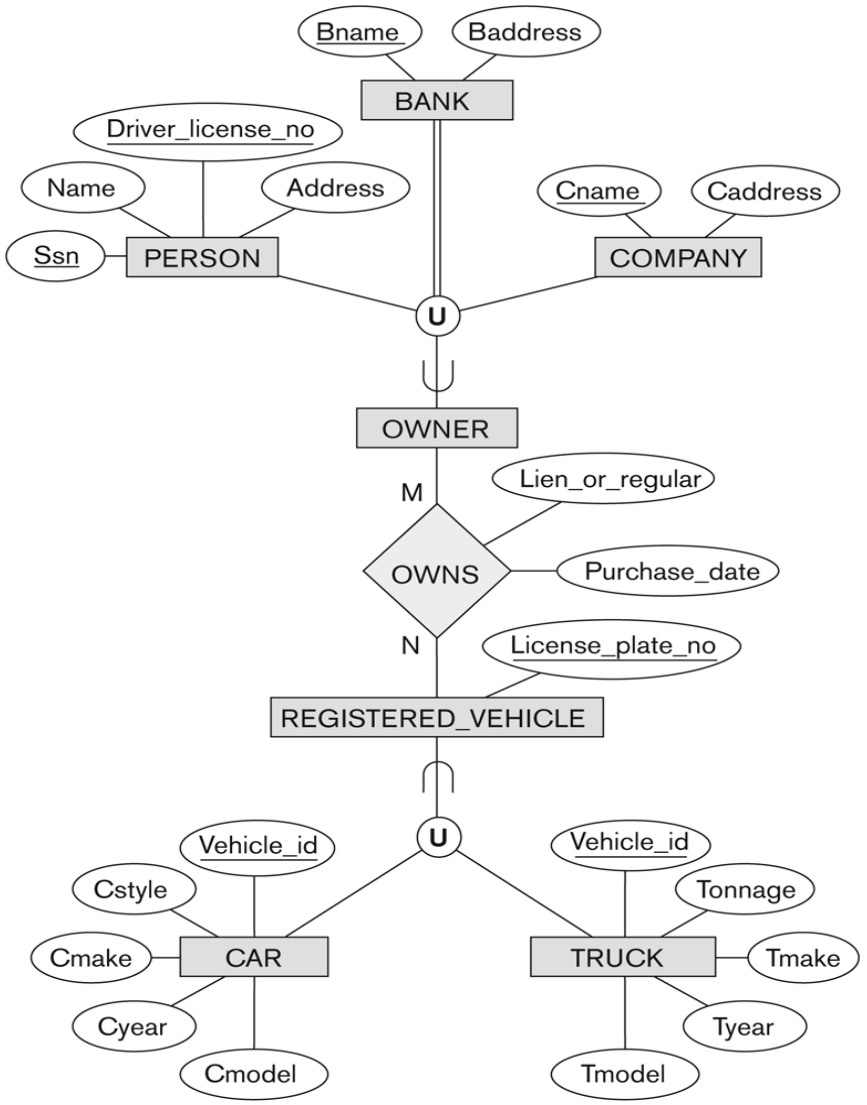
Owner_id is surrogate key

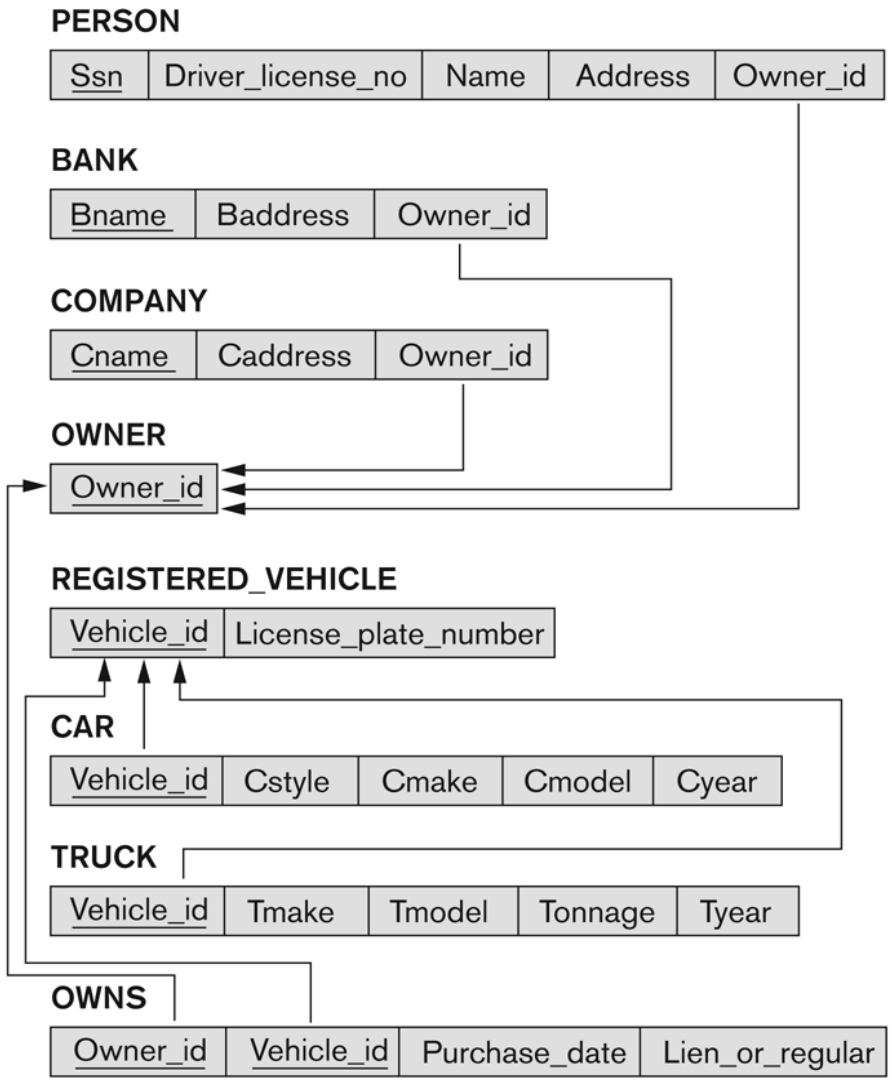
Example (Registered_Vehicle)





Example


Bname
Baddress
BANK
Driver_license_no
Name
Address
Ssn
PERSON
Cname
Caddress
COMPANY
U
OWNER
M
Lien_or_regular
OWNS
Purchase_date
N
License_plate_no
REGISTERED_VEHICLE
U
Vehicle_id
Cstyle
Cmake
CAR
Cyear
Cmodel
Vehicle_id
Tonnage
TRUCK
Tmake
Tyear
Tmodel


PERSON
Ssn
Driver_license_no
Name
Address
Owner_id
BANK
Bname
Baddress
Owner_id
COMPANY
Cname
Caddress
Owner_id
OWNER
Owner_id
REGISTERED_VEHICLE
Vehicle_id
License_plate_number
CAR
Vehicle_id
Cstyle
Cmake
Cmodel
Cyear
TRUCK
Vehicle_id
Tmake
Tmodel
Tonnage
Tyear
OWNS
Owner_id
Vehicle_id
Purchase_date
Lien_or_regular

Exercise: Art Museum Database