Updated 11/17/2022

Topic 2: Database design
L20: Translating EERDs

Wolfgang Gatterbauer

CS3200 Database design (fa22)

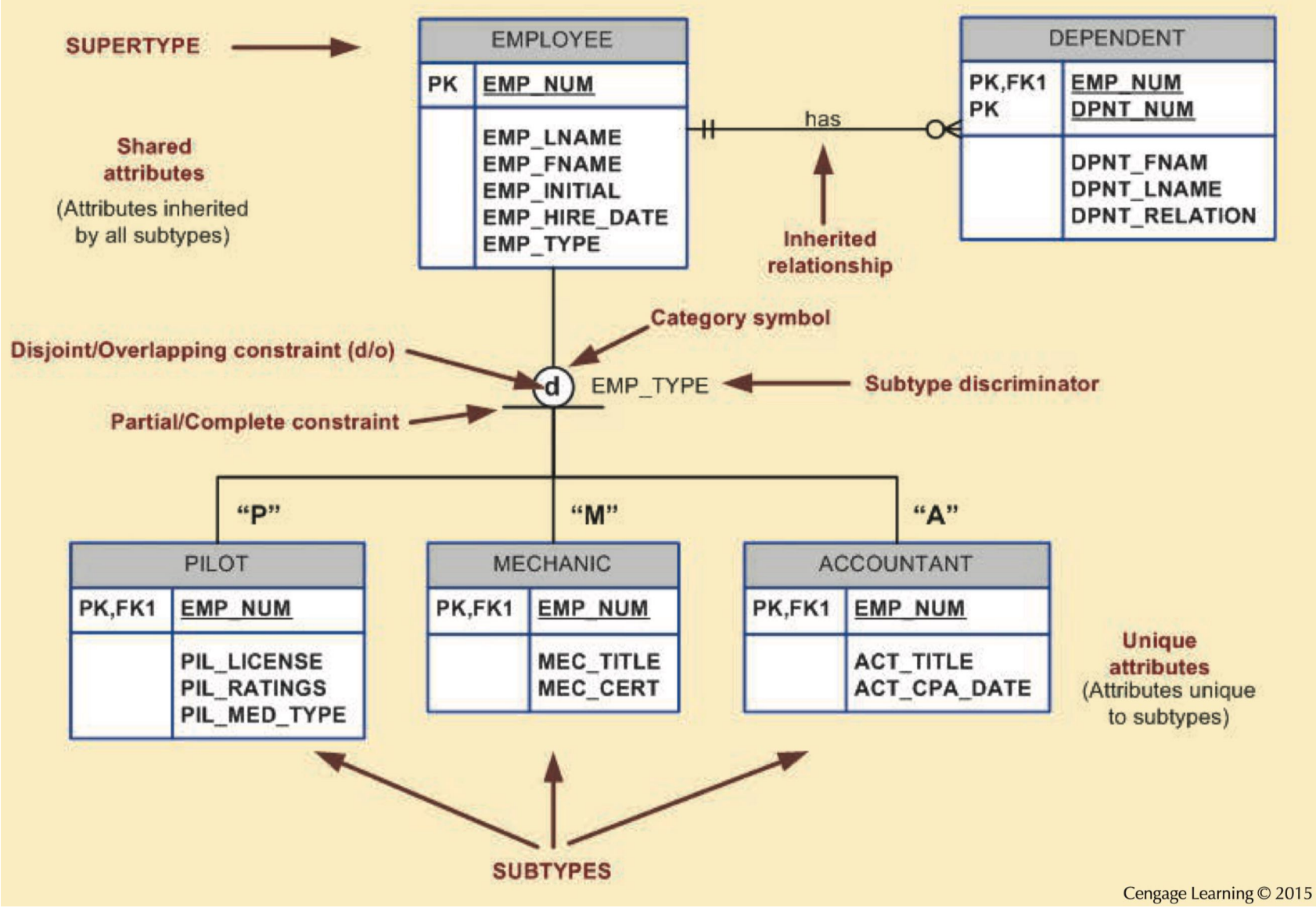
https://northeastern-datalab.github.io/cs3200/fa22s3/

11/16/2022

Class warm-up

- Last class summary
- Exam discussion next Monday. Please schedule make-up exam ASAP.
- I prepared more in-class examples today.
  - Thus we cover less material but practice more.
  - Class calendar will be adjusted as we go

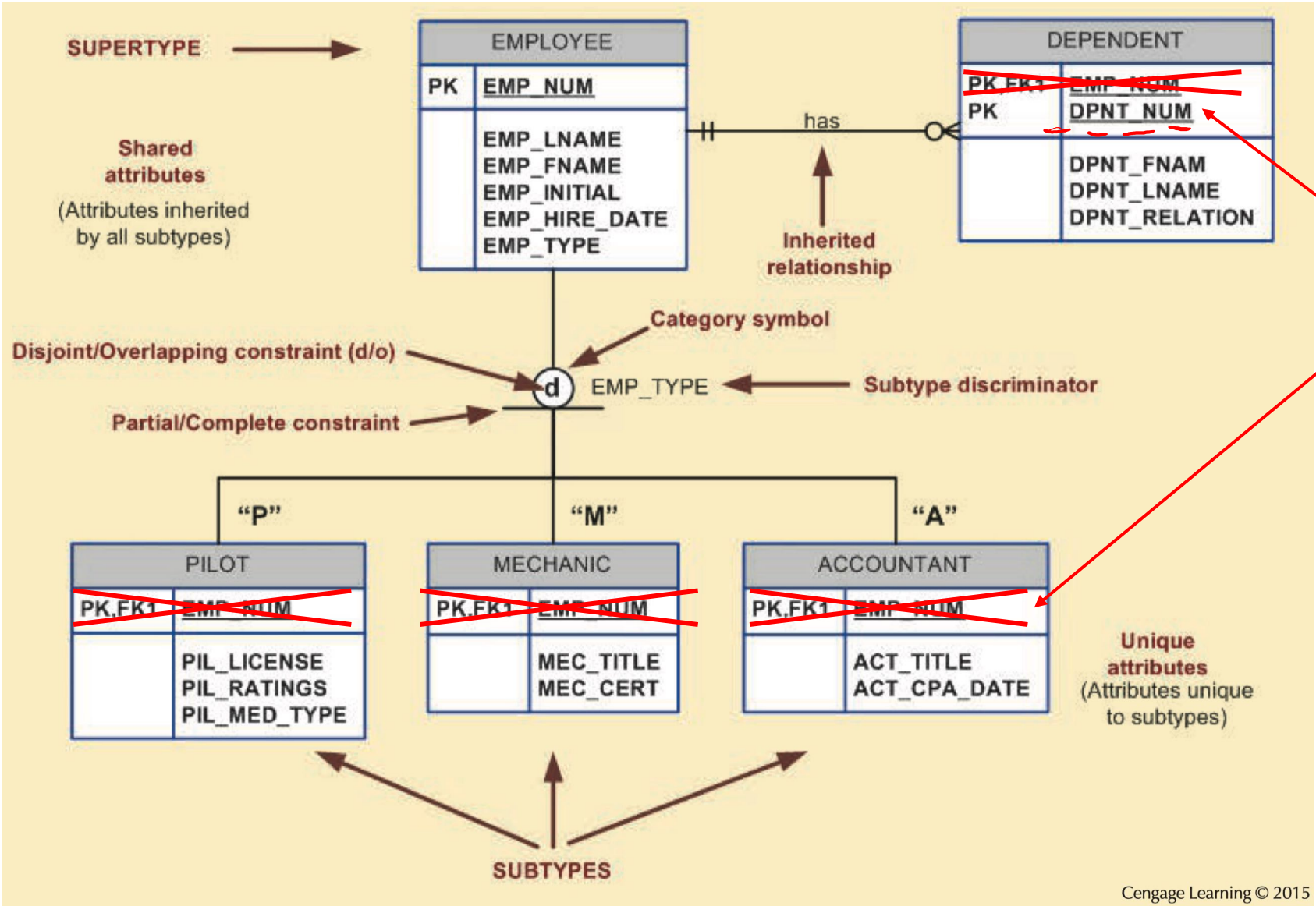
Please do not do that! It is wrong as ERD (also according to our textbook)



Do you see any problem ?

Source: Coronel, Morris, "Database systems -- design, implementation, and management", 11th ed, 2015. Fig 5-2.
Wolfgang Gatterbauer. Database design: https://northeastern-datalab.github.io/cs3200/

Please do not do that! It is wrong as ERD (also according to our textbook)

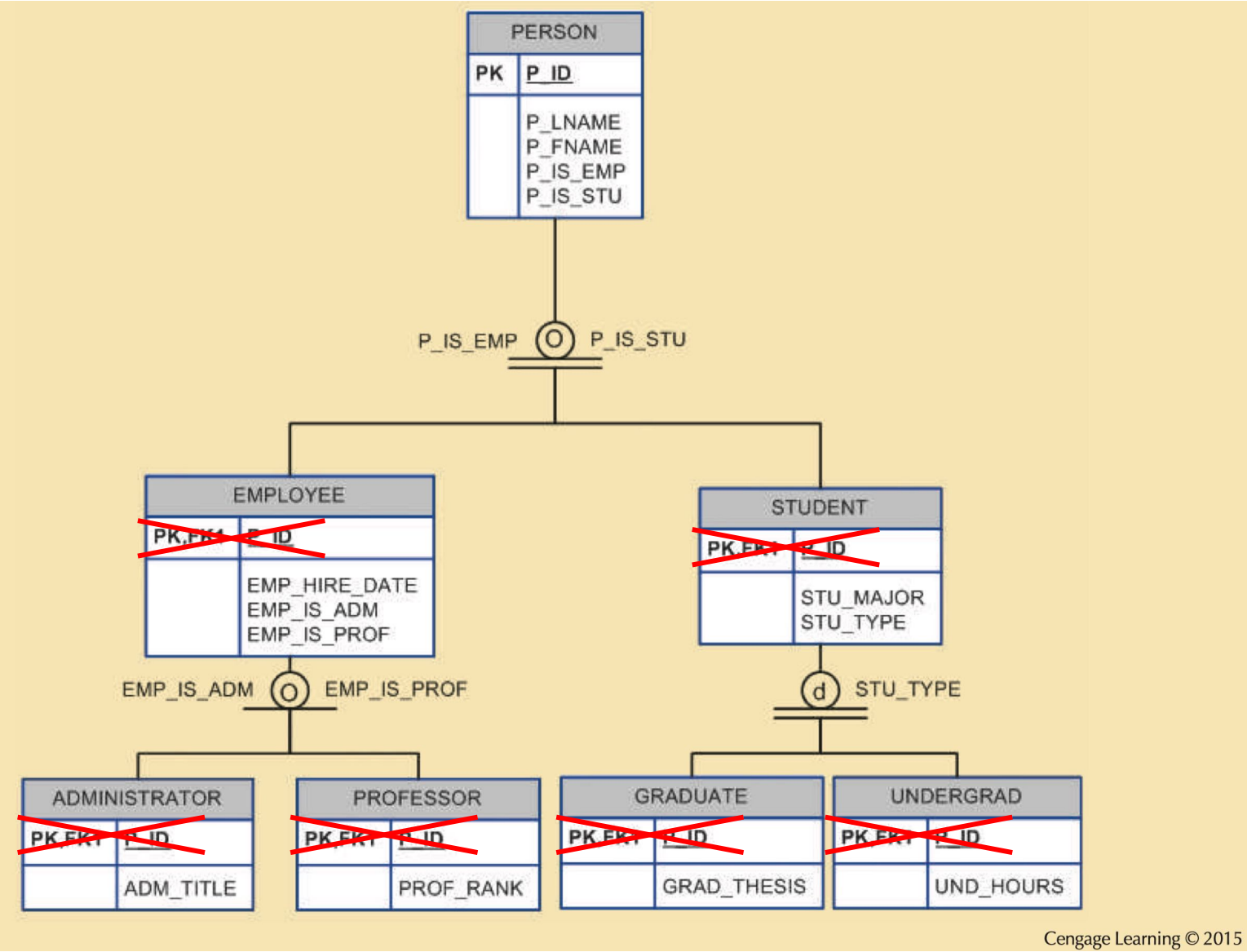


We do *not* show Fks in ERDs because:

1. they are redundant, and we want to avoid redundancy
2. they are a concept from relational tables, not ERDs

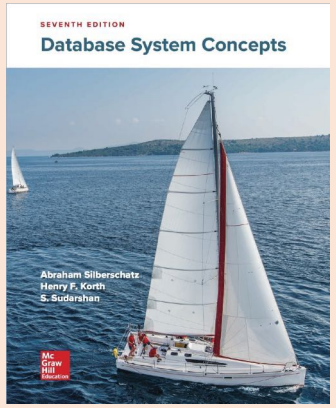
Source: Coronel, Morris, "Database systems -- design, implementation, and management", 11th ed, 2015. Fig 5-2.

Please do not do that! It is wrong as ERD (also according to our textbook)



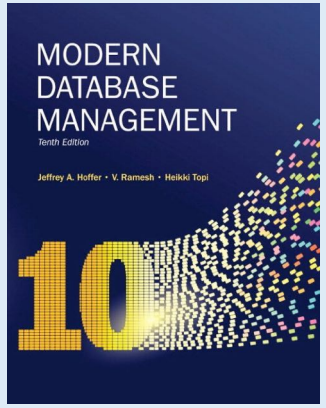
Source: Coronel, Morris, "Database systems -- design, implementation, and management", 11th ed, 2015. Fig 5-4
Wolfgang Gatterbauer. Database design: https://northeastern-datalab.github.io/cs3200/

Different sources, different notations



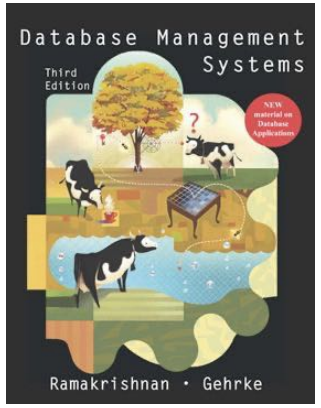
[Silberschatz+'20]

SDK arrows

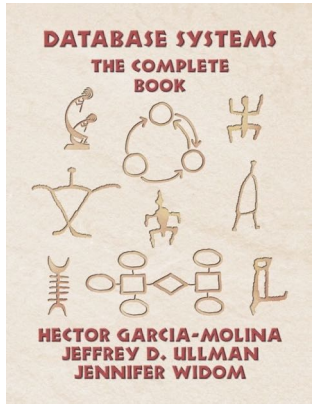


[Hoffer+'10]

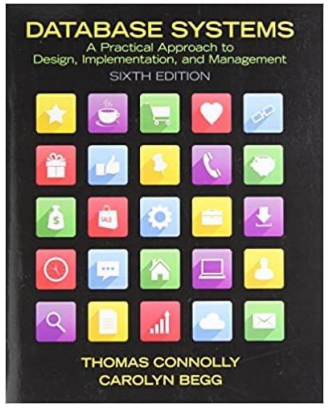
Crow's foot



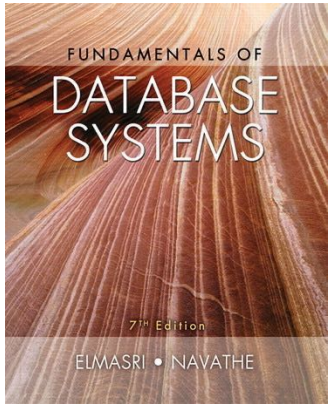
[Cow book'03]



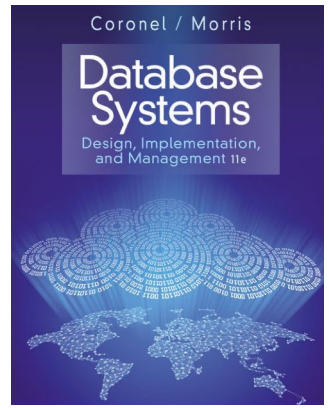
[Stanford book'08]



[Connolly+'15]



[Elmasri+'15]



[Coronel+'15]

"SDK" [Silberschatz+'20]: Silberschatz, Korth, Sudarshan. Database system concepts, 7th ed, 2020. https://www.db-book.com/db7

[Hoffer+'10]: Hoffer, Ramesh, Topi. Modern Database Management, 10th ed, 2010. https://www.pearson.com/us/higher-education/product/Hoffer-Modern-Database-Management-10th-Edition/9780136088394.html

[Cow book'03]: Ramakrishnan, Gehrke, Database Management Systems, 3rd ed, 2003. http://pages.cs.wisc.edu/~dbbook/

[Stanford book'08]: Garcia-Molina, Ullman, Widom. Database Systems: The Complete Book, 2nd ed, 2008. http://infolab.stanford.edu/~ullman/dscb.html

[Connolly+'15]: Connolly, Begg. Database systems: A practical approach to design, implementation, and management, 6th ed, 2015. https://www.pearson.com/us/higher-education/program/Connolly-Database-Systems-A-Practical-Approach-to-Design-Implementation-and-Management-6th-Edition/PGM116956.html

[Elmasri+'15]: Elmasri, Navathe. Fundamentals of Database Systems, 7th ed, 2015. https://www.pearson.com/us/higher-education/program/Elmasri-Fundamentals-of-Database-Systems-7th-Edition/PGM189052.html

[Coronel+'15]: Coronel, Morris. Database systems: design, implementation, and management, 11th ed, 2015. https://www.cengage.com/c/database-systems-11e-coronel-morris/9781285196145

Wolfgang Gatterbauer. Database design: https://northeastern-datalab.github.io/cs3200/

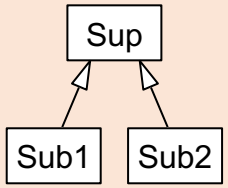
Notations for specialization ("ISA relationship")

Participation (or covering) constraint (optional=partial | mandatory=total)

Partial-overlapping *Partial-disjoint* *Total-overlapping* *Total-disjoint*

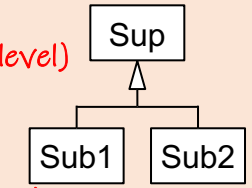
[Silberschatz+'20]

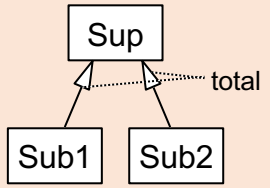
Please use the notation from our textbook even though I will use slides with various notation

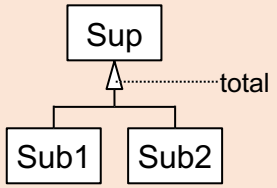


Super entity (more generalized, higher-level)

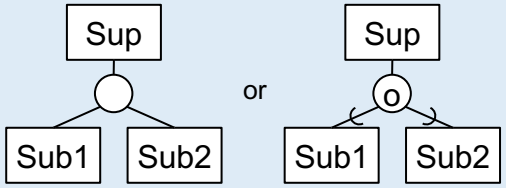
Sub entity (more specialized, lower-level)

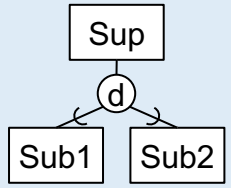


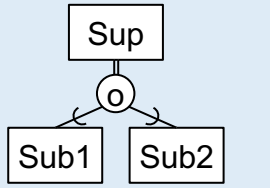


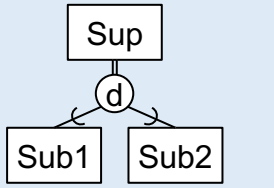


[Elmasri+'15], [Hoffer+'10]

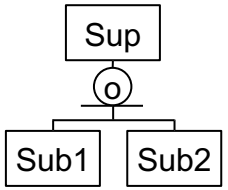


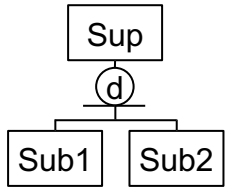


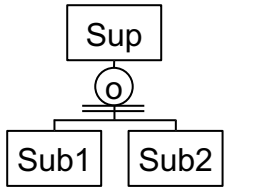


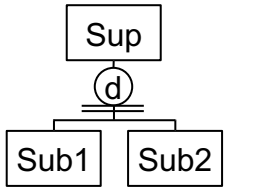


[Coronel+'15]



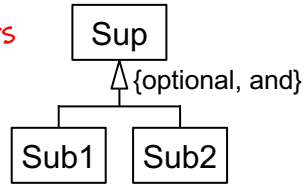


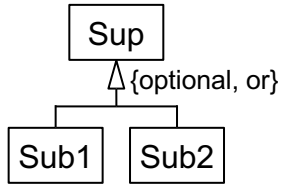


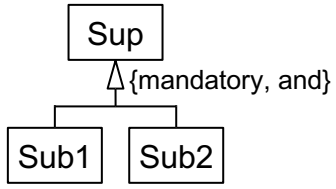


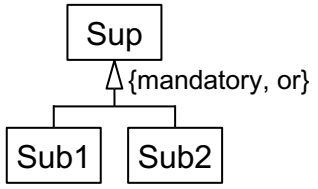
UML *[Connolly+'15]*

Overlap (or disjoint) constraints (or=disjoint | and=overlapping)





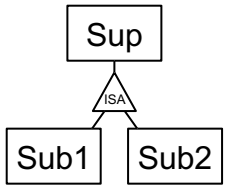




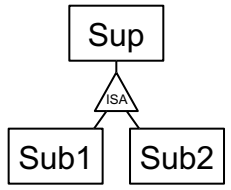
[Stanford book'03]

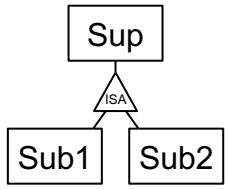
also by Gradiance

[Cow book'03]

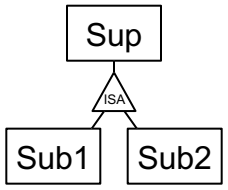


Sub1 overlaps Sub2

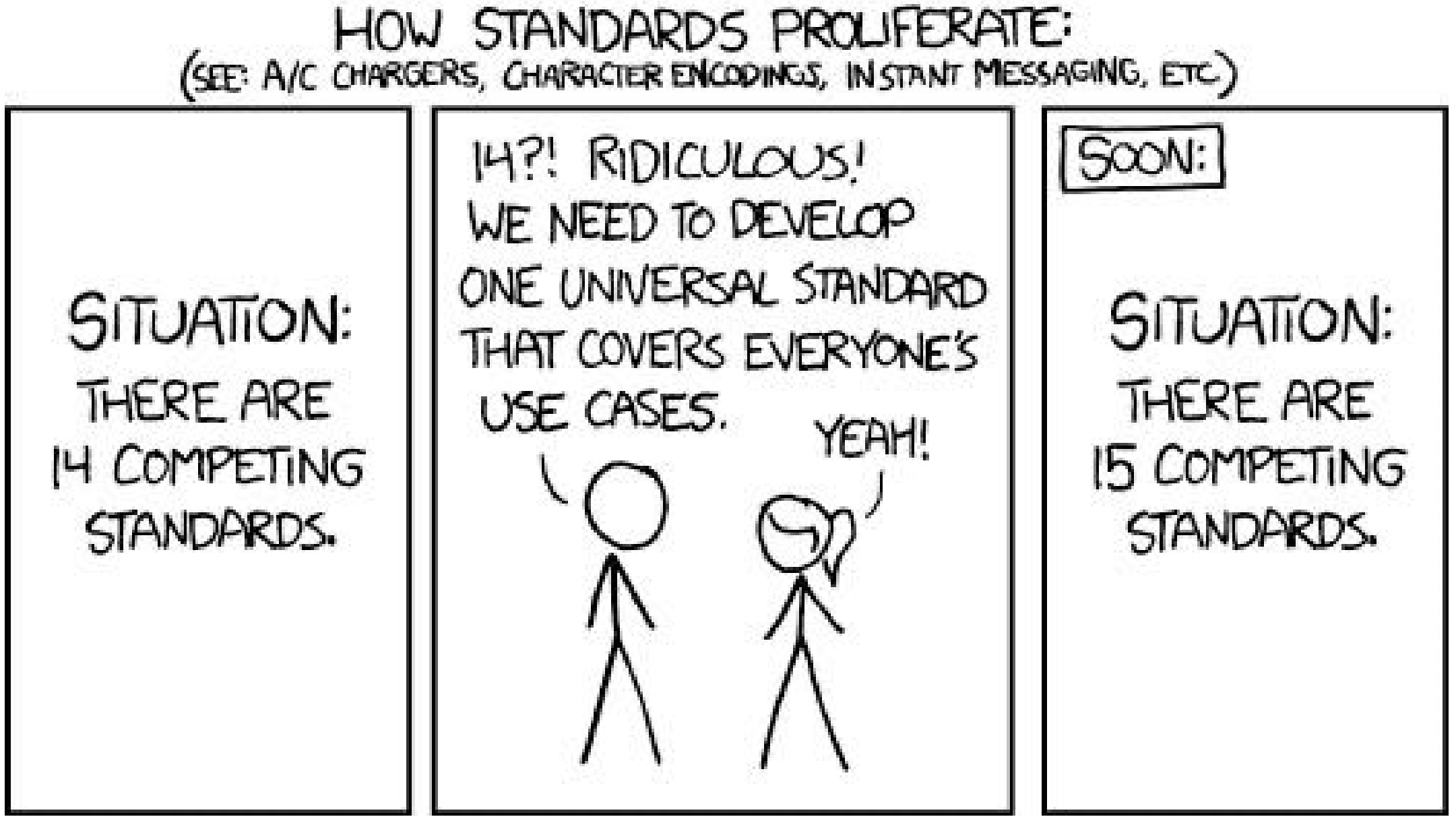




Sub1 overlaps Sub2
Sub1 and Sub2 cover Sup



Sub1 and Sub2 cover Sup



Source and further pointers: https://xkcd.com/927/, https://www.explainxkcd.com/wiki/index.php/927:_Standards

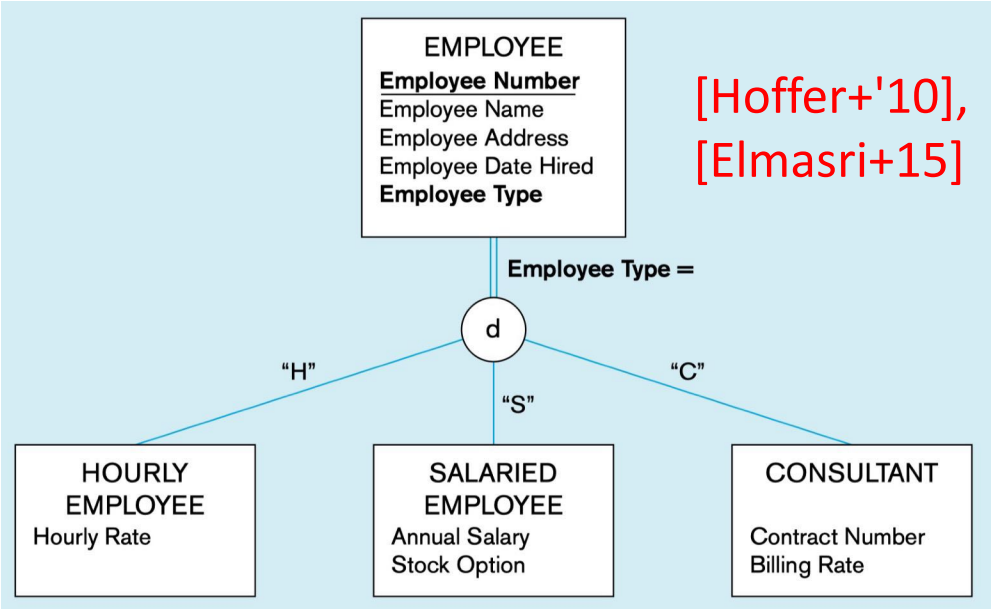
Relational Translation of Enhanced ER diagrams

Mapping Supertype/Subtype relationships

- One relation for supertype and for each subtype
- Supertype attributes (including identifier and subtype discriminator if used) go into supertype relation
- Subtype attributes go into each subtype; primary key of supertype relation also becomes primary key of subtype relation
- 1:1 relationship established between supertype and each subtype, with supertype as primary table (FK from subtype to supertype)

- Note: This is not as standard a mapping process as the previous examples were. The approach described is commonly used though

Translating Supertype/Subtype relationships

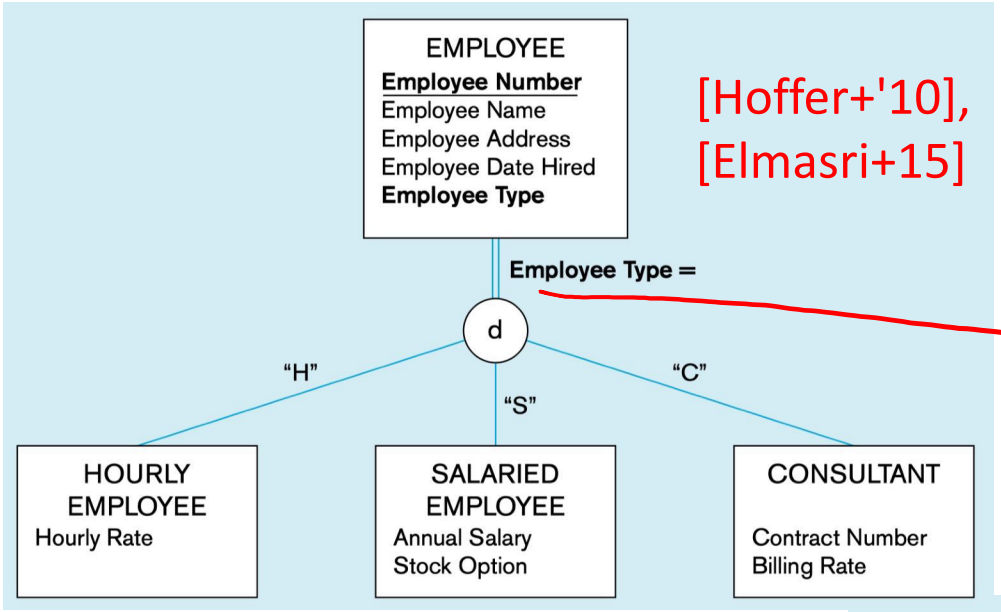


[Hoffer+'10], [Elmasri+15]

How to translate into relational tables ?

Source: Hoffer, Ramesh, Topi, "Modern database management," 12th ed, 2016. Figures 4-20, 4-21
Wolfgang Gatterbauer. Database design: https://northeastern-datalab.github.io/cs3200/

Translating Supertype/Subtype relationships



[Hoffer+'10], [Elmasri+15]

How to we define the employeetype attribute ? $\in \{H, S, C\}$

N.N.

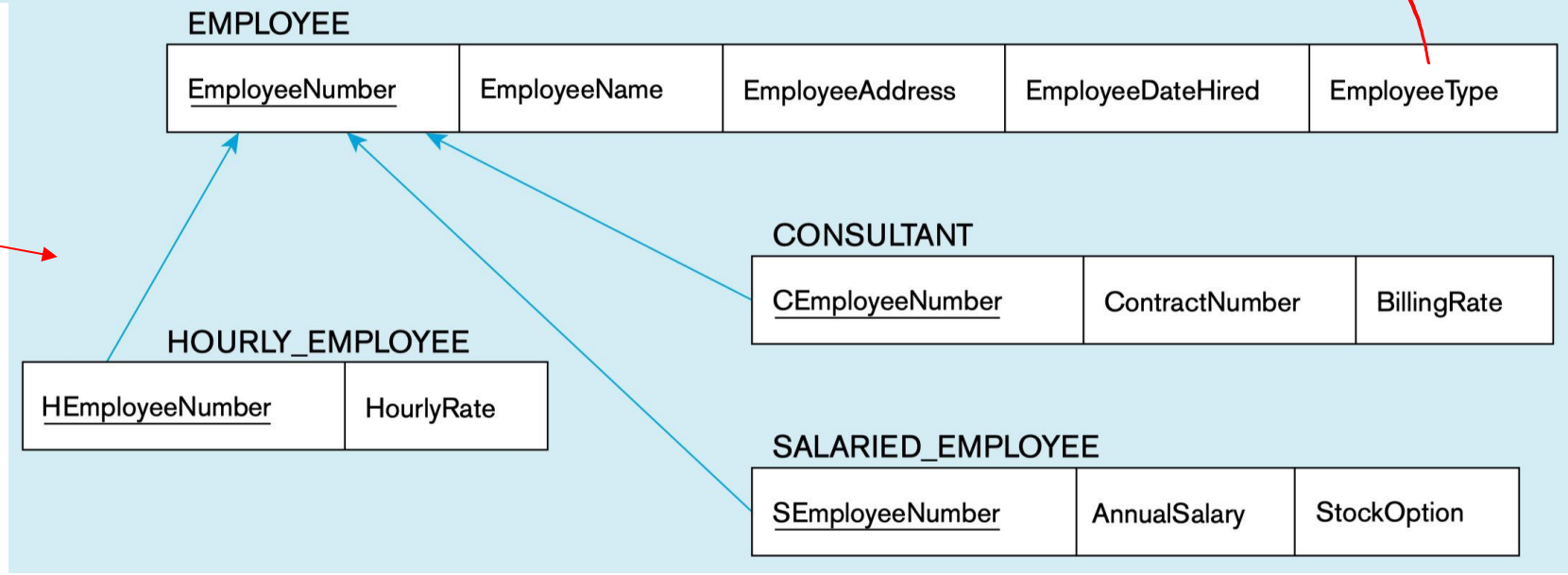
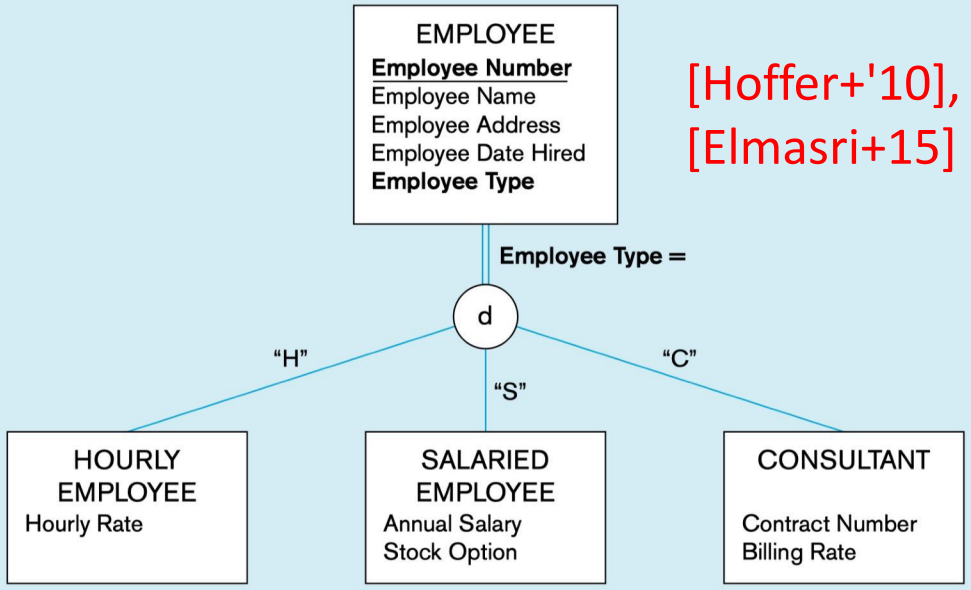


Figure here copied from [Hoffer+'10]. You can either put attributes vertically (as I usually draw them) or horizontally (as here and other copied figures from [Hoffer+10])

Source: Hoffer, Ramesh, Topi, "Modern database management," 12th ed, 2016. Figures 4-20, 4-21

Translating Supertype/Subtype relationships

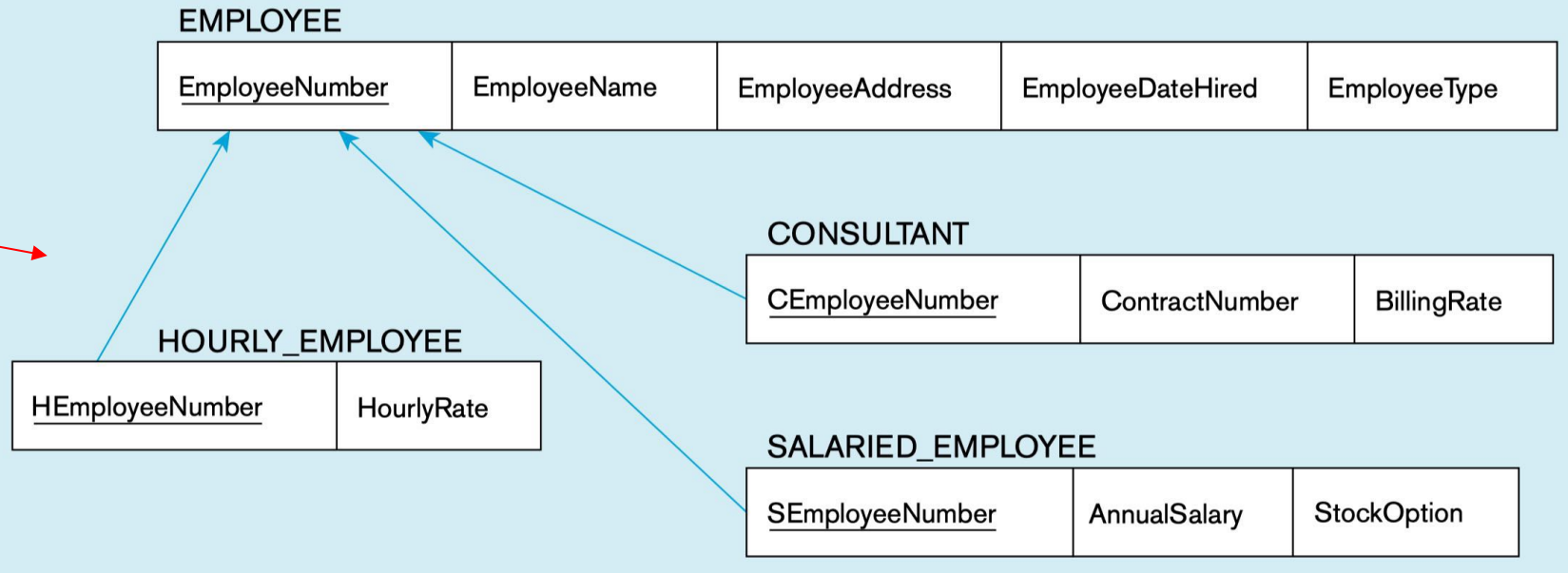


[Hoffer+'10], [Elmasri+15]

```
create table Employee(
    num int PRIMARY KEY,
    name varchar(20),
    datehired date,
    etype varchar(1) not null,
CHECK (etype in ('C', 'H', 'S')));

create table Consultant(
    cnum int PRIMARY KEY,
    contract int,
    billingrate decimal(5,2),
FOREIGN KEY (cnum) REFERENCES Employee);
```

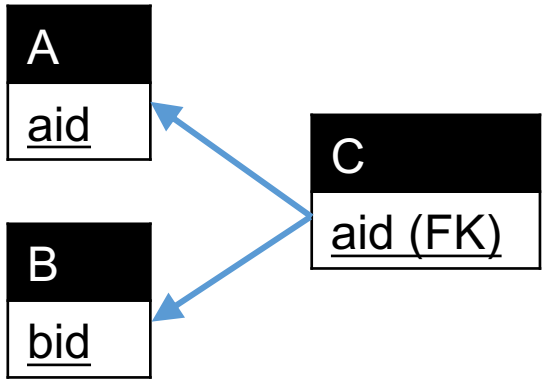
```
INSERT INTO Employee VALUES (1, 'Alice', '11-16-2022', 'C');
INSERT INTO Consultant VALUES (1, 234, 50);

INSERT INTO Employee VALUES (2, 'Bob', '11-16-2022', 'H');
INSERT INTO Hourly_employee VALUES (2, 30);
```

Figure here copied from [Hoffer+'10]. You can either put attributes vertically (as I usually draw them) or horizontally (as here and other copied figures from [Hoffer+10])



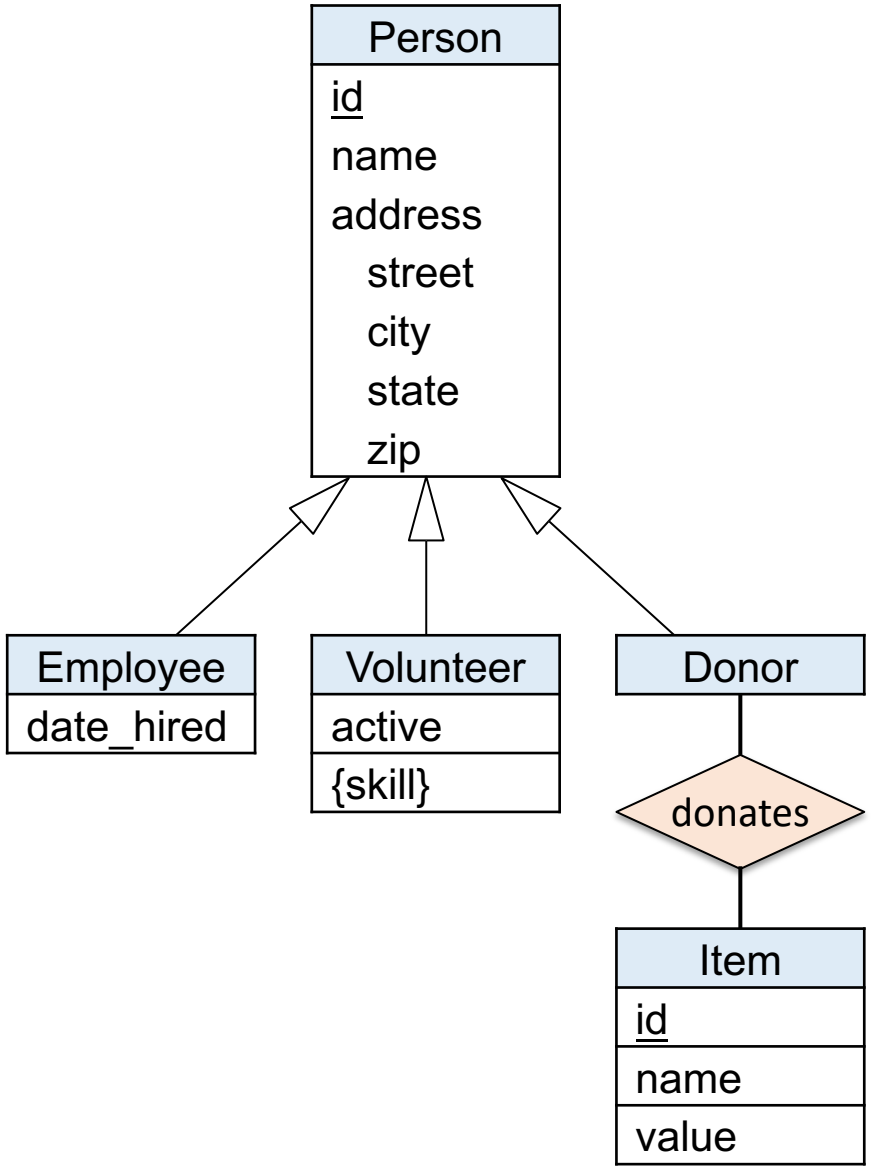
Source: Hoffer, Ramesh, Topi, "Modern database management," 12th ed, 2016. Figures 4-20, 4-21

One attribute can be part of multiple FKs




```
create table A(
    aid int PRIMARY KEY);

create table B(
    bid int PRIMARY KEY);

create table C(
    cid int PRIMARY KEY,
FOREIGN KEY (cid) REFERENCES A,
FOREIGN KEY (cid) REFERENCES B);
```

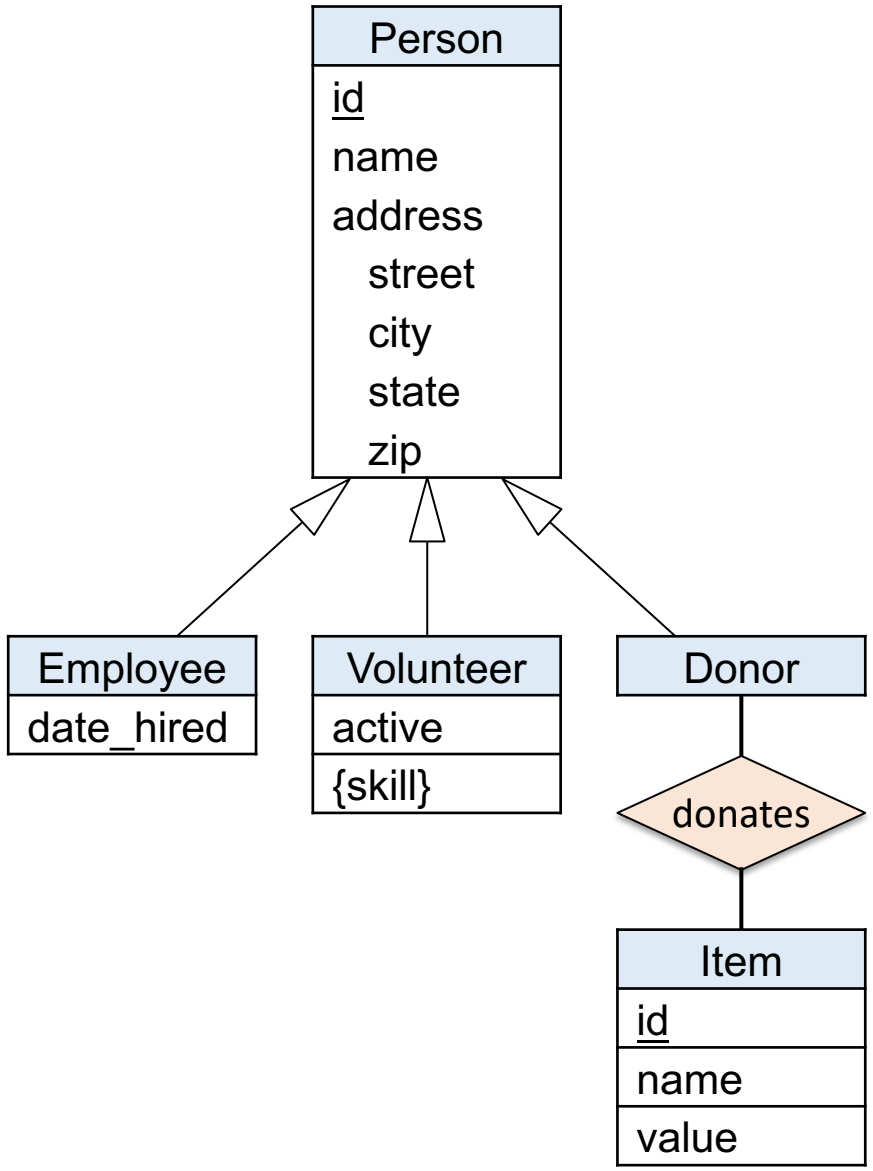
In-Class Exercise

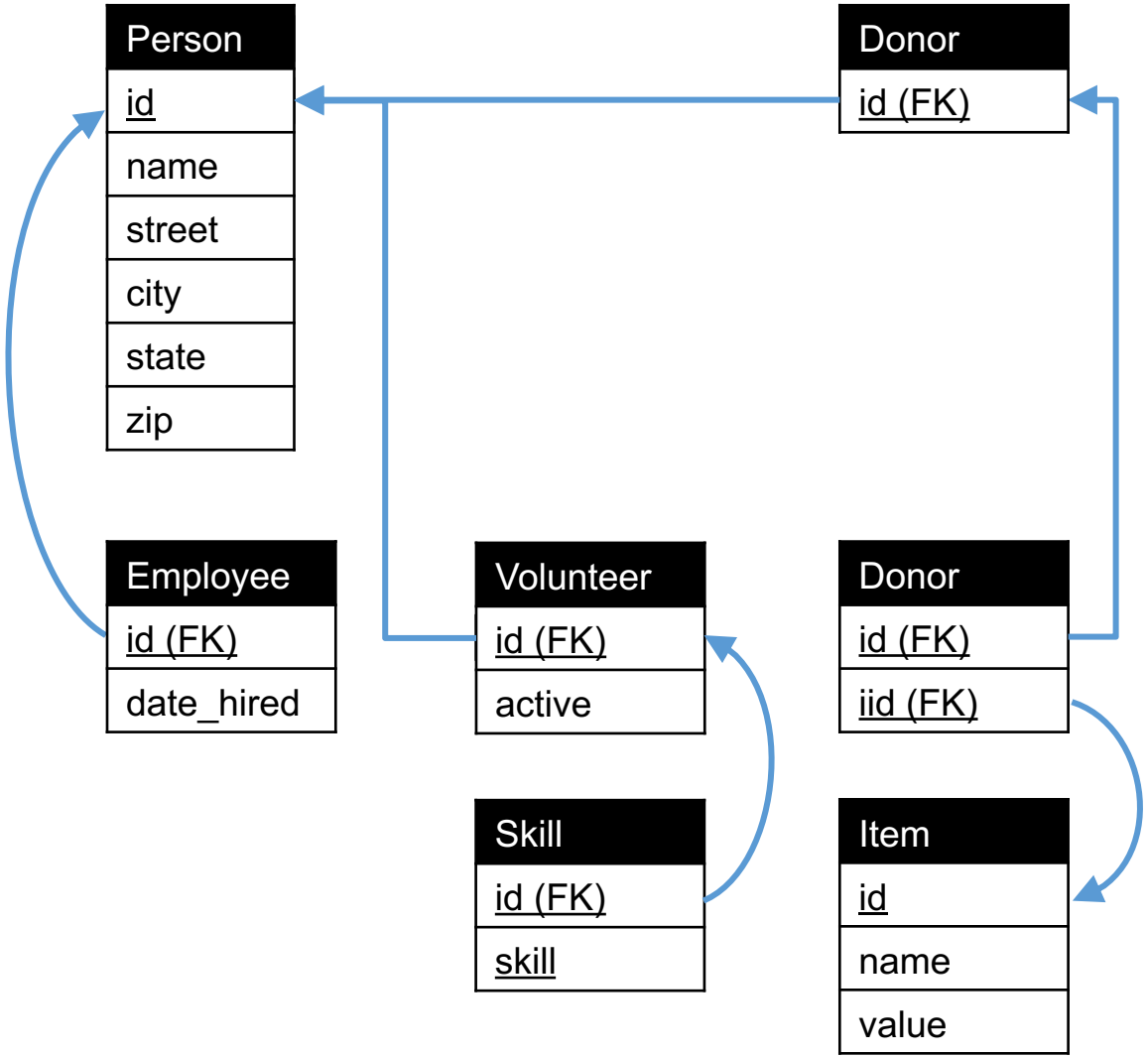


Translate the ERD on the left into a relational schema, by being as faithful to the diagram as possible, and by avoiding redundant tables or attributes if possible.

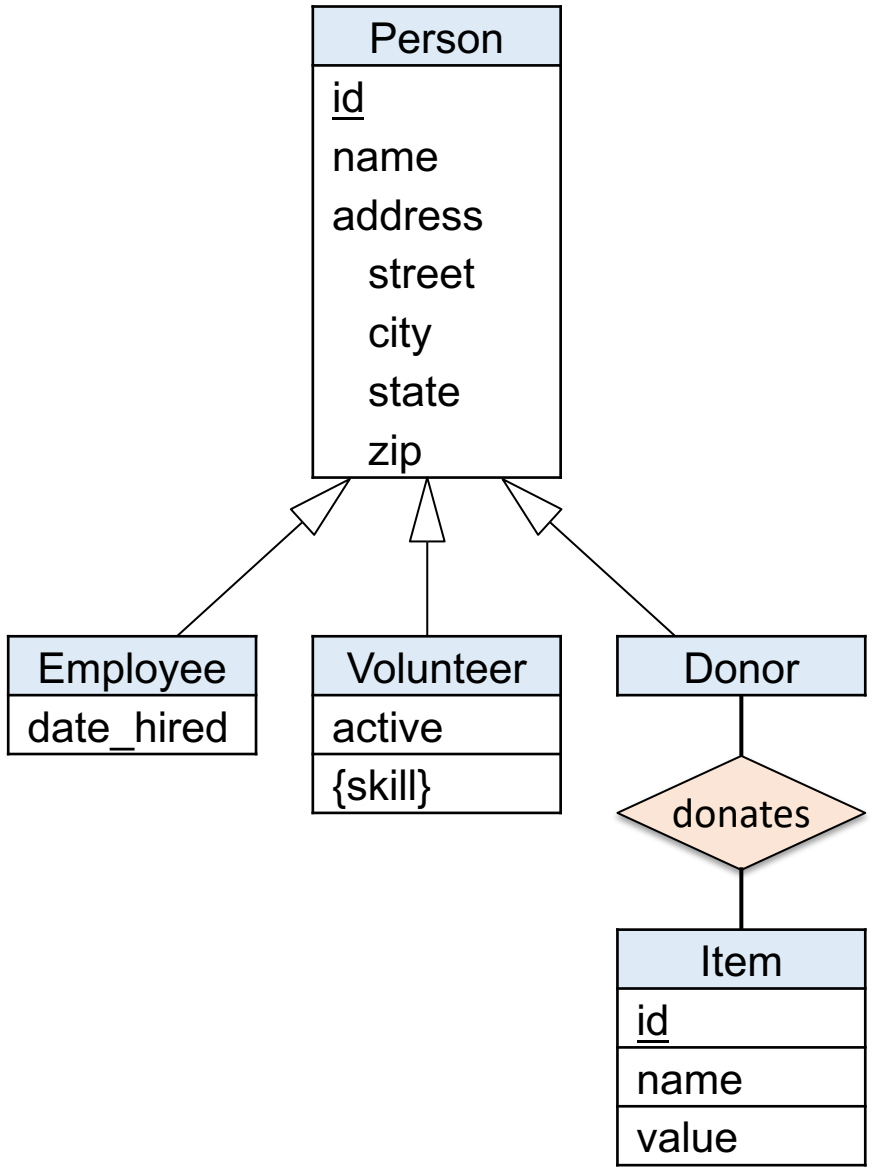
?

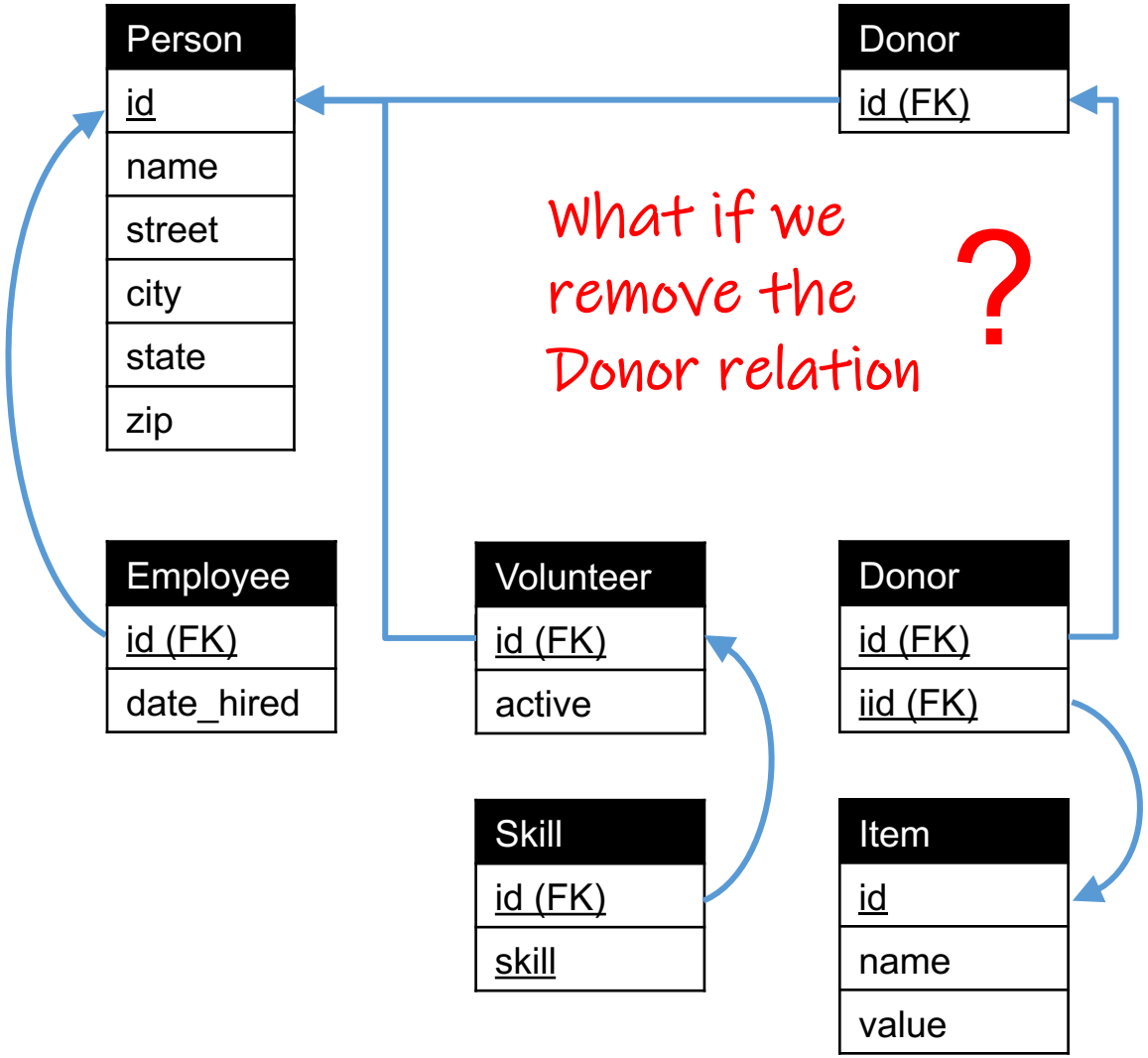
In-Class Exercise Solution



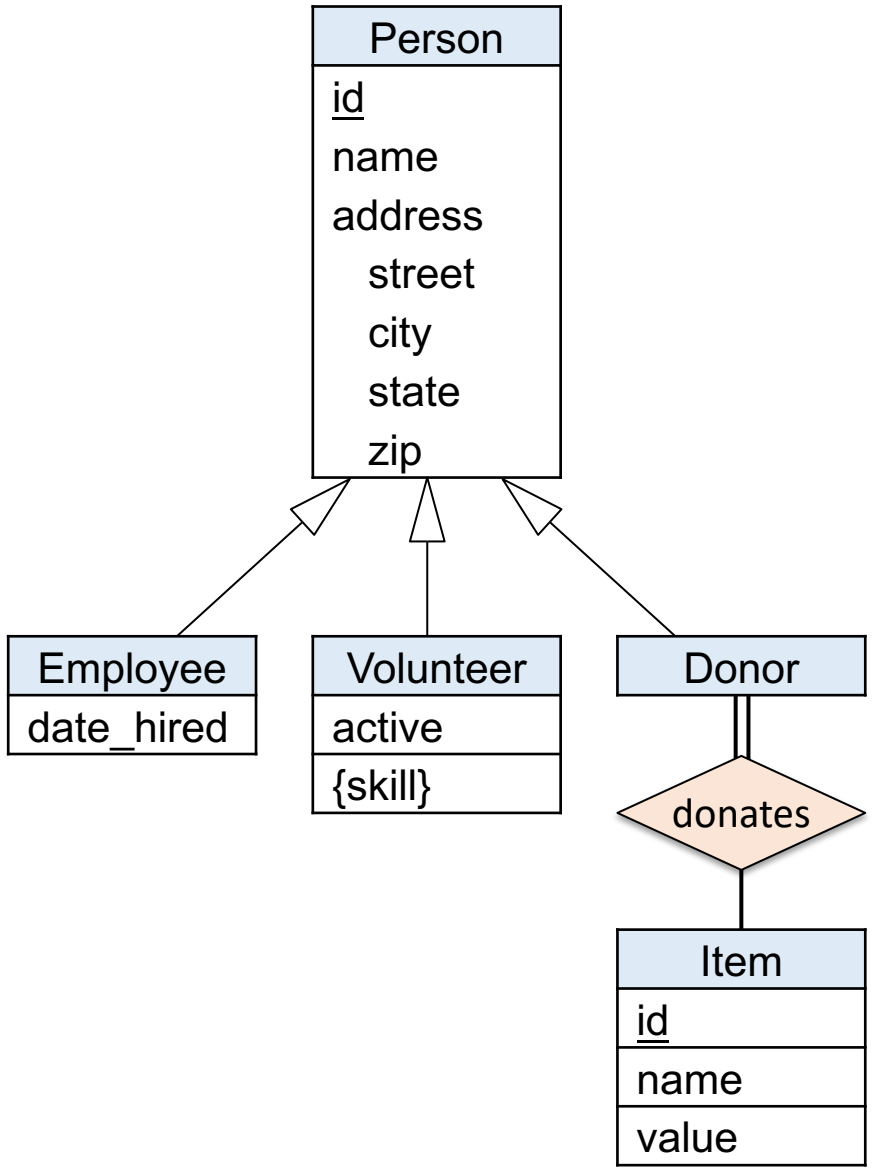


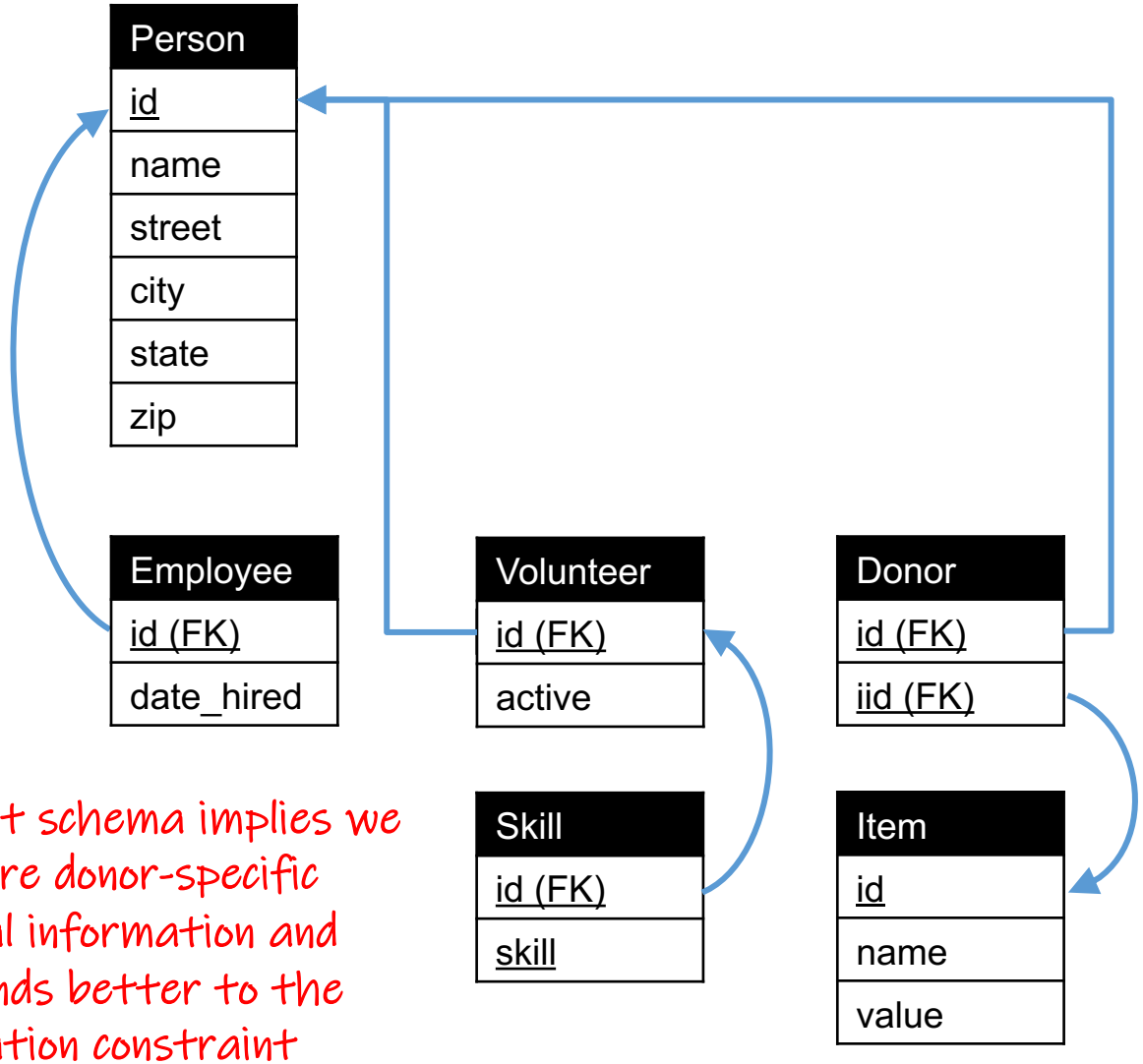
In-Class Exercise Solution





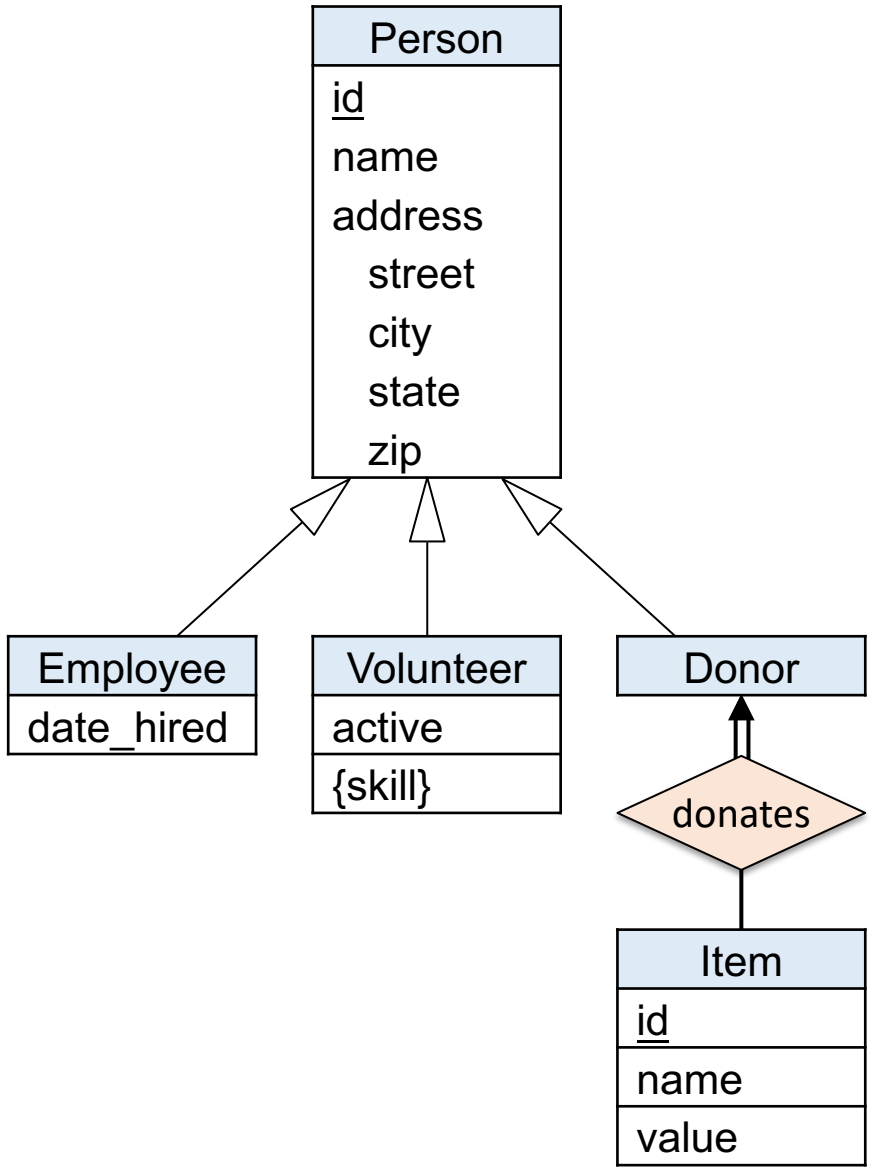
In-Class Exercise Solution

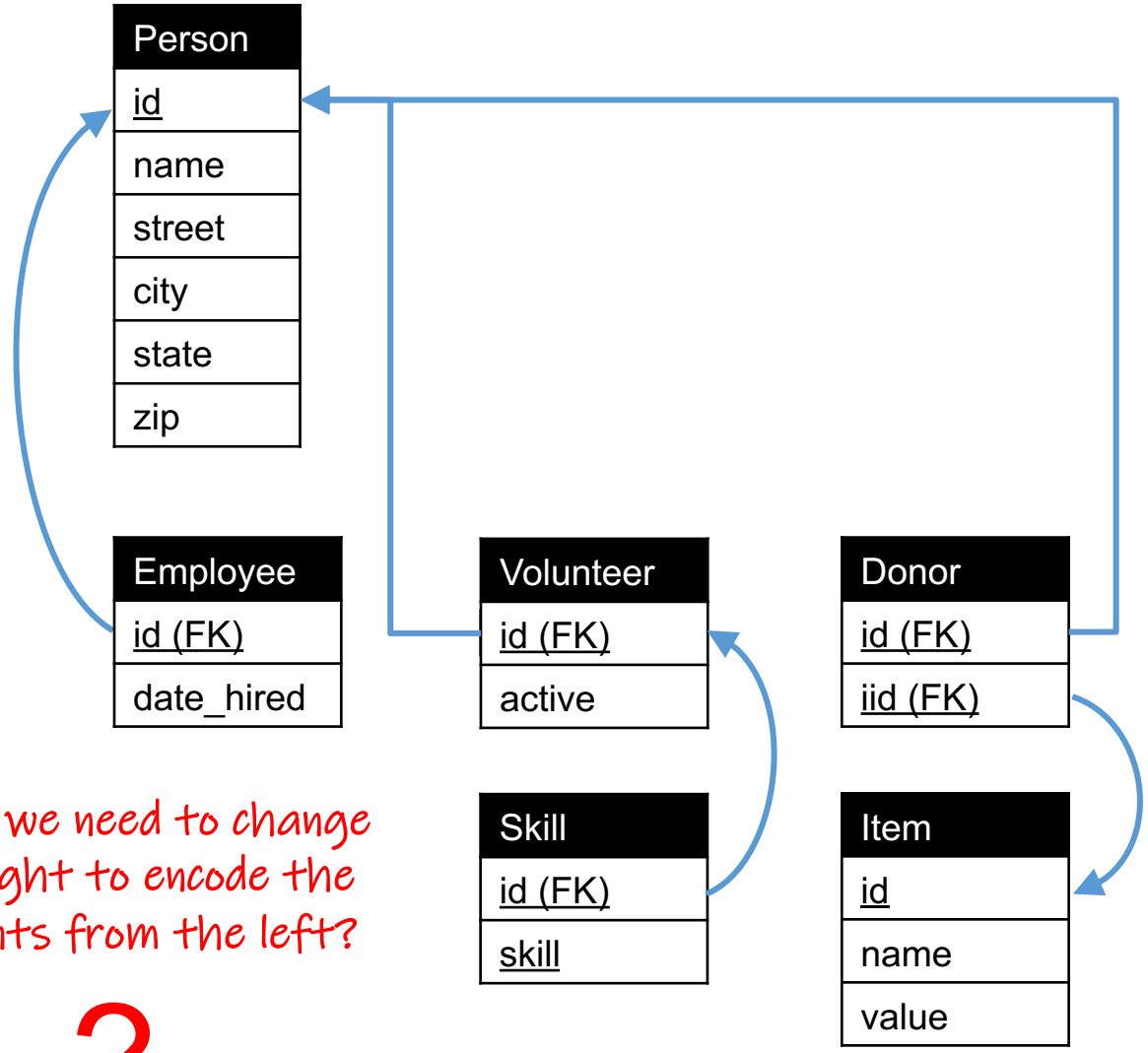




The right schema implies we can't store donor-specific additional information and corresponds better to the participation constraint shown on the left

In-Class Exercise Solution

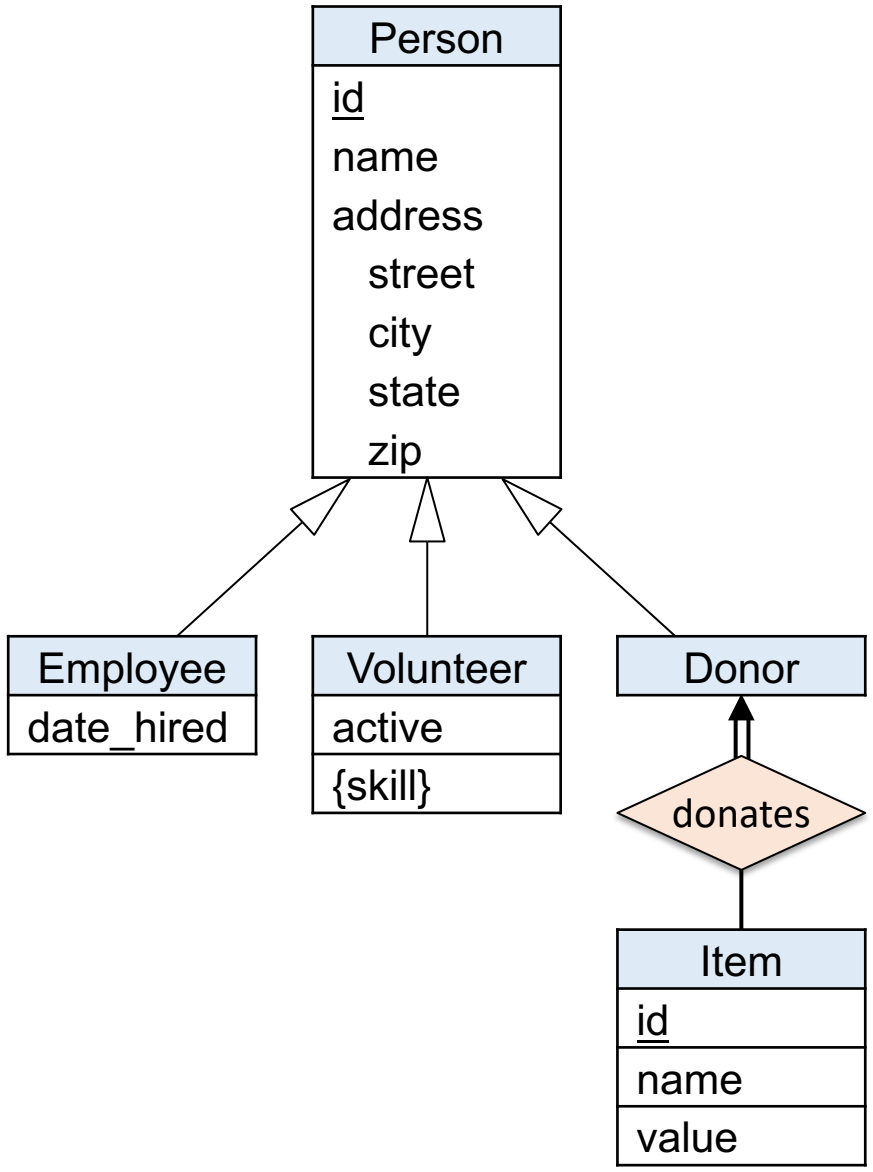


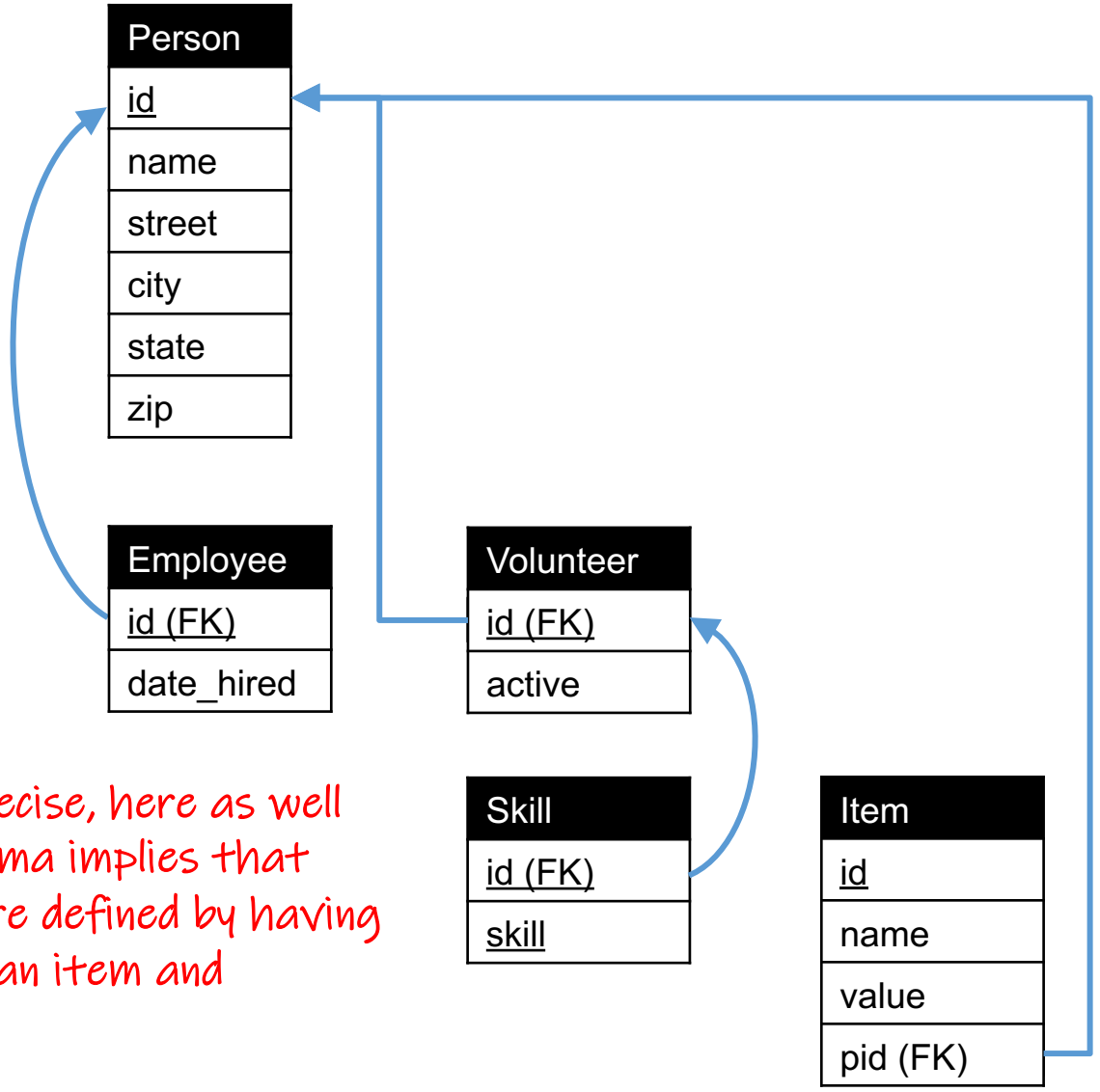


What do we need to change on the right to encode the constraints from the left?

?

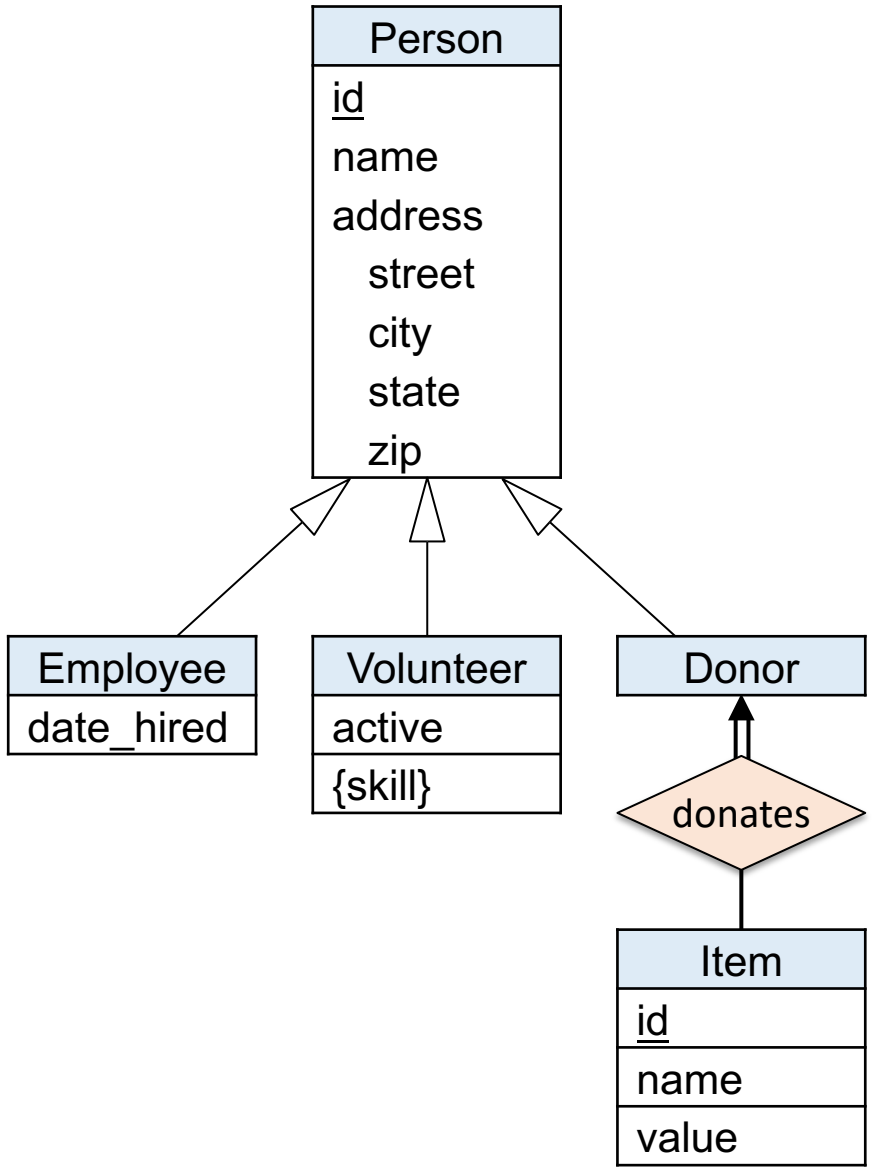
In-Class Exercise Solution

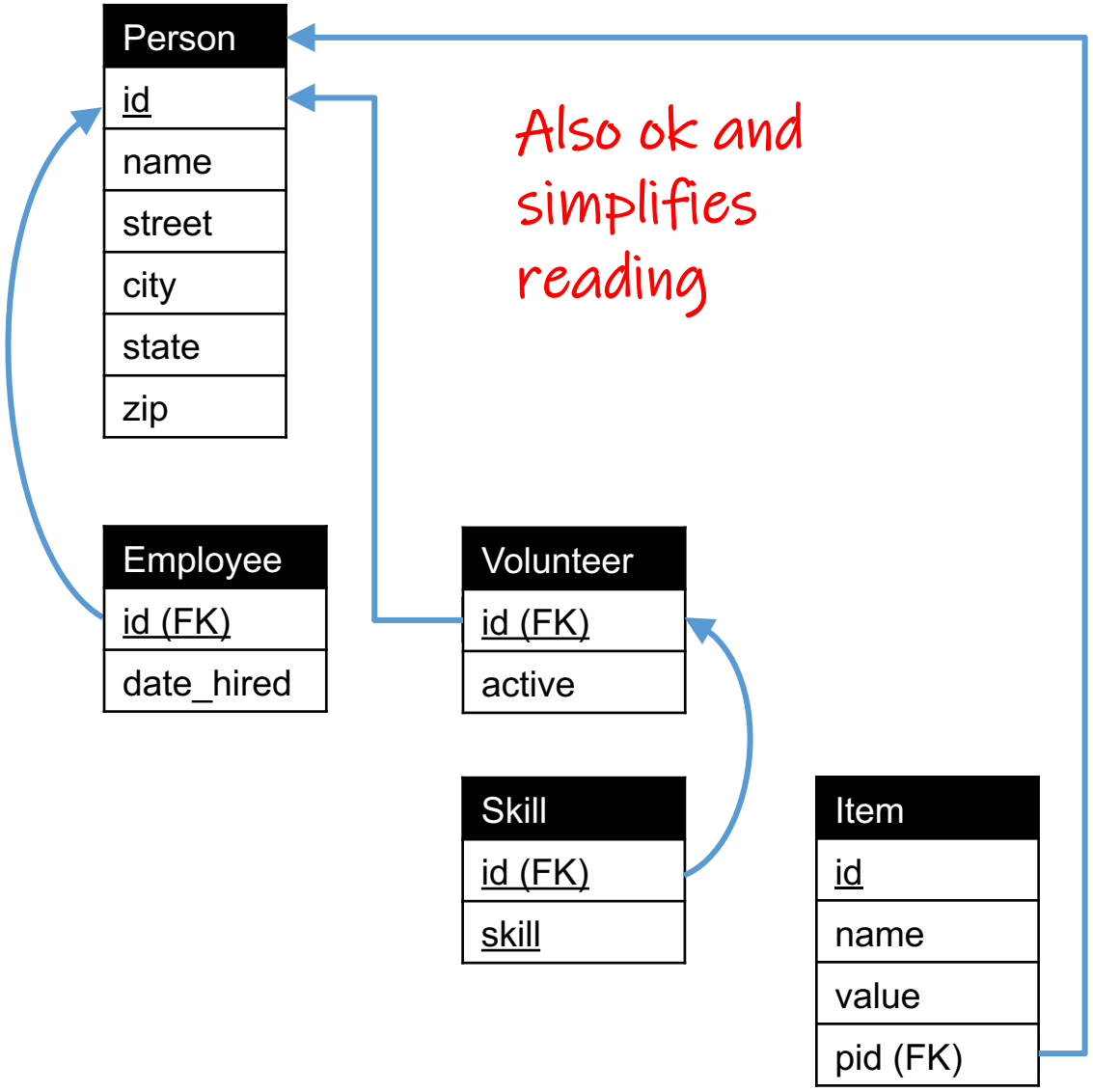




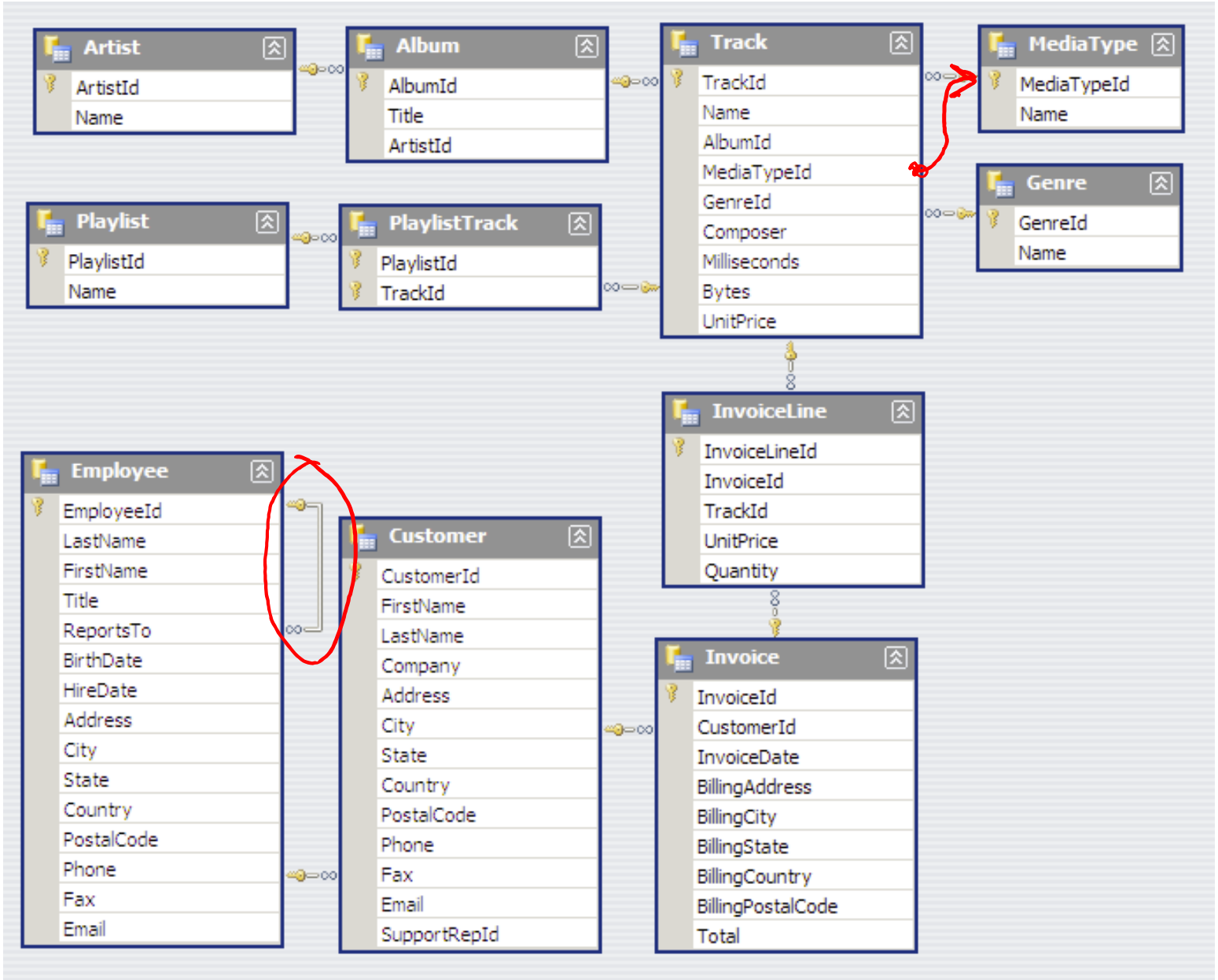
To be precise, here as well the schema implies that donors are defined by having donated an item and

In-Class Exercise Solution





319 Chinook database



Notice that join conditions are not shown correctly here. You may have to look at the actual schema definition in SQL ☺

Source: https://chinookdatabase.codeplex.com/
Wolfgang Gatterbauer. Database design: https://northeastern-datalab.github.io/cs3200/

Specialization "lattice" [Elmasri+15]

I will not ask you about lattices!

[Hoffer+'10], [Elmasri+15]

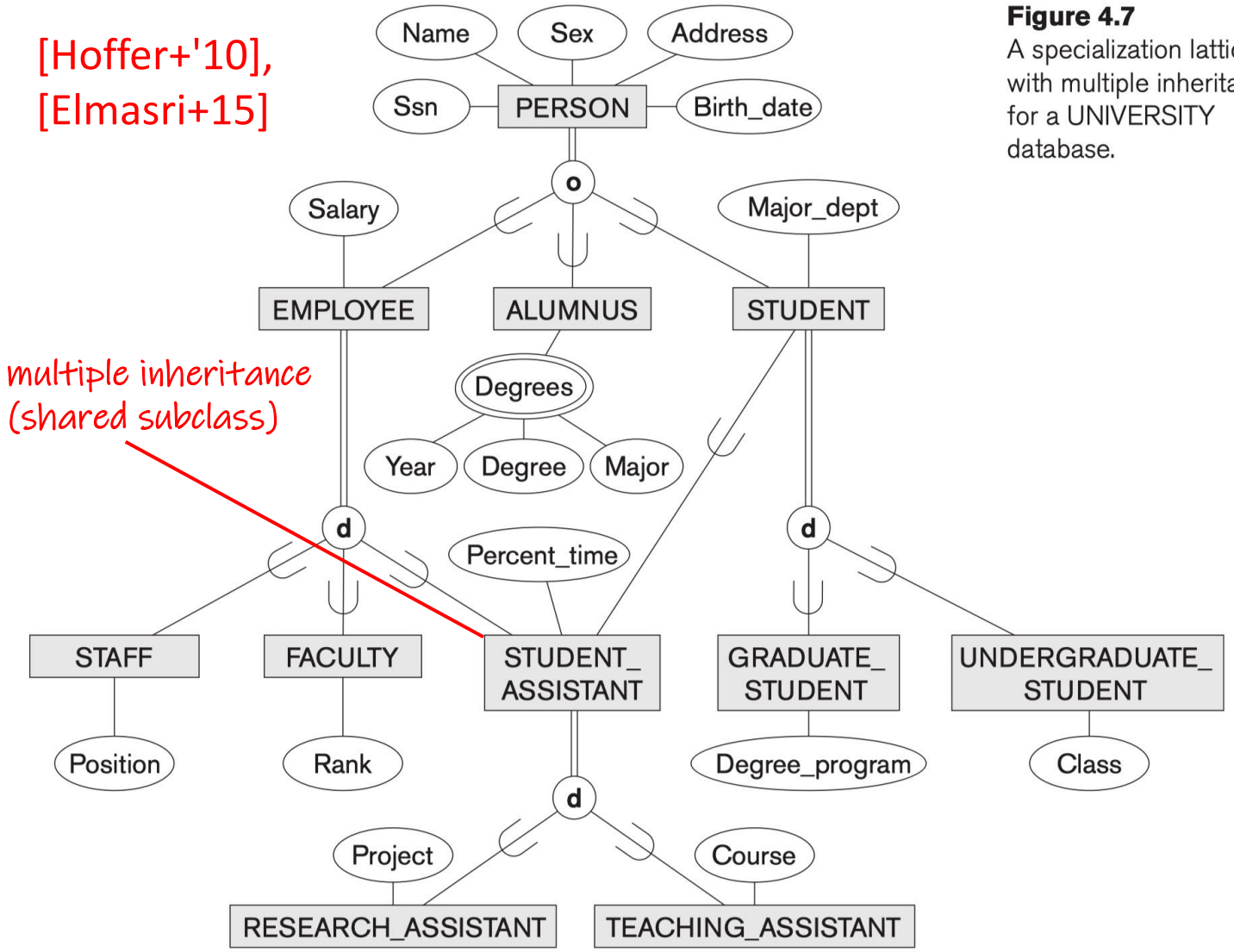


Figure 4.7
A specialization lattice with multiple inheritance for a UNIVERSITY database.

Lattice: special type of Partial Order (Poset) in which any two elements have a unique supremum (least upper bound) and unique infimum (greatest lower bound)

Source: Elmasri, Navathe. Fundamentals of Database Systems, 7th ed, 2015. Fig 4.7
Wolfgang Gatterbauer. Database design: https://northeastern-datalab.github.io/cs3200/

Specialization "lattice" [Elmasri+15]

I will not ask you about lattices!

[Hoffer+'10], [Elmasri+15]

Lattice: special type of Partial Order (Poset) in which any two elements have a unique supremum (least upper bound) and unique infimum (greatest lower bound)

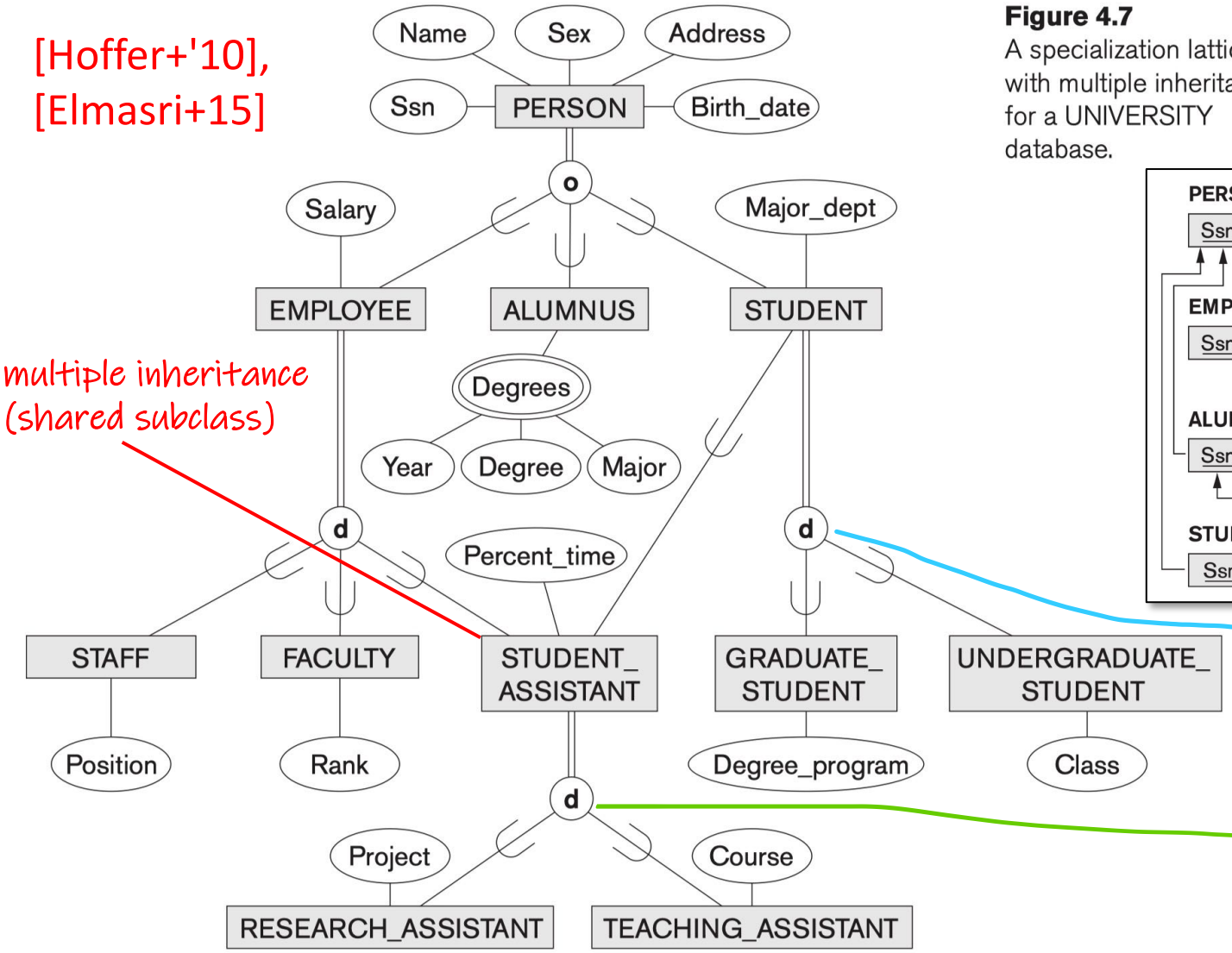


Figure 4.7
A specialization lattice with multiple inheritance for a UNIVERSITY database.

multiple inheritance (shared subclass)

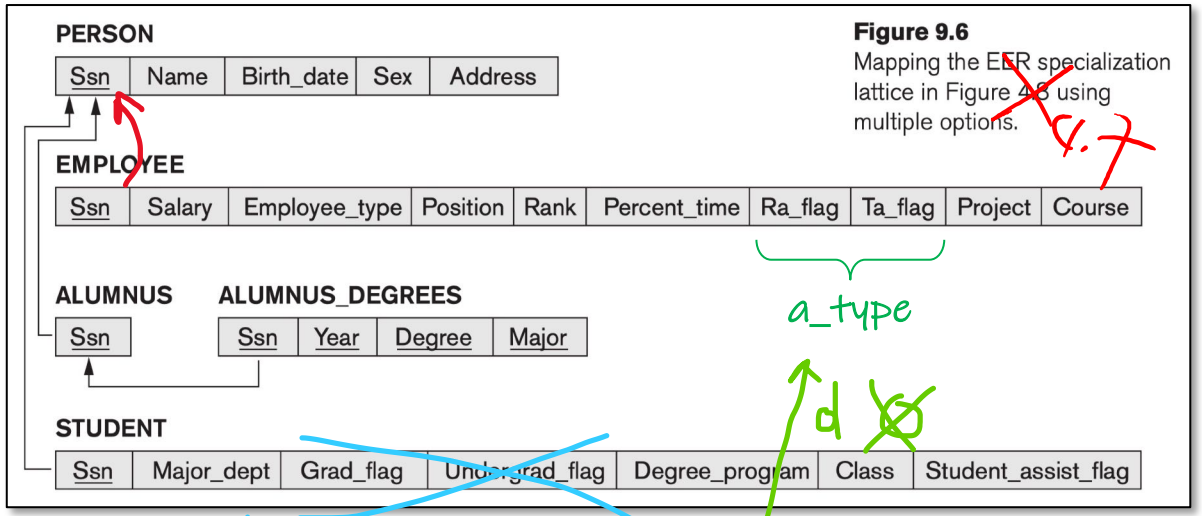


Figure 9.6
Mapping the EER specialization lattice in Figure 4.8 using multiple options.

Source: Elmasri, Navathe. Fundamentals of Database Systems, 7th ed, 2015. Fig 4.7
Wolfgang Gatterbauer. Database design: https://northeastern-datalab.github.io/cs3200/