Updated 11/2/2022

Topic 2: Database design
L14: ER modeling

Wolfgang Gatterbauer

CS3200 Database design (fa22)

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

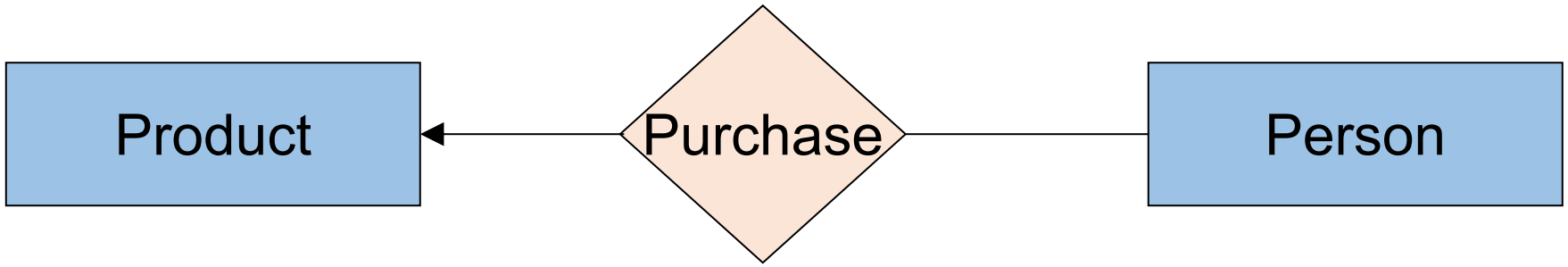
10/26/2022

Class warm-up

- Last class summary
- Project overview: web page
  - We stop 15min early / group self-assignment
- Thanks for feedback
  - homework groups: everything goes
  - more interactive questions in class ("checkpoints", "examples")
  - Please continue use our various options for feedback. Constructive feedback ("I suggest doing X instead of Y because it helps me do Z") is more helpful than feedback without concrete suggestions for changes ("exam question format was confusing")
- We continue with Database Design, hands-on

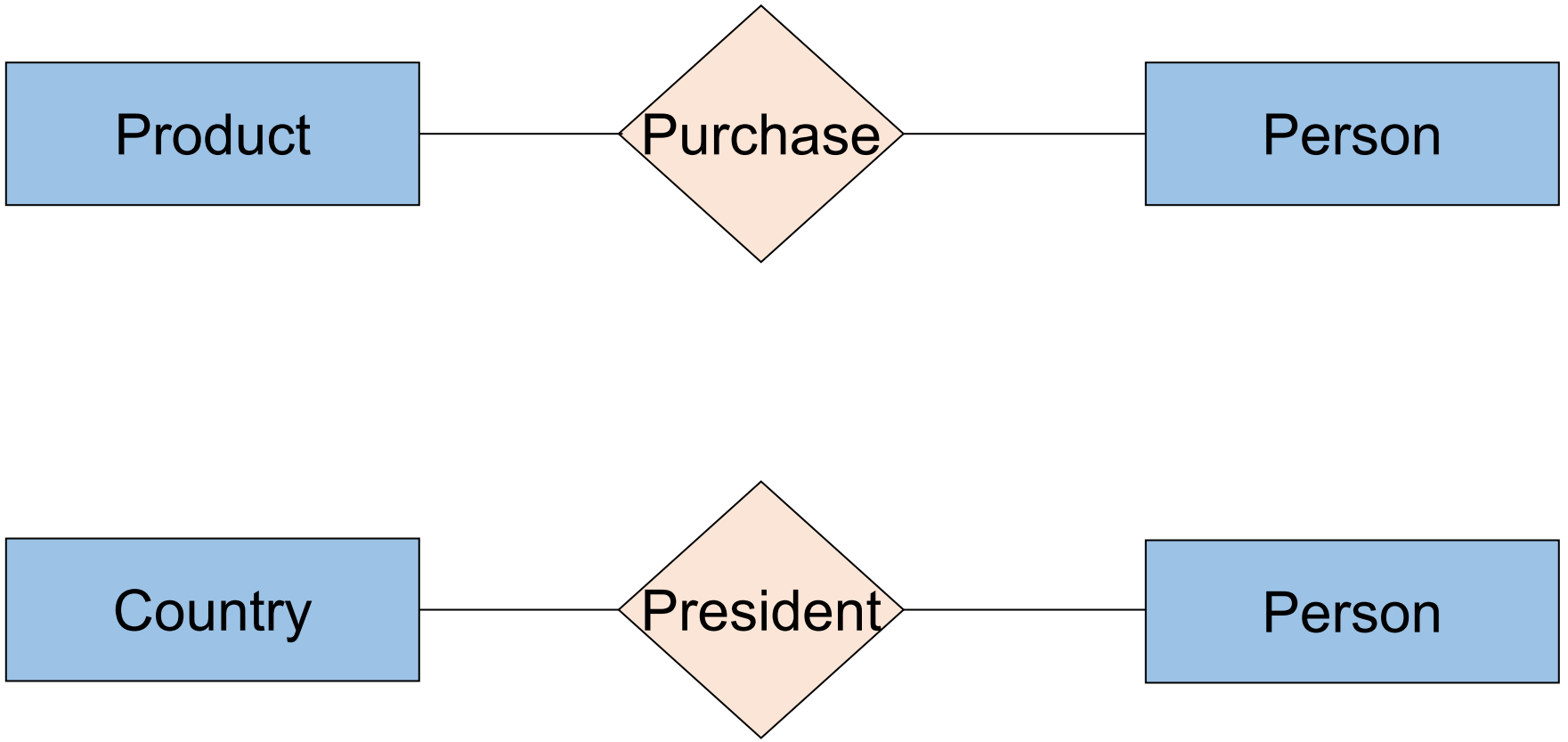
More Practice

Design Practice: What's Wrong?

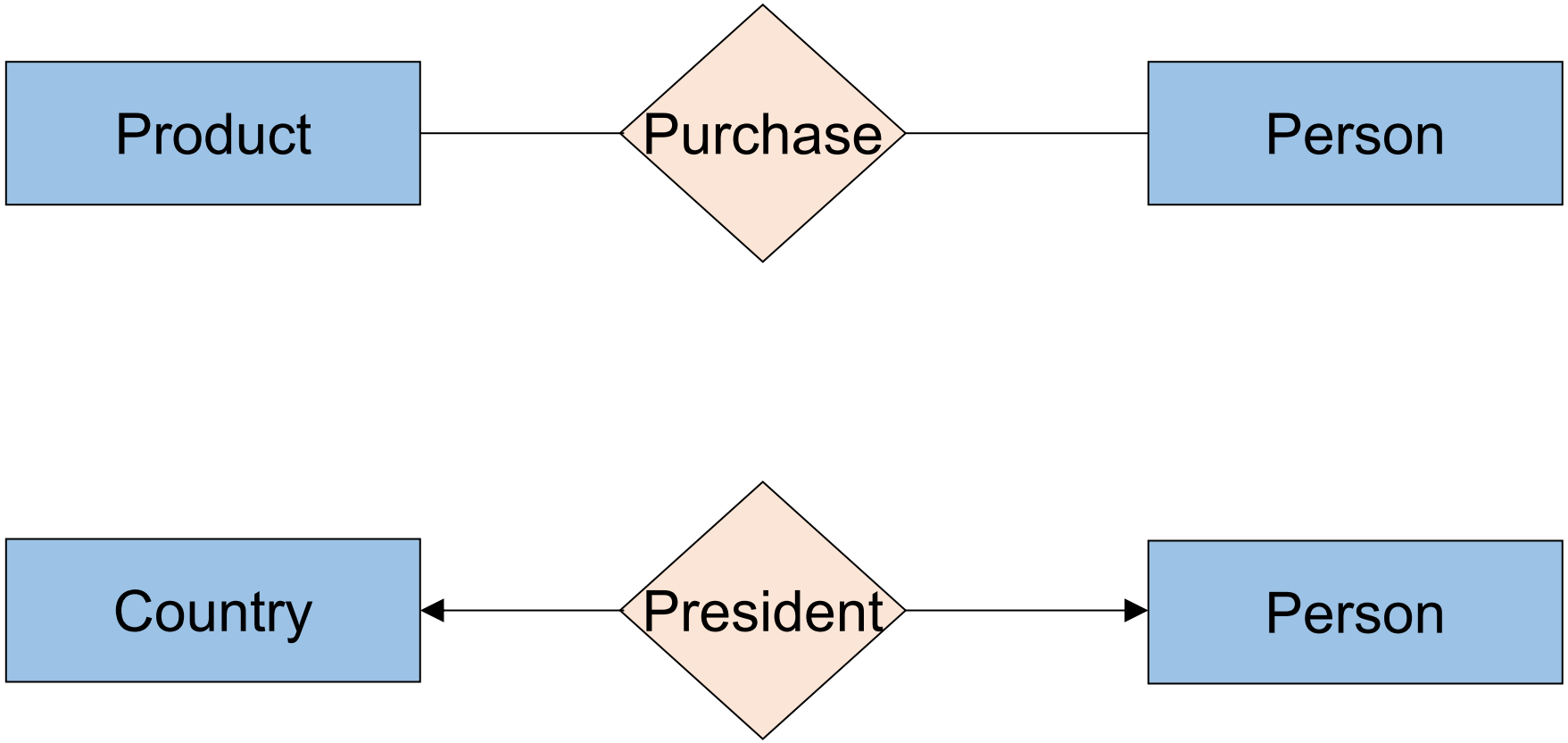


?

Design Practice: What's Wrong?



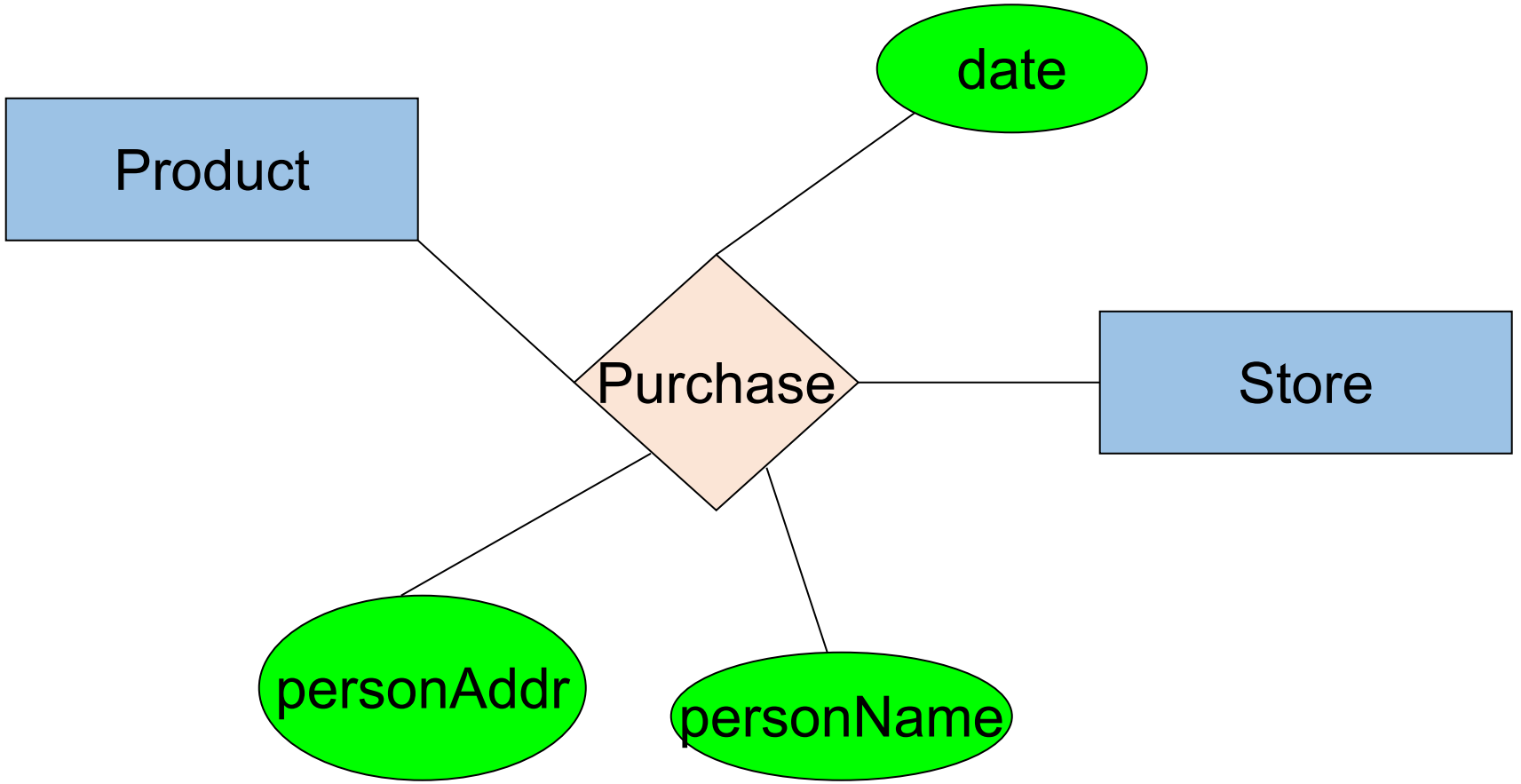
Design Practice: What's Wrong?



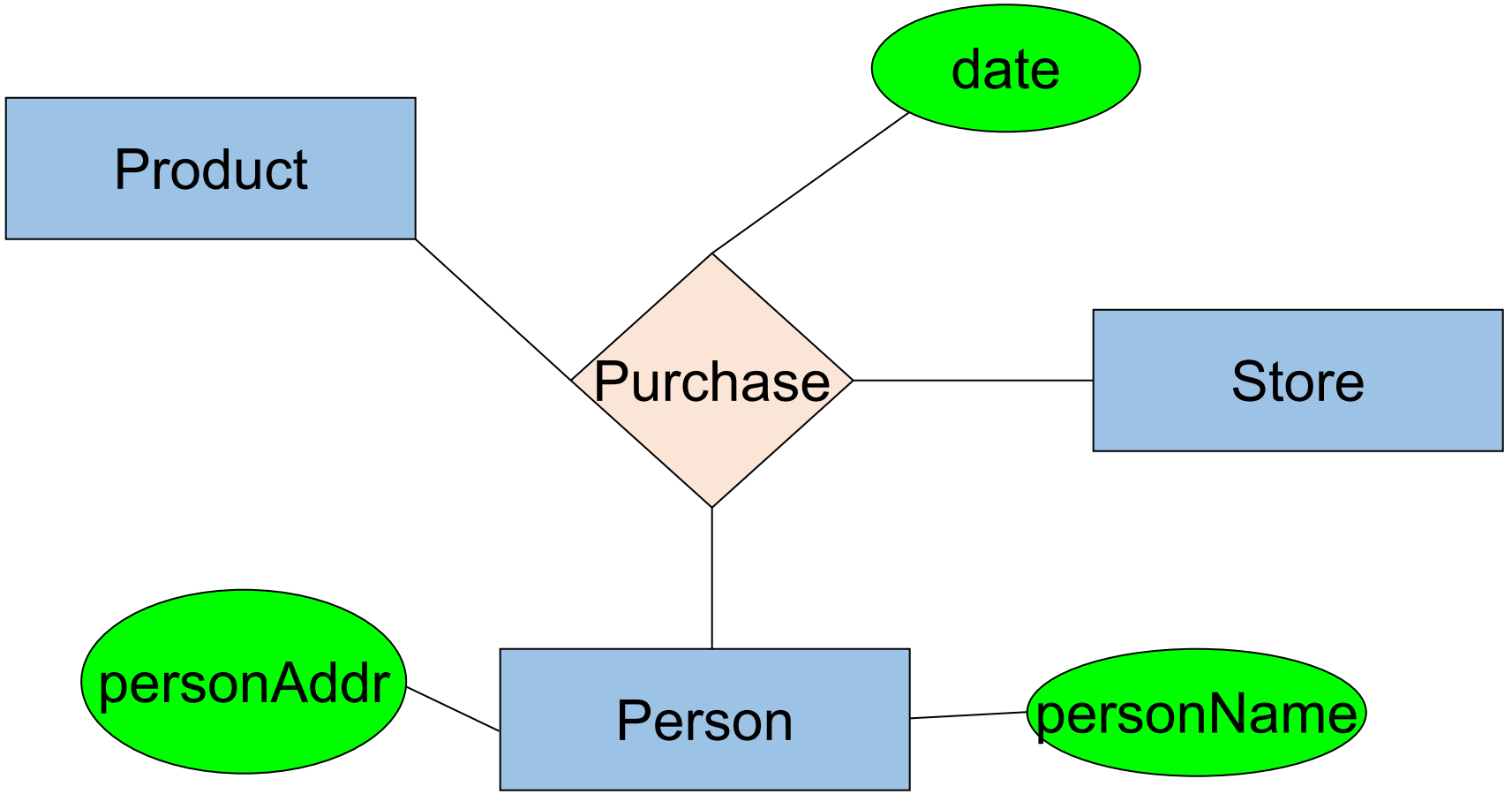
Moral: be faithful to the specifications of the application!

Design Practice: What's Wrong?

?

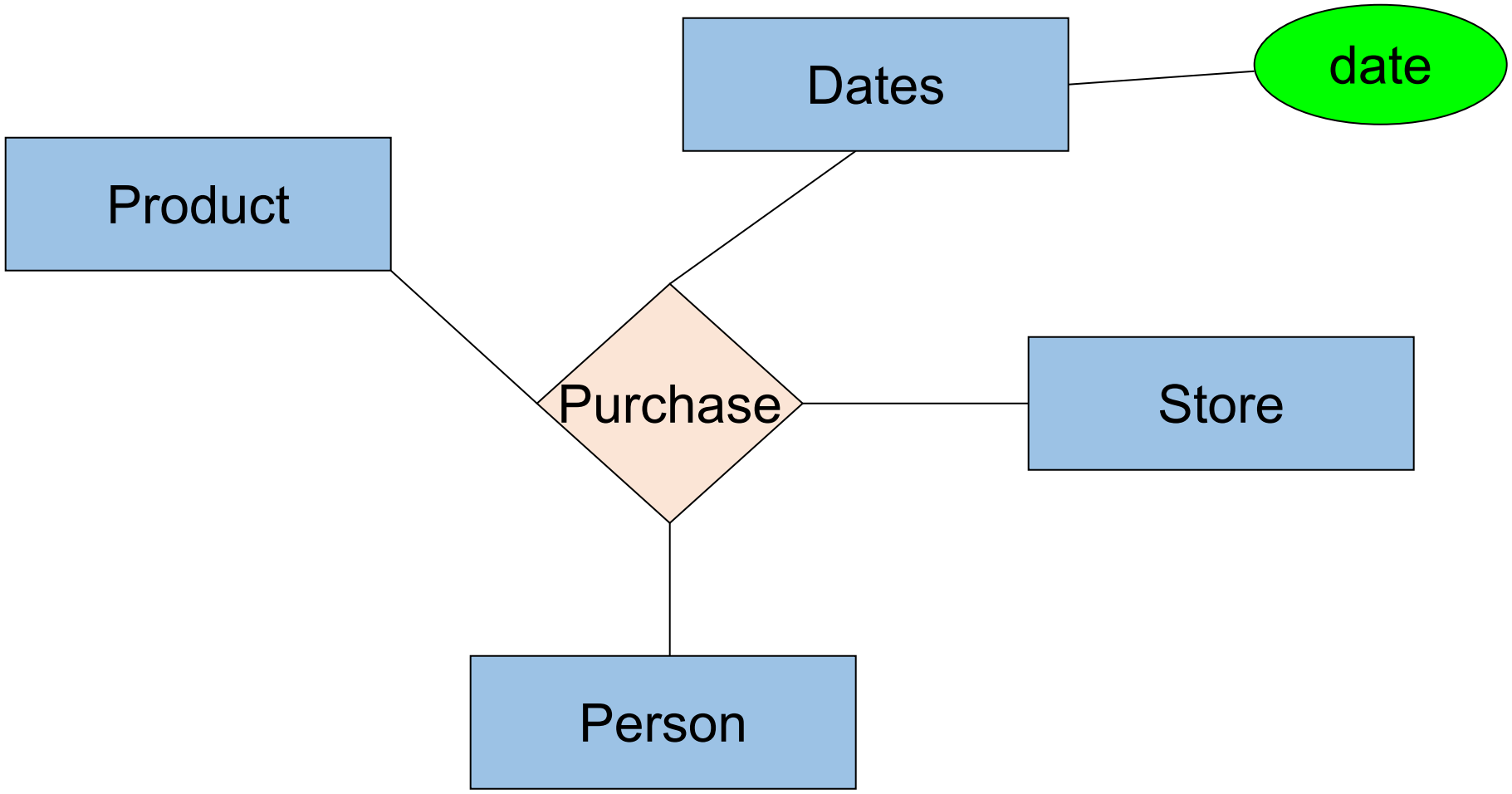


Design Practice: What's Wrong?

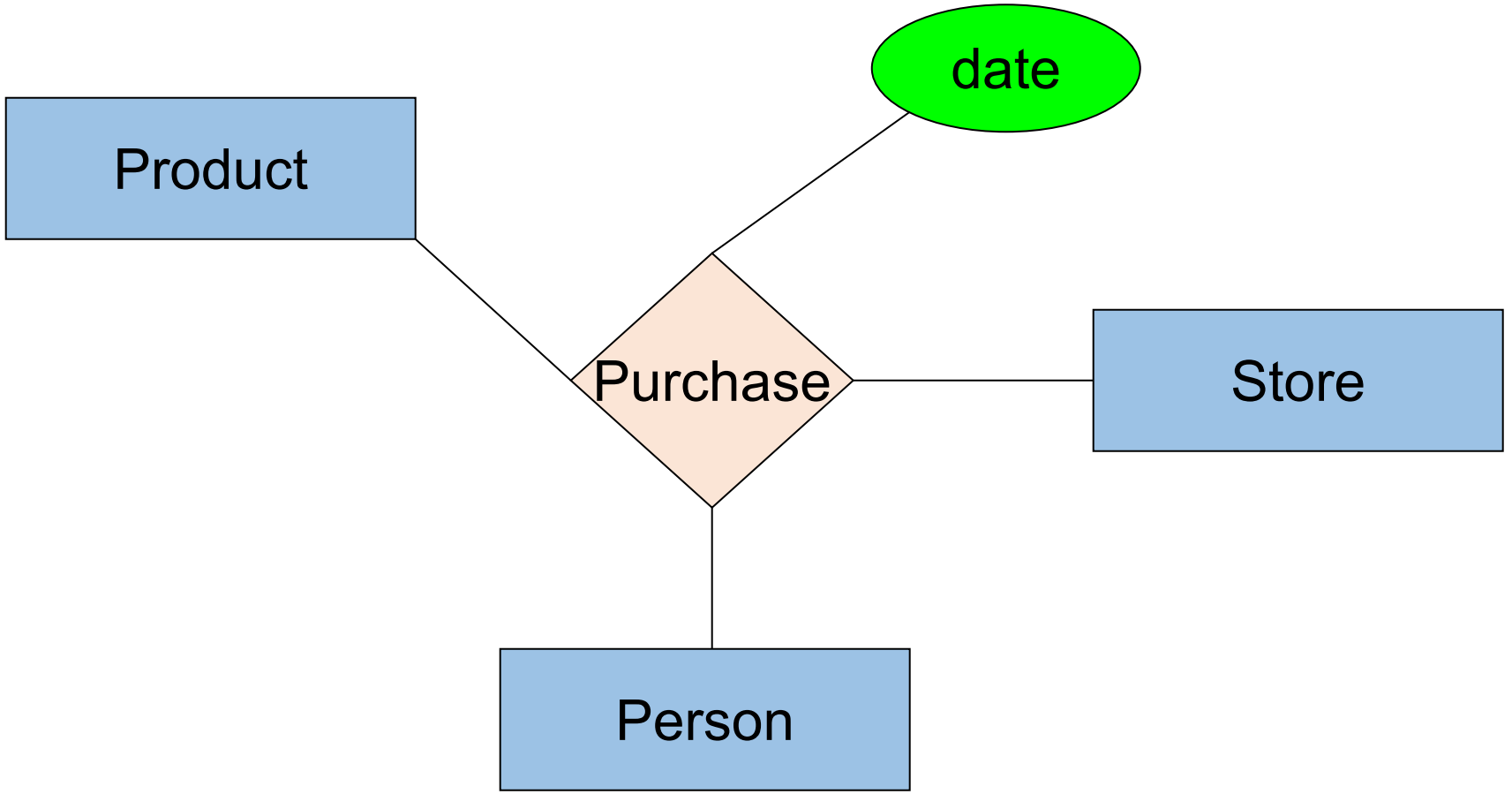


Design Practice: What's Wrong?

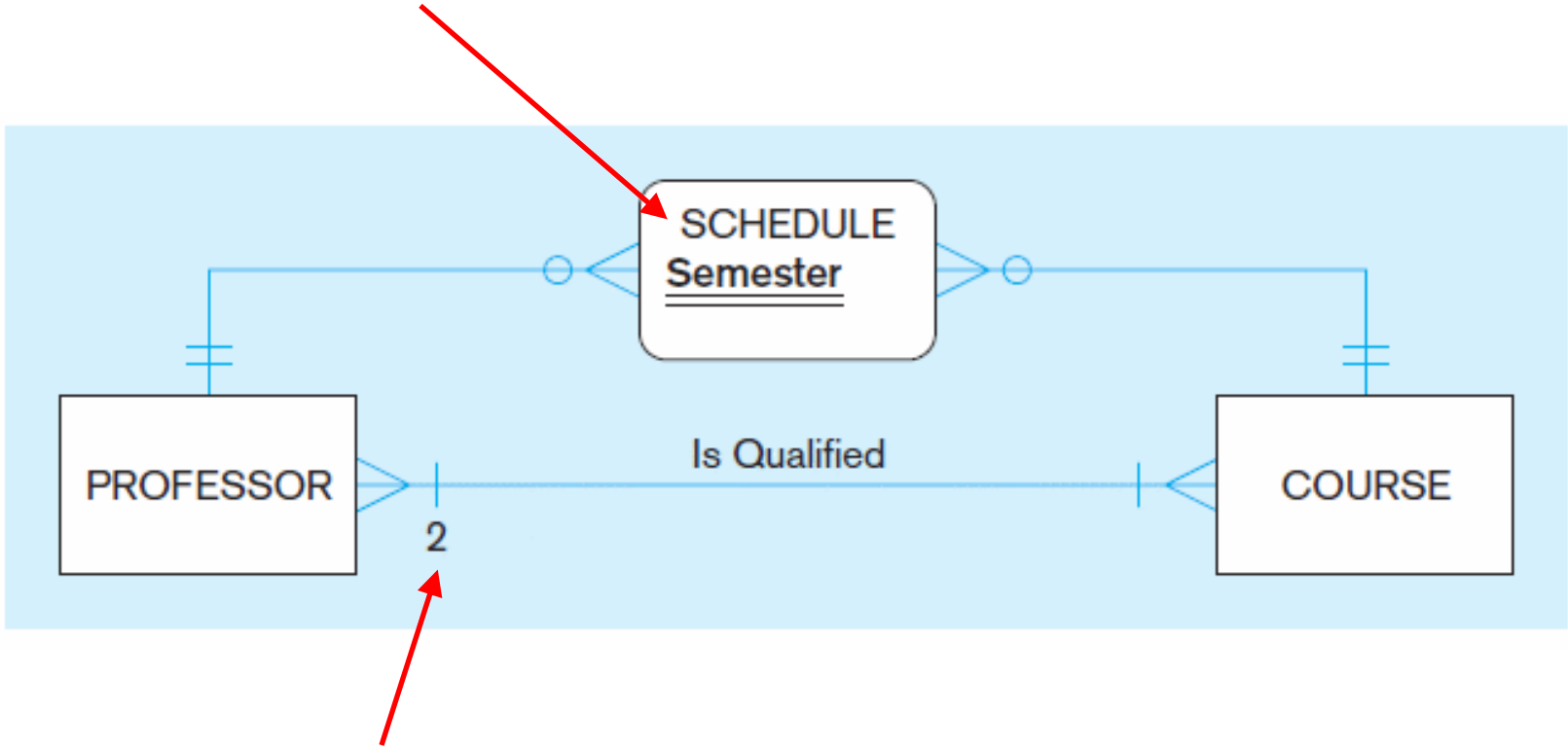
?



Design Practice: What's Wrong?



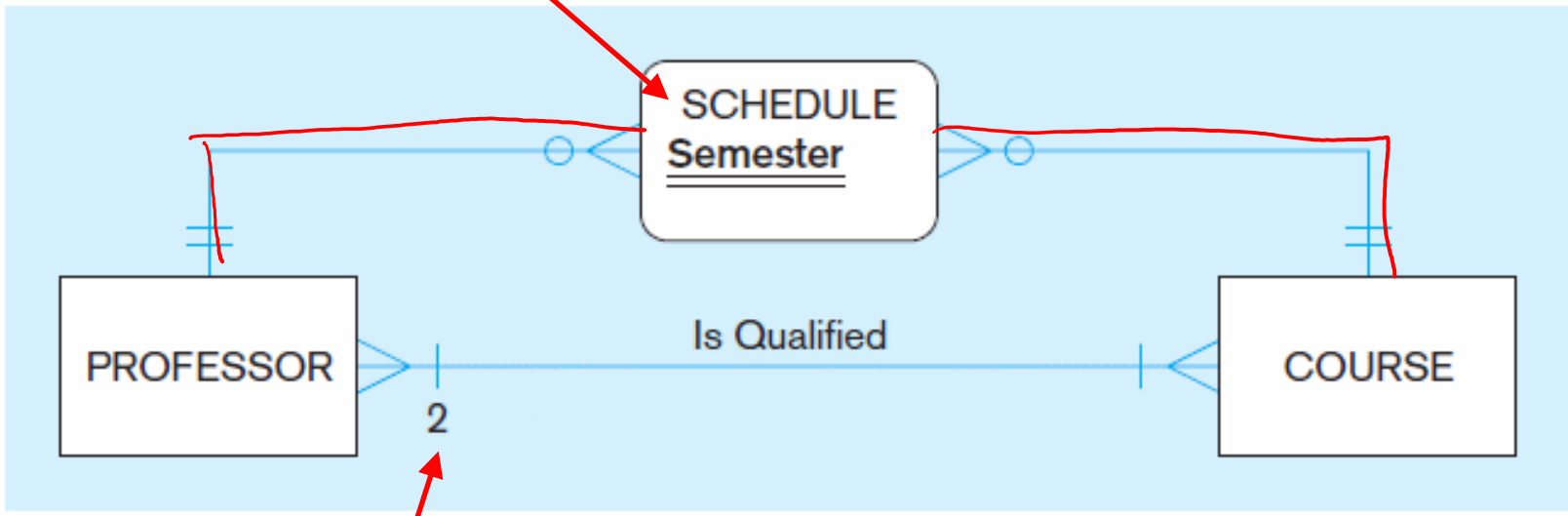
Multiple relationships



Source: Hoffer, Ramesh, Topi, "Modern database management," 10th ed, 2010. Fig 2.21.
Wolfgang Gatterbauer. Database design: https://northeastern-datalab.github.io/cs3200/

Multiple relationships

Associative entity: Surrogate identifier would loose the business rule that the combination of the PROFESSOR identifier, COURSE identifier, and Semester must be unique for each SCHEDULE instance

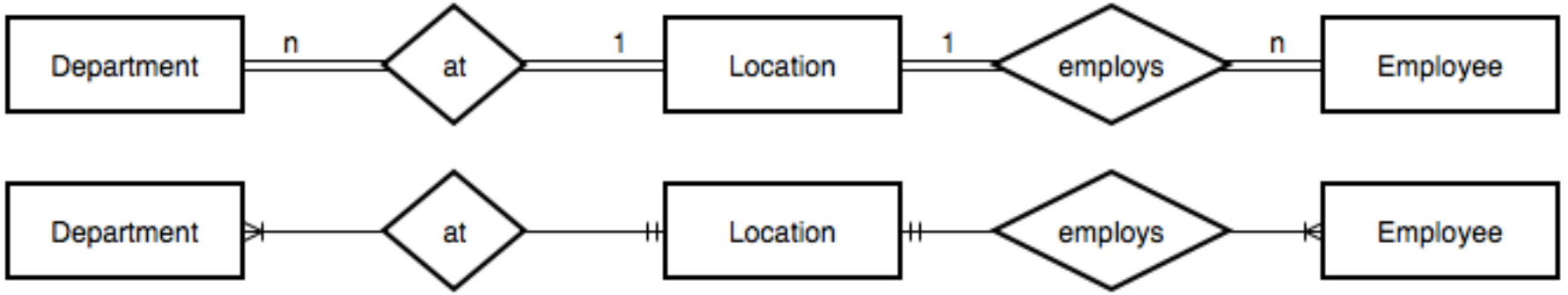


Here, min cardinality constraint is 2: At least two professors must be qualified to teach each course

Source: Hoffer, Ramesh, Topi, "Modern database management," 10th ed, 2010. Fig 2.21.
Wolfgang Gatterbauer. Database design: https://northeastern-datalab.github.io/cs3200/

Connection Traps (join path problems)

We have a problem

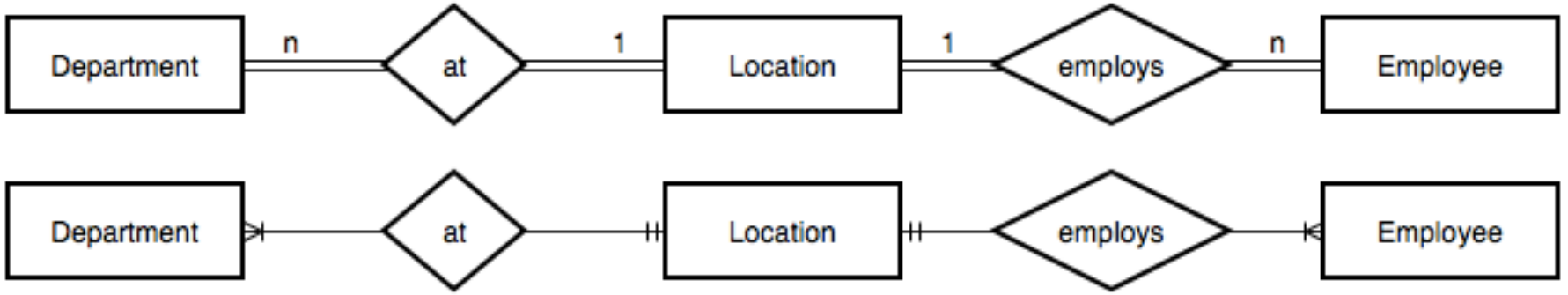


Where is the problem

?

Source: https://www.techtud.com/example/traps-entity-relationship-diagrams-fan-trap-and-chasm-trap
Wolfgang Gatterbauer. Database design: https://northeastern-datalab.github.io/cs3200/

We have a problem: a "fan trap"

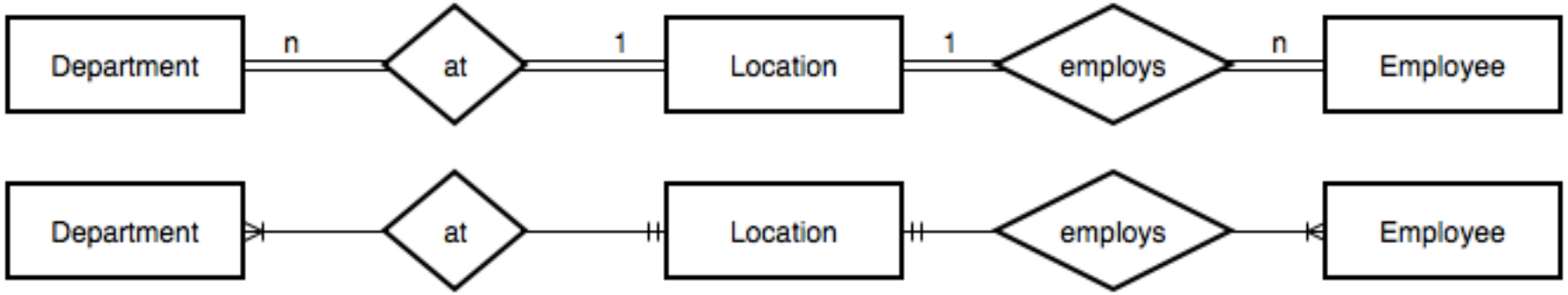


For which department does a particular employee work? ?

Source: https://www.techtud.com/example/traps-entity-relationship-diagrams-fan-trap-and-chasm-trap
Wolfgang Gatterbauer. Database design: https://northeastern-datalab.github.io/cs3200/

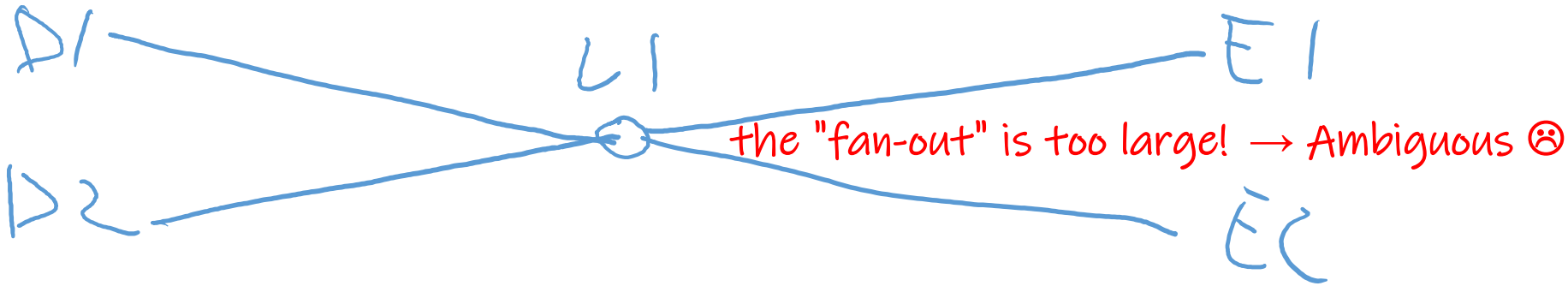
We have a problem: a "fan trap"

ER diagram (entity types)



For which department does a particular employee work? ?

- instances
- occurrences
- actual entities

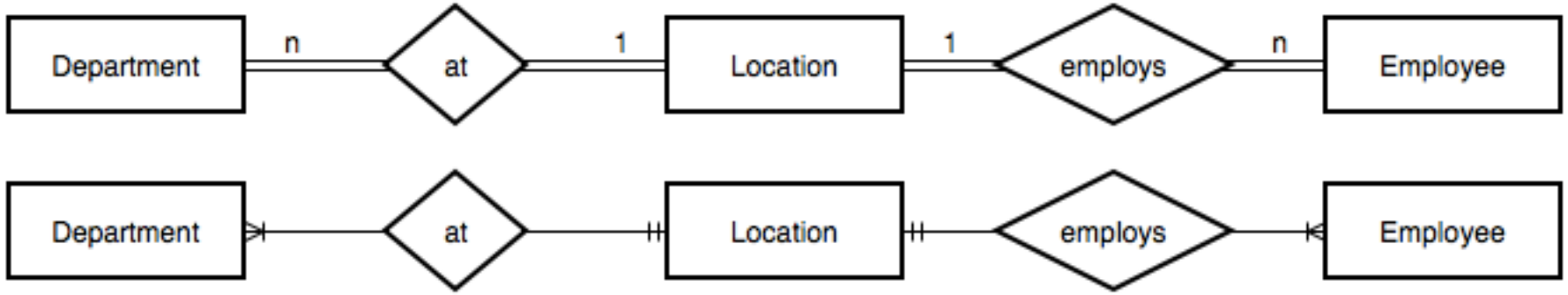


the "fan-out" is too large! → Ambiguous ☹ !

Source: https://www.techtud.com/example/traps-entity-relationship-diagrams-fan-trap-and-chasm-trap
Wolfgang Gatterbauer. Database design: https://northeastern-datalab.github.io/cs3200/

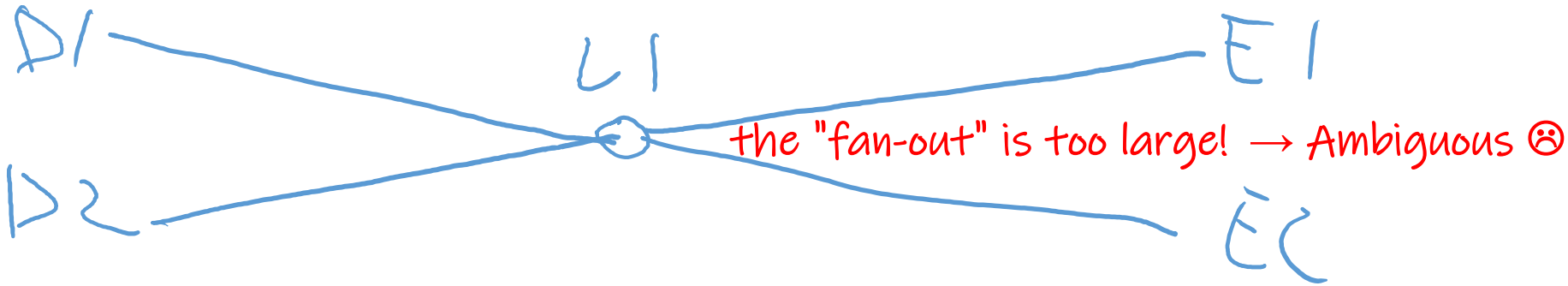
How to resolve a "fan trap"

ER diagram (entity types)



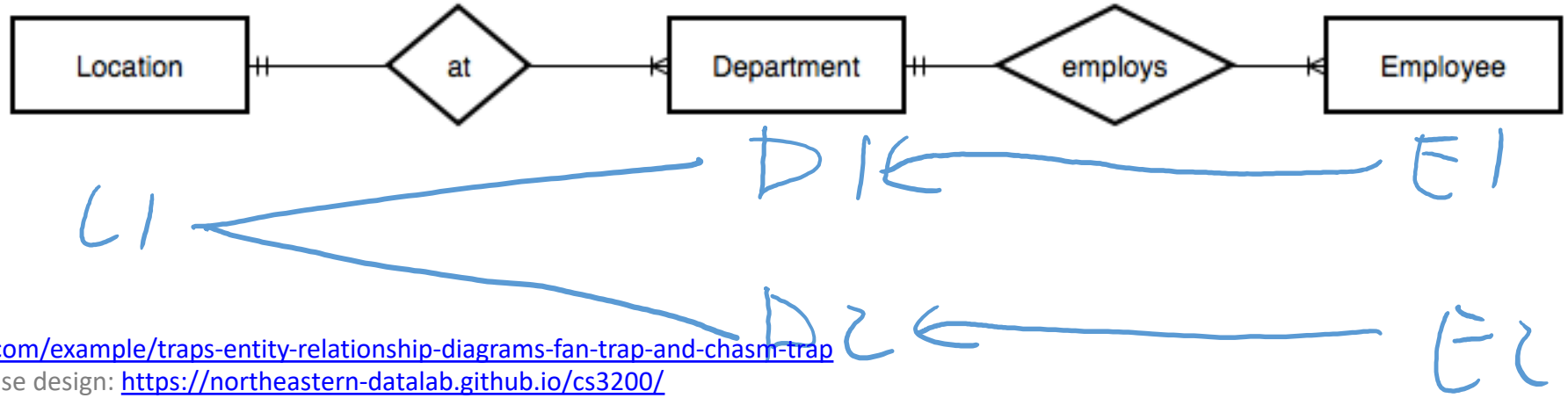
For which department does a particular employee work? ?

- instances
- occurrences
- actual entities



the "fan-out" is too large! → Ambiguous ☹ !

Solution: Flip from N:1, 1:N to 1:N, 1:n



Source: https://www.techtud.com/example/traps-entity-relationship-diagrams-fan-trap-and-chasm-trap
Wolfgang Gatterbauer. Database design: https://northeastern-datalab.github.io/cs3200/

Another fan trap, different notation

Which member of staff works at a particular branch?

A single division has one or more staff

A single division operates one or more branches

Source: Connolly, Begg: Database systems, 4th ed, p. 364, 2005.
Wolfgang Gatterbauer. Database design: https://northeastern-datalab.github.io/cs3200/

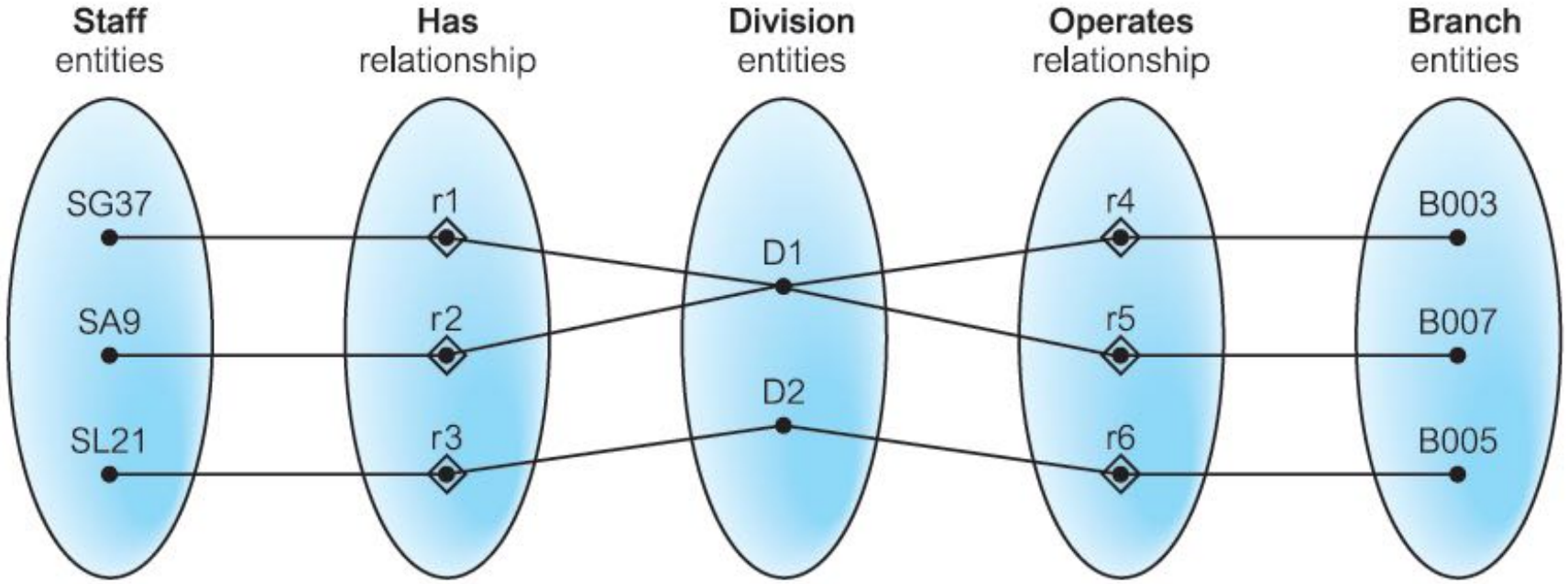
Another fan trap, different notation

Which member of staff works at a particular branch? ?

A single division has one or more staff

A single division operates one or more branches

Where does staff SG37 work? ?



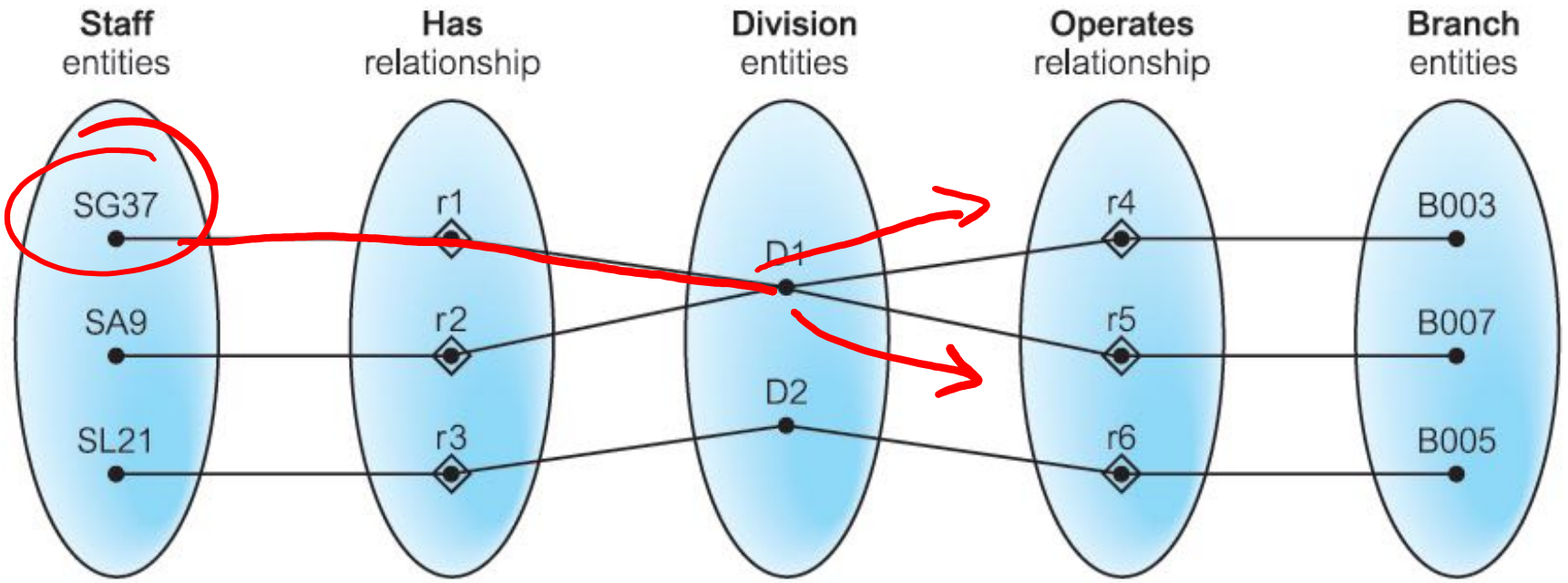
Another fan trap, different notation

Which member of staff works at a particular branch?

A single division has one or more staff

A single division operates one or more branches

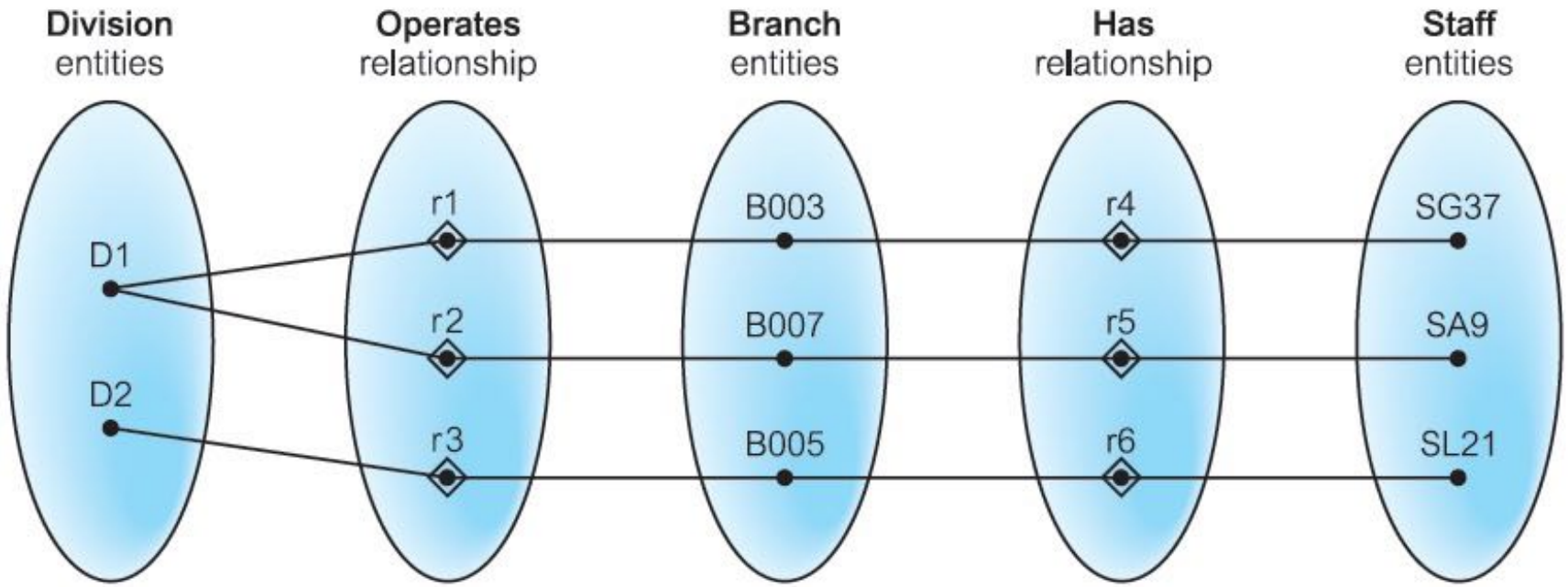
Where does staff SG37 work?



Fan Trap: Where a model represents a relationship between entity types, but the pathway between certain entity occurrences is ambiguous (**too many join paths**).

May exist when two or more 1:n relationships fan out from the same entity

Restructuring the model helps here again



Solution: here restructuring helped. More general solution: add a new relationship

Another problem with the "solved fan trap"

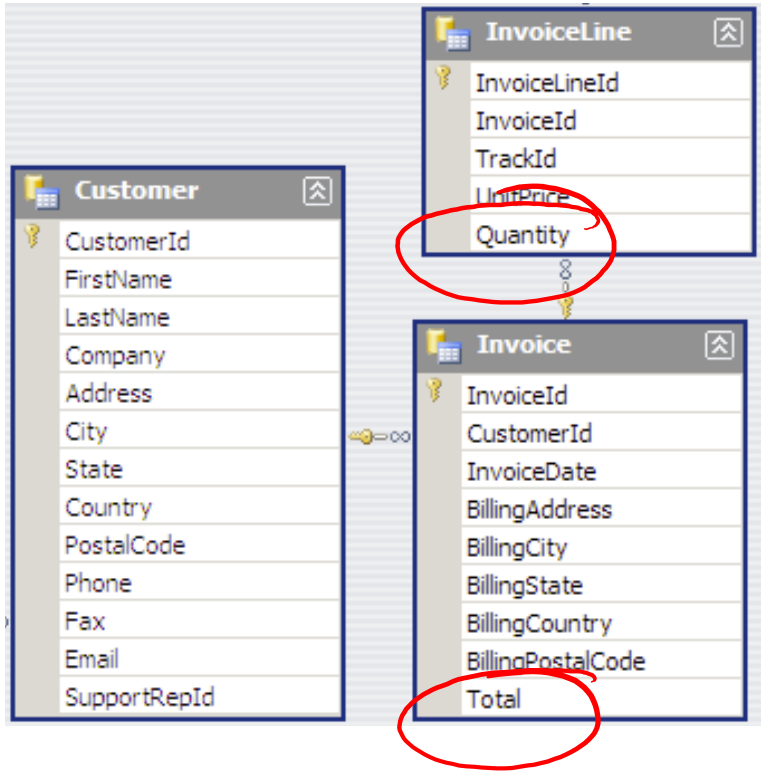
319

Chinook: For each customer with at least one purchase, sum the total and the quantities purchased.
[FirstName, LastName, sum_total, sum_quantity]

?

```
SELECT  C.firstname, C.lastname,
        sum(total) sum_total, sum(quantity) sum_quantity
FROM
```

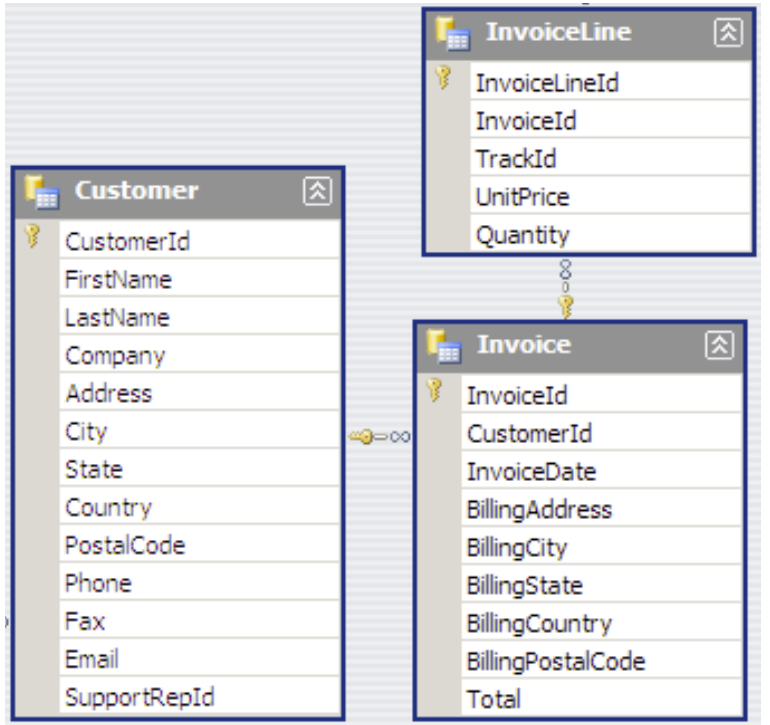
?



Another problem with the "solved fan trap"

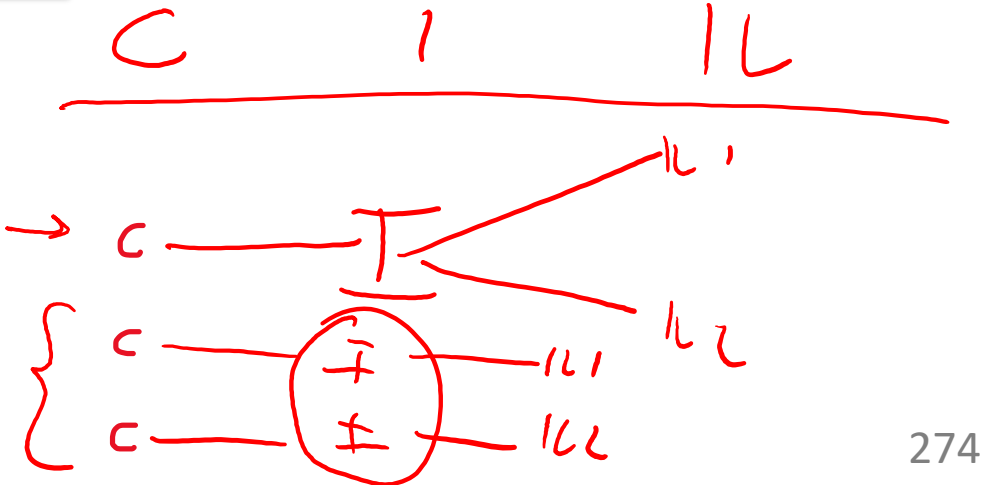
Chinook: For each customer with at least one purchase, sum the total and the quantities purchased.
[FirstName, LastName, sum_total, sum_quantity]

?




```
SELECT  C.firstname, C.lastname,
        sum(total) sum_total, sum(quantity) sum_quantity
FROM Customer C
JOIN Invoice I ON C.customerId=I.customerId
JOIN InvoiceLine IL ON I.invoiceId=IL.invoiceId
GROUP BY C.customerid, C.firstname, C.lastname
ORDER BY sum_total desc
```

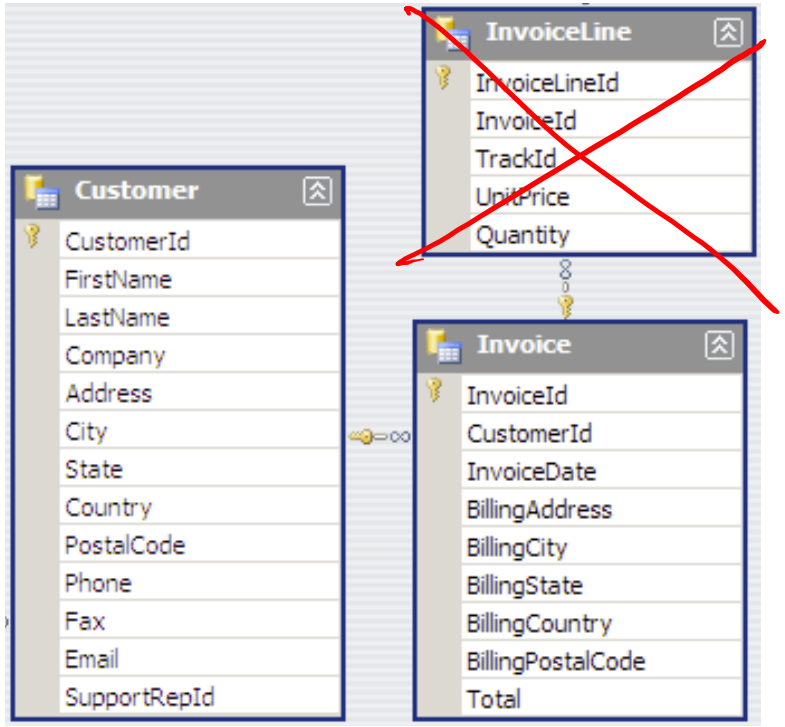
firstname	lastname	sum_total	sum_quantity
Helena	Holý	502.62	38
Richard	Cunningham	474.62	38
...	...	...	...



Another problem with the "solved fan trap"

Chinook: For each customer with at least one purchase, sum the total and the quantities purchased.
[FirstName, LastName, sum_total, sum_quantity]

?



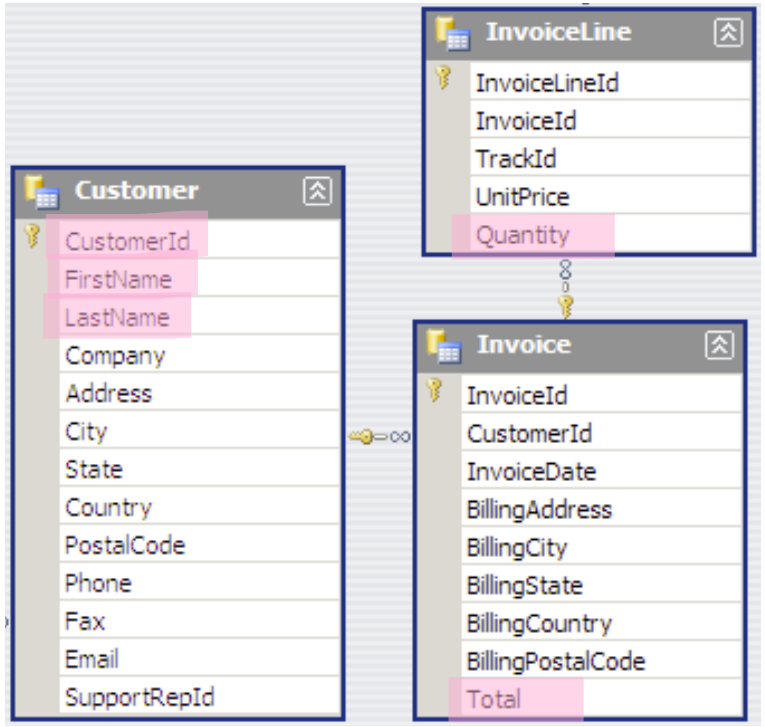

```
SELECT  C.firstname, C.lastname,
        sum(total) sum_total, sum(quantity) sum_quantity
FROM Customer C
JOIN Invoice I ON C.customerId=I.customerId
JOIN InvoiceLine IL ON I.invoiceId=IL.invoiceId
GROUP BY C.customerid, C.firstname, C.lastname
ORDER BY sum_total desc
```

firstname	lastname	sum_total
Helena	Holý	49.62
Richard	Cunningham	47.62
...	...	...

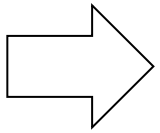
Another problem with the "solved fan trap"

Chinook: For each customer with at least one purchase, sum the total and the quantities purchased.
[FirstName, LastName, sum_total, sum_quantity]

!




```
SELECT    C.firstname, C.lastname, sum_total, sum_quantity
FROM Customer C
JOIN   (SELECT customerid, sum(total) sum_total
        FROM  Invoice I
        GROUP BY customerid) X on C.customerid=X.customerid
JOIN   (SELECT customerid, sum(quantity) sum_quantity
        FROM  Invoice I JOIN InvoiceLine IL
        ON I.invoiceId=IL.invoiceId
        GROUP BY customerid) Y on C.customerid=Y.customerid
ORDER BY sum_total desc
```



firstname	lastname	sum_total	sum_quantity
Helena	Holý	49.62	38
Richard	Cunningham	47.62	38
...	...	...	...

Careful when you aggregate over several different tables!

We have yet another problem

Now we seem to have avoided a fan trap. Do we still have a problem? ?

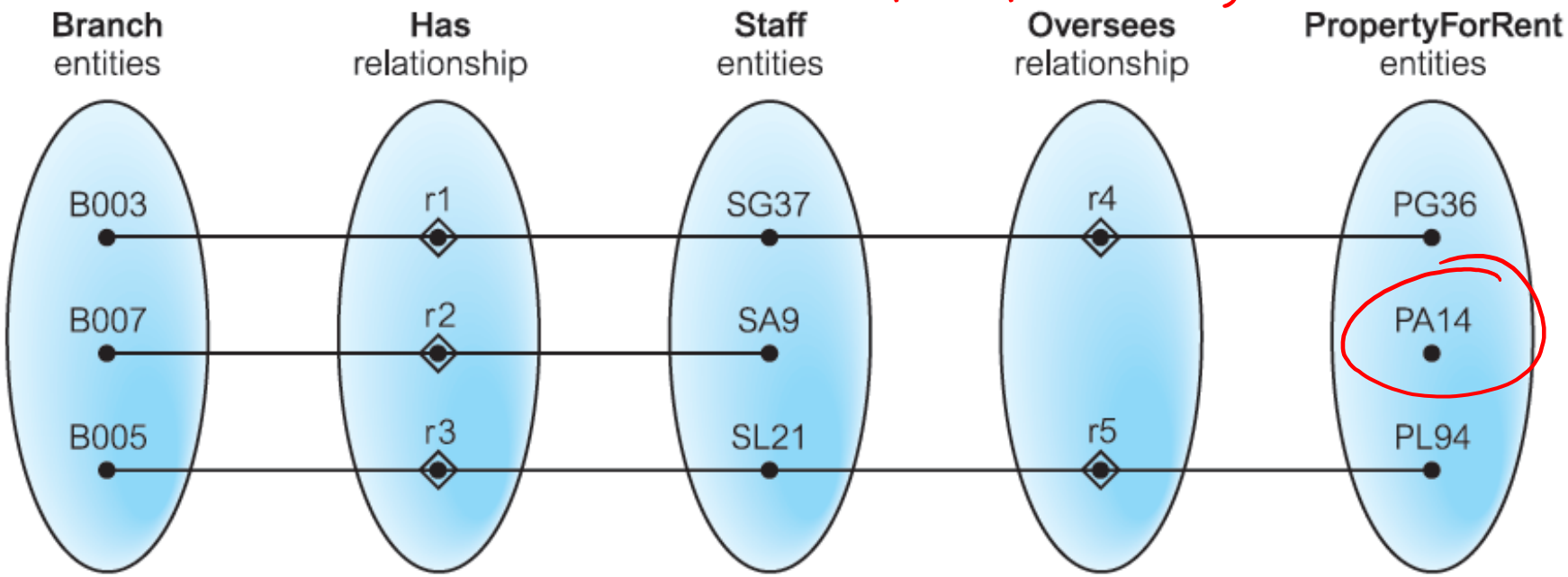
Yet another problem: "chasm traps"

A property managed at a branch may not yet be assigned to a staff

Now we seem to have avoided a fan trap. Do we still have a problem? ?

Yet another problem: "chasm traps"

A property managed at a branch may not yet be assigned to a staff

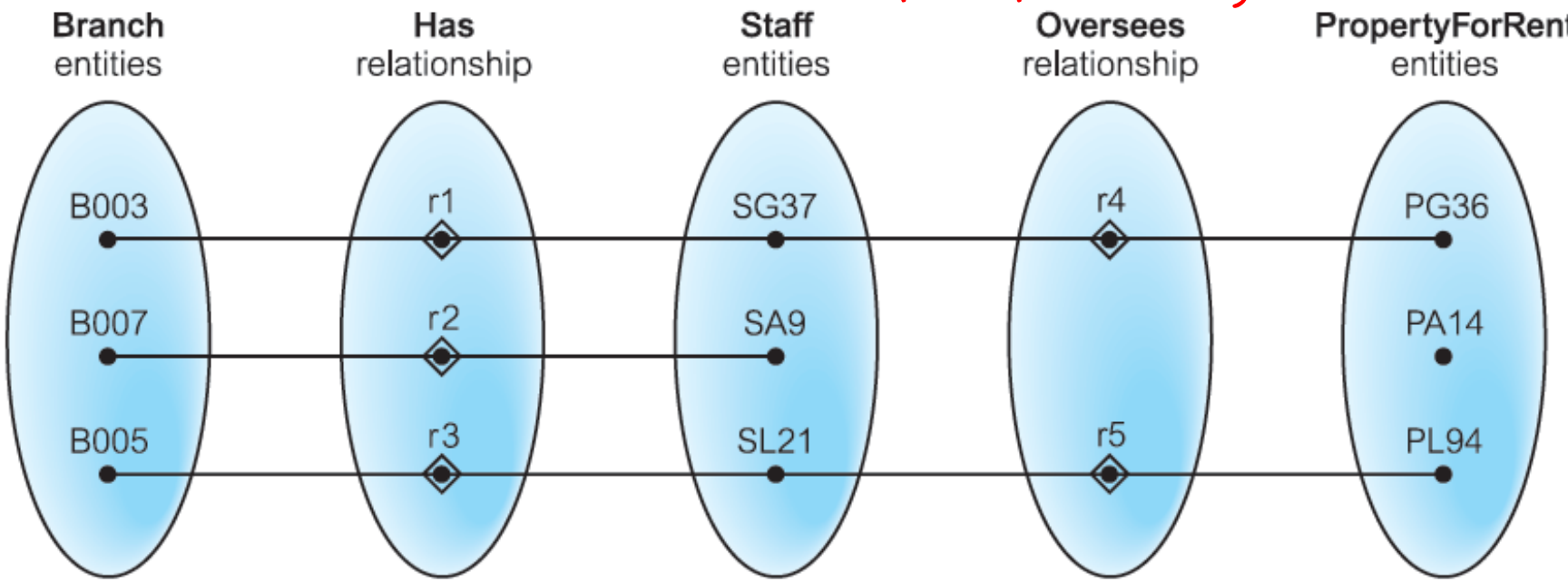


Where is property PA14 available? ?

Source: Connolly, Begg: Database systems, 4th ed, p. 364, 2005.
Wolfgang Gatterbauer. Database design: https://northeastern-datalab.github.io/cs3200/

Yet another problem: "chasm traps"

A property managed at a branch may not yet be assigned to a staff

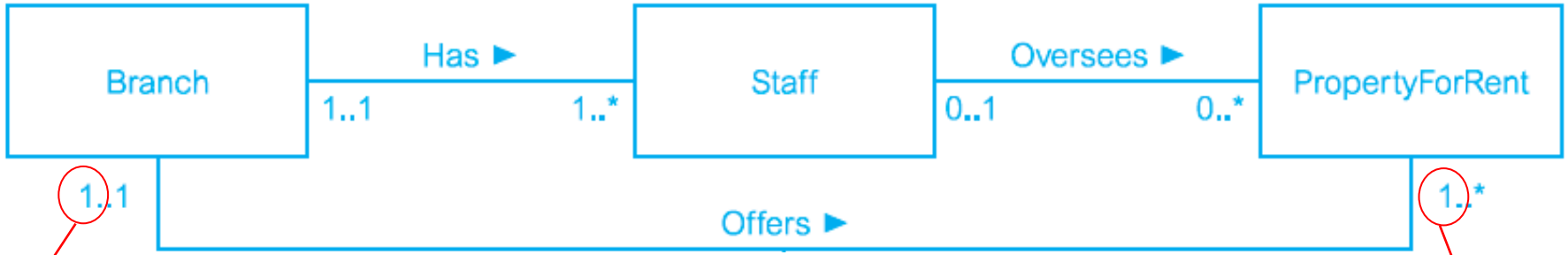


Where is property PA14 available? ?

Chasm Trap: Where a model suggests the existence of a relationship between entity types, but the pathway does not exist between certain entity occurrences (**a join path is lost**).
May exist when there is a relationship with optional participation between the related entities.

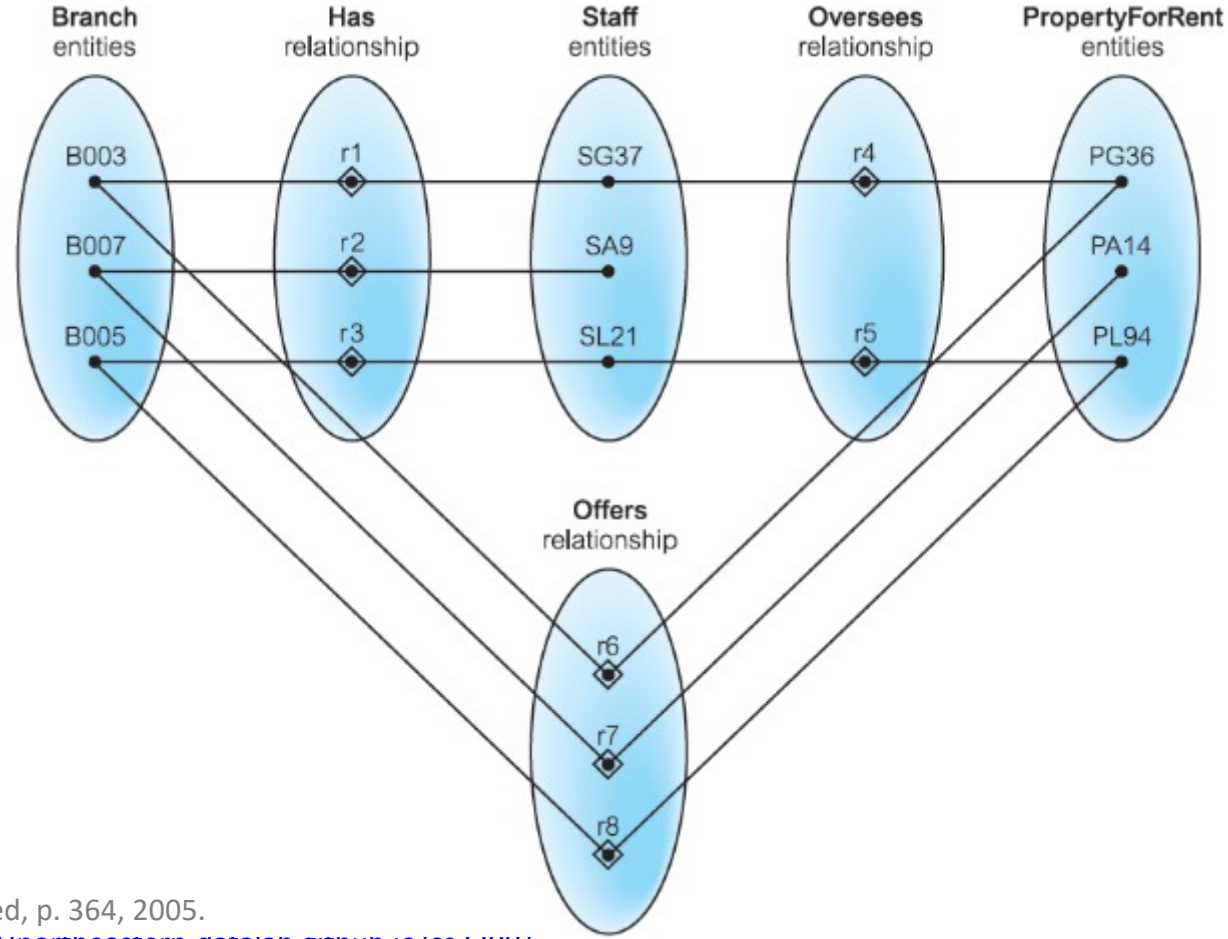
Source: Connolly, Begg: Database systems, 4th ed, p. 364, 2005.
Wolfgang Gatterbauer. Database design: https://northeastern-datalab.github.io/cs3200/

Adding a relationship helps here



An assignment to a branch is required (mandatory).

"0..*" is also possible, and actually preferred

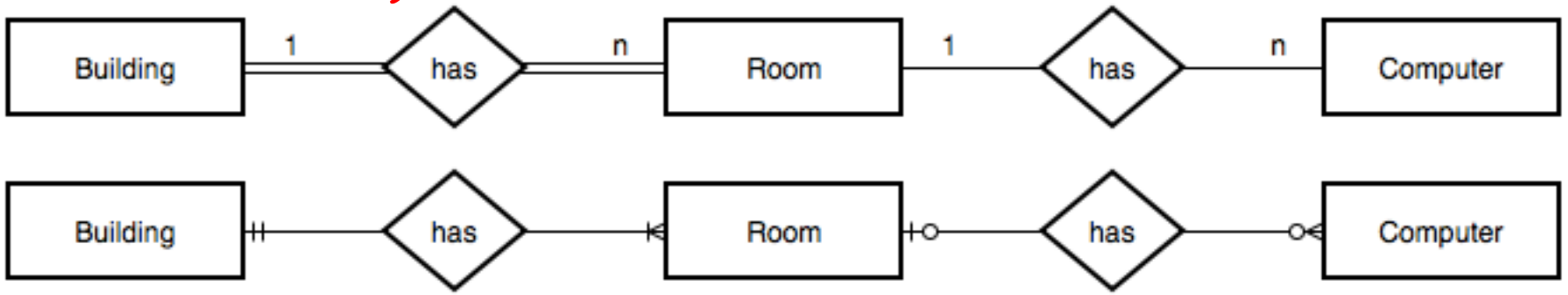


Source: Connolly, Begg: Database systems, 4th ed, p. 364, 2005.
Wolfgang Gatterbauer. Database design: https://northeastern-datalab.github.io/cs3200/

Chasm Trap

In which building are all computers, even those outside the rooms (e.g., in the reception counter)?

?

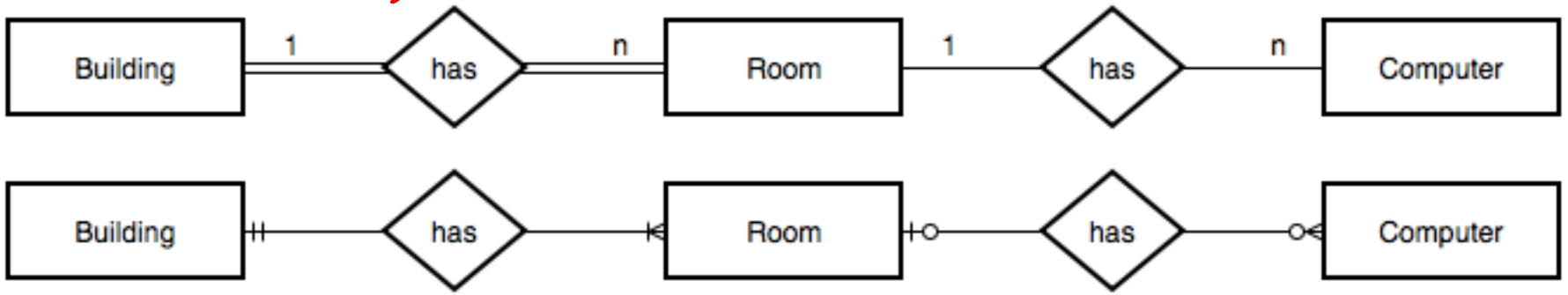


Source: https://www.techtud.com/example/traps-entity-relationship-diagrams-fan-trap-and-chasm-trap
Wolfgang Gatterbauer. Database design: https://northeastern-datalab.github.io/cs3200/

Chasm Trap

In which building are all computers, even those outside the rooms (e.g., in the reception counter)?

?



→ Failure to capture all the relationships that exist in the real world in the model.

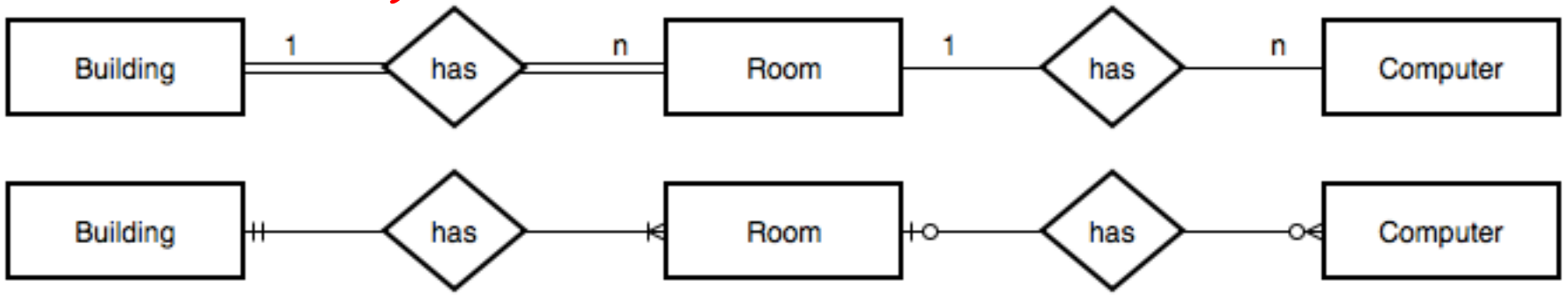
!

Source: https://www.techtud.com/example/traps-entity-relationship-diagrams-fan-trap-and-chasm-trap
Wolfgang Gatterbauer. Database design: https://northeastern-datalab.github.io/cs3200/

Chasm Trap

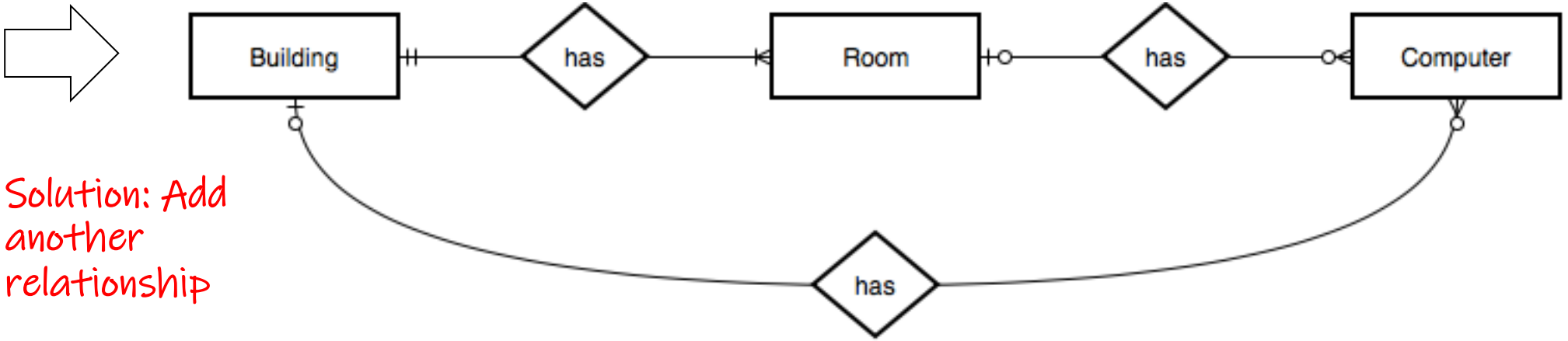
In which building are all computers, even those outside the rooms (e.g., in the reception counter)?

?



→ Failure to capture all the relationships that exist in the real world in the model.

!

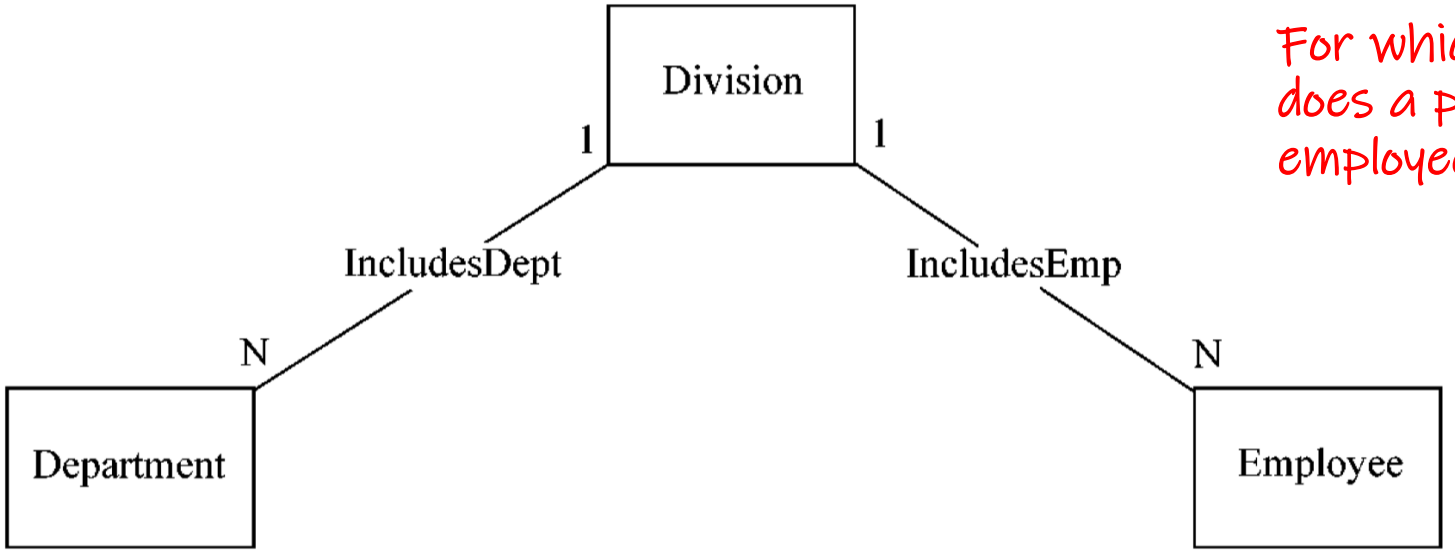


Solution: Add another relationship

Source: https://www.techtud.com/example/traps-entity-relationship-diagrams-fan-trap-and-chasm-trap
Wolfgang Gatterbauer. Database design: https://northeastern-datalab.github.io/cs3200/

Fan trap

ER diagram (entity types)

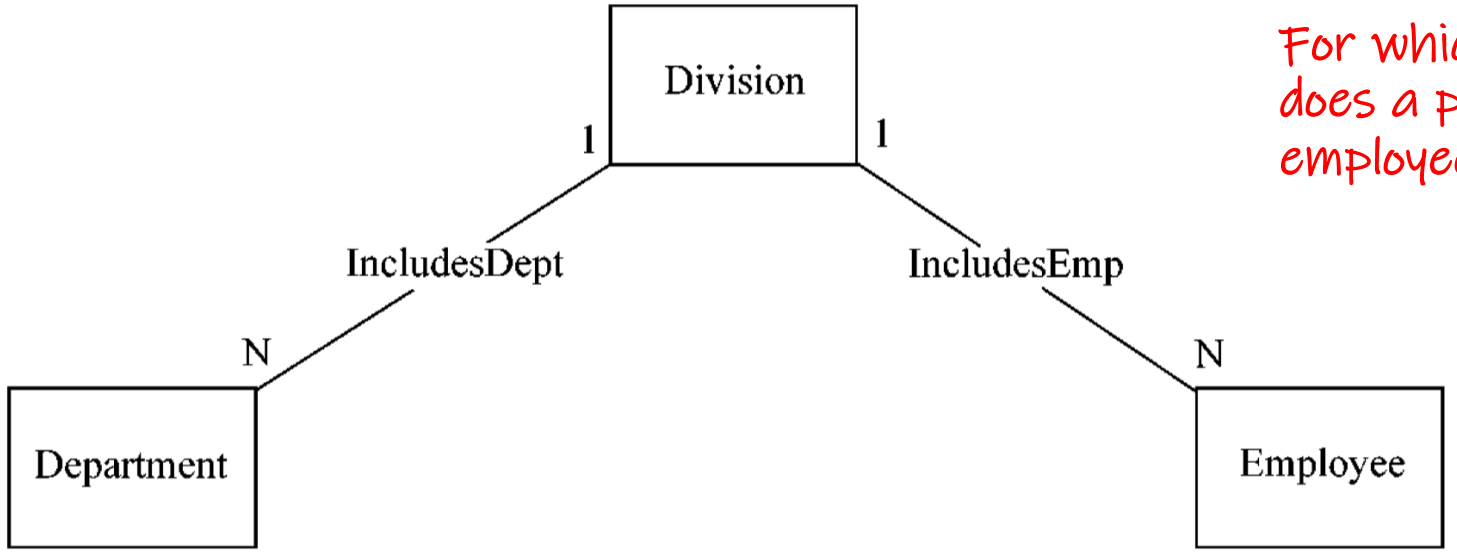


For which department does a particular employee work?

?

Fan trap

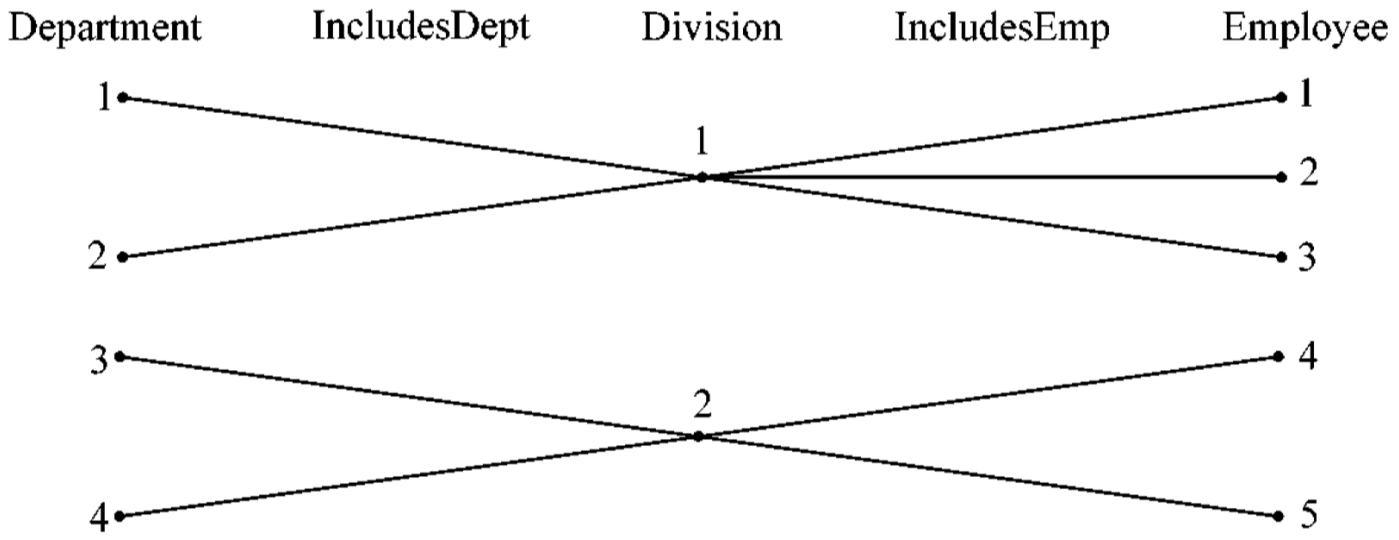
ER diagram (entity types)



For which department does a particular employee work?

?

- instances
- occurrences
- actual entities



Ambiguous ☹ !

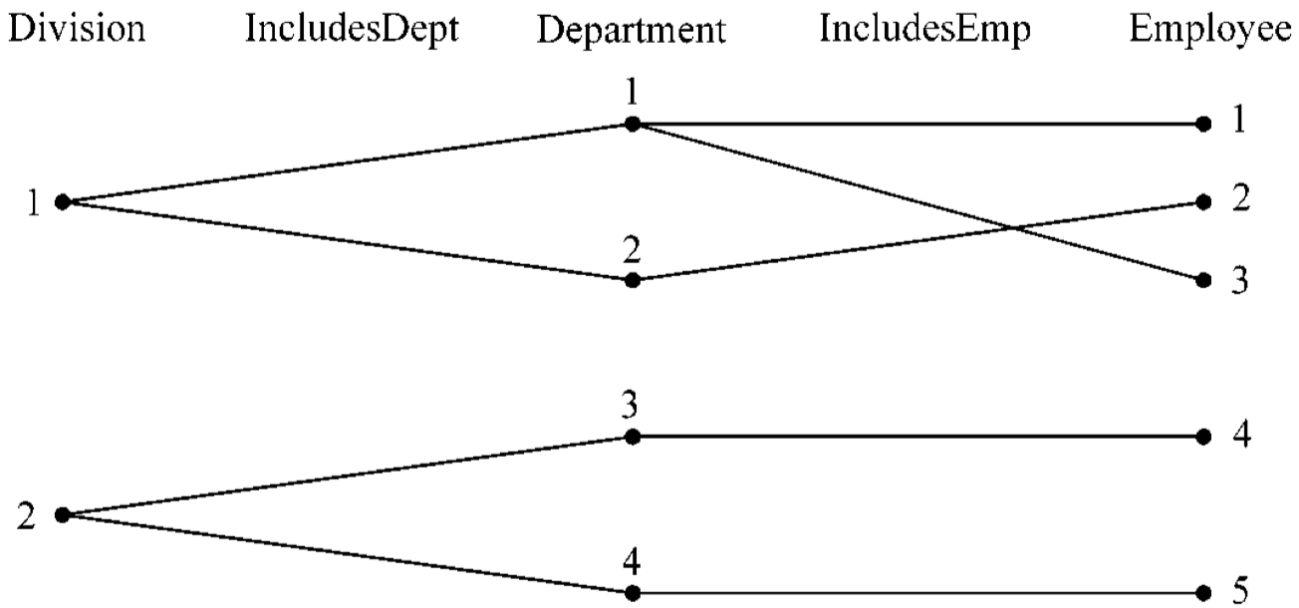
Source: Howe. Data Analysis for Database Design, 3rd ed, p. 117, 2001.
Wolfgang Gatterbauer. Database design: https://northeastern-datalab.github.io/cs3200/

Fan trap resolved

ER diagram (entity types)

Now relationships are unambiguous ☺ !

- instances
- occurrences
- actual entities



Source: Howe. Data Analysis for Database Design, 3rd ed, p. 117, 2001.
Wolfgang Gatterbauer. Database design: https://northeastern-datalab.github.io/cs3200/

Another resolution?

?

ER diagram (entity types)

Another resolution? Not really!

ER diagram (entity types)

Explain why the structure above contains a potential chasm trap. ?

Another resolution? Not really!

ER diagram (entity types)

Explain why the structure above contains a potential chasm trap.

?

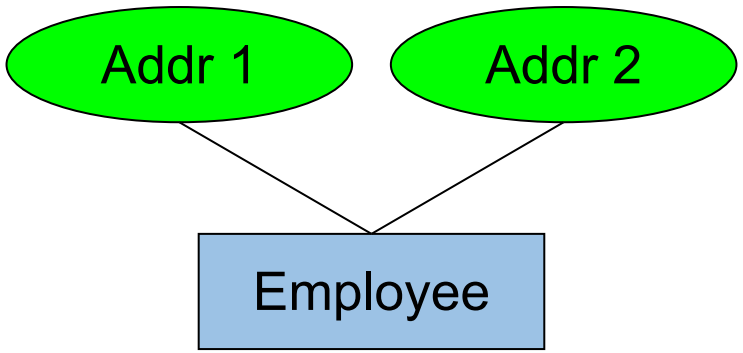
The association between a division and a department depends on there being at least one employee in the department.

Thus some instances of the 1:many relationship between division and department may be lost (may have fallen of "the chasm")

Examples: Entity vs. Attribute

How do we handle employees with multiple addresses?

Should address be an attribute? ?

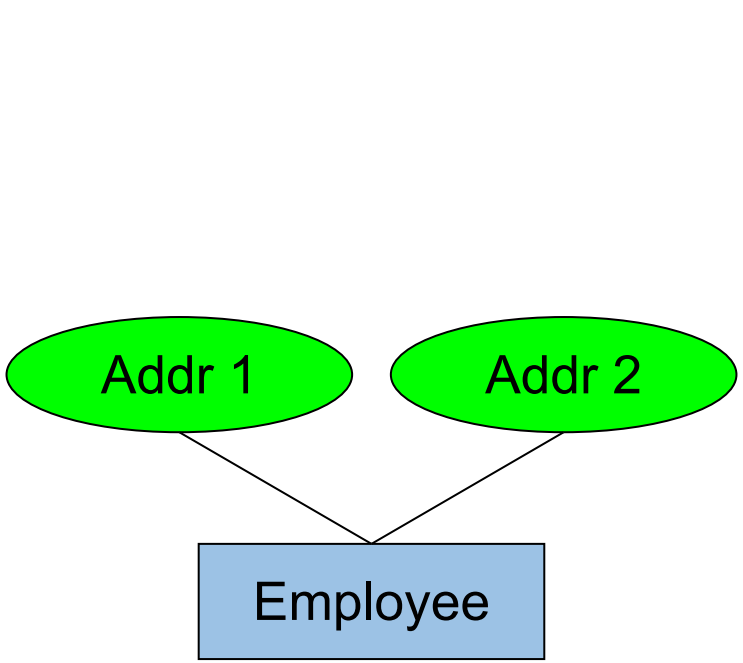


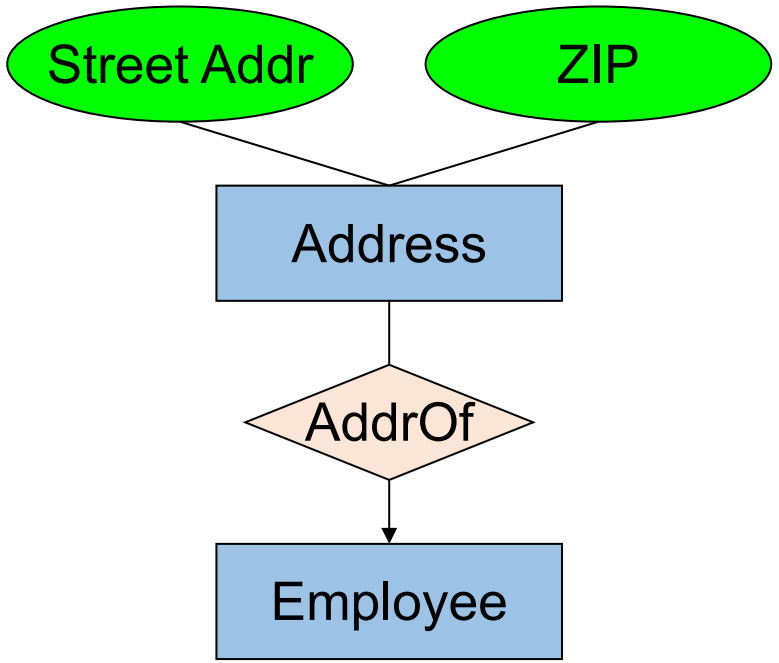
Examples: Entity vs. Attribute

How do we handle employees with multiple addresses?

Should address be an attribute? ?

Or be an entity? ?





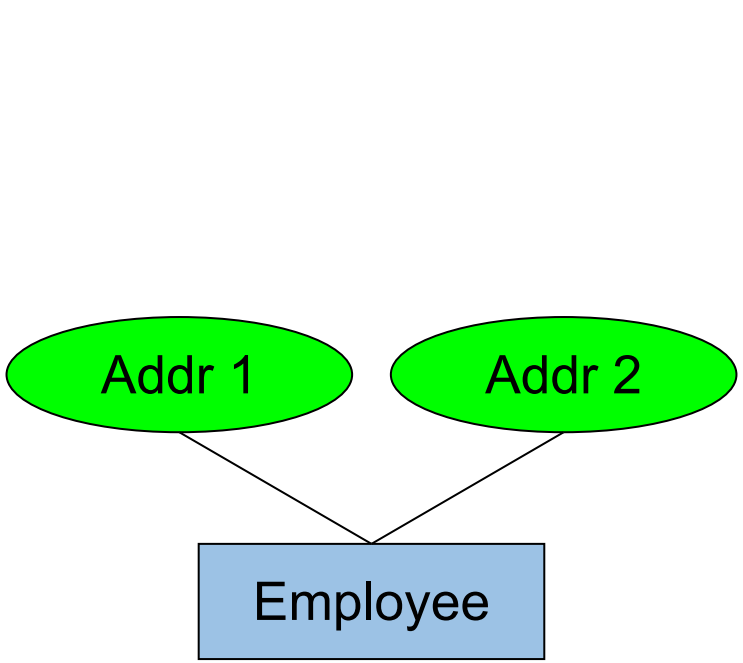
Examples: Entity vs. Attribute

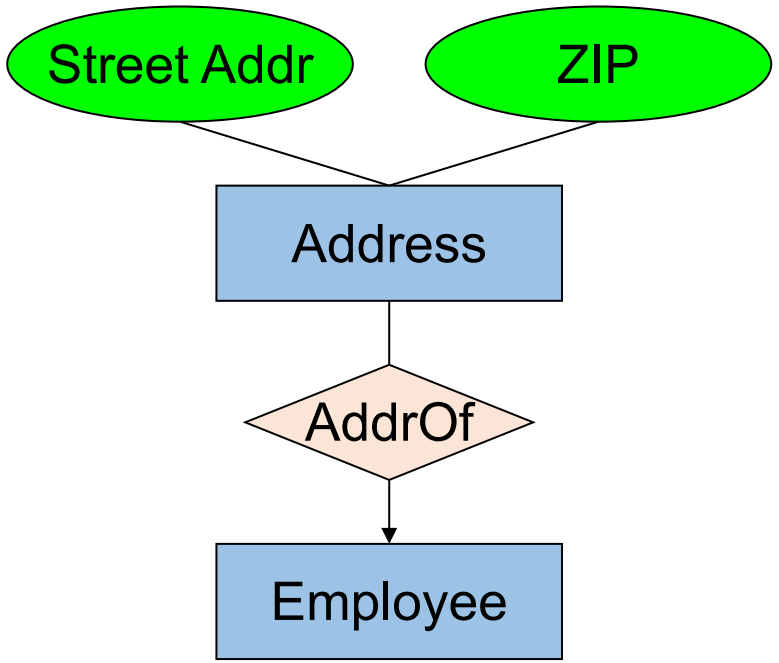
How do we handle employees with multiple addresses?

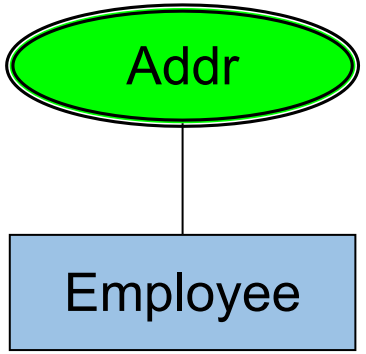
Should address be an attribute? ?

Or be an entity? ?

Or as a multivalued attribute! ?







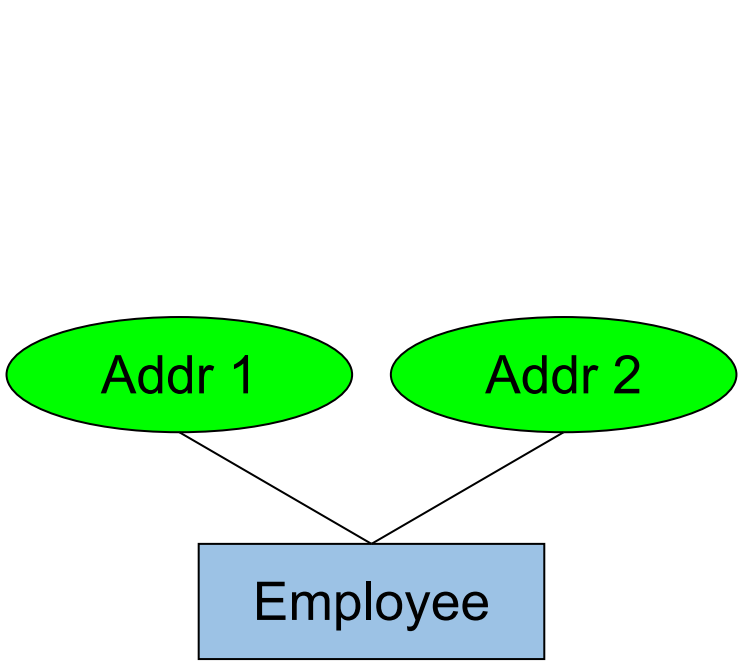
Examples: Entity vs. Attribute

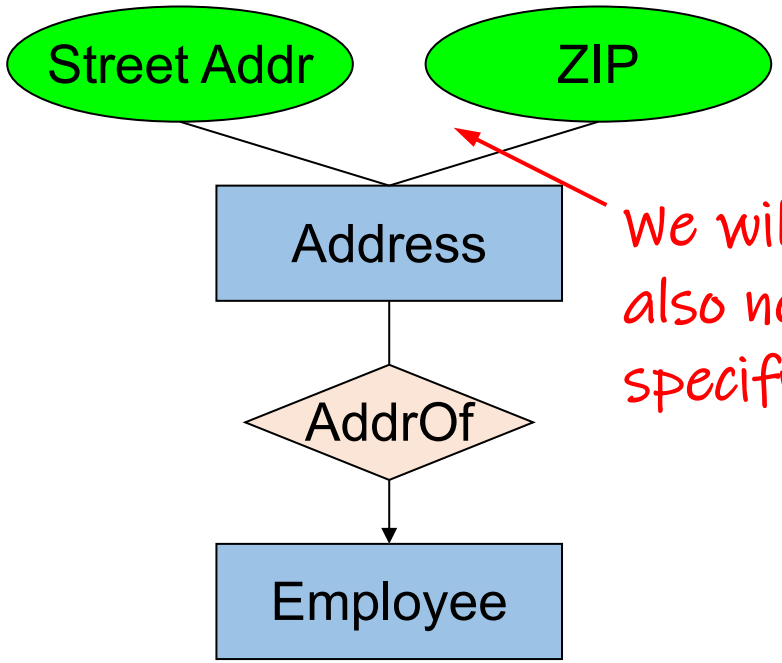
Preferred when internal structure of the address (e.g. zip code, state) is useful

Should address be an attribute? ?

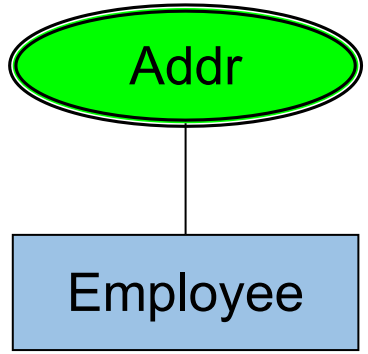
Or be an entity? ?

Or as a multivalued attribute! ?





We will then also need to specify a PK!



In general, when we want to record several values, we choose new entity or model as multivalued attributes!

Examples: Unary Degree Relationship

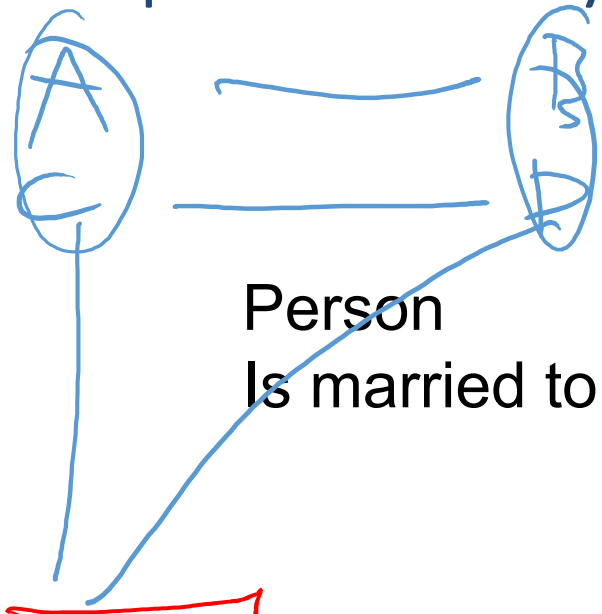
?

Person
Is married to

Managers
manage other
employees

Team
Stands After

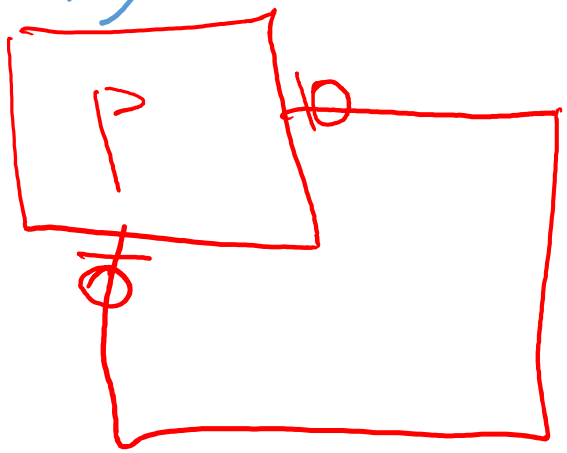
Examples: Unary Degree Relationship

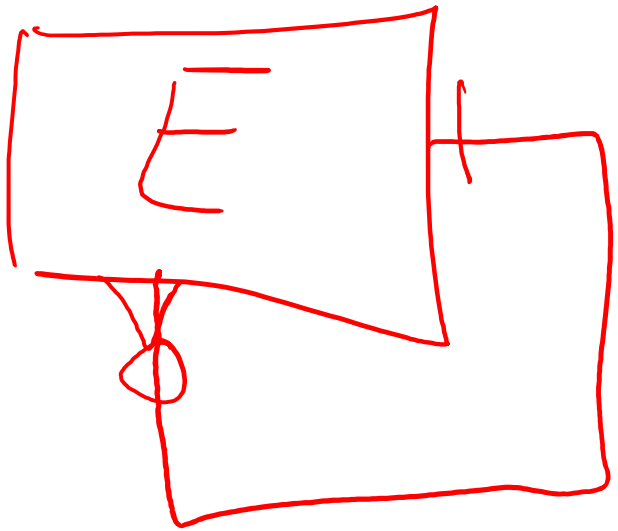


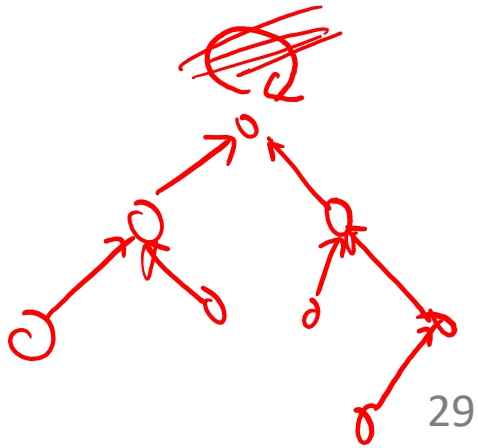
Person
Is married to

Managers
manage other
employees

Team
Stands After







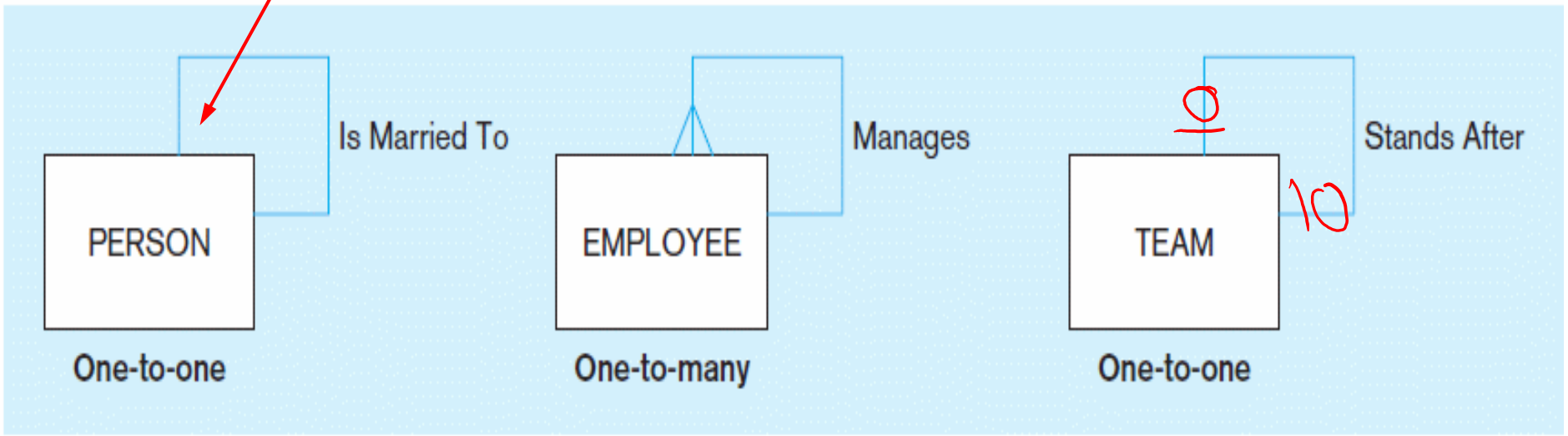
Examples: Unary Degree Relationship

Notice that this notation is ambiguous: according to our textbook, that would be many-to-many (no arrow = no constraint). Here is meant 1:1

Person
Is married to

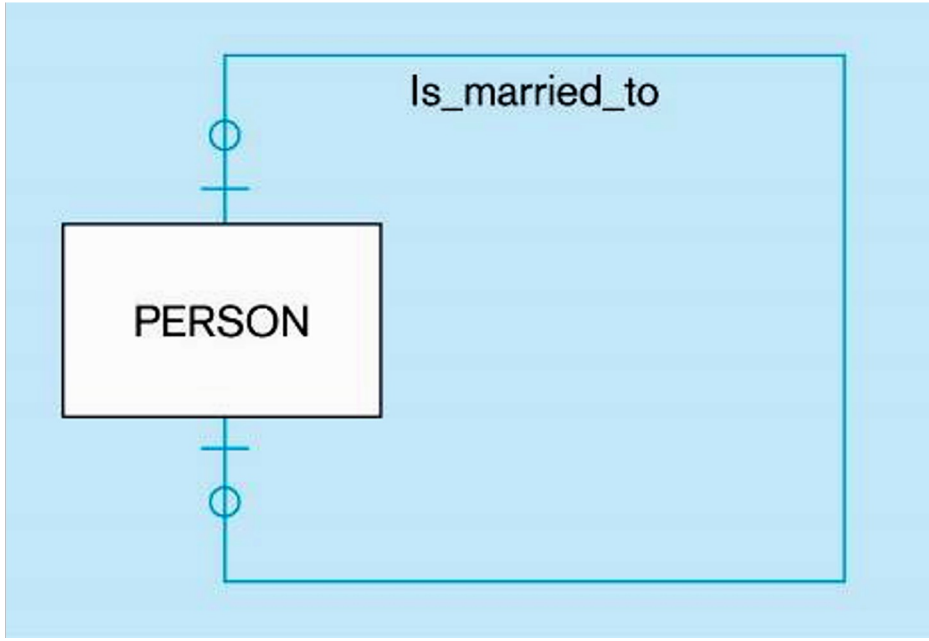
Managers
manage other
employees

Team
Stands After

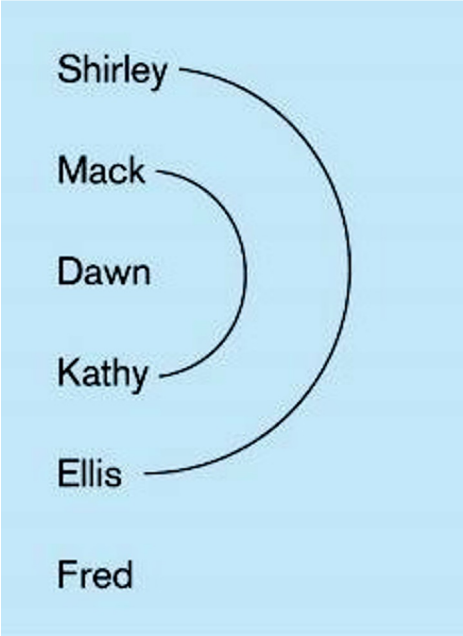


Example: Married to with participation

type (or entity set)

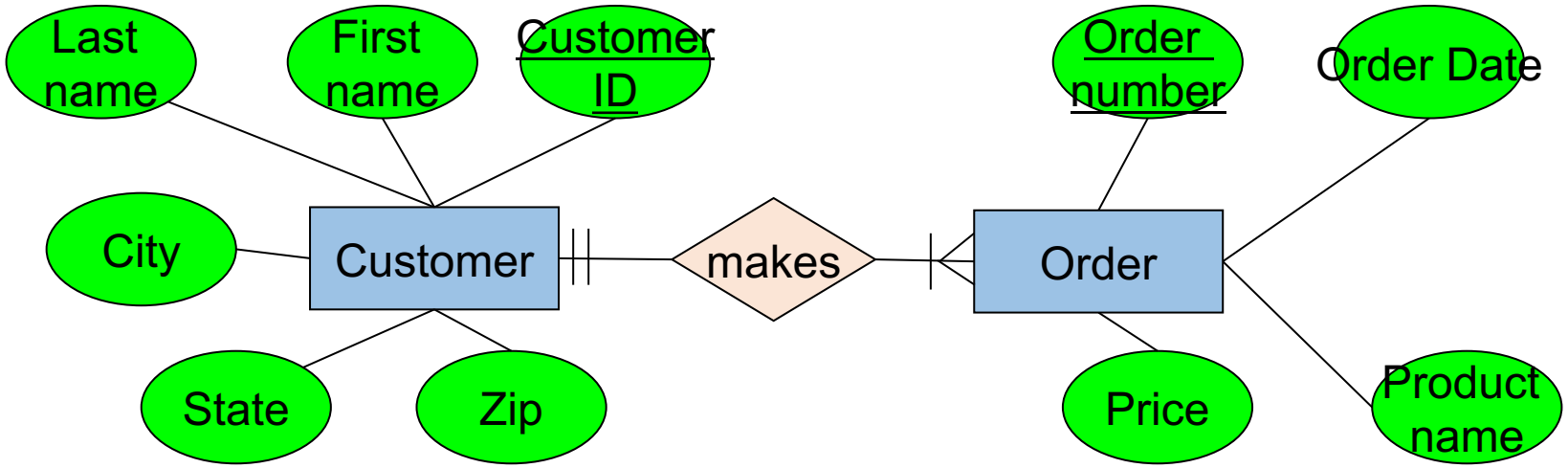


instance (or entity)



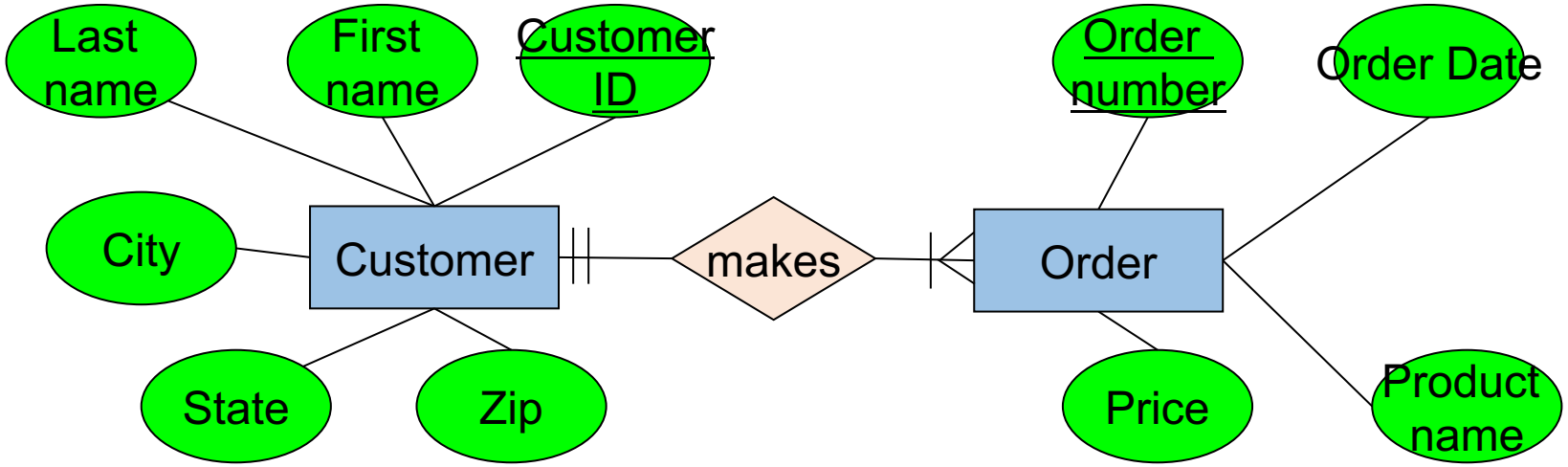
Source: Hoffer, Ramesh, Topi, "Modern database management," 10th ed, 2010.
Wolfgang Gatterbauer. Database design: https://northeastern-datalab.github.io/cs3200/

There is a problem with our ERD



?

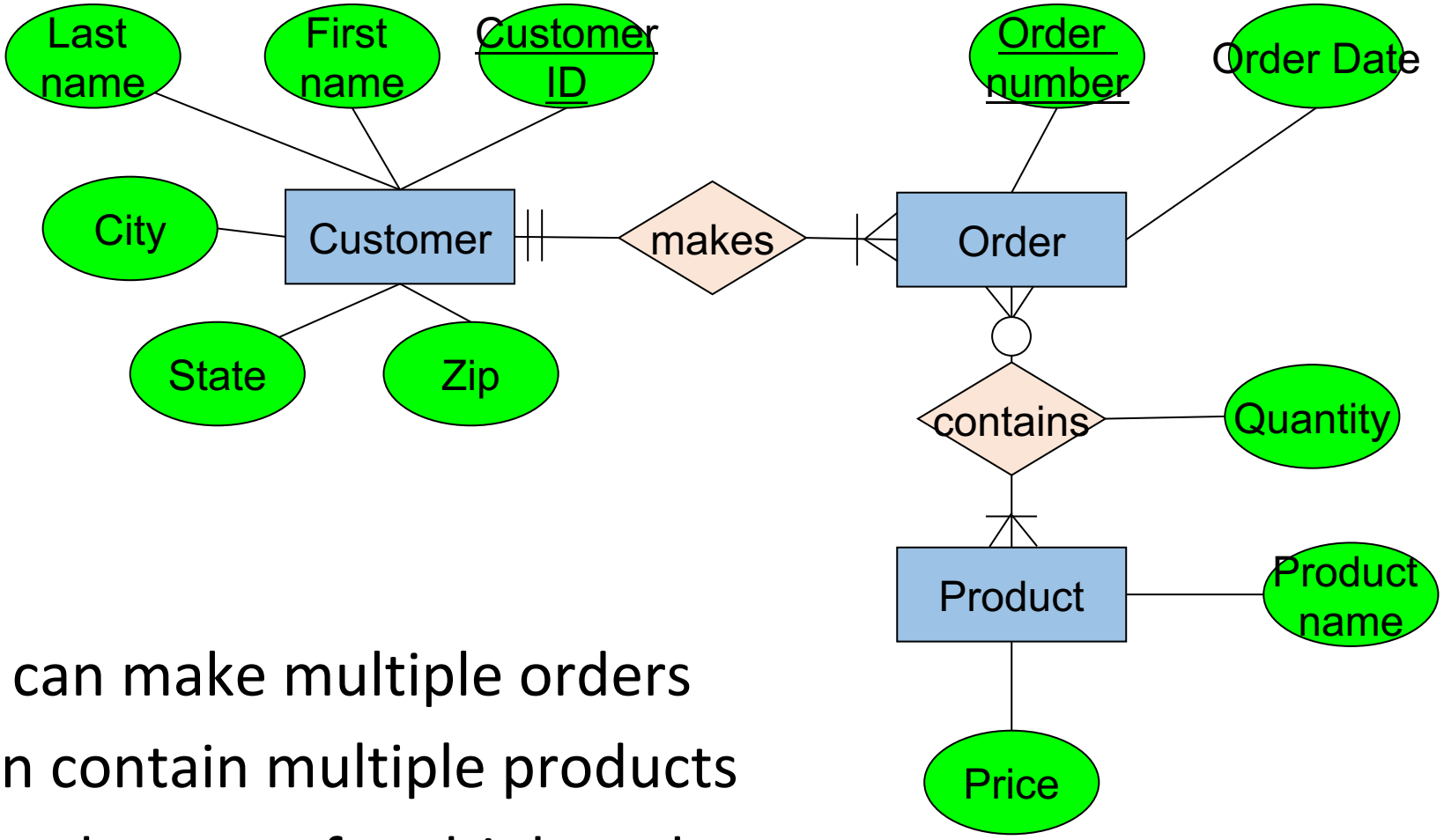
There is a problem with our ERD



This assumes every order contains only one product.
So if I want two products, I have to make two orders!

The problem: Product is defined as an attribute, not an entity.
(Because we didn't define our requirements clearly enough?)

Here is a solution



- Now
  - A customer can make multiple orders
  - An order can contain multiple products
  - A product can be part of multiple orders

Example: multiple relationships

For this exercise, ignore attributes:

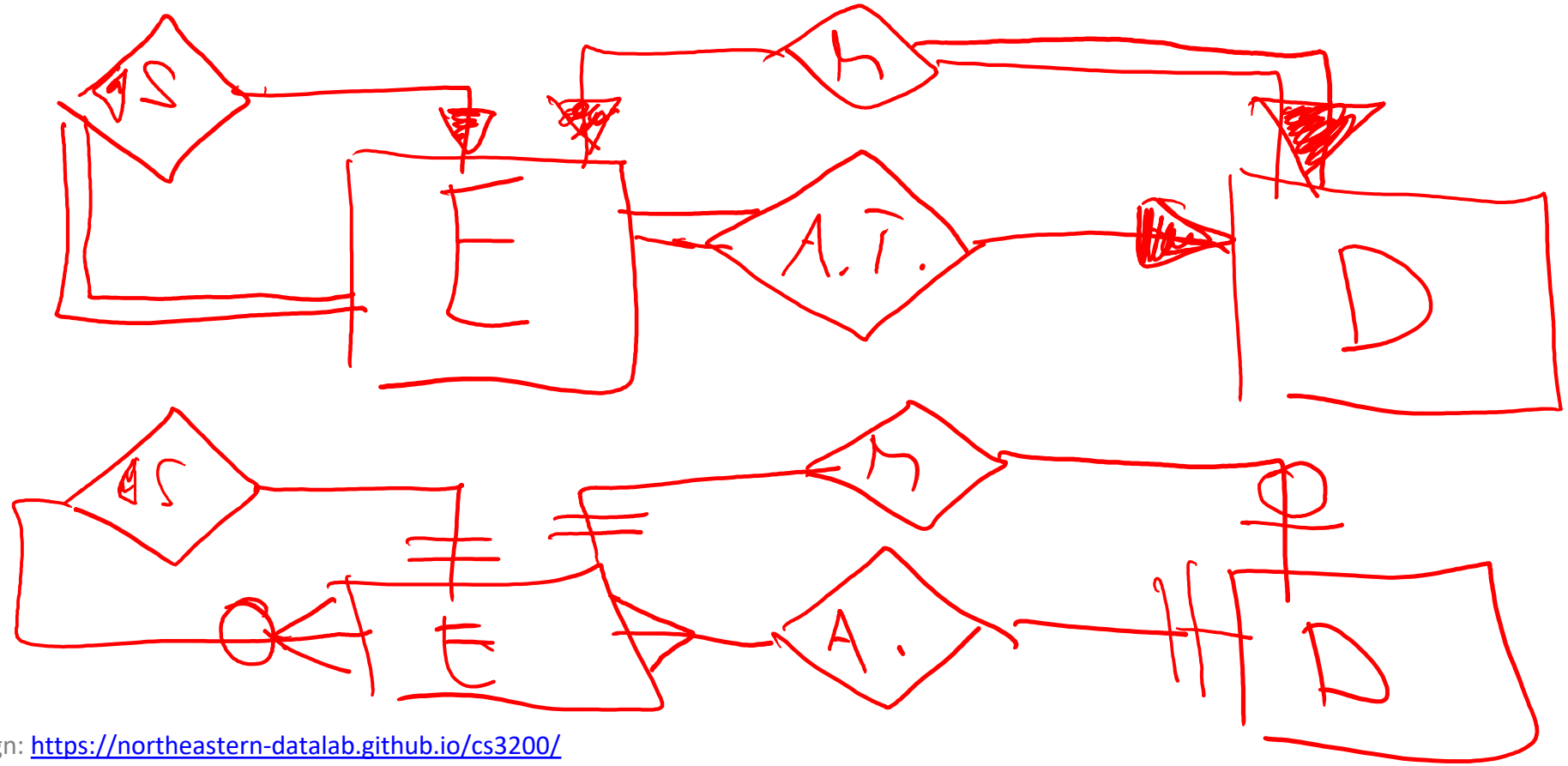
- Each employee is assigned to one department
- Each employee has one supervisor
- Each department is manged by one manager

?

Example: multiple relationships

For this exercise, ignore attributes:

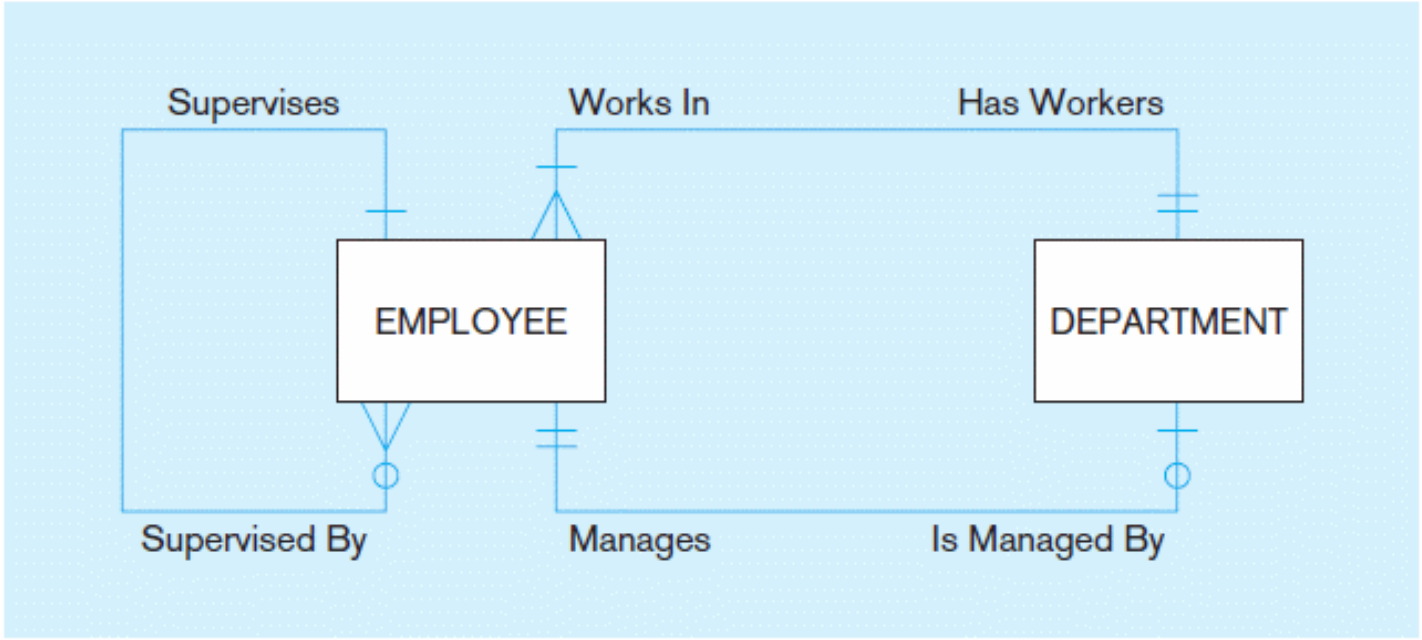
- Each employee is assigned to one department
- Each employee has one supervisor
- Each department is manged by one manager



Example: multiple relationships

For this exercise, ignore attributes:

- Each employee is assigned to one department
- Each employee has one supervisor
- Each department is manged by one manager



Recall: Entities can be related to one another in more than one way

Source: Fig. 2-21, Hoffer, Ramesh, Topi, "Modern database management," 10th ed, 2010.
Wolfgang Gatterbauer. Database design: https://northeastern-datalab.github.io/cs3200/