L11: ER modeling 3

CS3200 Database design (fa18 s2)

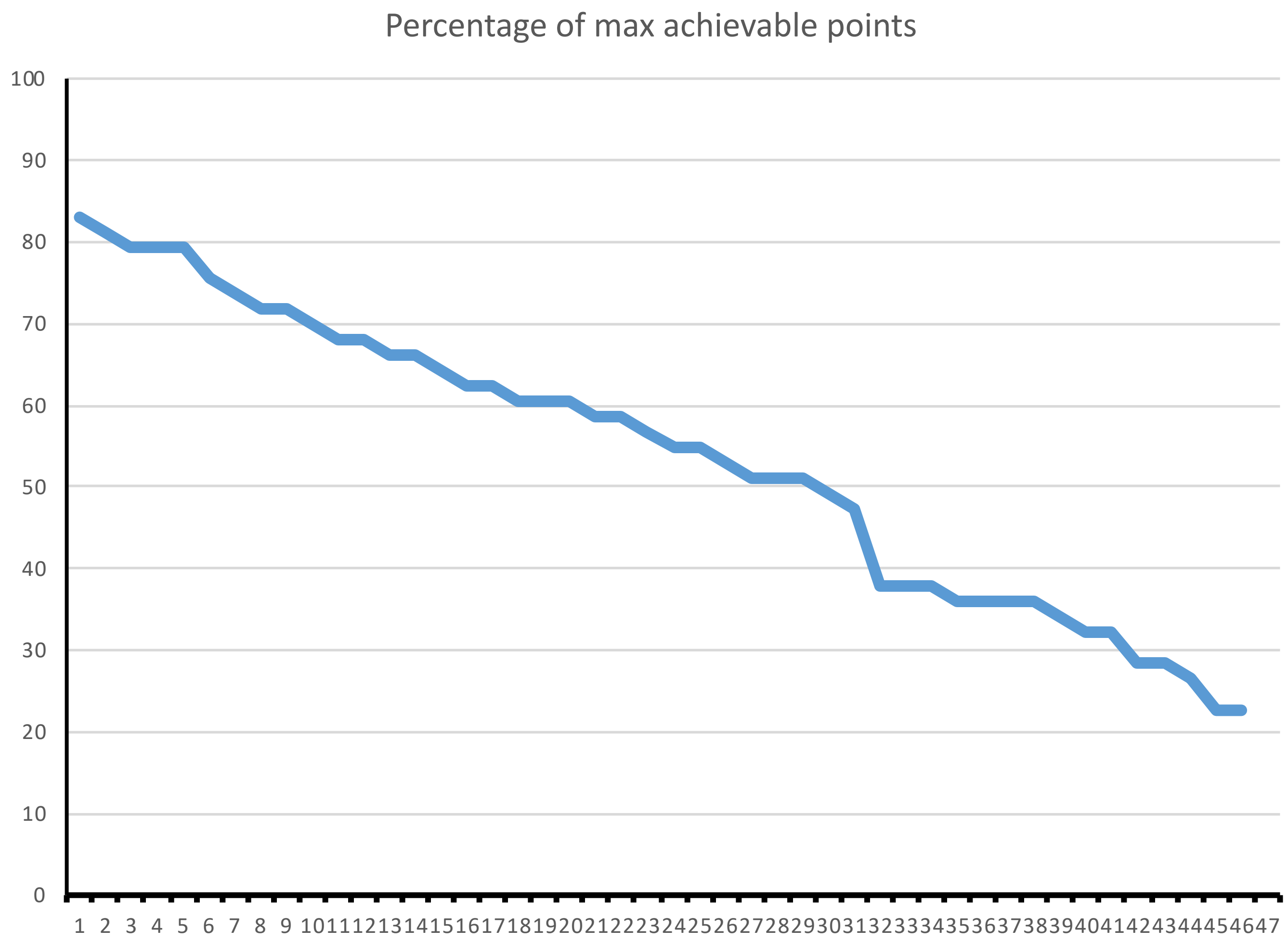
https://northeastern-datalab.github.io/cs3200/

Version 10/15/2018

Announcements!

- Keep bringing your name plates!
- Feedback:
  - Exam1 statistics:
    - next time: include point breakdown at beginning, fewer questions
  - Speed in class
    - about 1.5 weeks slower
  - Focus more on concepts, less syntax
    - that's my main goal
  - Homework and Gradiance: new content
    - think co-op, experiential learning

Exam 1

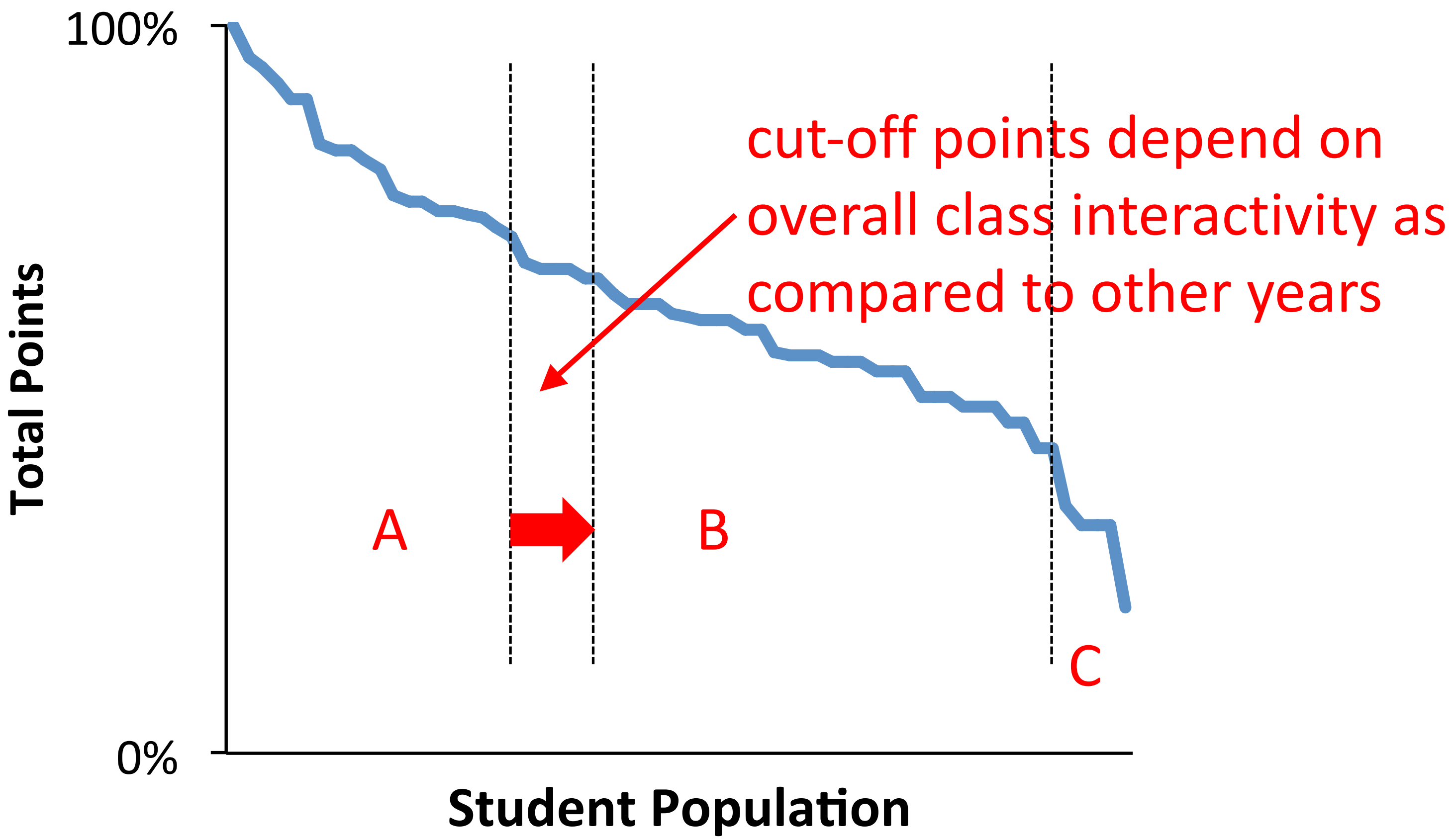


Percentage of max achievable points
100
90
80
70
60
50
40
30
20
10
0
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47

Grading Philosophy

Actual point distribution from a past final exam: long, but fair!

- no fixed percentages (e.g., top 30% get A)
- no fixed cut-offs (e.g., 80/100 points for A)



I will not disclose the actual cut-off points. Don't ask for an exception.

CS3200: Anonymous feedback

Your comments will help me (Wolfgang) tailor the course as we go along. I am the only one who can read these comments. Notice that you can also post anonymous comments to Piazza where everyone can see your comments. Thanks very much for filing this out!

Your name

Optional, only if you want me to get back to you

Your answer

1. Content

Do you understand what we are doing?

	1	2	3	4	5	6	7	8	9	10	
No clue what is going on	○	○	○	○	○	○	○	○	○	○	Super clear

2. Speed

How is the pace of the course?

	1	2	3	4	5	6	7	8	9	10	
Sooooooooo slow	○	○	○	○	○	○	○	○	○	○	Way too fast

3. Keep (+)

What is working well for you? What is your favorite part of this class and of my teaching?

Your answer

4. Change (-)

What specific suggestions do you have for changes to improve the course or how I teach it? Anything that you have seen in other classes you wished I adopted as well? Any part of the class content you like us to focus more on?

Your answer

5. Help (?)

Which topic from the class preparation do you like us to focus on more? Any particular question you have about the course but prefer to ask anonymously and not visible on Piazza?

Your answer

Content

Row Labels	Count of Content
5	1
6	1
7	1
8	5
9	4
Grand Total	**12**

Last semester:

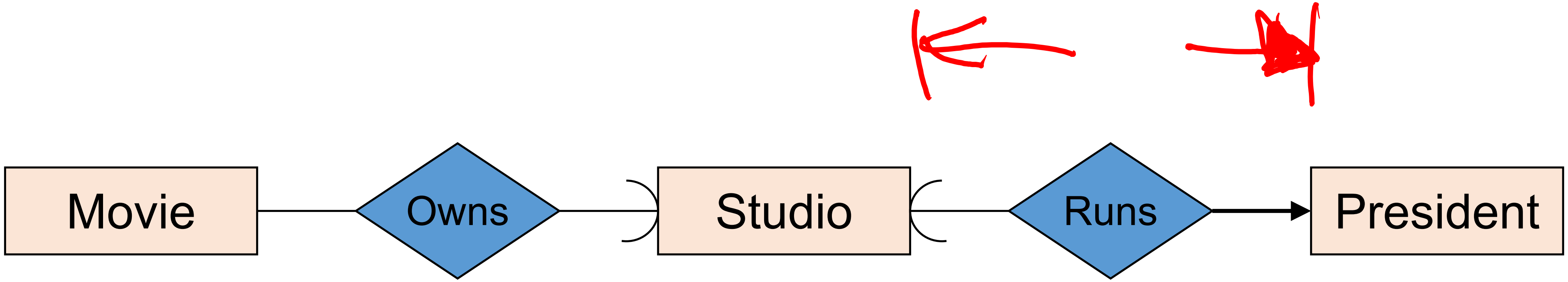
Row Labels	Count of What
4	1
5	3
6	5
7	7
8	12
9	4
10	2
Grand Total	**34**

Speed

Row Labels	Count of Speed
4	1
5	2
6	1
7	3
8	5
Grand Total	**12**

Row Labels	Count of Speed
4	2
5	9
6	8
7	9
8	5
9	1
Grand Total	**34**

Stanford arrow notation (used by Gradiance)



A studio can have at most one president

Each president must run exactly one studio (that exists in the studio entity set)

Transform it into crow feet!

"Referential integrity": a value appearing in one context must also appear in another

Weak
(or dependent)
Entities

Strong vs. Weak (Dependent) Entities

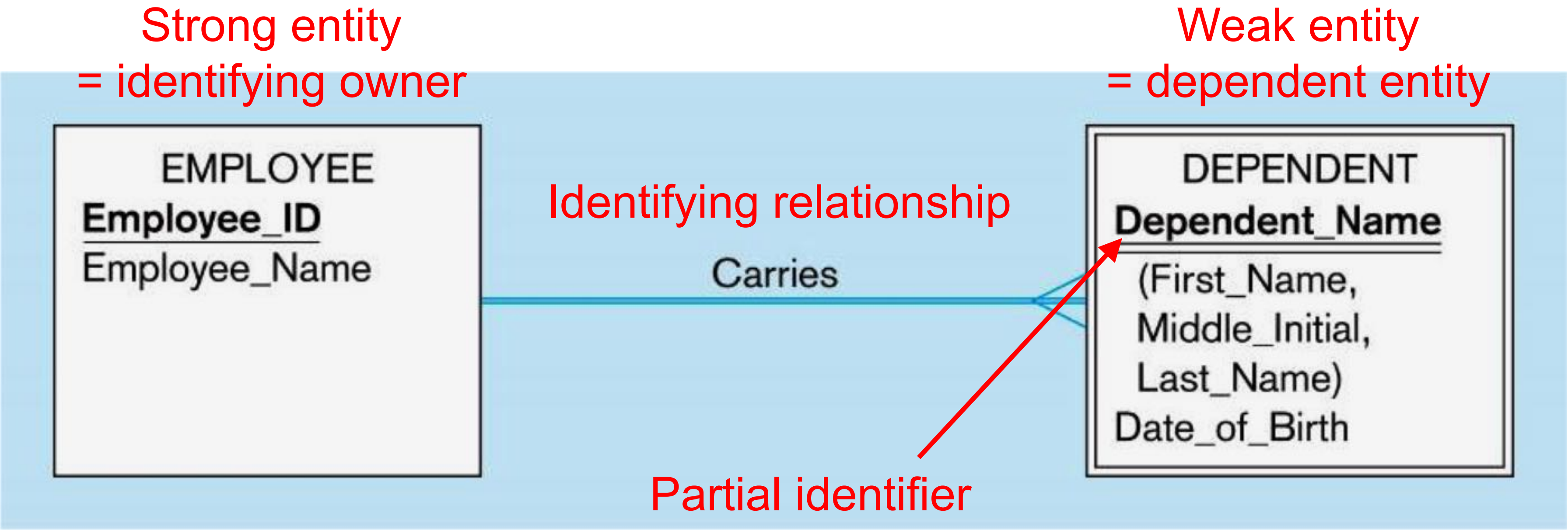
- **Strong entities**
  - Can be identified ("exist") independently of other types of entities
  - Have their own unique identifier
- **Weak entities**
  - Dependent on a strong entity,
    cannot exist on their own
    (better: cannot be identified independently)
  - Do not have unique identifiers: PK overlaps with parent's PK
  - (represented with double-line rectangle)

Entity sets are weak when part of their identifier comes from classes to which they are related

- **Identifying relationship**
  - Links strong entities to weak entities
  - Represented with double line relationship

Example: Strong and Weak Entities

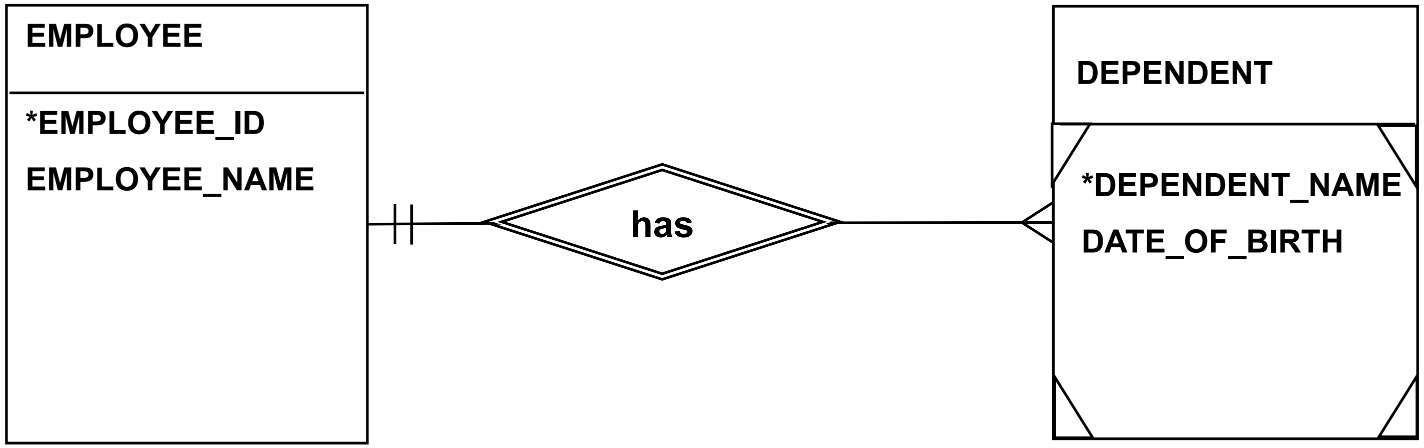
- Employee caries one dependent
  - Employee: ID, name
  - Dependent: name, Date_of_Birth

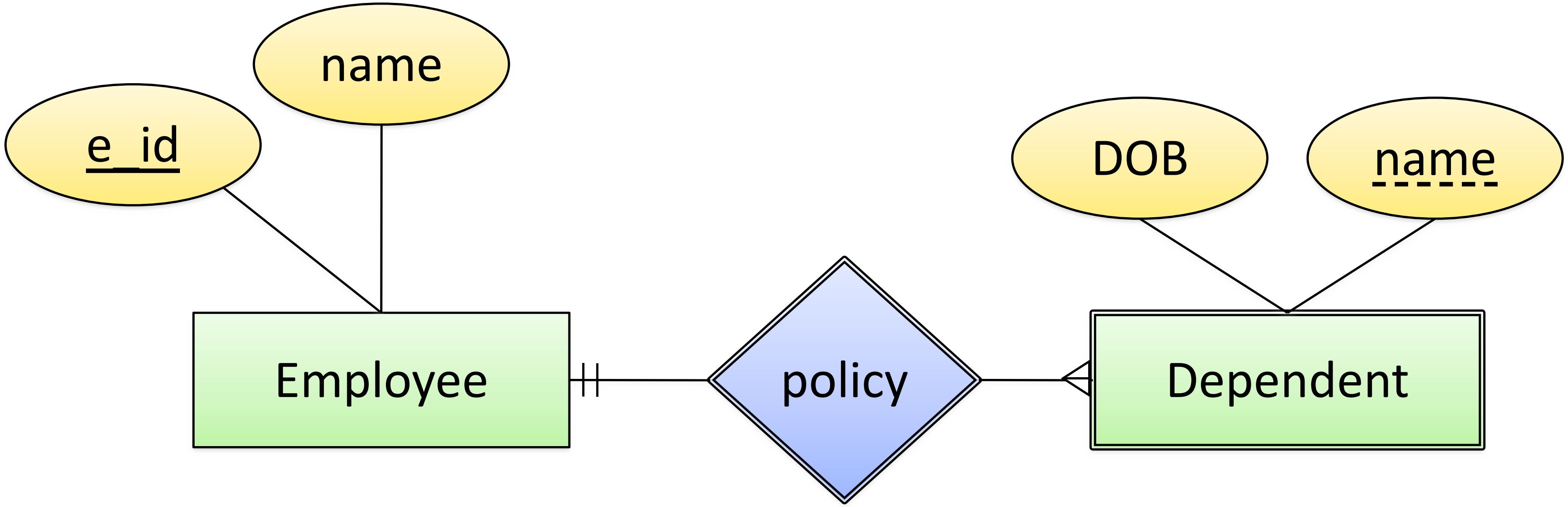


Note we need both EMPLOYEE_ID and DEPENDENT_NAME to uniquely identify a dependent

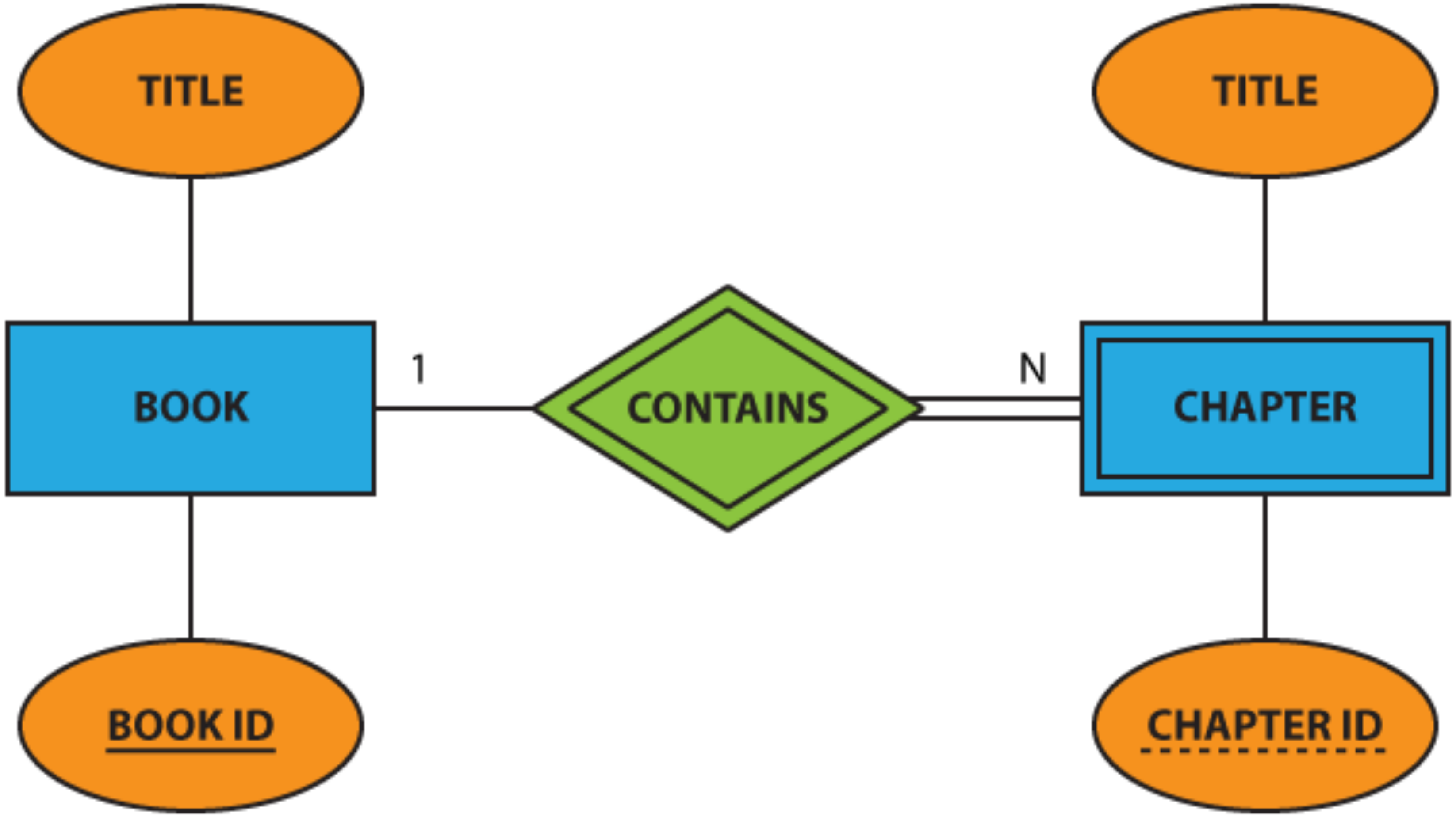
Source: Fig 2.5. Hoffer, Ramesh, Topi, "Modern database management," 10th ed, 2010.

Alternative notations for same scenario





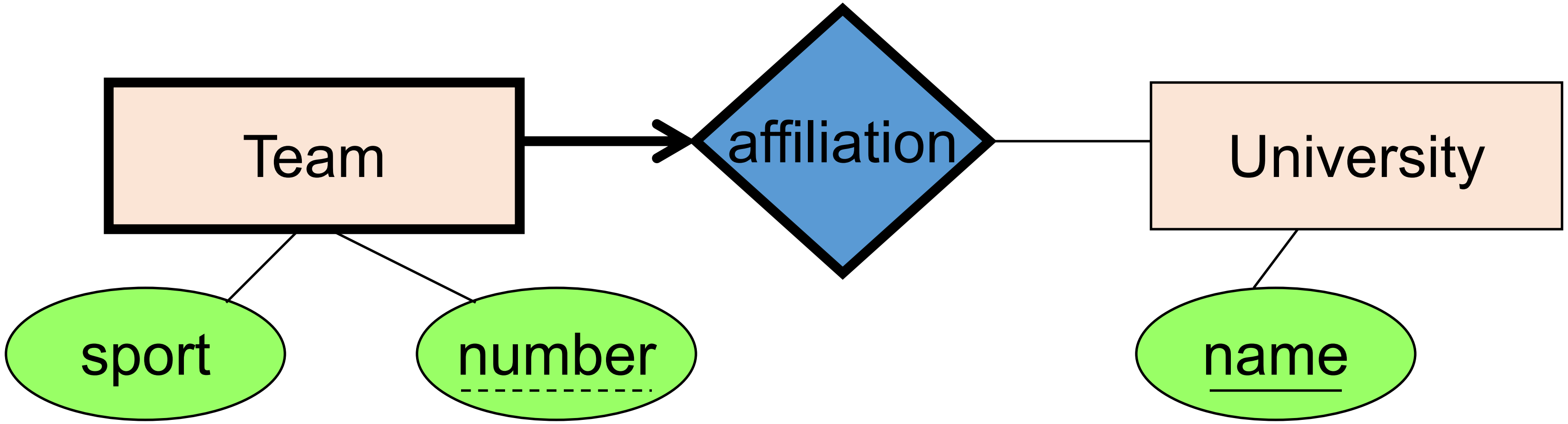
Participation constraints and weak entities



Source: http://www.vertabelo.com/blog/technical-articles/chen-erd-notation

Weak Entity Sets

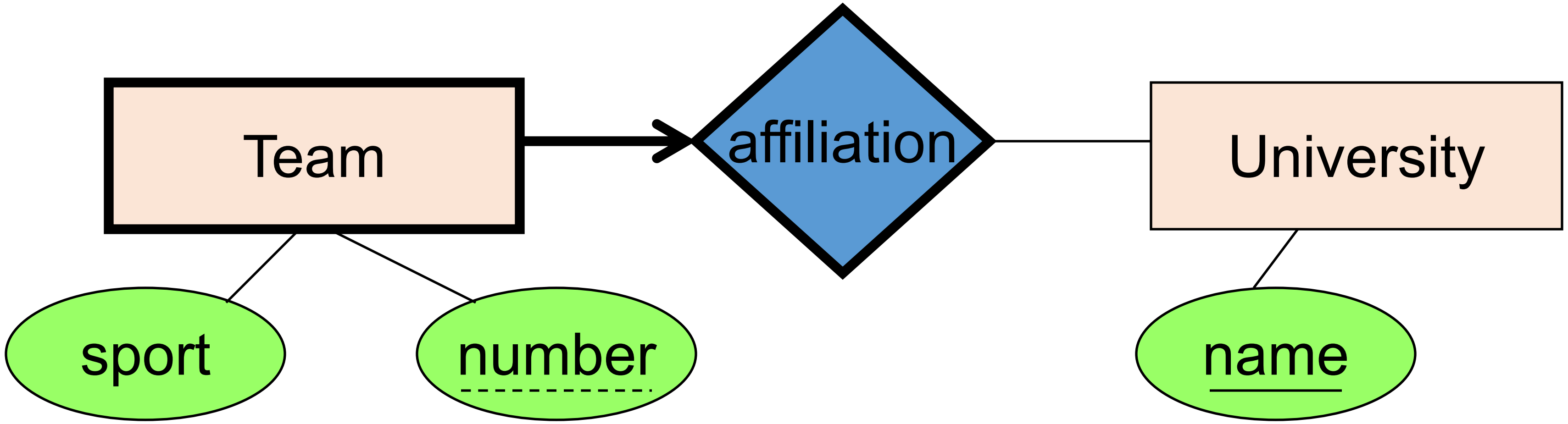
Entity sets are *weak* when their key comes from other classes to which they are related.



"Football team" v. "***The Northeastern*** Football team" *(E.g., BU has a football team too, sort of)*

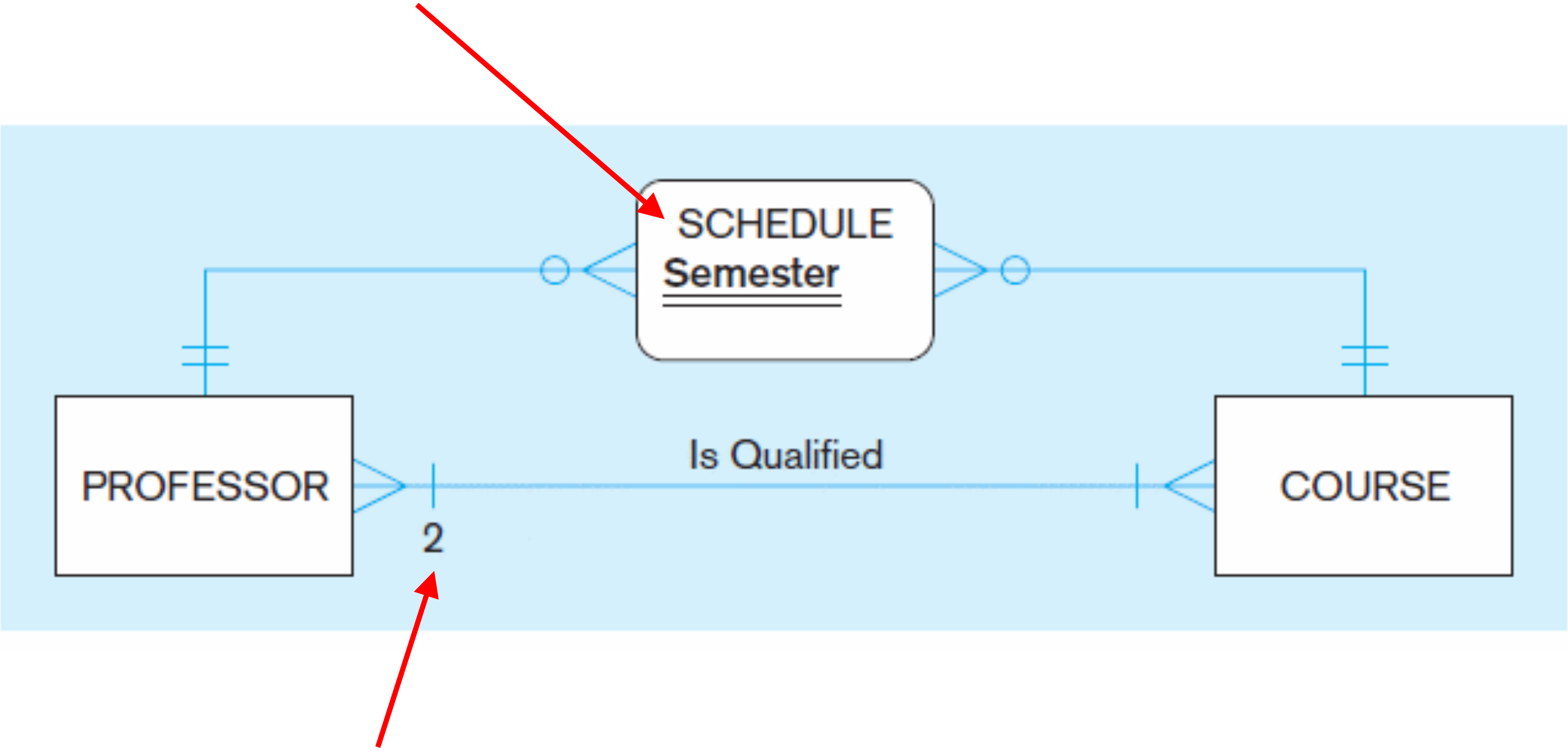
Weak Entity Sets

Entity sets are <u>*weak*</u> when their key comes from other classes to which they are related.



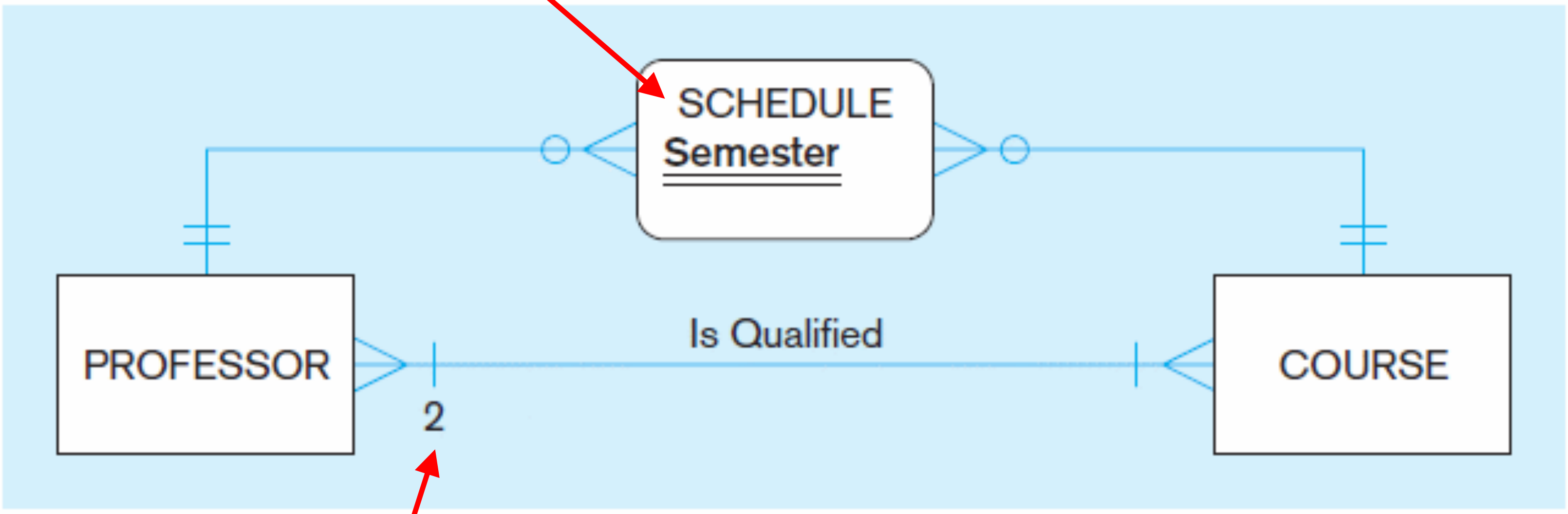
- number is a <u>*partial key*</u>. (denote with dashed underline).
- University is called the <u>*identifying owner*</u>.
- Participation in affiliation must be <u>total</u>. Why?

Multiple relationships



Source: Fig 2-21, Hoffer, Ramesh, Topi, "Modern database management," 10th ed, 2010.

Multiple relationships

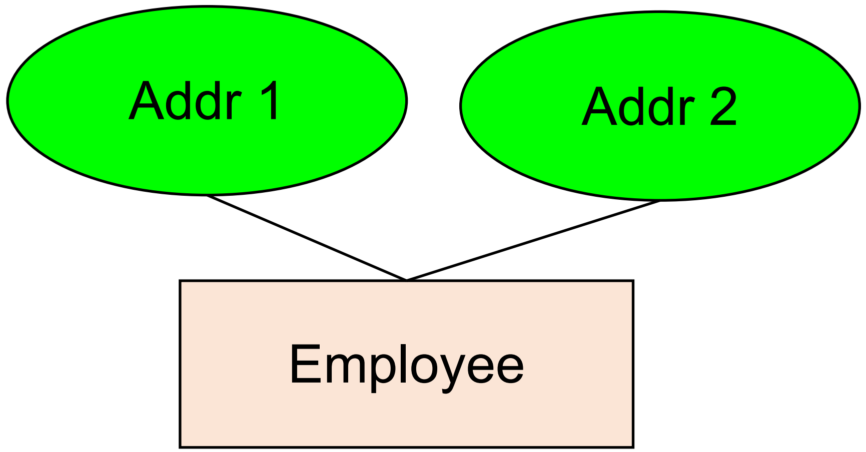


Here, min cardinality constraint is 2: At least two professors must be qualified to teach each course

Source: Fig 2-21, Hoffer, Ramesh, Topi, "Modern database management," 10th ed, 2010.

Examples: Entity vs. Attribute

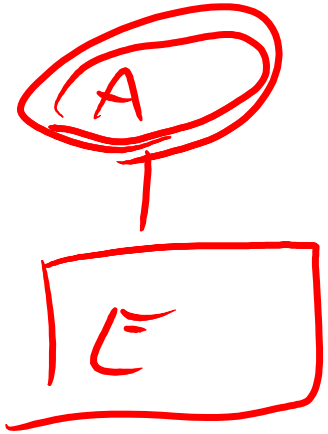
Should address (A) be an attribute?



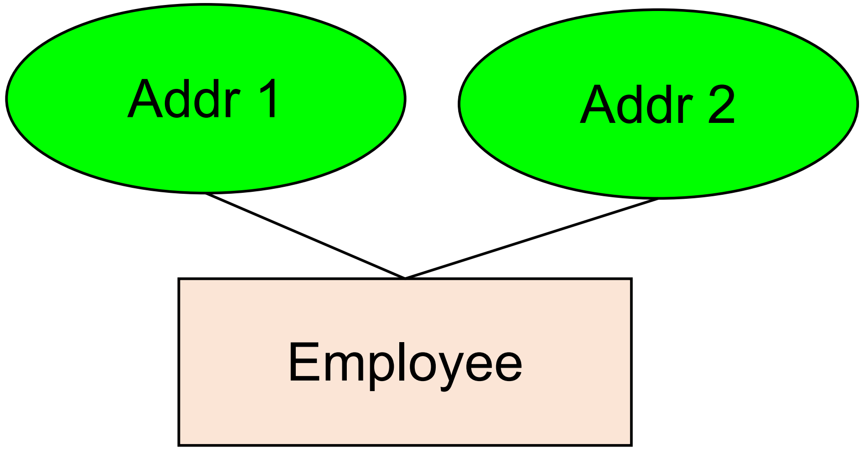
How do we handle employees with multiple addresses here?

How do we handle addresses where internal structure of the address (e.g. zip code, state) is useful?

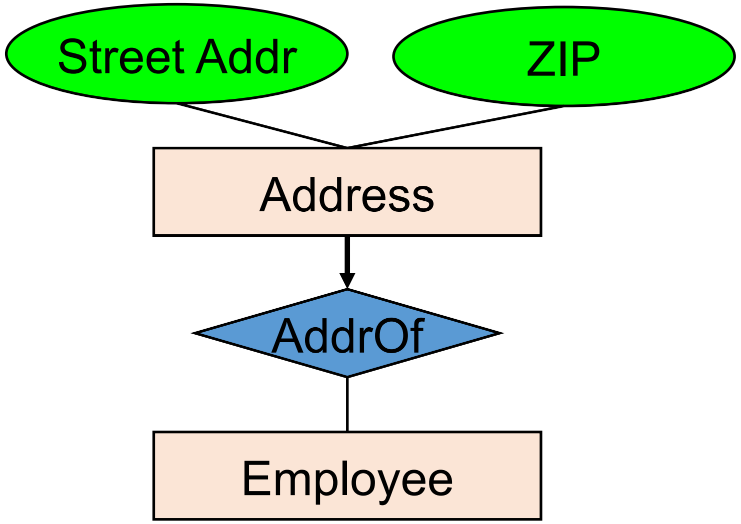
Examples: Entity vs. Attribute



Should address (A) be an attribute?

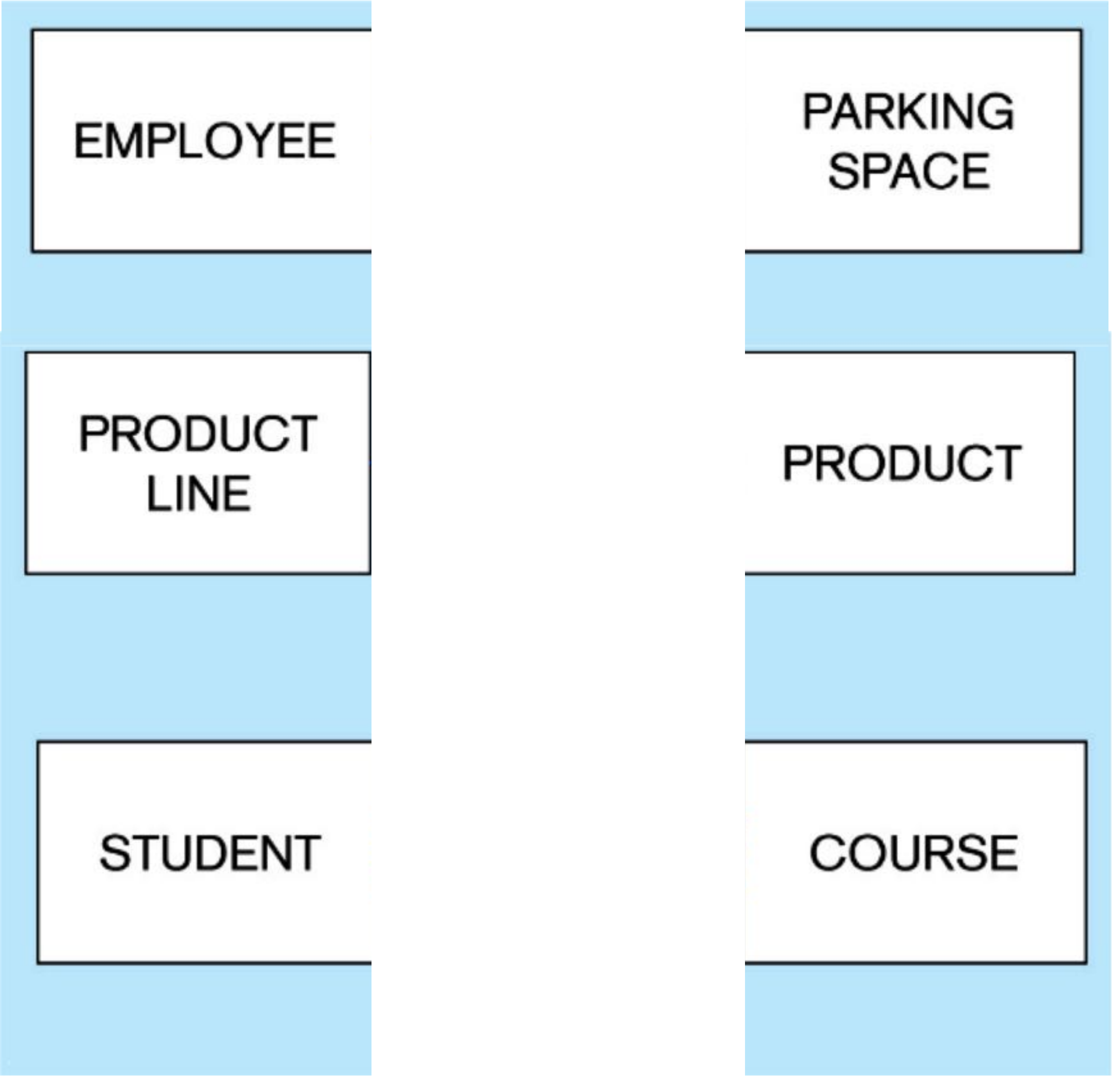


Or (B) be an entity?



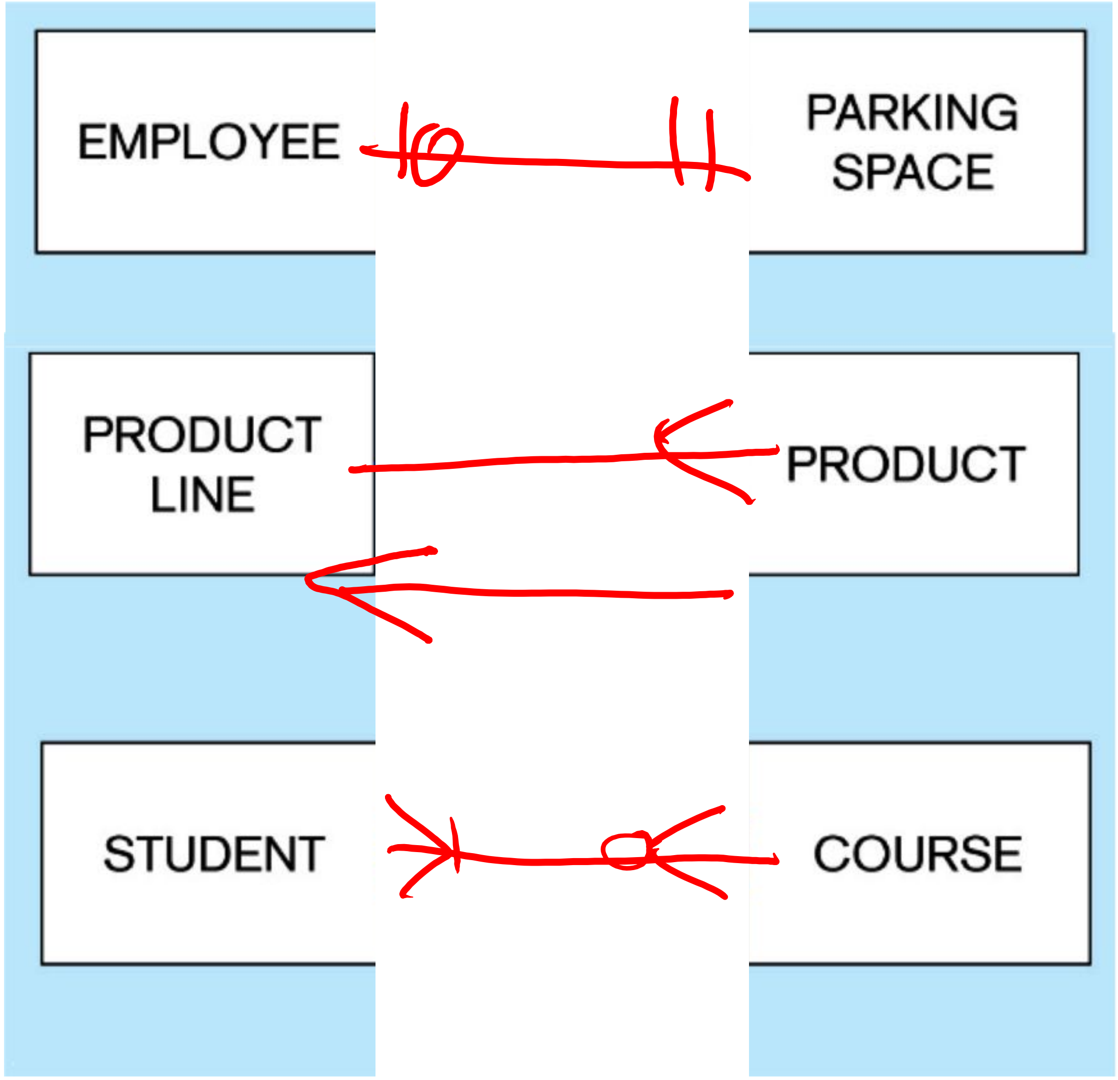
In general, when we want to record several values, we choose new entity

Example: Binary Relationships

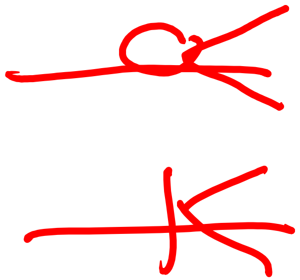


Source: Hoffer, Ramesh, Topi, "Modern database management," 10th ed, 2010.

Example: Binary Relationships

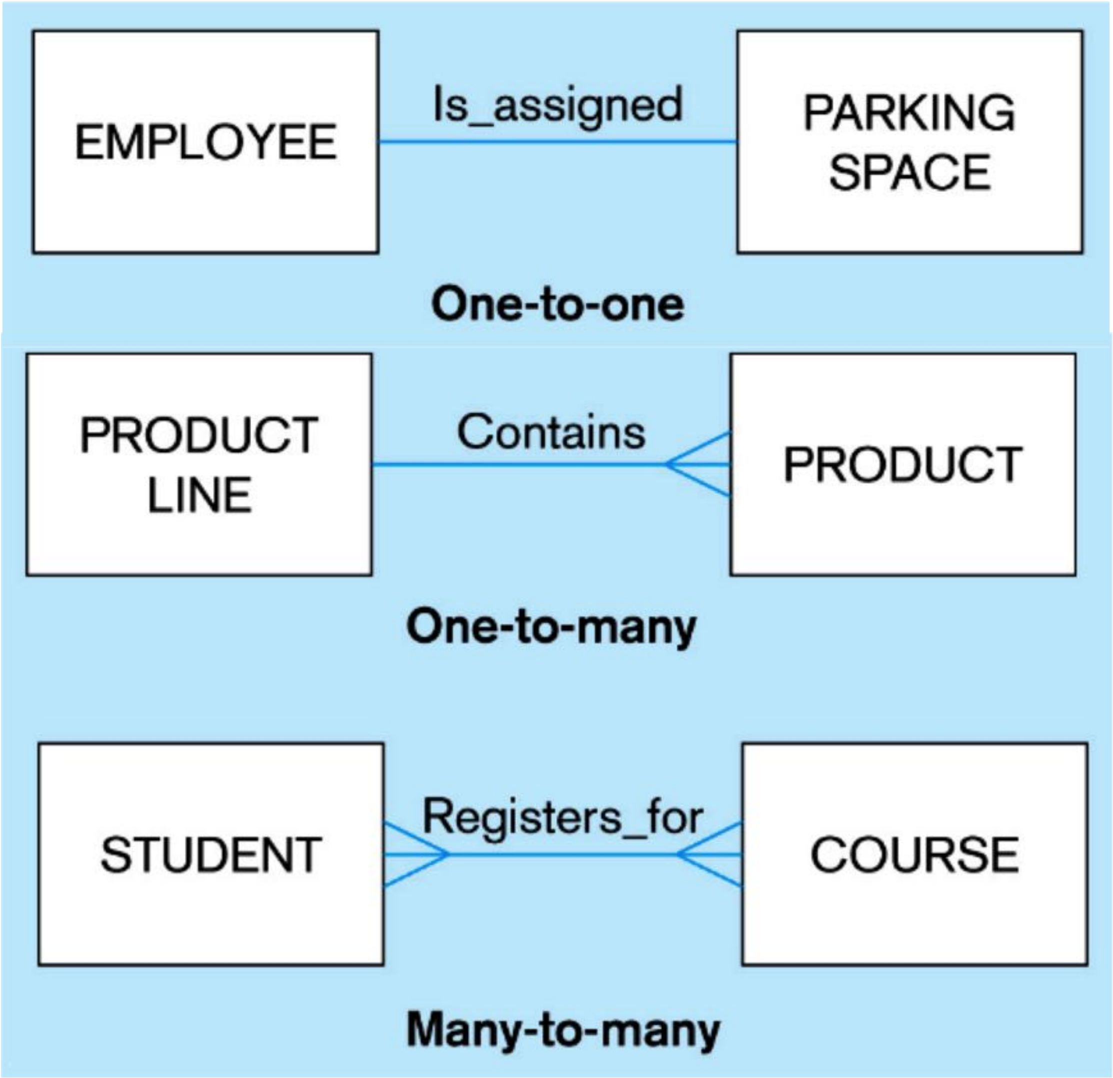


STANFORD ARROW



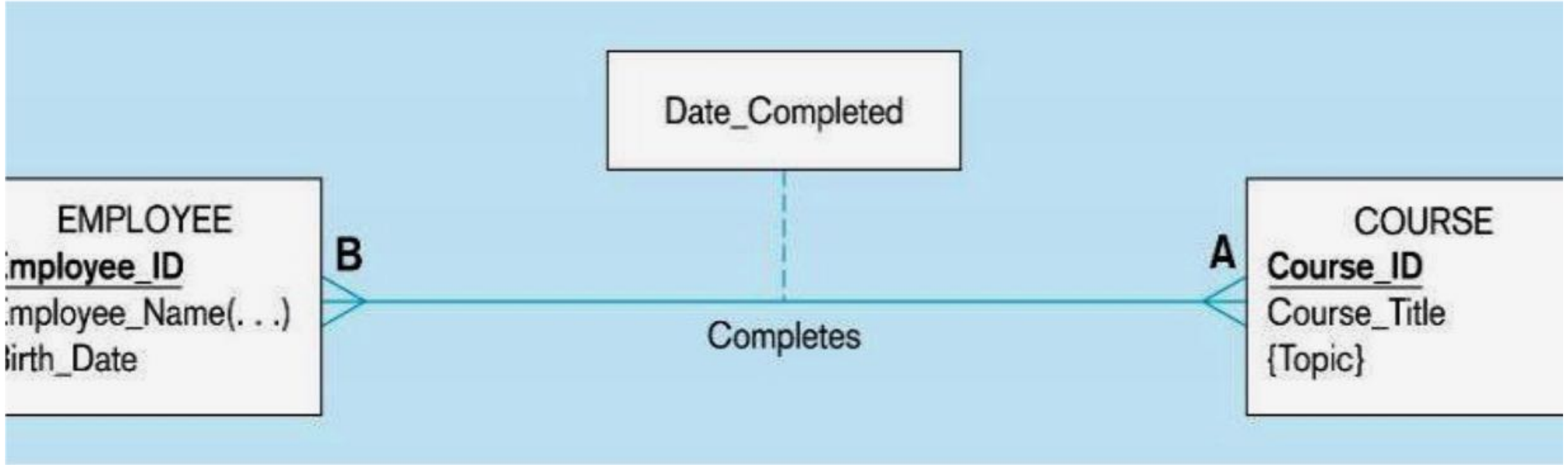
Source: Hoffer, Ramesh, Topi, "Modern database management," 10th ed, 2010.

Example: Binary Relationships



Source: Hoffer, Ramesh, Topi, "Modern database management," 10th ed, 2010.

Example: Binary Relationship With An Attribute



- The date completed attribute pertains specifically to the employee's completion of a course
- It is an attribute of the relationship, not either entity in isolation

Source: Hoffer, Ramesh, Topi, "Modern database management," 10th ed, 2010.

Examples: Unary Degree Relationship

Person
Is married to

Employee
Manages

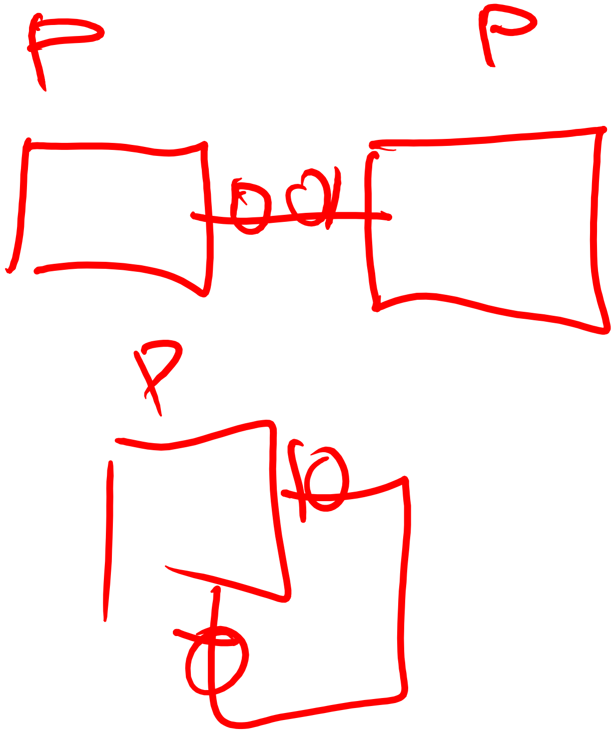
Team
Stands After

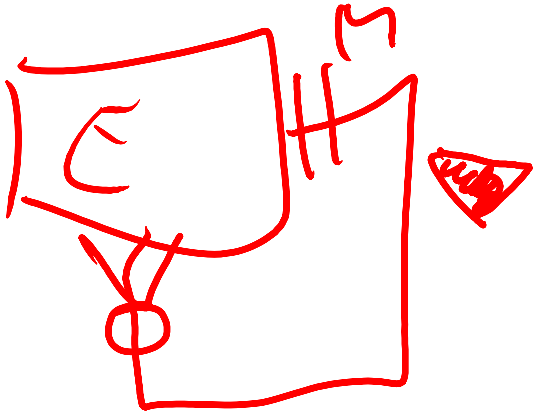
Examples: Unary Degree Relationship

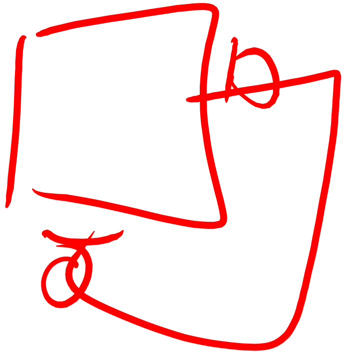
Person
Is married to

Employee
Manages

Team
Stands After





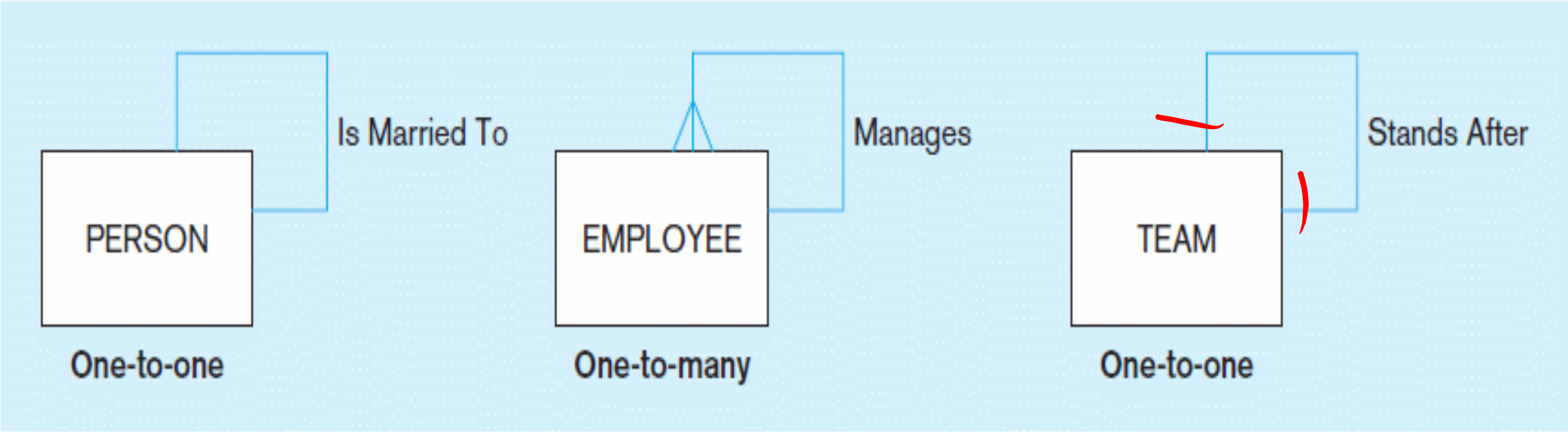


Examples: Unary Degree Relationship

Person
Is married to

Employee
Manages

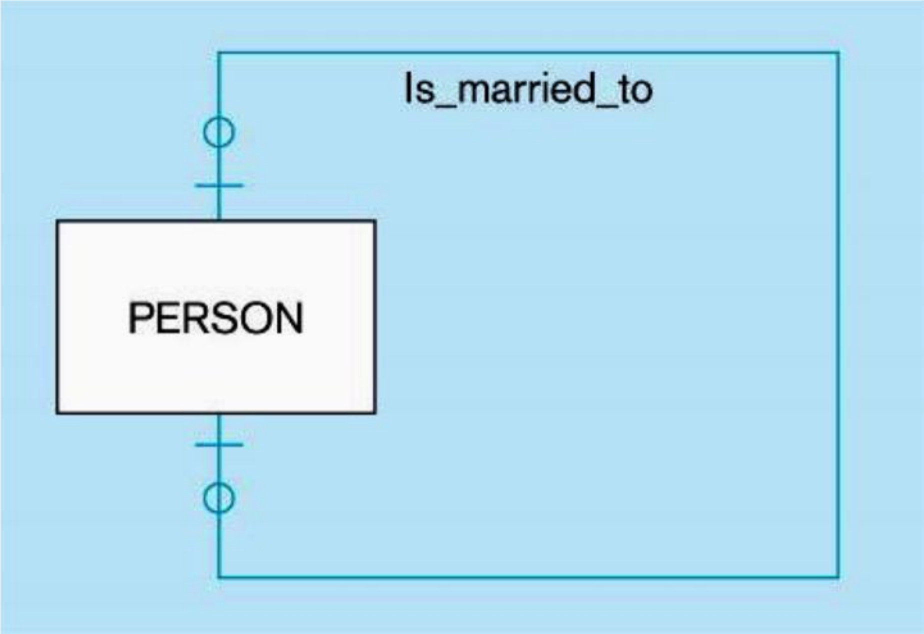
Team
Stands After



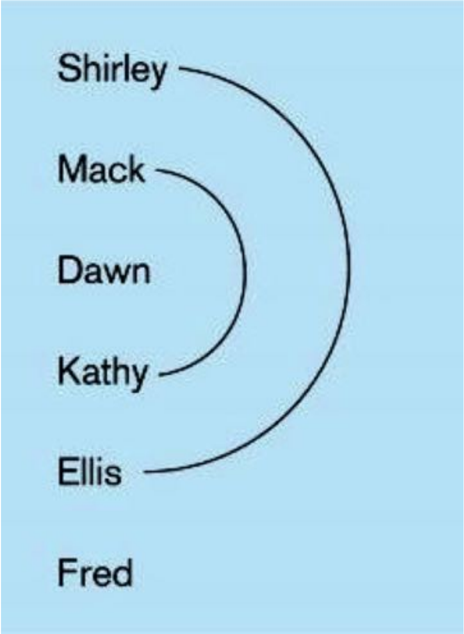
Source: Hoffer, Ramesh, Topi, "Modern database management," 10th ed, 2010.

Example: Married to with participation

type

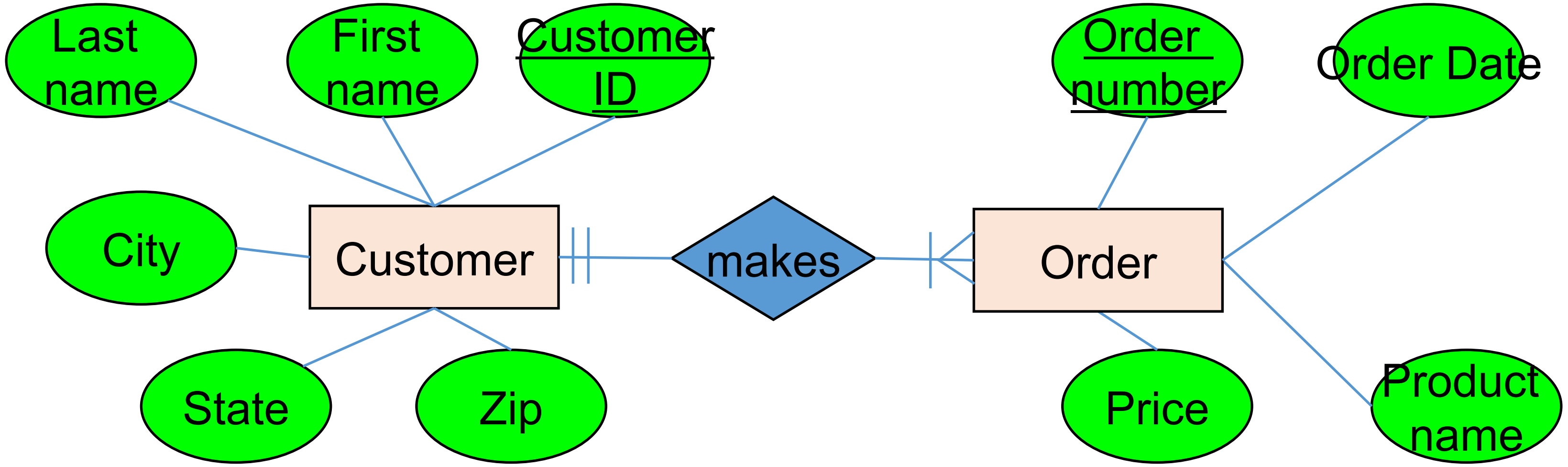


instance

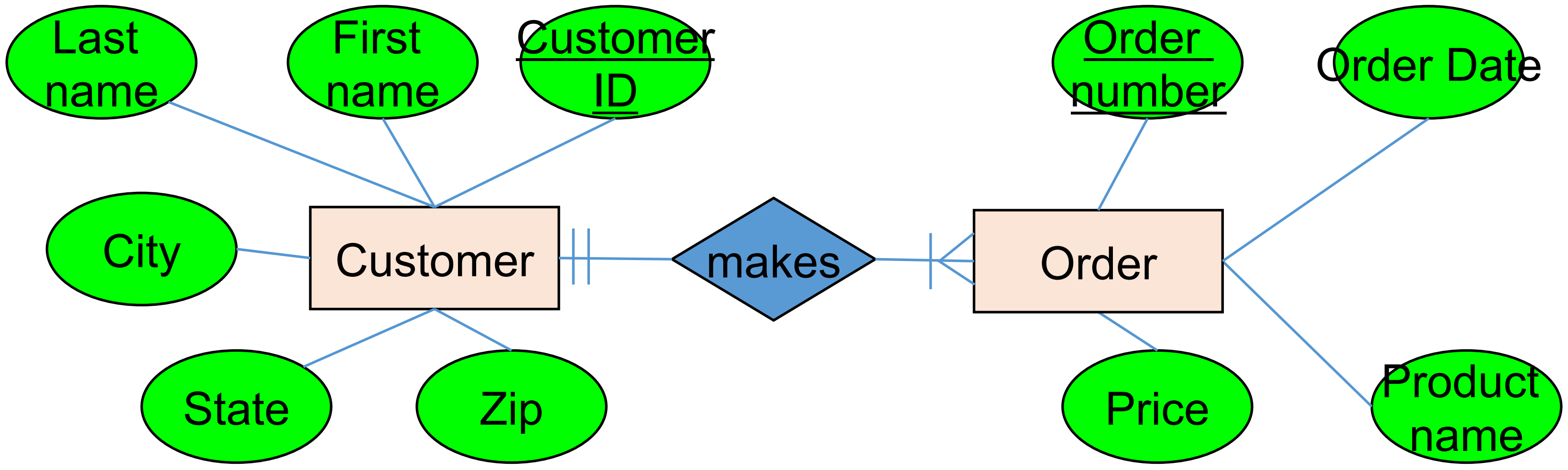


Source: Hoffer, Ramesh, Topi, "Modern database management," 10th ed, 2010.

There is a problem with our ERD



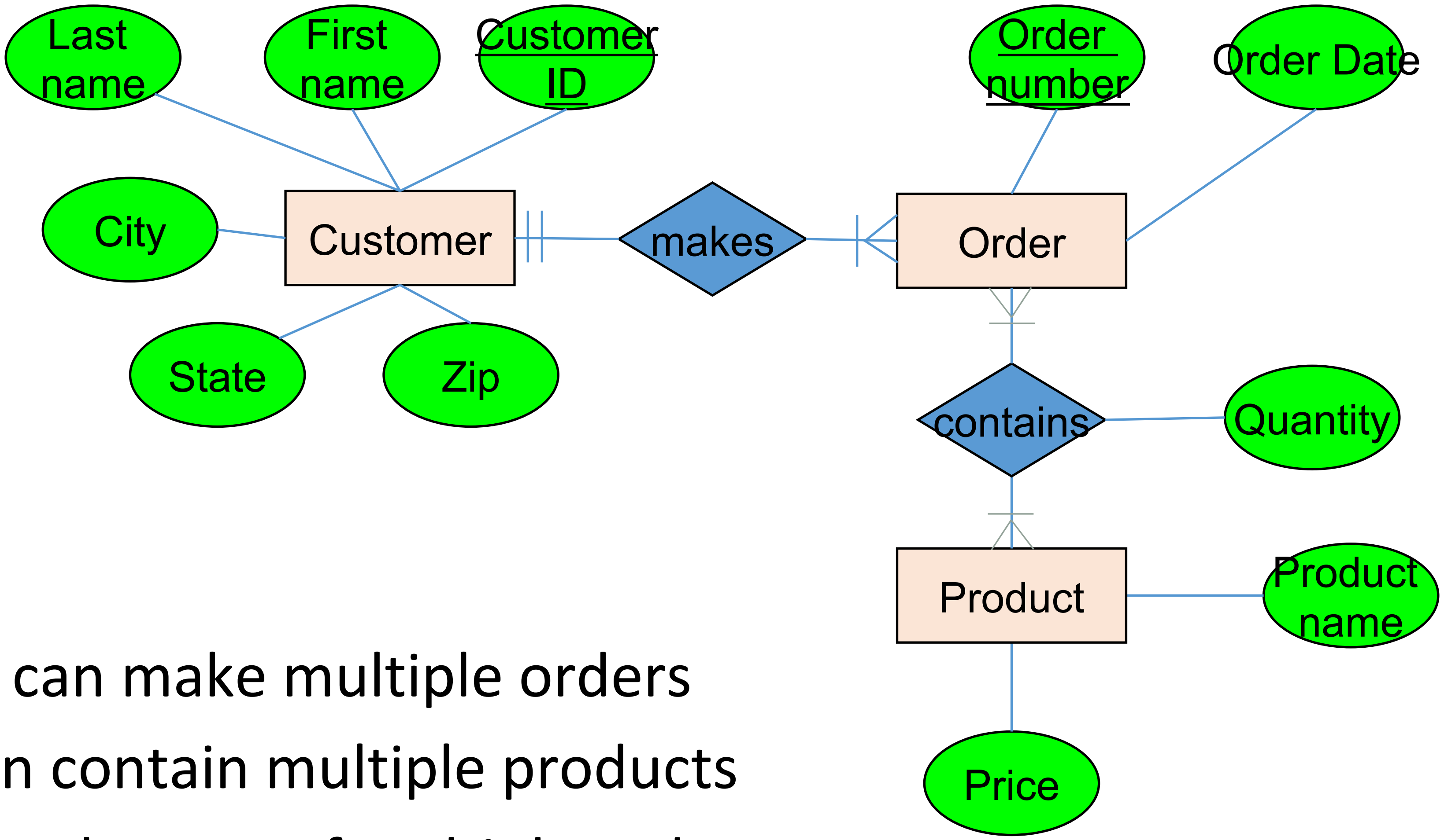
There is a problem with our ERD



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

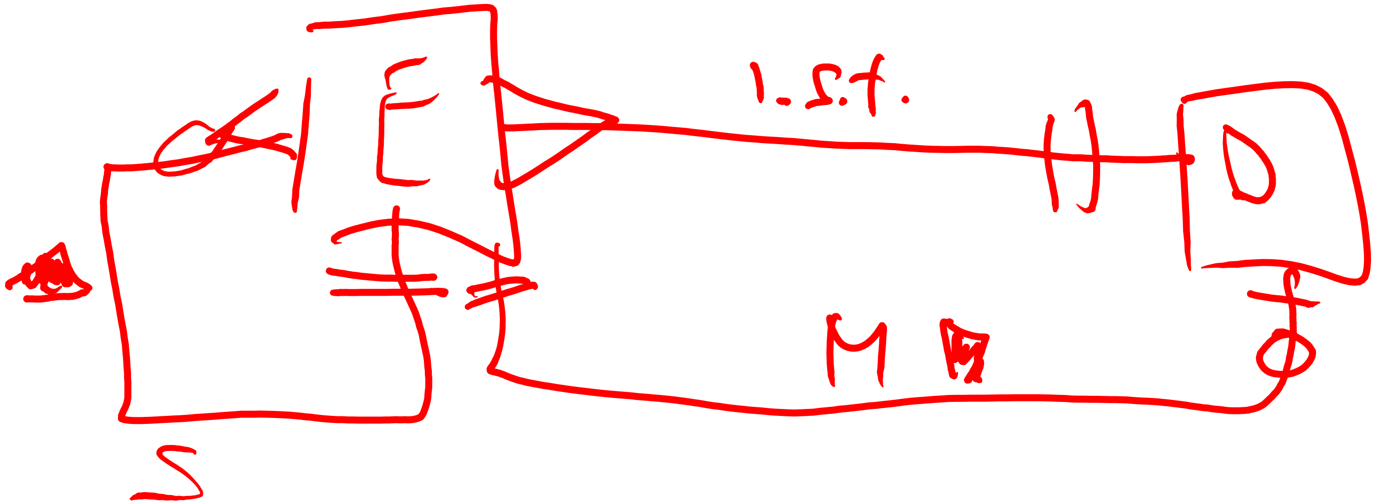


- Now
  - A customer can make multiple orders
  - An order can contain multiple products
  - A product can be part of multiple orders

Example: multiple relationships

For this exercise, ignore attributes:

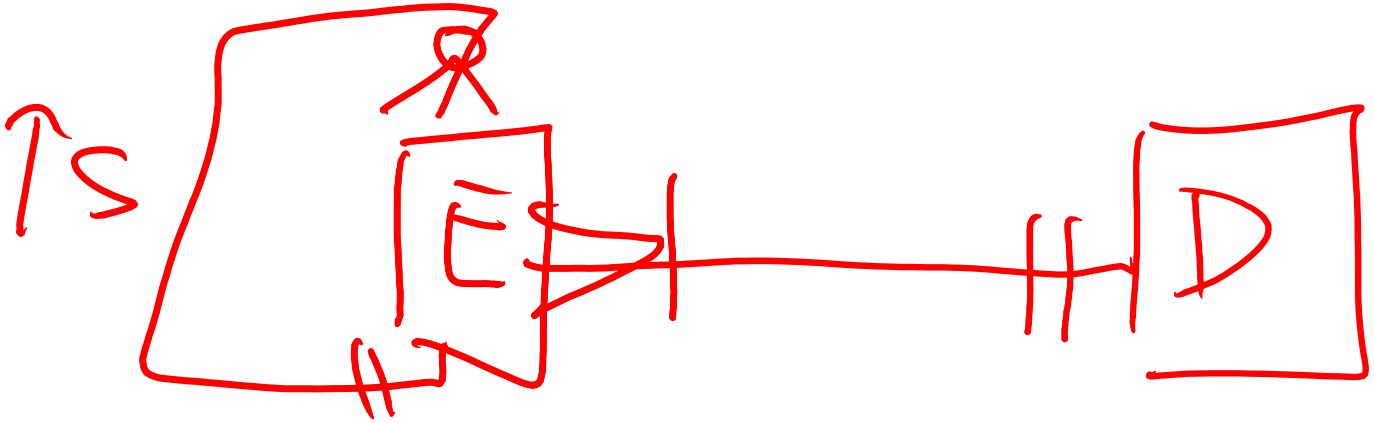
- Each employee is assigned to one department
- Each employee has one supervisor
- Each department is manged by one manager

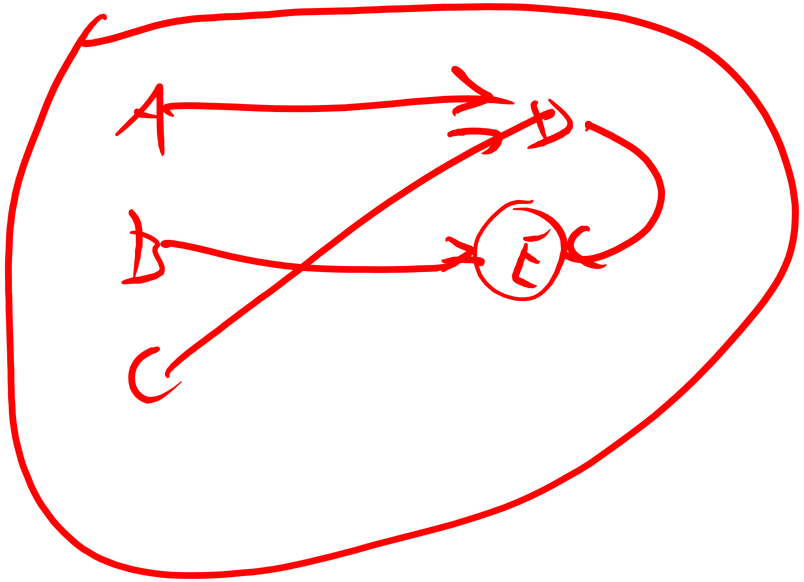


Example: multiple relationships

For this exercise, ignore attributes:

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

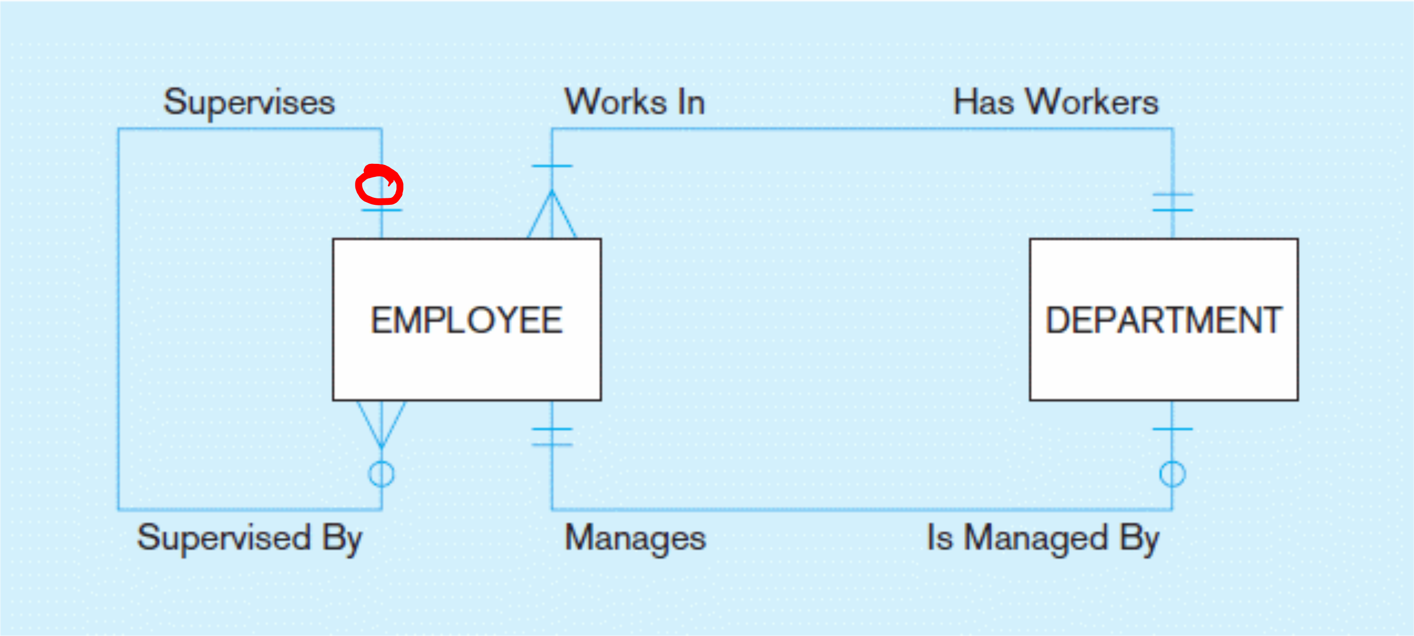




Example: multiple relationships

For this exercise, ignore attributes:

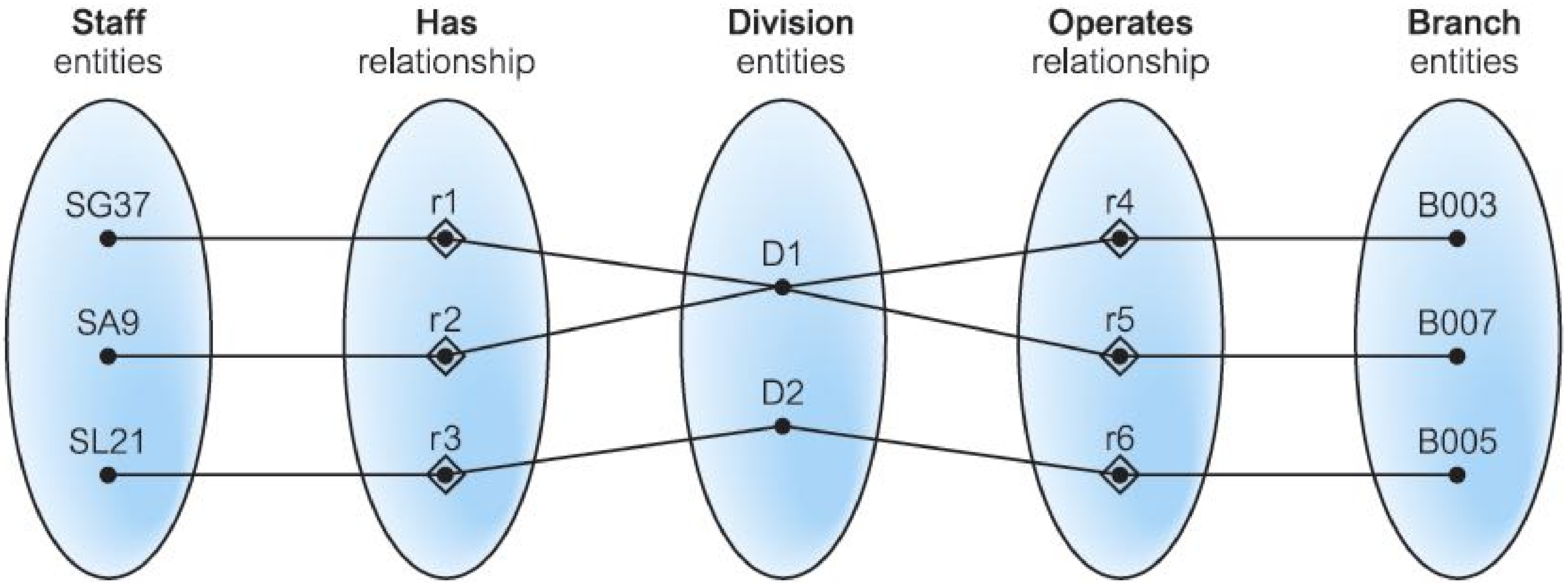
- Each employee is assigned to one department
- Each employee has one supervisor
- Each department is manged by one manager



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.

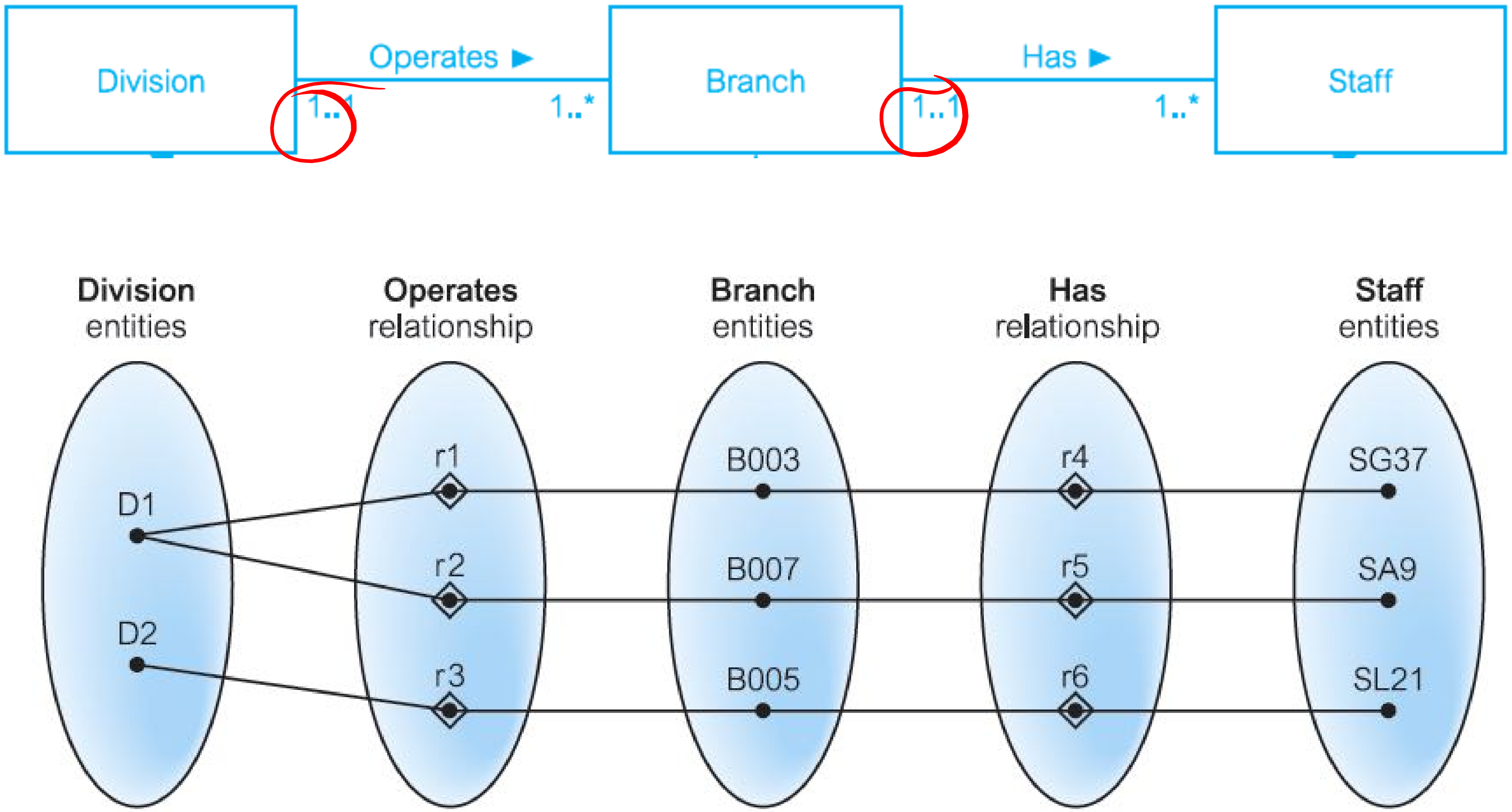
We have a problem



Fan Trap: Where a model represents a relationship between entity types, but the pathway between certain entity occurrences is ambiguous. May exist when two or more 1:n relationships <u>fan out from the same entity</u>

Source: Connolly, Begg: Database systems, 4th ed, p. 364, 2005.

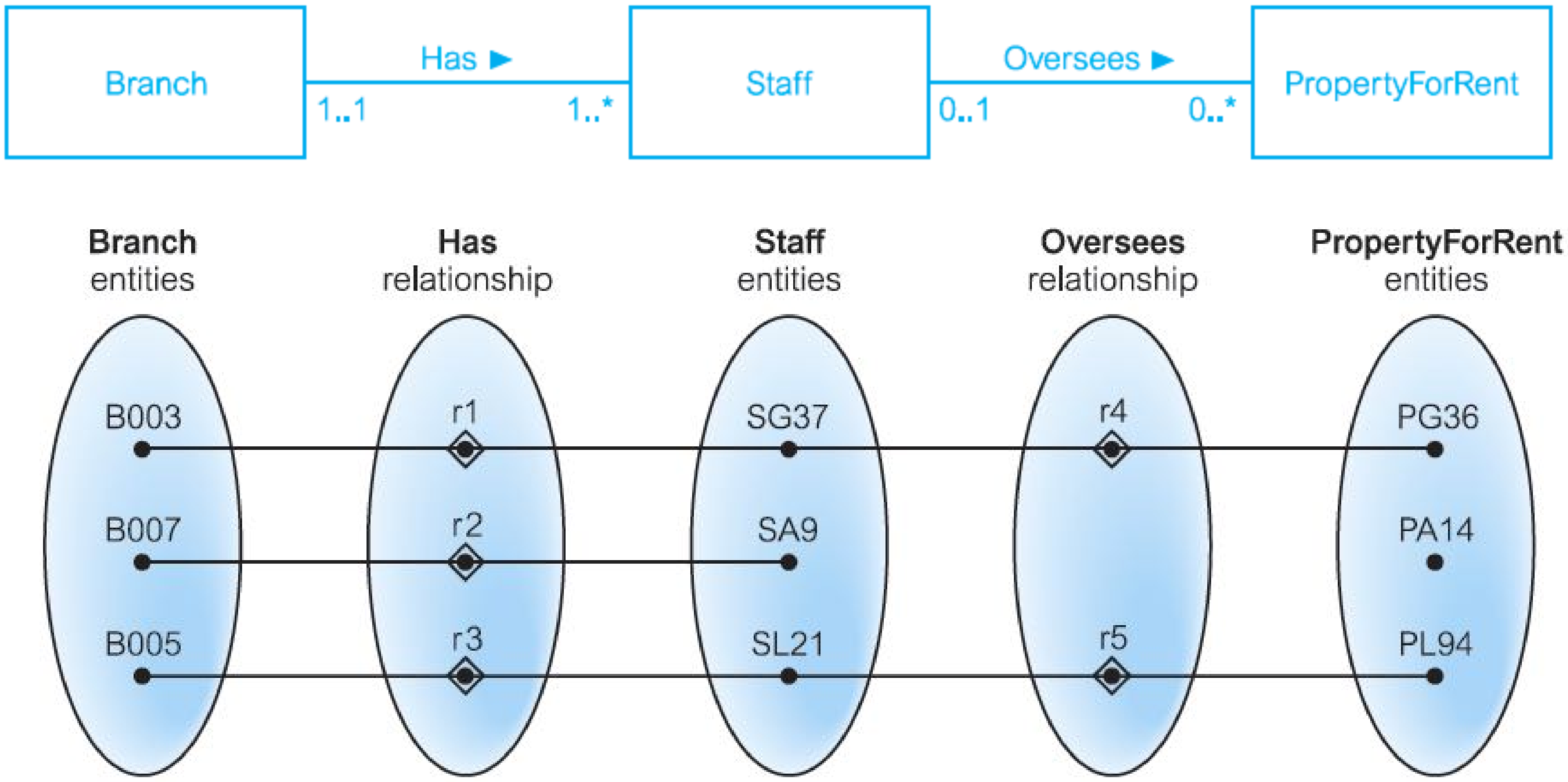
Restructuring the model helps (in this case)



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

Source: Connolly, Begg: Database systems, 4th ed, p. 364, 2005.

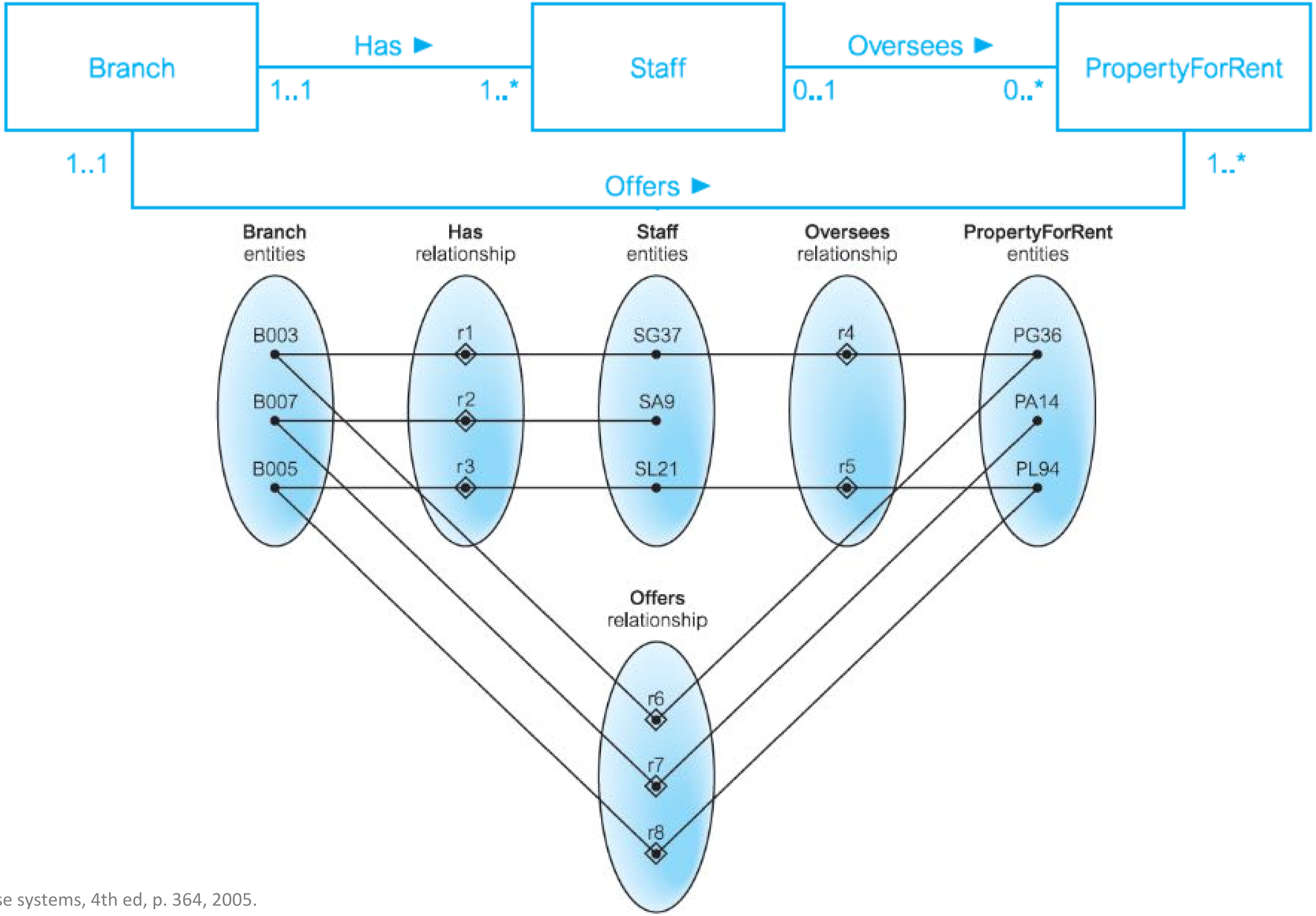
We have another problem



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

Source: Connolly, Begg: Database systems, 4th ed, p. 364, 2005.

Adding a relationship helps here

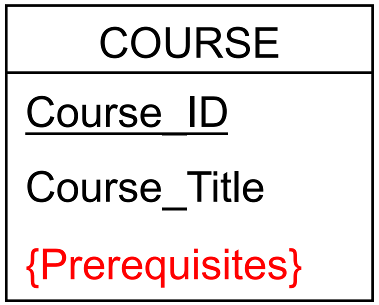


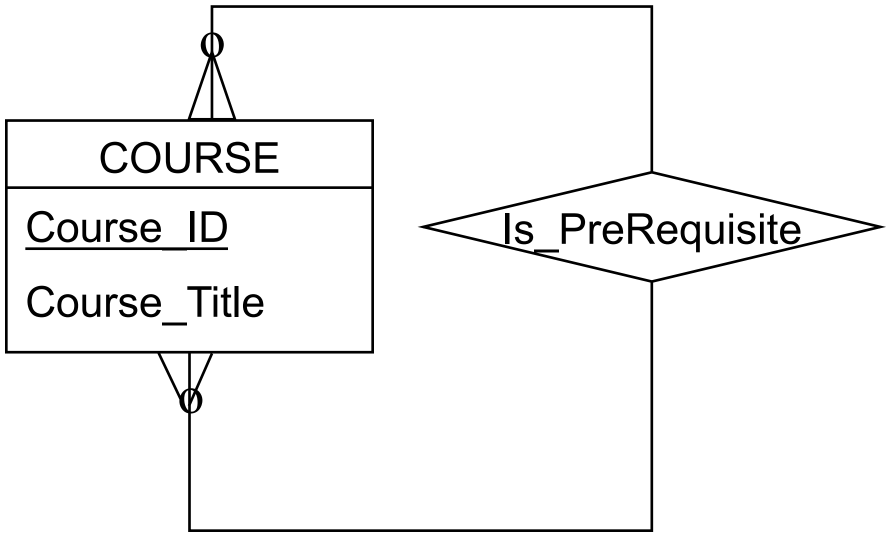
Source: Connolly, Begg: Database systems, 4th ed, p. 364, 2005.

1. Multivalued attributes represented as relationships

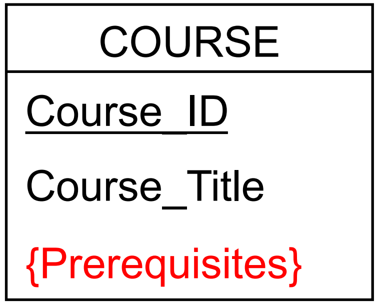
COURSE
<u>Course_ID</u>
Course_Title
{Prerequisites}

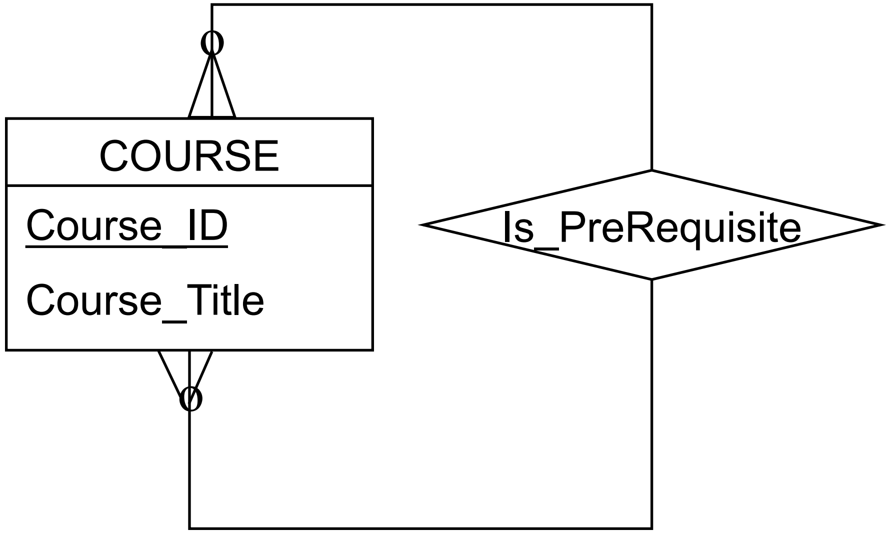
1. Multivalued attributes represented as relationships



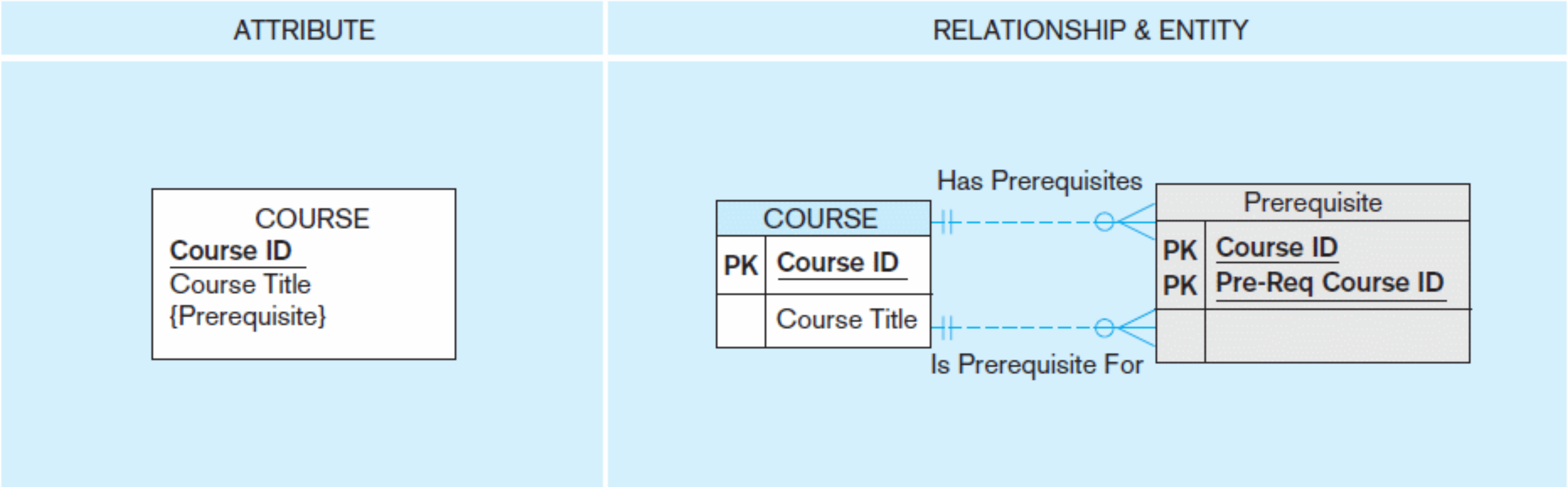


1. Multivalued attributes represented as relationships





Notation used by Microsoft Visio:



Source: Fig 2.15: Hoffer, Ramesh, Topi, "Modern database management," 10th ed, 2010.

2. Multivalued attributes can be represented as entities

EMPLOYEE

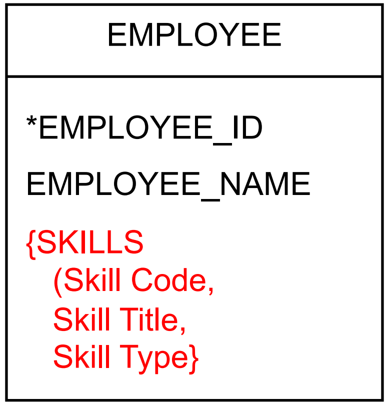
*EMPLOYEE_ID

EMPLOYEE_NAME

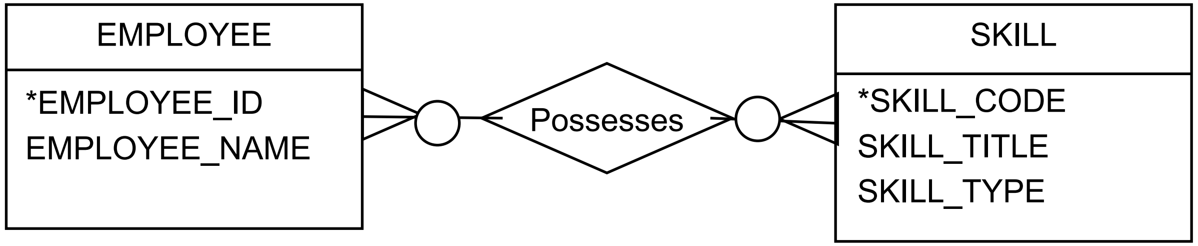
{SKILLS
(Skill Code,
Skill Title,
Skill Type}

composite

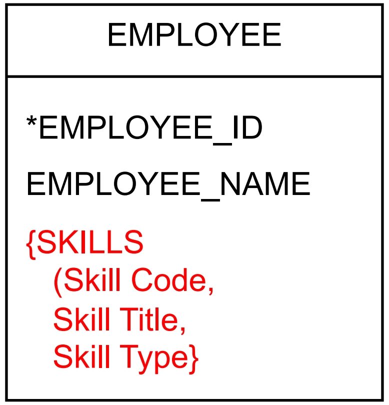
2. Multivalued attributes can be represented as entities



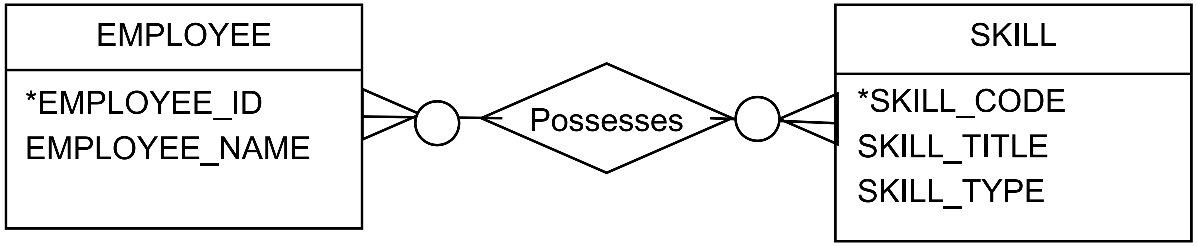
composite



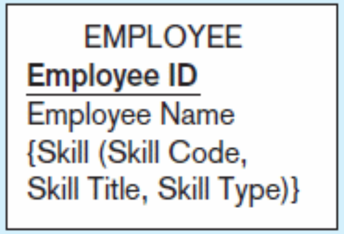
2. Multivalued attributes can be represented as entities

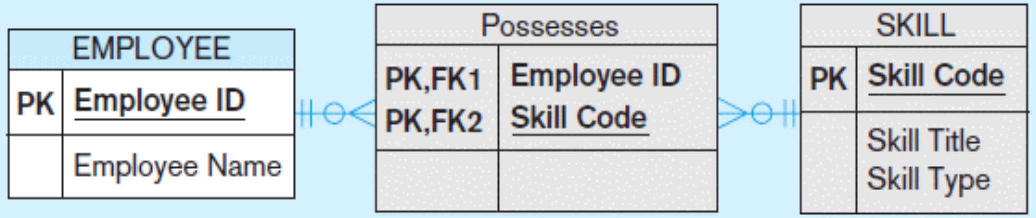


composite



Notation used by Microsoft Visio



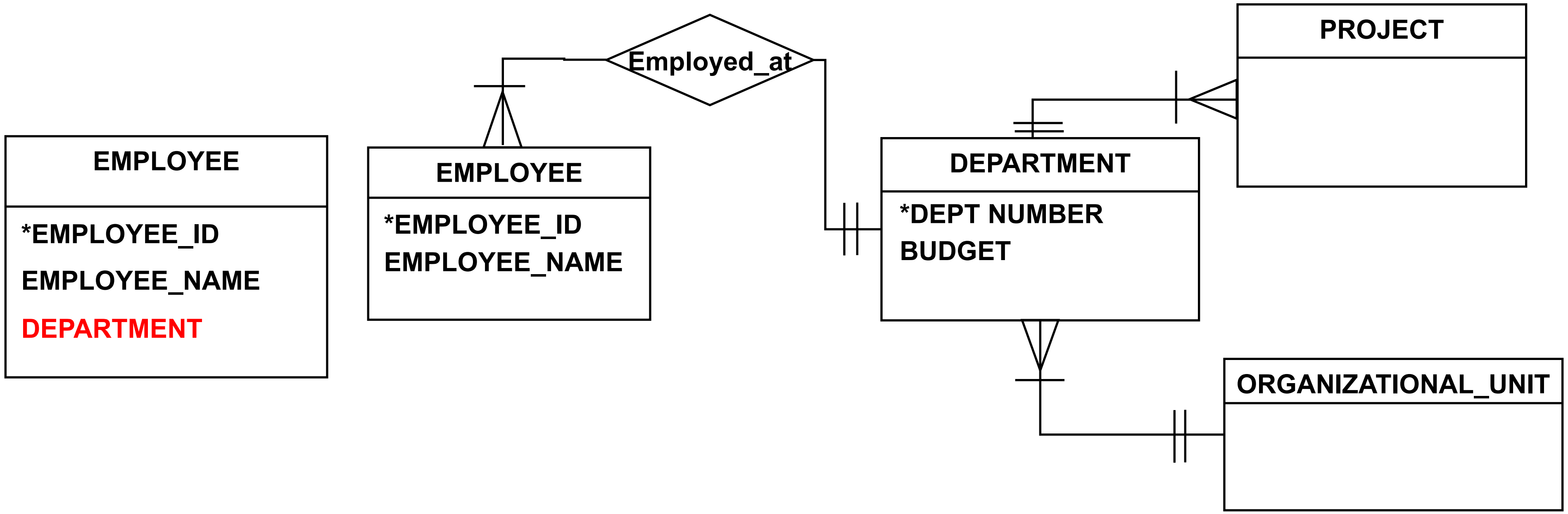


Source: Fig 2.15: Hoffer, Ramesh, Topi, "Modern database management," 10th ed, 2010.

3. Attribute vs.

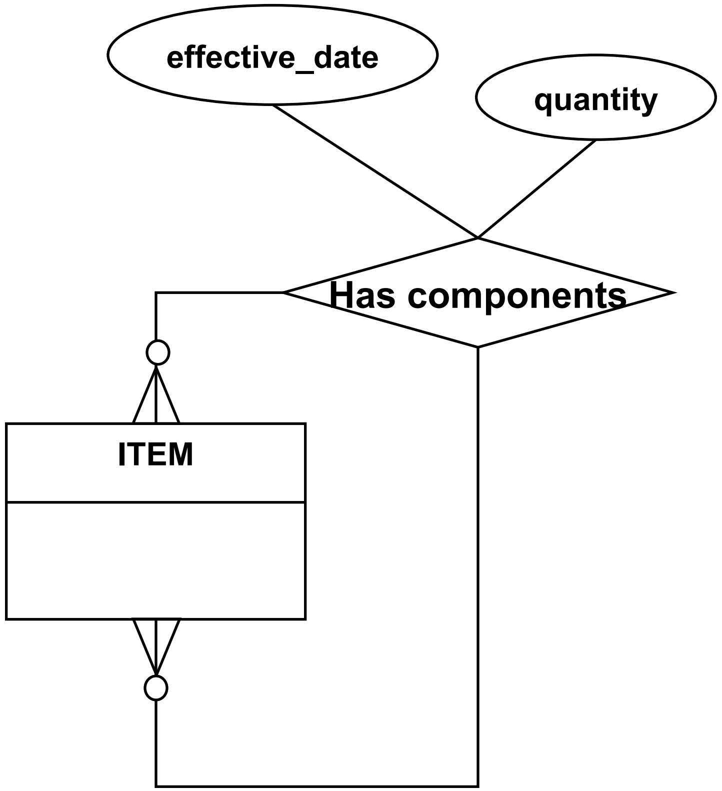
EMPLOYEE
*EMPLOYEE_ID
EMPLOYEE_NAME
DEPARTMENT

3. Attribute vs. Multiple Entities



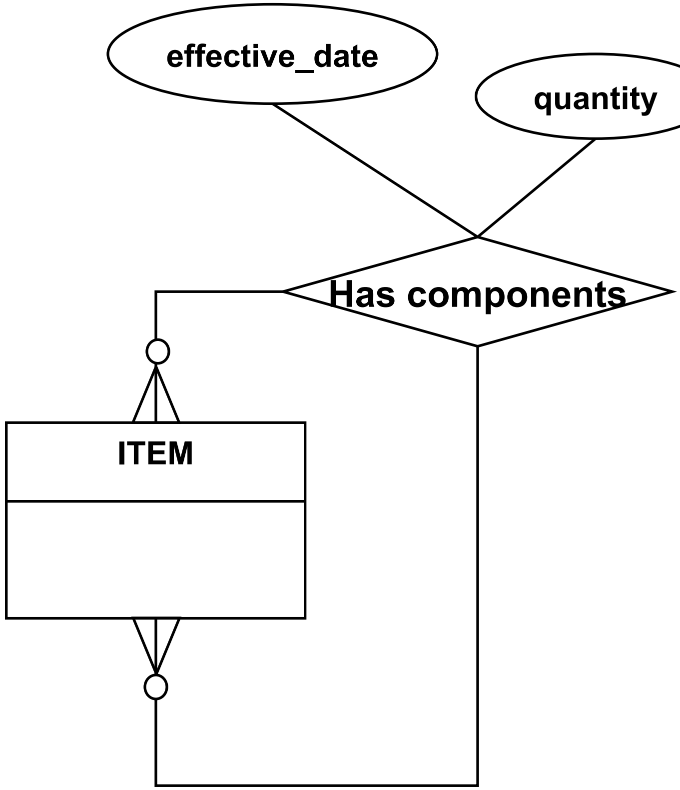
Bill-of-materials (BOM) structure

Relationship

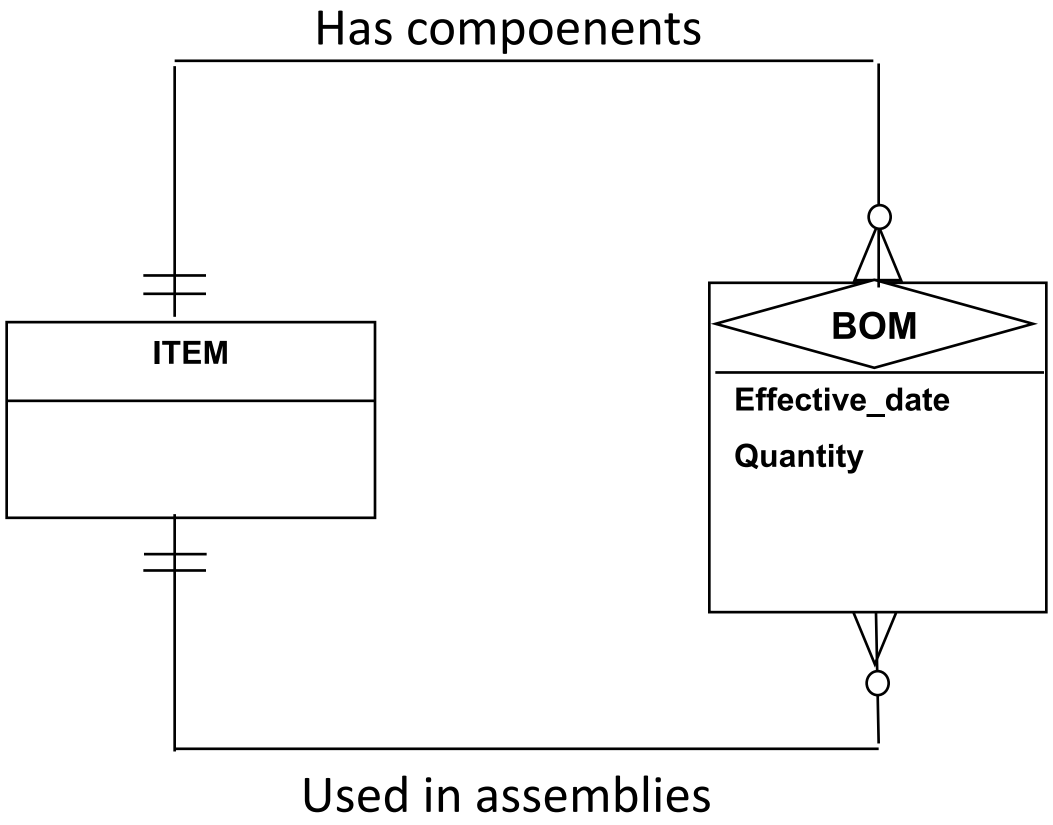


Bill-of-materials (BOM) structure

Relationship



Associative entity

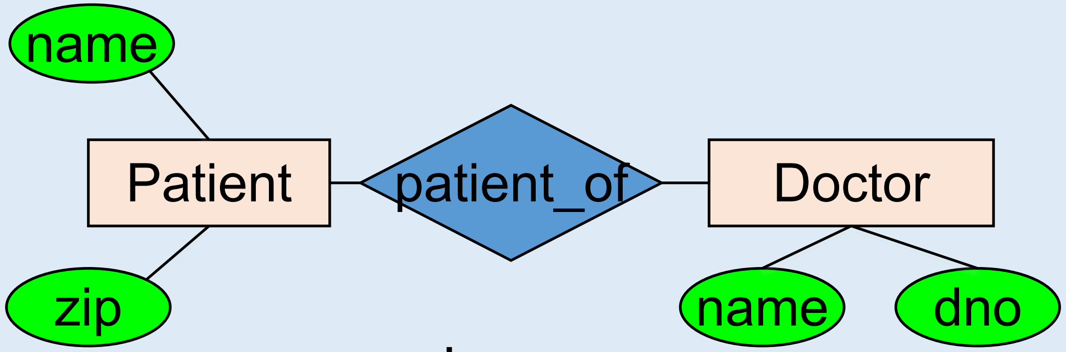


"Relational modeling": From ERDs to Relations

Data modeling and Database Design Process

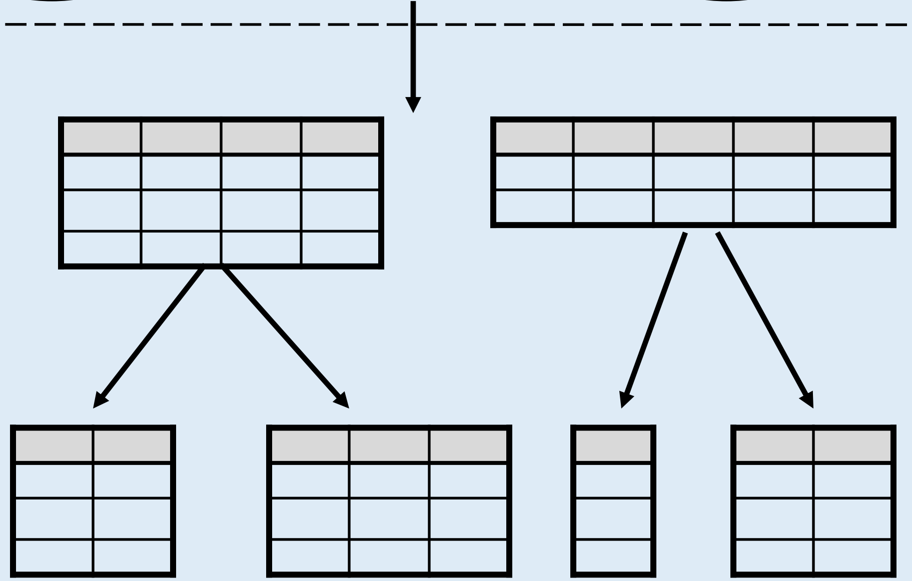
1. ER Diagram

Conceptual Model:
("technology independent")
describe main data items



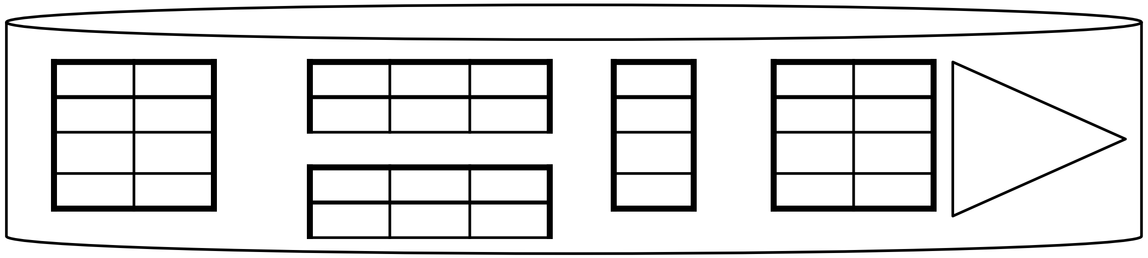
2. Relational Database Design

Logical Model
("for relational databases"):
Tables, Constraints
Functional Dependencies
Normalization:
Eliminates anomalies



3. Database Implementation

Physical Model
Physical storage details
Result: Physical Schema

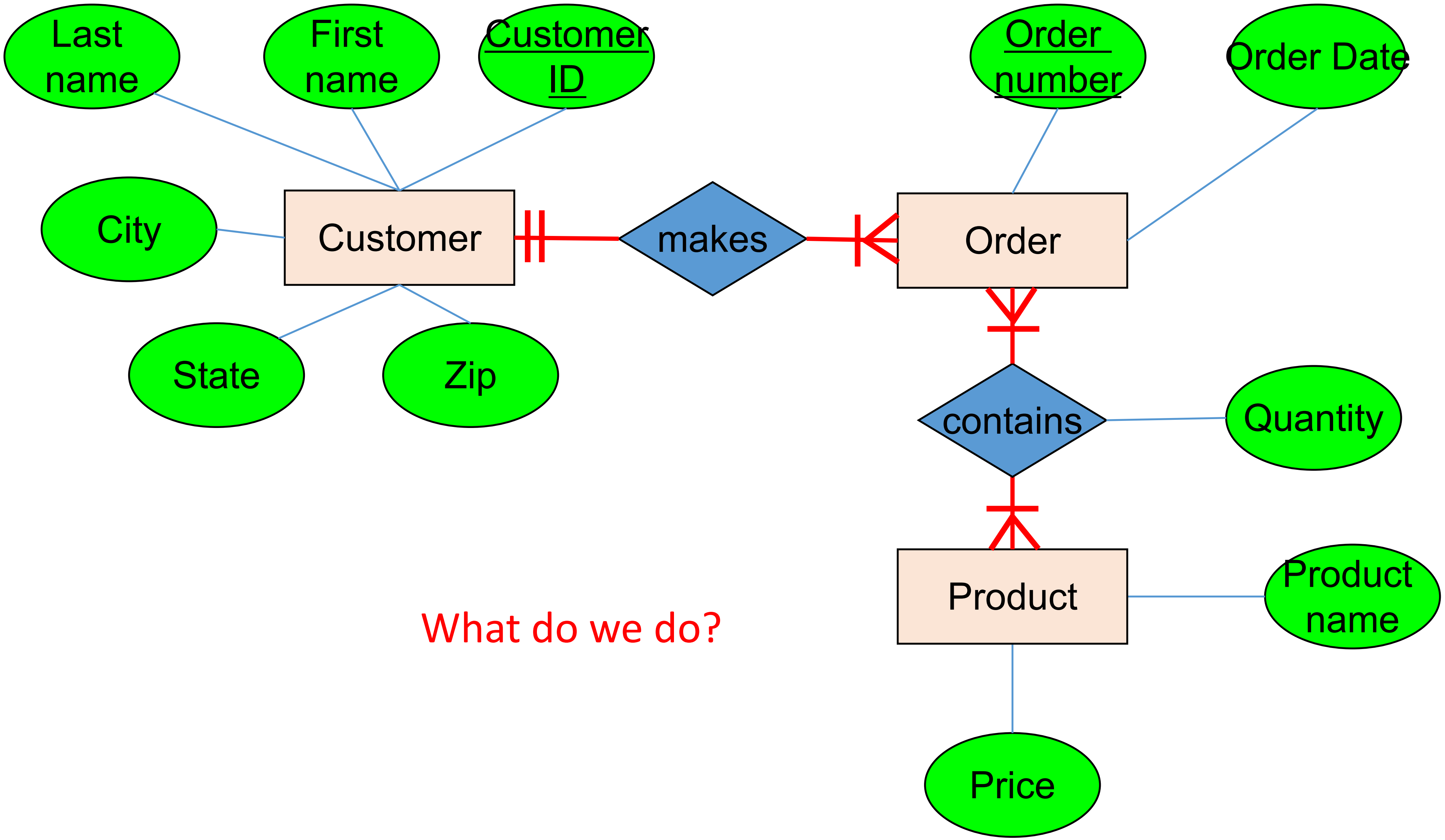


From E/R Diagrams to Relational Schema

- Key concept
  - Entity sets become relations, Relationships can become relations (tables in RDBMS)
  - Tables are connected with <u>foreign key constraints</u>

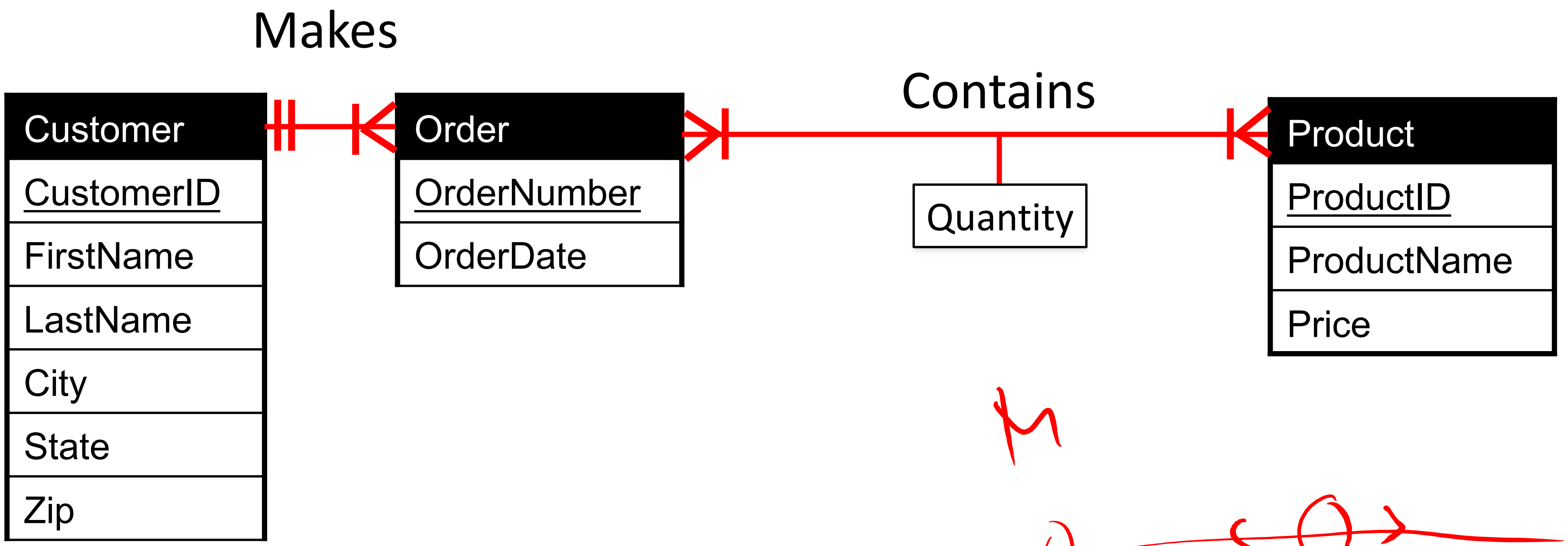
- A database schema
  - A map of the tables and fields (attributes) in the database
  - This is what is implemented in the database management system
  - Part of the "design" process

ERD (Chen notation)

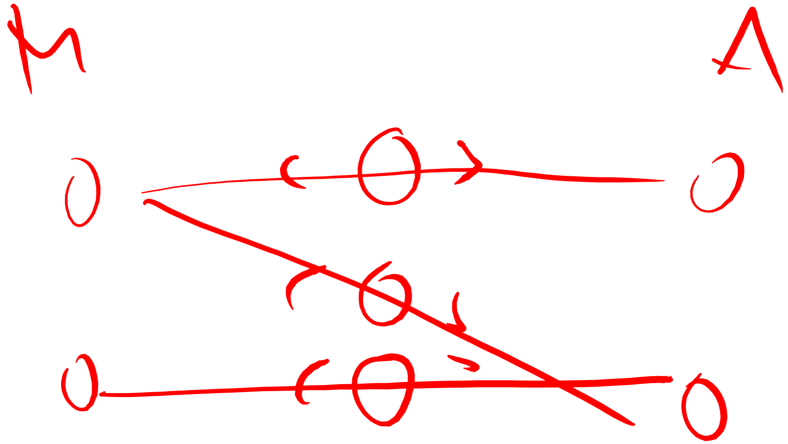


What do we do?

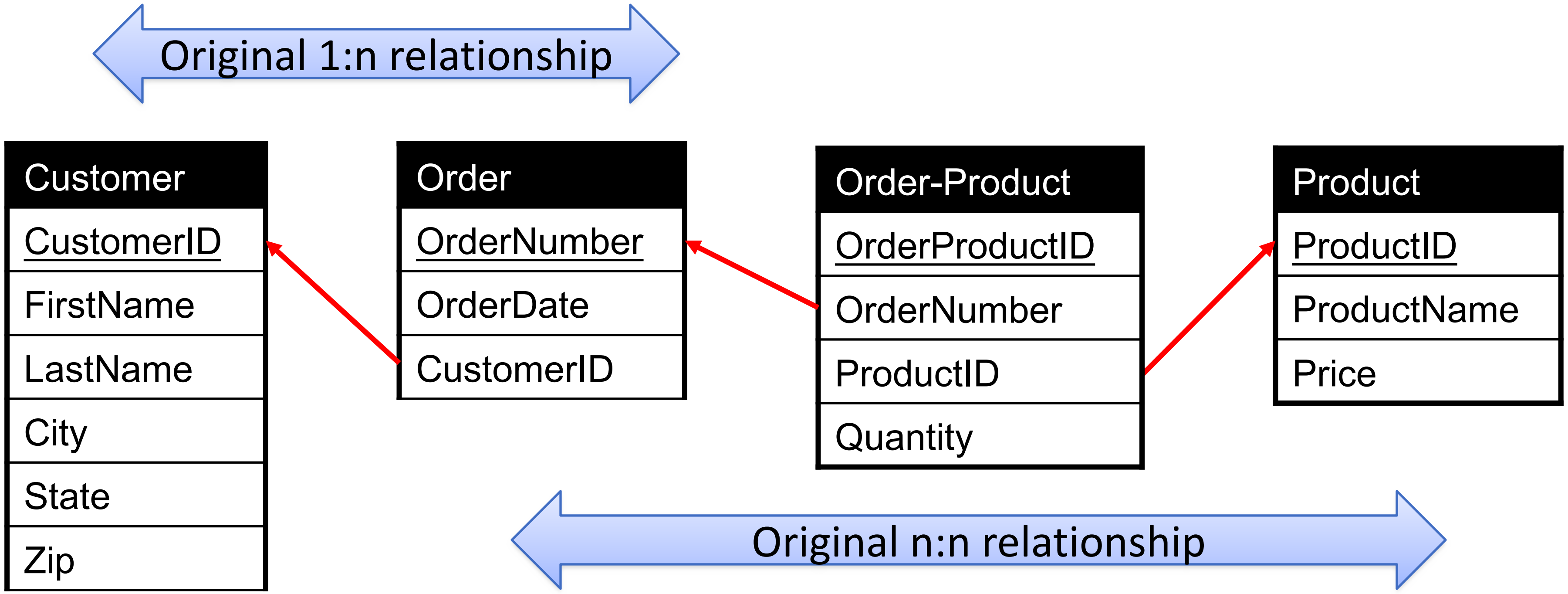
ERD (UML / crow-feet notation)



What do we do?

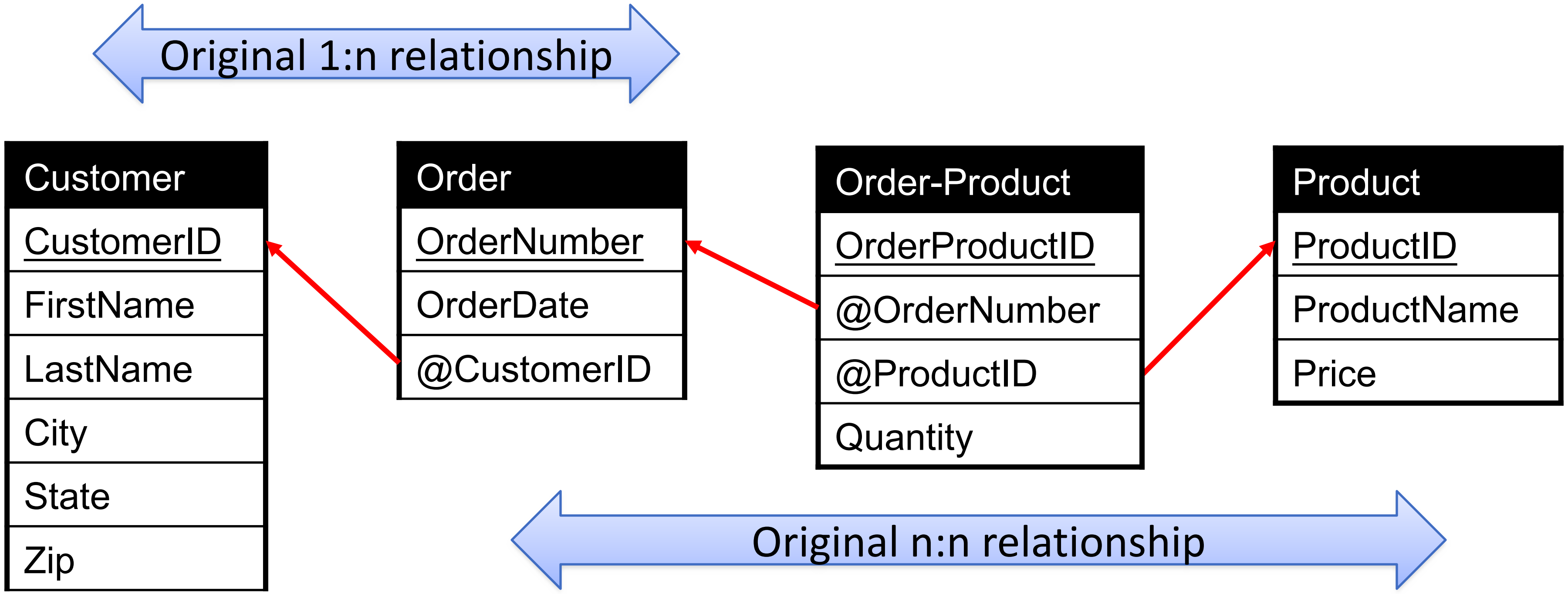


Relational schema (order database)



- Order-Product is a decomposed many-to-many relationship
  - Order-Product has a 1:n relationship with Order and Product
  - Now an order can have multiple products, and a product can be associated with multiple orders

Relational schema (order database)



- Order-Product is a decomposed many-to-many relationship
  - Order-Product has a 1:n relationship with Order and Product
  - Now an order can have multiple products, and a product can be associated with multiple orders

The Rules

- Create a table for every **entity**
- Create table fields for every entity's **attributes**
- Implement **relationships** between the tables
  - 1:1 relationships: primary key field of one table put into other table as foreign key field
  - <u>1:many relationships</u>: primary key field of "1" table put into "many" table as foreign key field
  - <u>many:many relationships</u>:
    - Create new table!
    - 1:many relationships with original table

CAST in our IMDB movie database