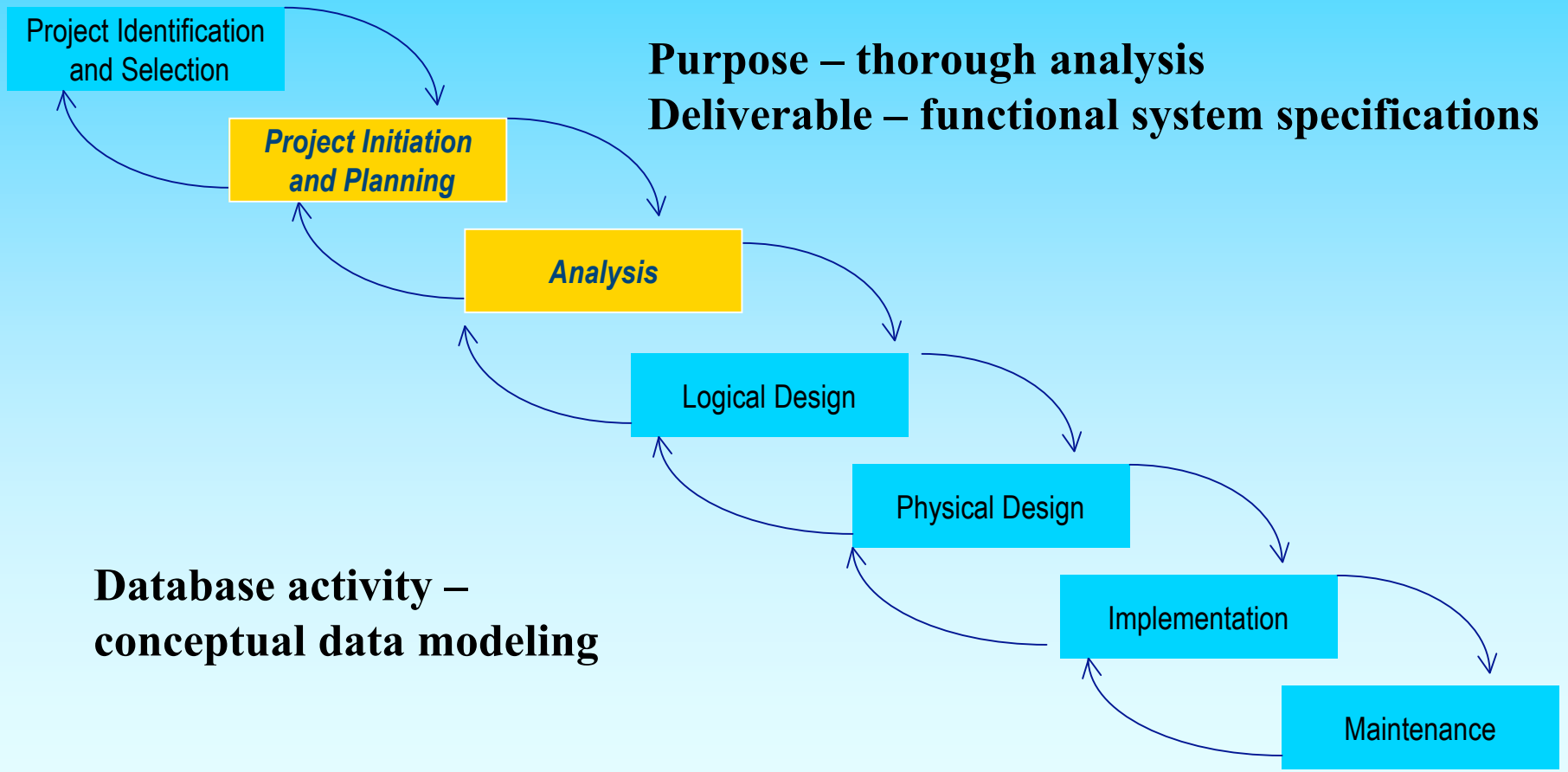
Chapter 3: Modeling Data in the Organization

Modern Database Management

7th Edition

Jeffrey A. Hoffer, Mary B. Prescott, Fred R. McFadden

SDLC Revisited – Data Modeling is an Analysis Activity (figures 2-4, 2-5)

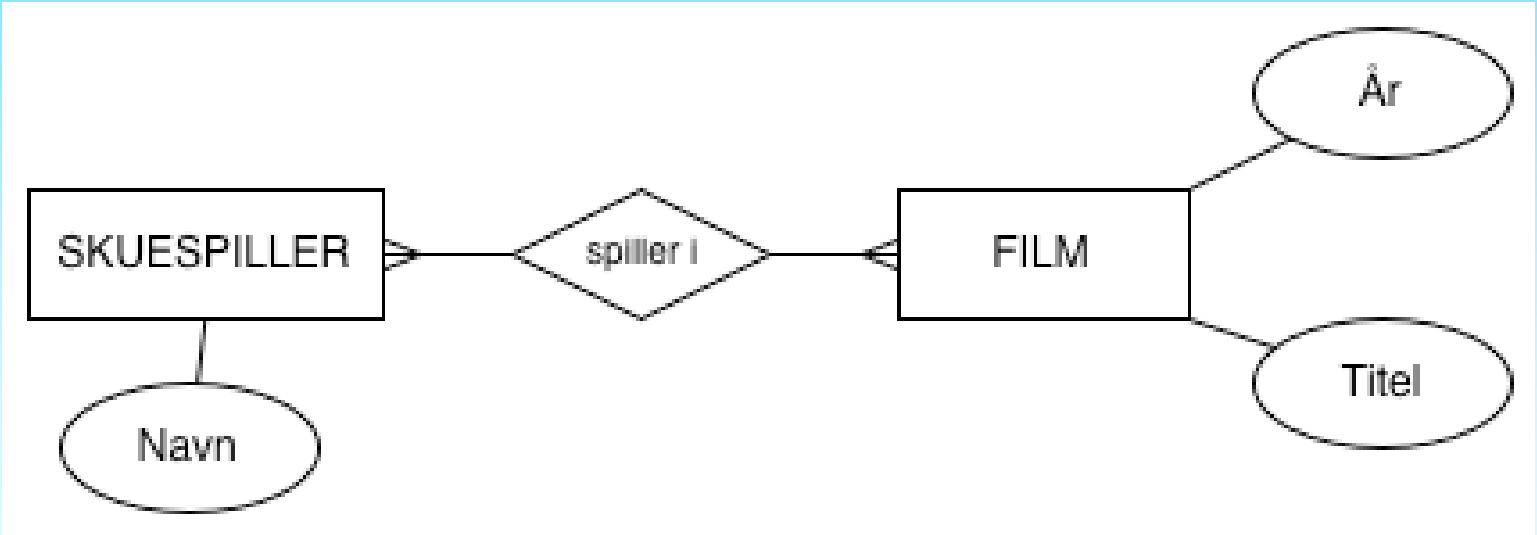


Purpose – thorough analysis
Deliverable – functional system specifications

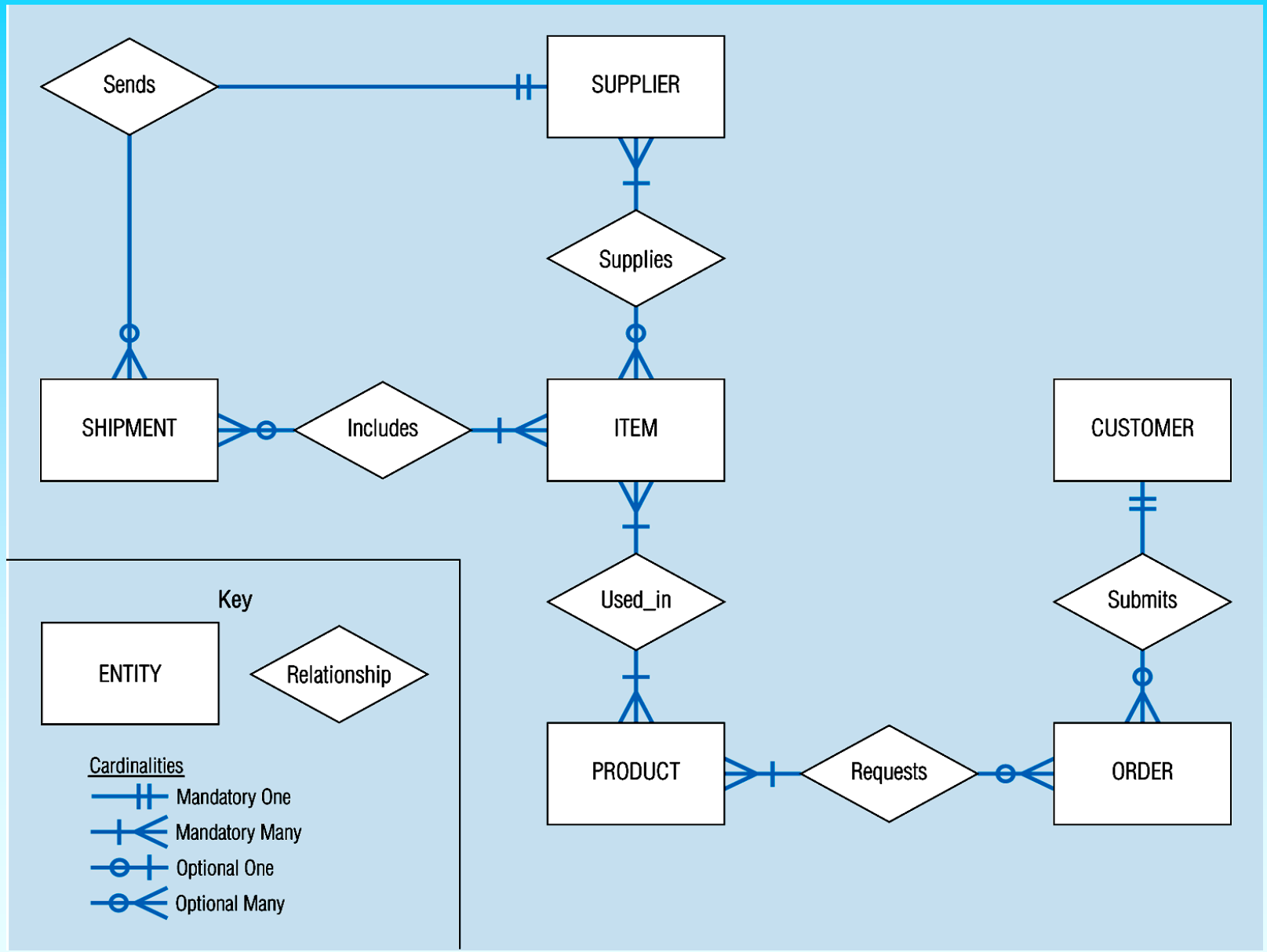
Database activity – conceptual data modeling

Simpelt E-R diagram

Lavet i Omnigraffle, som man selv gratis kan installere på Mac.
ITU Maclab er i 2A54.



Sample E-R Diagram (Figure 3-1)

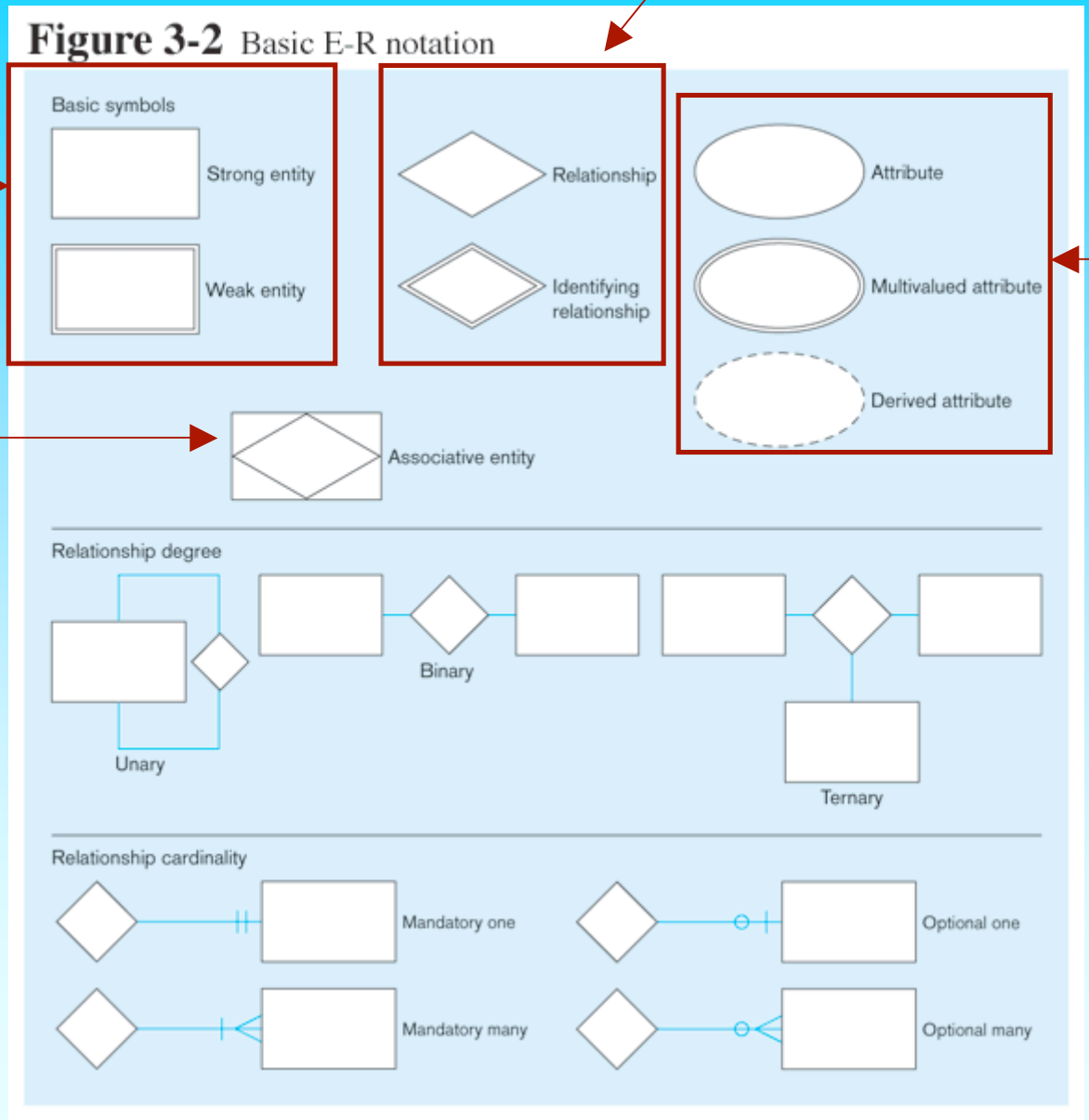


Relationship symbols

Entity symbols

Attribute symbols

A special entity that is also a relationship



Relationship degrees specify number of entity types involved

Relationship cardinalities specify how many of each entity type is allowed

What Should an Entity Be?

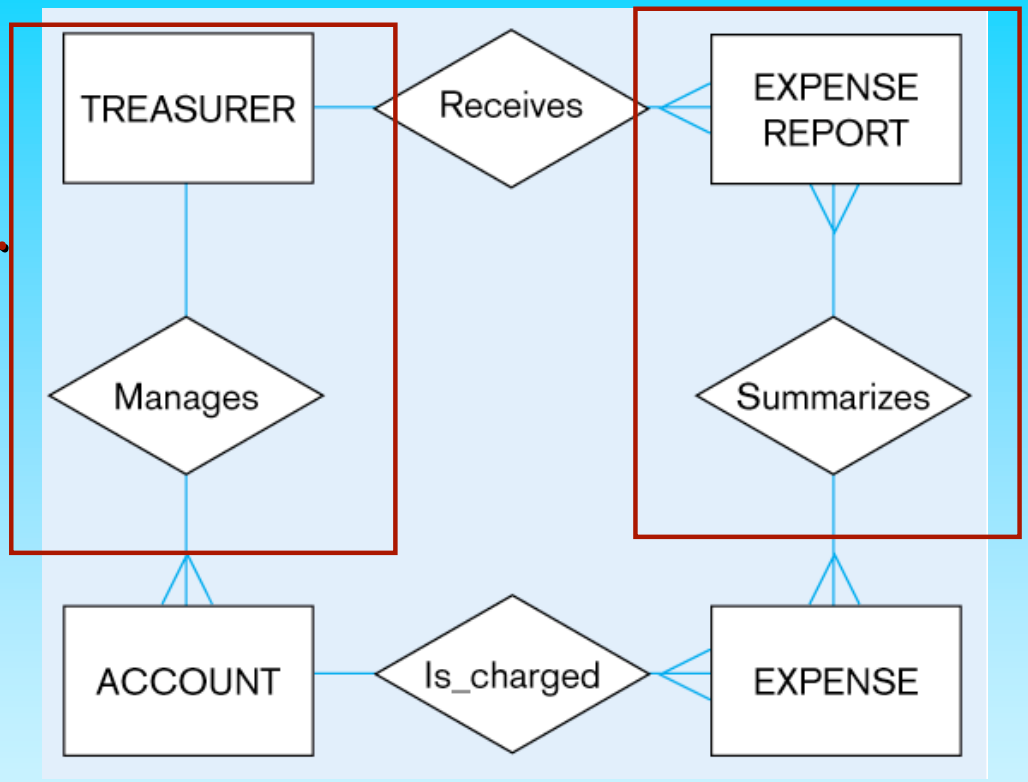
- SHOULD BE:
  - An object that will have many instances in the database
  - An object that will be composed of multiple attributes
  - An object that we are trying to model
- SHOULD NOT BE:
  - A user of the database system
  - An output of the database system (e.g. a report)

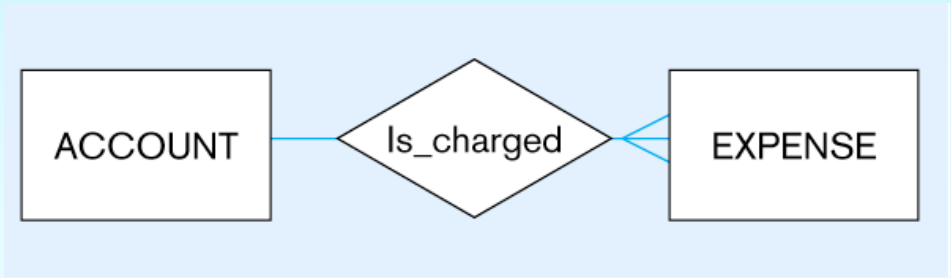
Figure 3-4

Inappropriate entities

System user

System output





Appropriate entities

Problemsession (5-10 min)

- Vi vil designe en database til en lille butik med information om kunderne, order og kredit:
  - Navne og adresser
  - Telefonnumre
  - Kundetyper (privat eller forretning)
  - Aktuelle ordrer
  - Kundenumre
  - Maximum kredit
  - Aktuel kredit

Tegn et muligt E-R diagram for databasen.

More on Relationships

- Relationship Types vs. Relationship Instances
  - The relationship type is modeled as the diamond and lines between entity types…the instance is between specific entity instances
- Relationships can have attributes
  - These describe features pertaining to the association between the entities in the relationship
- Two entities can have more than one type of relationship between them (multiple relationships)
- Associative Entity – combination of relationship and entity

Figure 3-10a Relationship type and instances - Relationship type (Completes)

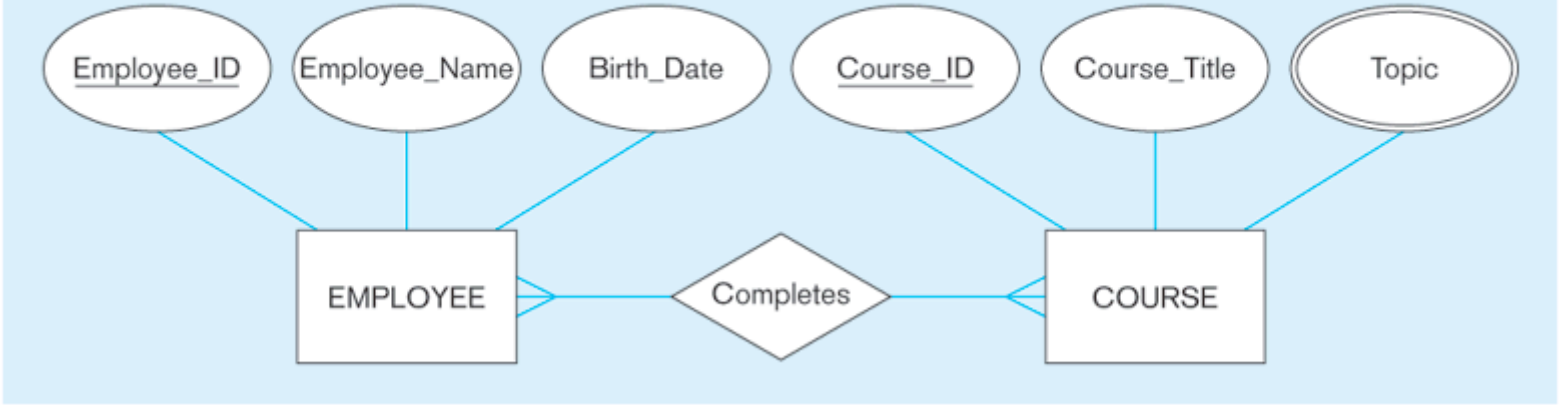
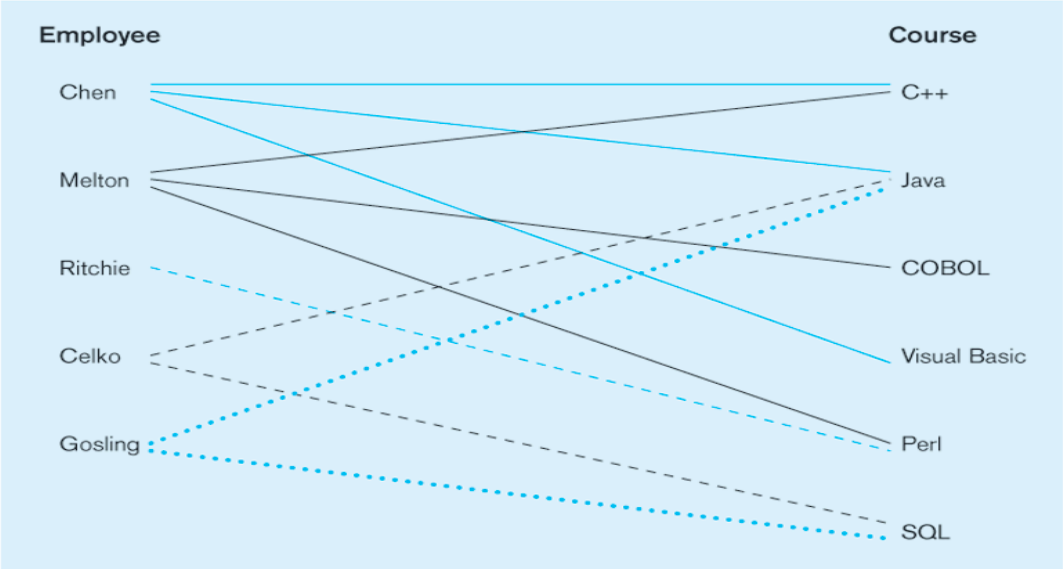


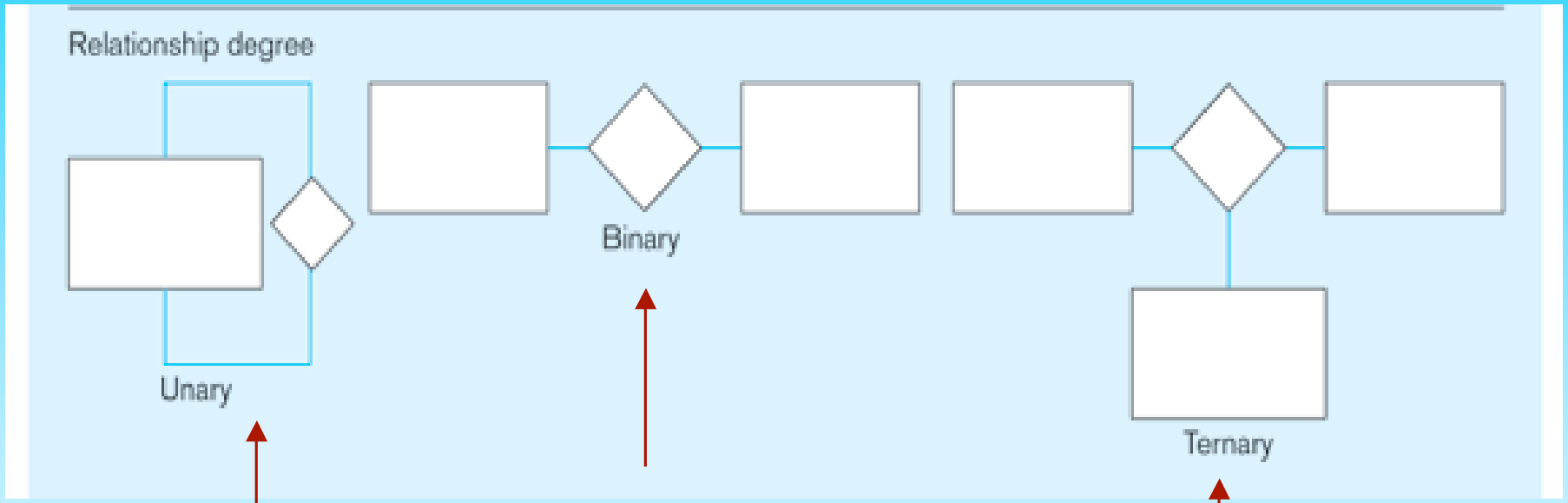
Figure 3-10b Relationship type and instances - Relationship instances



Degree of Relationships

- Degree of a relationship is the number of entity types that participate in it
  - Unary Relationship
  - Binary Relationship
  - Ternary Relationship

Degree of relationships – from Figure 3-2



One entity related to another of the same entity type

Entities of two different types related to each other

Entities of three different types related to each other

Cardinality of Relationships

- One-to-One
  - Each entity in the relationship will have exactly one related entity
- One-to-Many
  - An entity on one side of the relationship can have many related entities, but an entity on the other side will have a maximum of one related entity
- Many-to-Many
  - Entities on both sides of the relationship can have many related entities on the other side

Cardinality Constraints

- Cardinality Constraints - the number of instances of one entity that can or must be associated with each instance of another entity
- Minimum Cardinality
  - If zero, then optional
  - If one or more, then mandatory
- Maximum Cardinality
  - The maximum number

Figure 3-12a Examples of relationships of different degrees - Unary relationships

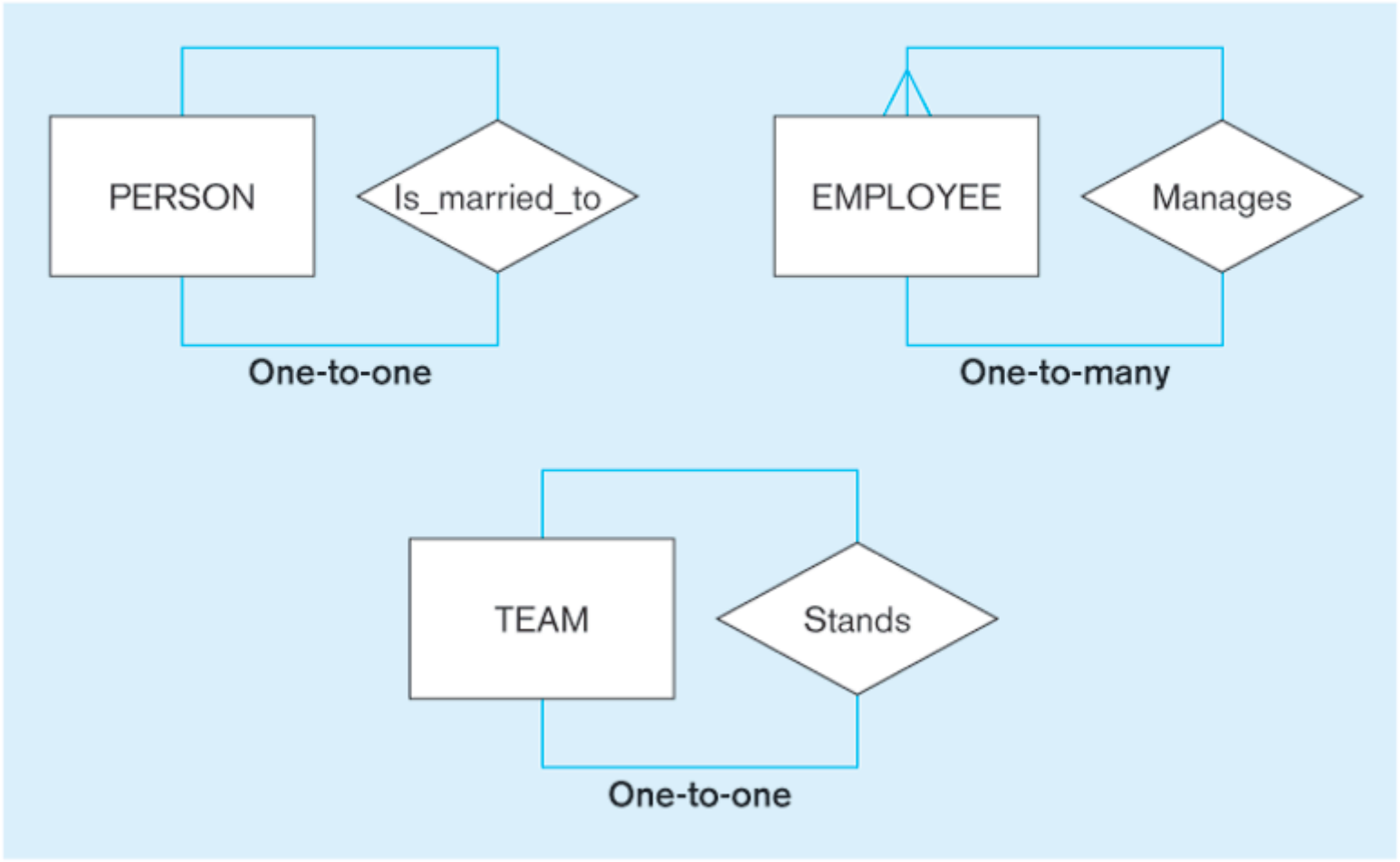
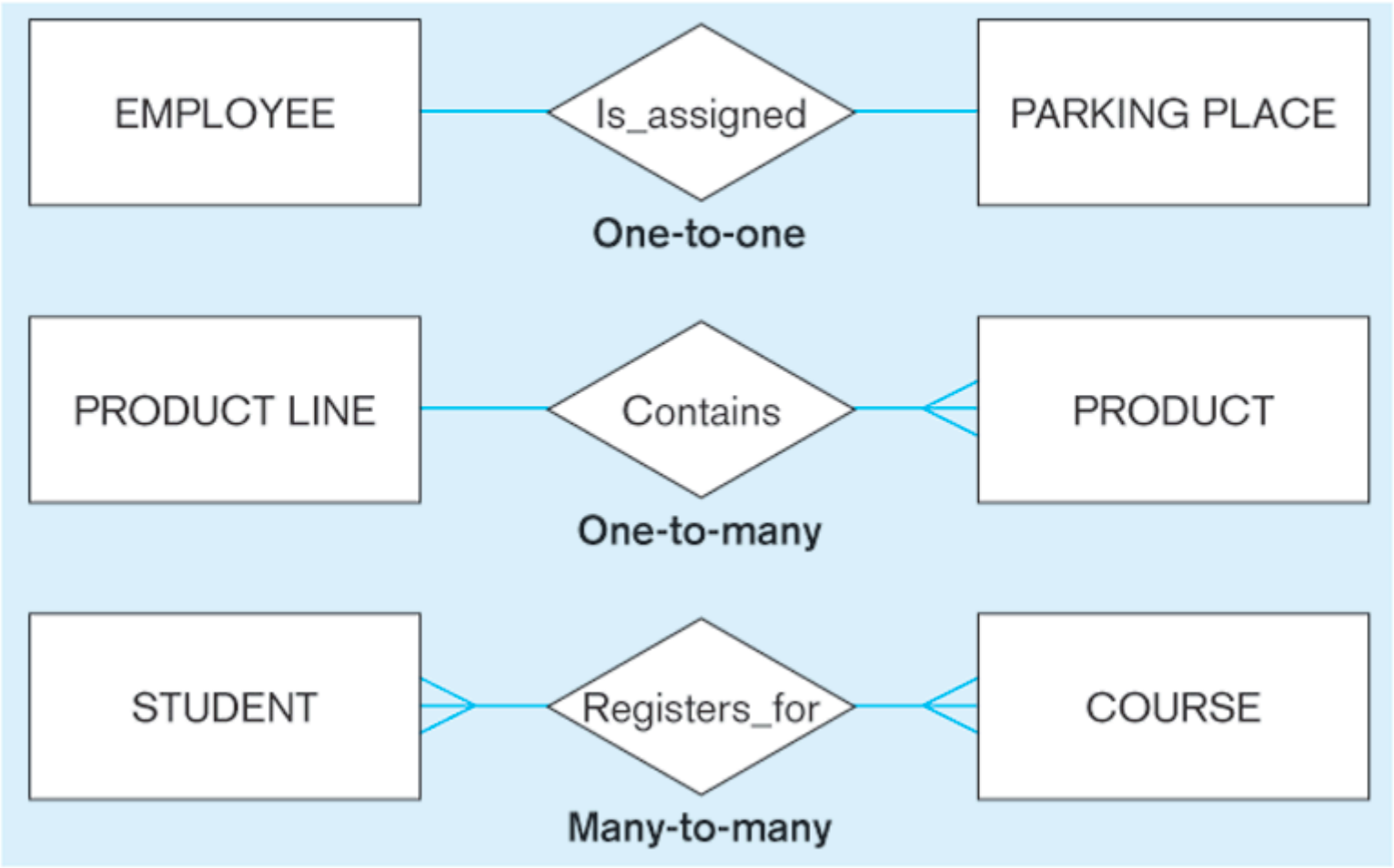
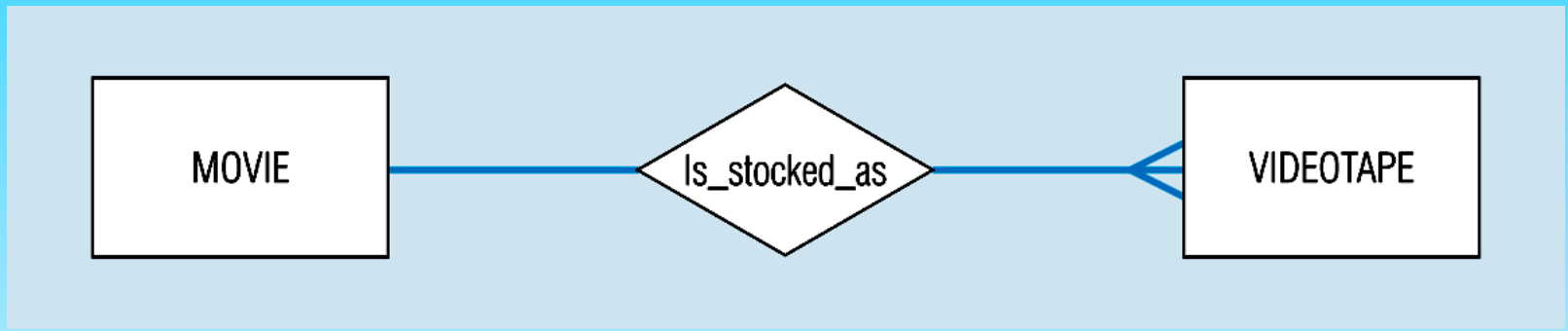


Figure 3-12b Examples of relationships of different degrees - Binary relationships



Basic relationship with only maximum cardinalities showing – Figure 3-16a



Mandatory minimum cardinalities – Figure 3-17a

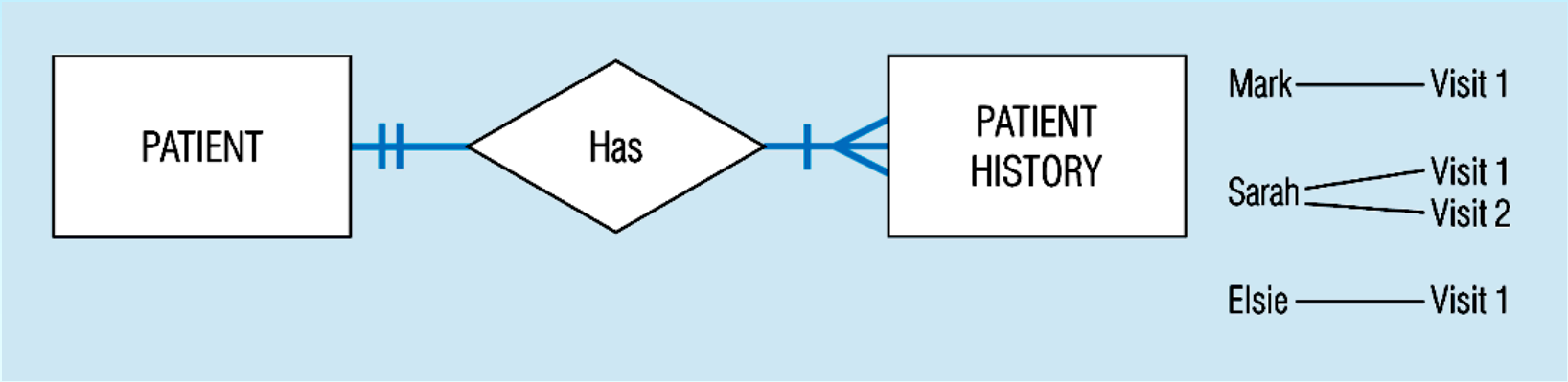


Figure 3-17c
Optional cardinalities with unary degree, one-to-one relationship

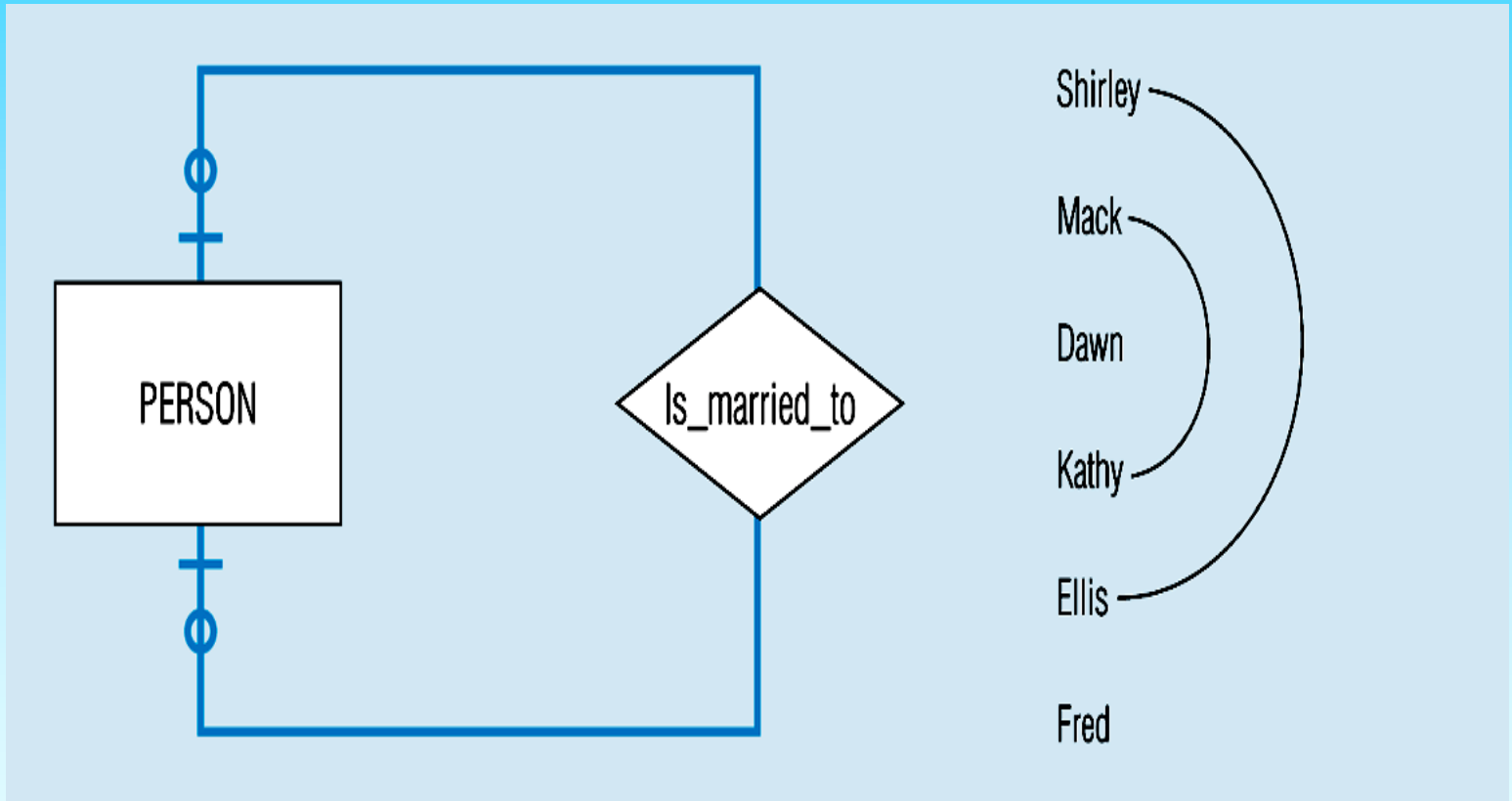
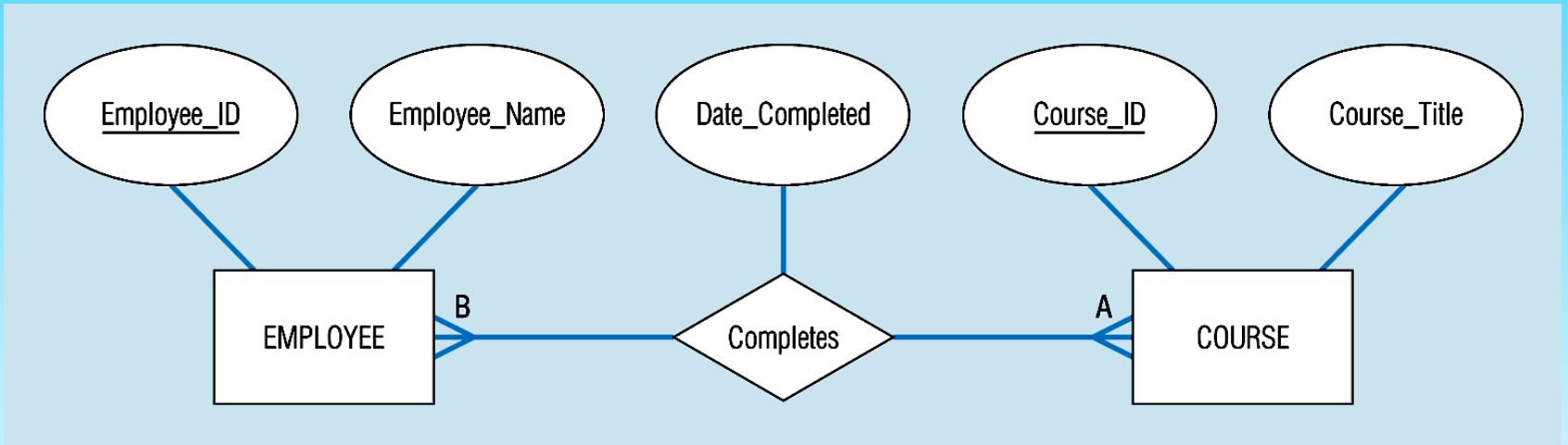


Figure 3-11a A binary relationship with an attribute



Here, the date completed attribute pertains specifically to the employee's completion of a course…it is an attribute of the *relationship*

Figure 3-12c -- A ternary relationship with attributes

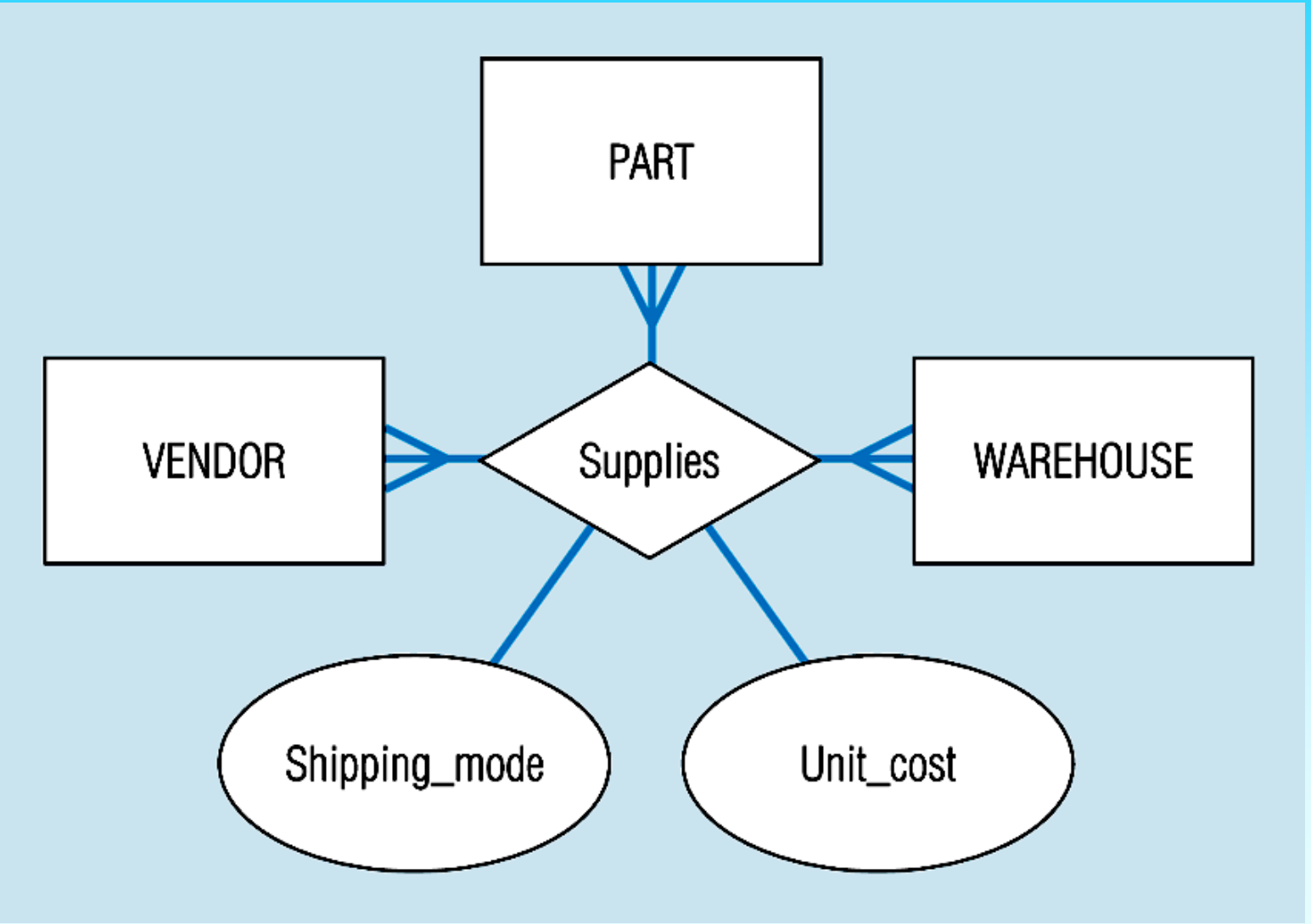
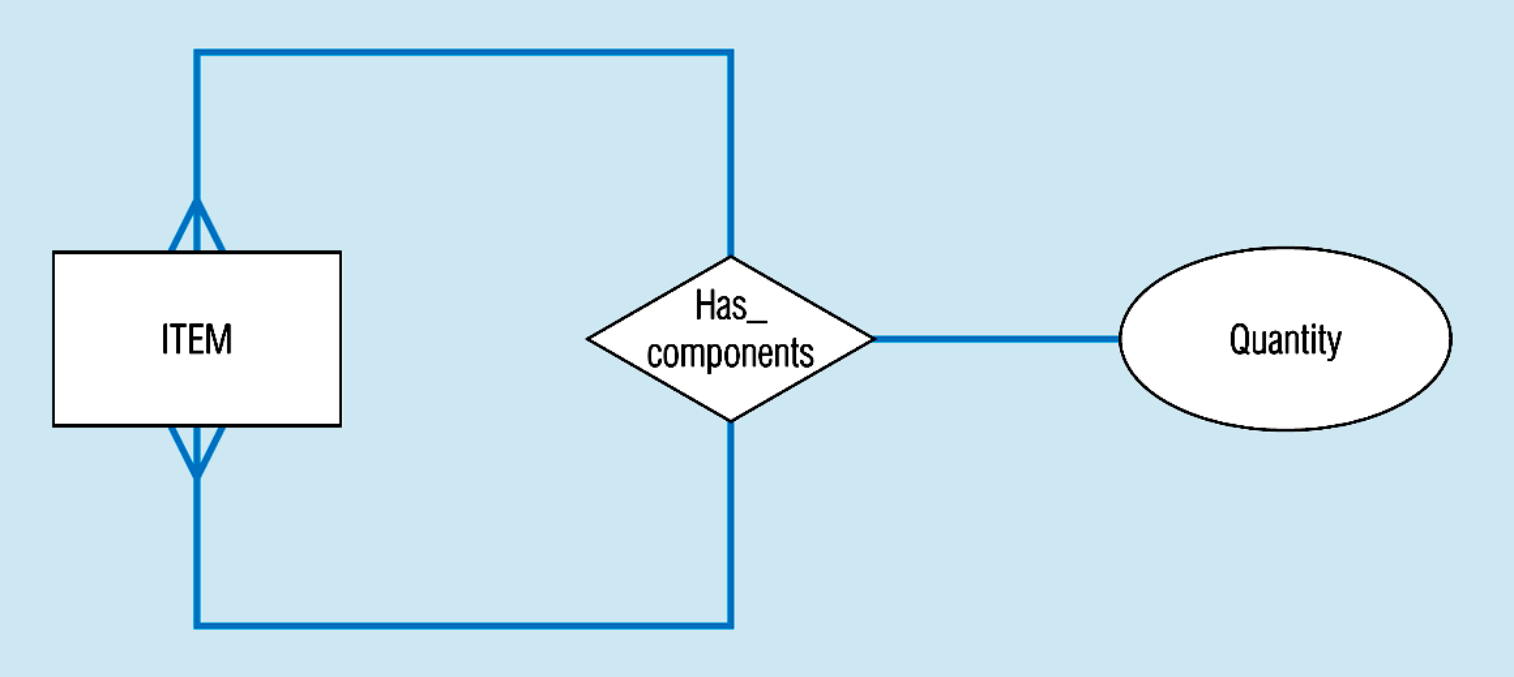


Figure 3-13a – A unary relationship with an attribute.
This has a many-to-many relationship



Representing a bill-of -materials structure

Figure 3-13b Representing a bill-of-materials structure -
Two ITEM bill-of-materials structure instances

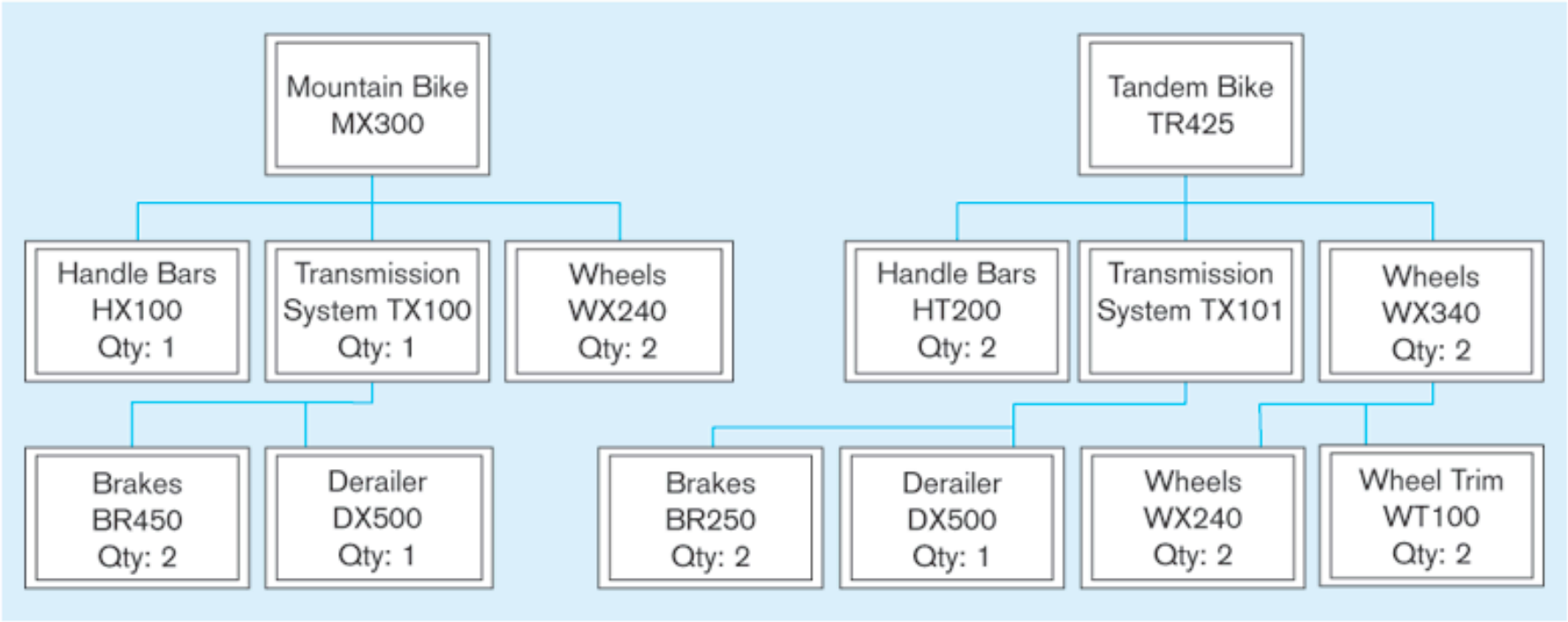
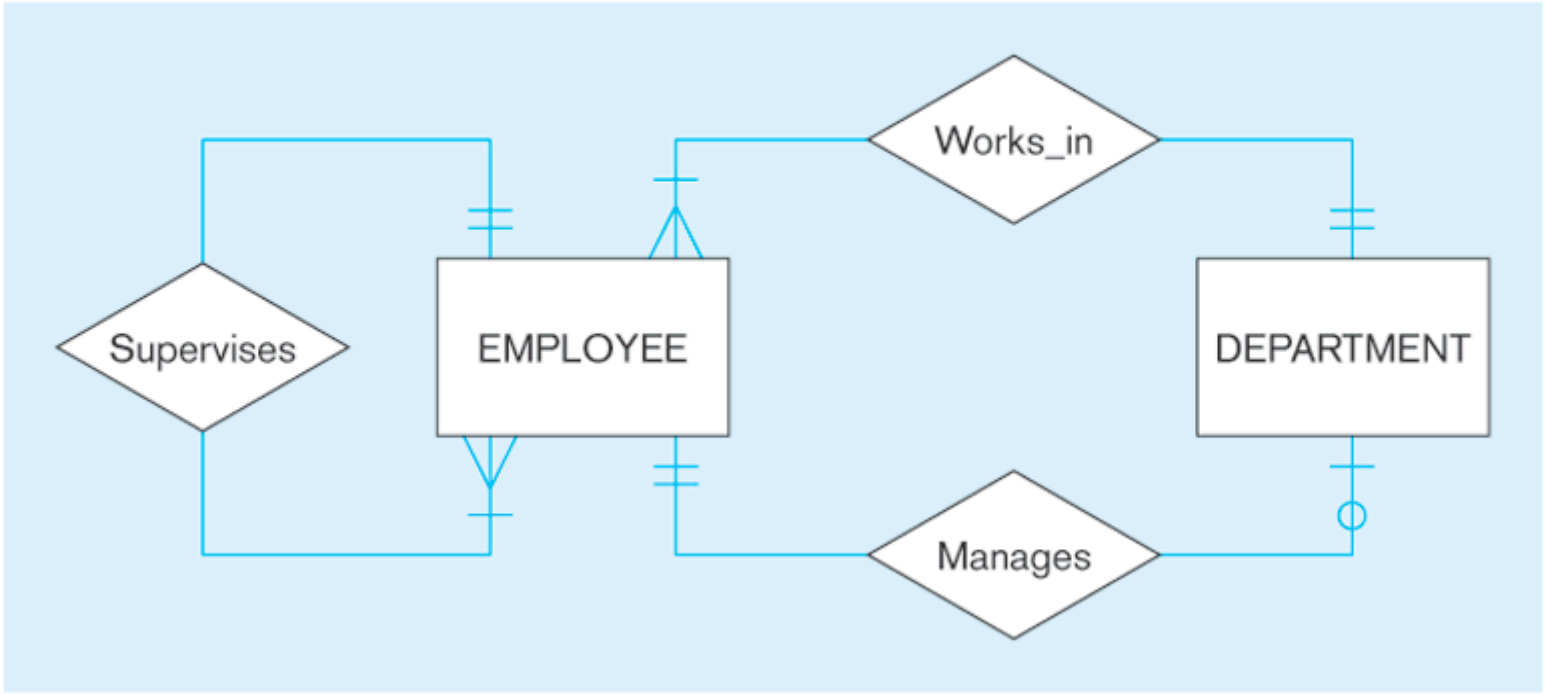
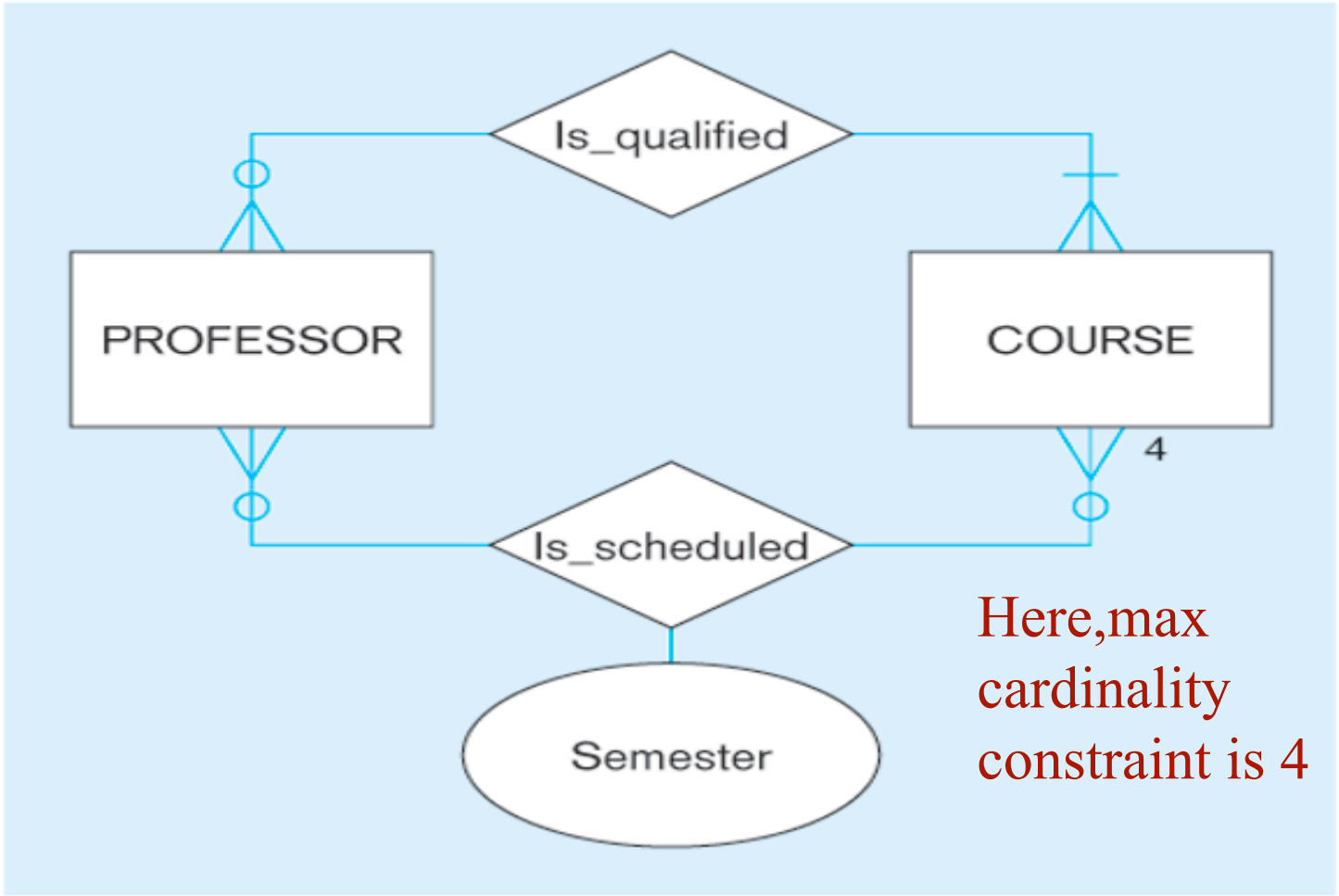


Figure 3-21a Examples of multiple relationships - Employees and departments

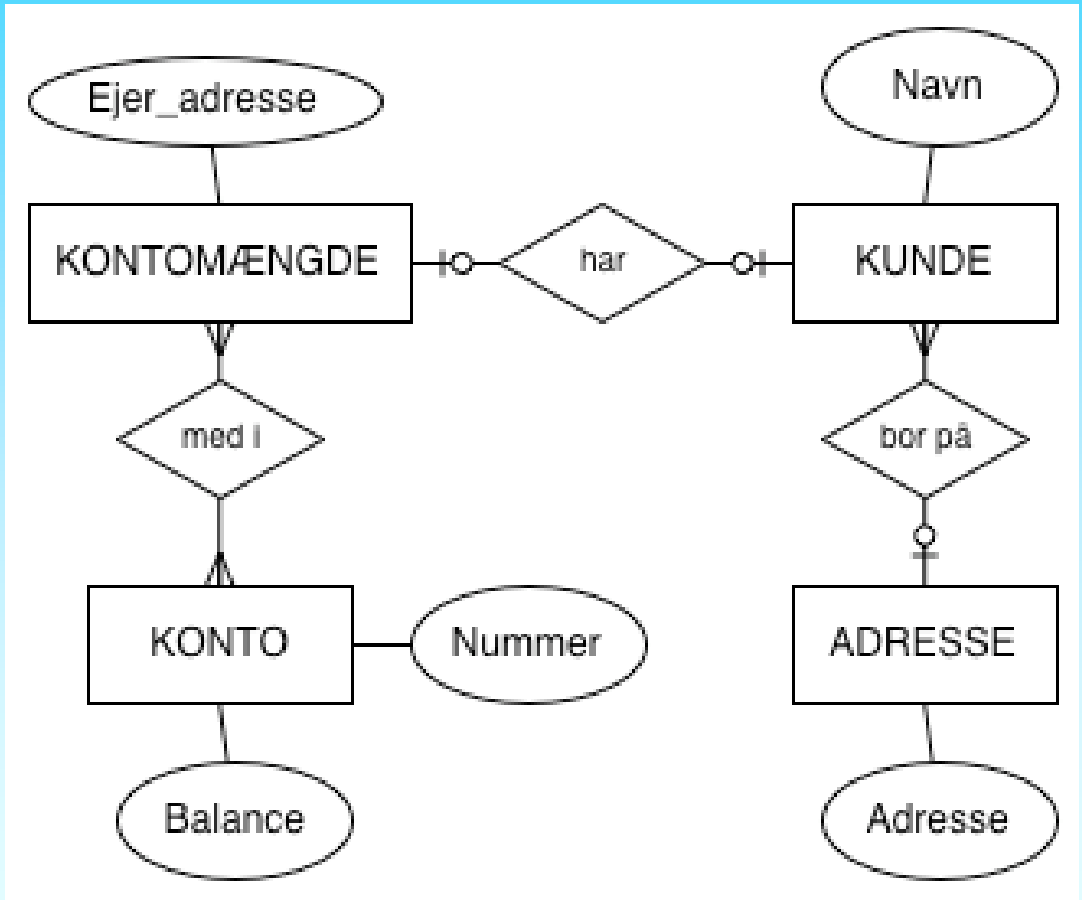


Entities can be related to one another in more than one way

Figure 3-21b Examples of multiple relationships - Professors and courses (fixed upper limit constraint)



E-R diagram til problemsession



Attributes

- Attribute - property or characteristic of an entity type
- Classifications of attributes:
  - Required versus Optional Attributes
  - Simple versus Composite Attribute
  - Single-Valued versus Multivalued Attribute
  - Stored versus Derived Attributes
  - Identifier Attributes

Identifiers (Keys)

- Identifier (Key) - An attribute (or combination of attributes) that uniquely identifies individual instances of an entity type
- Simple Key versus Composite Key
- Candidate Key – an attribute that could be a key…satisfies the requirements for being a key

Characteristics of Identifiers

- Will not change in value
- Will not be null
- No intelligent identifiers (e.g. containing locations or people that might change)
- Substitute new, simple keys for long, composite keys

Figure 3-7 – A **composite** attribute

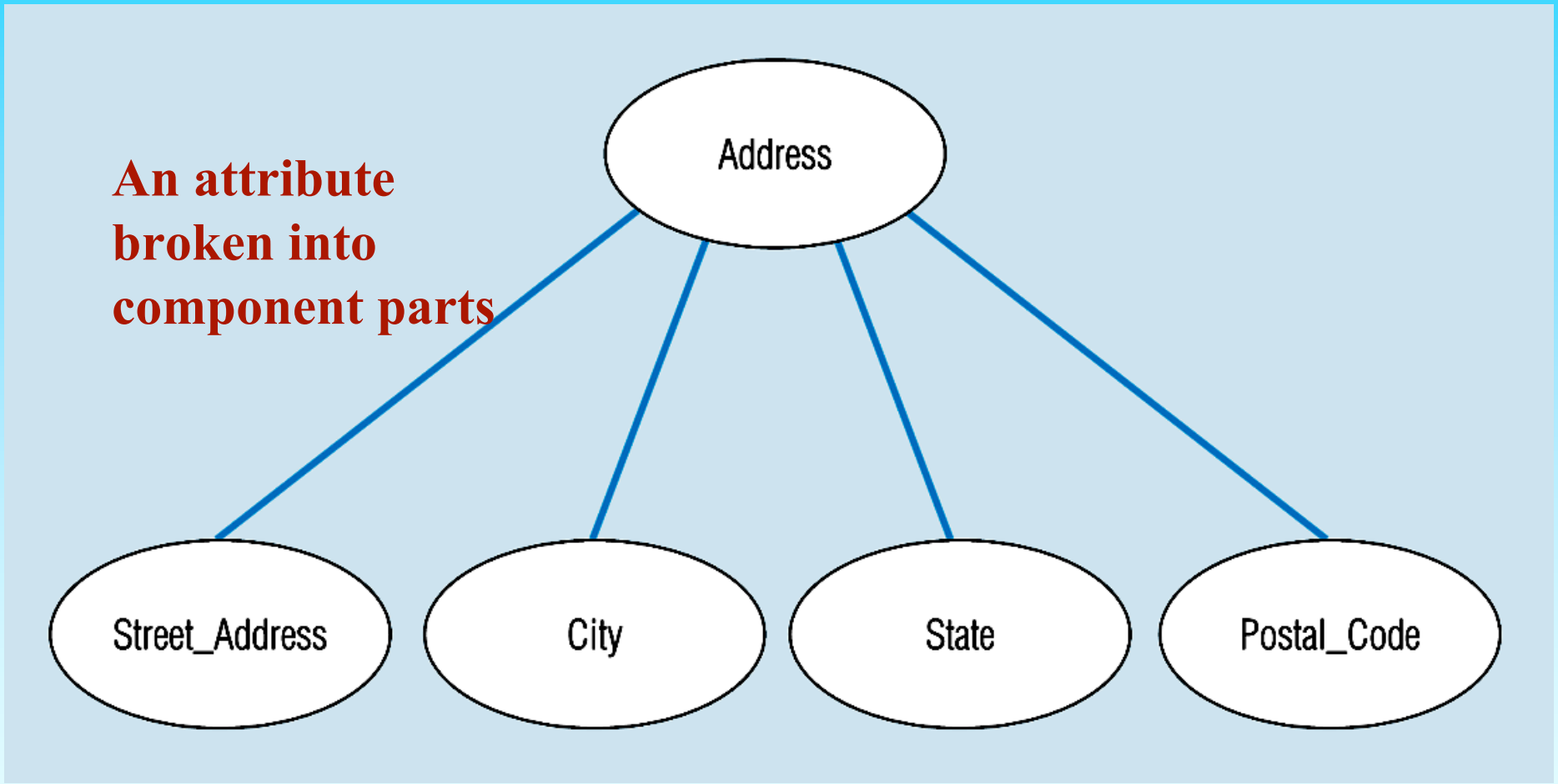


Figure 3-9a – Simple key attribute

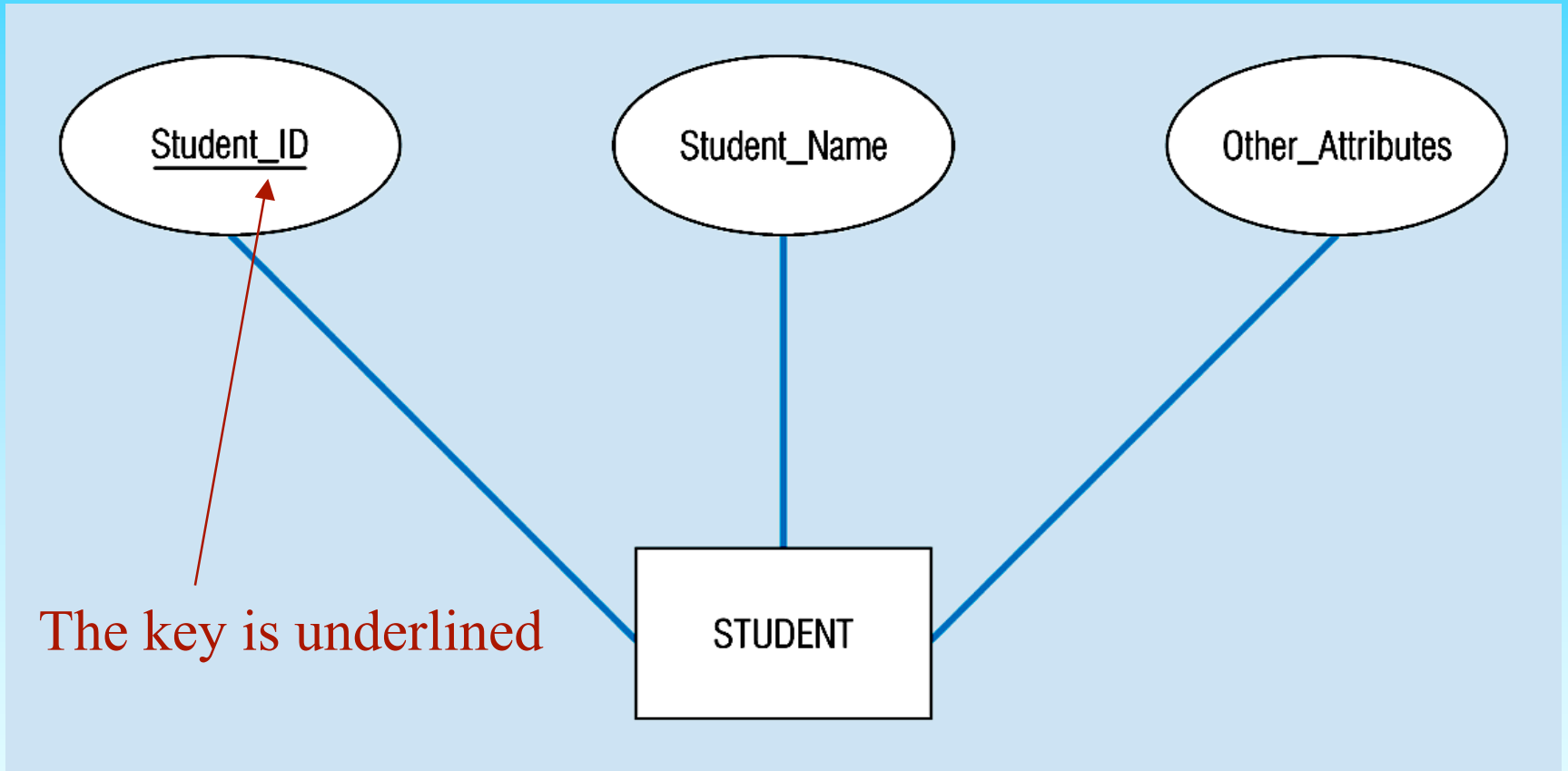


Figure 3-9b – Composite key attribute

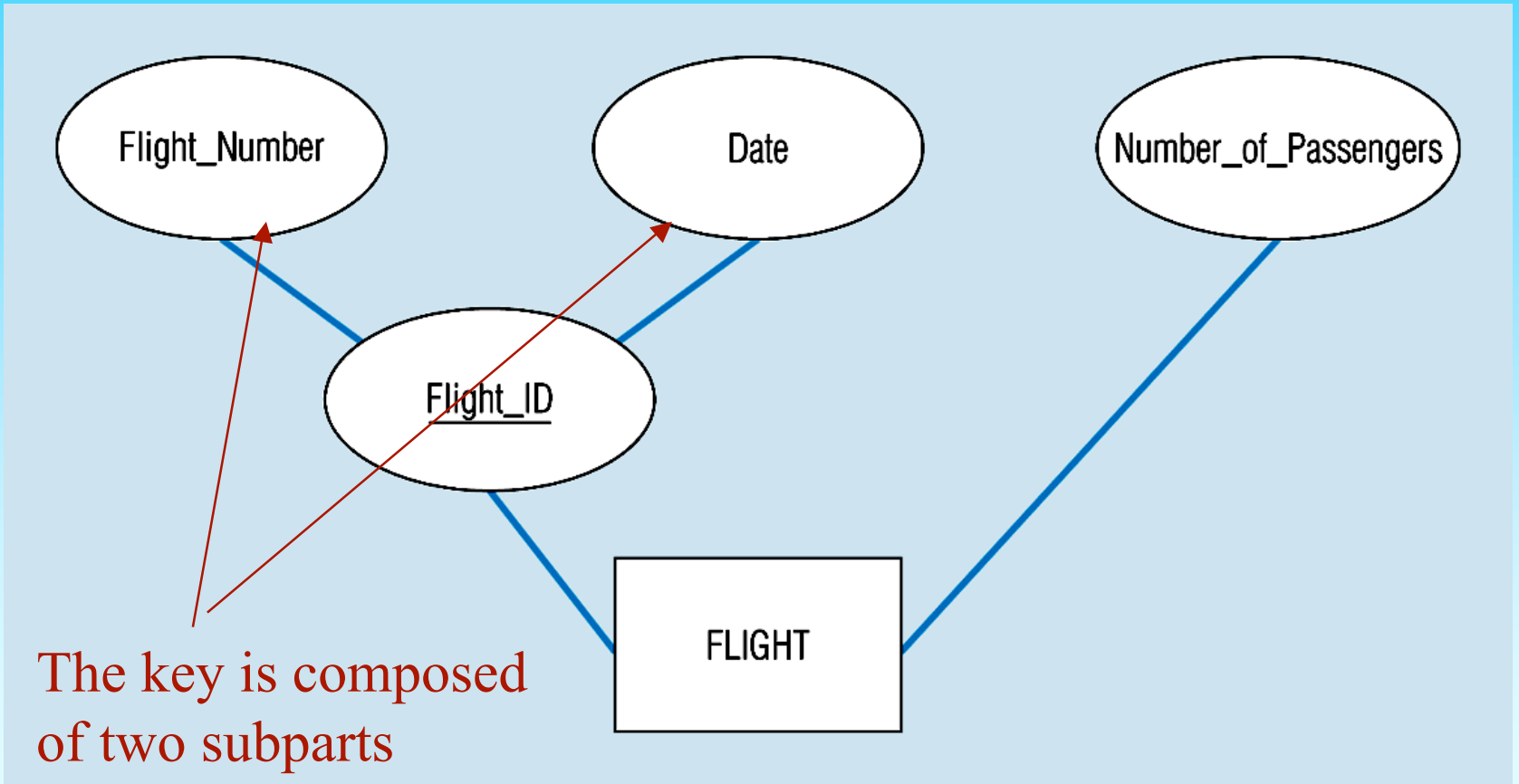


Figure 3-8 – Entity with a multivalued attribute (Skill) and derived attribute (Years_Employed)

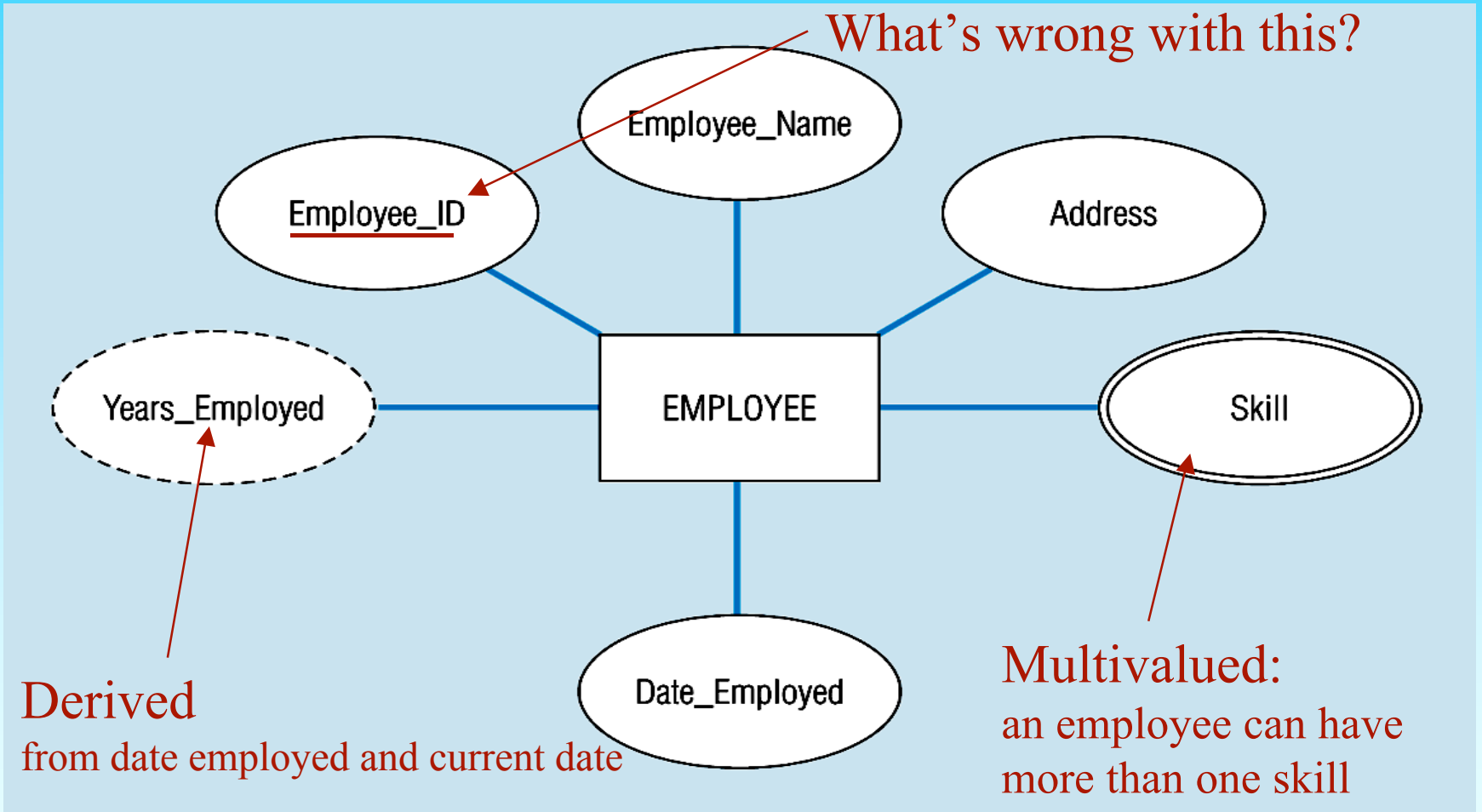
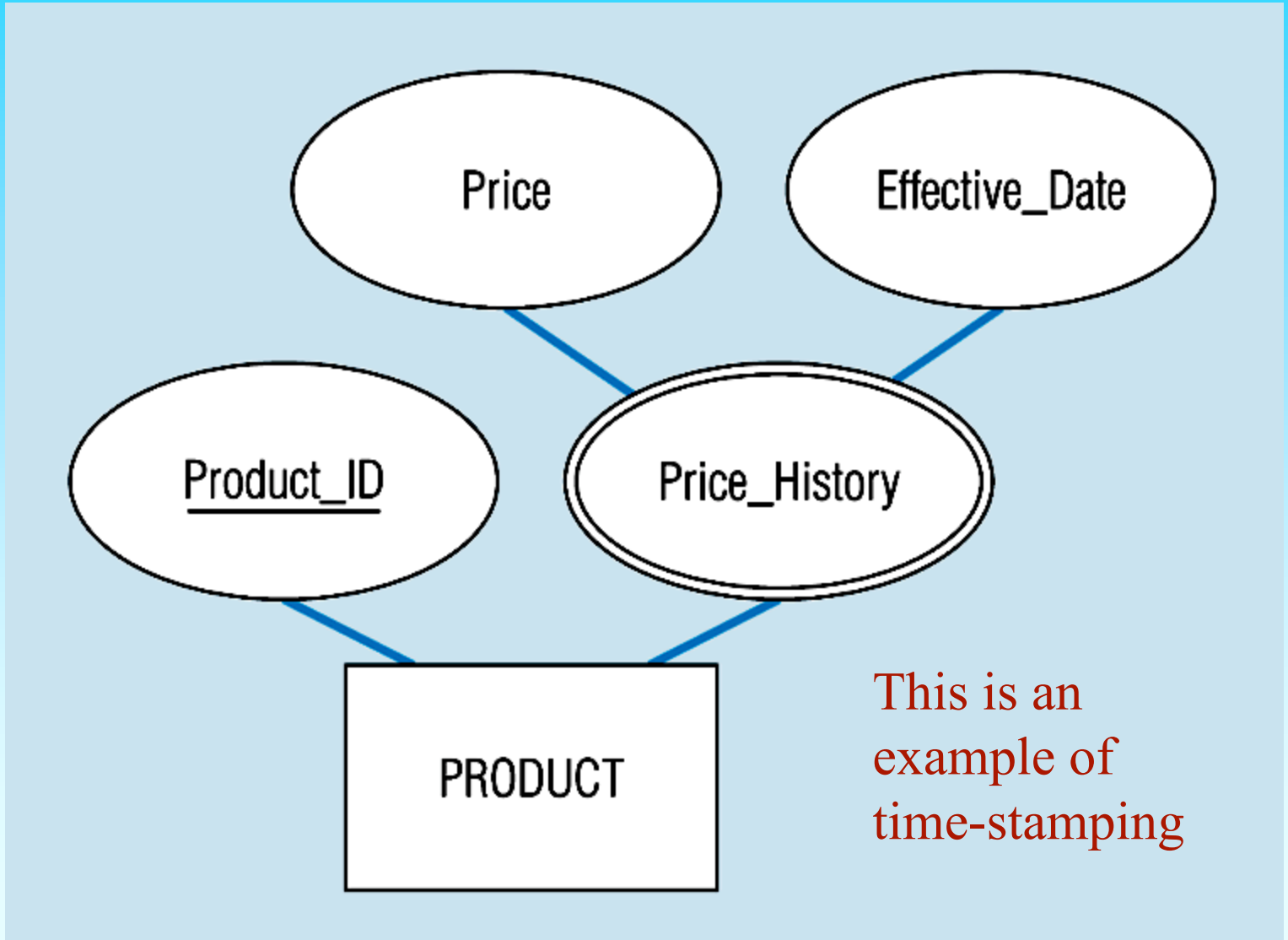


Figure 3-19 – An attribute that is both multivalued and composite



Multivalued attributes can be represented as relationships

Figure 3-15a Using relationships to link related attributes - Multivalued attribute versus relationships via bill-of-materials structure

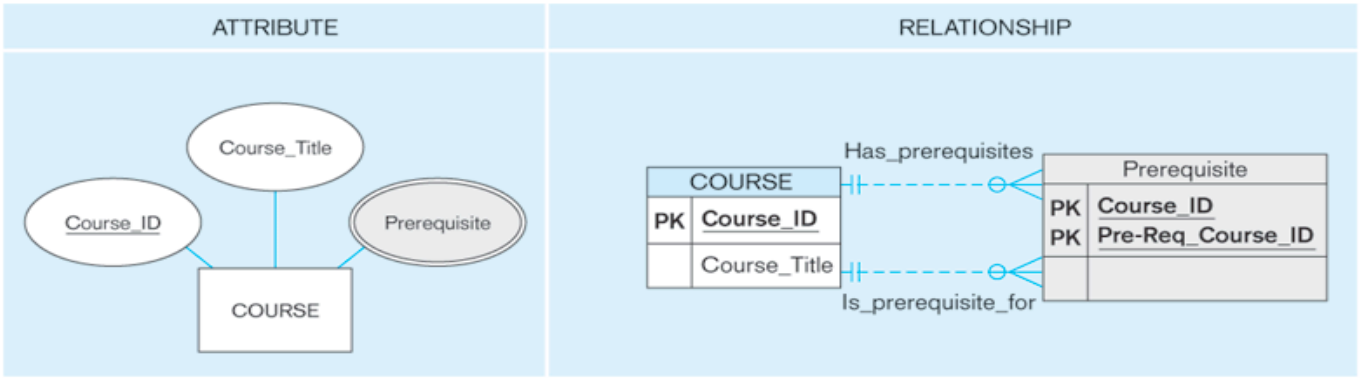
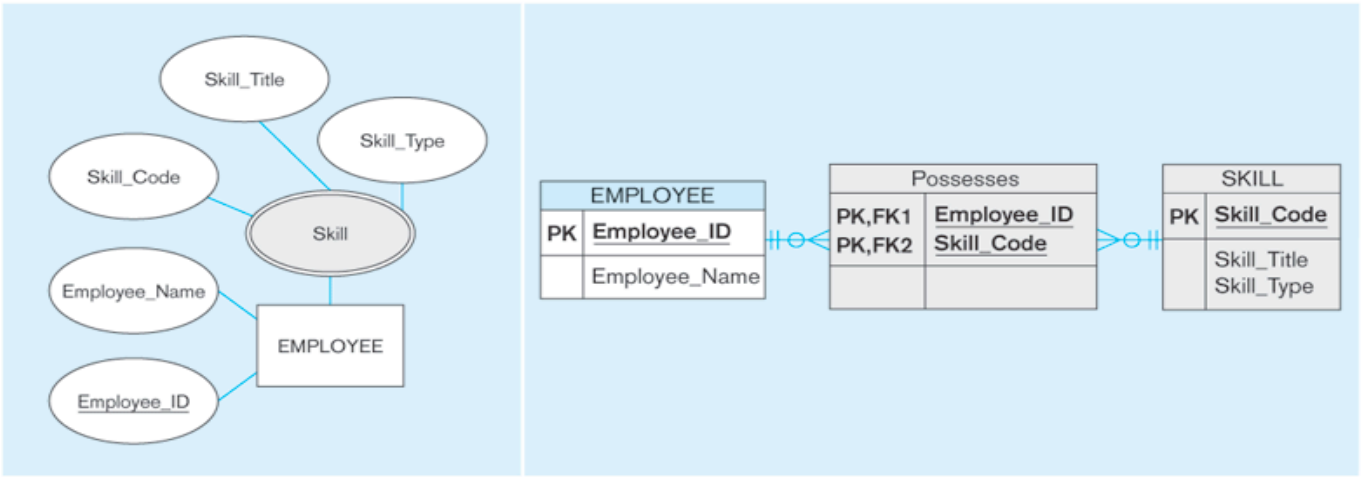


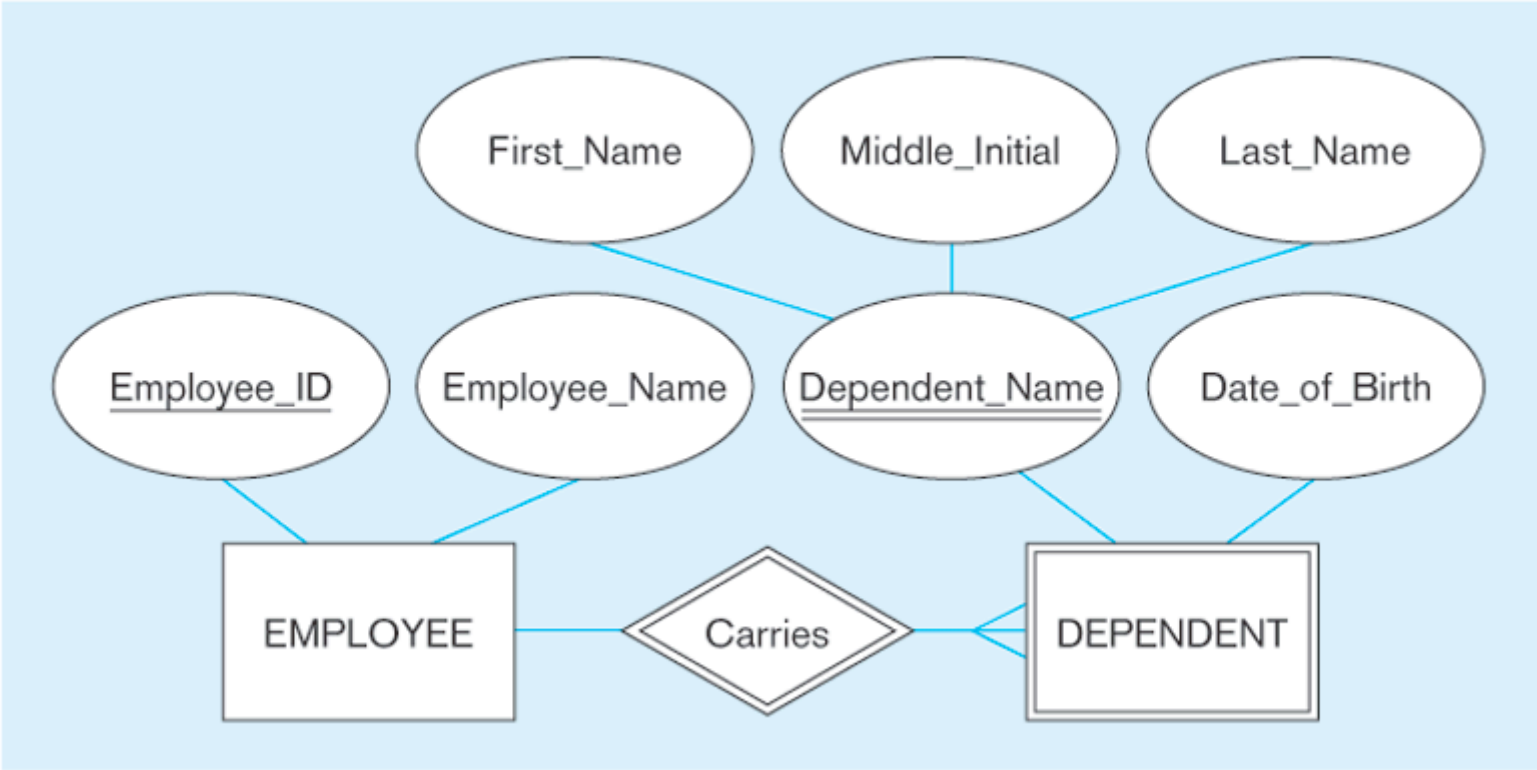
Figure 3-15b Using relationships to link related attributes - Composite, multivalued attribute versus relationship



Strong vs. Weak Entities, and Identifying Relationships

- Strong entities
  - exist independently of other types of entities
  - has its own unique identifier
  - represented with single-line rectangle
- Weak entity
  - dependent on a strong entity…cannot exist on its own
  - does not have a unique identifier
  - represented with double-line rectangle
- Identifying relationship
  - links strong entities to weak entities
  - represented with double line diamond

Figure 3-5a Example of a weak entity and its identifying relationship - E-R notation

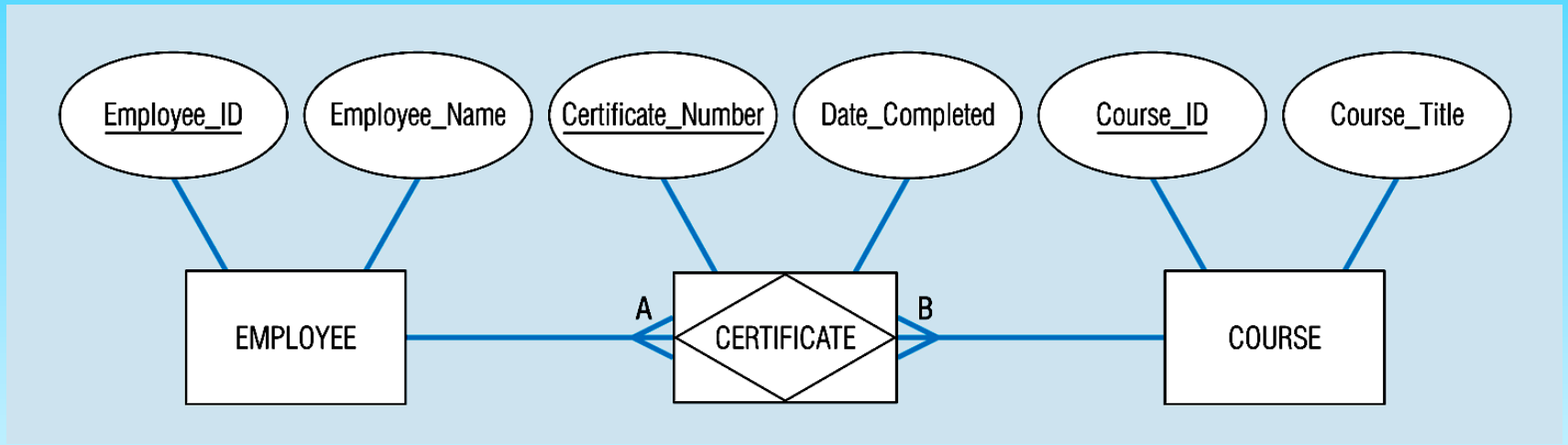


Strong entity Identifying relationship Weak entity

Associative Entities

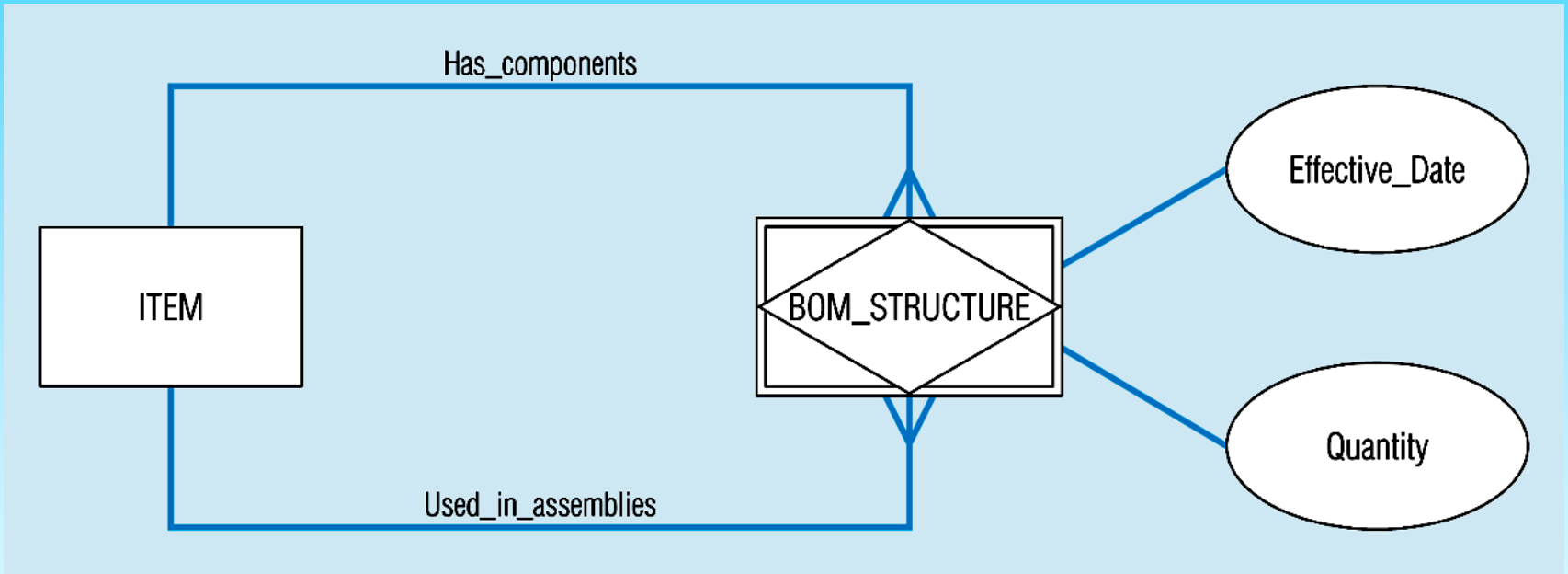
- It's an **entity** – it has attributes
- AND it's a **relationship** – it links entities together
- When should a *relationship with attributes* instead be an *associative entity*?
  - All relationships for the associative entity should be many
  - The associative entity could have meaning independent of the other entities
  - The associative entity preferably has a unique identifier, and should also have other attributes
  - The associative entity may participate in other relationships other than the entities of the associated relationship
  - Ternary relationships should be converted to associative entities

Figure 3-11b – An associative entity (CERTIFICATE)



Associative entity involves a rectangle with a diamond inside. Note that the many-to-many cardinality symbols face toward the associative entity and not toward the other entities

Figure 3-13c – An associative entity – bill of materials structure



This could just be a relationship with attributes…it's a judgment call

Figure 3-18 – Ternary relationship as an associative entity