HALMSTAD UNIVERSITY

DS4001 Databases (7.5 credits)

Lecture 6 – Entity-Relationship Diagrams

Yuantao Fan

yuantao.fan@hh.se

Halmstad University

Enhanced Entity-Relationship (EER) Model

- Why EER?
  - Class/subclass relationships (ISA) and type inheritance
  - Specialization and generalization
- Example p109
  - Secretary, Technician, Engineer, Manager are employees

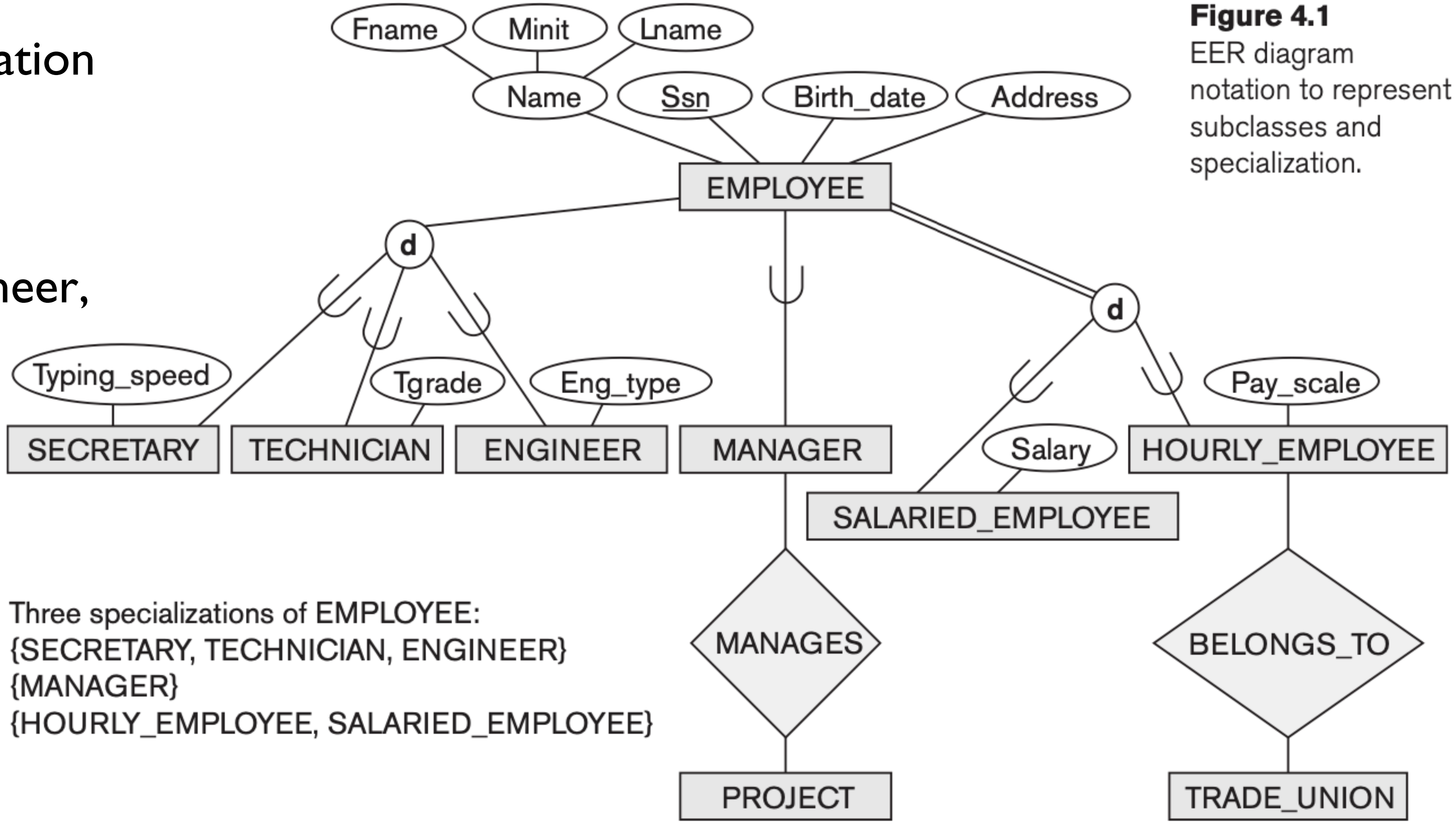
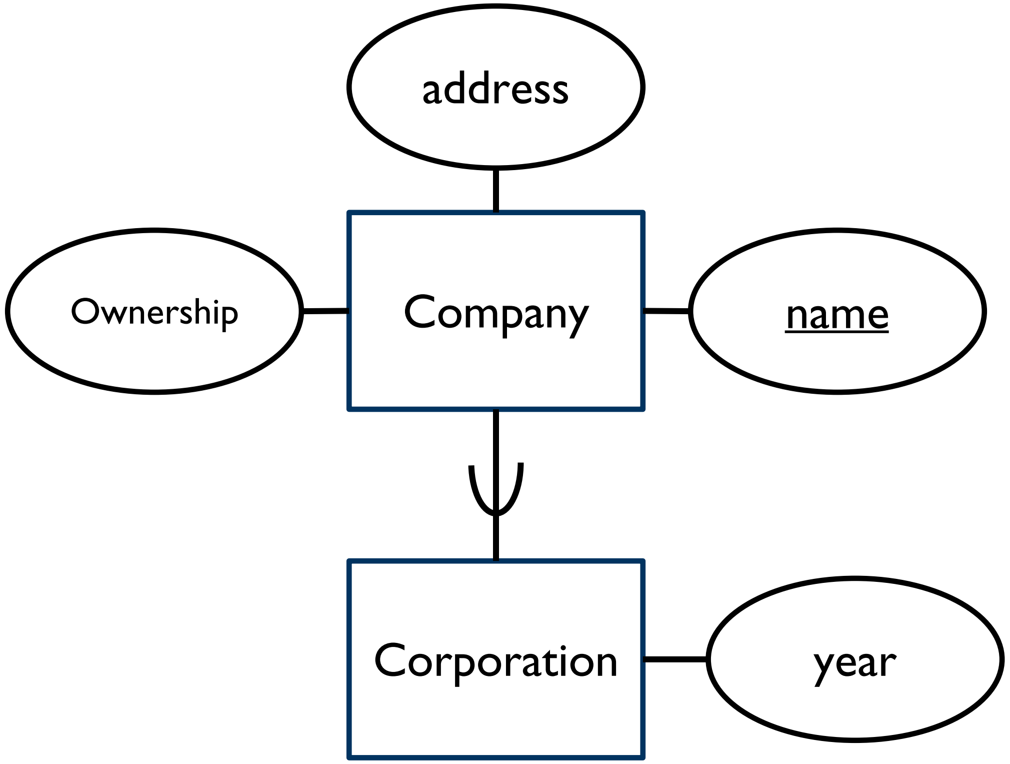


Figure 4.1
EER diagram notation to represent subclasses and specialization.

Inheritance in EER

- ISA relationships, ISA stands for "is a"
- Example
  - Corporations are a special kind of companies, they have a year in addition to all properties of other companies
- Corporation is a subentity
- Campany is its superentity
- Note that corporation do not has key attributes
- Subentities can never have key attributes of their own

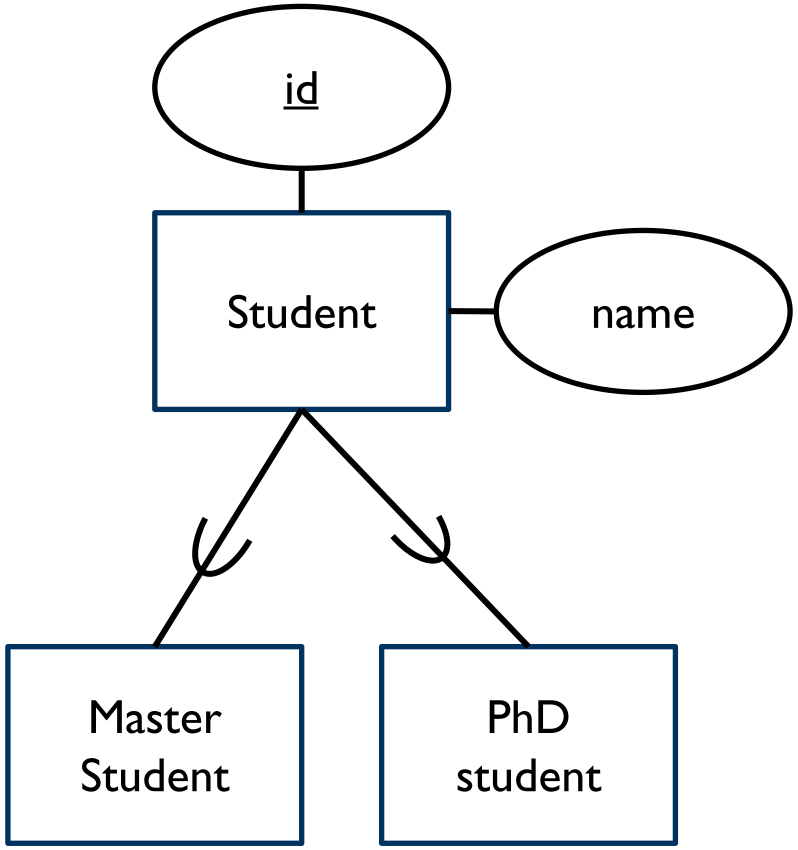


Companies(<u>name</u>, address)
Corporations(<u>name</u>, year)
name>companies.name

Multiple subentities

- In EER diagram, a instance can be a member of several subentities
  - If it is also a member of the superentity
- Example – four cases
  - X is only a student
  - X is a student and PhD student
  - X is a student and Master student
  - X is all three types of student

Students(id, name)
PhDStudents(id)
id -> Students.id
MasterStudents(id)
id ->Student.id



Specification

- Specialization is the process of defining a set of subclasses of an entity type
  - this entity type is called the superclass of the specialization.

- Examples
  - A set of subclasses {SECRETARY, ENGINEER, TECHNICIAN}
  - Another subclasses {SALARIED_EMPLOYEE, HOURLY_EMPLOYEE}

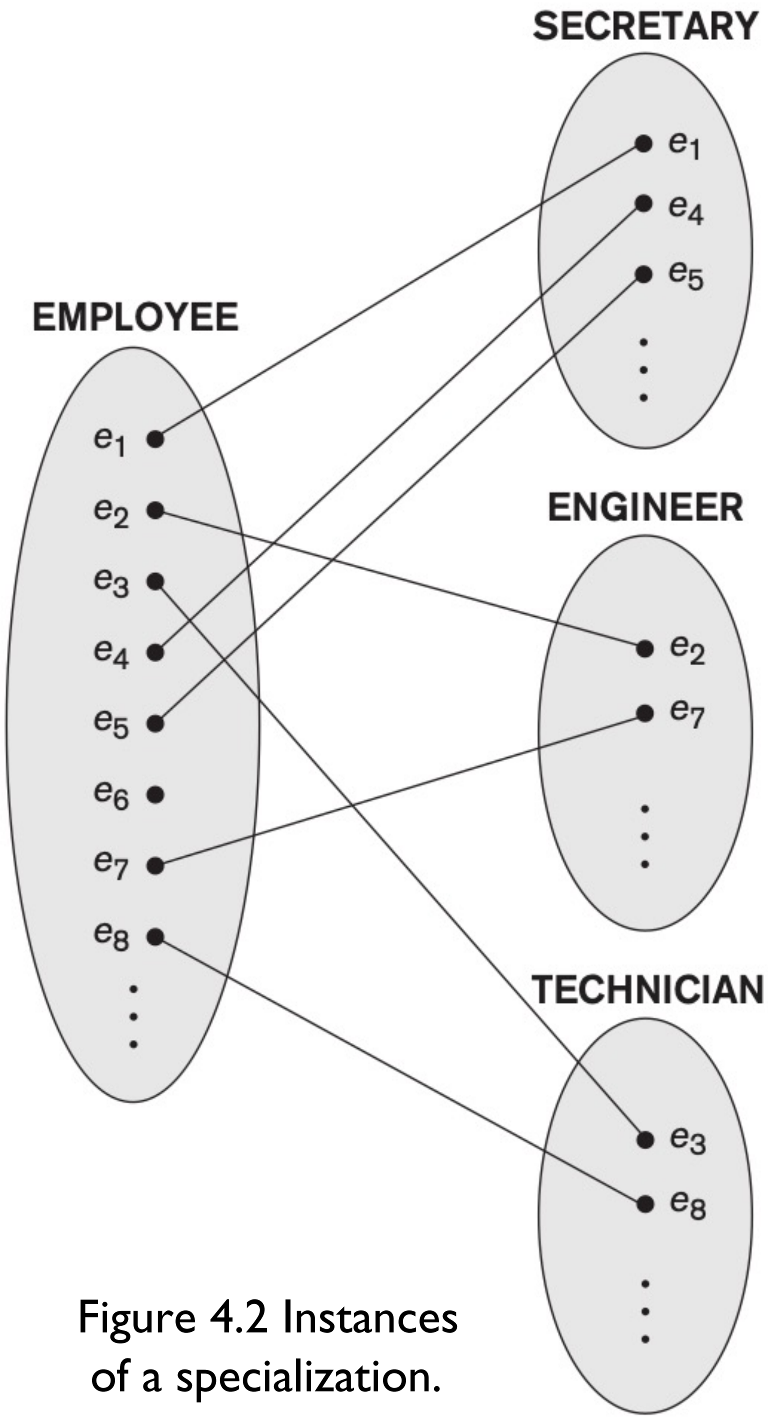
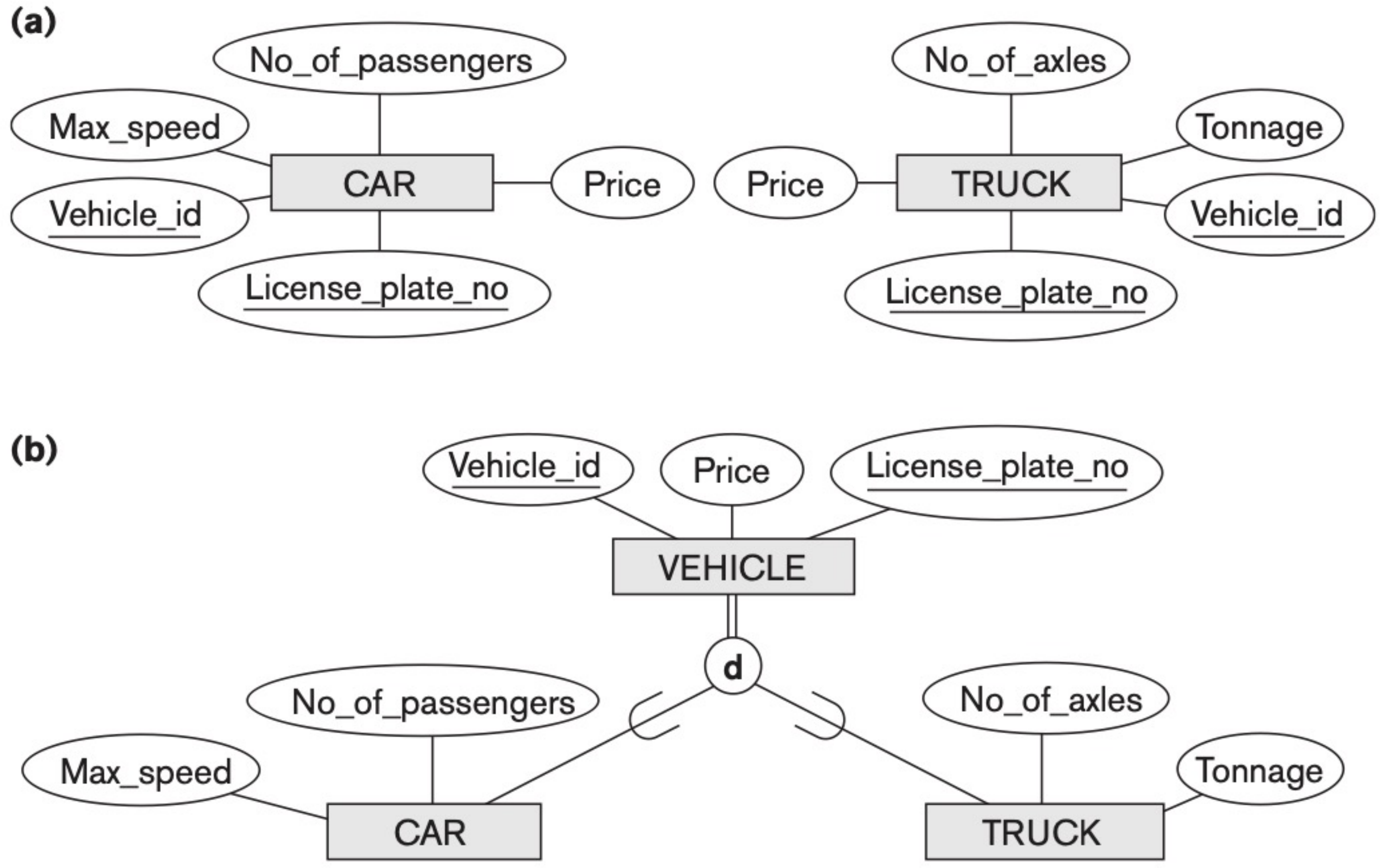


Figure 4.2 Instances of a specialization.

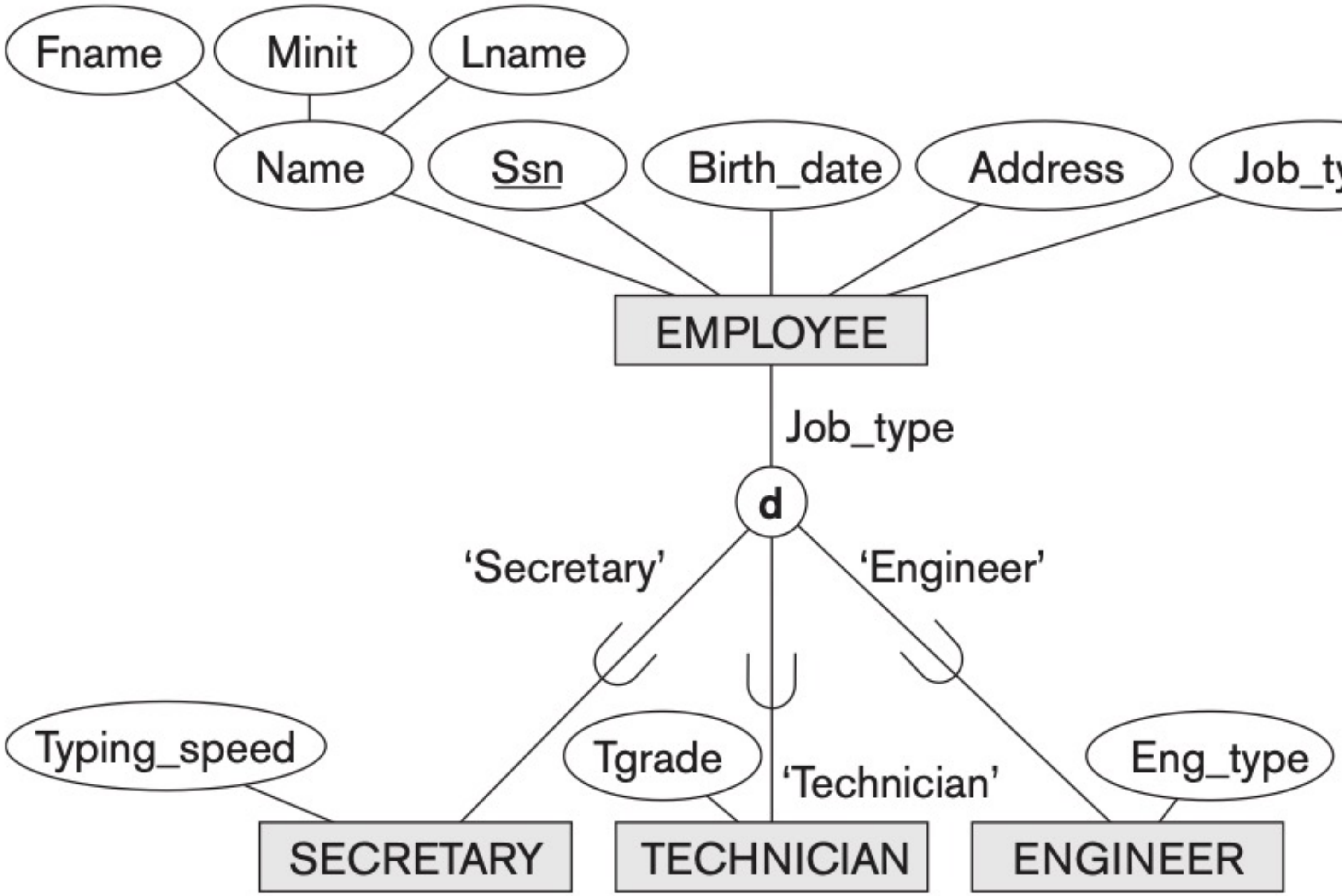
Generalization

- Generalization is a reverse process of abstraction in which
  - we suppress the differences among several entity types
  - identify their common features
  - and generalize them into a single superclass of which the original entity types are special subclasses

Figure 4.3
Generalization. (a) Two entity types, CAR and TRUCK.
(b) Generalizing CAR and TRUCK into the superclass VEHICLE.

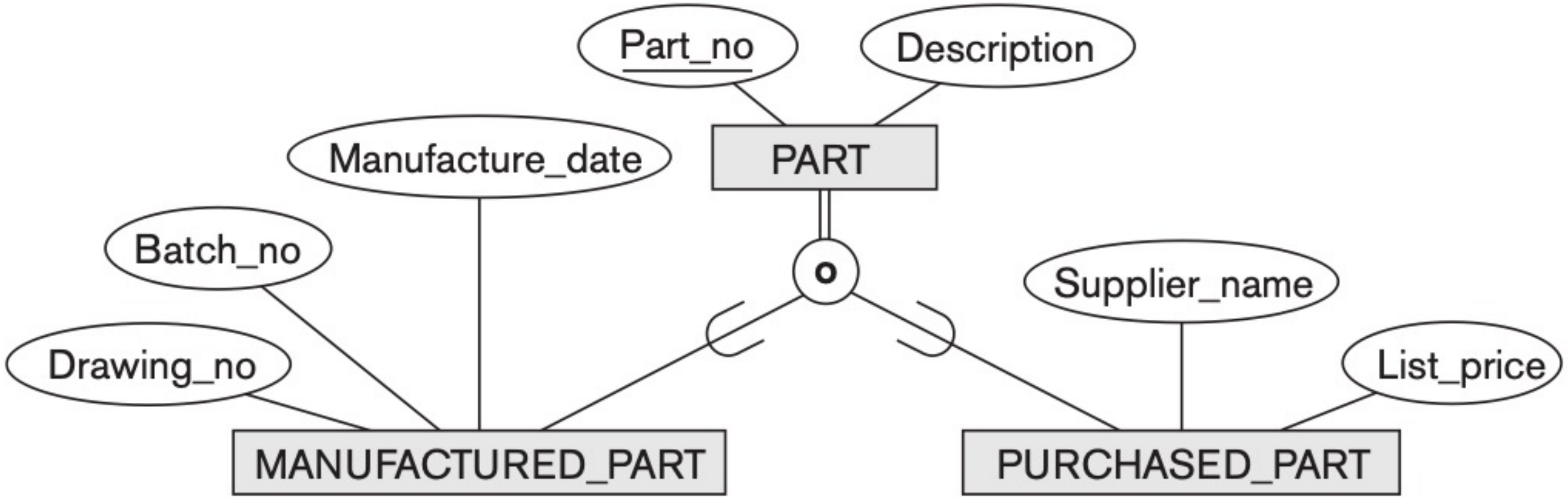


EER diagram notation for an attribute-defined specialization



P114, Figure 4.4

EER diagram notation for an overlapping (nondisjoint) specialization



P115, Fig 4.5

- If the subclasses are not constrained to be disjoint, their sets of entities may be overlapping;
  - that is, the same (real-world) entity may be a member of more than one subclass of the specialization.
- Displayed by placing “o” in the circle

Example

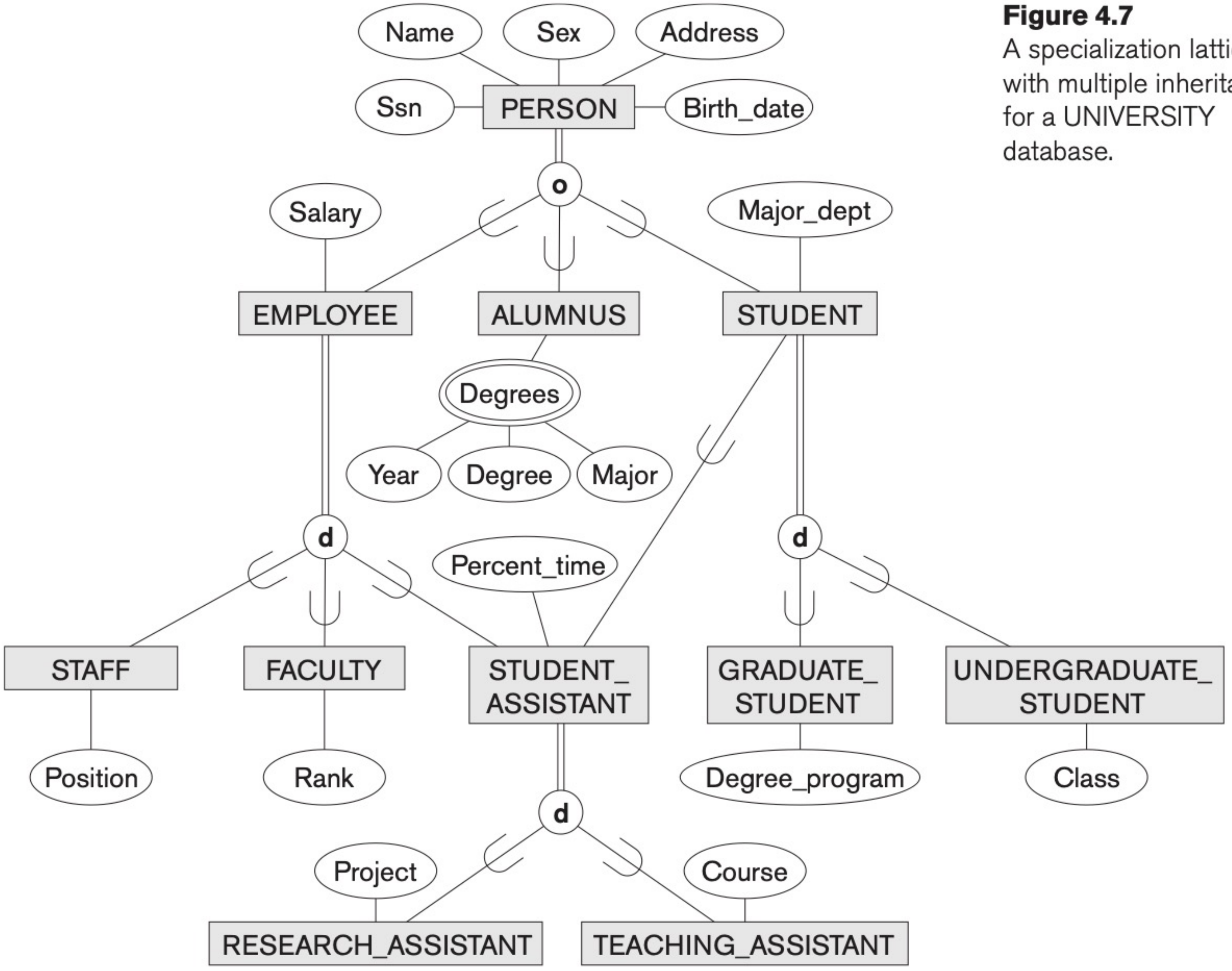
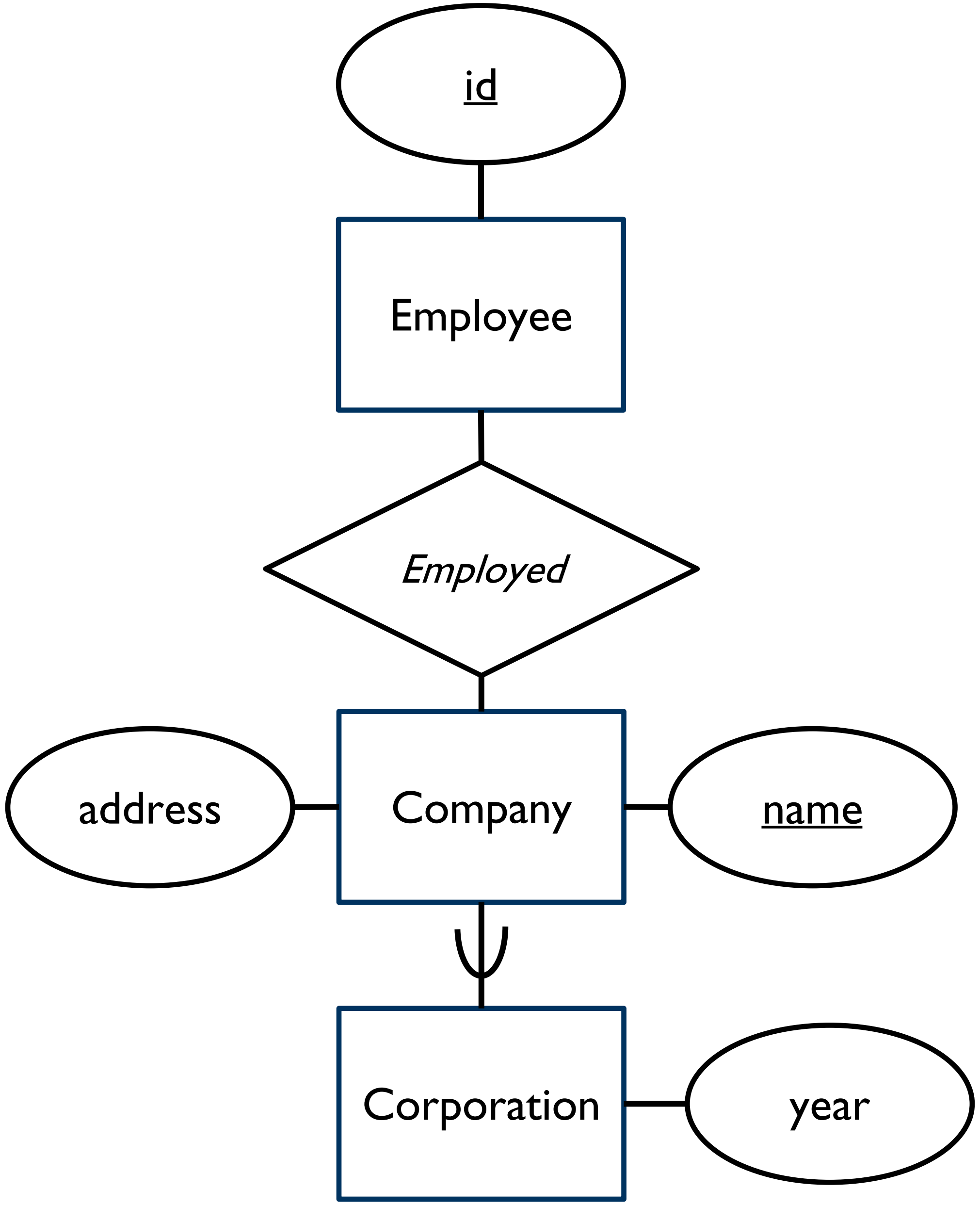


Figure 4.7
A specialization lattice with multiple inheritance for a UNIVERSITY database.

Inheritance in EER

- Employees are employed by companies
- Can employees employed by corporation?

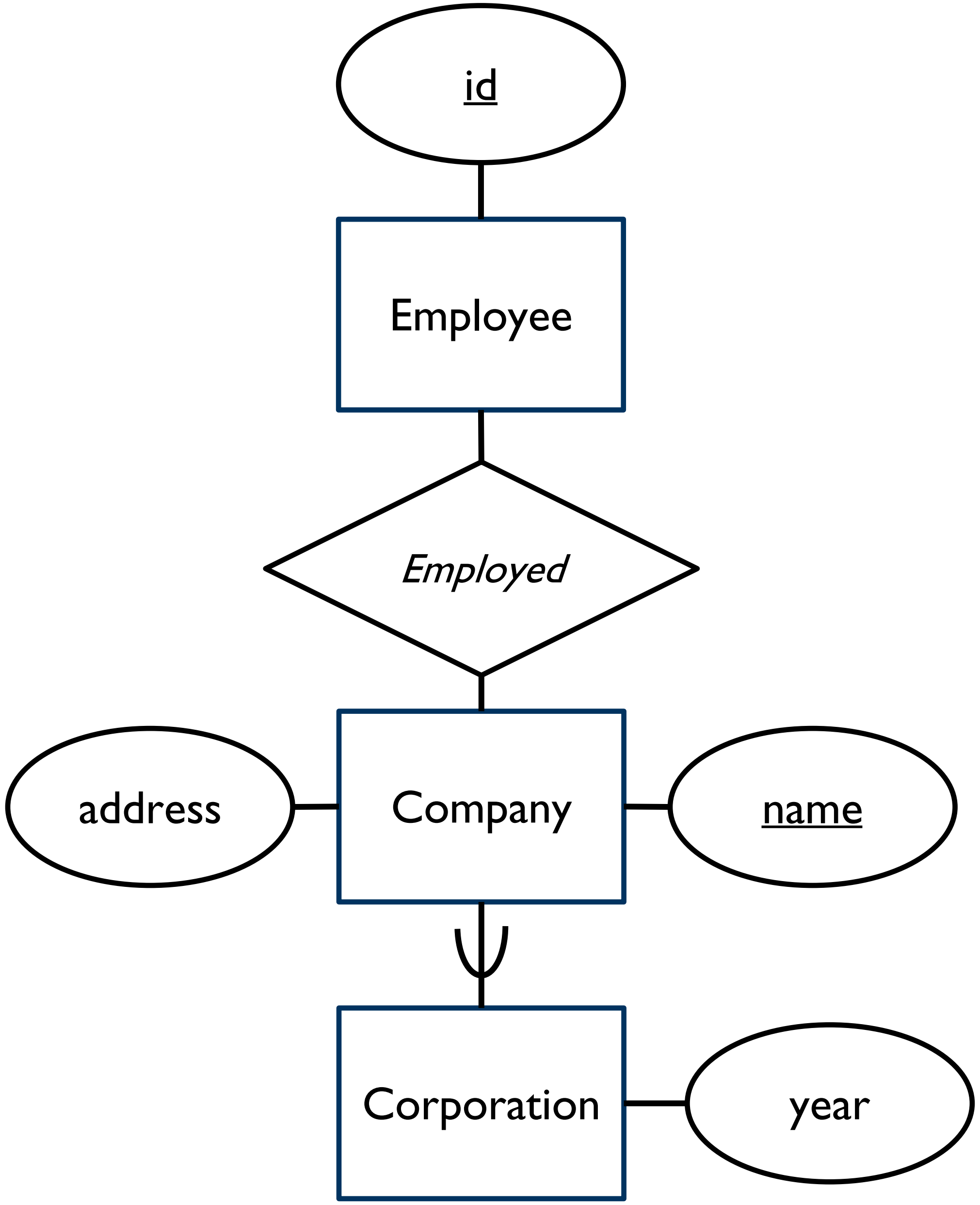


Inheritance in EER

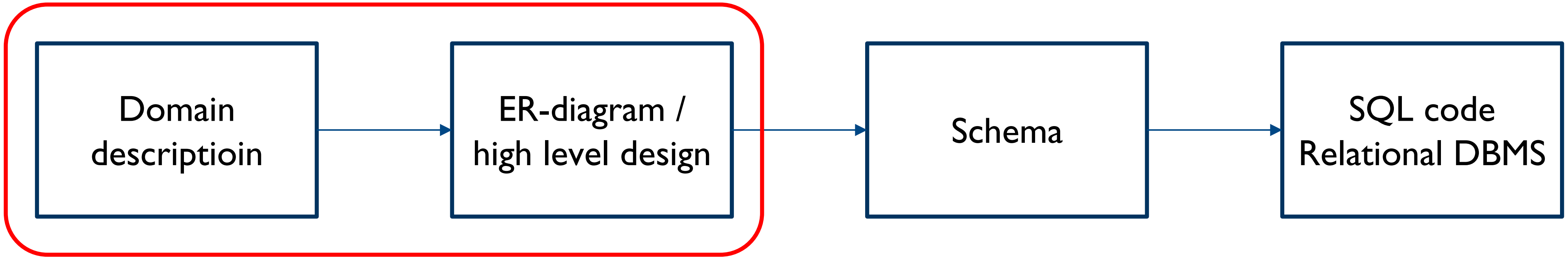
- Employees are employed by companies
- Can employees employed by corporation?

Companies(<u>name</u>, address)
Corporations(<u>name</u>, year)
    name>companies.name
Employees(<u>id</u>)
Employed(<u>employee</u>, <u>company</u>)
    employee -> Employees.id
    company -> Companies.name

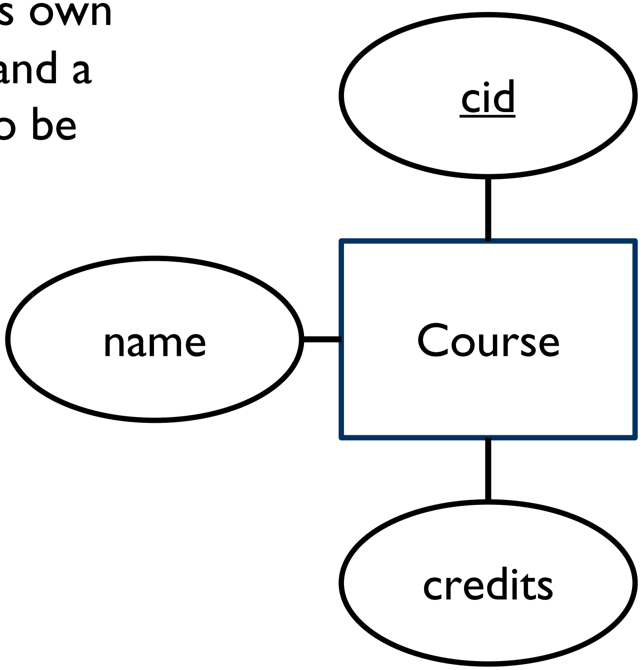
- Identifying ISA relationships
  - "A is a B", e.g. PhD student is a student; car is a vehicle



From ER model to relational schema



"Each Course has its own name, a unique ID, and a number of credits to be acquired."



- Start with identiying entity first, then relationships
  - Start with identifying entities that does not depending on others (e.g. keys of other entities)
    - First Superentities, then subentities
  - Start with entities on the exactly-one side for entities of many-to-exactly-one relationship
  - Then many-to-many relationship

A domain example

- Assume that your clients ask you to build a database for their companies in a enterprise group

"Please create a databse of companies of similar organisational structure. The companies have employees, and employees are divided into different deparment, within its company. Each deparment in the same company should have a unique name. Some employees have supervisor. Some employees are assigned as managers with a special title. Managers have access to company cars. Please also make an inventory of company cars and who can have access to it, and for which time period they have the access. "

Database Nomalization

- (Re-)structuring a relational database
  - to reduce data redundancy and improve data integrity
  - in accordance with a series of normal forms
  - proposed by Edgar F. Codd, relational model 1970

- Normalization entails organizing the columns (attributes) and tables (relations) of a database to ensure
  - dependencies are properly enforced by database integrity constraints

- Normal forms
  - 1NF – (1970)
  - 2NF & 3NF – (1971)
  - BCNF – Boyce and Codd Normal Form (1975)

- If a relational database is often described as normalized if it meets the 3rd normal form
  - Free from Insert, update, and deletion anomalies

Anomalies

- Insertion anomaly
  - Facts that do not obtain all columns info. Can not be inserted.
- Update anomaly
  - Same information can be stored on multiple roles, then update may cause logical inconsistency
- Deletion anomaly
  - Deletion of data representing certain facts may cause deletion of data representing the completely different facts

Faculty and Their Courses

Faculty ID	Faculty Name	Faculty Hire Date	Course Code
389	Dr. Giddens	10-Feb-1985	ENG-206
407	Dr. Saperstein	19-Apr-1999	CMP-101
407	Dr. Saperstein	19-Apr-1999	CMP-201

424	Dr. Newsome	29-Mar-2007	?

Employees' Skills

Employee ID	Employee Address	Skill
426	87 Sycamore Grove	Typing
426	87 Sycamore Grove	Shorthand
519	94 Chestnut Street	Public Speaking
519	96 Walnut Avenue	Carpentry

Faculty and Their Courses

Faculty ID	Faculty Name	Faculty Hire Date	Course Code
389	Dr. Giddens	10-Feb-1985	ENG-206
407	Dr. Saperstein	19-Apr-1999	CMP-101
407	Dr. Saperstein	19-Apr-1999	CMP-201

DELETE

Source: https://en.wikipedia.org/wiki/Database_normalization

Normal Forms

- 1NF
  - Each column has a single value
- 2NF
  - 1NF + has valid (single col. works) primary key
- 3NF
  - 2NF + no Functional Dependencies between attributes not in keys
- BCNF
  - 3NF + attributes depend only on keys
- 4NF:
  - 3NF + No violating Multiple Valued Dependencies

Constraint (informal description in parentheses)	UNF (1970)	1NF (1970)	2NF (1971)	3NF (1971)	EKNF (1982)	BCNF (1974)	4NF (1977)	ETNF (2012)	5NF (1979)	DKNF (1981)	6NF (2003)
Unique rows (no duplicate records)[4]	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Scalar columns (columns cannot contain relations or composite values)[5]	✗	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Every non-prime attribute has a full functional dependency on a candidate key (attributes depend on the *complete* primary key)[5]	✗	✗	✓	✓	✓	✓	✓	✓	✓	✓	✓
Every non-trivial functional dependency either begins with a superkey or ends with a prime attribute (attributes depend *only* on the primary key)[5]	✗	✗	✗	✓	✓	✓	✓	✓	✓	✓	✓
Every non-trivial functional dependency either begins with a superkey or ends with an elementary prime attribute (a stricter form of 3NF)	✗	✗	✗	✗	✓	✓	✓	✓	✓	✓	—
Every non-trivial functional dependency begins with a superkey (a stricter form of 3NF)	✗	✗	✗	✗	✗	✓	✓	✓	✓	✓	—
Every non-trivial multivalued dependency begins with a superkey	✗	✗	✗	✗	✗	✗	✓	✓	✓	✓	—
Every join dependency has a superkey component[8]	✗	✗	✗	✗	✗	✗	✗	✓	✓	✓	—
Every join dependency has only superkey components	✗	✗	✗	✗	✗	✗	✗	✗	✓	✓	—
Every constraint is a consequence of domain constraints and key constraints	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗
Every join dependency is trivial	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓

Source: https://en.wikipedia.org/wiki/Database_normalization

UNF

Raw initial data

Title	Author	Author Nationality	Format	Price	Subject	Pages	Thickness	Publisher	Publisher Country	Publication Type	Genre ID	Genre Name
Beginning MySQL Database Design and Optimization	Chad Russell	American	Hardcover	49.99	MySQL Database Design	520	Thick	Apress	USA	E-book	1	Tutorial

Adding a primary key, a prerequisite to conform to the relational model

ISBN	Title	Author	Author Nationality	Format	Price	Subject	Pages	Thickness	Publisher	Publisher Country	Publication Type	Genre ID	Genre Name
1590593324	Beginning MySQL Database Design and Optimization	Chad Russell	American	Hardcover	49.99	MySQL Database Design	520	Thick	Apress	USA	E-book	1	Tutorial

1NF

- To satisfy First normal form, each column of a table must have a single value.

ISBN	Title	Author	Author Nationality	Format	Price	Subject	Pages	Thickness	Publisher	Publisher Country	Publication Type	Genre ID	Genre Name
1590593324	Beginning MySQL Database Design and Optimization	Chad Russell	American	Hardcover	49.99	MySQL Database Design	520	Thick	Apress	USA	E-book	1	Tutorial

Book

ISBN	<u>Title</u>	<u>Format</u>	Author	Author Nationality	Price	Pages	Thickness	Publisher	Publisher country	Genre ID	Genre Name
1590593324	Beginning MySQL Database Design and Optimization	Hardcover	Chad Russell	American	49.99	520	Thick	Apress	USA	1	Tutorial

Subject

<u>ISBN</u>	Subject name
1590593324	MySQL
1590593324	Database
1590593324	Design

Satisfying 2NF

Book

<u>Title</u>	<u>Format</u>	Author	Author Nationality	Price	Pages	Thickness	Genre ID	Genre Name	*Publisher ID*
Beginning MySQL Database Design and Optimization	Hardcover	Chad Russell	American	49.99	520	Thick	1	Tutorial	*1*
Beginning MySQL Database Design and Optimization	E-book	Chad Russell	American	22.34	520	Thick	1	Tutorial	*1*
The Relational Model for Database Management: Version 2	E-book	E.F.Codd	British	13.88	538	Thick	2	Popular science	*2*
The Relational Model for Database Management: Version 2	Paperback	E.F.Codd	British	39.99	538	Thick	2	Popular science	*2*

Book

<u>Title</u>	Author	Author Nationality	Pages	Thickness	Genre ID	Genre Name	*Publisher ID*
Beginning MySQL Database Design and Optimization	Chad Russell	American	520	Thick	1	Tutorial	*1*
The Relational Model for Database Management: Version 2	E.F.Codd	British	538	Thick	2	Popular science	*2*

Format - Price

<u>Title</u>	<u>Format</u>	Price
Beginning MySQL Database Design and Optimization	Hardcover	49.99
Beginning MySQL Database Design and Optimization	E-book	22.34
The Relational Model for Database Management: Version 2	E-book	13.88
The Relational Model for Database Management: Version 2	Paperback	39.99

Publisher

<u>Publisher ID</u>	Name	Country
1	Apress	USA
2	Addison-Wesley	USA

Satisfying 3NF

Book

Title	Author	Author Nationality	Pages	Thickness	Genre ID	Genre Name	*Publisher ID*
Beginning MySQL Database Design and Optimization	Chad Russell	American	520	Thick	1	Tutorial	*1*
The Relational Model for Database Management: Version 2	E.F.Codd	British	538	Thick	2	Popular science	*2*

Format - Price

Title	Format	Price
Beginning MySQL Database Design and Optimization	Hardcover	49.99
Beginning MySQL Database Design and Optimization	E-book	22.34
The Relational Model for Database Management: Version 2	E-book	13.88
The Relational Model for Database Management: Version 2	Paperback	39.99

Publisher

Publisher ID	Name	Country
1	Apress	USA
2	Addison-Wesley	USA

Book

Title	Author	Pages	Thickness	Genre ID	*Publisher ID*
Beginning MySQL Database Design and Optimization	Chad Russell	520	Thick	1	*1*
The Relational Model for Database Management: Version 2	E.F.Codd	538	Thick	2	*2*

Author

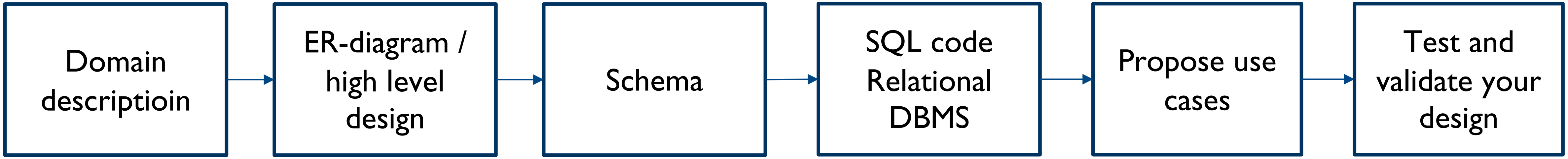
Author	Author Nationality
Chad Russell	American
E.F.Codd	British

Genre

Genre ID	Genre Name
1	Tutorial
2	Popular science

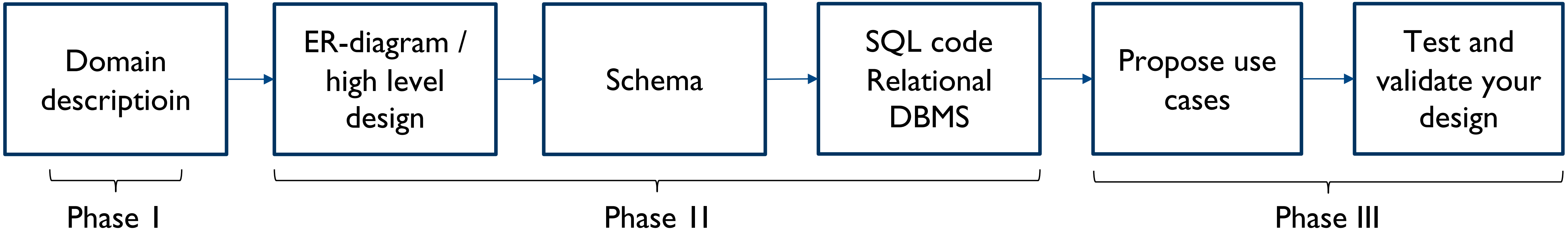
Project Introduction

- Design and implement a database for a specific domain of your choice
  - Investigate the domain chosen, gather relevant information
  - Written a domain description
  - Translate the description to ER-diagram, and schema
  - Implment the design with SQL code
- Propose use cases and demostrate how the implementation works
  - Propose use case for storing, and for querying data
  - Test and validate your design
- Document and present the design processes and the results



Project Timeline

- Design and implement a database for a specific domain of your choice
  - Investigate the domain chosen, gather relevant information
  - Written a domain description (hand-in - end of week 4, 2023.02.12)
  - Translate the description to ER-diagram, and schema
  - Implment the design with SQL code (preliminary code - week 5, 2023.02.19)
- Propose use cases and demostrate how the implementation works
  - Propose use case for storing, and for querying data
  - Test and validate your design (Use cases and result - week 6, 2023.02.26)
- Document and present the design processes and the results (book time – week 7)



Lab 2 Introduction

- Objective & learning outcome
  - Learn how to design databases via ER-diagram
    - Learn how to translate domain description to ER-diagram
    - Identify different components and how they are related in the diagram
    - Know how to translate ER-diagram to schema
  - Learn how to impelment databases given ER-diagrams

- Content
  - Design ER-diagram based on domain description
  - Produce schema given ER-diagram
  - Write SQL code to implement given ER-diagrams

Lab 2 Introduction

- Domain description and ER-diagram
- The Company-employee example
- The Café example
- The University room booking system example