#sdudk

ER & EER diagrams and mapping
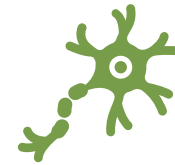
Status

#sdudk

Create Tables

Create Relationships

SQL Queries

Design databases

Data - Normalize to 4th normal form

Interviews, text, conversations?

Domain model vs. ER Diagram

→ ISO/IEC/IEEE 24765 : Systems and software engineering – defines the vocabulary as:
  → **domain model** "a product of domain analysis that provides a representation of the requirements of the domain."
  → **entity-relationship diagram** "a diagram that depicts a set of real-world entities and the logical relationships among them."
→ But a domain model can evolve into an ER diagram.

The process

- → DBMS independent works with entities and concepts.
- → DBMS specific works with SQL and Tables.
- → The mapping in the middle is the process of working from the ER diagrams to the table mapping.

SDU❧ Jakob Hviid   **4**

In relation to UP

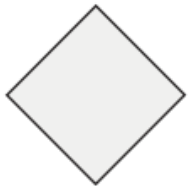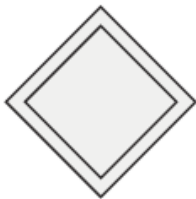| Phase	Artifact
Business Modelling	Domain Model (None from DM)
Requirements	(None from DM)
Analysis	ER, EER, UML (Entities only)
Design	UML (Tables)
Implementation	SQL Creation Script
Test	(None from DM)
Deployment	(None from DM)

sdu.dk

#sdudk

ER Components 1/3

Entity
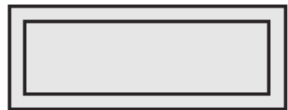
Weak Entity

Relationship

Indentifying Relationship

→ An entity type is strong if its existence does not depend on another entity type. Otherwise, the entity type is weak.
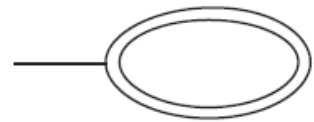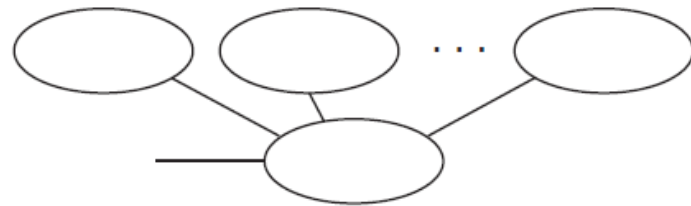
ER Components 2/3

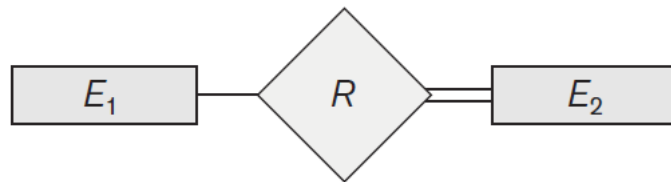Attribute

Key Attribute

Multivalued Attribute

→ ...

Composite Attribute

・・・

Derived Attribute

ER Components 3/3

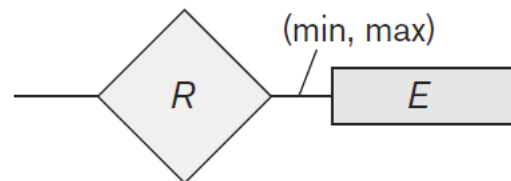E_1 — R — E_2 Total Participation of E_2 in R
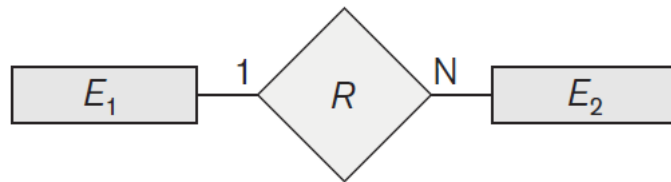
E_1 — 1 R N — E_2 Cardinality Ratio 1 : N for $E_1 : E_2$ in R

R — (min, max) — E   Structural Constraint (min, max) on Participation of E in R

→ Total participation can also be understood as E2 HAS to have one entry.

→ Example, a teacher MUST teach a class (or they are not a teacher).

Cardinalities

——————————————|—	One
——————————————<	Many
——————————————||—	One and only one
——————————————o|—	Zero or one
——————————————|<	One or many
——————————————o<	Zero or many

Crow's foot notation

→ 1 - * , one to many, 1:N

→ * - 1, many to one, N:1

→ * - *, many to many M:N

→ 1 - 1, one to one 1:1

→ 0..1 - 1, zero/one to one, 0..1:1

And there are many alternatives
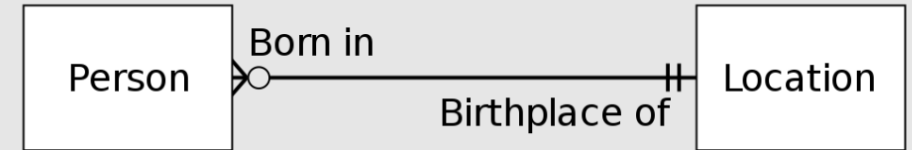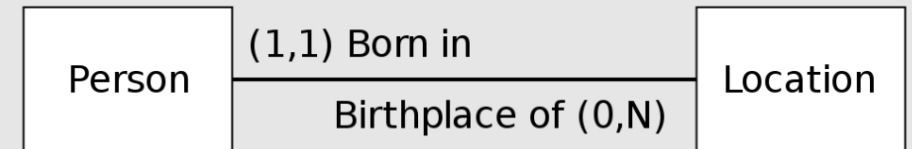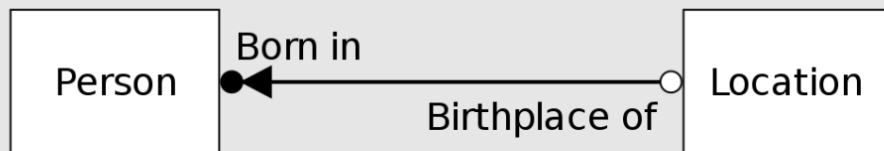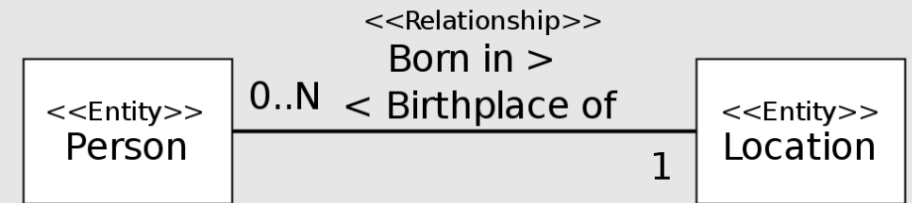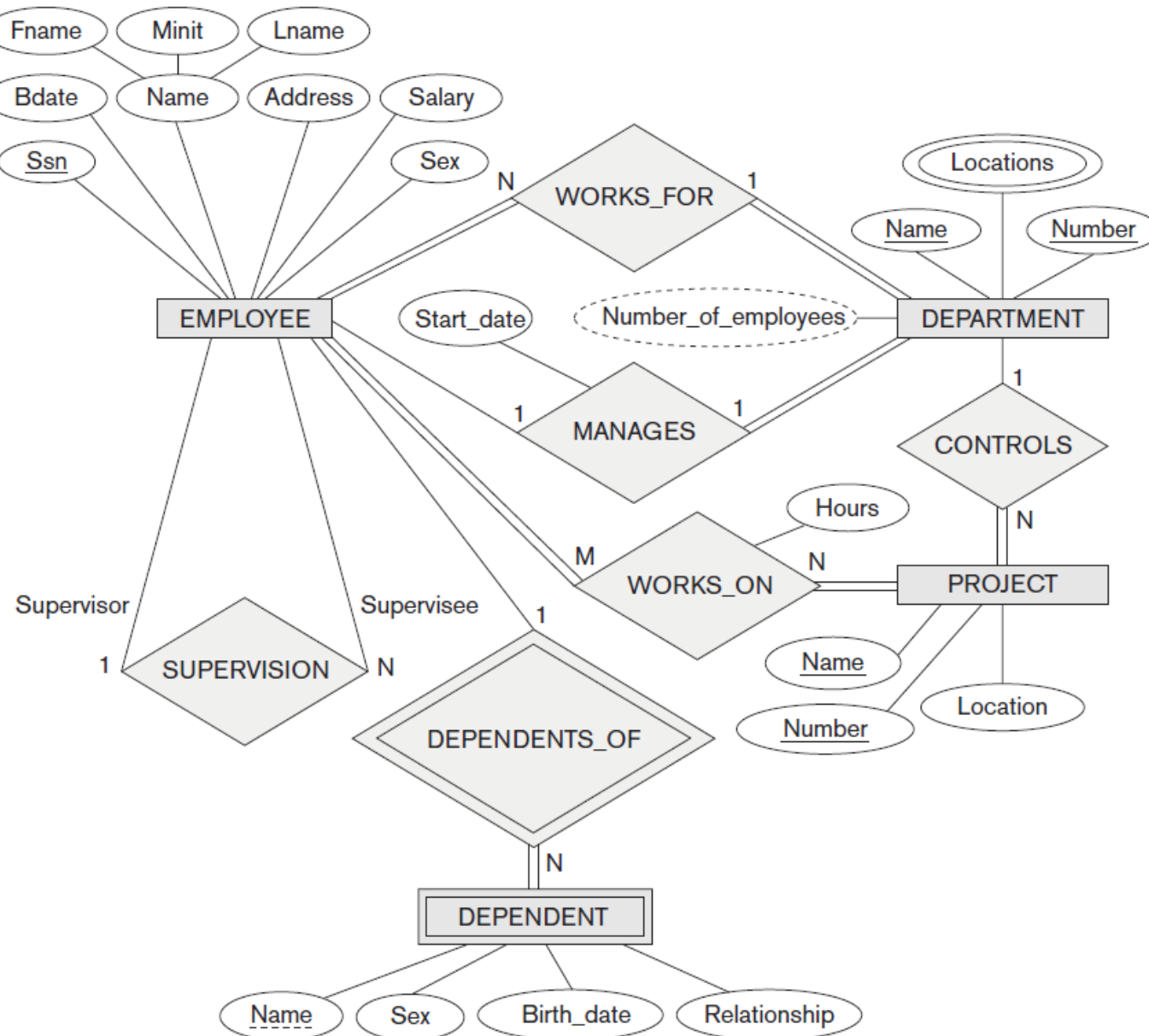
ER Diagram Example

→ Entity types (strong/weak)

→ Relationship types (Identifying or not)

→ Recursive relationship

→ Attribute types (composite, key, derived, multiple values, etc.)

→ Cardinalities

ER Diagram Example
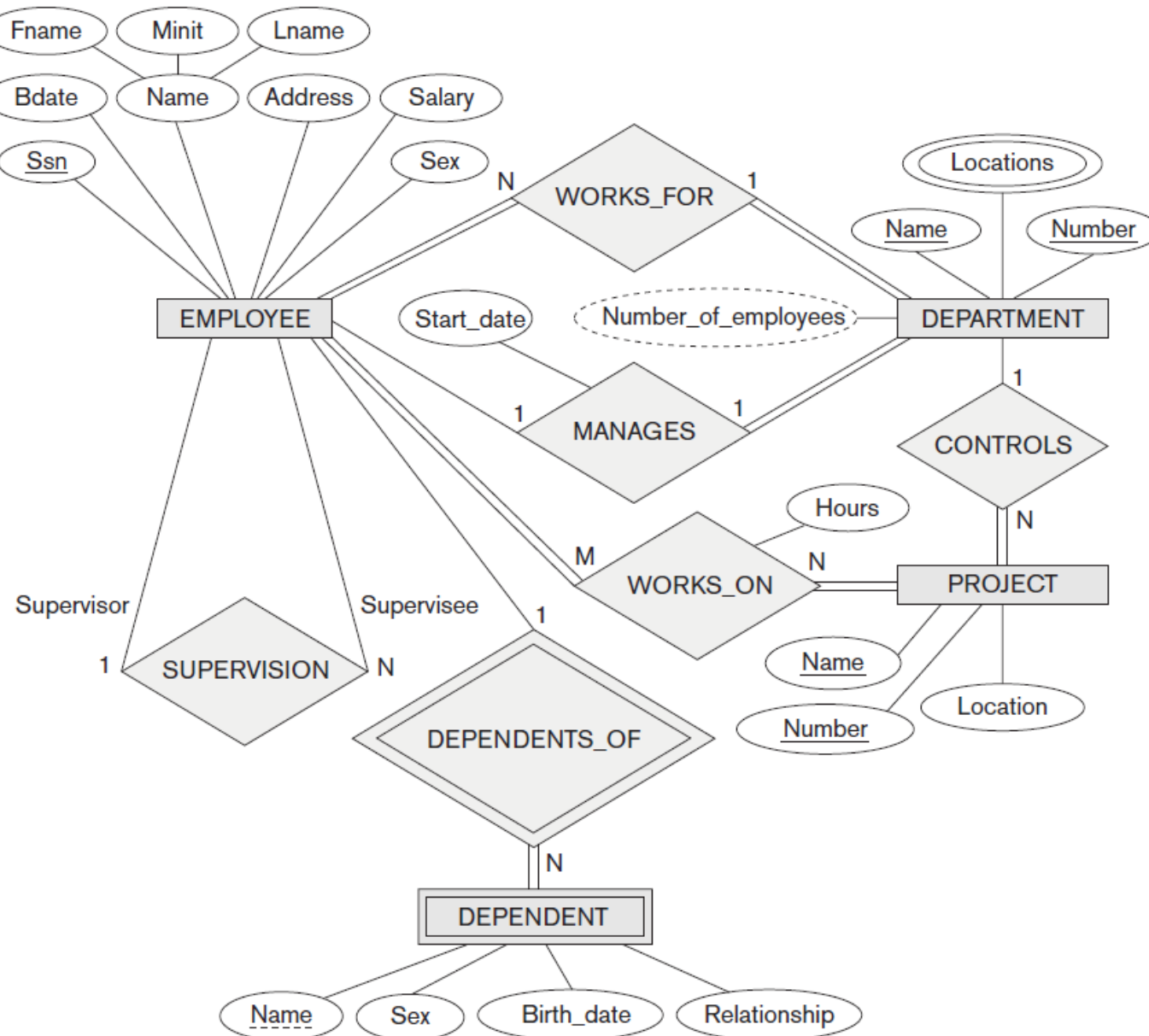
→ Entity types (strong/weak)

→ Relationship types (Identifying or not)

→ Recursive relationship

→ Attribute types (composite, key, derived, multiple values, etc.)

→ Cardinalities

Alternative structural constraints

→ Min, Max notation

sdu.dk

#sdudk

Figure 4.1
EER diagram notation to represent subclasses and specialization.

Three specializations of EMPLOYEE:
{SECRETARY, TECHNICIAN, ENGINEER}
{MANAGER}
{HOURLY_EMPLOYEE, SALARIED_EMPLOYEE}
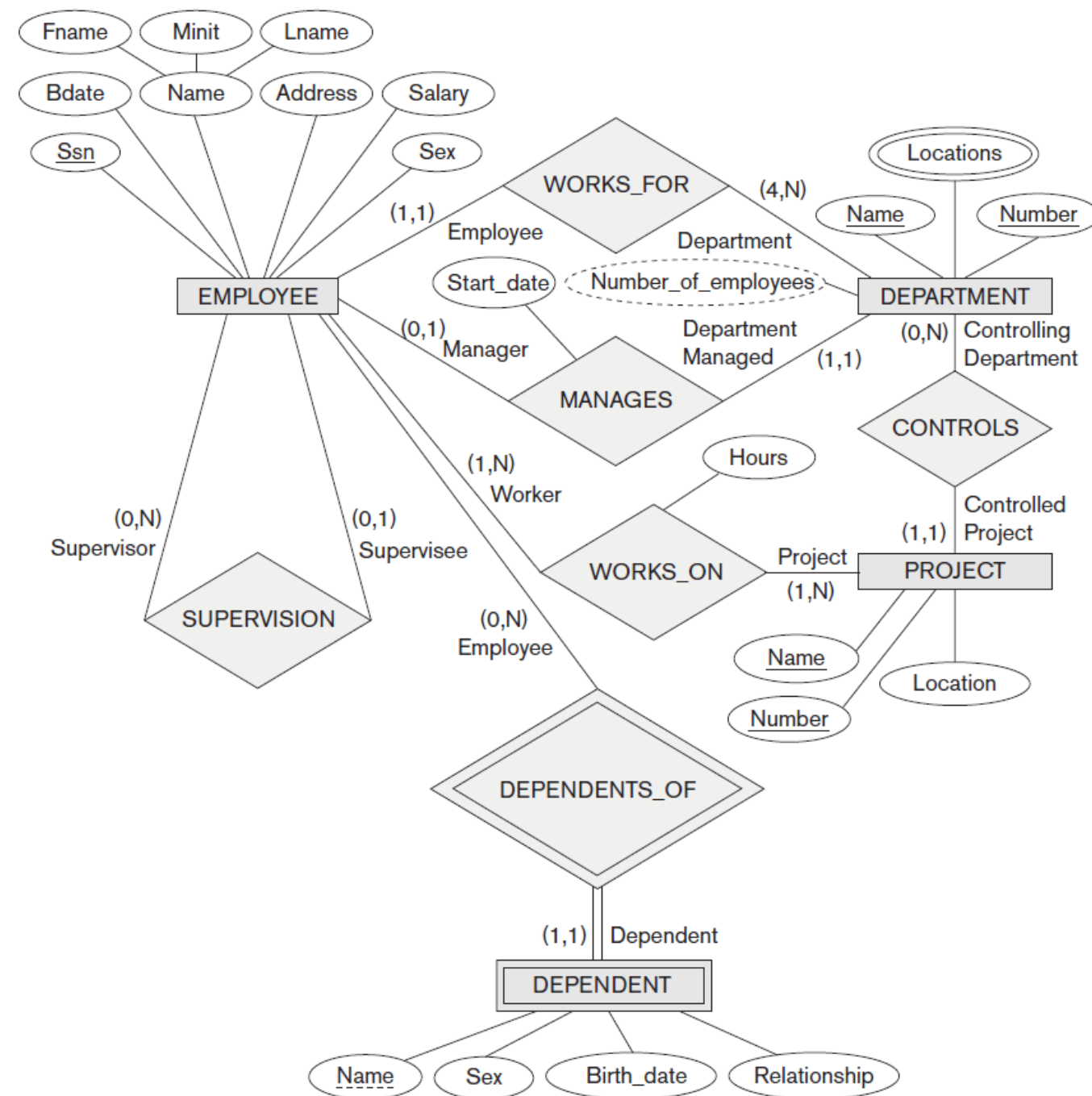
EER: Enhanced entity-relationship diagram
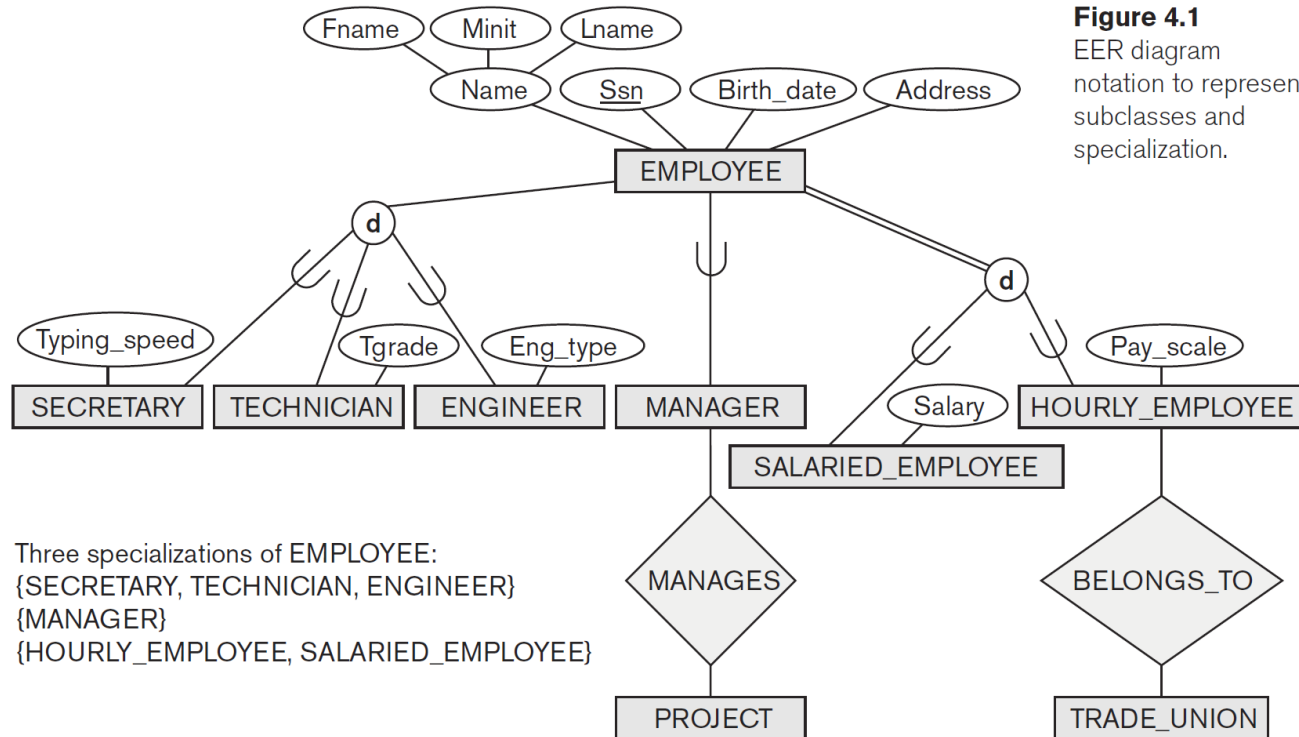
→ Expands with the following components:

→ Attribute or relationship inheritances
→ Category or union types
→ Specialization and generalization
→ Subclasses and superclasses

SDU ✿ **Jakob Hviid**   **14**

Inheritance

Figure 4.7
A specialization lattice with multiple inheritance for a UNIVERSITY database.
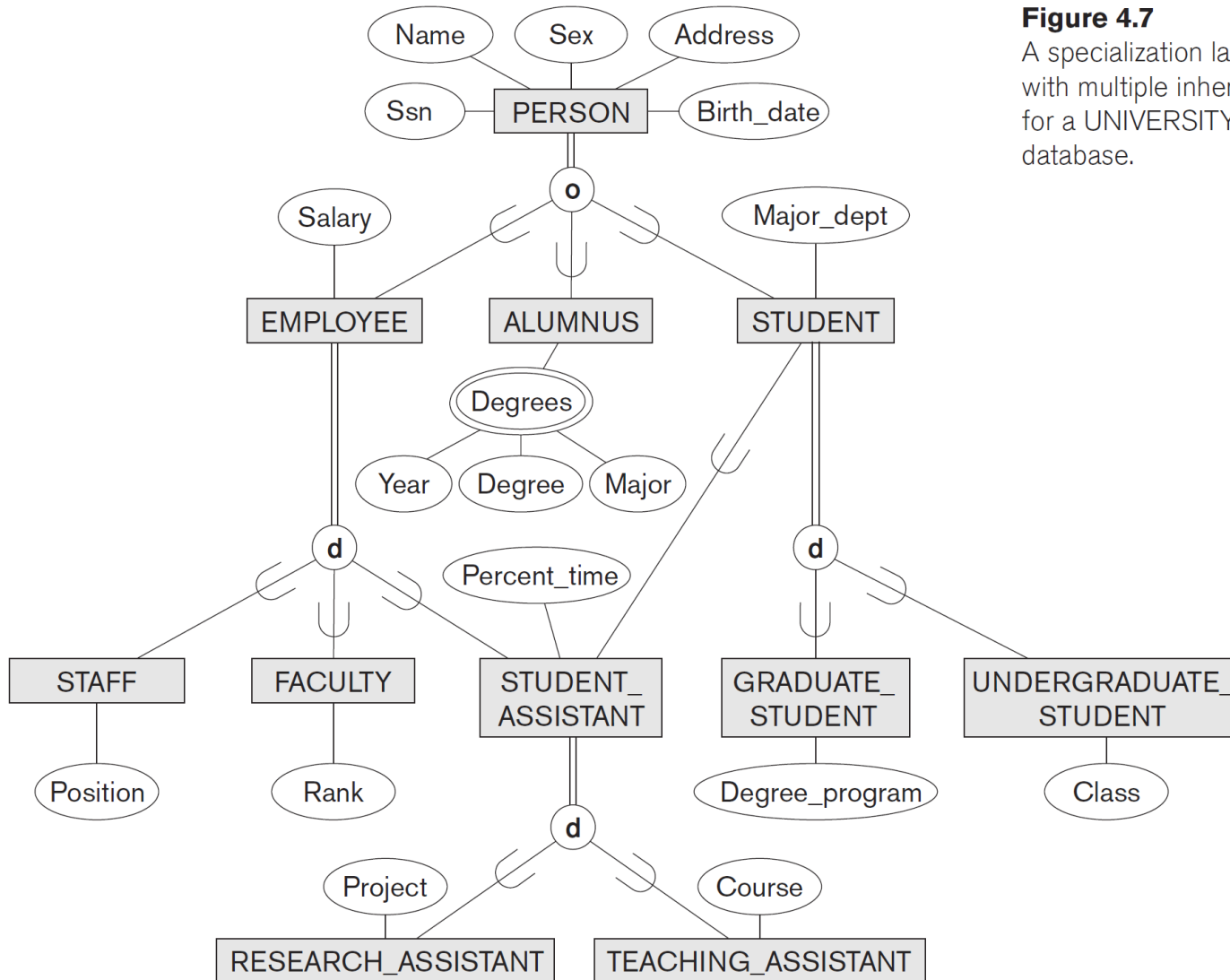
→ Disjoint
  → Is one of
→ Overlapping
  → Can be any of, and more than one.

UML Notation
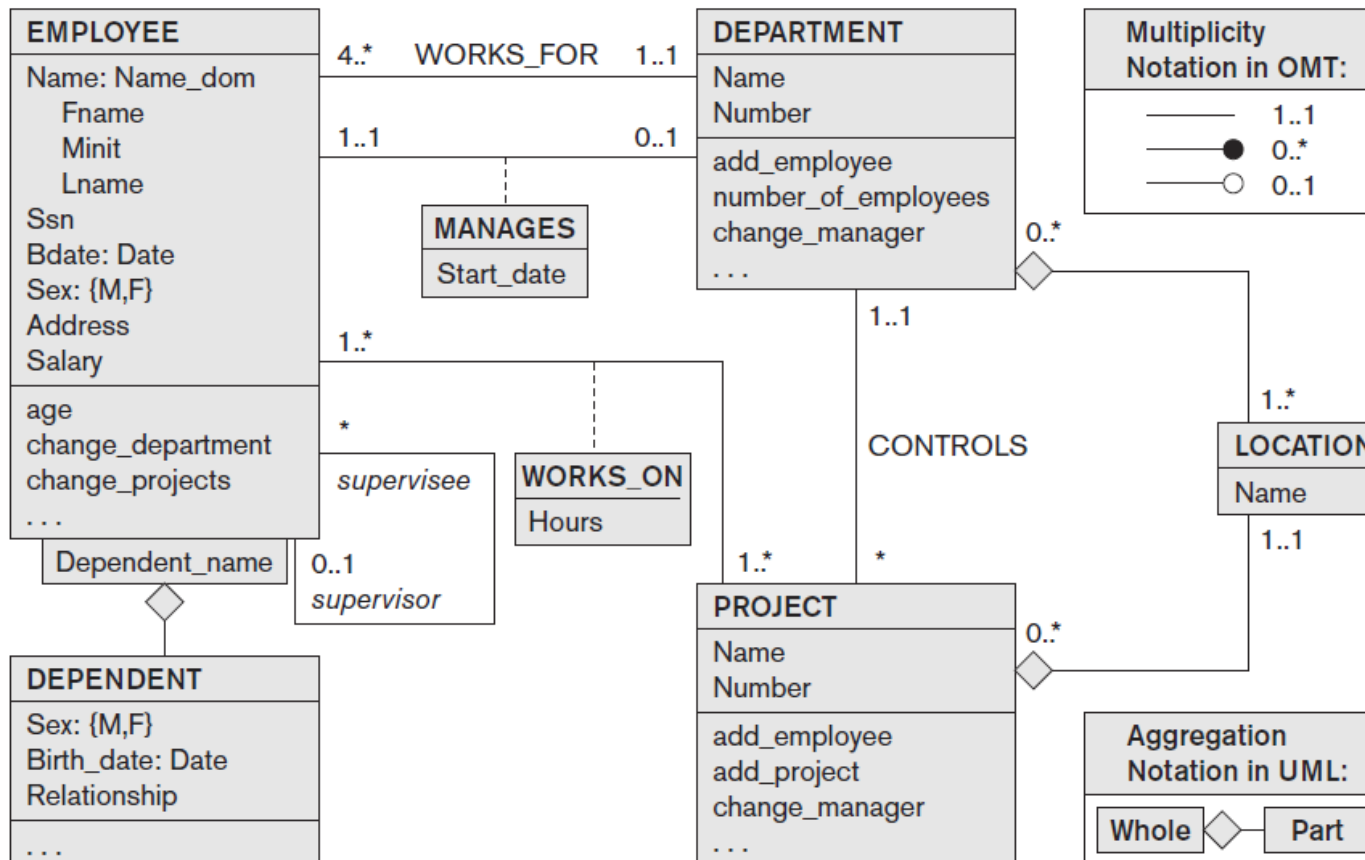
→ Often preferred due to the common use of UML for class diagrams.

→ Notation differs.

UML Entity vs Table


```
        <<Entity>>
         Students
----------------------------
+ Name
+ Email



----------------------------
                    1
                    │
                    ▼
                    *
         <<Entity>>
          Course
----------------------------
+ Name
```
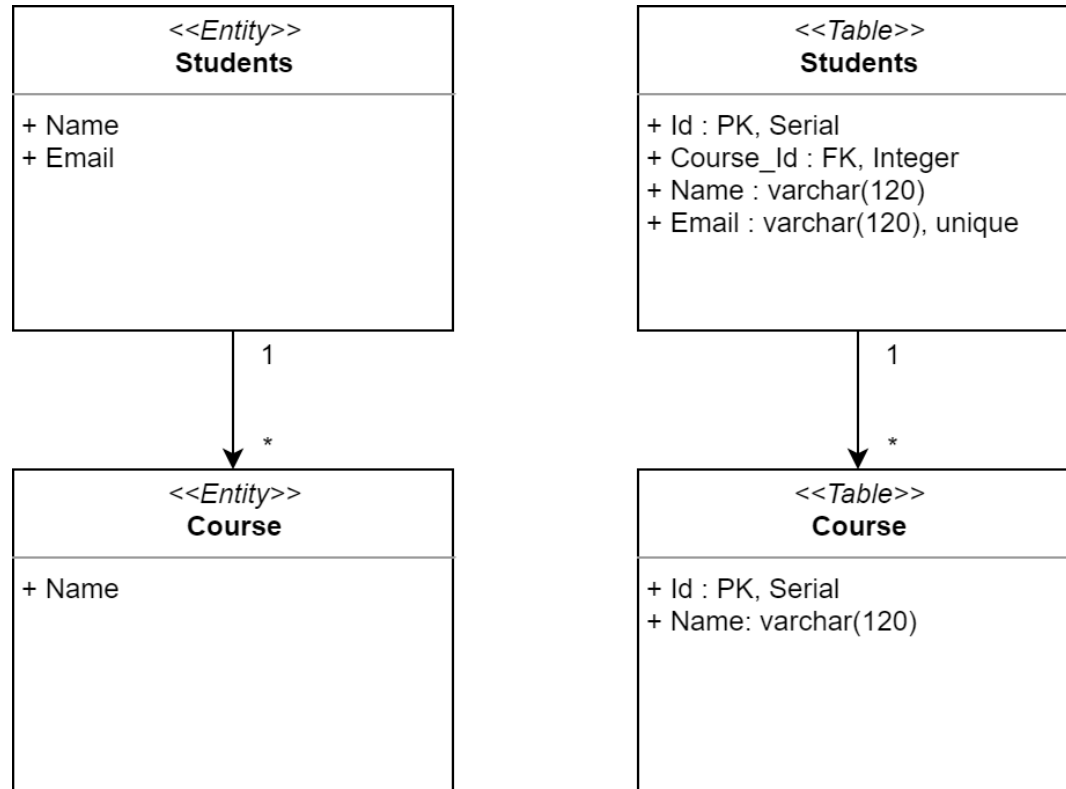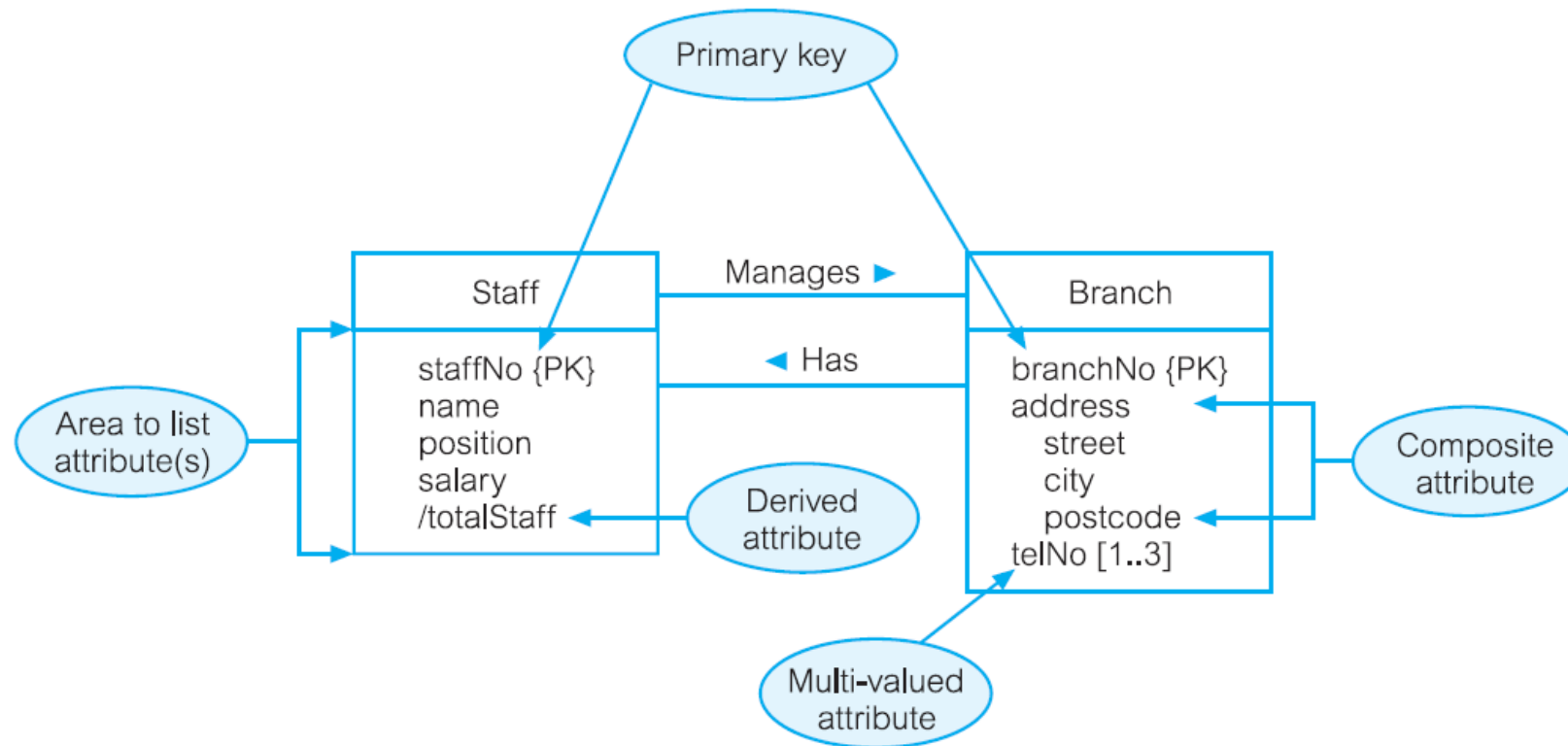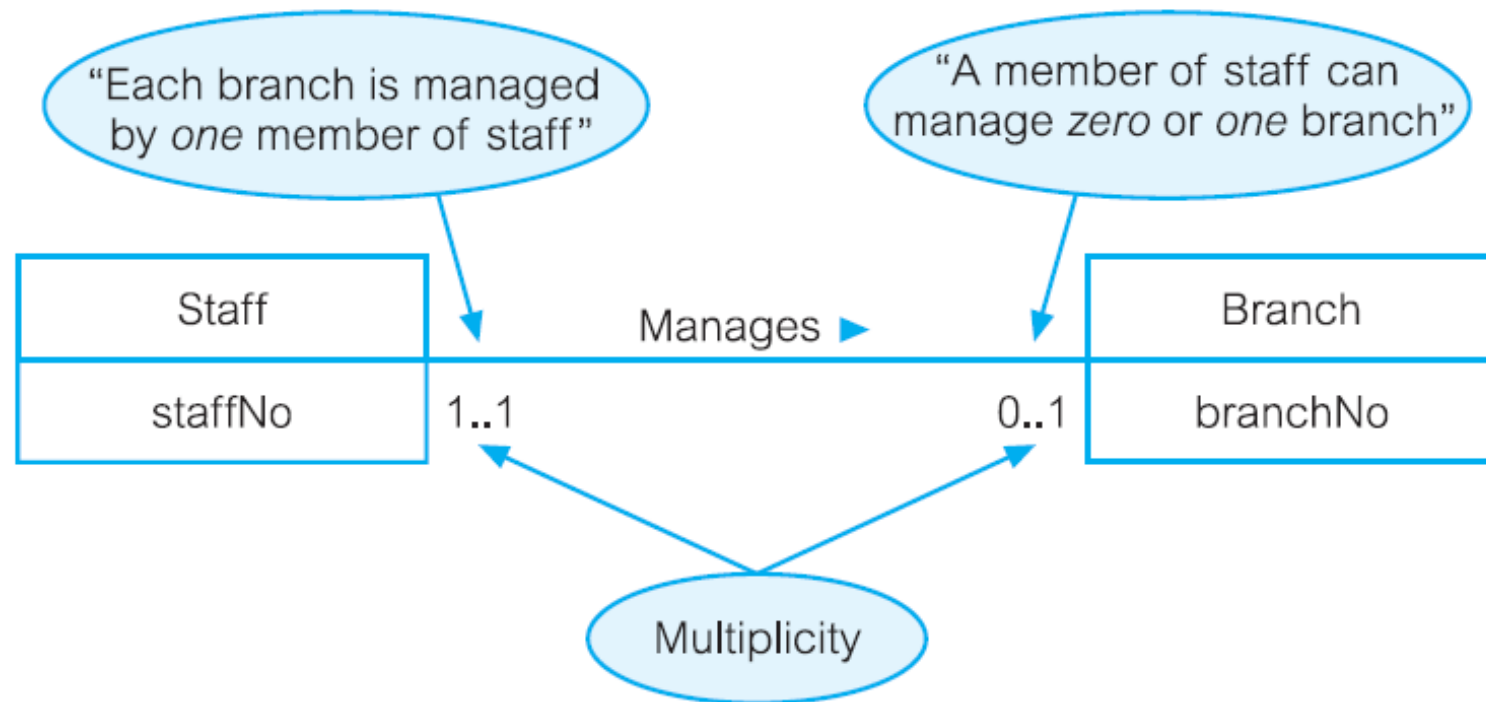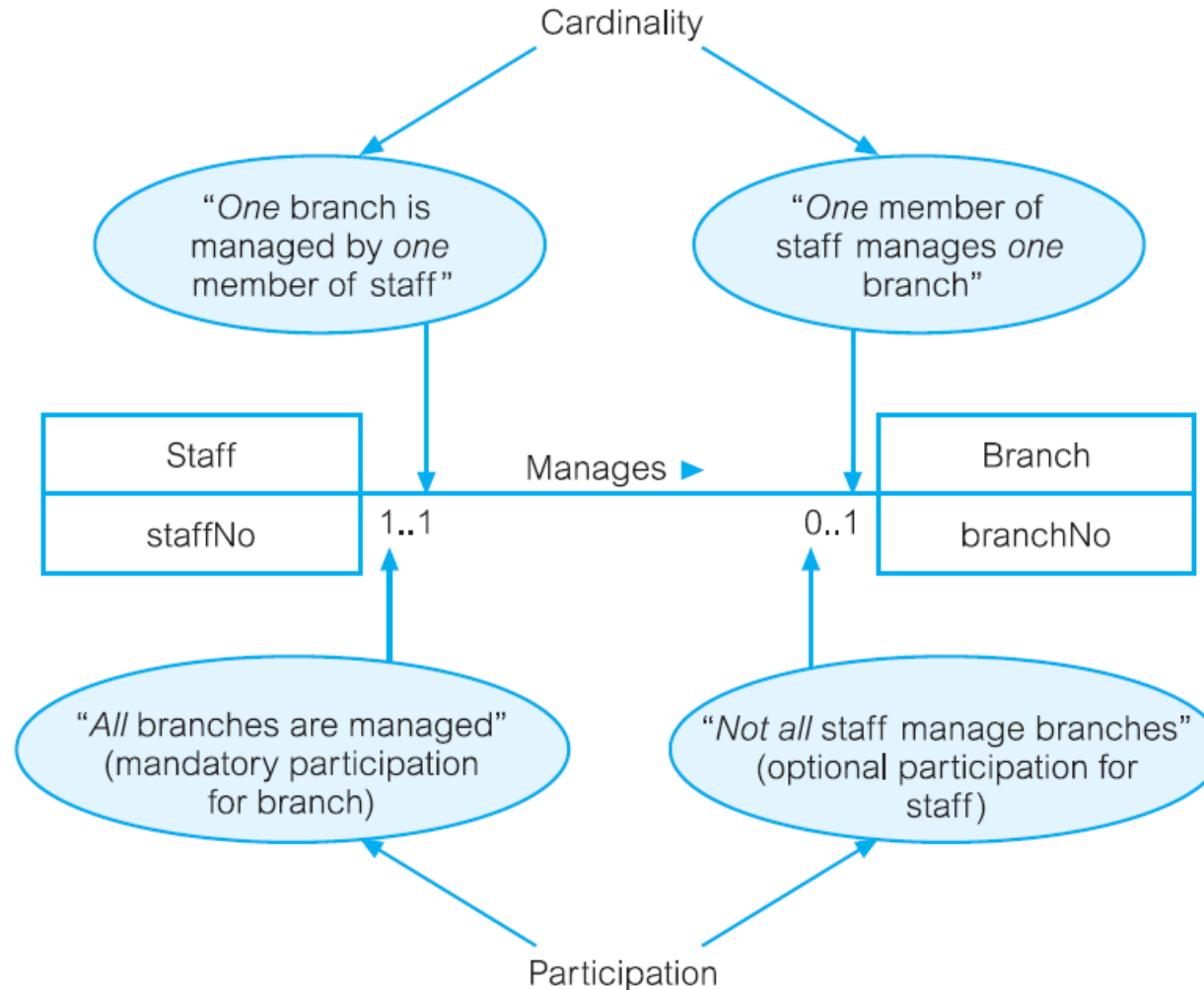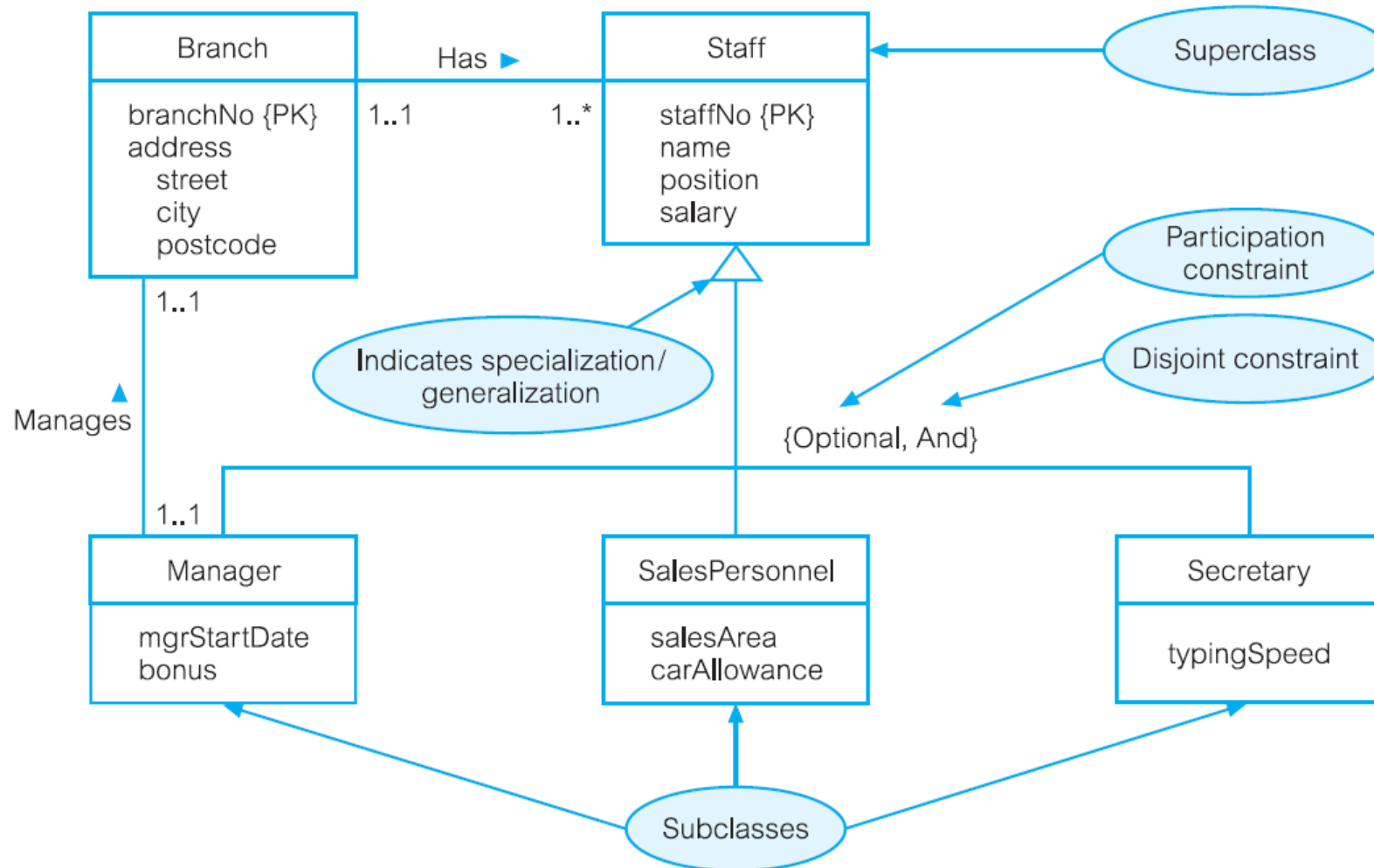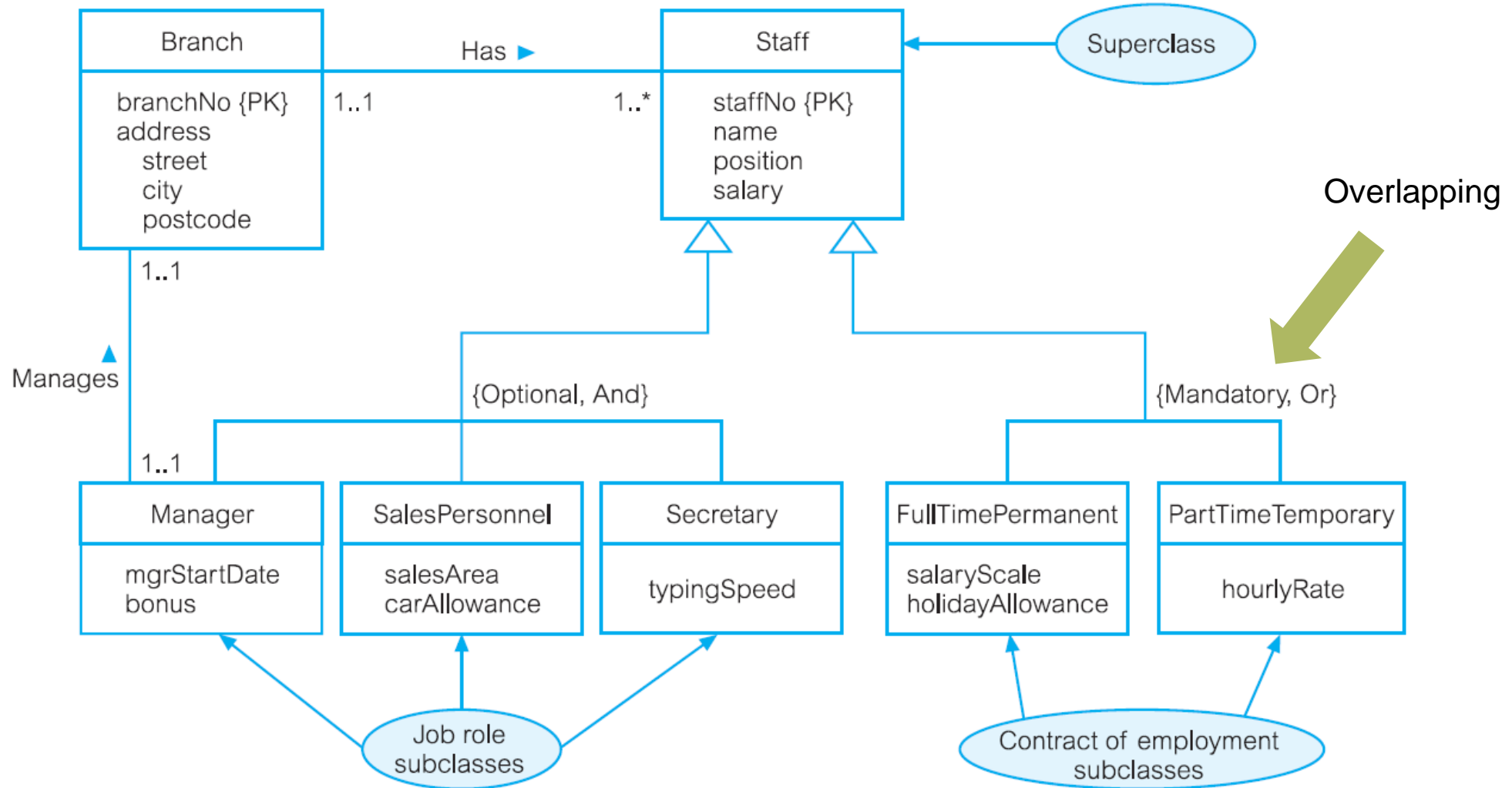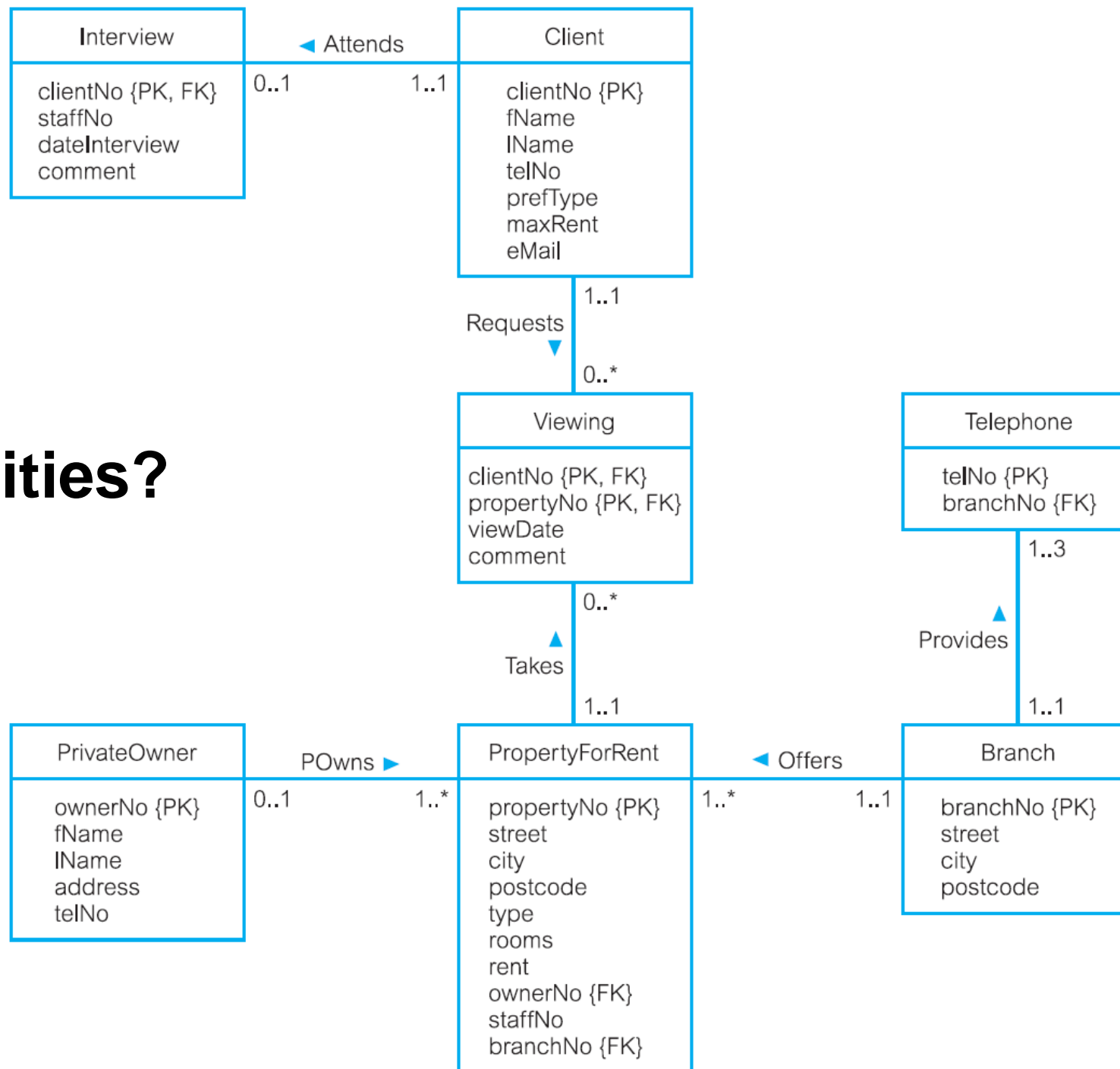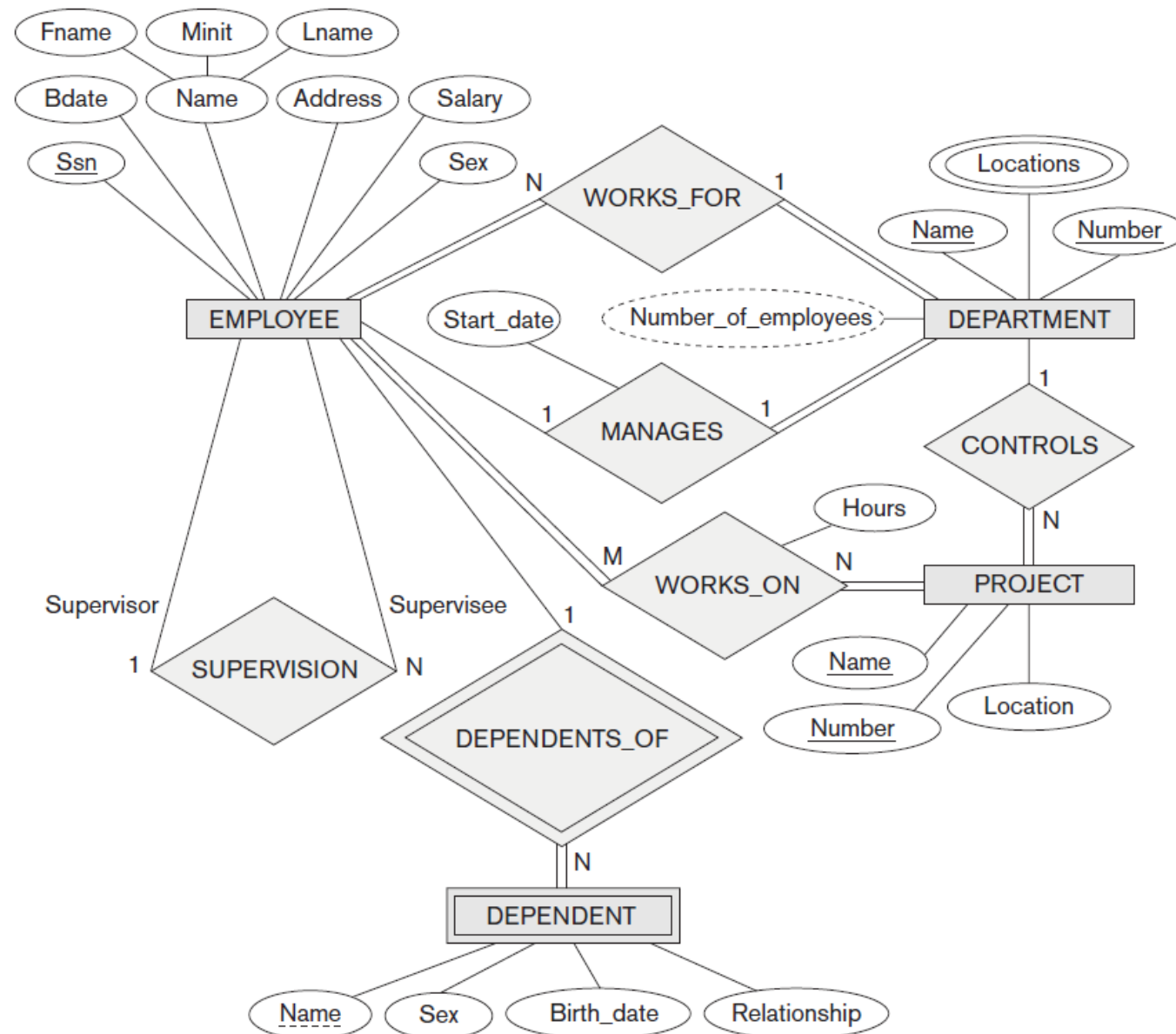
```
          <<Table>>
          Students
----------------------------
+ Id : PK, Serial
+ Course_Id : FK, Integer
+ Name : varchar(120)
+ Email : varchar(120), unique

----------------------------
                    1
                    │
                    ▼
                    *
          <<Table>>
           Course
----------------------------
+ Id : PK, Serial
+ Name: varchar(120)
```

→ If no << >> is present, we are talking about an entity.

→ **Please be <u>explicit</u> in this course about what type it is!**

Standard UML Notation

Source:
Applying UML and Patterns, Larman, 2009

Sample UML Notation

Class Diagram

Attributes in UML

Cardinalities in UML - 1/2

Cardinalities in UML – 2/2

EER in UML – 1/2

EER in UML – 2/2

Tables or entities?

Mapping ER to Tables

→ Step 1: Mapping of Regular Entity Types.

→ Step 2: Mapping of Weak Entity Types.

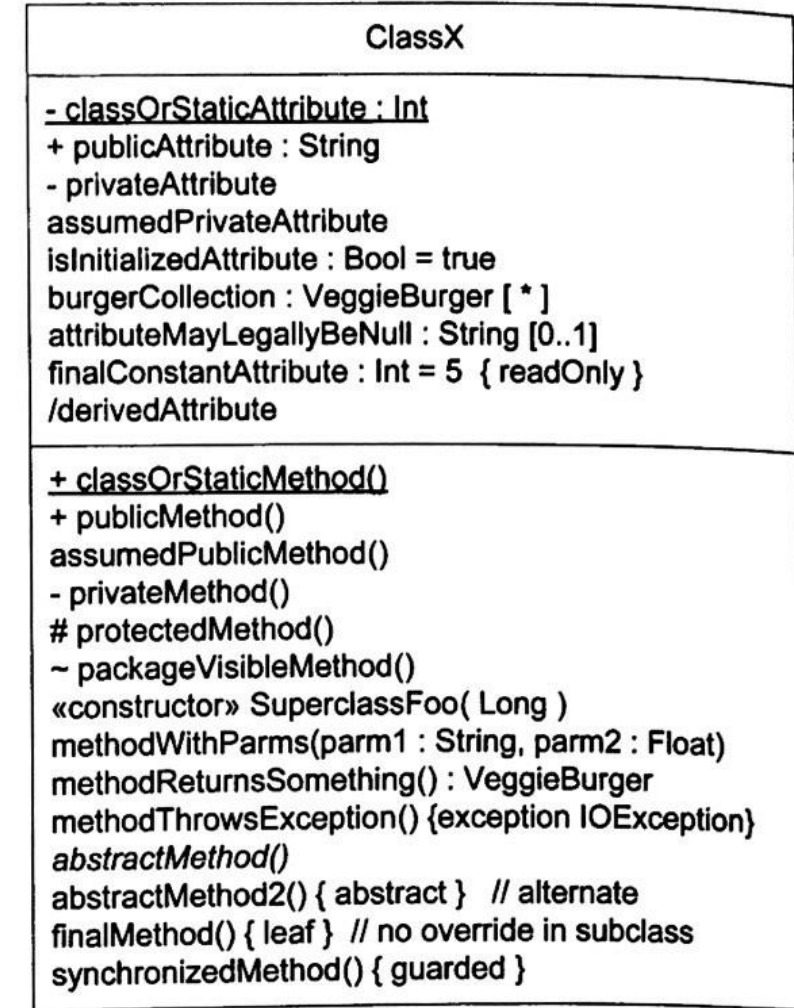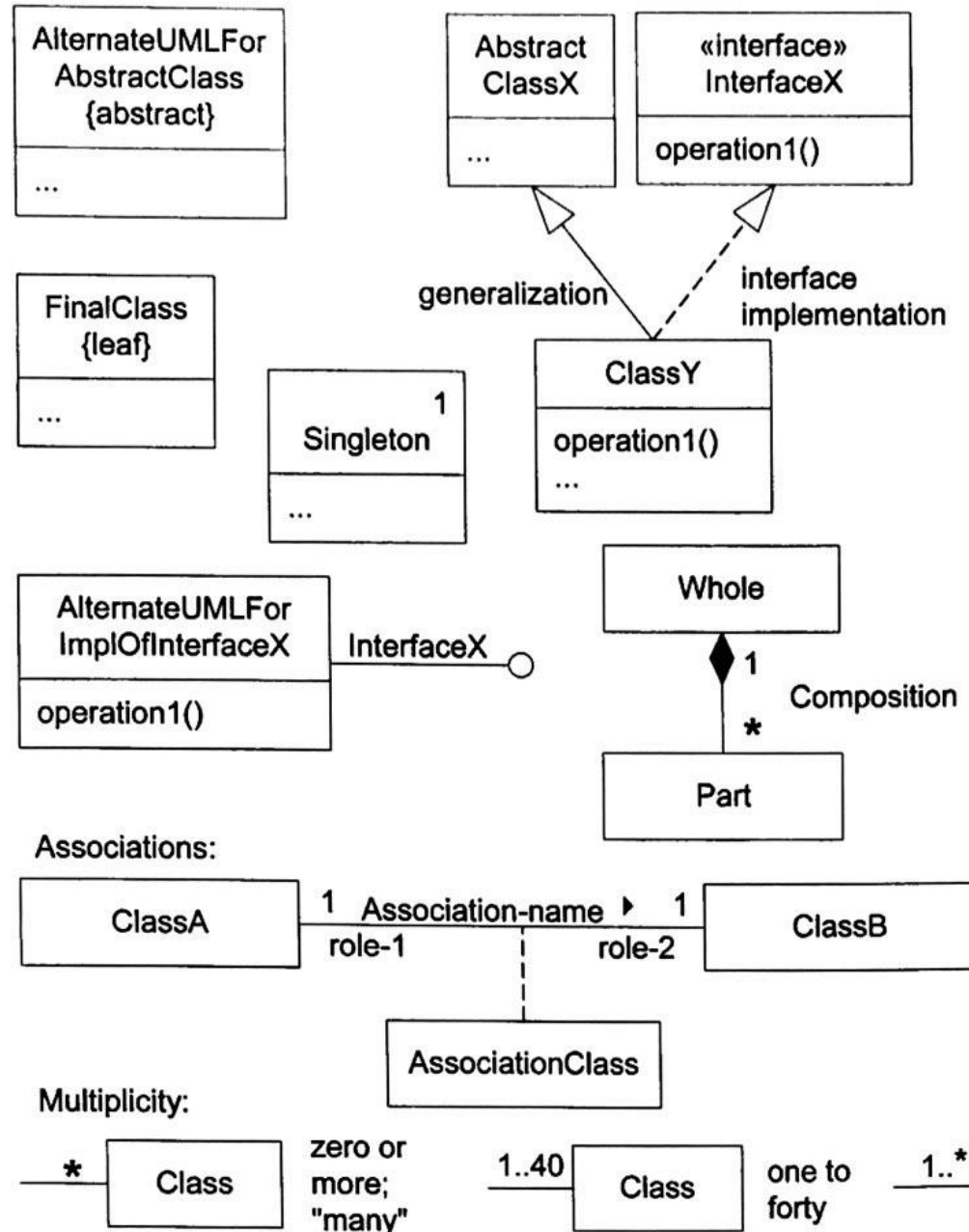→ Step 3: Mapping of Binary 1:1 Relationship Types.

  → Foreign key approach

  → Merged relation approach

  → Cross-reference or relationship relation approach

→ Step 4: Mapping of Binary 1:N Relationship Types.

  → The foreign key approach.

  → The relationship relation approach.

→ Step 5: Mapping of Binary M:N Relationship Types.

→ Step 6: Mapping of Multivalued Attributes.

→ Step 7: Mapping of N-ary Relationship Types.

Figure 9.3

Illustration of some mapping steps.
(a) *Entity* relations after step 1.
(b) Additional *weak entity* relation after step 2.
(c) *Relationship* relations after step 5.
(d) Relation representing multivalued attribute after step 6.

(a) **EMPLOYEE**

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary

DEPARTMENT

Dname	Dnumber

PROJECT

Pname	Pnumber	Plocation

(b) **DEPENDENT**

Essn	Dependent_name	Sex	Bdate	Relationship

(c) **WORKS_ON**

Essn	Pno	Hours

(d) **DEPT_LOCATIONS**

Dnumber	Dlocation

EMPLOYEE

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dno

DEPARTMENT

Dname	Dnumber	Mgr_ssn	Mgr_start_date

DEPT_LOCATIONS

Dnumber	Dlocation

PROJECT

Pname	Pnumber	Plocation	Dnum

WORKS_ON

Essn	Pno	Hours

DEPENDENT

Essn	Dependent_name	Sex	Bdate	Relationship

Figure 9.2

Result of mapping the COMPANY ER schema into a relational database schema.

Maturing the UML diagram to tables

(a) EMPLOYEE

Ssn	Fname	Minit	Lname	Birth_date	Address	Job_type

SECRETARY

Ssn	Typing_speed

TECHNICIAN

Ssn	Tgrade

ENGINEER

Ssn	Eng_type

(b) CAR

Vehicle_id	License_plate_no	Price	Max_speed	No_of_passengers

TRUCK

Vehicle_id	License_plate_no	Price	No_of_axles	Tonnage

(c) EMPLOYEE

Ssn	Fname	Minit	Lname	Birth_date	Address	Job_type	Typing_speed	Tgrade	Eng_type

(d) PART

Part_no	Description	Mflag	Drawing_no	Manufacture_date	Batch_no	Pflag	Supplier_name	List_price

In relation to UP – Repeated

Phase	Artifact
Business Modelling	Domain Model (None from DM)
Requirements	(None from DM)
Analysis	ER, EER, UML (Entities only)
Design	UML (Tables)
Implementation	SQL Creation Script
Test	(None from DM)
Deployment	(None from DM)

ER model vs Relational Model (Tables)

Table 9.1   Correspondence between ER and Relational Models

ER MODEL	RELATIONAL MODEL
Entity type	*Entity* relation
1:1 or 1:N relationship type	Foreign key (or *relationship* relation)
M:N relationship type	*Relationship* relation and *two* foreign keys
n-ary relationship type	*Relationship* relation and *n* foreign keys
Simple attribute	Attribute
Composite attribute	Set of simple component attributes
Multivalued attribute	Relation and foreign key
Value set	Domain
Key attribute	Primary (or secondary) key

Process Recap

ER / EER / UML (Entity)

Table format / UML (Table)

Remember << Table >>

SQL Creation Script

```
CREATE TABLE account(
    user_id serial PRIMARY KEY,
    username VARCHAR (50) UNIQUE NOT NULL,
    password VARCHAR (50) NOT NULL,
    email VARCHAR (355) UNIQUE NOT NULL,
    created_on TIMESTAMP NOT NULL,
    last_login TIMESTAMP
);
```

sdu.dk

#sdudk

Live Demo (draw.io)

Pay attention, and don't try to replicate what I do right now!

You will have time to do that afterwards.

Exercises

→ Notes from the interview:
  → Vi i Kajak klubben vil gerne have et system til at holde styr på vores medarbejdere, medlemmer, nyhedsbreve, bådpladser, og udlejningsbåde.
  → Vores medlemmer får nuværende et medlemskort med Medlemsnummer og deres navn, og vi holder styr på informationerne i et Excel ark lige nu.
  → Arrangementer: Medlemmerne kan også logge på deres hjemmeside og melde sig til arrangementer. Her kan de se et navn på eventet, tiden det kører, samt et billede og en kortere tekst der beskriver eventet. Hvert event kan oprettes af ethvert medlem, og har et enkelt ansvarligt medlem for arrangementet, og multiple hjælpere. Disse hjælpere skal være medlemmer. På hjemmesiden kan man se telefonnummer på arrangøren og medhjælperne, så man kan kontakte dem før et event.
  → Nyhedsbreve: Det er ikke alle medlemmer der vil have nyhedsbreve. Nuværende ringer de ind og siger de ikke længere vil have dem. Nyhedsbrevene kan sendes både på e-mail og via snail-mail. Medarbejderne får også nyhedsbreve, men dette er et medarbejder nyhedsbrev, ikke det som medlemmerne får.
  → Udlejningsbåde kan lejes af folk uden for klubben også, og her kan man også bede om Nyhedsbreve (kun via email), men disse kunder har ikke et medlemsnummer.
→ Based on the interview notes, create the following:
  → ER Diagram - Important
  → Map to UML diagram (Table entities) – Optional, but recommended
  → Create the database - Important
→ If some information is missing, make assumptions about what attributes are needed for the database to make sense.

SDU✿ **Jakob Hviid**   **33**