Data Modelling/Data Base Systems

VU 184.685/VU 184.686, WS 2020

Data Base Design

Anela Lolić

Institute of Logic and Computation, TU Wien

Acknowledgements

The slides are based on the slides (in German) of Sebastian Skritek.

The content is based on Chapter 2 of
(Kemper, Eickler: Datenbanksysteme – Eine Einführung).

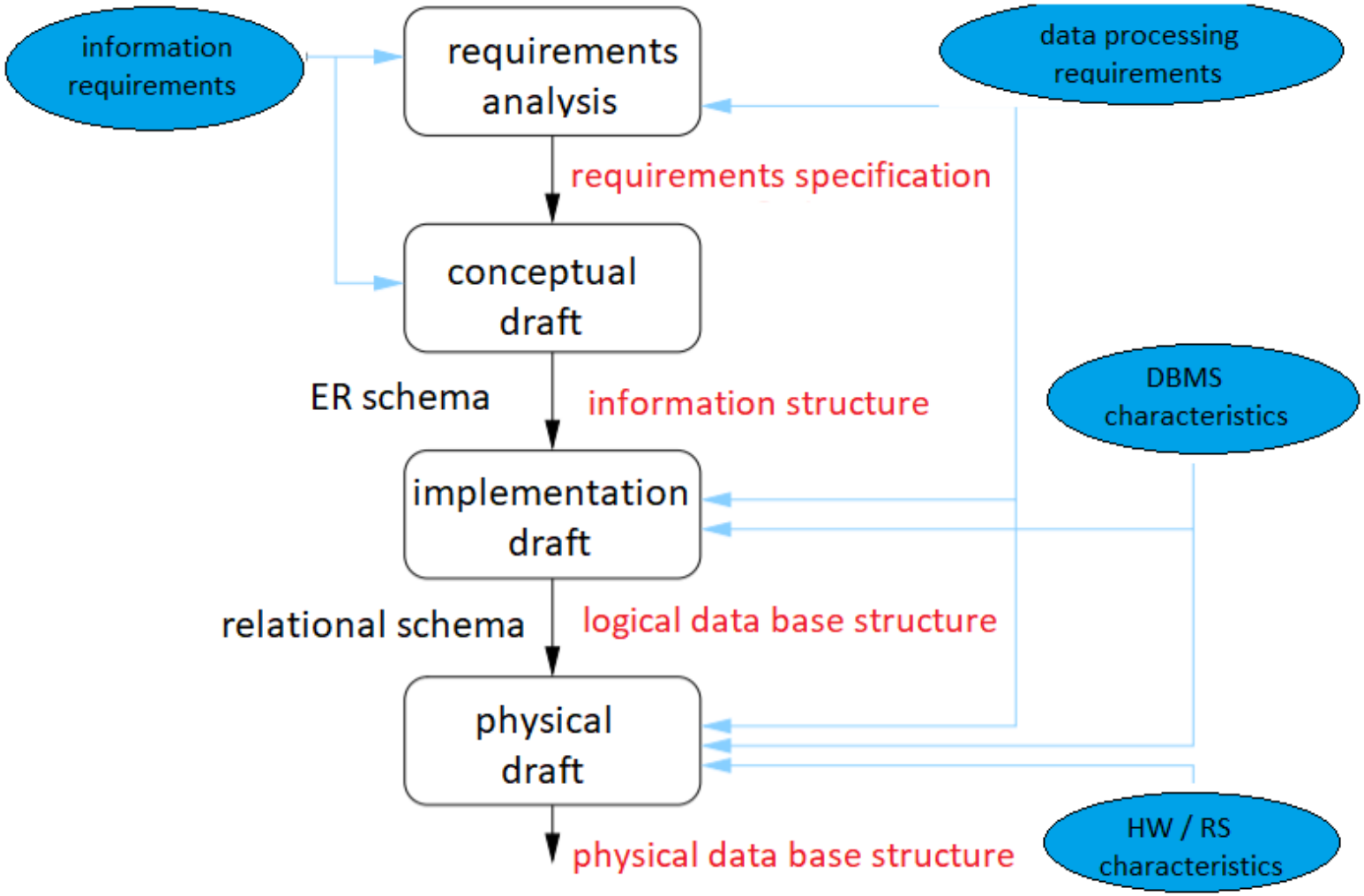
For related literature in English see Chapter 2 of
(Ramakrishnan, Gehrke: Database Management Systems).

Overview

1. Data Base Design Steps
2. The Entity-Relationship (ER) Model
3. View Integration and Consolidation

Overview

Data Base Design Steps



Requirements Analysis

1. identification of organization units
2. identification of the tasks to be supported
3. requirements plan
4. requirements collection
5. filtering (checking comprehensibility and uniqueness of information)
6. classification (objects, relations, operations, events)
7. formalization - construction of a functional specification:
   - information structure requirements: structured information about
     - objects
     - attributes
     - relationships
   - data processing requirements: structured information about process descriptions

Object and Attribute Description

objects

- university staff:
  - number: 1000
  - attributes:
    - persNr
    - name
    - salary
    - rank
- students:
  - number: 20.000
  - attributes:
    - matrNr
    - name
    - address

Object and Attribute Description

objects

- university staff:
  - number: 1000
  - attributes:
    - persNr
    - name
    - salary
    - rank
- students:
  - number: 20.000
  - attributes:
    - matrNr
    - name
    - address

attributes

- persNr
  - type: char
  - length: 9
  - domain: 0...999.999
  - number of iterations: 0
  - definedness: 100%
  - identifying: yes
- salary
  - type: decimal
  - length: (8,2)
  - number of iterations: 0
  - definedness: 90%
  - identifying: no

Relationship Description

relationship *examine*

- involved objects:
  - professors as examiner
  - students as examinee
  - lecture as assessment load
- attributes of the relationship:
  - date
  - time
  - grade
- number: 100 000 per year

Process Description

process: *issuing of a certificate*

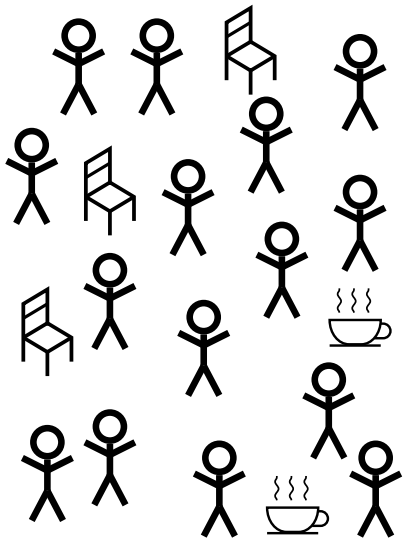
- frequency: semiannual
- required data:
  - exams
  - study regulations
  - information about students
  - ...
- priority: high
- amount of data to be processed:
  - 500 students
  - 3000 exams
  - 10 study regulations

Overview

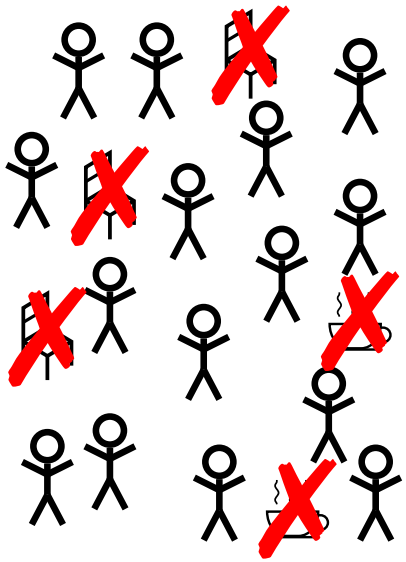
The Entity-Relationship (ER) Model

1. entities and relationships
2. roles and attributes
3. keys
4. cardinalities
5. (min,max)-notation
6. weak entities
7. generalization or specialization (EER)
8. aggregation (EER)

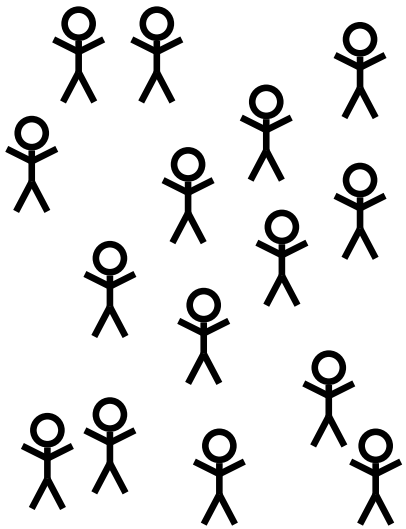
Conceptual Design Through ER Diagrams



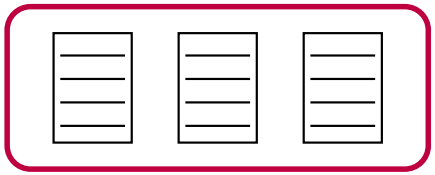
Conceptual Design Through ER Diagrams

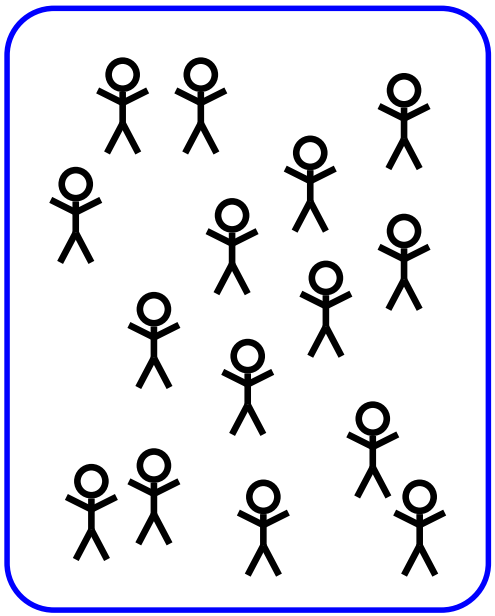


Conceptual Design Through ER Diagrams

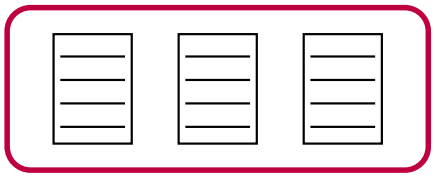


Conceptual Design Through ER Diagrams

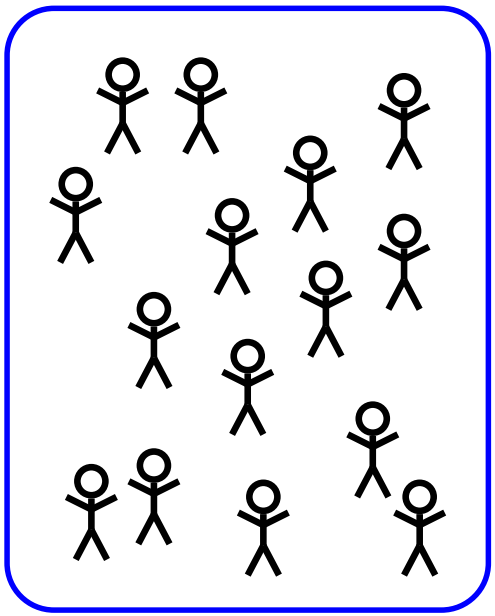




Conceptual Design Through ER Diagrams

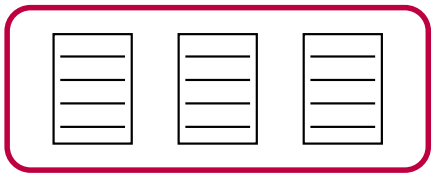


exam



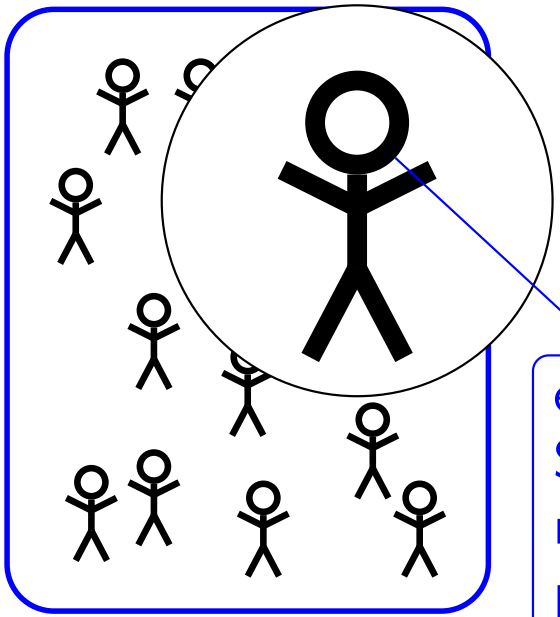
person

Conceptual Design Through ER Diagrams



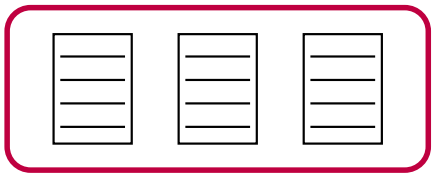
exam

person

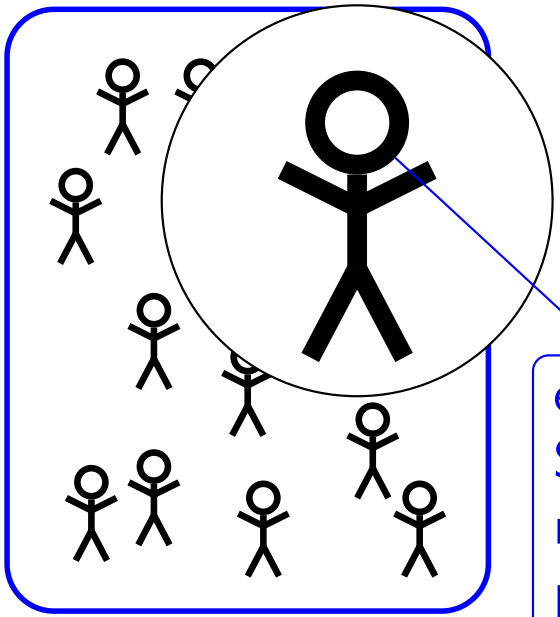


eyes: green
SVNR: 001
name: Alice
pet: cat

Conceptual Design Through ER Diagrams



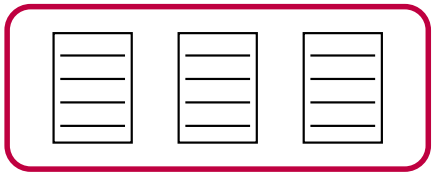
exam



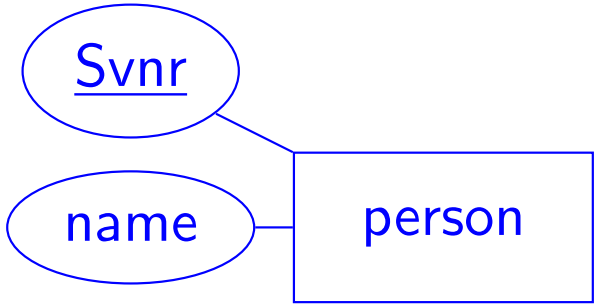
person

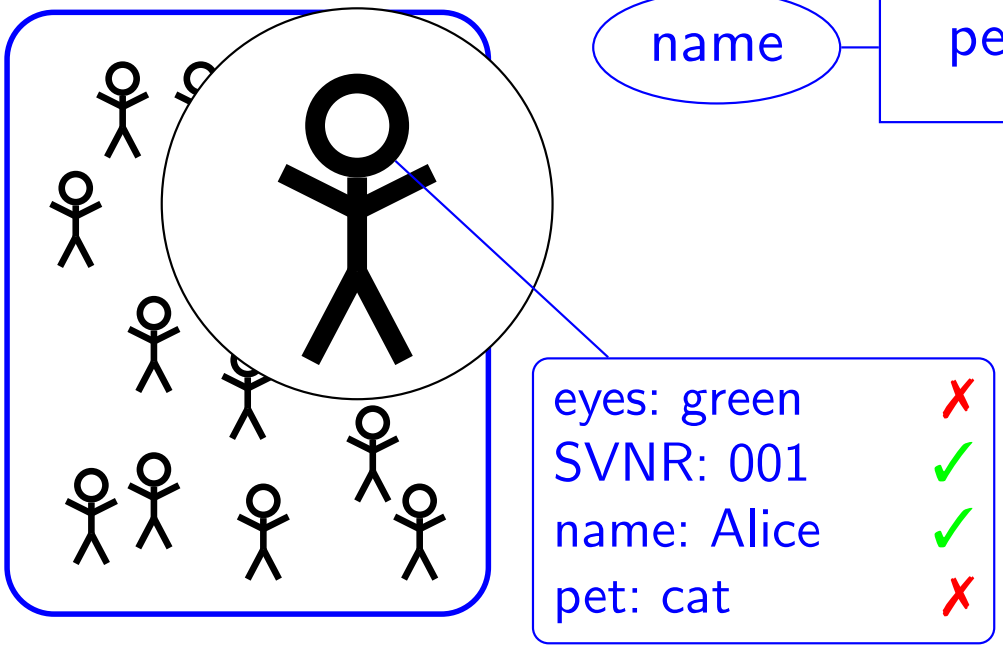
eyes: green ✗
SVNR: 001 ✓
name: Alice ✓
pet: cat ✗

Conceptual Design Through ER Diagrams

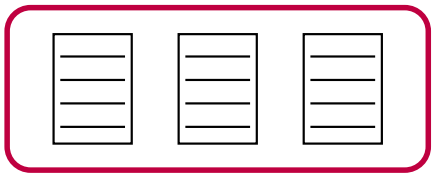


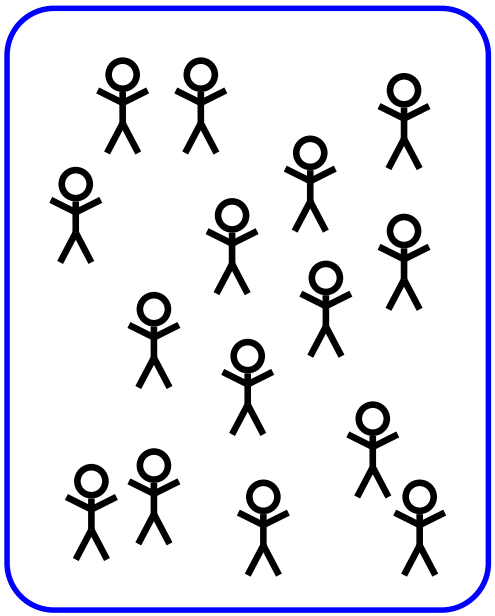
exam

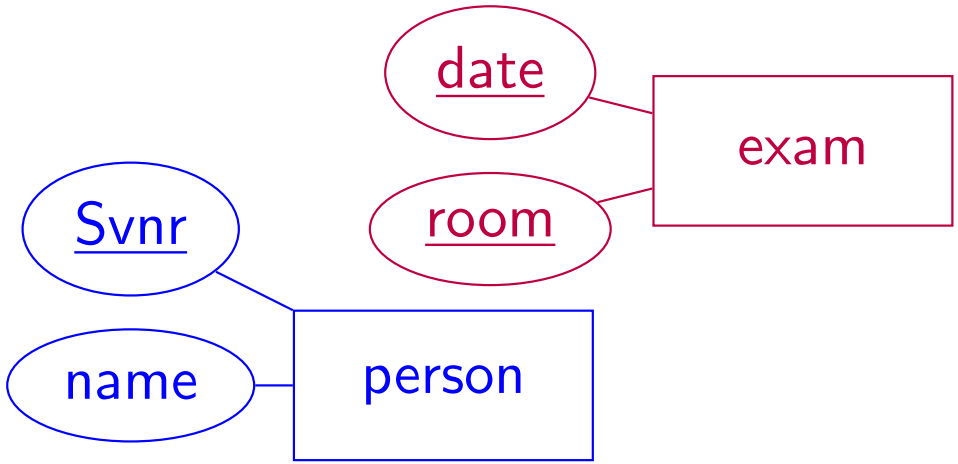




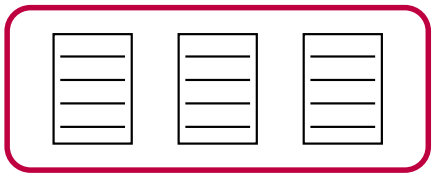
Conceptual Design Through ER Diagrams

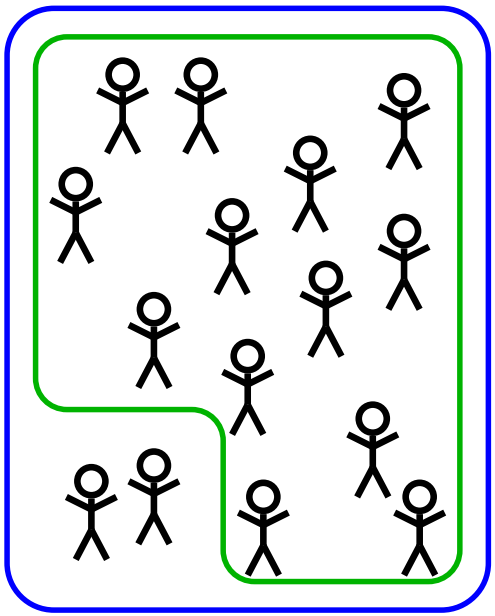


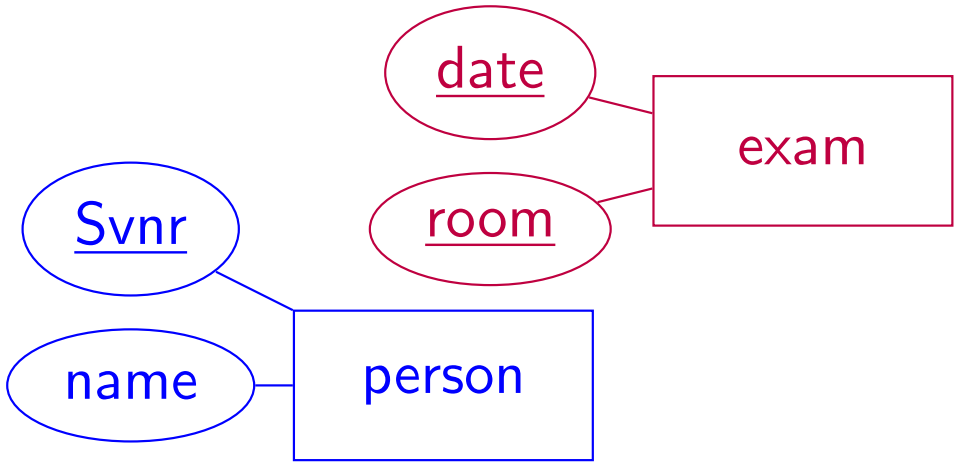


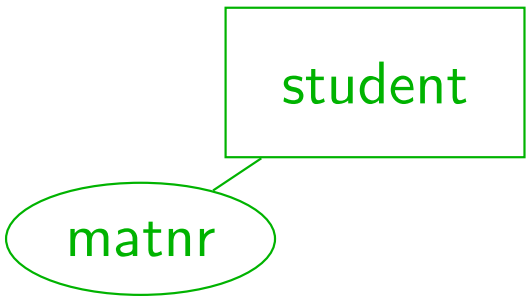


Conceptual Design Through ER Diagrams

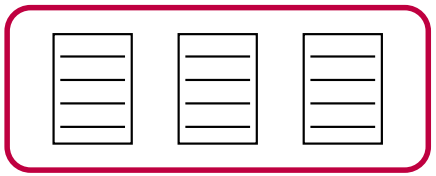


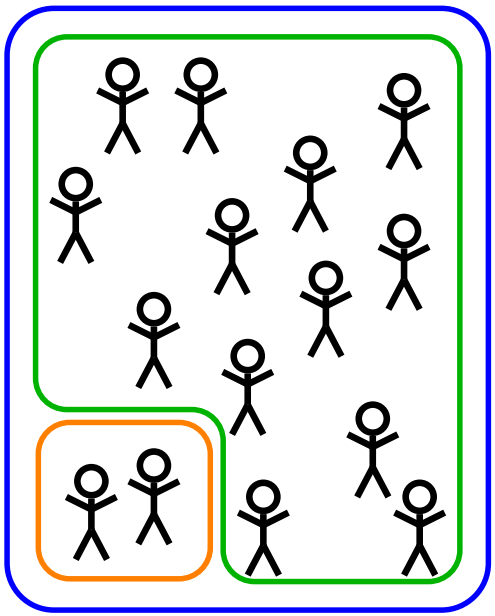


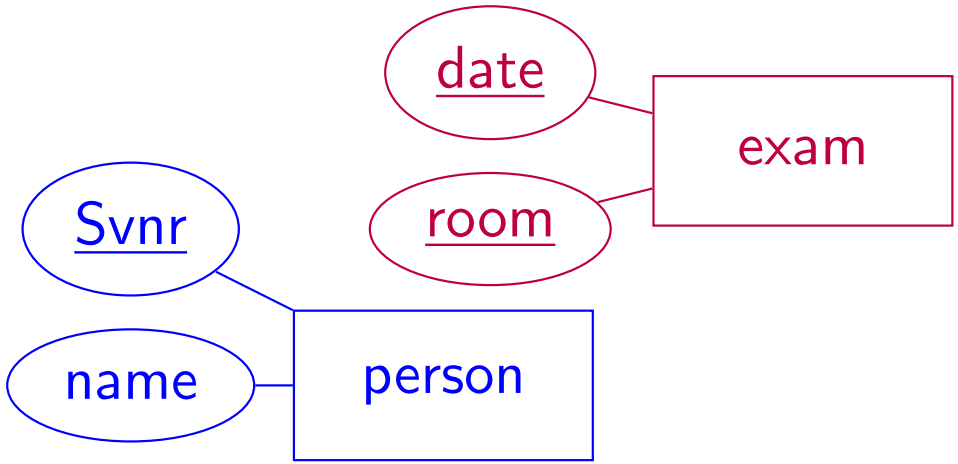


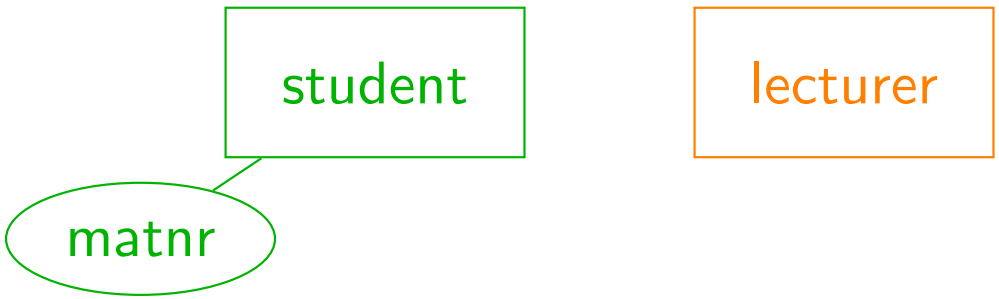


Conceptual Design Through ER Diagrams

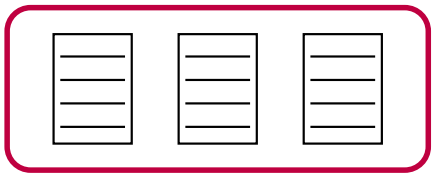


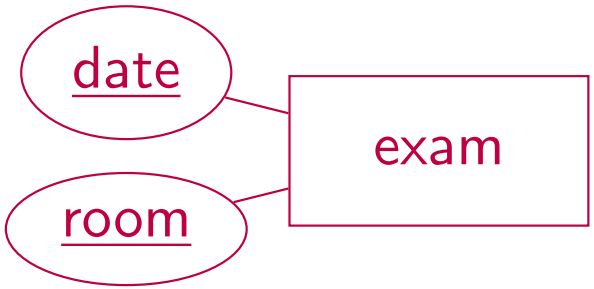


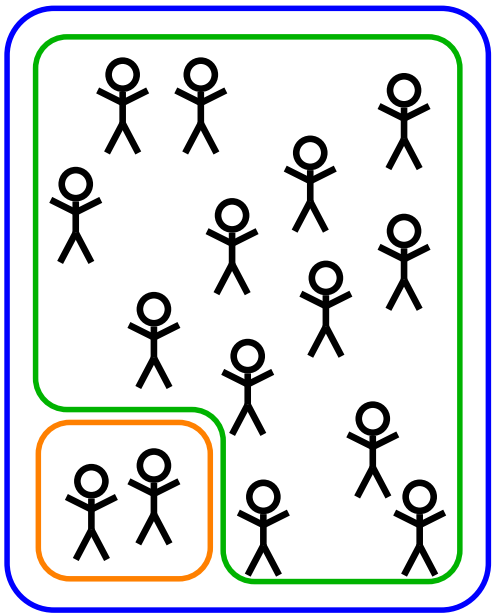


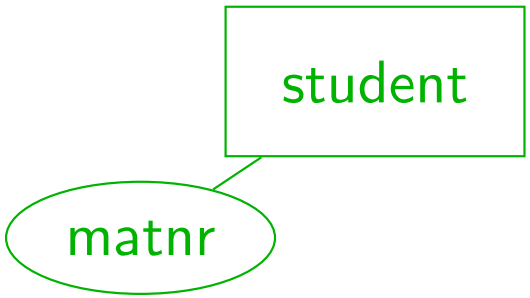


Conceptual Design Through ER Diagrams

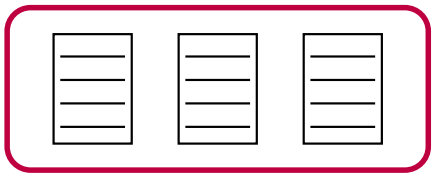


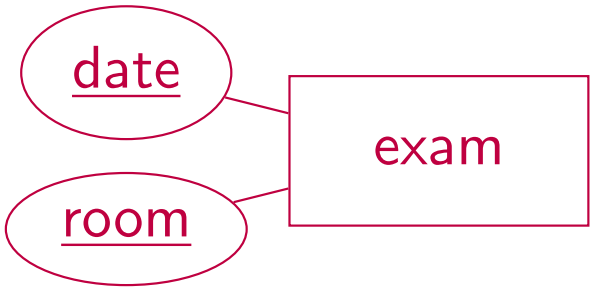


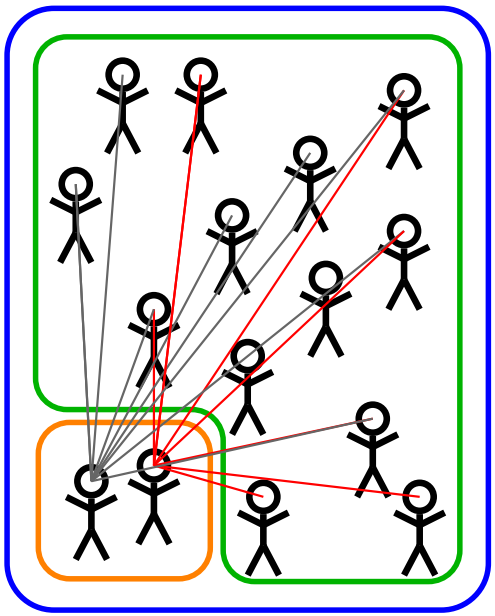


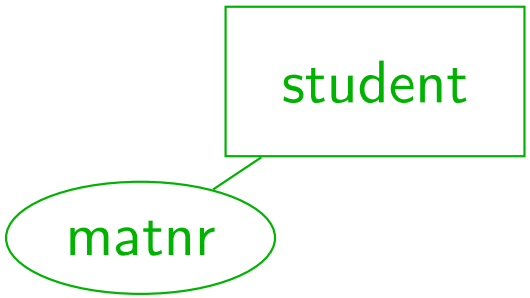


Conceptual Design Through ER Diagrams

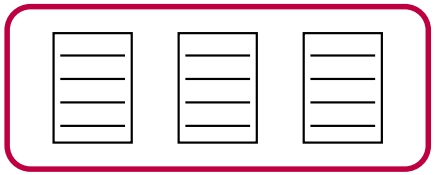


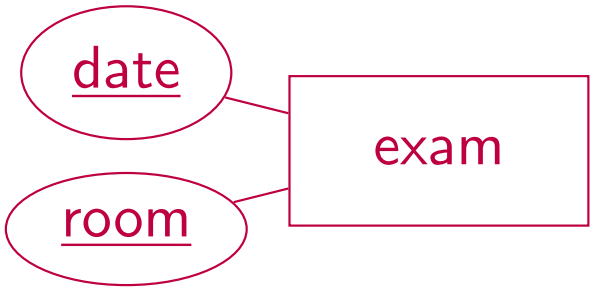


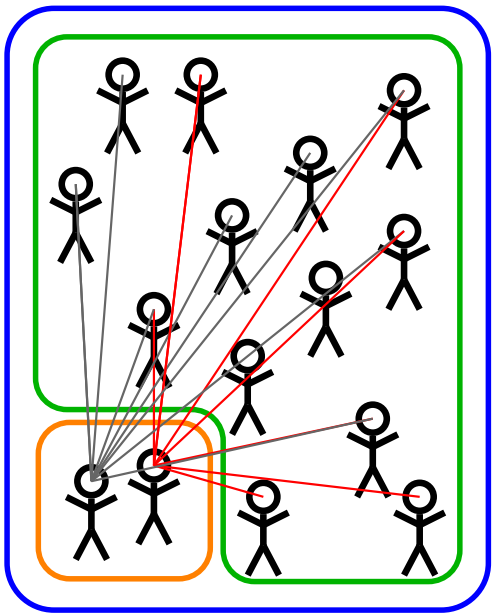


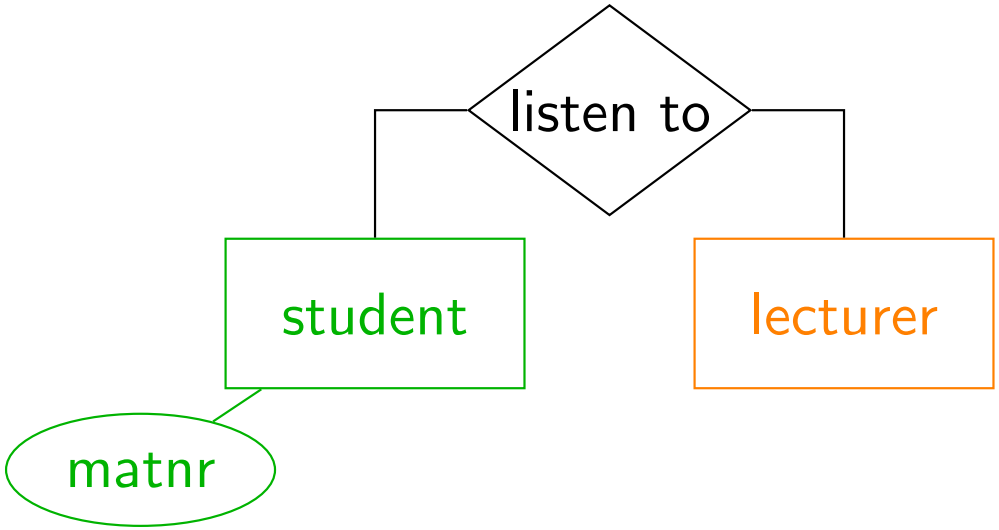


Conceptual Design Through ER Diagrams

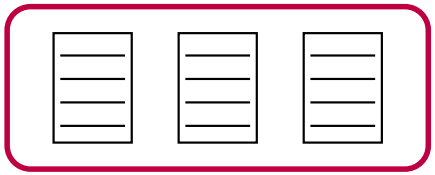


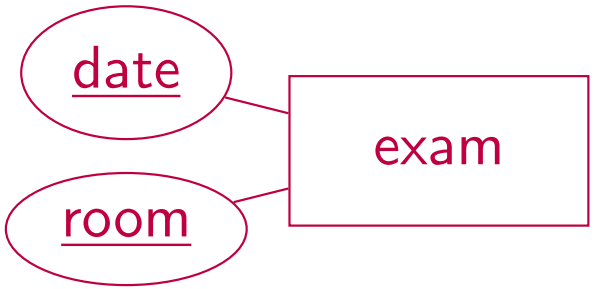


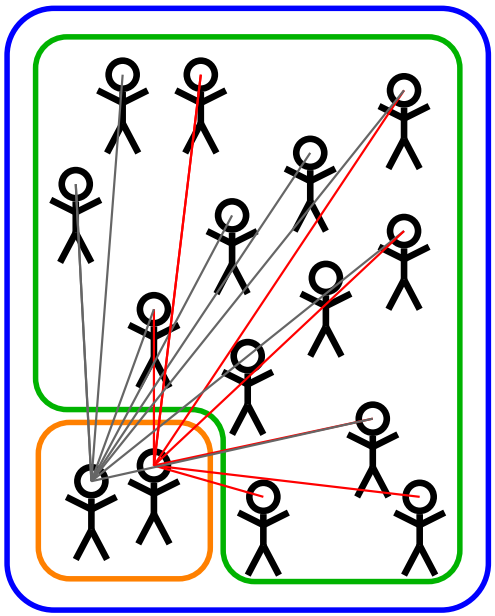


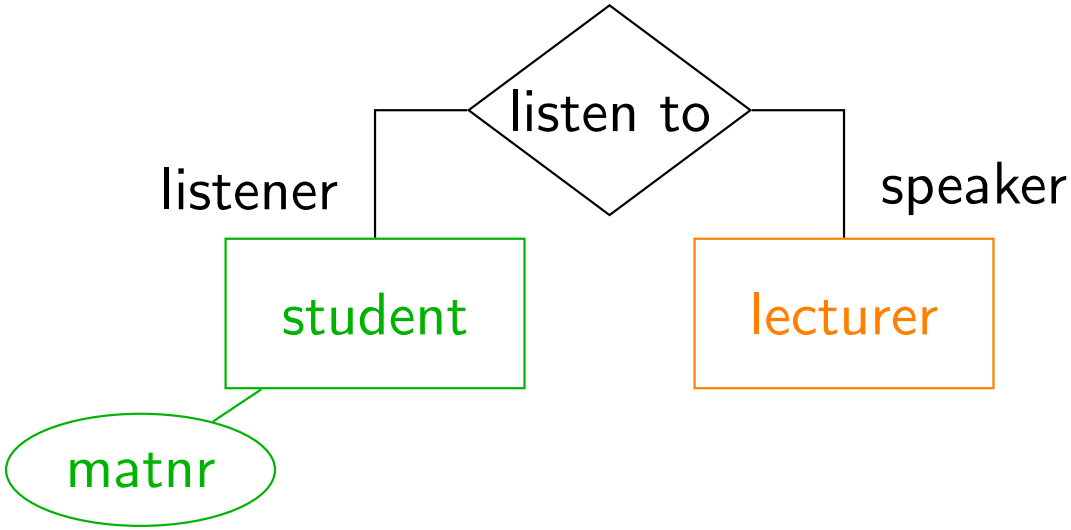


Conceptual Design Through ER Diagrams

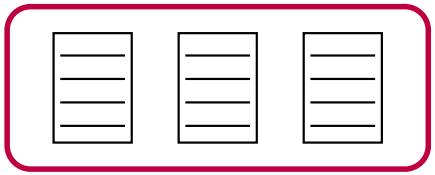


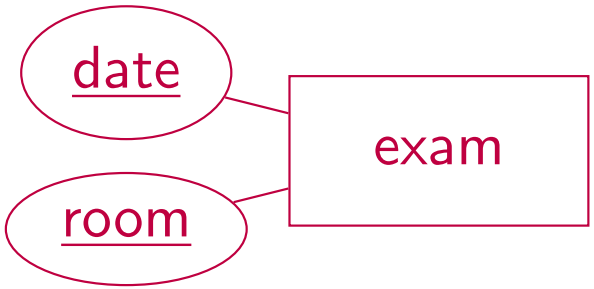


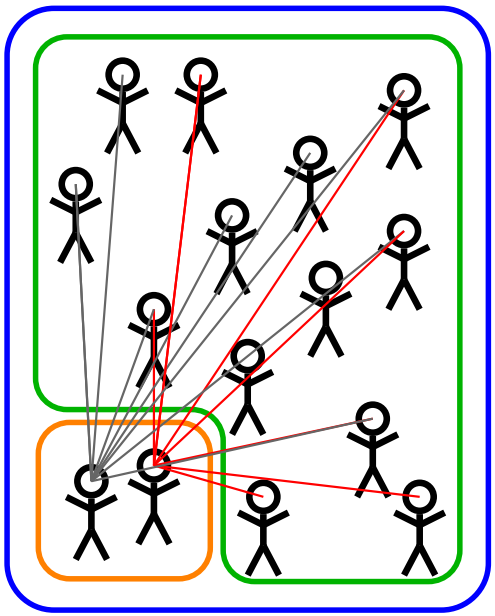


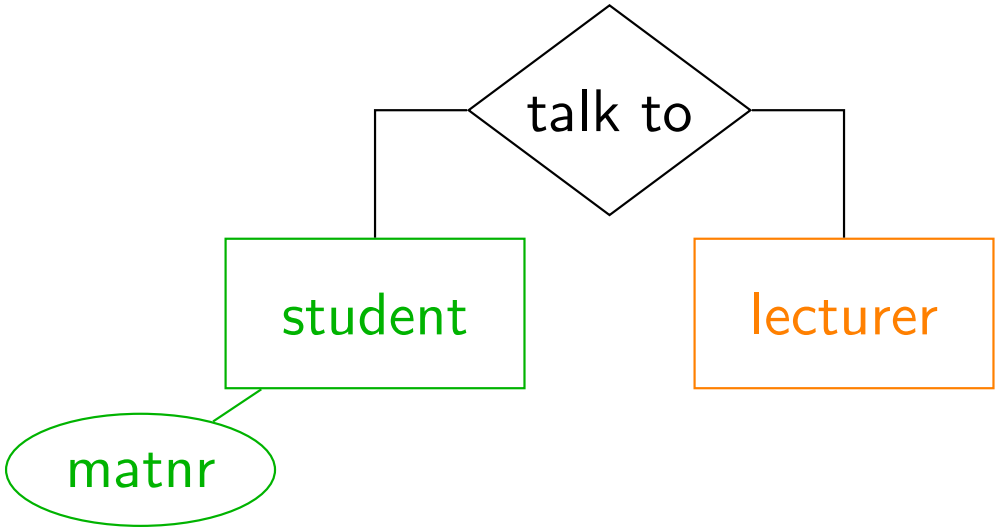


Conceptual Design Through ER Diagrams

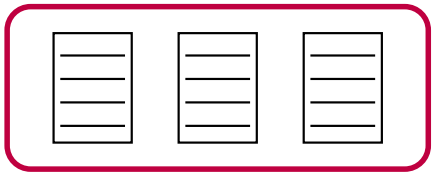


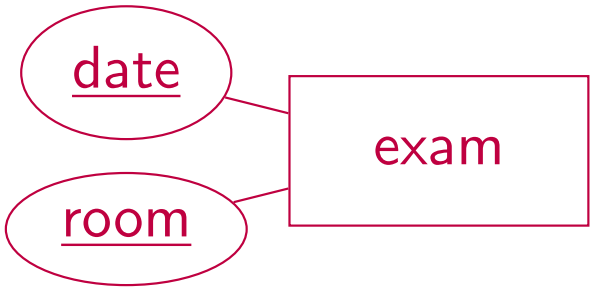


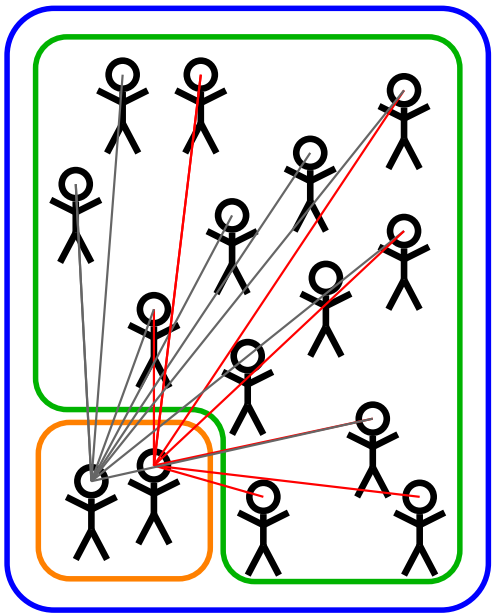


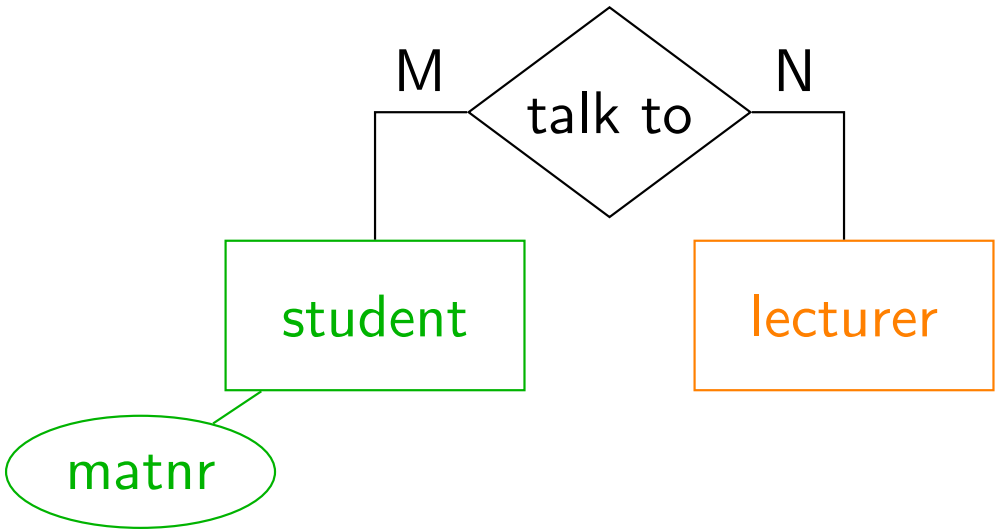


Conceptual Design Through ER Diagrams

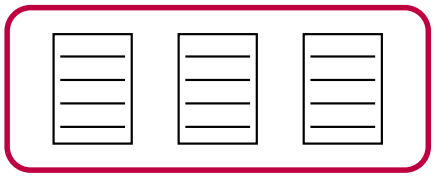


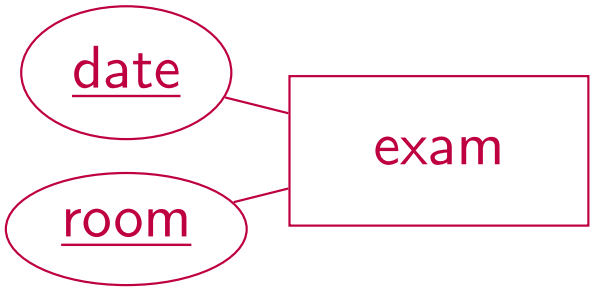


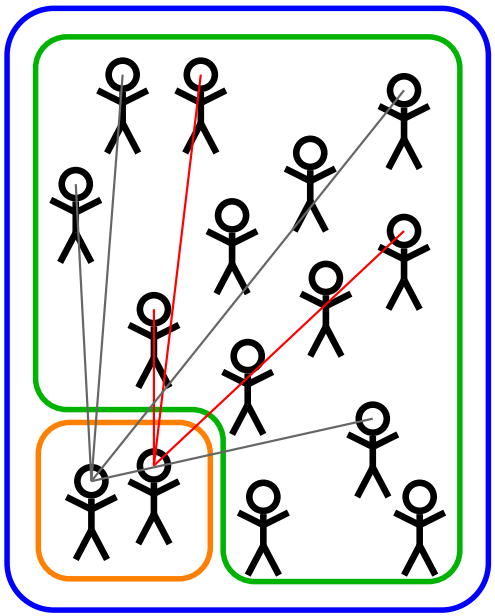


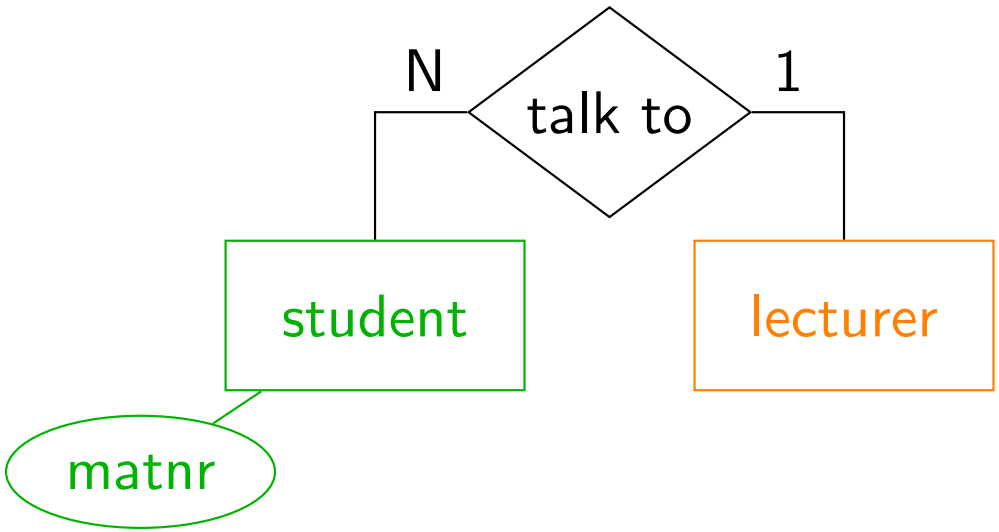


Conceptual Design Through ER Diagrams

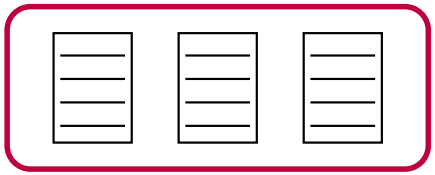


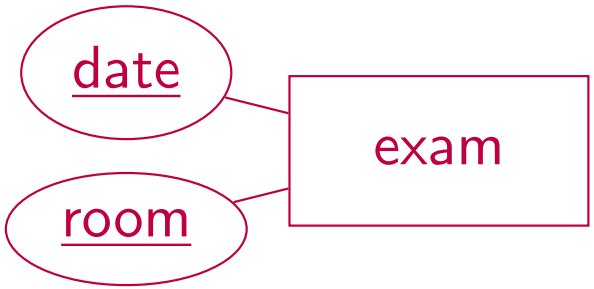


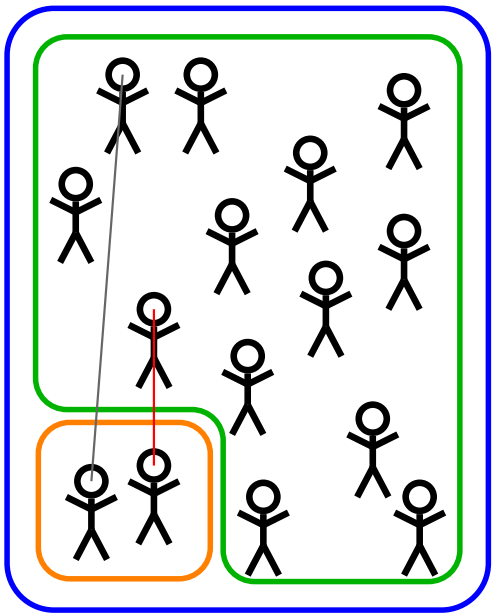


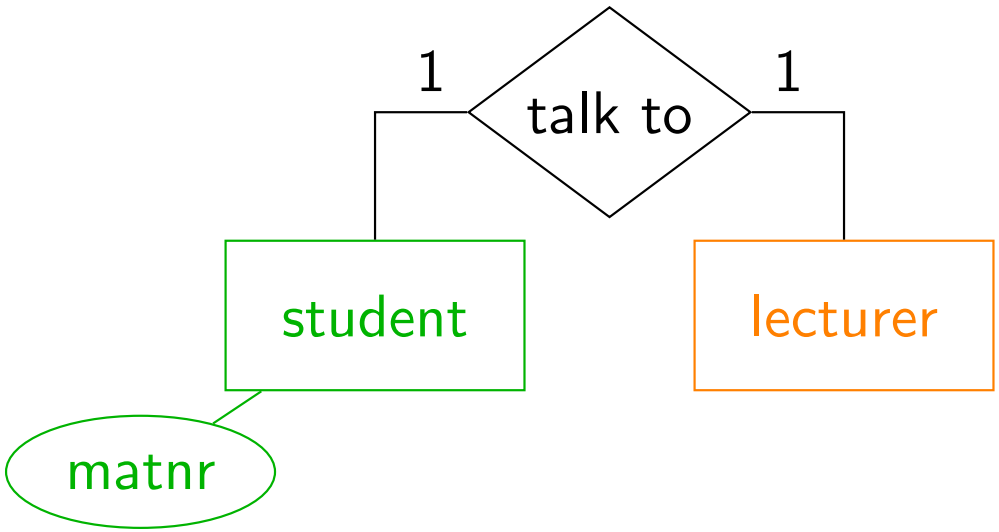


Conceptual Design Through ER Diagrams

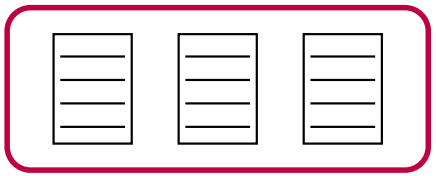


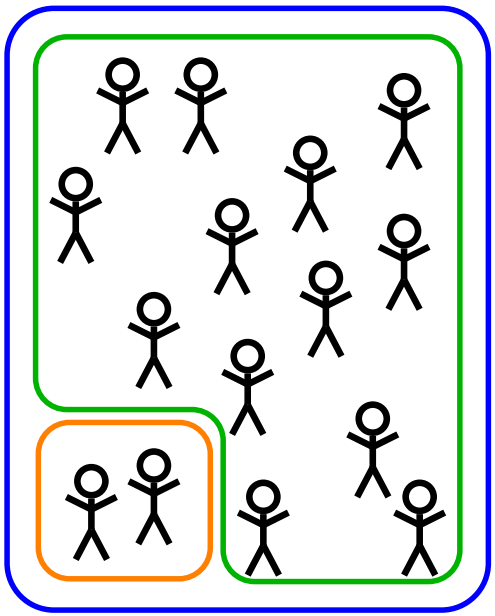


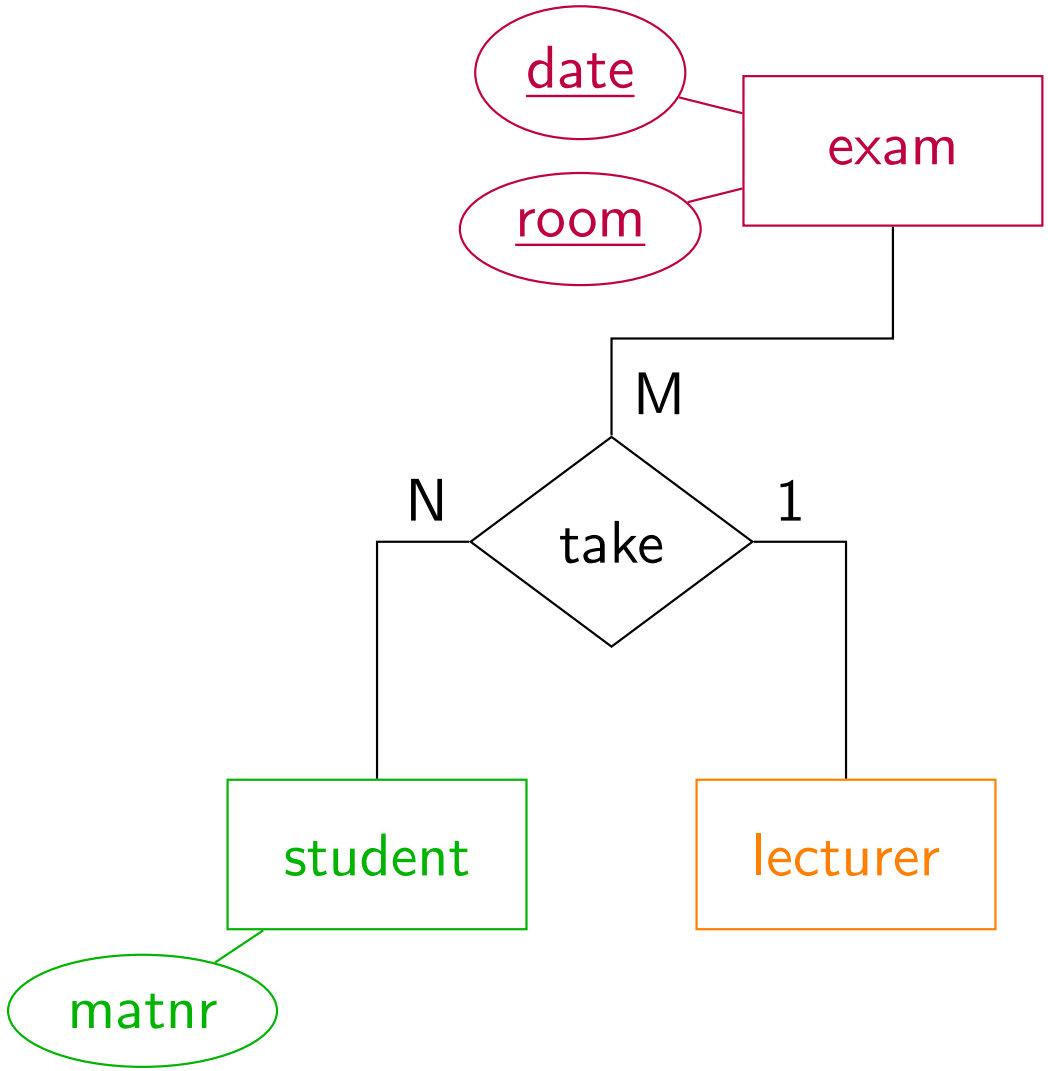




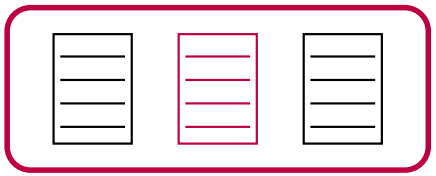
Conceptual Design Through ER Diagrams

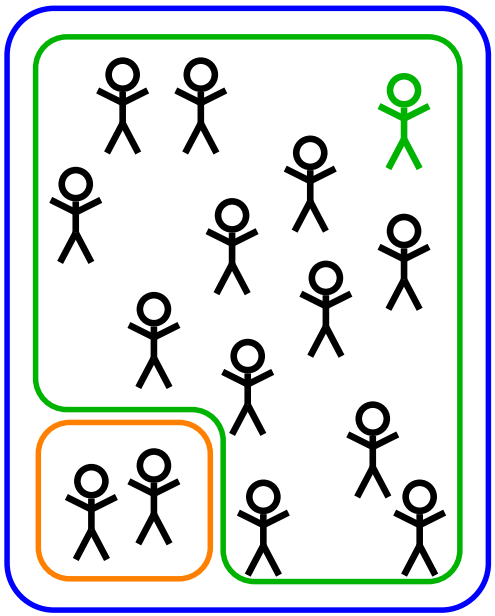


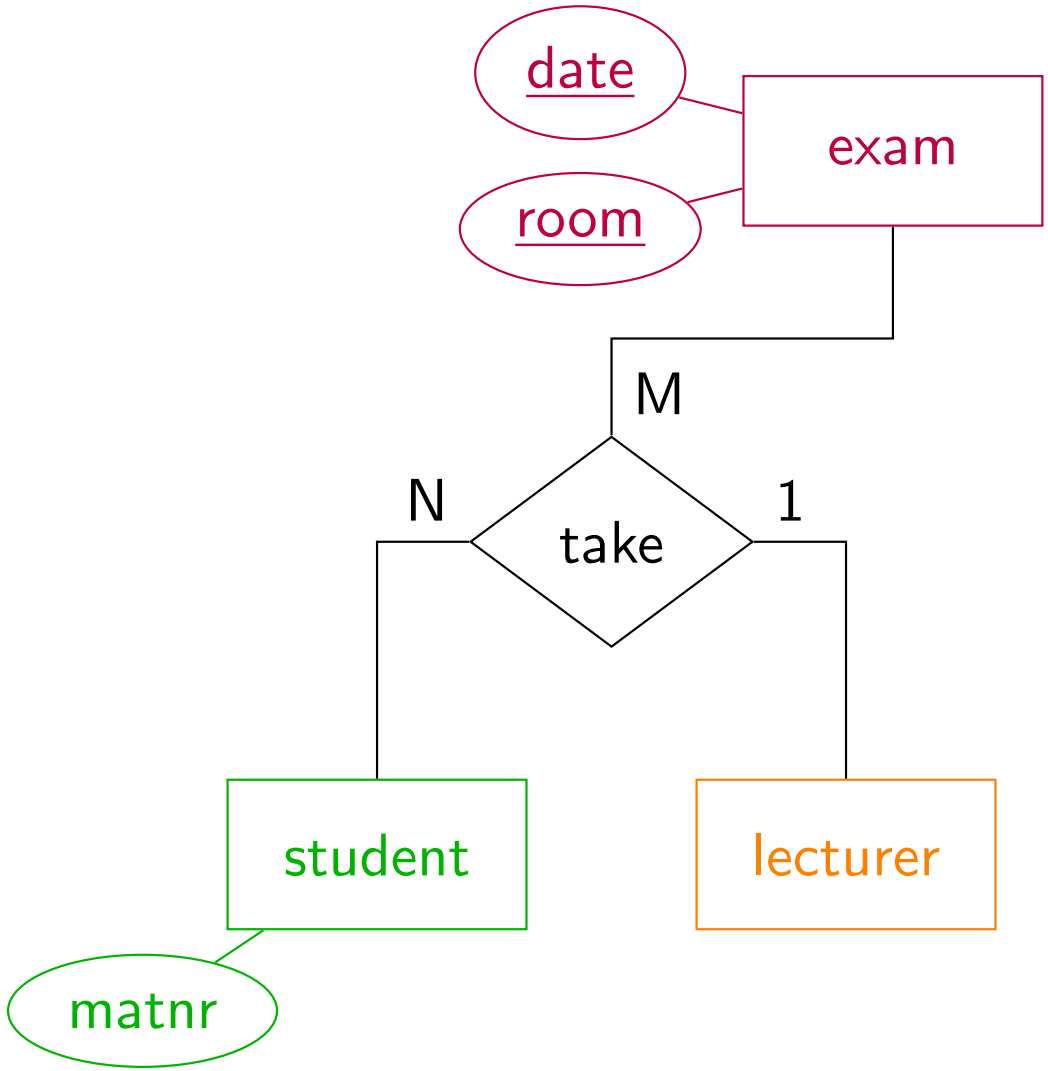




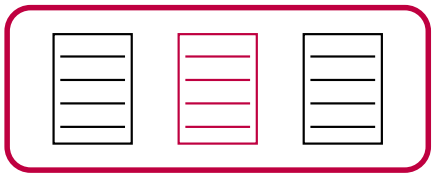
Conceptual Design Through ER Diagrams

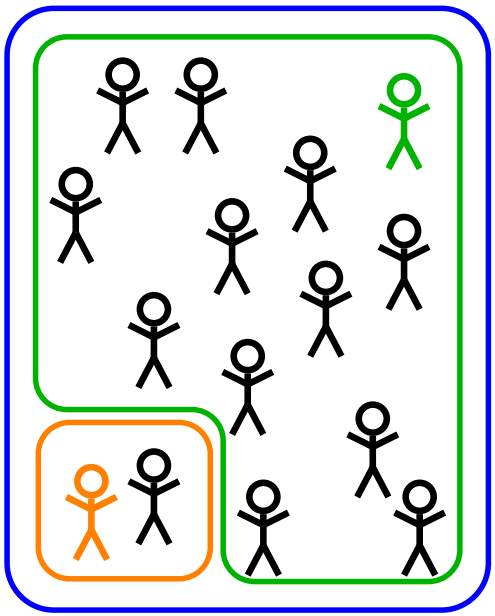


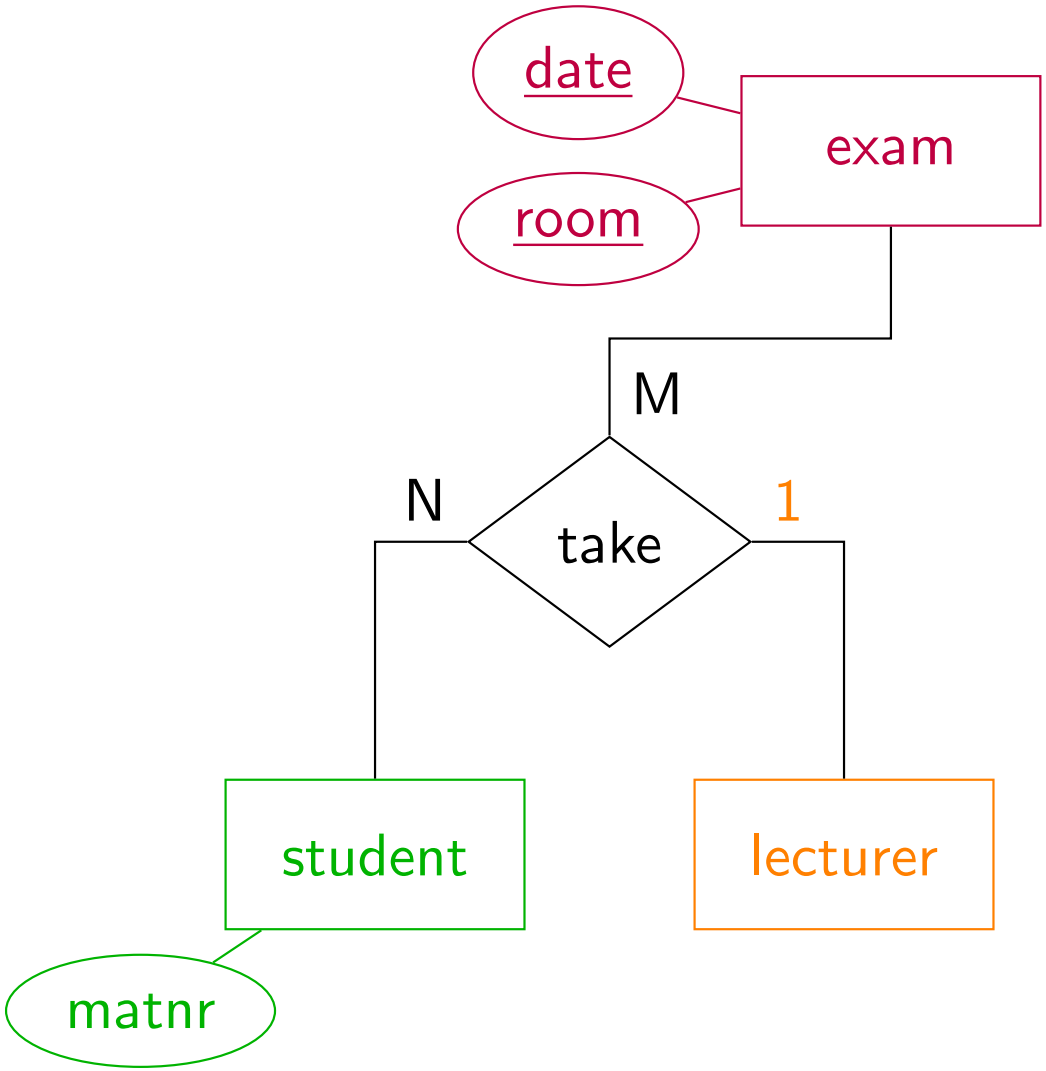




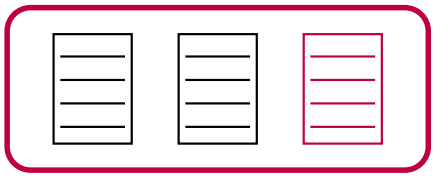
Conceptual Design Through ER Diagrams

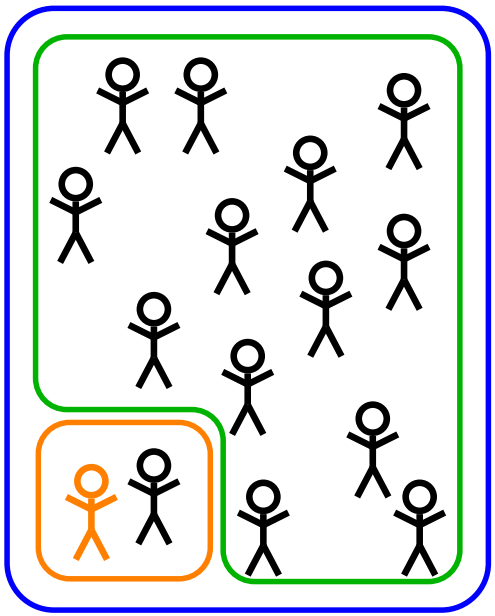


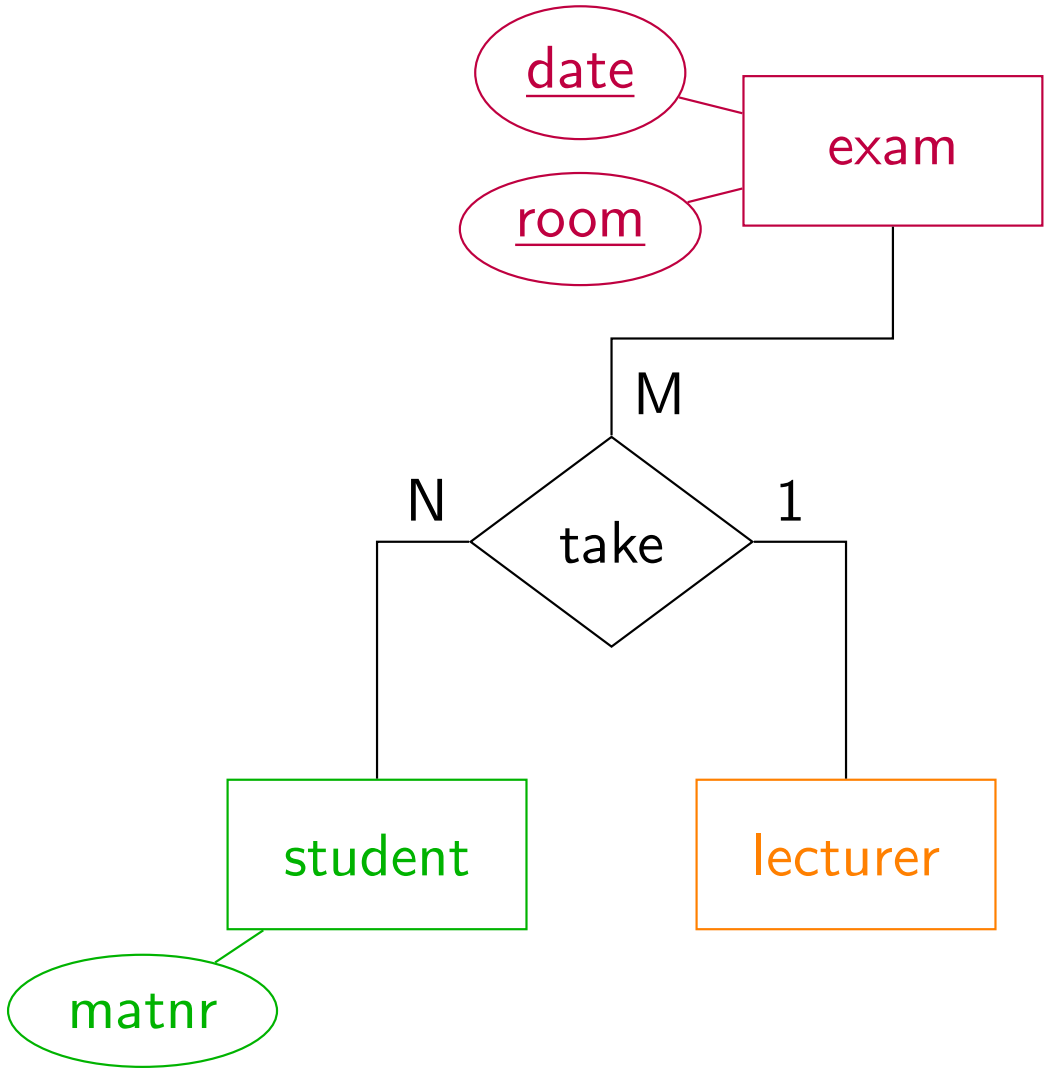




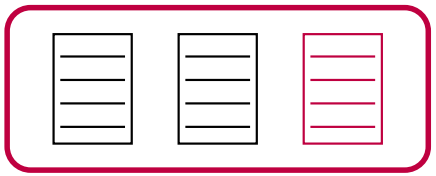
Conceptual Design Through ER Diagrams

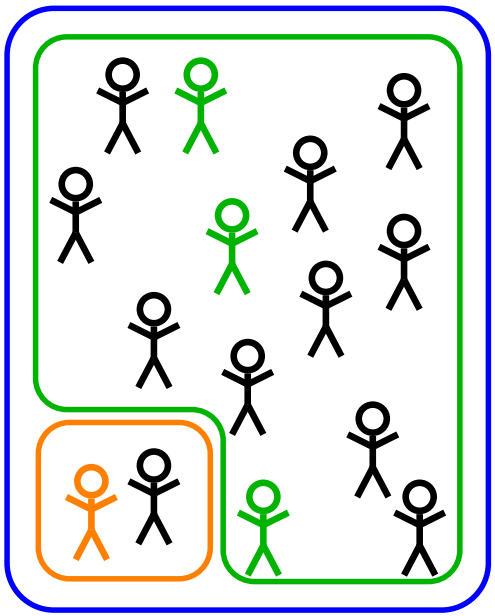


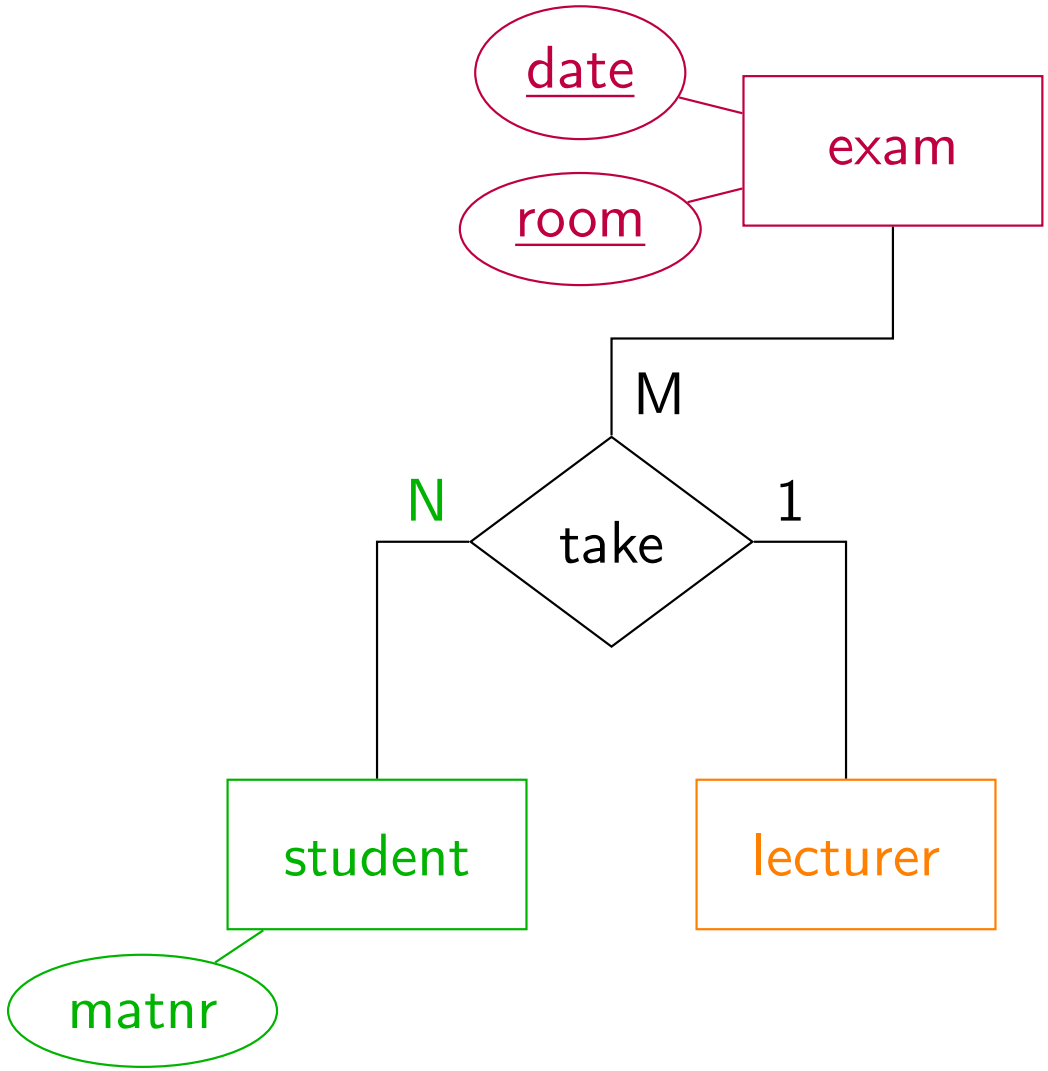




Conceptual Design Through ER Diagrams







Entities and Relationships

entities: clearly distinguishable concepts of the world to be modelled

relationships: link several entities

Example

entities:
- student Müller, Maier, Alice, Bob
- lectures Data Modelling, Data Base Systems
- exams DM 2.2.2016, DBS 12.12.2015

relationships:
- Müller attends Data Modelling, Maier attends Data Base Systems
- Alice takes exam DBS 12.12.2015

Entity Types and Relationship Types

entities and relationships are abstracted to homogeneous types

→ describe types and their relationships in the ER diagram

Entity Types and Relationship Types

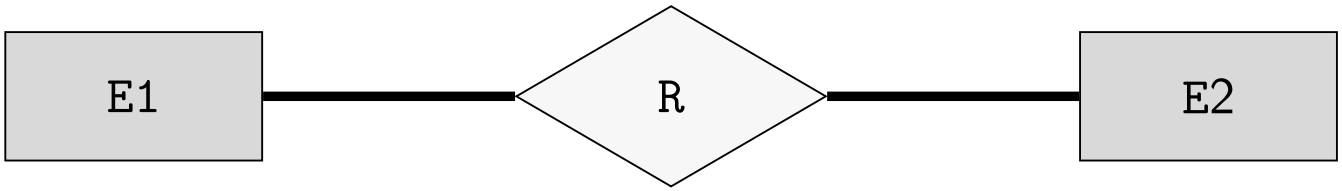
entities and relationships are abstracted to homogeneous types

→ describe types and their relationships in the ER diagram
entity types: rectangles

Entity Types and Relationship Types

entities and relationships are abstracted to homogeneous types

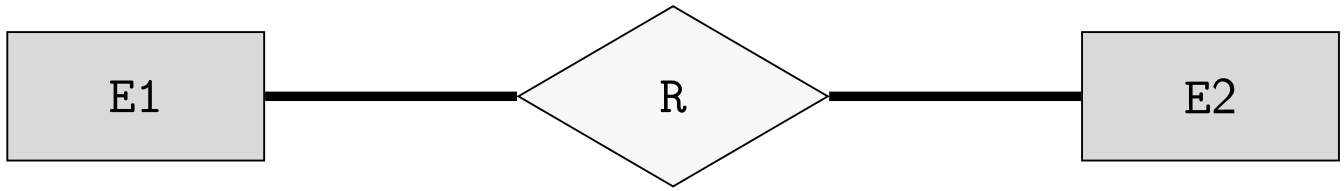
→ describe types and their relationships in the ER diagram
entity types: rectangles relationship types: diamonds



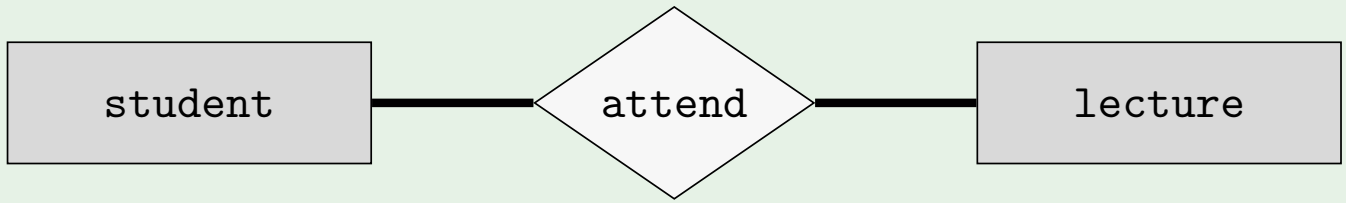
Entity Types and Relationship Types

entities and relationships are abstracted to homogeneous types

→ describe types and their relationships in the ER diagram
entity types: rectangles relationship types: diamonds

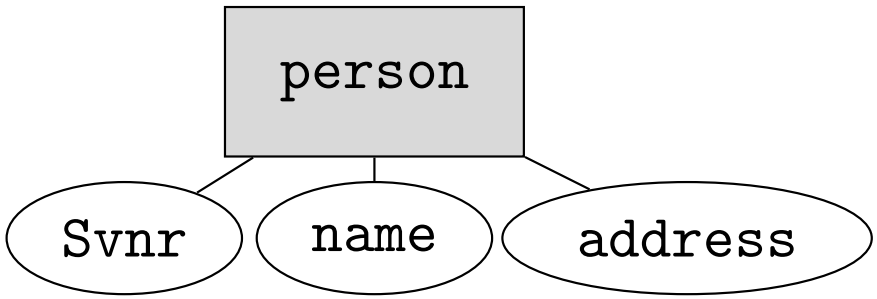


Example



Attributes and Keys

attributes characterize entity and relationship types

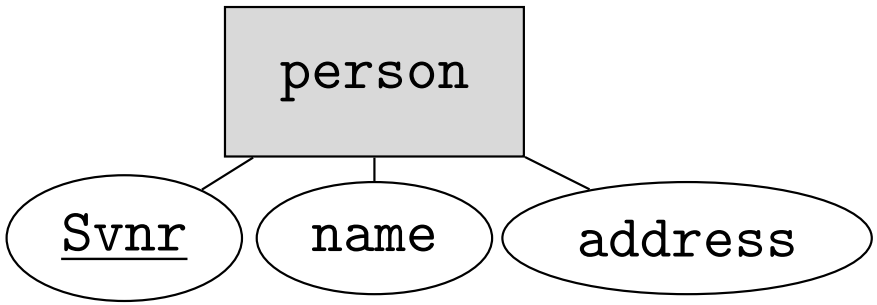


Attributes and Keys

attributes characterize entity and relationship types

keys are a (subset-)minimal set of attributes, whose values identify a unique entity among the entities of a type

- in the ER diagram: a key per entity type
- attributes contained in the key are underlined
- in case of several candidates: pick one key

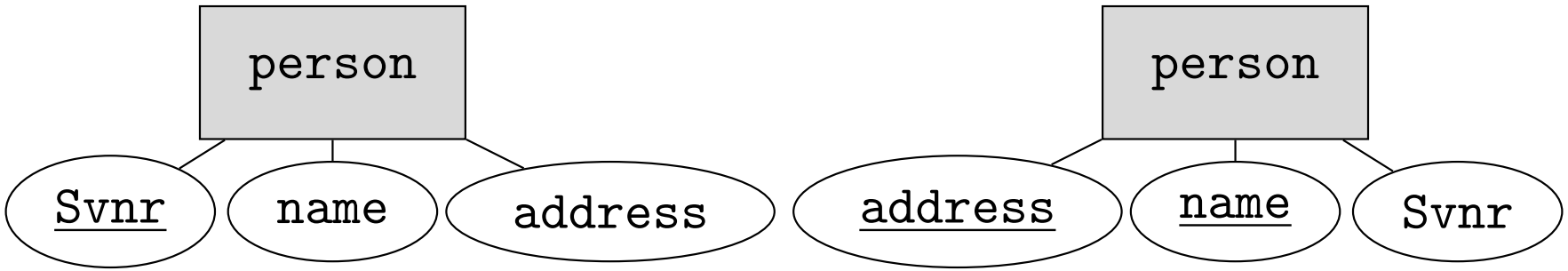


Attributes and Keys

attributes characterize entity and relationship types

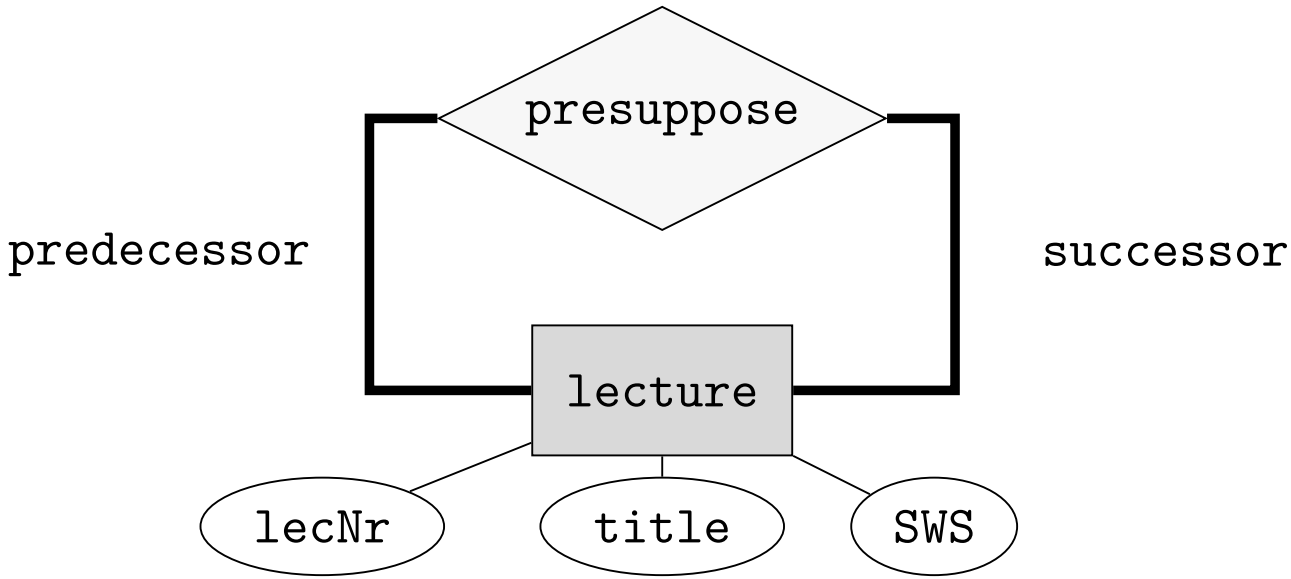
keys are a (subset-)minimal set of attributes, whose values identify a unique entity among the entities of a type

- in the ER diagram: a key per entity type
- attributes contained in the key are underlined
- in case of several candidates: pick one key

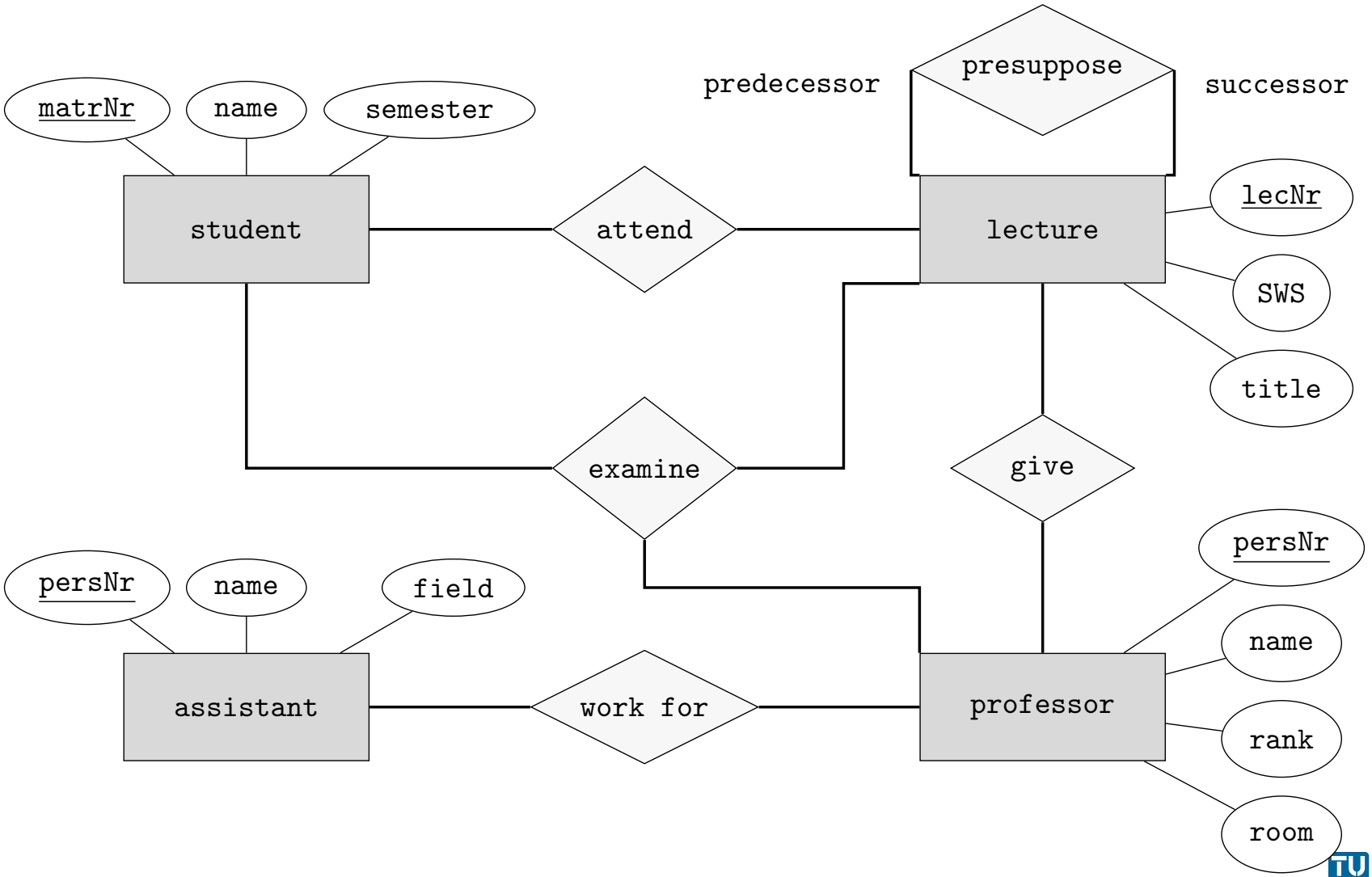


Roles

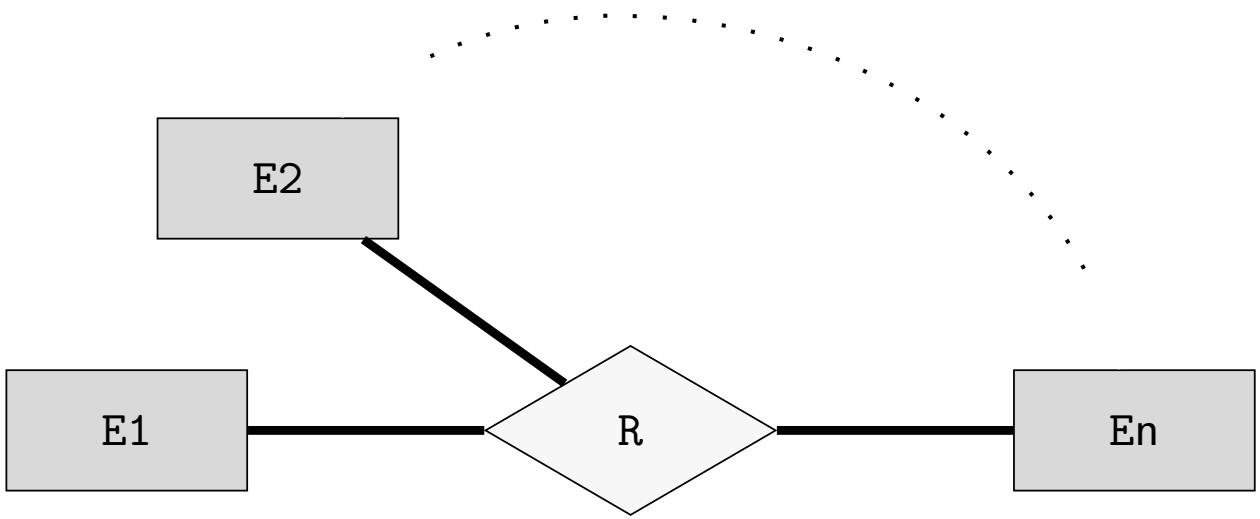
roles can be used to describe how entity types involved in a relationship behave



Example: University Schema

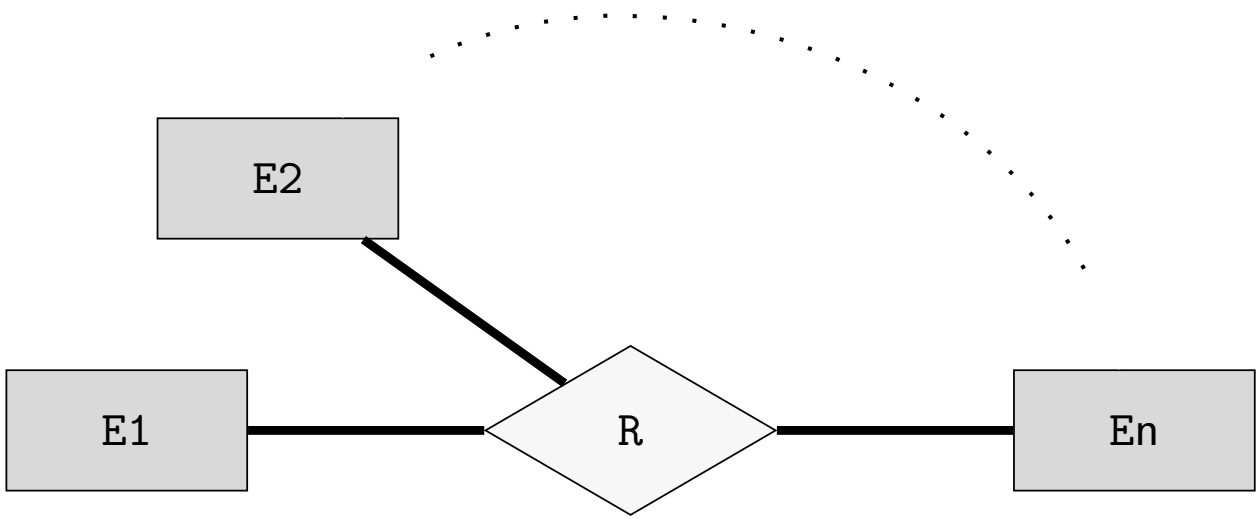


Relationships and Relationship Types – Formally



Relationships and Relationship Types – Formally

n-ary relationship: *n*-tuple of entities

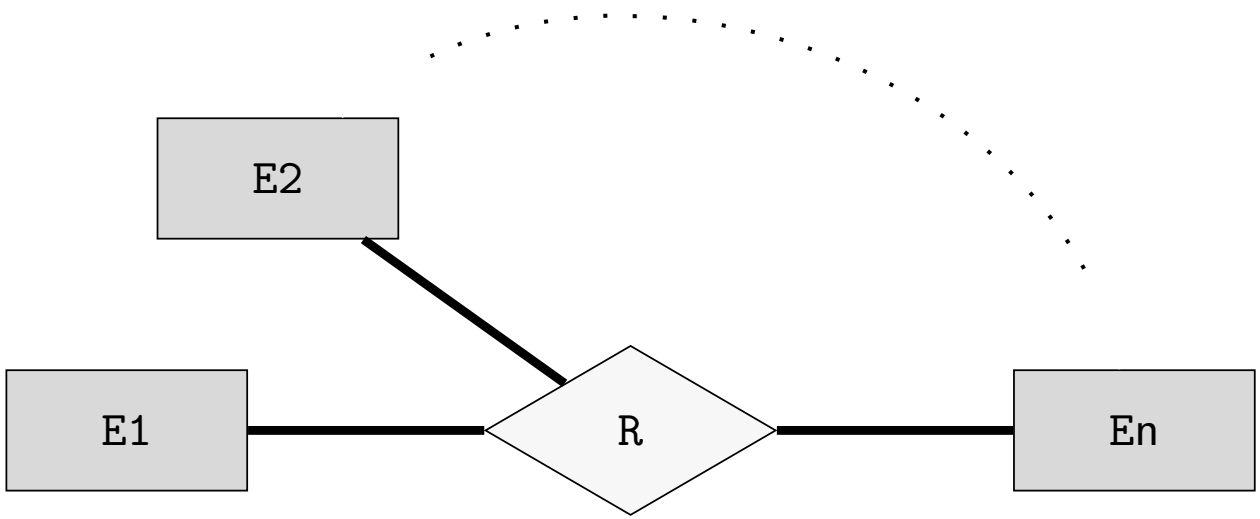


Relationships and Relationship Types – Formally

n-ary relationship: *n*-tuple of entities

n-ary relationship type: describes a set of *n*-ary relationships

$$R \subseteq E_1 \times E_2 \times \cdots \times E_n$$



Relationships and Relationship Types – Formally

Example

relationship:

- Alice attends Data Modelling
- Bob attends Data Base Systems
- Eve attends Data Modelling 2

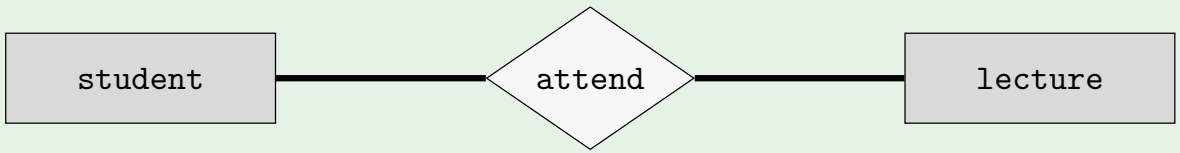
Relationships and Relationship Types – Formally

Example

relationship:

- Alice attends Data Modelling
- Bob attends Data Base Systems
- Eve attends Data Modelling 2

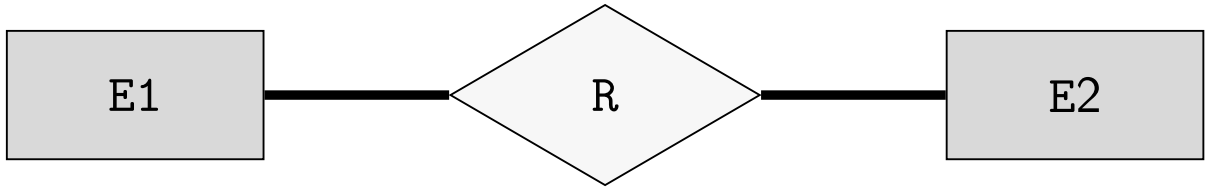
relationship type:



$$\text{attend} \subseteq \text{student} \times \text{lecture}$$

Cardinality of Binary Relationship Types

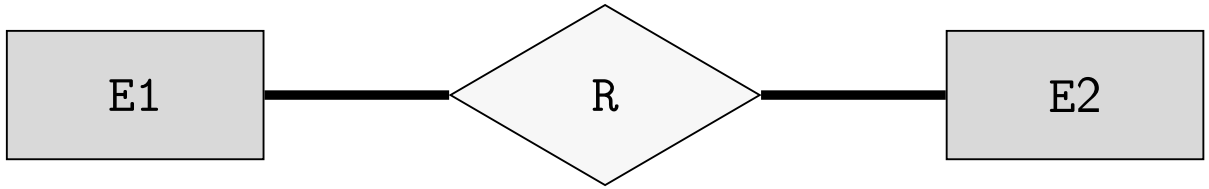
cardinalities describe the number of entities that can be in relationship with a specific entity



possible values:

Cardinality of Binary Relationship Types

cardinalities describe the number of entities that can be in relationship with a specific entity

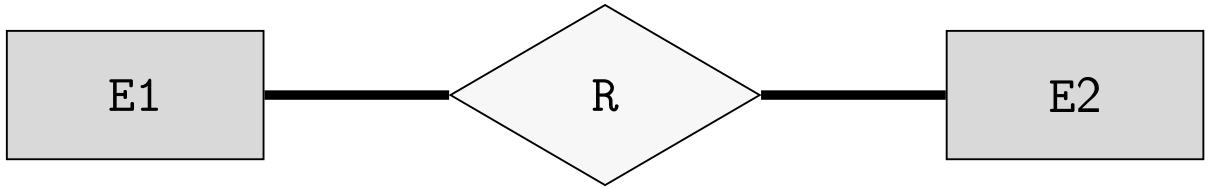


possible values:

1:1 each entity of type E1 with at most one entity of type E2 and vice versa

Cardinality of Binary Relationship Types

cardinalities describe the number of entities that can be in relationship with a specific entity

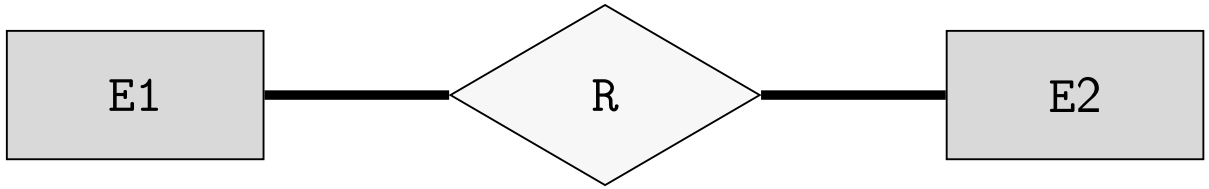


possible values:

- **1:1** each entity of type E1 with at most one entity of type E2 and vice versa
- **N:1** each entity of type E1 with at most one entity of type E2, no restrictions for E2

Cardinality of Binary Relationship Types

cardinalities describe the number of entities that can be in relationship with a specific entity

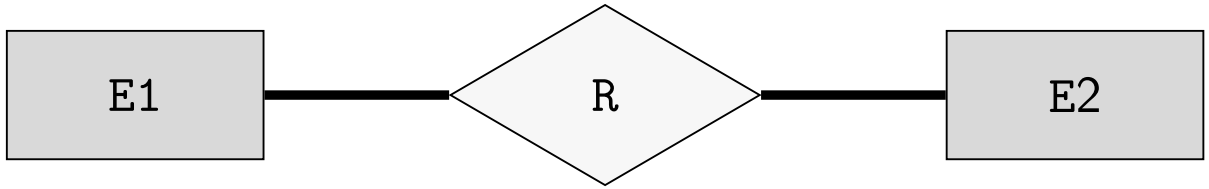


possible values:

- 1:1 each entity of type E1 with at most one entity of type E2 and vice versa
- N:1 each entity of type E1 with at most one entity of type E2, no restrictions for E2
- 1:N each entity of type E2 with at most one entity of type E1, no restriction for E1

Cardinality of Binary Relationship Types

cardinalities describe the number of entities that can be in relationship with a specific entity

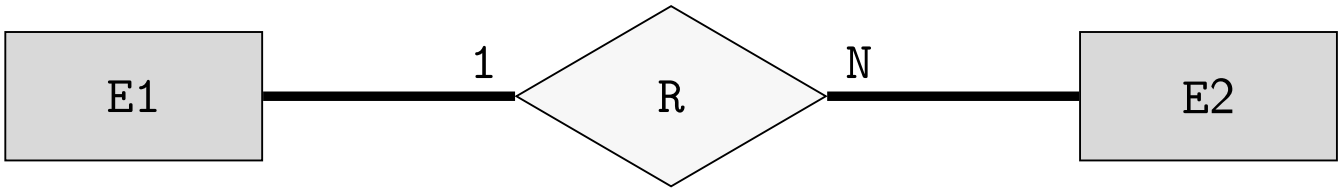


possible values:

- **1:1** each entity of type E1 with at most one entity of type E2 and vice versa
- **N:1** each entity of type E1 with at most one entity of type E2, no restrictions for E2
- **1:N** each entity of type E2 with at most one entity of type E1, no restriction for E1
- **N:M** no restrictions

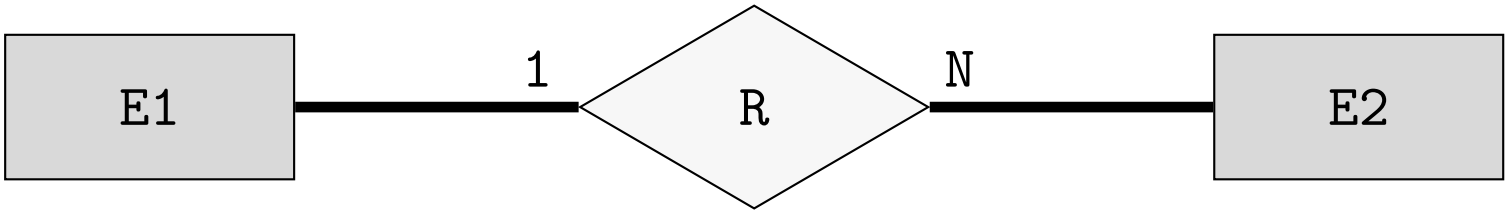
Cardinality of Binary Relationship Types

- notation: to each entity type we add (write) a value (1 or M,N,...) of the cardinality of the relationship type



- intuitive meaning: Fix the entity of a type. The value on the side of the other entity type determines the maximal number of entities of this type that can be in relationship with the fixed entity (M,N, ...: any number).

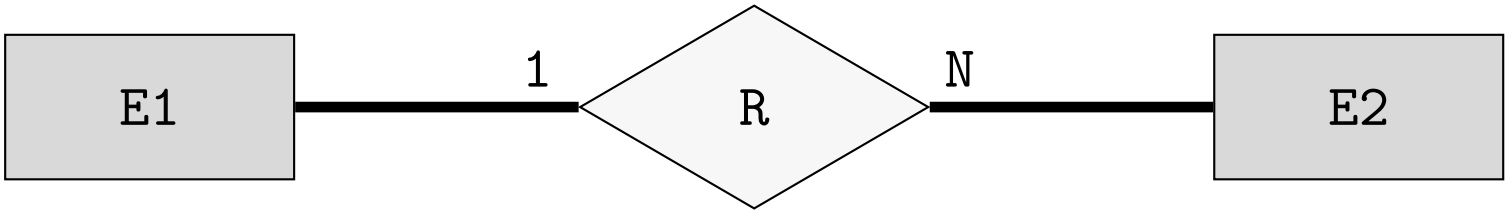
Cardinality of Binary Relationship Types



Example

- If we fix an entity of type E2, then it can be in relationship with at most one entity of type E1.

Cardinality of Binary Relationship Types

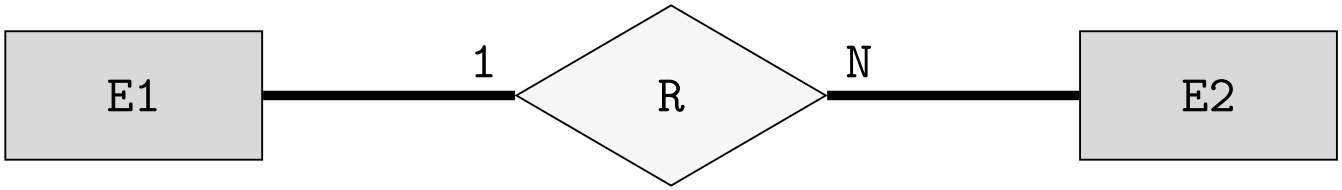


Example

- If we fix an entity of type E2, then it can be in relationship with at most one entity of type E1.
- If we fix an entity of type E1, then it can be in relationship with at most N (any number) entities of type E2.

Cardinalities – Formally

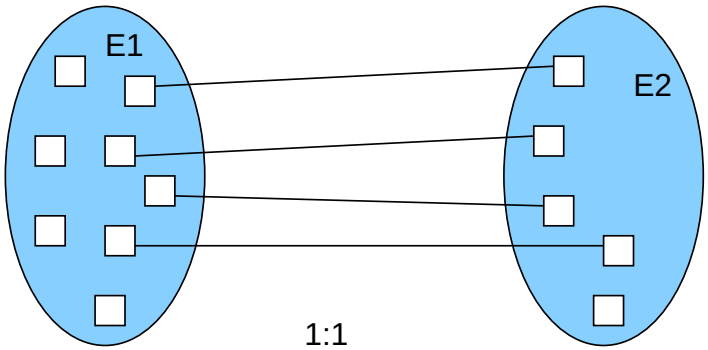
for cardinalities 1:1, N:1, and 1:N:
relationship type describes partial functions between entities



partial: not every entity has to be in relationship with another entity

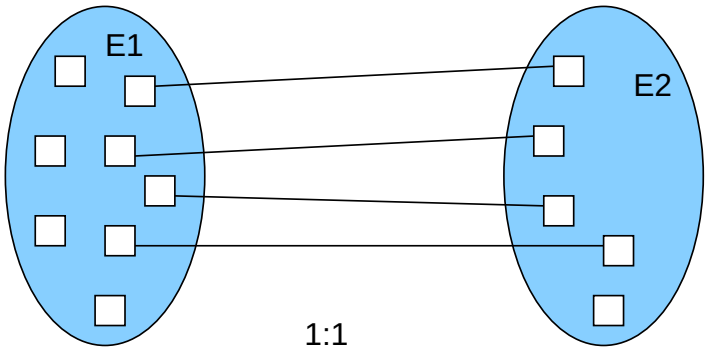
function: on the side "opposite of 1" every entity involved in the relationship is assigned to exactly one entity of the other type

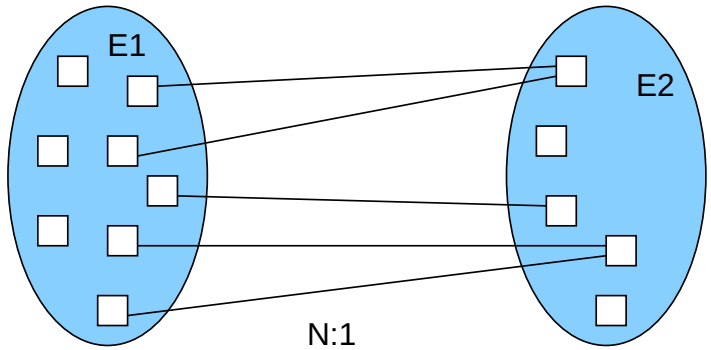
Partial Functions Defined by Cardinalities



$R : E1 \rightarrow E2$ resp. $R^{-1} : E2 \rightarrow E1$

Partial Functions Defined by Cardinalities

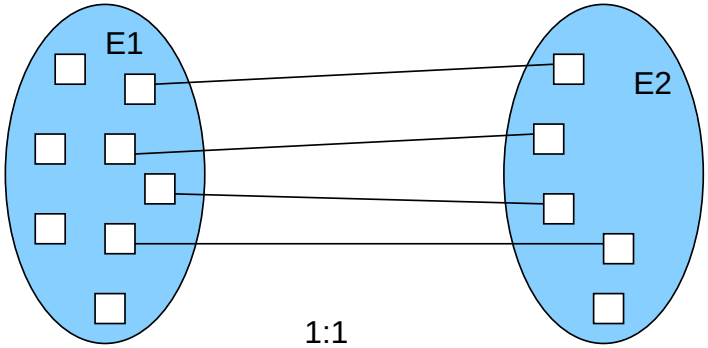


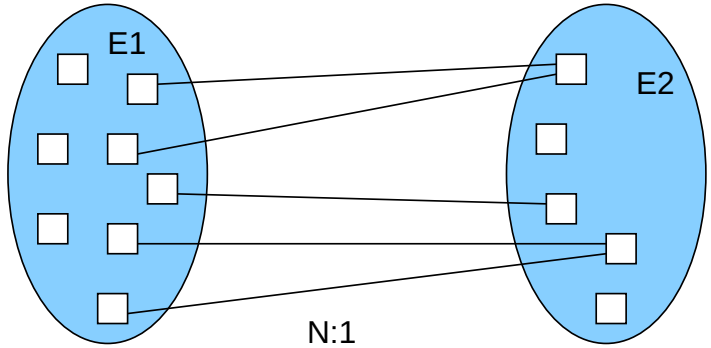


$R : E1 \rightarrow E2$ resp. $R^{-1} : E2 \rightarrow E1$

$R : E1 \rightarrow E2$

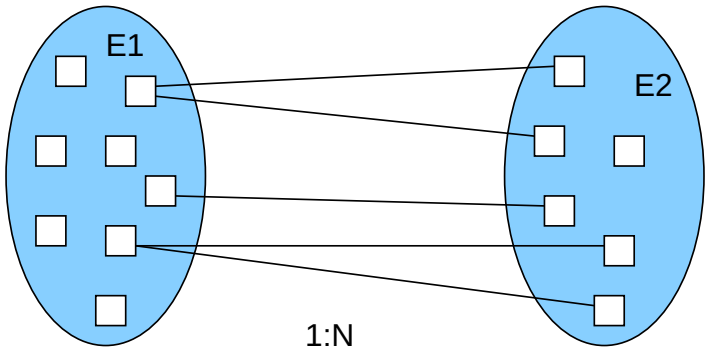
Partial Functions Defined by Cardinalities





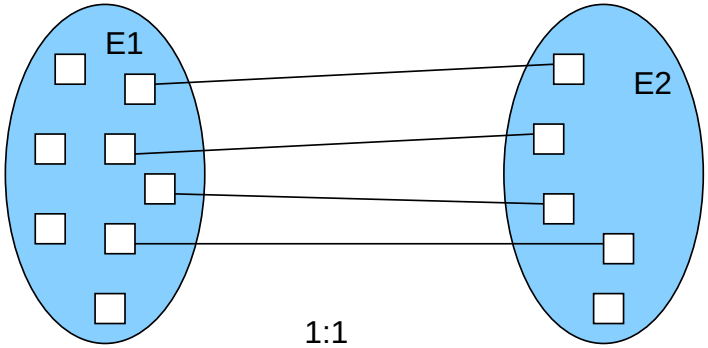
$R : E1 \rightarrow E2$ resp. $R^{-1} : E2 \rightarrow E1$

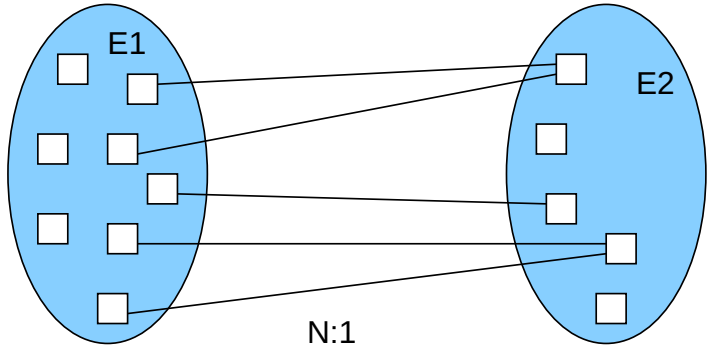
$R : E1 \rightarrow E2$



$R : E2 \rightarrow E1$

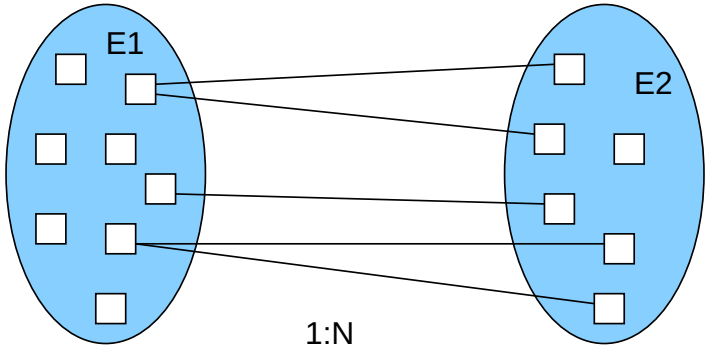
Partial Functions Defined by Cardinalities

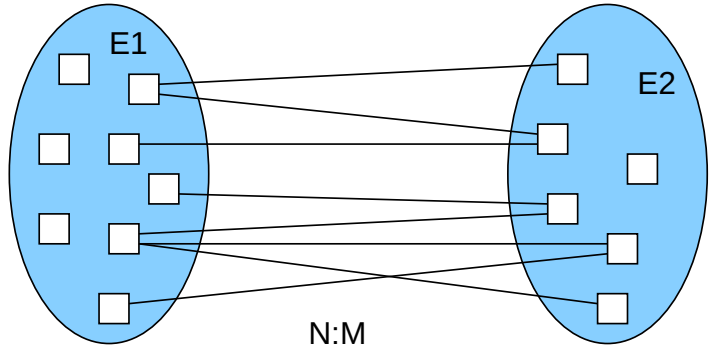




$R : E1 \rightarrow E2$ resp. $R^{-1} : E2 \rightarrow E1$

$R : E1 \rightarrow E2$





$R : E2 \rightarrow E1$

no partial function

Cardinalities for n-ary Relationship Types

generalization of binary relationship types

- intuitive meaning: if we fix an entity of all but one involved types, then the value on the side of the missing type determines the maximal number of entities of this type that can be in relationship with the $n-1$-tuple of entities.

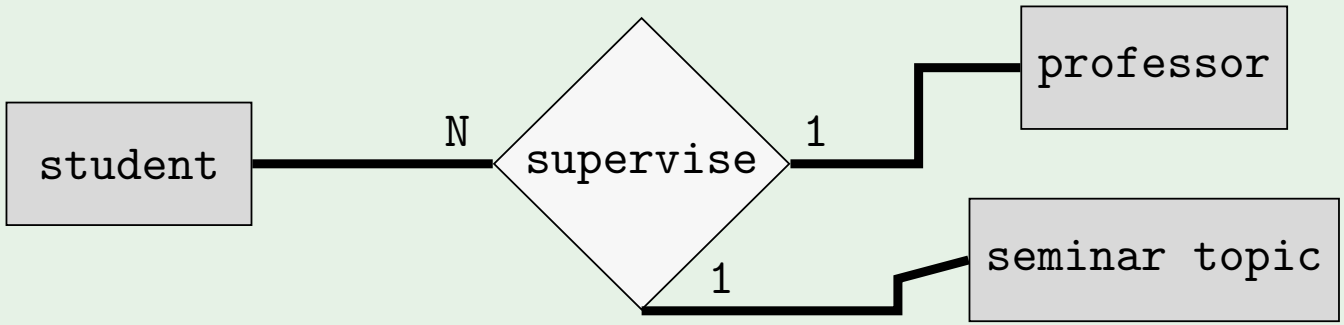
- formally: Let R be a relationship between several entities $E_1, \ldots, E_n$, where the cardinality of the entity $E_k, 1 \leq k \leq n$ is specified with "1", then R determines the following partial function:

$$R : E_1 \times E_2 \times \cdots \times E_{k-1} \times E_{k+1} \times \cdots \times E_n \rightarrow E_k$$

Cardinalities for n-ary Relationship Types

Example

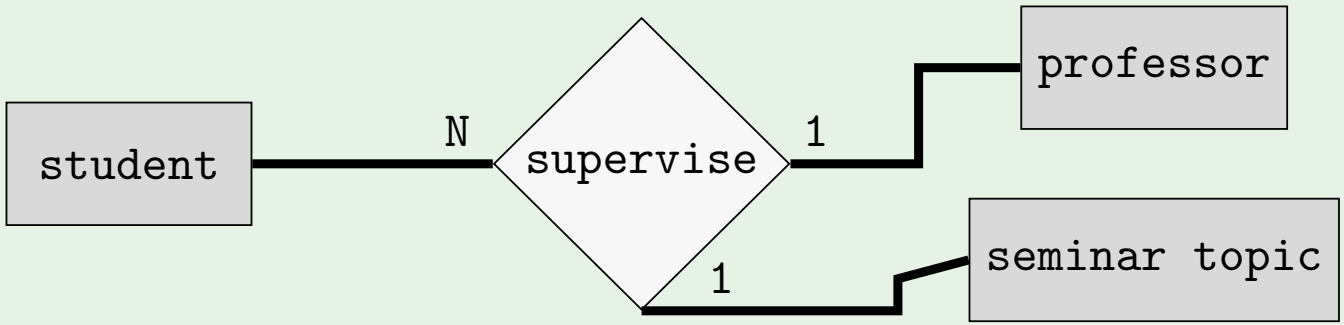
Consider the relationship type `supervise` between the entity types `student`, `professor`, `seminar topic`.



Cardinalities for n-ary Relationship Types

Example

Consider the relationship type `supervise` between the entity types `student`, `professor`, `seminar topic`.

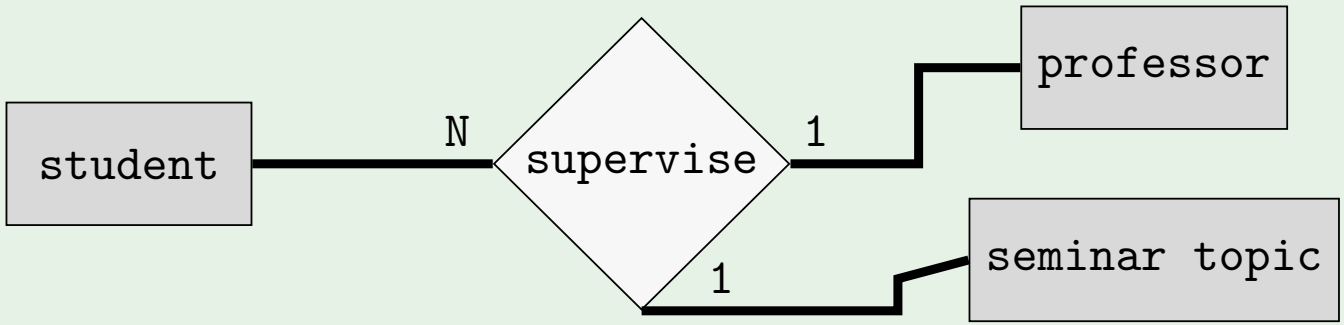


supervise: professor $\times$ student $\rightarrow$ seminar topic

Cardinalities for n-ary Relationship Types

Example

Consider the relationship type `supervise` between the entity types `student`, `professor`, `seminar topic`.



supervise: professor $\times$ student $\rightarrow$ seminar topic

supervise: seminar topic $\times$ student $\rightarrow$ professor

Cardinalities for n-ary Relationship Types

Example (ctd.)

supervise: professor $\times$ student $\rightarrow$ seminar topic

Cardinalities for n-ary Relationship Types

Example (ctd.)

supervise: professor $\times$ student $\rightarrow$ seminar topic

this means: students are allowed to work on only one seminar topic per supervisor (professor)

Cardinalities for n-ary Relationship Types

Example (ctd.)

supervise: professor $\times$ student $\rightarrow$ seminar topic
this means: students are allowed to work on only one seminar topic per supervisor (professor)

supervise: seminar topic $\times$ student $\rightarrow$ professor

Cardinalities for n-ary Relationship Types

Example (ctd.)

supervise: professor $\times$ student $\rightarrow$ seminar topic
this means: students are allowed to work on only one seminar topic per supervisor (professor)

supervise: seminar topic $\times$ student $\rightarrow$ professor
this means: students are allowed to work on a seminar topic supervised by only one professor (= no reuse of topics)

Cardinalities for n-ary Relationship Types

Example (ctd.)

supervise: professor $\times$ student $\rightarrow$ seminar topic
this means: students are allowed to work on only one seminar topic per supervisor (professor)

supervise: seminar topic $\times$ student $\rightarrow$ professor
this means: students are allowed to work on a seminar topic supervised by only one professor (= no reuse of topics)

still possible:

- professors are allowed to assign the same seminar topic to several students

Cardinalities for n-ary Relationship Types

Example (ctd.)

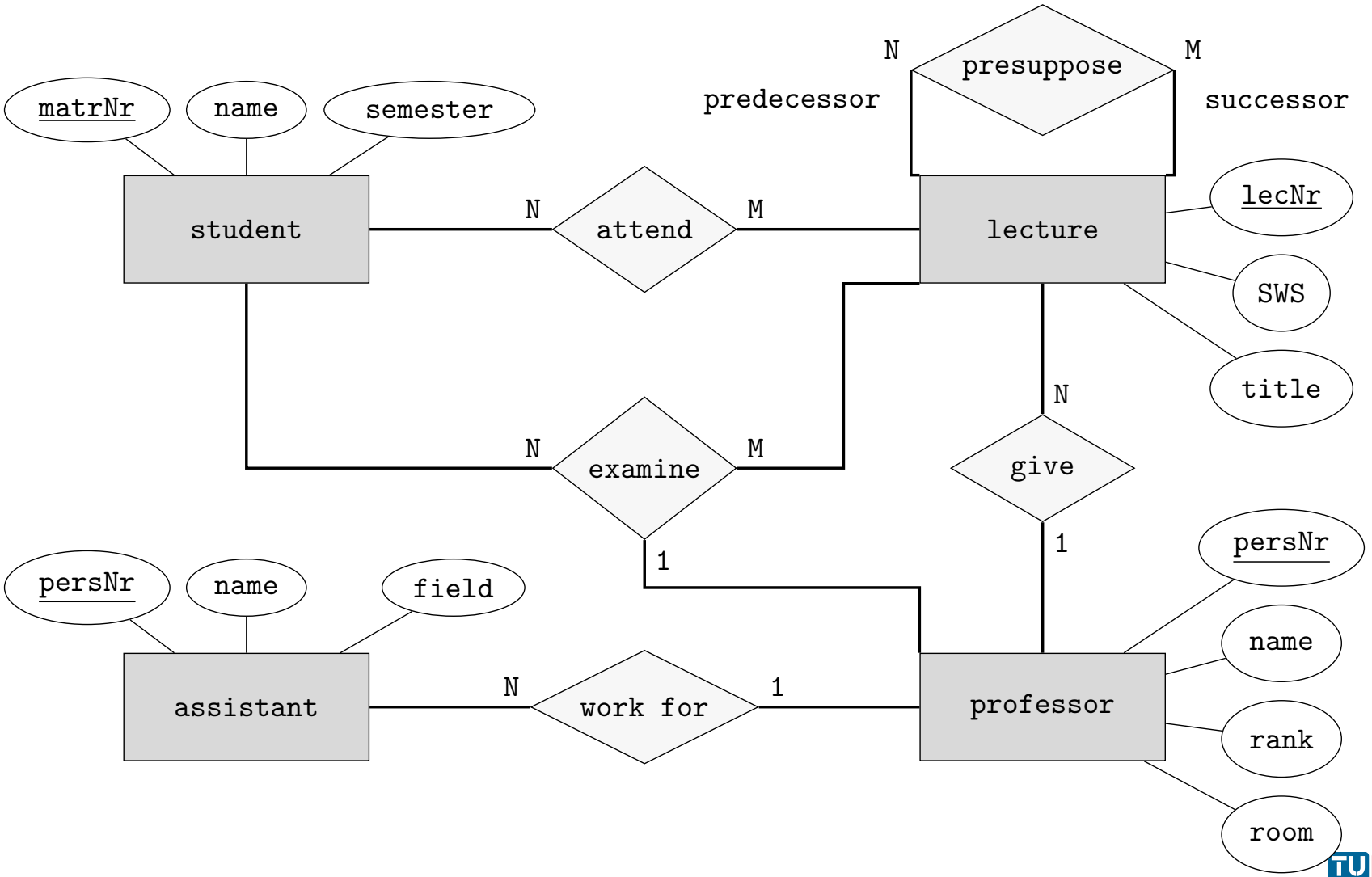
supervise: professor $\times$ student $\rightarrow$ seminar topic
this means: students are allowed to work on only one seminar topic per supervisor (professor)

supervise: seminar topic $\times$ student $\rightarrow$ professor
this means: students are allowed to work on a seminar topic supervised by only one professor (= no reuse of topics)

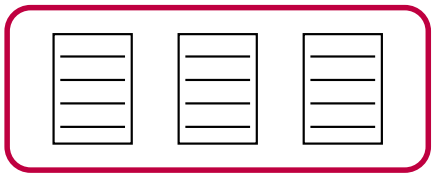
still possible:

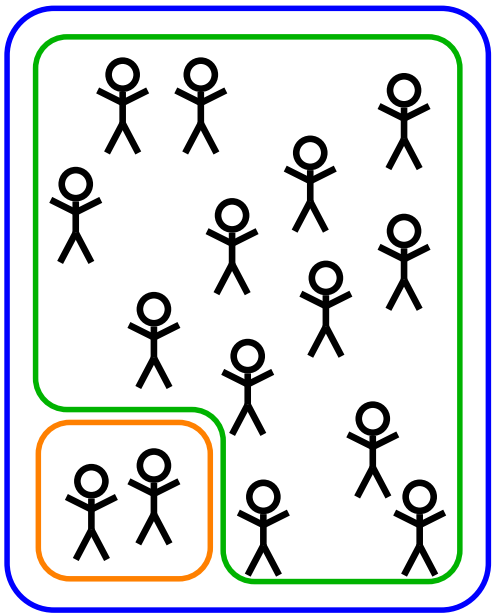
- professors are allowed to assign the same seminar topic to several students
- a seminar topic can be assigned by several professors - but only to different students

Example: University Schema with Cardinalities



Conceptual Design Through ER Diagrams





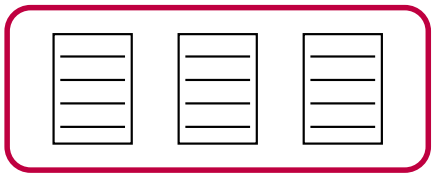
person

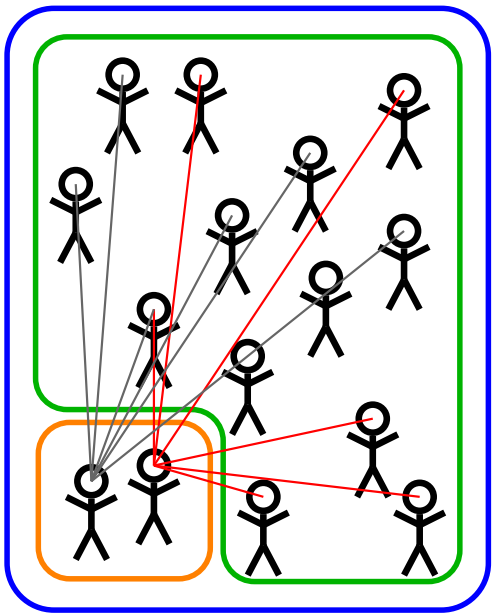
student

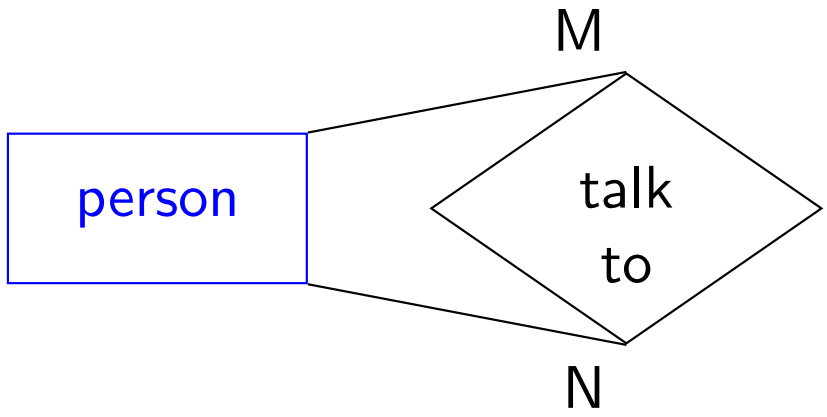
lecturer

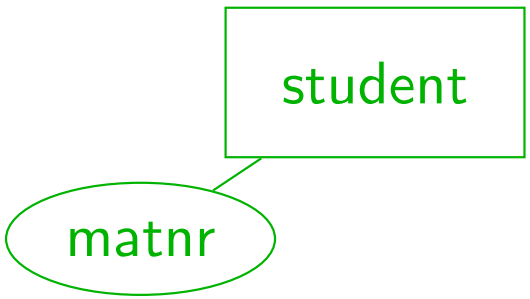
matnr

Conceptual Design Through ER Diagrams

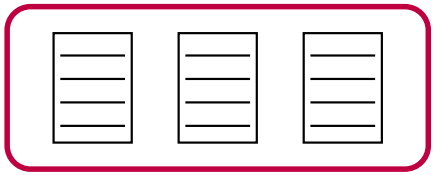


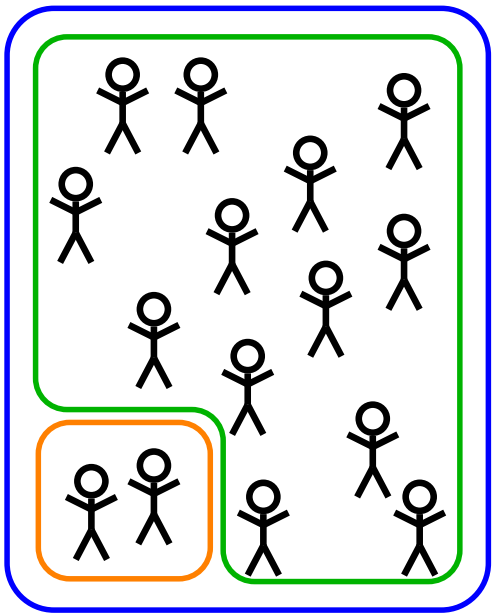


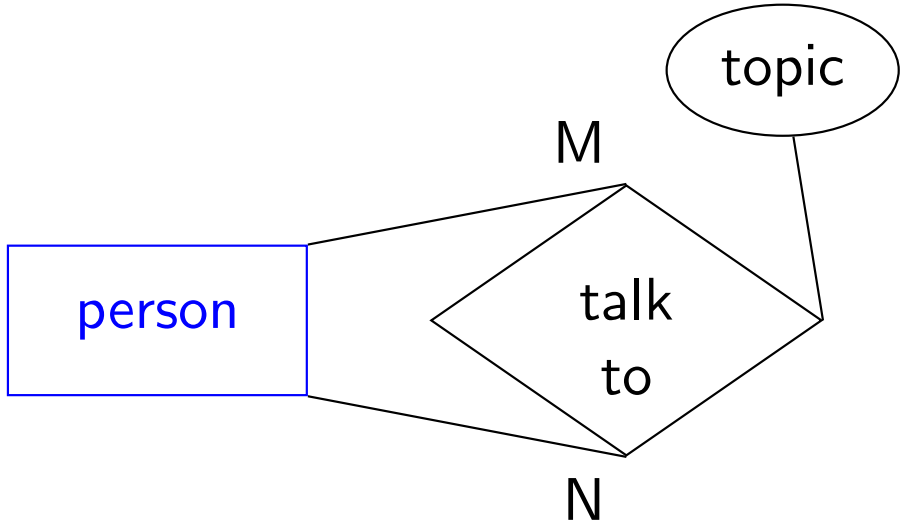


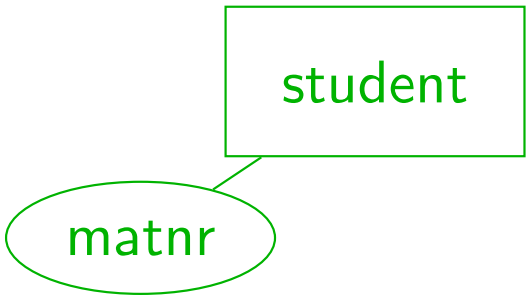


Conceptual Design Through ER Diagrams

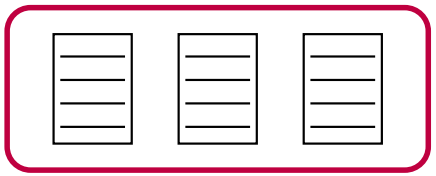


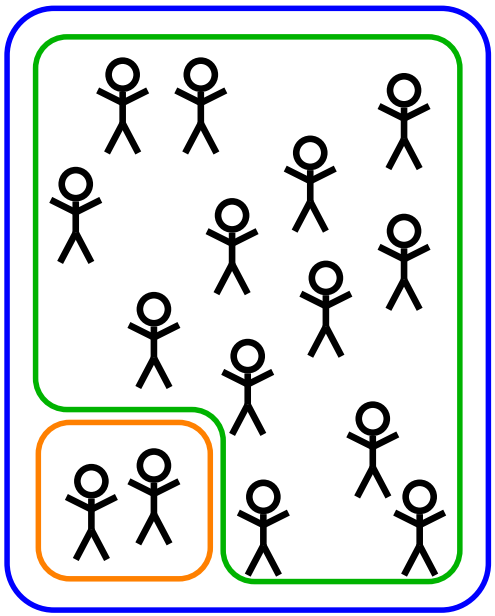


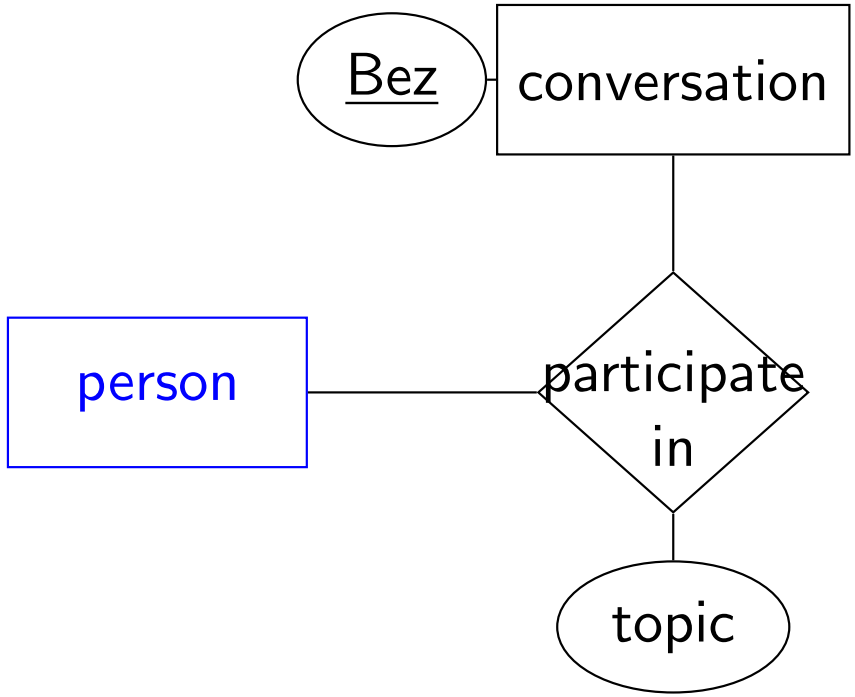


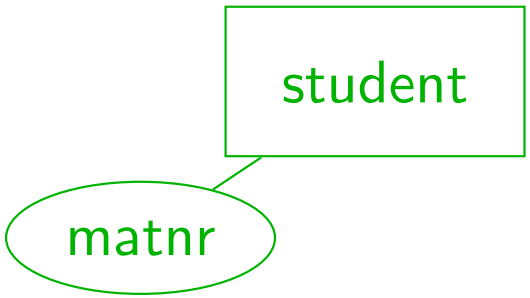


Conceptual Design Through ER Diagrams

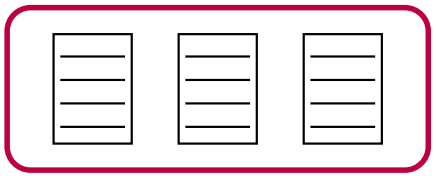


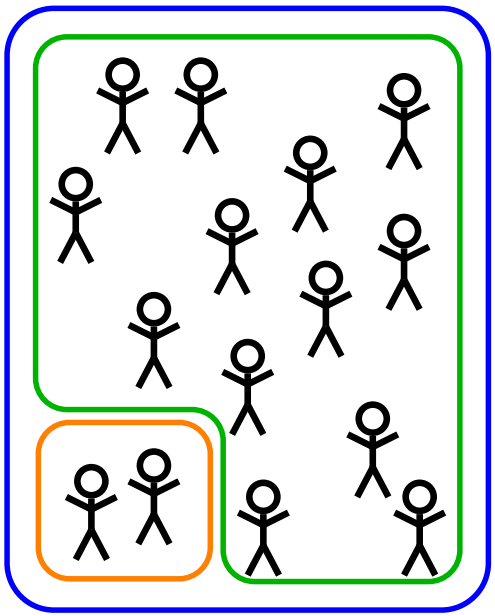


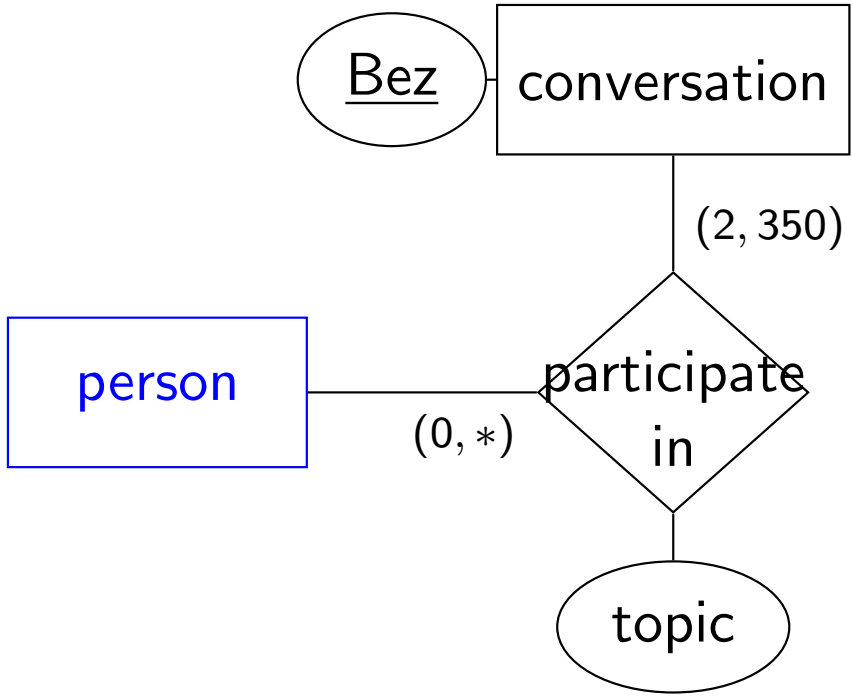


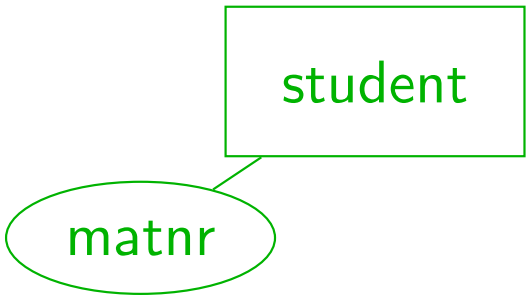


Conceptual Design Through ER Diagrams





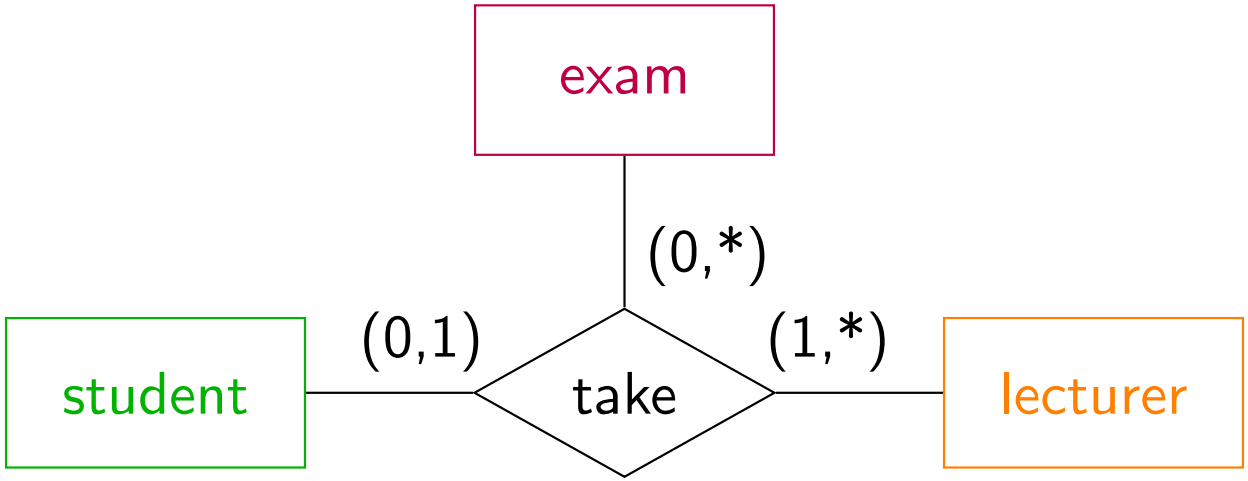




Example: The (min,max)-Notation

"all students take at most one exam"

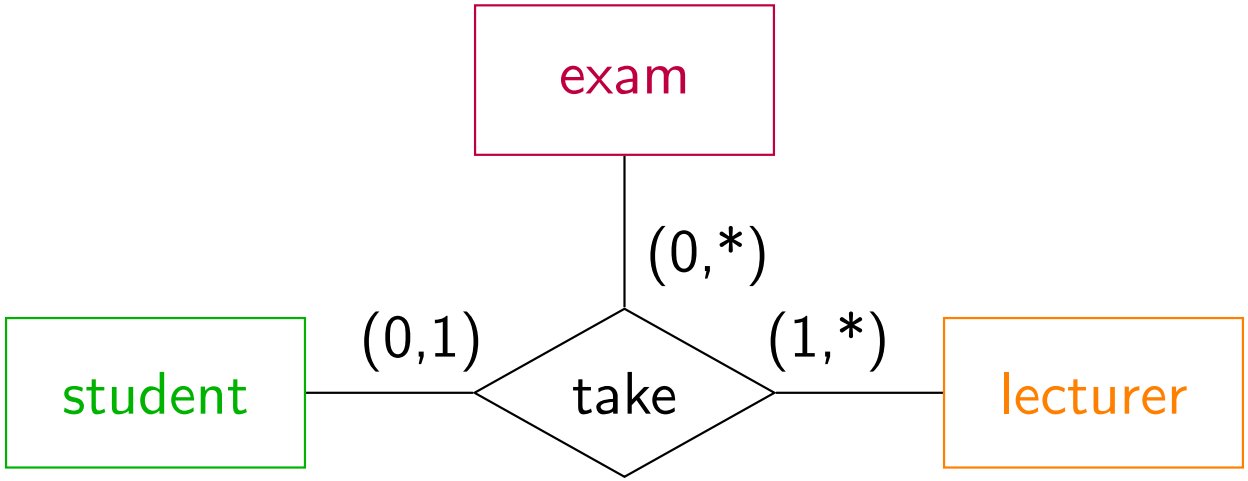
"all lecturers hold at least one exam"



Example: The (min,max)-Notation

"all students take at most one exam"

"all lecturers hold at least one exam"

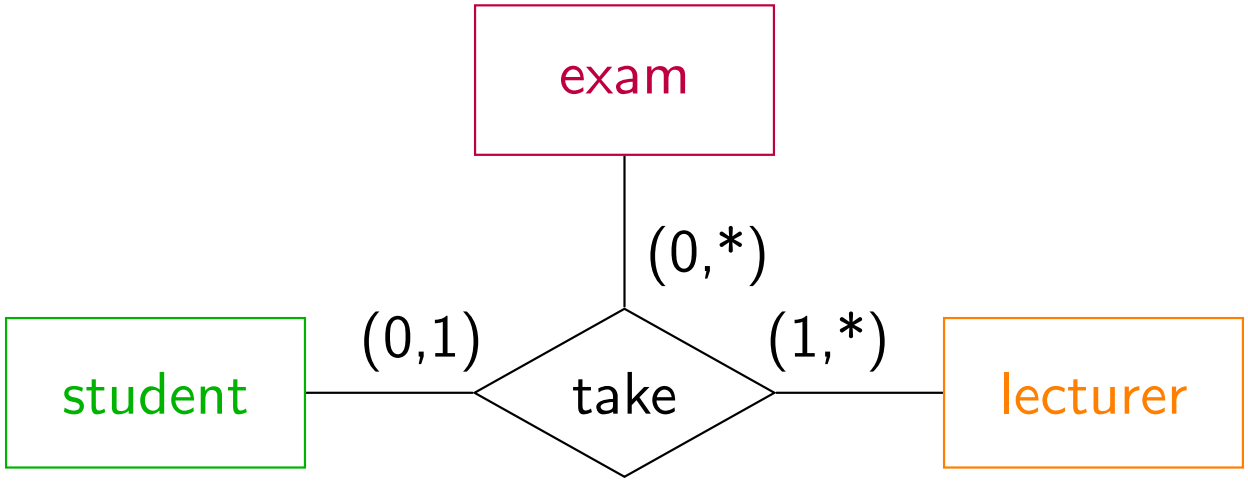


ok:

Example: The (min,max)-Notation

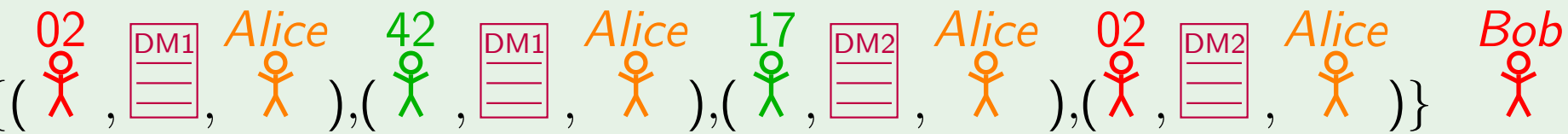
"all students take at most one exam"

"all lecturers hold at least one exam"

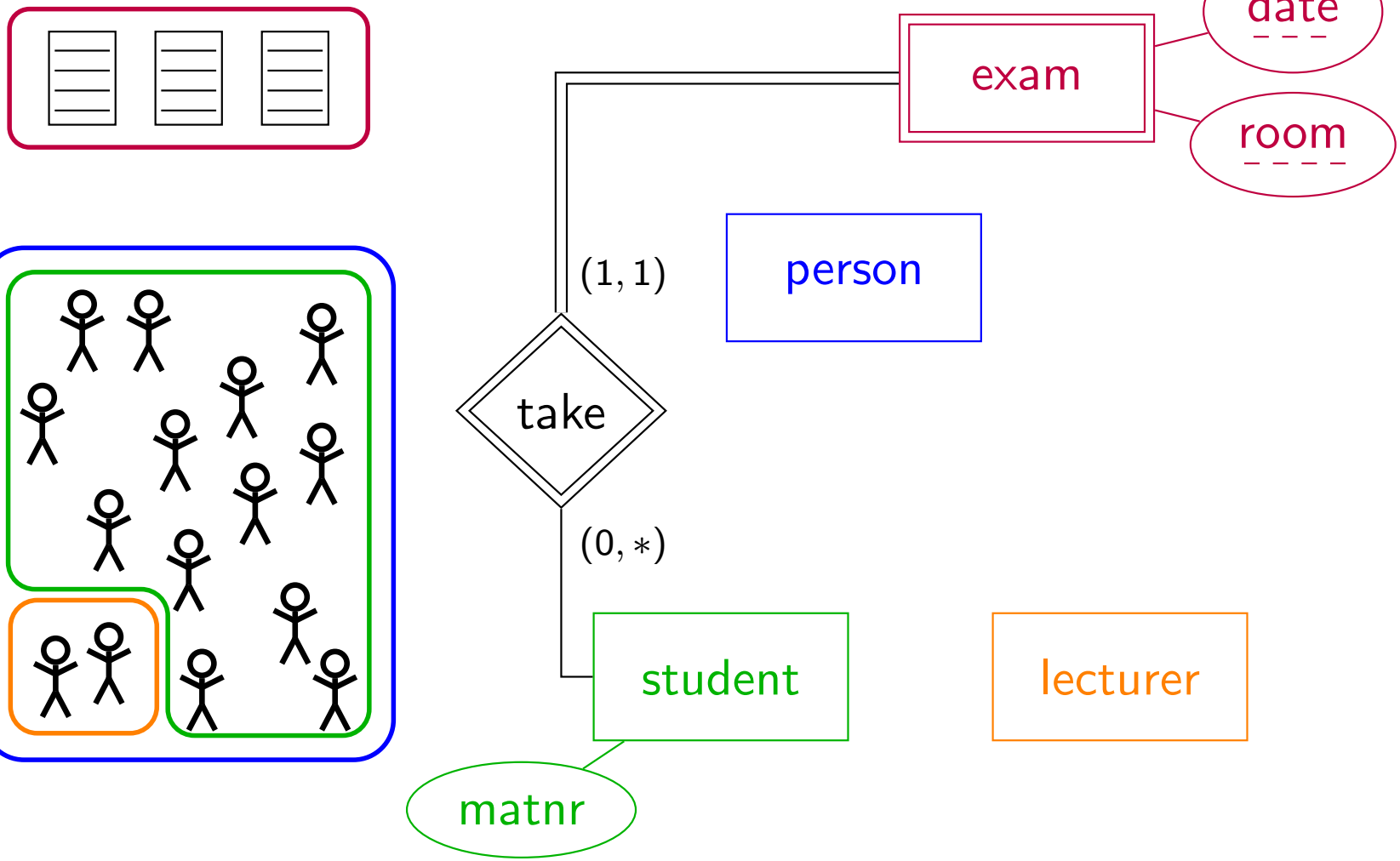


ok:

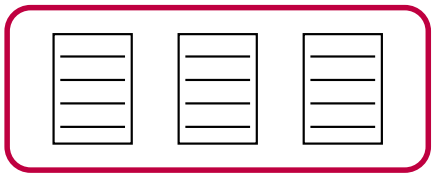
not ok:

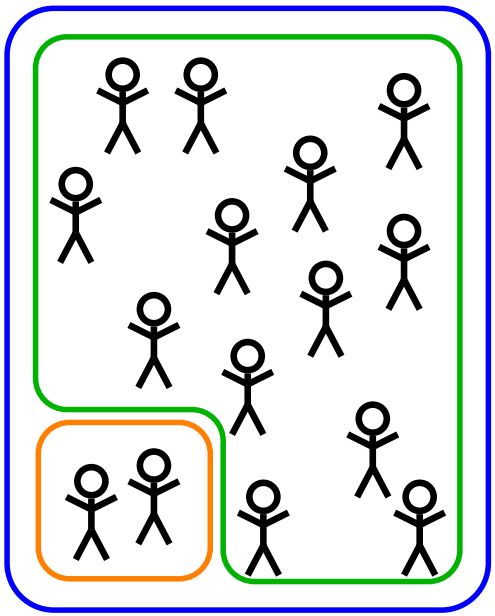


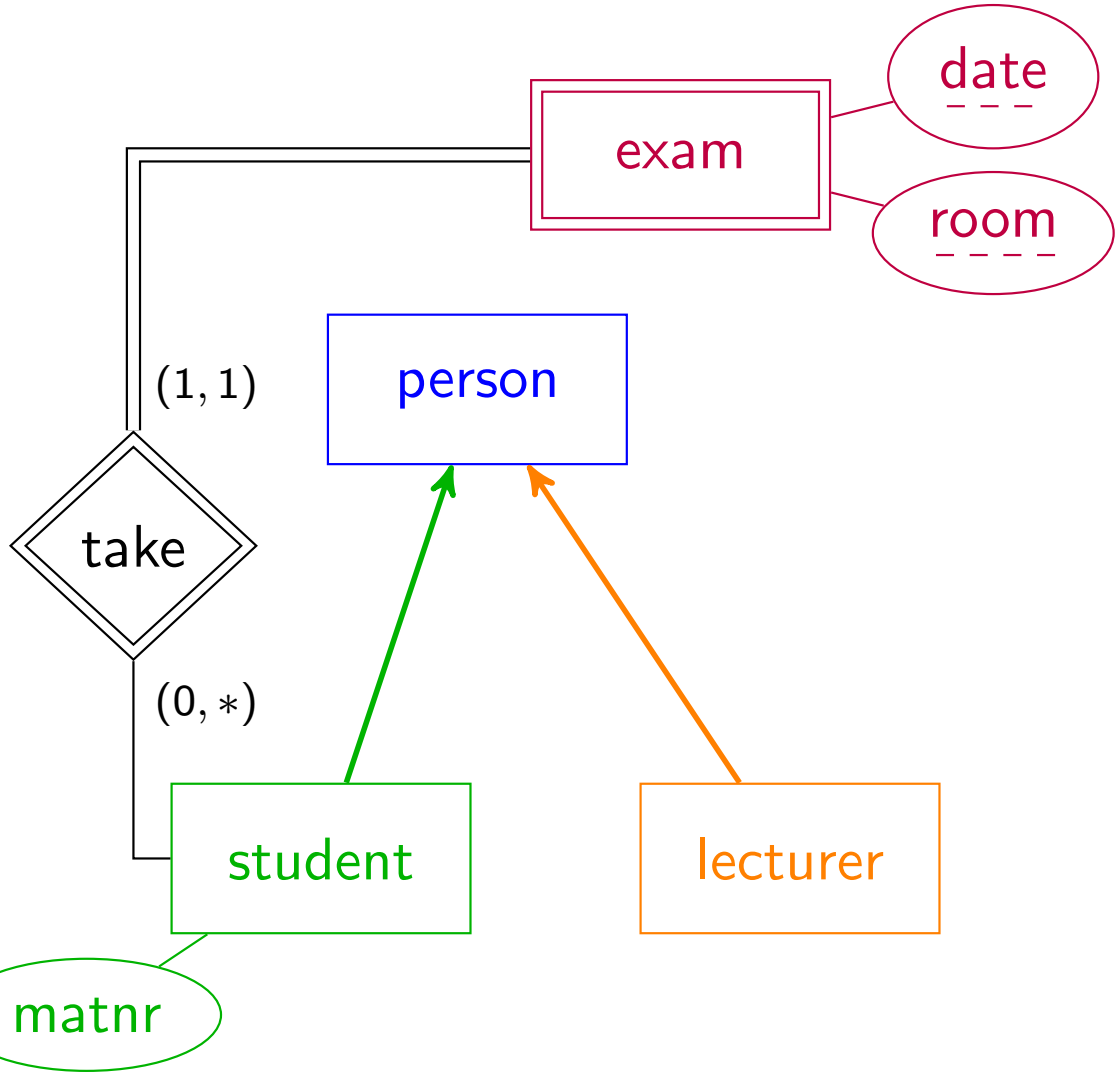
Conceptual Design Through EER Diagrams



Conceptual Design Through EER Diagrams







(min,max)-Notation

alternative notation of cardinalities and relationship types

given:
n-ary relationship type R between the entity types $E_1, \ldots E_n$

R defines the relationship $R \subseteq E_1 \times \cdots \times E_i, \times \cdots \times E_n$

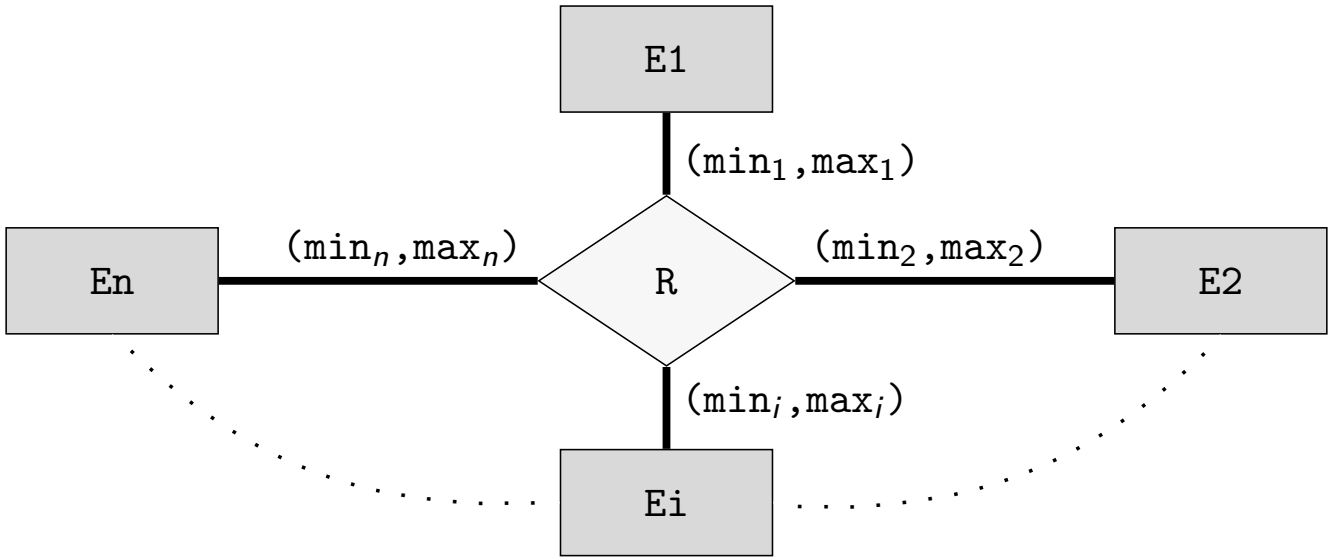
(min,max)-notation: determines for all entities of type E_i the minimal/maximal number of times they are allowed to occur in R

"1:N"-notation: determines for a specific $n-1$ tuple of entities the maximal number of entities of the remaining type it is allowed to be in relationship with

Expressive power is not comparable!

(min,max)-Notation

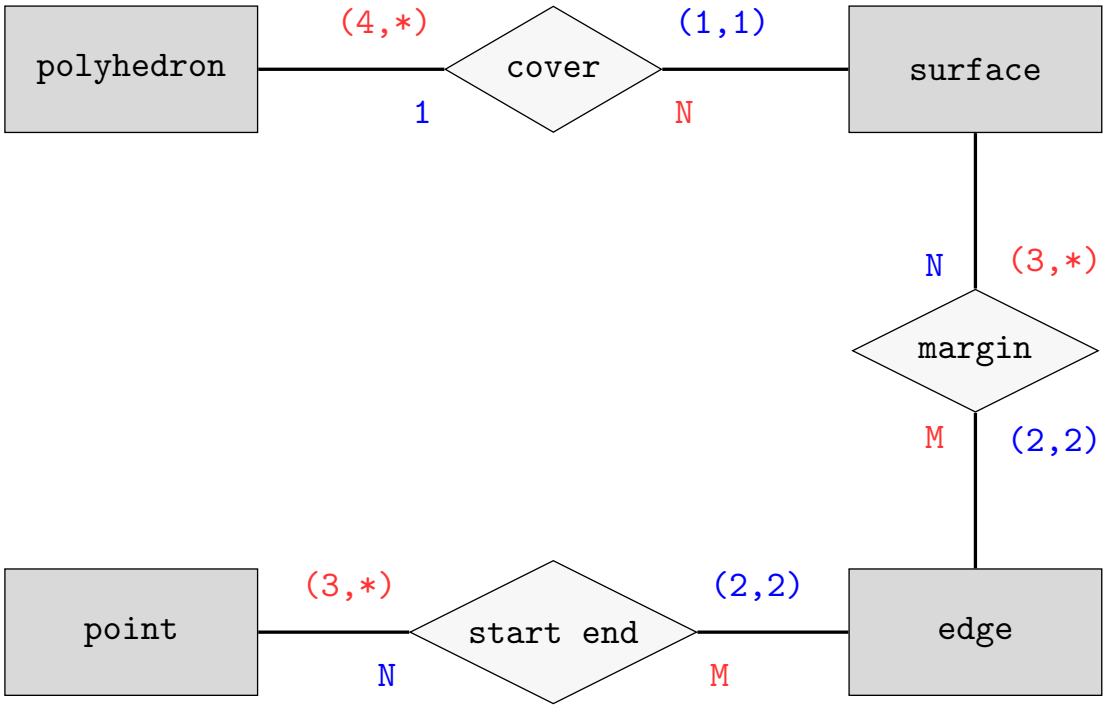
given:
n-ary relationship type R for the entity types $E_1, \ldots E_n$.

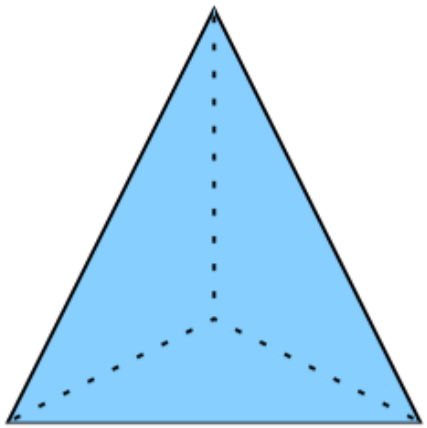


for every entity e_i of type E_i there is

- at least min_i tuple of the form $(\ldots, e_i, \ldots) \in R$
- at most max_i tuple of the form $(\ldots, e_i, \ldots) \in R$

Ex: (min,max)-Notation





minimal polyhedron

Weak Entities

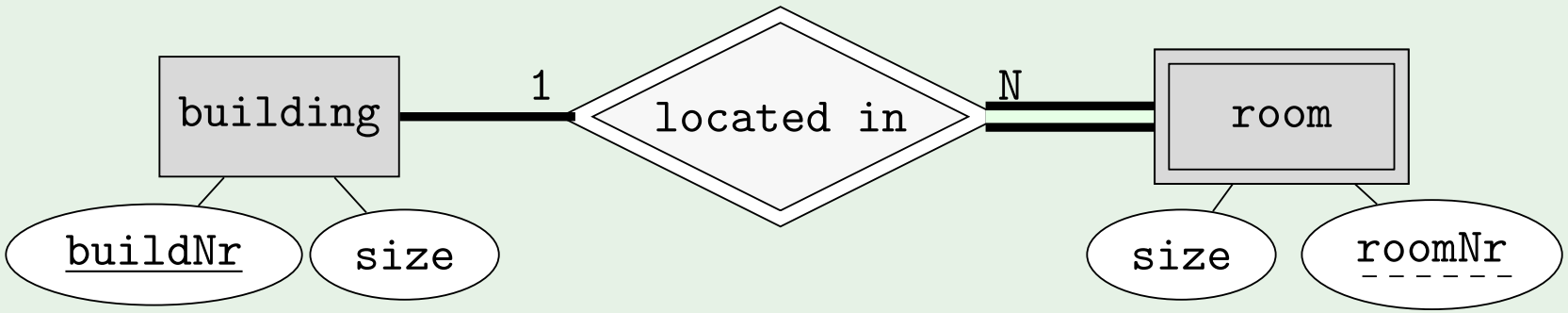
weak entities are entities whose existence depends on another (superior) entity; can be identified through a combination with the key of their respective superior entity

Weak Entities

weak entities are entities whose existence depends on another (superior) entity; can be identified through a combination with the key of their respective superior entity

Example

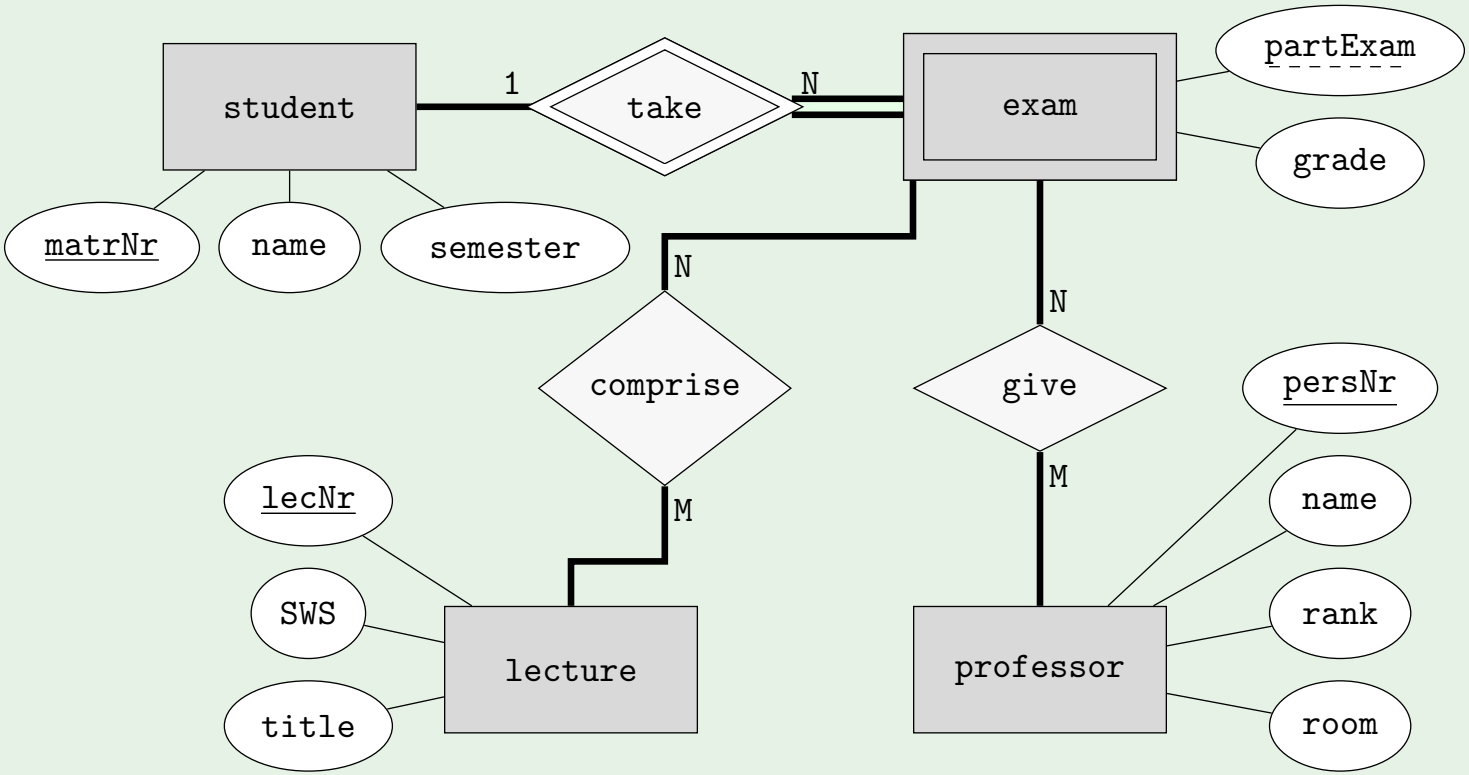
Consider a building with rooms and let the room number be unique only within the building. Then the key of the rooms is a combination of room and building number.



Weak Entities

Example

A diploma examination consists of several lectures, each of which is evaluated in partial exams by professors. An exam depends on the student, who takes the exam.



Generalization or Specialization (EER)

generalization is used to classify the entity types

Generalization or Specialization (EER)

generalization is used to classify the entity types

- common features of similar entity types can be assigned to a super type (P)

Generalization or Specialization (EER)

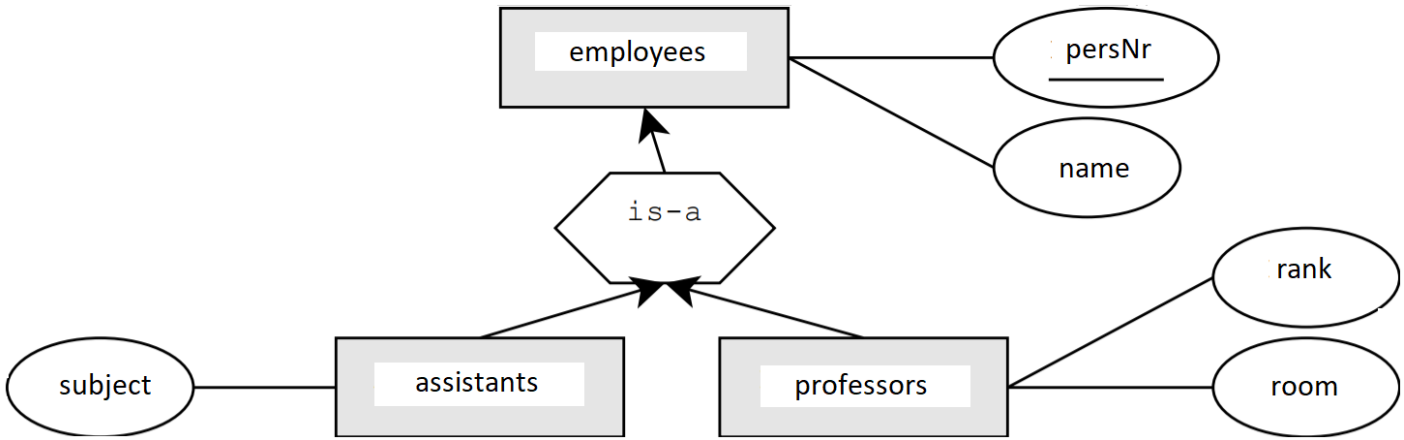
generalization is used to classify the entity types

- common features of similar entity types can be assigned to a super type (P)
- distinct features remain in the sub type (B)

Generalization or Specialization (EER)

generalization is used to classify the entity types

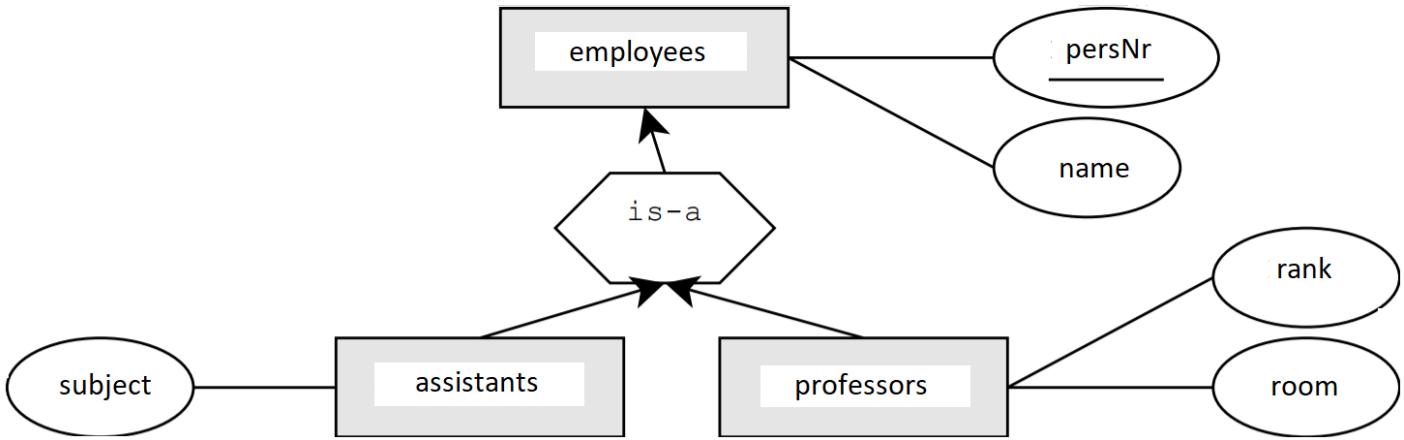
- common features of similar entity types can be assigned to a super type (P)
- distinct features remain in the sub type (B)



Generalization or Specialization (EER)

generalization is used to classify the entity types

- common features of similar entity types can be assigned to a super type (P)
- distinct features remain in the sub type (B)

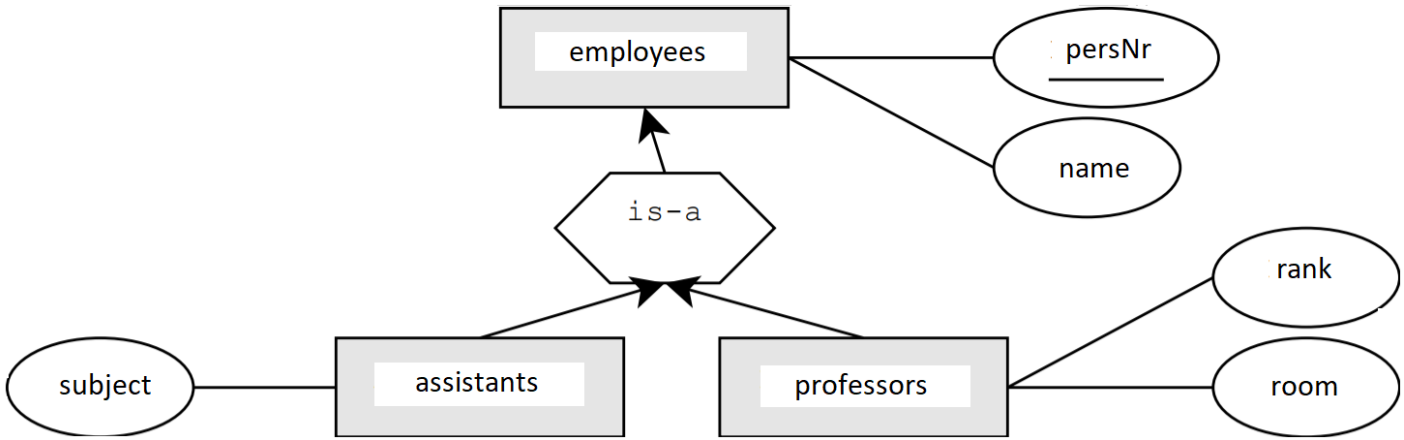


- disjoint generalization: $U_1 \cap U_2 = \emptyset$

Generalization or Specialization (EER)

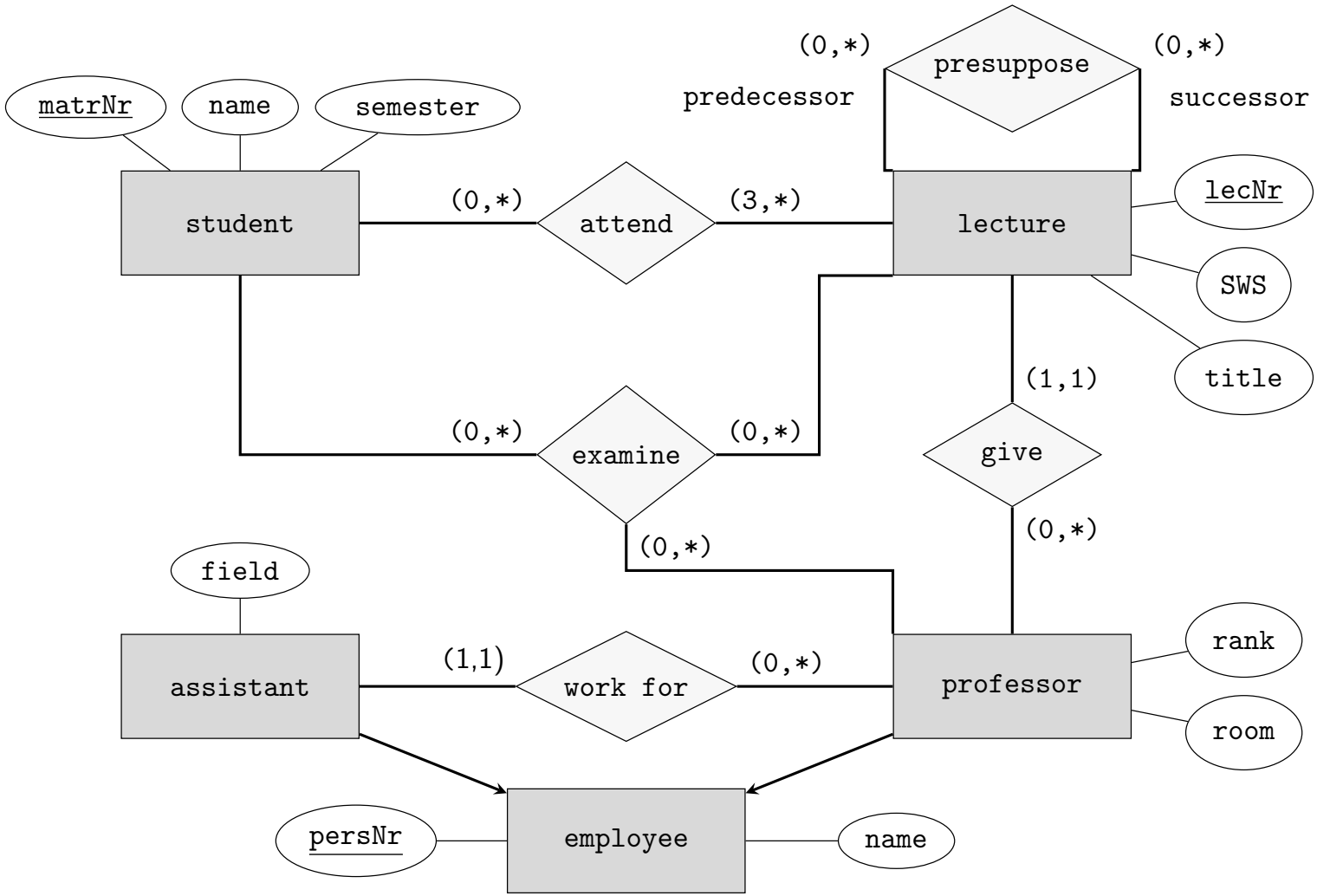
generalization is used to classify the entity types

- common features of similar entity types can be assigned to a super type (P)
- distinct features remain in the sub type (B)



- disjoint generalization: $U_1 \cap U_2 = \emptyset$
- complete generalization: $U_1 \cup U_2 = P$

Ex: University Schema with (min, max)-Notation and Generalization

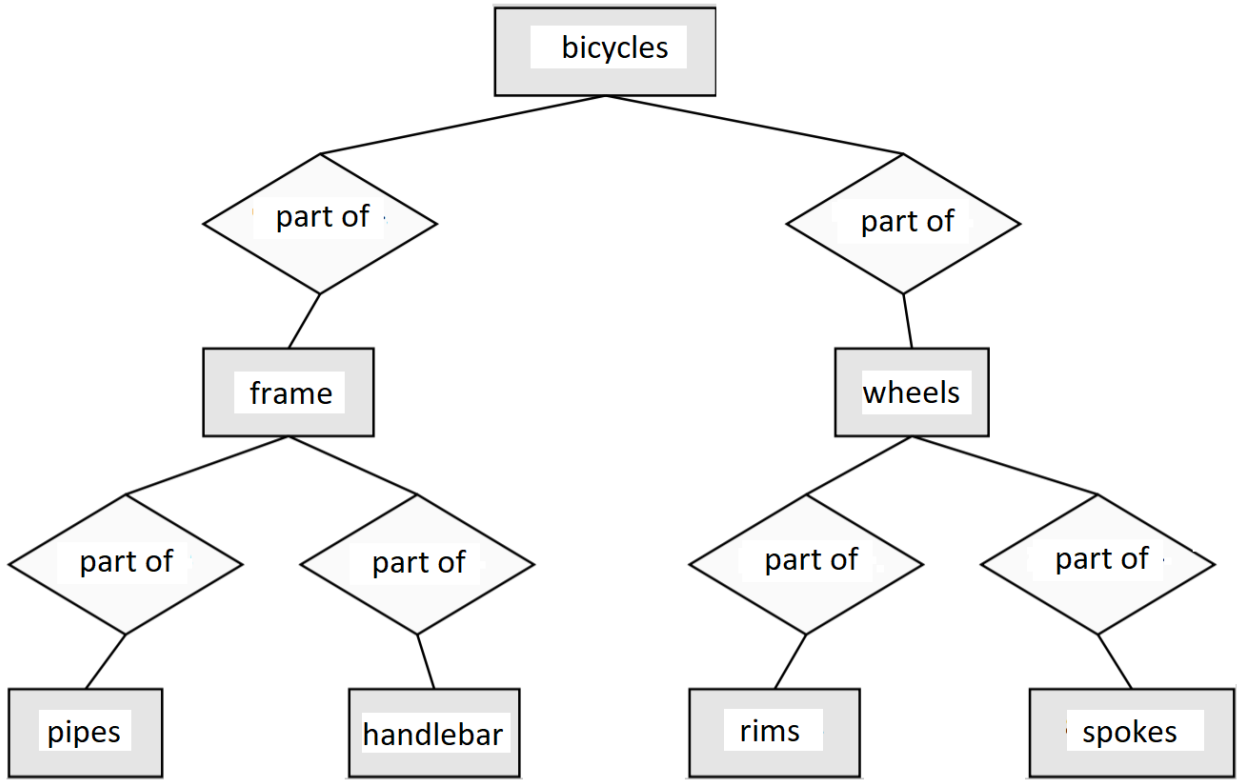


Aggregation (EER)

aggregation is used to assign distinct entity types to one another, whose totality (aggregation) defines a structured object type

Aggregation (EER)

aggregation is used to assign distinct entity types to one another, whose totality (aggregation) defines a structured object type



Overview

Overview

View Integration and Consolidation

huge applications: requirements analysis is split in different views

View Integration and Consolidation

huge applications: requirements analysis is split in different views

Example (university)

possible different views:

- lecturer view
- student view
- administration view
- ....

View Integration and Consolidation

huge applications: requirements analysis is split in different views

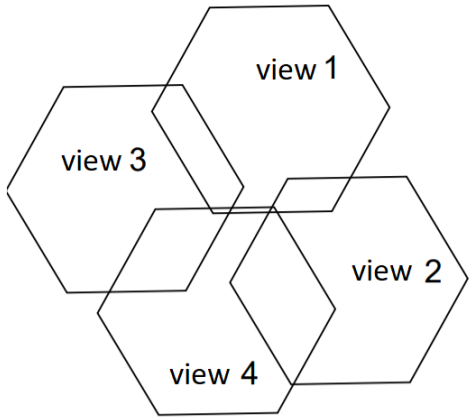
Example (university)

possible different views:

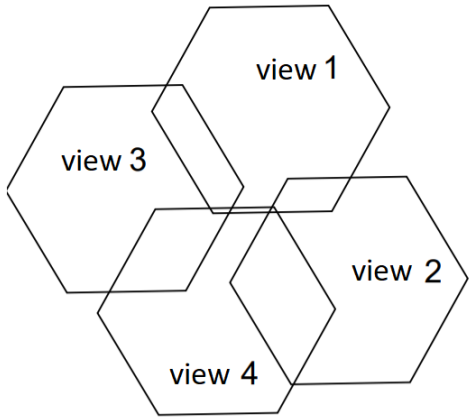
- lecturer view
- student view
- administration view
- ....

consolidation: construction of a global schema, which is redundancy-free, consistent and free from synonyms and homonyms

Consolidation Tree

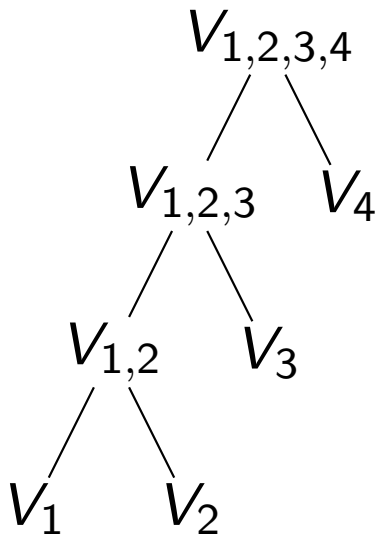


Consolidation Tree

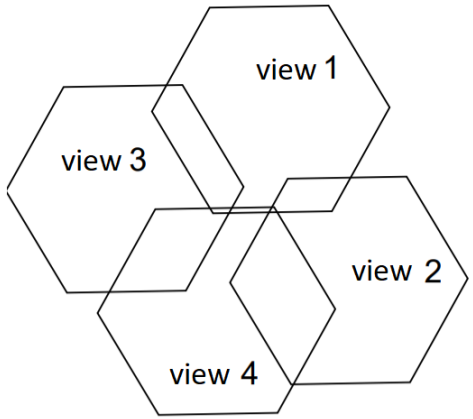


possible consolidation trees are:

- highest possible consolidation tree

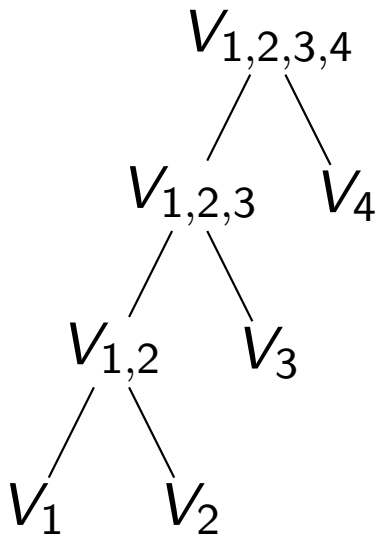


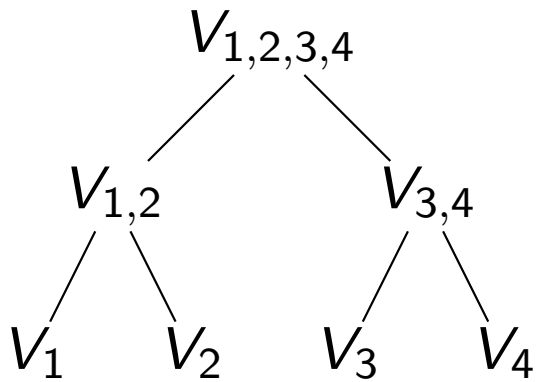
Consolidation Tree



possible consolidation trees are:

- highest possible consolidation tree
- lowest possible consolidation tree

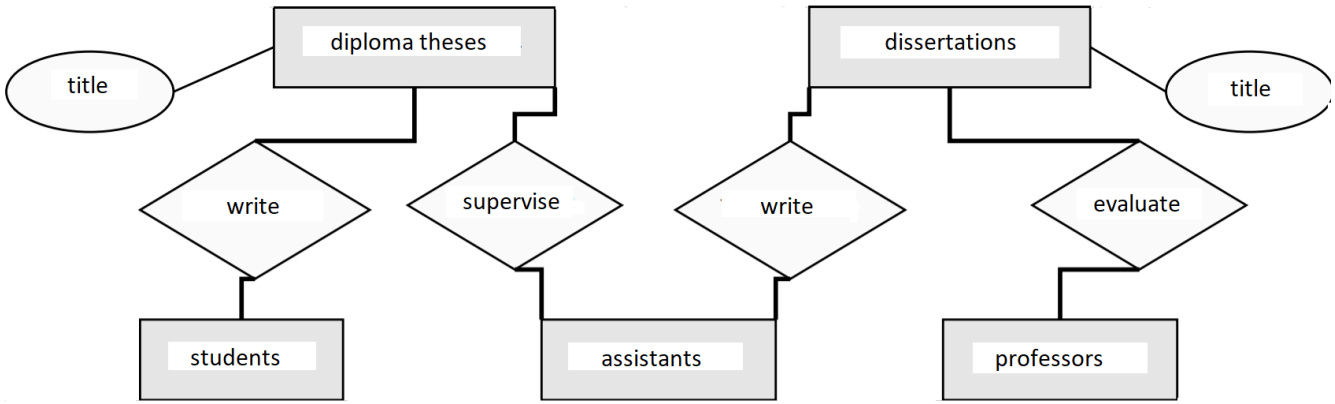




Ex: Consolidation

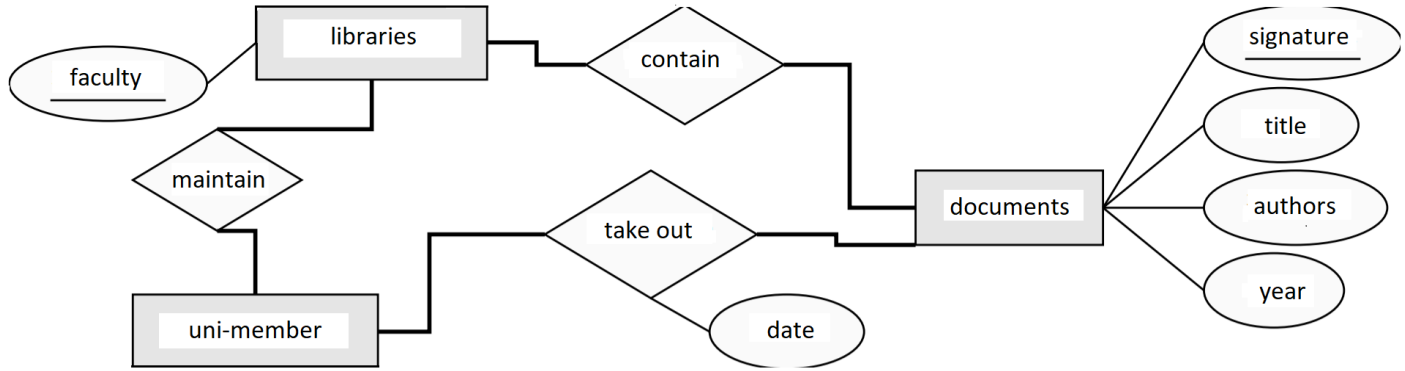
given are 3 views . . .

view 1: document creation as an examination



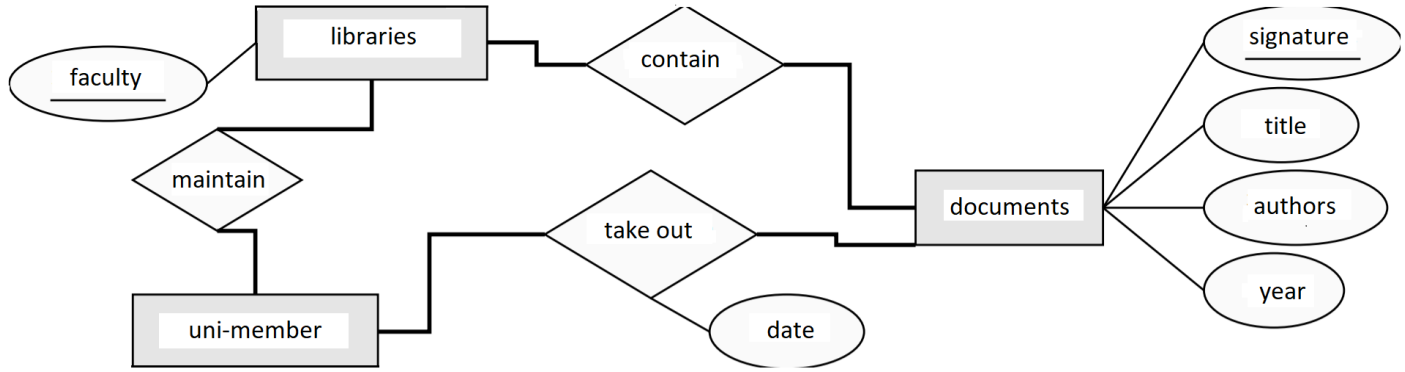
Ex: Consolidation

view 2: library management

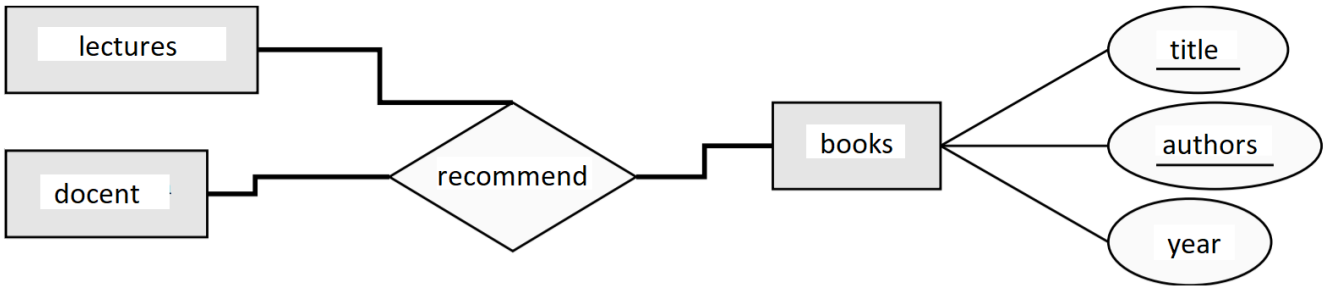


Ex: Consolidation

view 2: library management



view 3: book recommendations for lectures



Ex: Consolidation

. . . and the following information

- dissertations, diploma theses and books are specializations of documents of the library

Ex: Consolidation

. . . and the following information

- dissertations, diploma theses and books are specializations of documents of the library
- all documents are identified by the signature

Ex: Consolidation

. . . and the following information

- dissertations, diploma theses and books are specializations of documents of the library
- all documents are identified by the signature
- students, professors and assistants are generalized by uni-member

Ex: Consolidation

. . . and the following information

- dissertations, diploma theses and books are specializations of documents of the library
- all documents are identified by the signature
- students, professors and assistants are generalized by uni-member
- synonymous use of docent and professor

Ex: Consolidation

. . . and the following information

- dissertations, diploma theses and books are specializations of documents of the library
- all documents are identified by the signature
- students, professors and assistants are generalized by uni-member
- synonymous use of docent and professor
- all diploma theses and dissertations are maintained in the library

Ex: Consolidation

. . . and the following information

- dissertations, diploma theses and books are specializations of documents of the library
- all documents are identified by the signature
- students, professors and assistants are generalized by uni-member
- synonymous use of docent and professor
- all diploma theses and dissertations are maintained in the library
- faculty library: maintained by employees - revision of maintenance at specialization of uni-members

Ex: Consolidation

. . . and the following information

- dissertations, diploma theses and books are specializations of documents of the library
- all documents are identified by the signature
- students, professors and assistants are generalized by uni-member
- synonymous use of docent and professor
- all diploma theses and dissertations are maintained in the library
- faculty library: maintained by employees - revision of maintenance at specialization of uni-members
- relationships write and write in view 1 correspond to the authors of books in view 3

Ex: Consolidation