MOBI-DBS-B: Datenbanksysteme Datenbank-Modellierung

Vorlesung Sommersemester 2019

Tanya Braun, Universität zu Lübeck

Lehrauftrag SoSe 19, Universität Bamberg

Datenbank-Modellierung

Inhalte

- Anforderungsspezifikationen
- Entity-Relationship-Modell (ER)
- Vorgehen zur Erstellung
- Erweitertes ER-Modell (EER)
- Beziehung zu UML
- Dokumentation
- Qualitätskriterien

Bezug zu Phasen des DB-Entwurfs

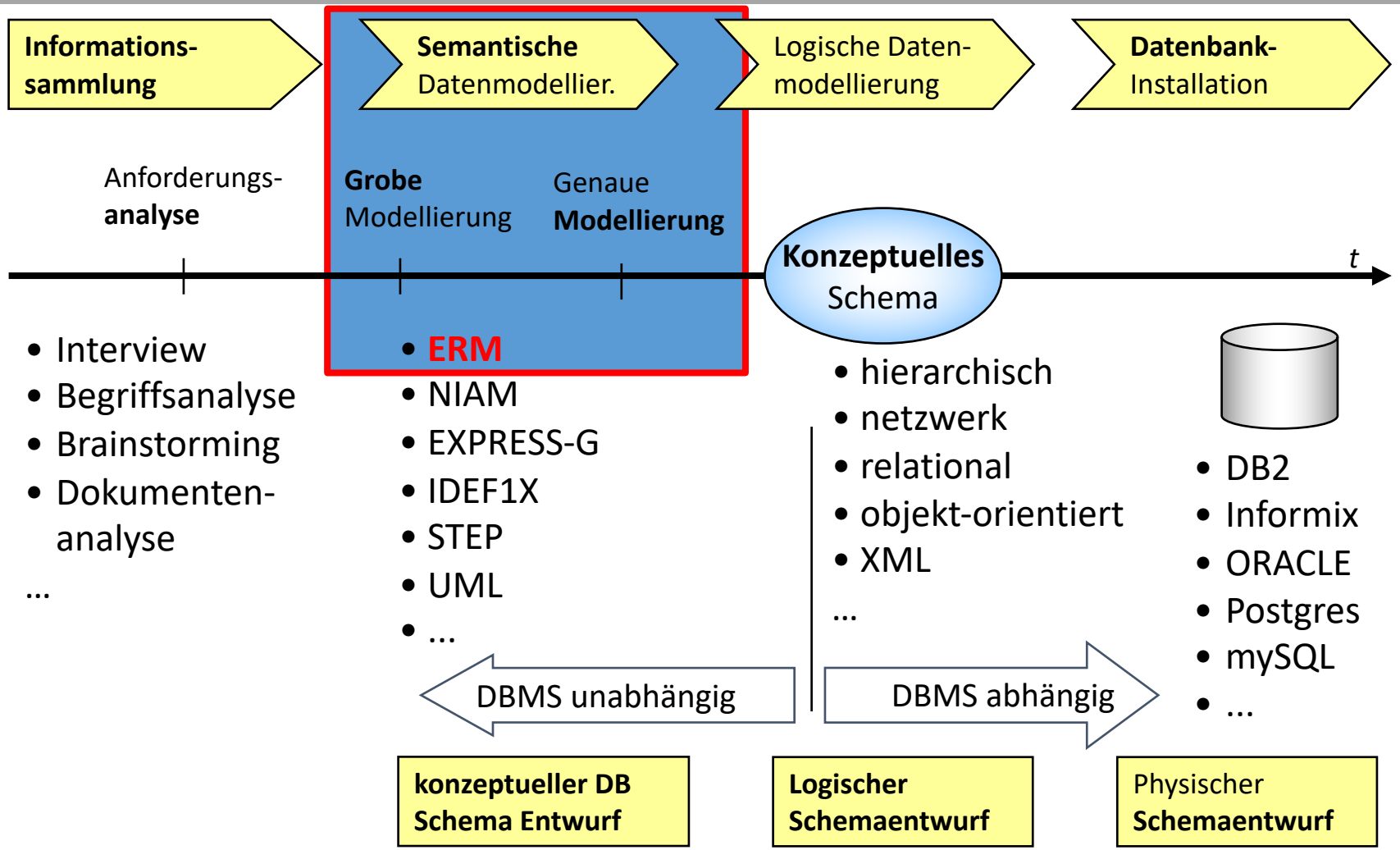
Kompetenzen

- Spezifikationen in Datenmodelle überführen
- Daten mit (E)ER modellieren
- ER-Modelle einem Kunden erklären
- Wenn UML bekannt: (E)ER-Modelle mit UML darstellen und wissen, wo die Grenzen/Unterschiede liegen
- Technische Probleme in Datenmodellen erkennen
- Mögliche inhaltliche Probleme in Datenmodellen erkennen und durch Fragen klären

Wie kommt man zum konkreten Datenbanksystem?

Phasen des DB-Entwurfs

Die Phasen des DB-Entwurfs



Modelle

Modellierung

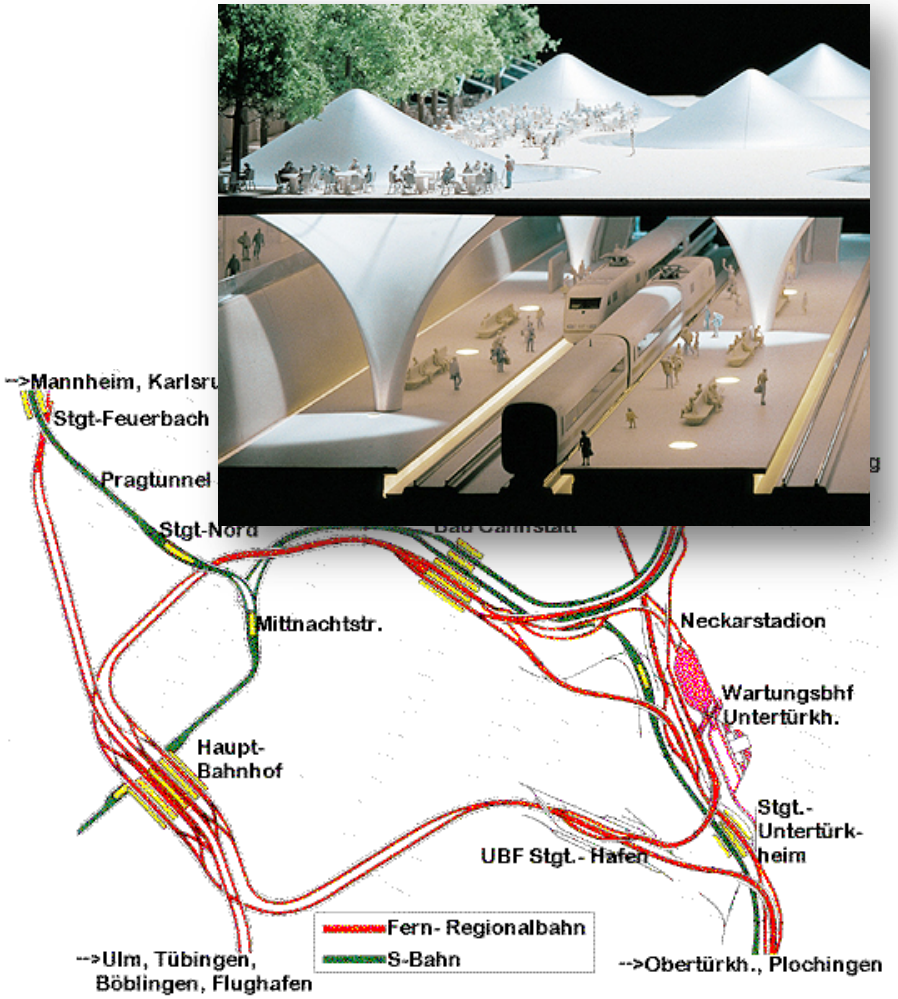
- Was ist ein Modell?
  - Abbild
  - Vorbild
- Typen von Modellen
  - Konkrete / abstrakte Modelle
    (Prototyp eines neuen Laptop / Erlösmodell)
  - Konkrete / abstrakte Originale
    (Laptop / Personalentwicklung)
- Und in der Informatik?

Was kann man modellieren?

- Struktur
  - Elemente eines Originals
- Eigenschaften
  - Attribute der Elemente
- Beziehungen
  - zwischen Elementen
- Verhalten
  - Dynamik der Elemente

Was davon bildet ein DB-Modell (Schema) ab?

Und was nicht?



Warum das Modell und nicht das Original?

- Operationen, die am Original nicht oder nur mit einem größeren Aufwand durchgeführt werden können
  - Simulation!
- Untersuchen und Verstehen komplexer Zusammenhänge
- Kommunikationsgrundlage
- Fixieren von Anforderungen

Verwendung bestimmt, was modelliert wird und was nicht:

- Gebäudemodell: optischer Eindruck
- Grundriss: Grundstücks- und Raumeinteilung
- Gewerkeplan: Bauabwicklung
- Kostenplan: Finanzierung

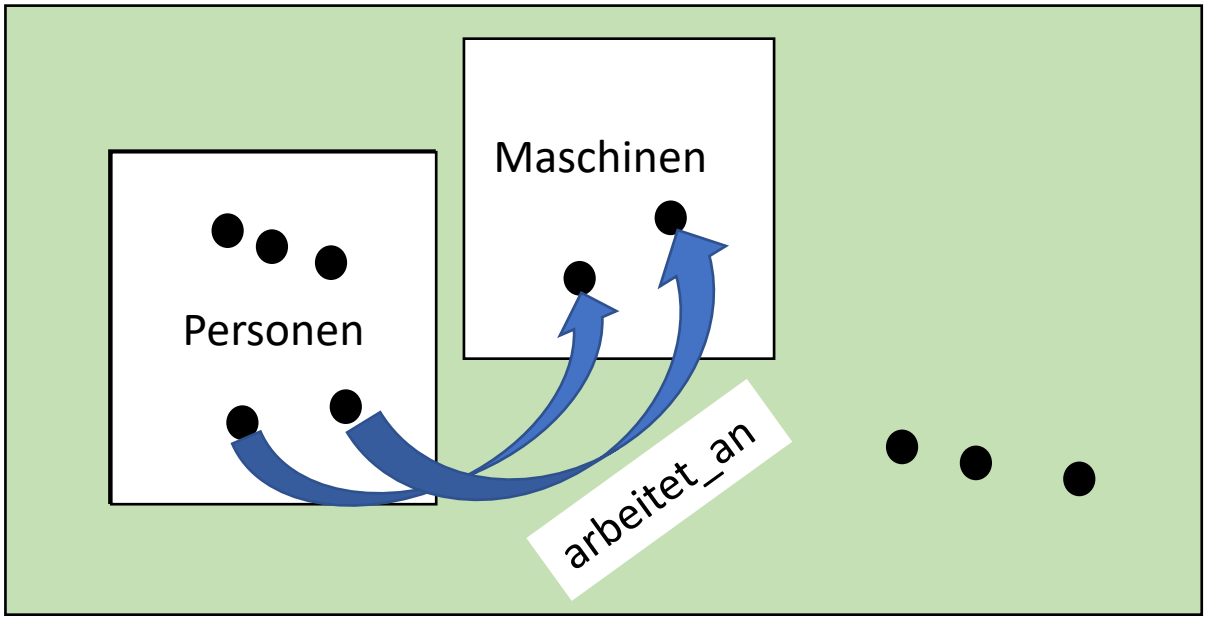
Verwendung sollte auch die Modellierungssprache bestimmen!

Entity-Relationship-Modell

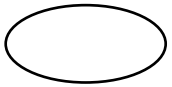
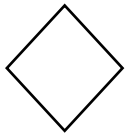
Konzeptuelle Modellierung

ER-Modellierung

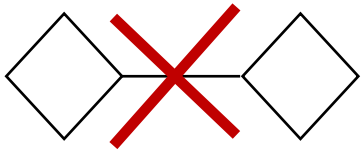
- Objekte (Entitäten) mit ähnlichen Eigenschaften können zu Mengen (Entitätstypen, Klassen) zusammengefasst werden
- Jedes Objekt ist „Instanz“ einer oder mehrerer Klassen
- Extension (Menge aller Instanzen einer Klasse)
- Objekte können in Beziehung gesetzt werden (Beziehungstyp, Relationship)

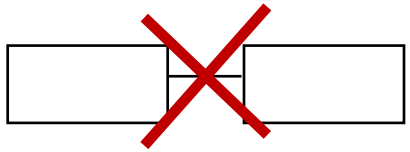


Grundlegende Elemente von ER-Diagrammen

- Objekttyp  (auch Entitätstyp oder Klasse genannt, Menge von Objekten)
- Werttyp  (für Basisdatentypen, Menge von Werten bzw. Literalen)
- Beziehungstyp  (Menge von Tupeln von Objekten)

- Elemente von ER-Diagrammen bilden einen bipartiten Graphen:



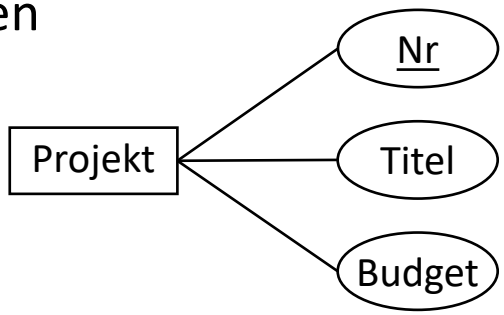


Verbindungen zwischen Symbolen der gleichen Typen sind nicht erlaubt.

P. Chen, The Entity-Relationship Model - Toward a Unified View of Data, ACM Transactions on Database Systems 1 (1), pp. 9–36, **1976**

Objekte/Entitäten und Attribute

- Entitäten
  - Wesentliche Konzepte („abgrenzbare Dinge“) realer oder gedachter Miniwelten
  - Haben eine eigenständige Existenz
- Attribute beschreiben Eigenschaften von Entitäten
- Beispiel:
  - Entität Projekt; wird beschrieben durch
    - Eine Nummer
    - Einen Titel
    - Das Budget



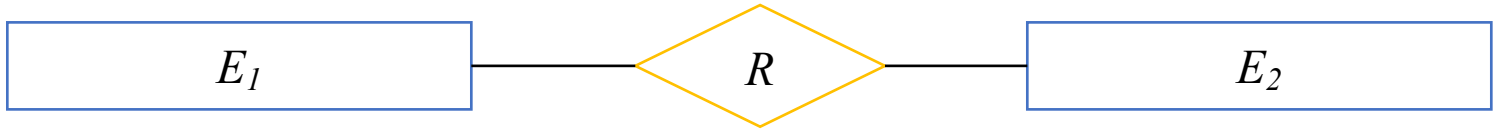
  - Mathematische Bedeutung von „Projekt“: Menge von Tupeln von Werten
    - Ein Tupel kann als „Aggregat“ von Basiswerten aufgefasst werden
    - (42, Flughafen, 100M), (21, Bahnhof, 50M)
- Bestimmte Attribute können Tupel eindeutig kennzeichnen
  - In der Graphik sind diese Attribute unterstrichen
  - Wir nennen die Attribute „Schlüssel“ -> starke Entität

Identifikation und Schlüssel

- Identifikation: Zwei grundlegende Ansätze in DB-Modellen
  - Referentielle Identifikation
    - Direkte Verweise auf Objekte (Zeiger in Programmiersprachen)
  - Assoziative Identifikation
    - Werte von Attributen oder Attributkombinationen, um sich eindeutig auf Objekte zu beziehen (Schlüssel: Ausweisnummer, Fahrgestellnummer, ...)
- Schlüssel
  - Attribute oder Attributkombinationen mit innerhalb einer Klasse eindeutigen Werten
  - Mehrere Schlüsselkandidaten möglich (Primärschlüssel, Sekundärschlüssel; dazu mehr später)

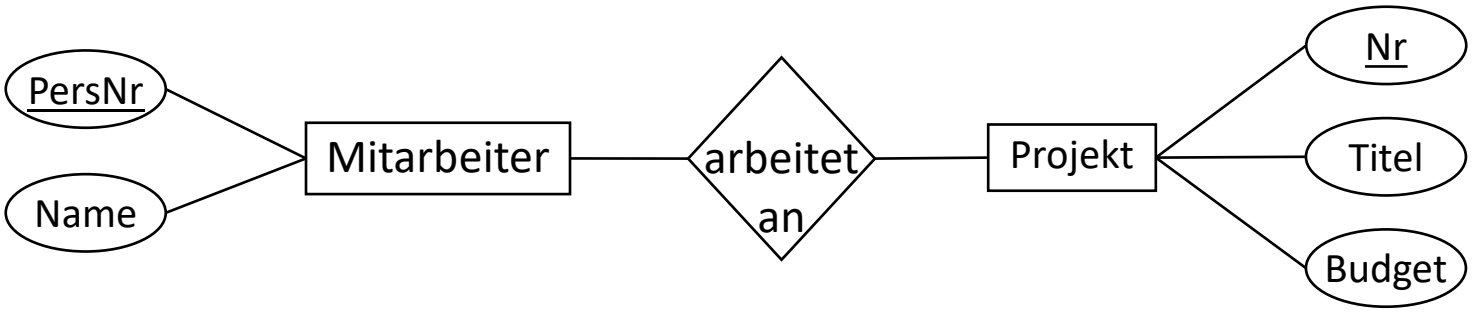
Assoziation/Relationship

- Objekte können miteinander in Beziehung gesetzt (assoziiert) werden:
  - Binäre (ternäre, ...) Beziehungen assoziieren zwei (drei, ...) Klassen oder Objekte
  - Allgemein: n-äre Beziehungen zwischen n Klassen oder Objekten, wobei n der Grad der Beziehung ist



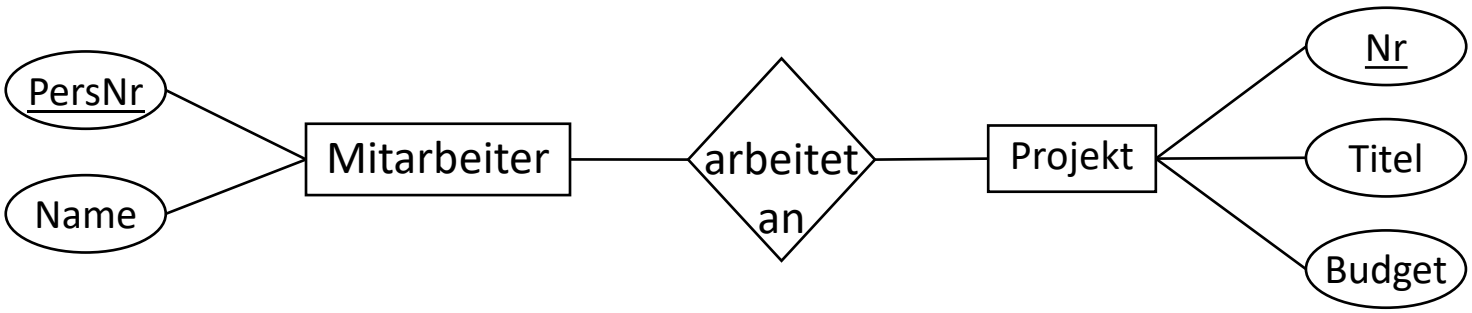
$$R \subseteq E_1 \times E_2$$

- Beispiel
  - An Projekten arbeiten Mitarbeiter
    - Mitarbeiter: Entität mit Attributen PersNr und Name

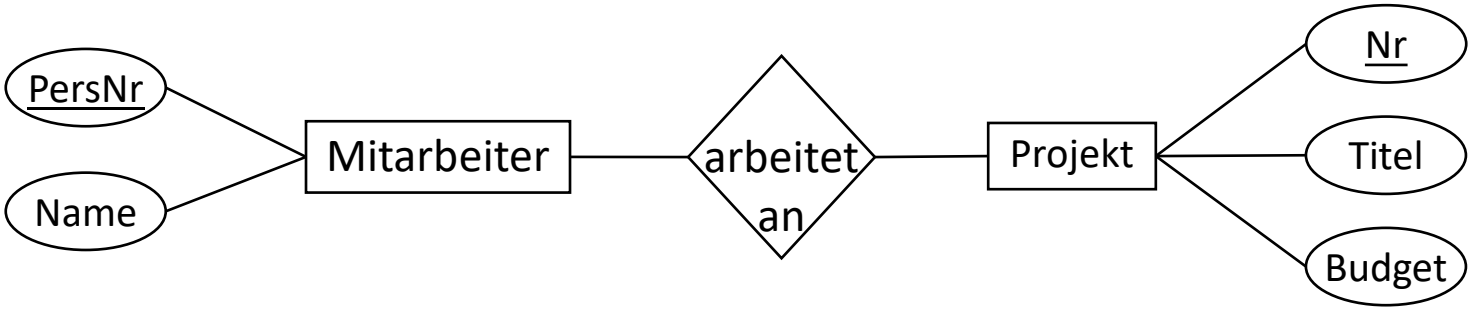


Assoziation/Relationship

- Schlüssel um Objekte in Assoziationen zu identifizieren
  - Schlüssel stellen als Attributwerte Beziehungen zu anderen Objekten her (Fremdschlüssel; auch dazu mehr später)
  - Referenzierte Objekte müssen existieren (→ referentielle Integrität)
- Beispiel
  - An Projekten arbeiten Mitarbeiter
    - Mitarbeiter: Entität mit Attributen PersNr und Name
  - Mathematische Bedeutung: Menge an verknüpften Tupeln
    - ((4711, Mustermann), (42, Flughafen, 100M))
    - Unter Zuhilfenahme der Schlüssel: (42, 4711)



Identifikation und referentielle Integrität



- Beispiel: Projekt
  - Projekte werden durch eine Nummer eindeutig identifiziert
  - Zwei Möglichkeiten zur Identifikation von Projekten innerhalb der Assoziation „arbeitet an“
  - Annahme: Projekt 54 existiert nicht → referentielle Integrität verletzt!

Referentielle Identifikation

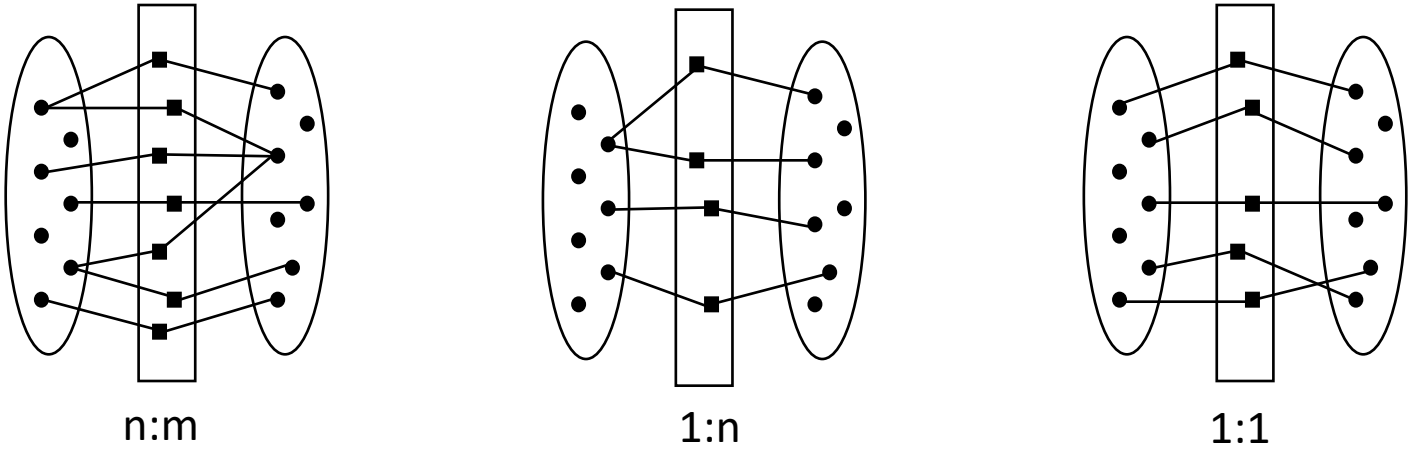
Projekt	Mitarbeiter
	...
	...
NULL	...
...	...

Assoziative Identifikation

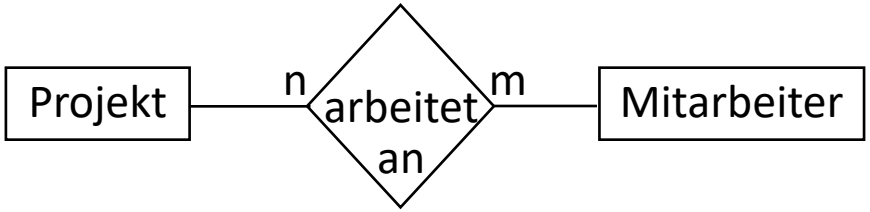
Projekt	Mitarbeiter
42	...
21	...
54	...
....	...

Funktionalitäten

- Funktionalitätsangaben definieren Einschränkungen:

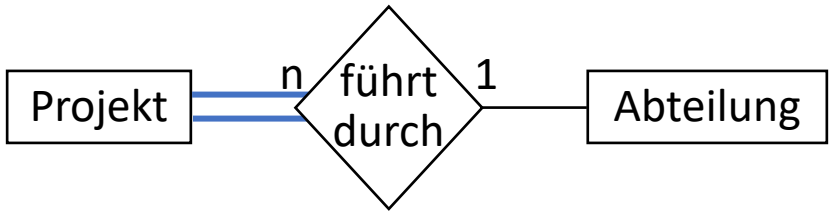


- Beispiel
  - An Projekten arbeiten Mitarbeiter. Ein Mitarbeiter kann an mehreren Projekten arbeiten. Jedes Projekt wird von beliebig vielen Mitarbeitern bearbeitet.

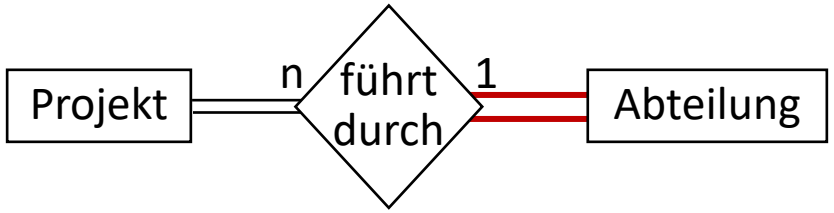


Partizipation

- Projekte werden von Abteilungen durchgeführt. Jedes Projekt muss einer Abteilung zugeordnet sein. Eine Abteilung kann mehrere Projekte ausführen.



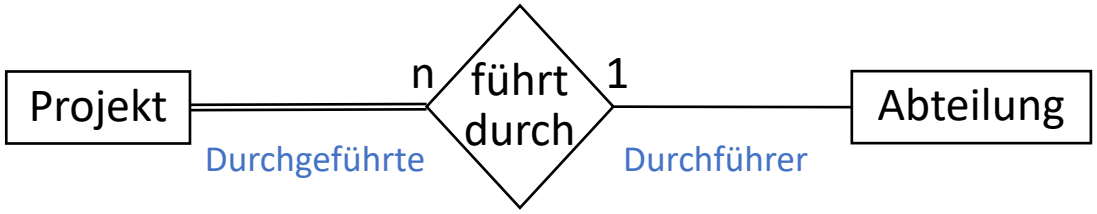
- Was bedeutet die Änderung?

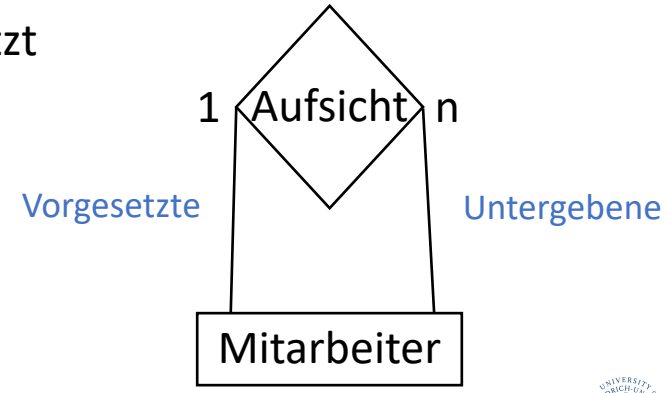


Partizipation

- Totale Partizipation: Jede Instanz einer Klasse muss mit einer Instanz der zweiten Klasse in Beziehung stehen (═══)
- Partielle Partizipation: Eine Instanz einer Klasse kann in Beziehung zu einer Instanz der zweiten Klasse stehen (———)

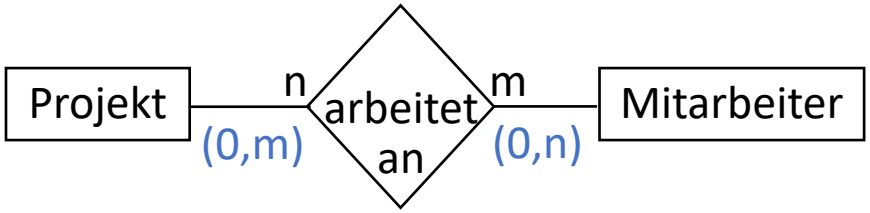
- Rollennamen (Namen für die Argumente der Relation) identifizieren die Menge der Instanzen, die mit einer anderen Instanz in Beziehung stehen.
  - Rollen können als abgeleitete Attribute verstanden werden, die die Menge der Instanzen als Attributwerte besitzen.
  - Besonders bei rekursiven Assoziationen eingesetzt



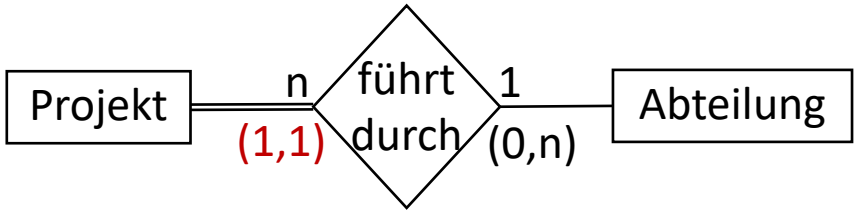


Funktionalitäten: (min, max)-Kardinalitäten

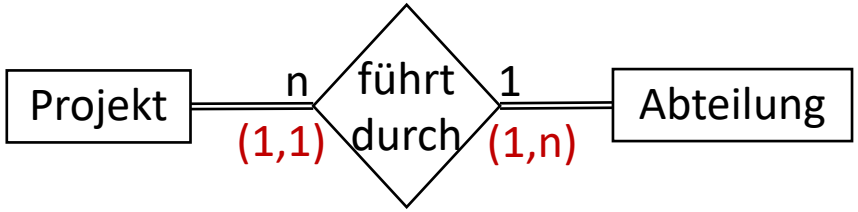
- Explizite Angabe der Kardinalitäten



  - Totale Partizipation von Projekt:



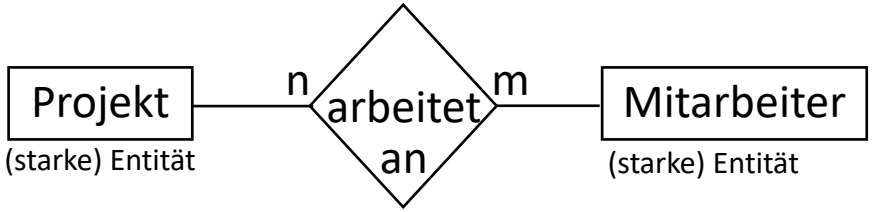
  - Totale Partizipation beider Entitäten:



Bemerkung: In der Literatur findet man weitere Beschriftungsregeln.

Schwache, existenzabhängige Entitäten

- Starke Entitäten
  - Besitzen ausgezeichnete Attribute zur eindeutigen Identifikation (Schlüssel)

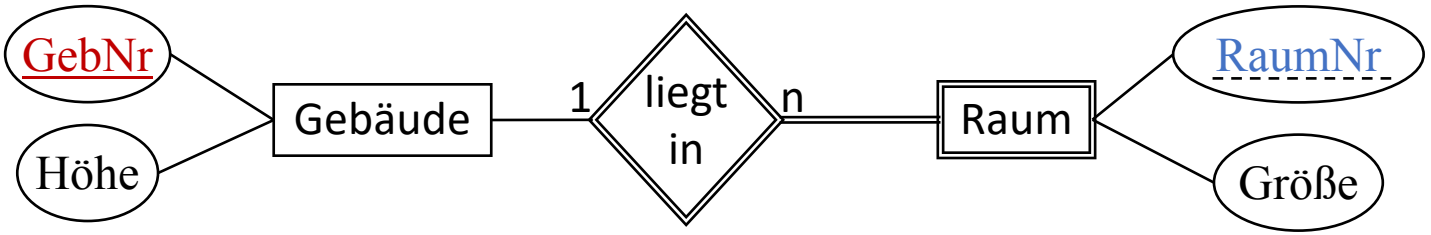


- Schwache Entitäten
  - Benötigen identifizierende Einheit zur Identifikation
  - Starke Entität mit schwacher Entität in Beziehung
  - Die Beziehung „Angehörige von“ stellt die Verbindung zur identifizierenden Einheit (Angestellter) her

Schwache Entitäten und Schlüssel

- Beziehung zwischen starken und schwachem Typ ist immer 1:n (oder 1:1 in seltenen Fällen)
  - Warum kann das keine n:m-Beziehung sein?
- Schwache Entitäten haben partiellen Schlüssel
  - Wird eindeutig mit Schlüssel der identifizierenden Entität

- Noch ein Beispiel
  - RaumNr ist nur innerhalb eines Gebäudes eindeutig
  - Eindeutiger Schlüssel ist: GebNr und RaumNr

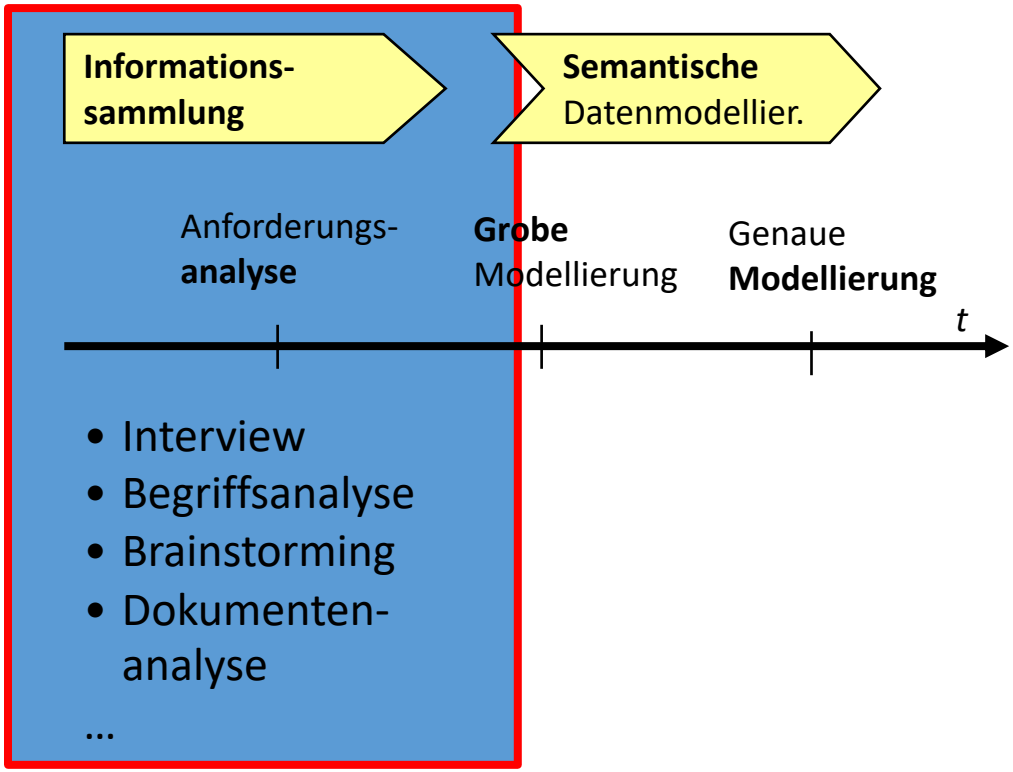


Von der Anforderungsdefinition

Zum ER-Diagramm

Anforderungsdefinition

- Gegeben: Anforderungsdefinition
  - Woher?



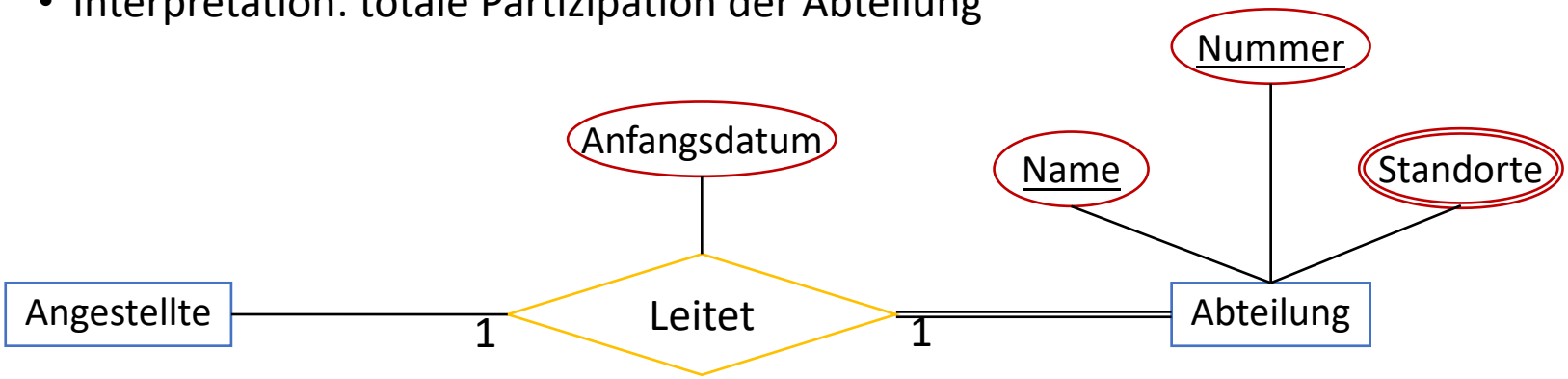
- Gesucht: ER-Diagramm

Anwendungsdefinition „Firma“ …

1. Die Firma ist in Abteilungen organisiert. Jede Abteilung hat eine eindeutige Bezeichnung, eine eindeutige Nummer und einen bestimmten Angestellten, der die Abteilung leitet. Es wird das Anfangsdatum verfolgt, ab dem dieser Angestellte die Leitung der Abteilung übernommen hat. Eine Abteilung verfügt über mehrere Standorte.
2. Eine Abteilung kontrolliert eine Reihe von Projekten, die jeweils einen Namen, eine eindeutige Nummer und einen einzigen Standort haben.
3. Zu jedem Angestellten sollen Name, Sozialversicherungsnummer, Adresse, Gehalt, Geschlecht und Geburtsdatum gespeichert werden. Ein Angestellter wird einer Abteilung zugewiesen, kann aber an mehreren Projekten arbeiten, die nicht unbedingt alle von der gleichen Abteilung kontrolliert werden. Dabei wird die Stundenzahl pro Woche verfolgt, die ein Angestellter an jedem Projekt arbeitet. Es wird auch der unmittelbare Vorgesetzte eines Angestellten gespeichert.
4. Zu Versicherungszwecken sollen die Familienangehörigen jedes Mitarbeiters gespeichert werden. Hierzu werden für jeden Angehörigen Vorname, Geschlecht, Geburtsdatum und Verwandtschaftsgrad zum jeweiligen Angestellten erfasst.

Anwendungsdefinition „Firma“ …

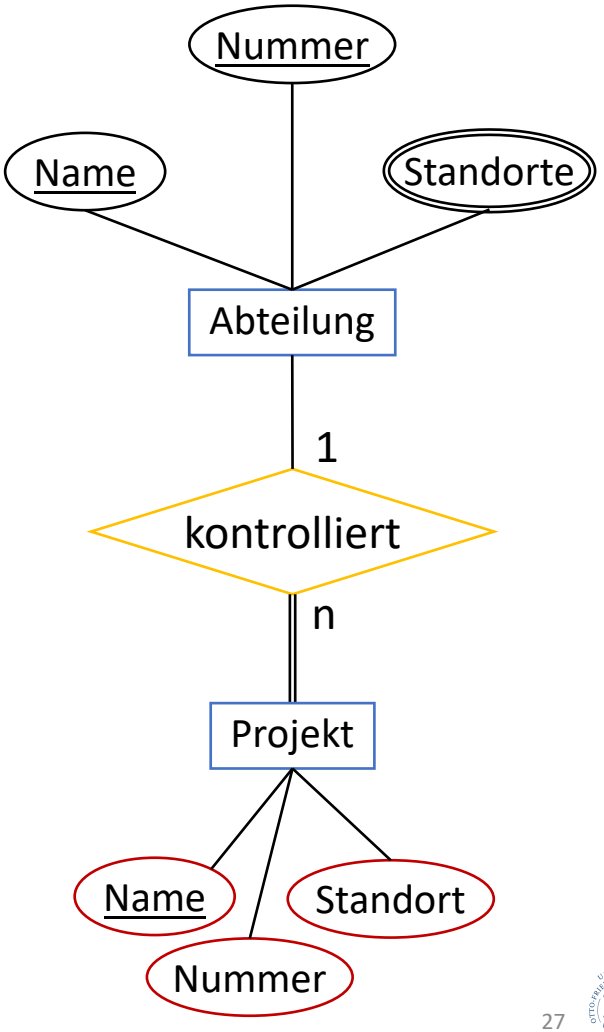
1. Die Firma ist in Abteilungen organisiert. Jede Abteilung hat eine eindeutige Bezeichnung, eine eindeutige Nummer und einen bestimmten Angestellten, der die Abteilung leitet. Es wird das Anfangsdatum verfolgt, ab dem dieser Angestellte die Leitung der Abteilung übernommen hat. Eine Abteilung verfügt über mehrere Standorte.
   - Interpretation: totale Partizipation der Abteilung



- Anforderungen nicht immer eindeutig
  - Standort auch als Entität denkbar

Anwendungsdefinition „Firma“ …

2. Eine Abteilung kontrolliert eine Reihe von Projekten, die jeweils einen Namen, eine eindeutige Nummer und einen einzigen Standort haben.
   - Interpretation: totale Partizipation der Projekte

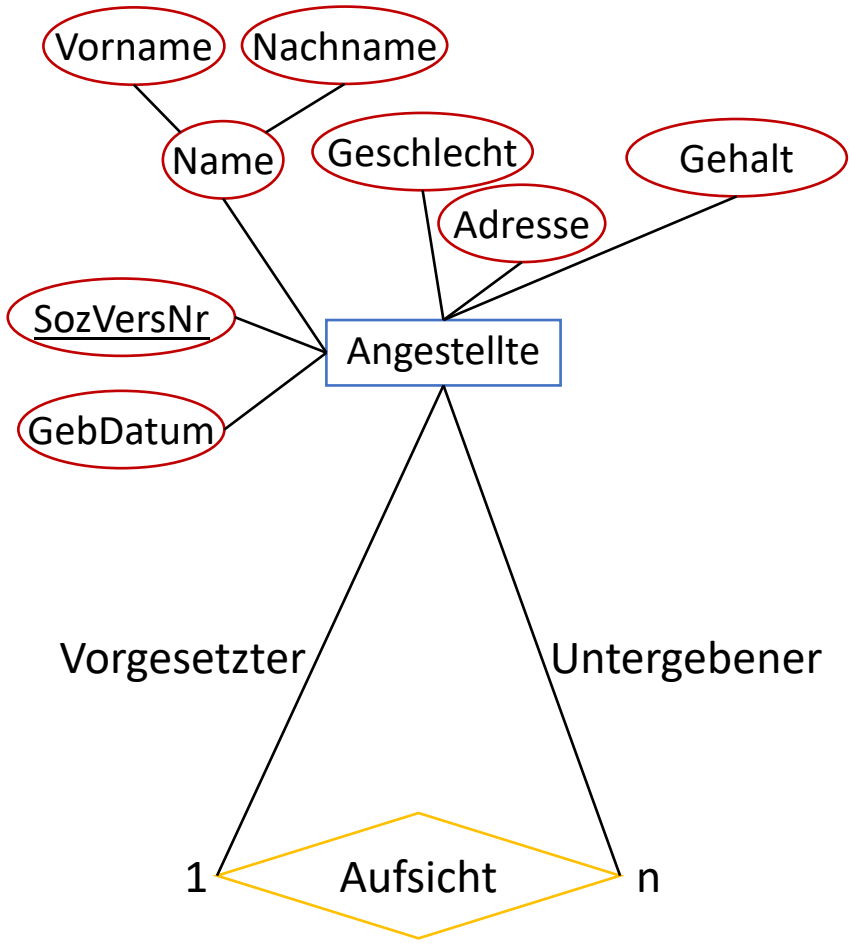


- Mehrwertige vs. einwertige Attribute
  - Standorte: mehrere Orte pro Abteilung

  - Standort: ein Ort pro Projekt

Anwendungsdefinition „Firma“ …

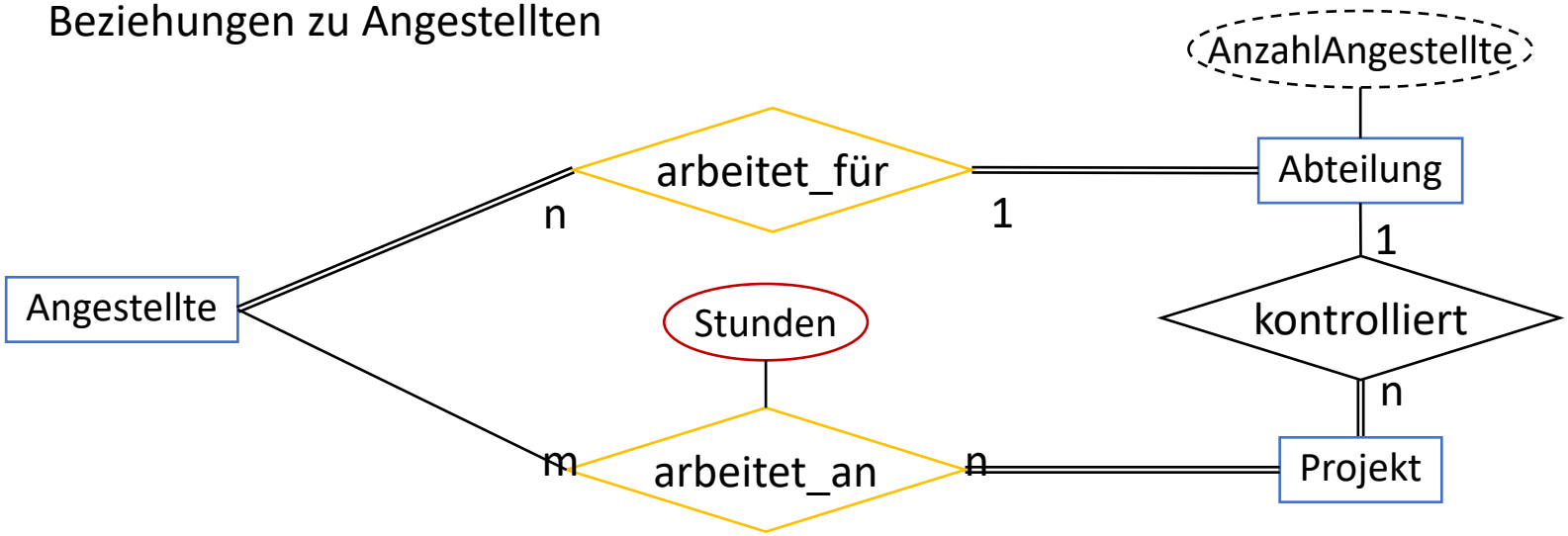
3. Zu jedem Angestellten sollen Name, Sozialversicherungsnummer, Adresse, Gehalt, Geschlecht und Geburtsdatum gespeichert werden. … Es wird auch der unmittelbare Vorgesetzte eines Angestellten gespeichert.
   - Interpretation: Name als mehrwertiges Attribut, auch bei anderen Attributen denkbar



- Atomare vs. zusammengesetzte Attribute
  - Atomar: unteilbar (nicht sinnvoll zerlegbar); z.B. SozVersNr
  - Zusammengesetzt: aus mehreren Attributen bestehend; z.B. Name besteht aus Vorname und Nachname

Anwendungsdefinition „Firma“ …

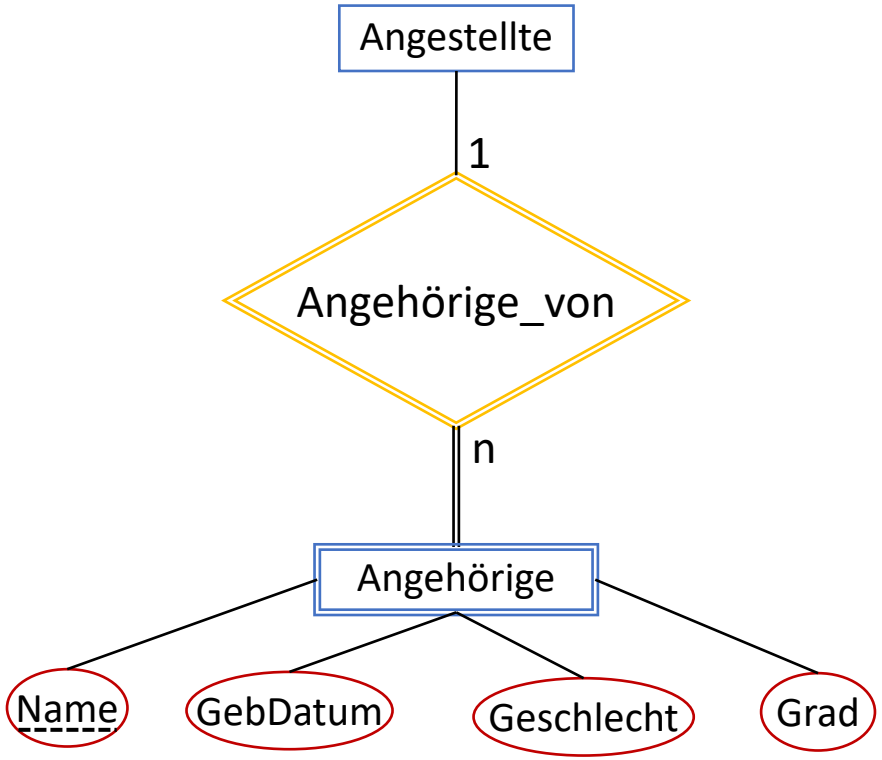
3. … Ein Angestellter wird einer Abteilung zugewiesen, kann aber an mehreren Projekten arbeiten, die nicht unbedingt alle von der gleichen Abteilung kontrolliert werden. Dabei wird die Stundenzahl pro Woche verfolgt, die ein Angestellter an jedem Projekt arbeitet...
   - Interpretation: totale Partizipation der Abteilung und der Projekte in den Beziehungen zu Angestellten



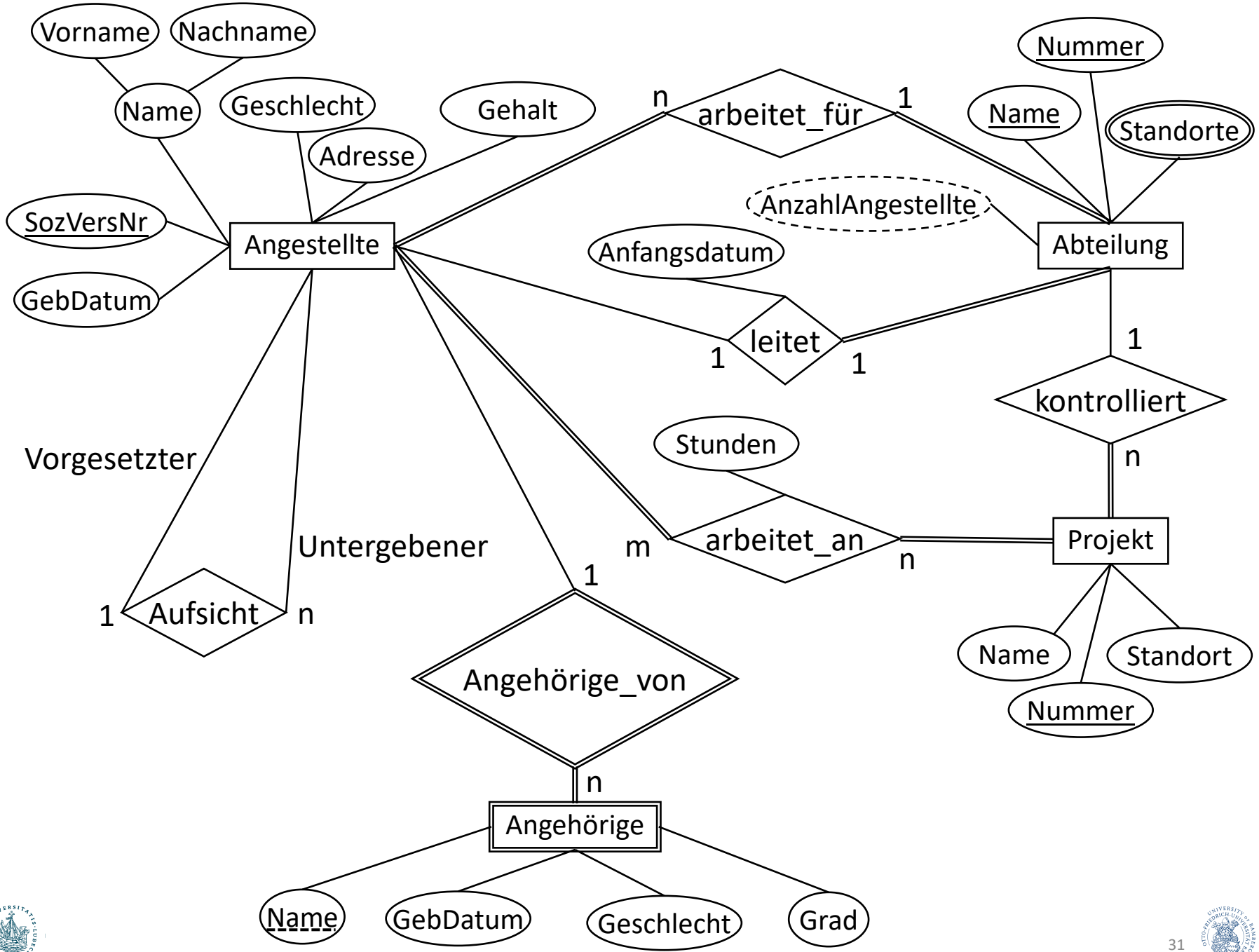
- Abgeleitetes Attribut: kann berechnet werden

Anwendungsdefinition „Firma“ …

4. Zu Versicherungszwecken sollen die Familienangehörigen jedes Mitarbeiters gespeichert werden. Hierzu werden für jeden Angehörigen Vorname, Geschlecht, Geburtsdatum und Verwandtschaftsgrad zum jeweiligen Angestellten erfasst.



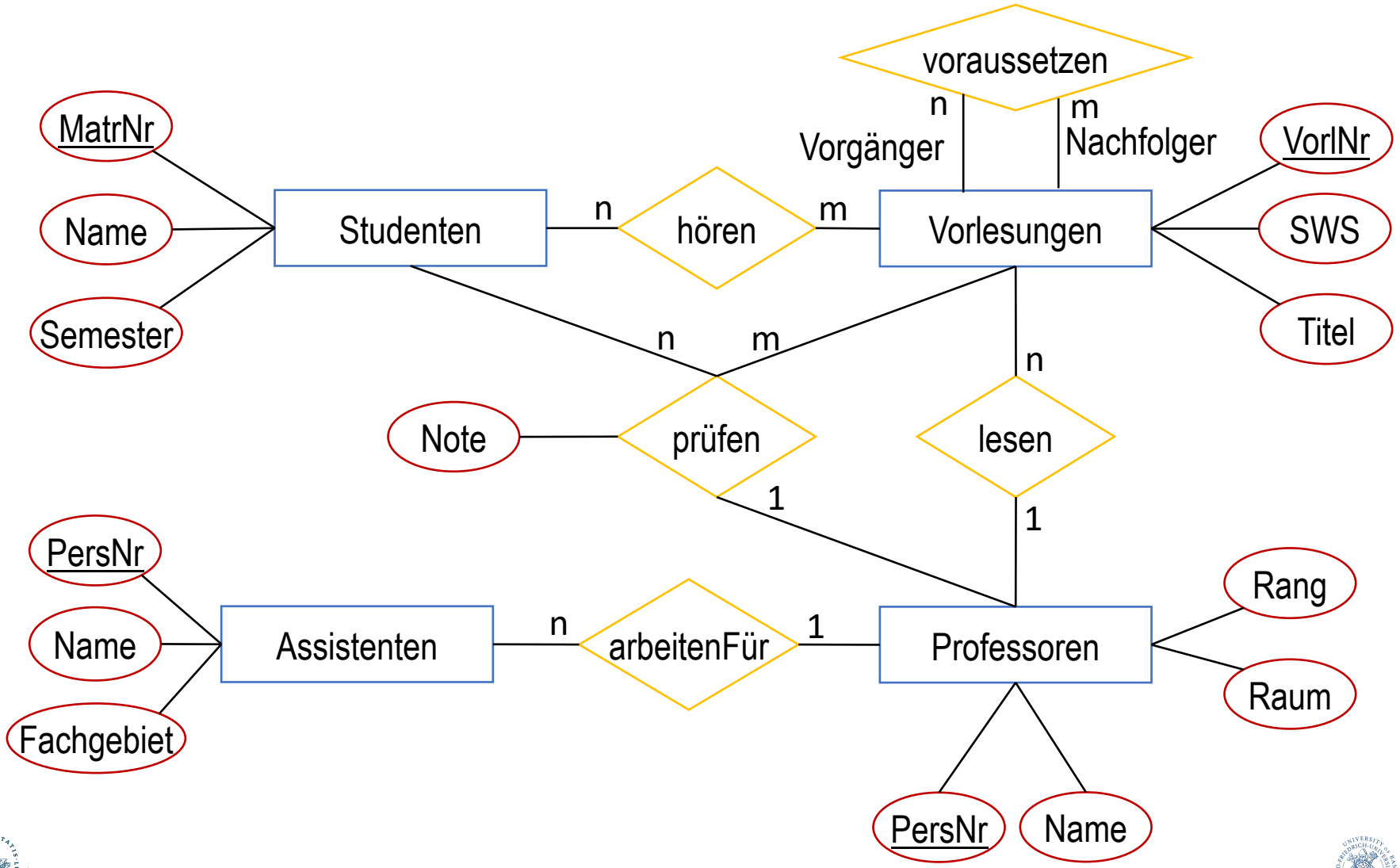
- Partielle Schlüsselattribute
  - Name ist nicht eindeutig ohne den dazugehörigen Angestellten


Vorname
Nachname
Name
Geschlecht
Gehalt
Adresse
SozVersNr
GebDatum
Angestellte
n
arbeitet_für
1
Nummer
Name
Standorte
AnzahlAngestellte
Abteilung
Anfangsdatum
leitet
1
1
1
kontrolliert
n
Stunden
Vorgesetzter
Untergebener
m
arbeitet_an
n
Projekt
1
Aufsicht
1
n
Name
Standort
Nummer
Angehörige_von
n
Angehörige
Name
GebDatum
Geschlecht
Grad

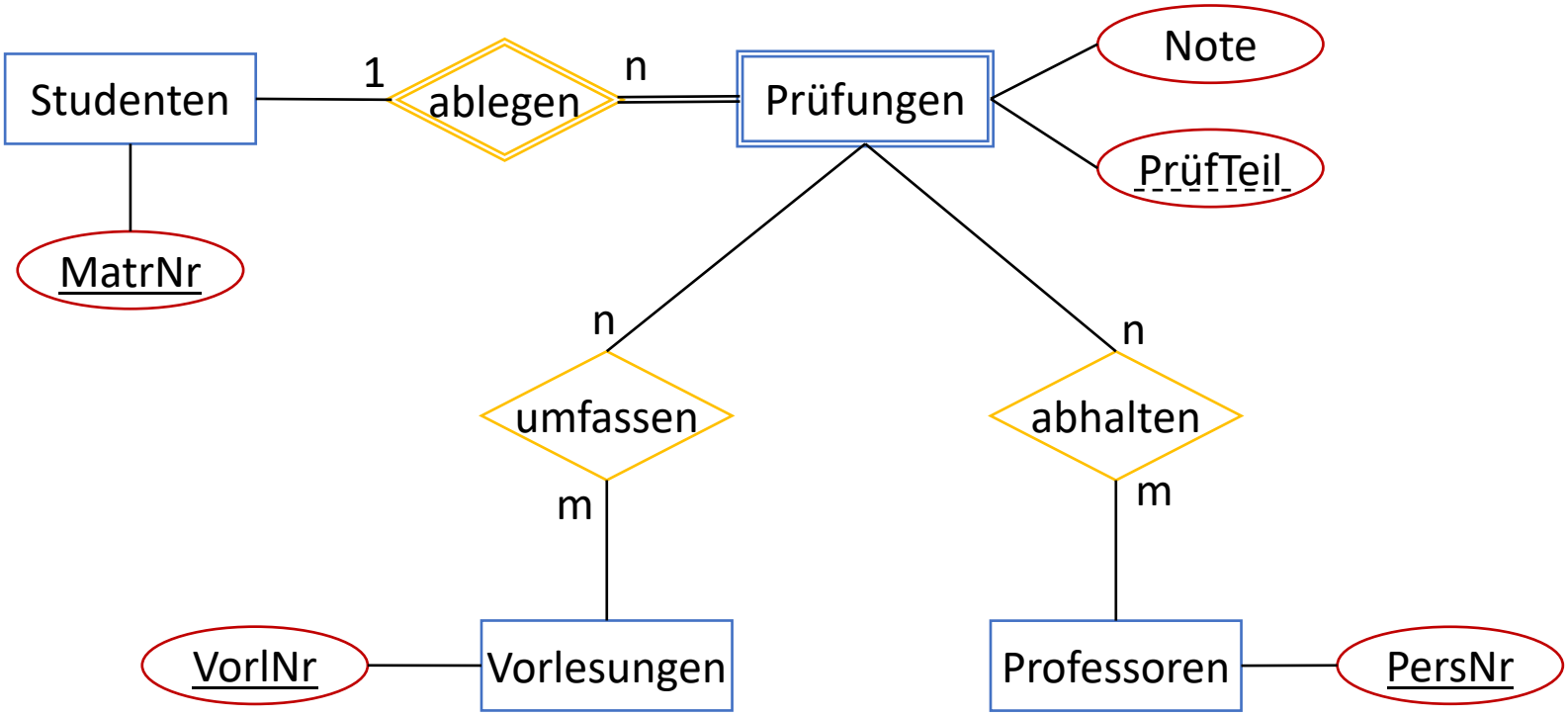
Und anders herum?

Interpretation von ER-Diagrammen

Universitätsschema

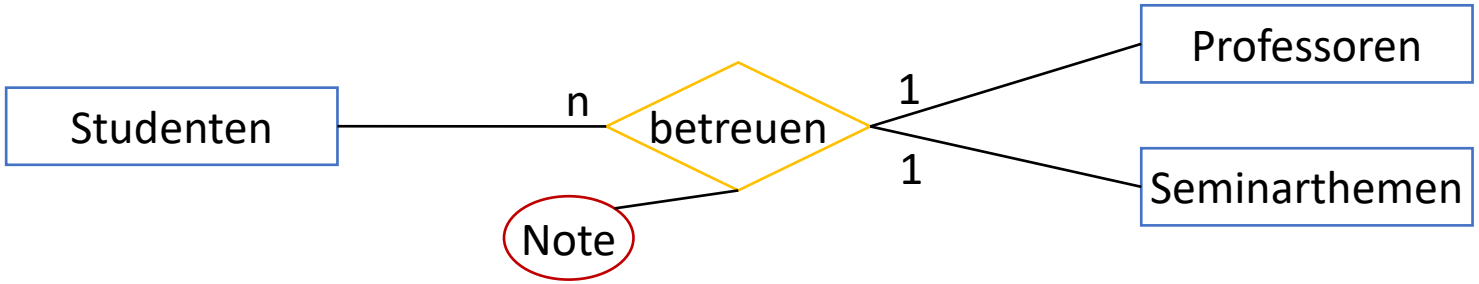


Prüfungen



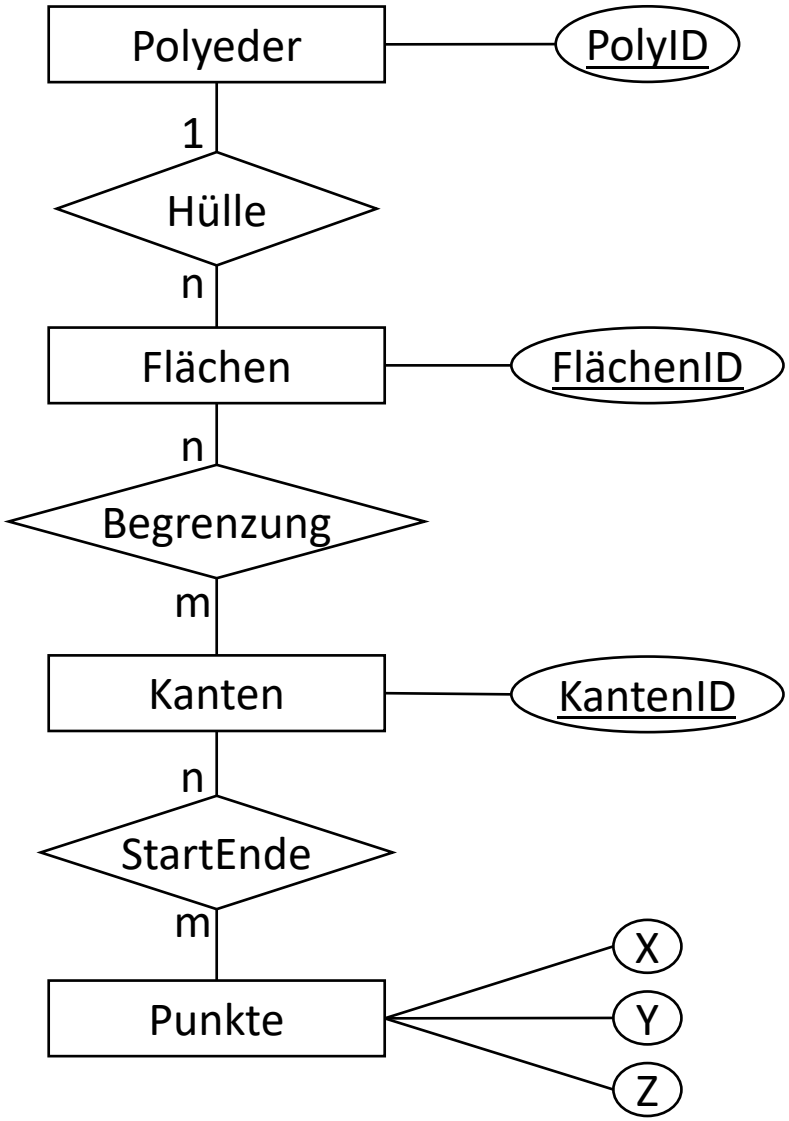
- Mehrere Prüfer in einer Prüfung
- Mehrere Vorlesungen werden in einer Prüfung abgefragt

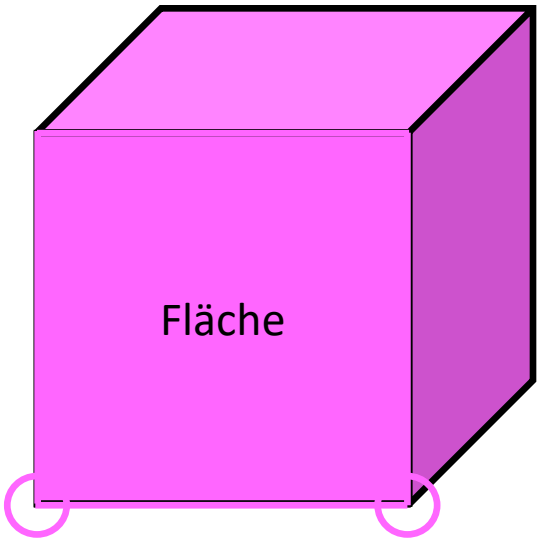
N-stellige Beziehung: *betreuen*



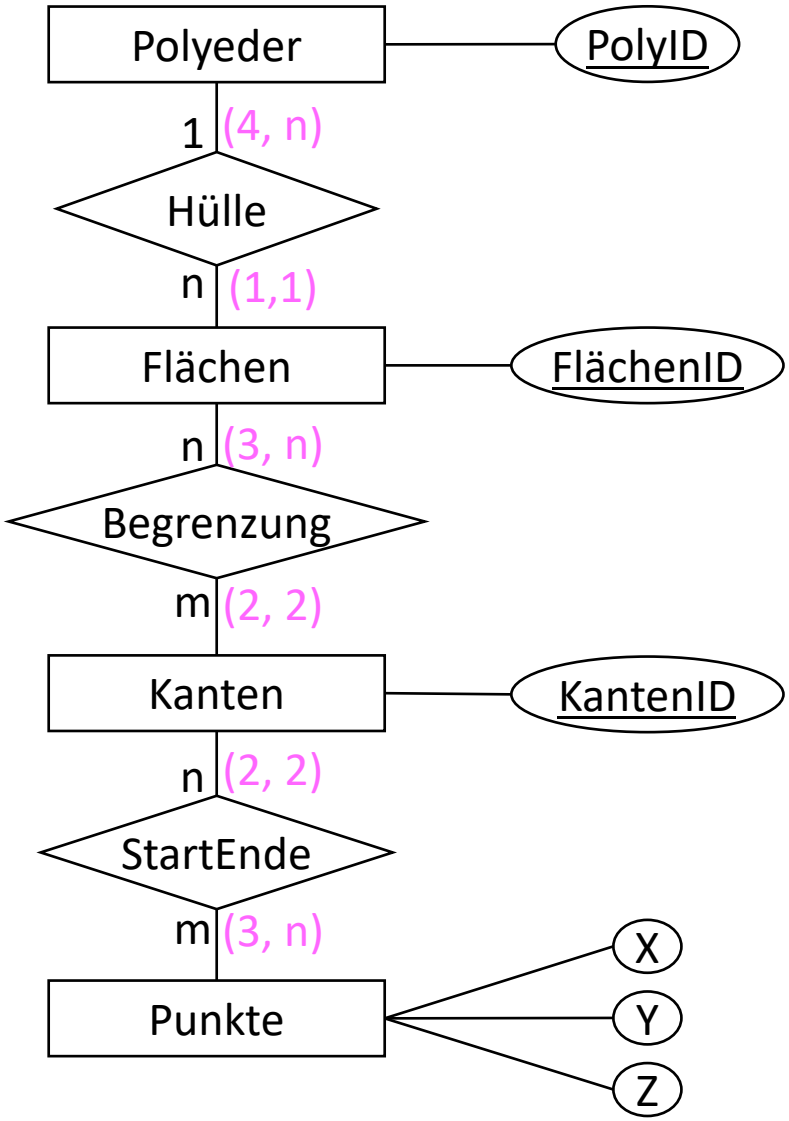
- Erzwungene Konsistenzbedingungen
  - Studenten dürfen bei demselben Professor bzw. derselben Professorin nur ein Seminarthema „ableisten“ (damit ein breites Spektrum abgedeckt wird).
  - Studenten dürfen dasselbe Seminarthema nur einmal bearbeiten – sie dürfen also nicht bei anderen Professoren ein schon einmal erteiltes Seminarthema nochmals bearbeiten.
- Mögliche Zustände
  - Professoren können dasselbe Seminarthema „wiederverwenden“ – also dasselbe Thema auch mehreren Studenten erteilen.
  - Ein Thema kann von mehreren Professoren vergeben werden – aber an unterschiedliche Studenten.

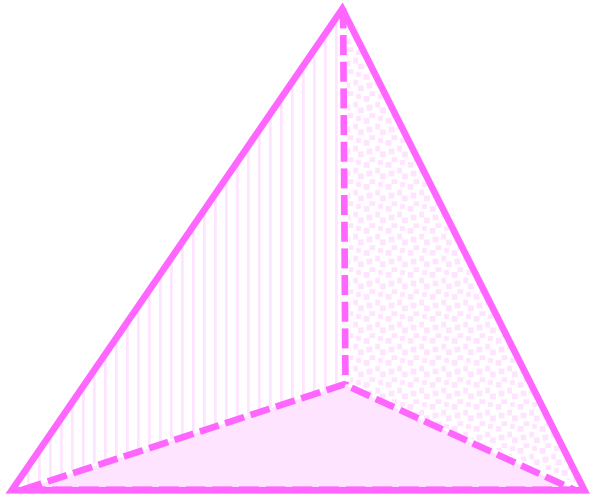
Komplex-strukturierte Entitäten





Komplex-strukturierte Entitäten





Kleinster Polyeder

Rückblick

- Modellierung
- ER-Modellierung
  - Notation
    - Entitäten
    - Assoziationen/Beziehungen
    - Attribute
    - Schlüssel
    - Funktionalitäten/Kardinalitäten
  - Von der Anforderungsdefinition zum ER-Diagramm
  - ER-Diagramm lesen