



## Kapitelübersicht

- ▶▶ Grundlagen von Datenbankmodellen
- ▶▶ Entity-Relationship-Modelle
- ▶▶ Objektorientierte Modelle: UML (nur in den Übungen)



## Grundlagen von Datenbankmodellen

### Begriff Datenbankmodell

*Ein Datenbankmodell ist ein System von Konzepten zur Beschreibung von Datenbanken. Es legt Syntax und Semantik von Datenbankbeschreibungen für ein Datenbanksystem fest.*

Datenbankbeschreibungen = Datenbankschemata

**Datenbankmodelle I**

Ein Datenbankmodell legt fest

1. **statische Eigenschaften**
  - (a) Objekte
  - (b) Beziehungeninklusive der Standard-Datentypen, die Daten über die Beziehungen und Objekte darstellen können,
2. **dynamische Eigenschaften wie**
  - (a) Operationen
  - (b) Beziehungen zwischen Operationen,sowie
3. **Integritätsbedingungen an**
  - (a) Objekte
  - (b) Operationen.

**Datenbankmodelle II**

**Klassische Datenbankmodelle sind speziell geeignet für**

- große Informationsmengen mit relativ starrer Struktur und**
- die Darstellung statischer Eigenschaften und Integritätsbedingungen (also die Bereiche 1(a), 1(b) und 3(a)).**



**Gegenüberstellung DB- und Programmiersprachenkonzepte:**

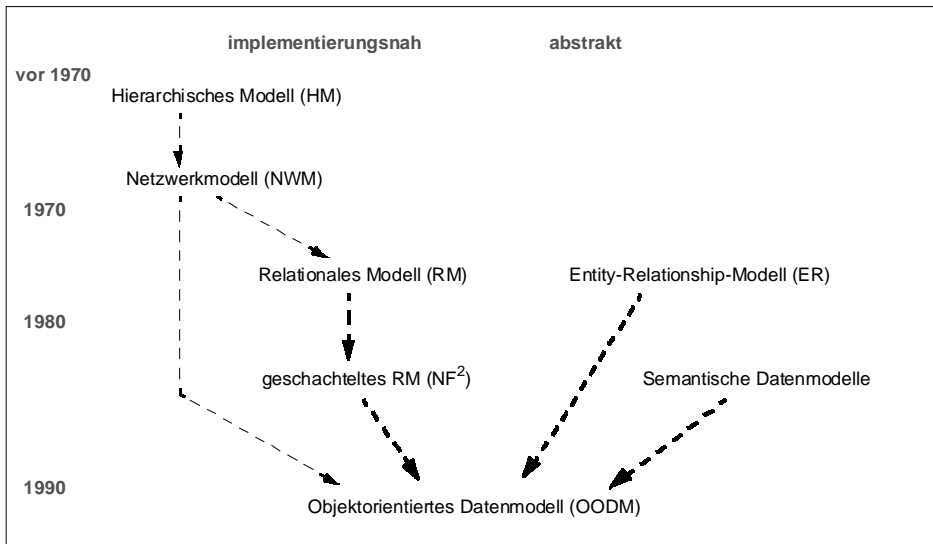
Datenbankkonzept	Typsystem einer Programmiersprache
Datenbank (DB)	Werte
Datenbankschema (DB-Schema)	Variablendeklaration mit Typdefinition
Datenbankmodell	Typsystem der Sprache
Datenbanksystem (DBS) = DB + DBMS	Programm mit Datenbestand
Datenbankmanagementsystem (DBMS)	Entwicklungs- und Laufzeitumgebung

**Achtung:**

Diese Gegenüberstellung vernachlässigt die Operationen auf Daten und gilt in erster Linie für den Vergleich klassischer DBS mit imperativen Programmiersprachen.



**Historische Einordnung von Datenmodellen**



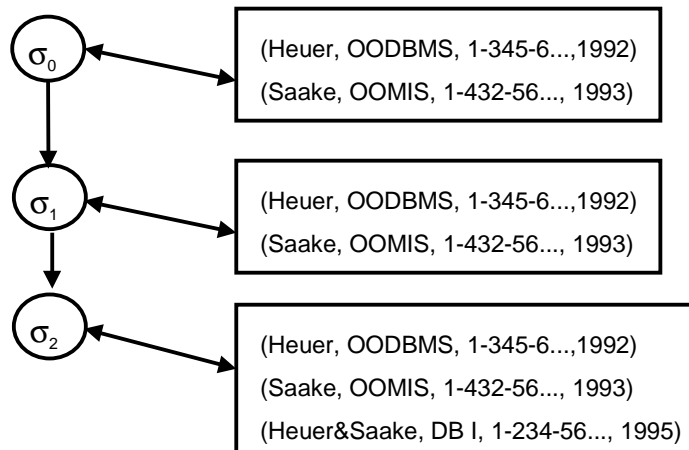


**Semantikfestlegung für Datenbankmodelle**

- ❑ **Wertebereiche:**
  - abstrakte Datentypen
- ❑ **Datenbankzustände**
  - Modell einer prädikatenlogischen Beschreibung
- ❑ **Gesamtsemantik**
  - Zustandsfolgen



**Formalisierung durch Zustandsfolgen**



**Semantikfestlegung am Beispiel I**

Trägermengen für *mögliche* Werte:  $\mu$

- $\mu(\mathbf{integer}) = \mathbb{Z}$  (die ganzen Zahlen)
- $\mu(\mathbf{string}) = C^*$  (Folgen von Zeichen aus  $C = \{a, b, \dots, z, A, B, \dots, Z\}$ )
- $\mu(\mathbf{set}(z)) = 2^{\mu(z)}$   
(die Potenzmenge über den Werten des Parameterdatentyps  $z$ , oder anders ausgedrückt die Mengen aller Teilmengen von möglichen Werten in  $\mu(z)$ )
- $\mu(\mathbf{tuple}(z_1, \dots, z_n)) = \mu(z_1) \times \dots \times \mu(z_n)$   
(das kartesische Produkt der Parameterwertebereiche)

**Semantikfestlegung am Beispiel II**

Datenbankentwicklung:

$$\underline{\sigma} = (\sigma_0, \sigma_1, \dots, \sigma_i, \dots).$$

Bedeutung einer DB-Variablen ( $T$ : Zeitachse):

$$\underline{\sigma}(\mathit{db}): T \rightarrow \mu(\mathit{typ}(\mathit{db}))$$

**Semantikfestlegung am Beispiel III**

Beispiel der Bücher-Datenbank:

$$\sigma(\text{Bücher}): T \rightarrow 2^{C^* \times C^* \times C^* \times \mathbb{Z}}$$

Ein konkreter Zustandswert zum Zeitpunkt 42:

$$\sigma(\text{Bücher})(42) = \{(\text{Heuer}, 00\text{DB}, 1-453-, 1992), \\ (\text{Saake}, 00\text{SIS}, 1-321-, 1993)\}$$

**Entity-Relationship-Modelle**

P.P. Chen im Jahre 1976

**Entity:** Objekt der realen oder der Vorstellungswelt, über das Informationen zu speichern sind

z. B. Vorlesungsveranstaltungen, Buch, Lehrperson, ...

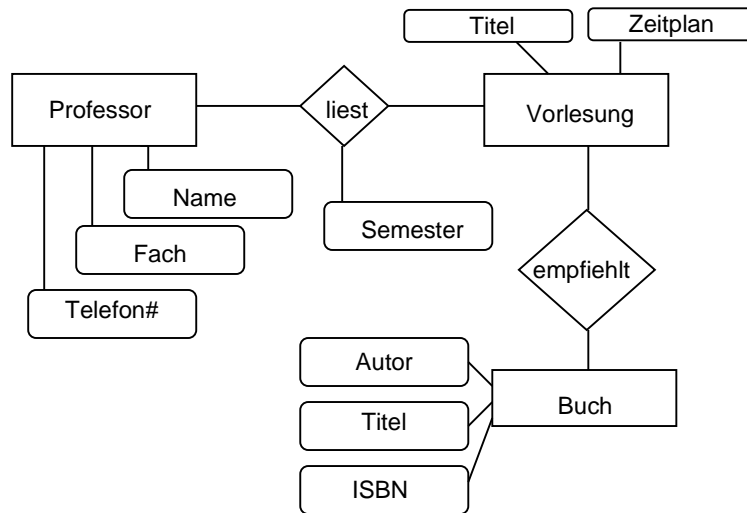
Auch Informationen über Ereignisse: Prüfungen, ...

**Relationship:** Beziehungen zwischen Entities, z. B. eine Lehrperson hält eine Vorlesung

**Attribut:** Eigenschaft von Entities oder Beziehungen, z. B. die ISBN eines Buches, der Titel einer Vorlesung, oder das Semester, in dem eine Vorlesung gehalten wird



Ein einfaches Beispiel



ER-Modellierungskonzepte und ihre Semantik I

Werte

$\mu(\text{int})$  : der Wertebereich  $\mathbb{Z}$  (die ganzen Zahlen) mit +, -, x, ÷, =, <, ...

$\mu(\text{string})$  : der Wertebereich  $C^*$  (Folgen von Zeichen aus der Menge C) mit +, =, <, ...

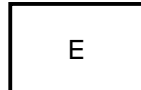
...

$\mu(D)$ : Interpretation von D, mögliche Werte einer Entity-Eigenschaft



ER-Modellierungskonzepte und ihre Semantik II

Entities: Entity-Typen, etwa  $E_1, E_2, \dots$



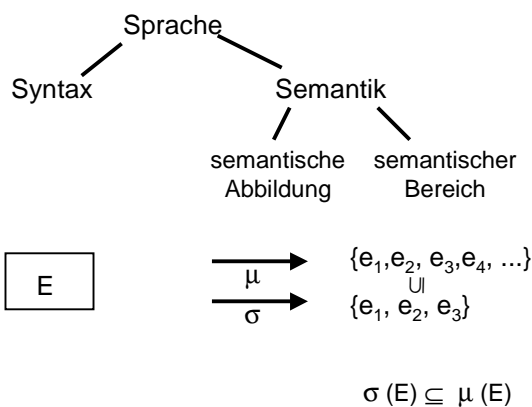
$\mu(E)$  Menge der *möglichen* Entities vom Typ E  
wird hier nicht festgelegt (etwa Menge isomorph zu natürlichen Zahlen)

$\sigma(E)$  Menge der *aktuellen* Entities vom Typ E in einem Zustand  $\sigma$   
( $\sigma$ , *Sigma*, für *state* (Zustand))

Aktuelle Entities müssen mögliche Elemente sein:  $\sigma(E) \subseteq \mu(E)$   
Ferner gefordert:  $\sigma(E)$  endlich



Erläuterung an konkreten Beispielen (1): Entity-Typen







ER-Modellierungskonzepte und ihre Semantik IV

mögliche Ausprägungen:

$$\mu(R) = \mu(E_1) \times \dots \times \mu(E_n)$$

aktuelle Beziehungen nur zwischen aktuellen Entities:

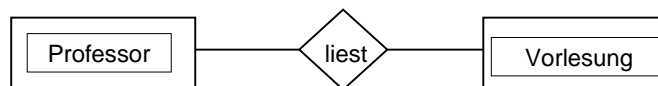
$$\sigma(R) \subseteq \sigma(E_1) \times \dots \times \sigma(E_n)$$

Rollennamen:

verheiratet(Frau : Person, Mann : Person)

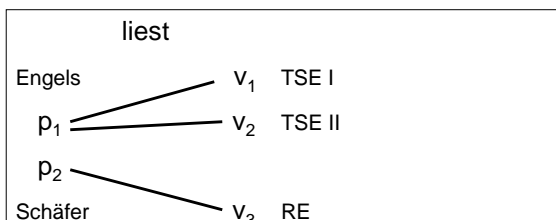


Erläuterung an konkreten Beispielen (3): Beziehungstypen



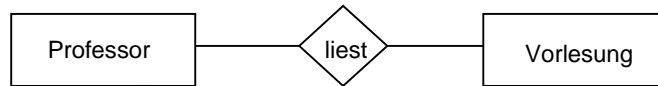
$$\text{liest} \xrightarrow{\mu} \mu(\text{Professor}) \times \mu(\text{Vorlesung})$$

$$\{p_1, p_2, p_3, \dots\} \times \{v_1, v_2, v_3, \dots\}$$





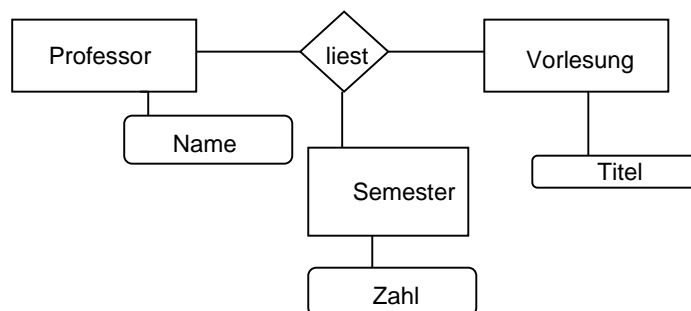
Erläuterung an konkreten Beispielen (4): Beziehungstypen



$$\begin{aligned} \sigma(\text{Professor}) \times \sigma(\text{Vorlesung}) &= \{(p_1, v_1), (p_1, v_2), (p_1, v_3), (p_2, v_1), \\ &\quad (p_2, v_2), (p_2, v_3)\} \\ \cup \\ \sigma(\text{liest}) &= \{(p_1, v_1), (p_1, v_2), (p_2, v_3)\} \end{aligned}$$



Erläuterung an konkreten Beispielen (5): mehrstellige Beziehungen

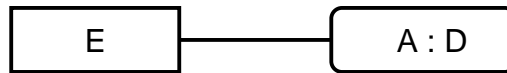


$$\begin{aligned} \sigma(\text{liest}) \subseteq \sigma(\text{Professor}) \times \sigma(\text{Vorlesung}) \times \sigma(\text{Semester}) \\ = \{(p_1, v_1, s_1), (p_1, v_1, s_2), \dots\} \end{aligned}$$



## ER-Modellierungskonzepte und ihre Semantik V

Attribute:

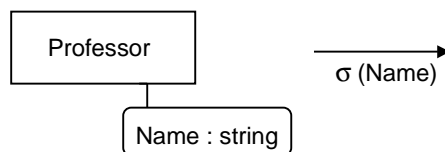


Semantik einer Attributdeklaration: Ein Attribut A eines Entity-Typen E ist im Zustand  $\sigma$  eine Abbildung

$$\sigma(A) : \sigma(E) \rightarrow \mu(D)$$



## Erläuterung an konkreten Beispielen (6): Attribute



$$\sigma(\text{Name}) : \sigma(\text{Professor}) \rightarrow \mu(\text{String})$$

$$\sigma(\text{Name}) : \{p_1, p_2\} \rightarrow C^*$$

$$\sigma(\text{Name})(p_1) = \text{Engels}$$

$$\sigma(\text{Name})(p_2) = \text{Schäfer}$$

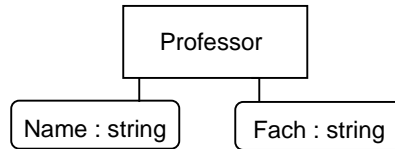
Erweiterung: neuer Zustand  $\sigma'$  mit Professor  $p_3$

$$\sigma'(\text{Professor}) = \{p_1, p_2, p_3\}$$

$$\sigma'(\text{Name})(p_3) = \text{Kastens}$$



## Erläuterung an konkreten Beispielen (7): Attribute



$$\sigma(\text{Fach}) \{p_1, p_2\} \rightarrow C^*$$

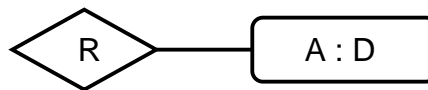
$$\sigma(\text{Fach}) (p_1) = \text{DB}$$

$$\sigma(\text{Fach}) (p_2) = \text{SWT}$$



## ER-Modellierungskonzepte und ihre Semantik VI

## Beziehungsattribute:



## Semantik:

$$\sigma(A) : \sigma(R) \rightarrow \mu(D)$$

## Textuelle Notation für Attribute:

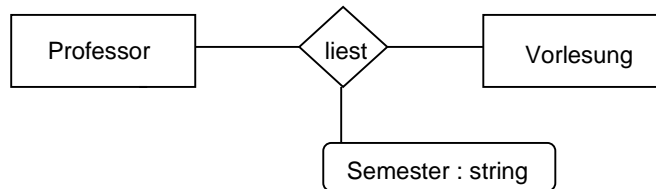
$$E(A_1 : D_1, \dots, A_m : D_m)$$

## bzw.

$$R(E_1, \dots, E_n; A_1, \dots, A_p)$$



## Erläuterung an konkreten Beispielen (8): Attribute von Beziehungstypen



$$\sigma(\text{Semester})(p_1, v_1) = \text{WS 00/01}$$

$$\sigma(\text{Semester})(p_2, v_3) = \text{SS00}$$


## Semantik eines ER-Schemas

*Jeder Zustand  $\sigma$  eines ER-Schemas ist eine Zuordnung*

$$E \longmapsto \sigma(E) \subseteq \mu(E)$$

$$R(E_1, \dots, E_n; \dots) \longmapsto \sigma(R) \subseteq \sigma(E_1) \times \dots \times \sigma(E_n)$$

$$E(\dots, A_i : D, \dots) \longmapsto \sigma(A_i) : \sigma(E) \rightarrow \mu(D), \dots$$

$$R(\dots; \dots, A_i : D, \dots) \longmapsto \sigma(A_i) : \sigma(R) \rightarrow \mu(D), \dots$$

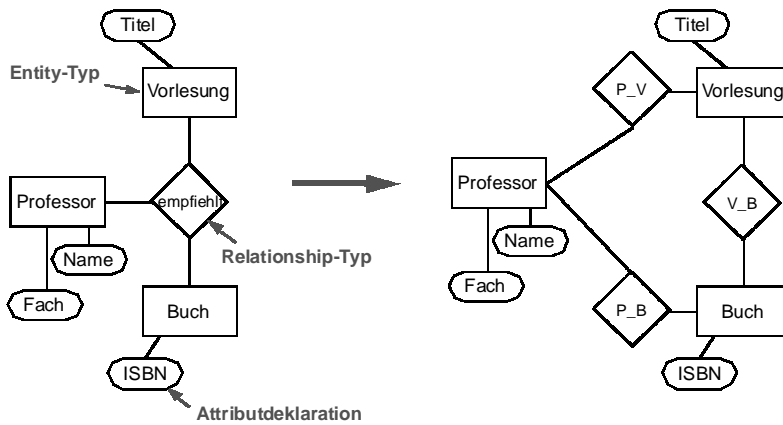
*bei gegebener fester Interpretation  $\mu$  der Datentypen durch Wertebereiche und der Entity-Typen durch vorgegebene Mengen möglicher Entities.*



Zweistellige versus mehrstellige Beziehungen

ER-Diagramm mit dreistelliger Beziehung

mögliche Umwandlung in zweistellige Beziehungen



Problem mit Umwandlung n-stelliger in 2-stellige Beziehungen:

empfiehl (korrekte Fassung)

Professor	Vorlesung	Buch (ISBN)
Heuer	DB 1	1-234...
Heuer	DB 2	9-876...
Saake	DB 1	9-876...
Saake	DB 2	9-876...

Ausprägungen der drei zweistelligen Beziehungen:

P_V		P_B		V_B	
Professor	Vorlesung	Professor	Buch	Vorlesung	Buch
Heuer	DB 1	Heuer	1-234...	DB 1	1-234...
Heuer	DB 2	Heuer	9-876...	DB 2	9-876...
Saake	DB 1	Saake	9-876...	DB 1	9-876...
Saake	DB 2				



**Problem mit Umwandlung n-stelliger in 2-stellige Beziehungen:**

empfeht (falsche, d.h. nicht beabsichtigte Fassung)

Professor	Vorlesung	Buch (ISBN)
Heuer	DB 1	1-234...
Heuer	DB 1	9-876...
Heuer	DB 2	9-876...
Saake	DB 1	9-876...
Saake	DB 2	9-876...



**Ausprägungen der drei zweistelligen Beziehungen:**

P\_V

Professor	Vorlesung
Heuer	DB 1
Heuer	DB 2
Saake	DB 1
Saake	DB 2

P\_B

Professor	Buch
Heuer	1-234...
Heuer	9-876...
Saake	9-876...

V\_B

Vorlesung	Buch
DB 1	1-234...
DB 2	9-876...
DB 1	9-876...



**Problem mit Umwandlung n-stelliger in 2-stellige Beziehungen:**

jetzt auch möglich:

P\_V

Professor	Vorlesung
Heuer	DB 1
Heuer	DB 2
Saake	DB 1
Saake	DB 2

P\_B

Professor	Buch
Heuer	1-234...
Heuer	9-876...
Saake	9-876...

V\_B

Vorlesung	Buch
DB 1	1-234...
DB 2	9-876...
DB 1	9-876...
<b>DB3</b>	<b>4-242...</b>

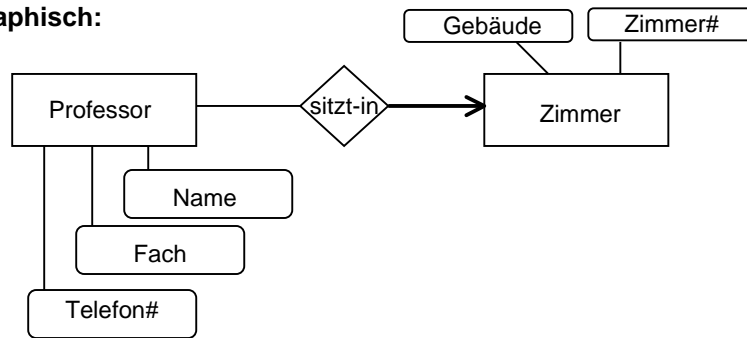


**Funktionale Beziehungen**

**Textuell:**

$$R : E_1 \rightarrow E_2$$

**Graphisch:**

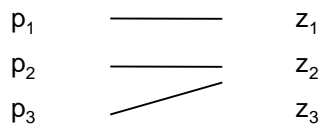
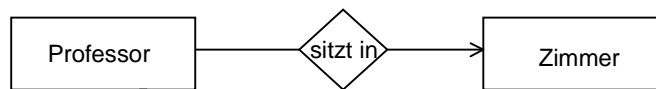


**Bedeutung:**

$$\sigma(R) : \sigma(E_1) \rightarrow \sigma(E_2)$$

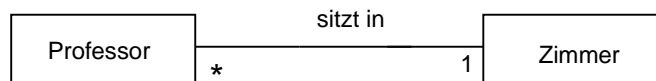


**Erläuterung an konkreten Beispielen (9): Funktionale Beziehungstypen**



$$\sigma(\text{sitzt in})(p_1) = z_1$$

UML-Notation:



**Identifizierung durch Schlüssel**

Für Entity-Typ  $E(A_1, \dots, A_m)$  sei Teilmenge  $\{S_1, \dots, S_k\}$  der gesamten Attribute gegeben, die **Schlüsselattribute**.

Es gilt:  $\{S_1, \dots, S_k\} \subseteq \{A_1, \dots, A_m\}$ .

In jedem Datenbankzustand identifizieren die aktuellen Werte der Schlüsselattribute eindeutig Instanzen des Entity-Typs E:

$$\forall e_1, e_2 \in \sigma(E) : (\sigma(S_1)(e_1) = \sigma(S_1)(e_2) \wedge \dots \wedge \sigma(S_k)(e_1) = \sigma(S_k)(e_2)) \Rightarrow (e_1 = e_2)$$

Notation: markieren durch Unterstreichung:

$$E(\dots, \underline{S}_1, \dots, \underline{S}_k, \dots)$$
**Erläuterung an konkreten Beispielen (10): Schlüsselattribute**

Professor	<u>Name</u>	Fach	Telefon#
	Engels	DB	3336
	Schäfer	SWT	3313
	Böttcher	DB	6600

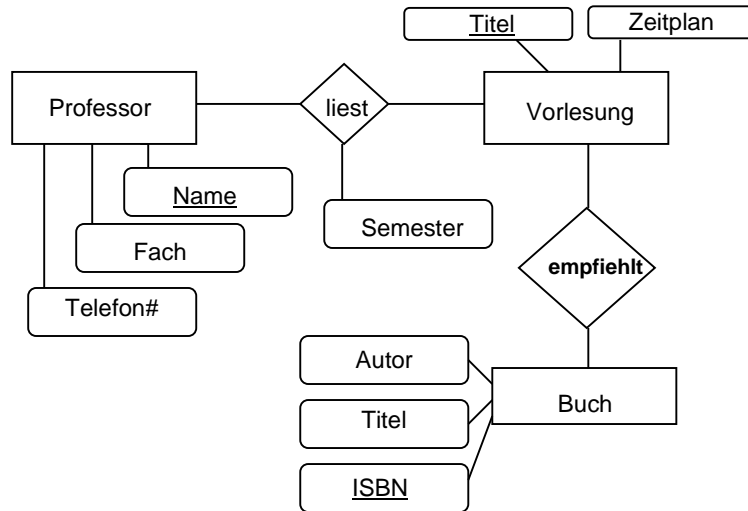
$$\{\text{Name}\} \subseteq \{\text{Name}, \text{Fach}, \text{Telefon}\}$$

$$p_1, p_2 \in \sigma(\text{Professor})$$

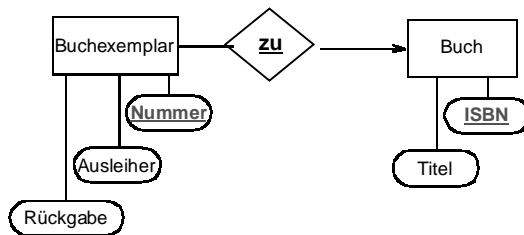
$$\sigma(\text{Name})(p_1) = \sigma(\text{Name})(p_2) \Rightarrow p_1 = p_2$$



Beispiel mit Schlüsselattributen



Abhängige Entity-Typen und Schlüsselattribute



- unterstrichene Attribute/funktionale Bez. sind die **Schlüssel** eines Entity-Typs:
  - ⇒ Buch hat den Schlüssel ISBN
  - ⇒ Buchexemplar hat den Schlüssel Nummer, zu -Verweis auf Buch
- Schlüsselattribute zweier Entities desselben Typs haben verschiedene Werte
- funktionale Beziehung als Schlüsselbestandteil legt **Existenzabhängigkeit** fest:
  - ⇒ Buchexemplar ohne zugehöriges Buch ist nicht erlaubt (= **weak Entity**)

**Ausprägungen abhängiger Entity-Typen:**

## Buchexemplar-Objekte

• Nummer: 1  
Ausleiher: Saake  
Rückgabe: 1.1.97

• Nummer: 2  
Ausleiher: Heuer  
Rückgabe: 1.1.99

• Nummer: 1  
Ausleiher: Heuer  
Rückgabe: 1.1.99

## Buch-Objekte

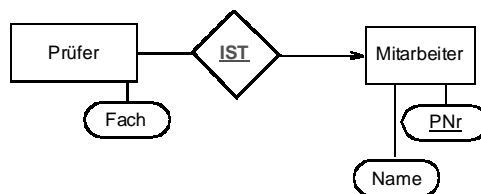
• ISBN: 1-234...  
Titel: OODB

• ISBN: 1-333...  
Titel: Zen for Computer Science

• ISBN: 1-345...  
Titel: Betriebssysteme

**Nicht erlaubt wäre z.B.:**

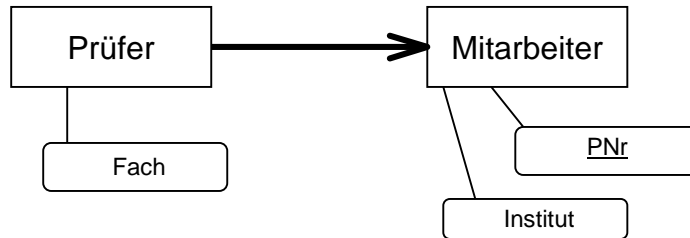
- beide Buchexemplare zu OODB-Titel haben dieselbe Nummer 1
- Buchexemplar hat mehr als ein oder kein zugehöriges Buch

**Die IST-Beziehung als Spezialfall einer funktionalen Beziehung**

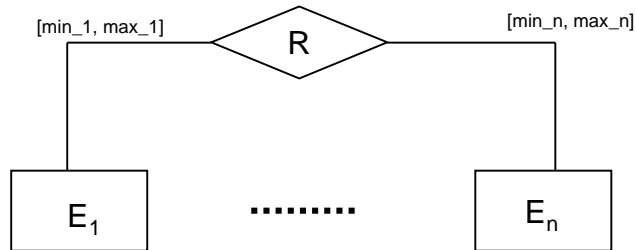
- jede Prüfer -Instanz ist genau einer Mitarbeiter -Instanz zugeordnet (Spezialfall eines abhängigen Entity-Typs)
- nicht jeder Mitarbeiter ist zugleich Prüfer, aber jeder Prüfer ist Mitarbeiter!
- IST-Beziehungstyp beschreibt also Mengeninklusion  $\sigma(E_1) \subseteq \sigma(E_2)$
- Stichwort: semantische Vererbung (siehe TSE I)
- Attribute des Entity-Typs Mitarbeiter treffen auch auf Prüfer zu („vererbte Attribute“): Prüfer(Name, PNr, Fach)
- nicht nur Attributdeklarationen, sondern auch konkrete Werte vererben sich



Die IST-Beziehung: Alternative Notation



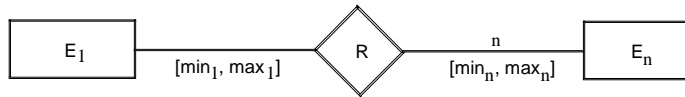
Kardinalitäten



- Notation für Kardinalitätsangaben an einem Beziehungstyp  
 $R(E_1, \dots, E_i[min_i, max_i], \dots, E_n)$
- Spezielle Wertangabe für  $max_i$  ist \*



## 2 verschiedene Semantikdefinitionen für Kardinalitäten



mit  $\min_i, \max_i \in 0, 1, \dots$  und \* für beliebig viele

□ **Semantikmöglichkeit 1 (für den Rest der Vorlesung) :**

die Anzahl von Beziehungen des Relationship-Typs R mit derselben Instanz  $e_i$  vom Typ  $E_i$  bewegt sich zwischen  $\min_i$  und  $\max_i$  :

$$\forall e_i \in E_i : \min_i \leq |\{ r \in R \mid r.E_i = e_i \}| \leq \max_i$$

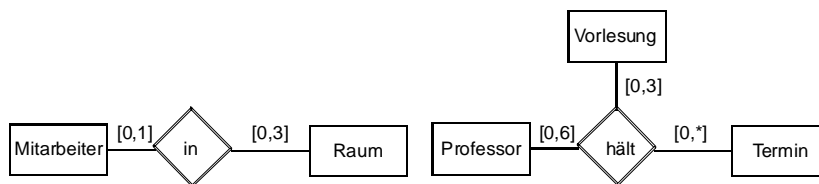
□ **Semantikmöglichkeit 2 :**

die Anzahl von Beziehungen des Relationship-Typs R, bei denen die Instanzen zu allen Entity-Typen bis auf  $E_i$  festgelegt sind, bewegt sich zwischen  $\min_i$  und  $\max_i$  :

$$\forall i \in 1, \dots, n : \forall (e_1, \dots, e_{i-1}, e_{i+1}, \dots, e_n) \in E_1 \times \dots \times E_{i-1} \times E_{i+1} \times \dots \times E_n : \min_i \leq |\{ r \in R \mid \forall j \in 1, \dots, n : j \neq i \wedge r.E_j = e_j \}| \leq \max_i$$



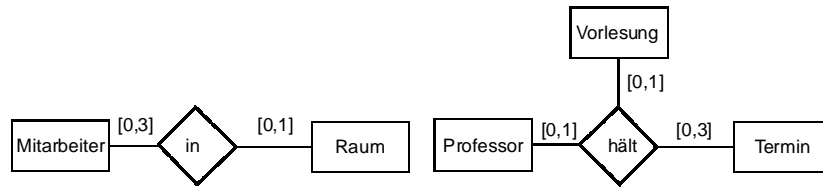
## Beispiele für Kardinalitätsangaben mit Semantikmöglichkeit 1



- ein Mitarbeiter arbeitet in höchstens einem Raum (der Uni)
- in einem Raum arbeiten höchstens 3 Mitarbeiter
- ein Professor hat maximal 6 Vorlesungstermine (pro Woche)
- eine Vorlesung wird an maximal drei Terminen (pro Woche) gehalten
- an einem Termin können beliebig viele Vorlesungen gehalten werden



Beispiele für Kardinalitätsangaben mit Semantikmöglichkeit 2

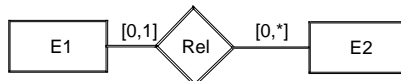


- ein Mitarbeiter arbeitet in höchstens einem Raum (der Uni)
- in einem Raum arbeiten höchstens 3 Mitarbeiter
- ein Professor hält an einem Termin höchstens eine Vorlesung
- eine Vorlesung wird an einem Termin höchstens von einem Professor gelesen
- ein Professor hält eine Vorlesung an maximal drei Terminen (pro Woche)

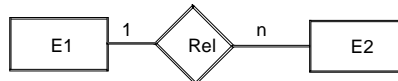


Vereinfachte Kardinalitätsangaben für binäre Beziehungen

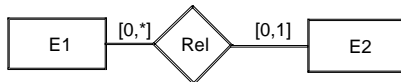
Kardinalitätsangabe nach Semantik 1:



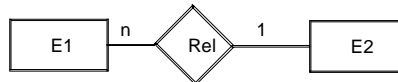
abkürzende Notation (Semantik 1):



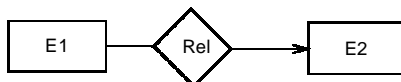
Kardinalitätsangabe nach Semantik 2:



abkürzende Notation (Semantik 2):



oder im ER-Modell ohne Kardinalitäten:



Fazit:

Vorsicht bei Kardinalitätsangaben in verschiedenen ER-Dialekten!

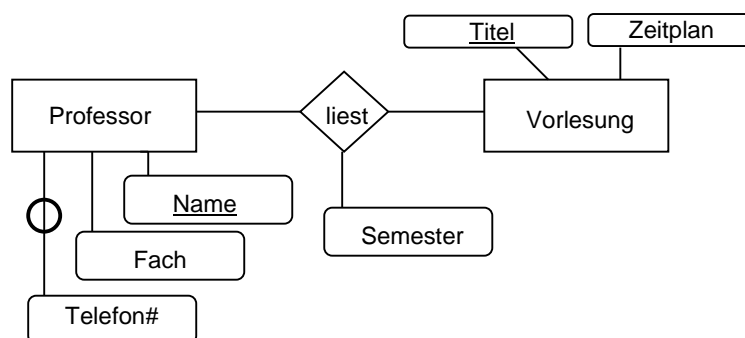


### Kardinalitäten

- [0,\*] ist Standardannahme.
- Die Angabe  $R(E_1[0,1], E_2)$  entspricht einer (partiellen) funktionalen Beziehung  $R : E_1 \rightarrow E_2$ , da jede Instanz aus  $E_1$  maximal einer Instanz aus  $E_2$  zugeordnet ist.
- Eine totale funktionale Beziehung wird durch  $R(E_1[1,1], E_2)$  modelliert.
- Für die Beziehung  $E_1$  IST  $E_2$  gilt:  $IST(E_1[1,1], E_2[0,1])$ .



### Optionalität von Attributen und Beziehungen

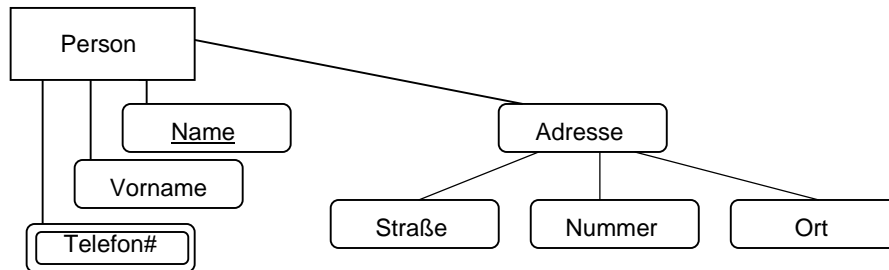


### Optionale Attribute im ER-Modell



**Weitere Konzepte**

**Strukturierte Attributwerte im ER-Modell**



**Strukturierte Werte im ER-Modell**

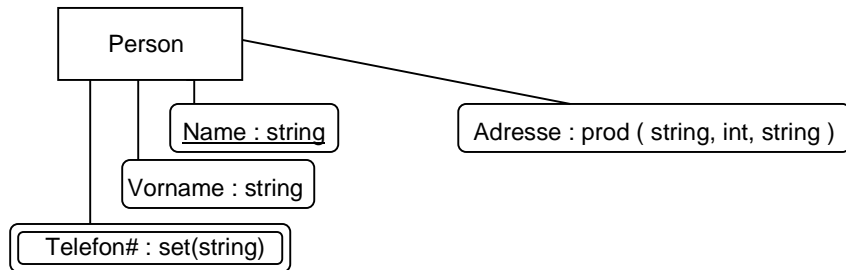
**Mengen- und tupelwertige Attribute im ER-Modell**

<b>prod:</b> Tupelbildung	<b>point = prod(real,real)</b>
<b>list:</b> Listen/Folgen von Werten	<b>polygon = list(point)</b>
<b>set:</b> Mengen	<b>holidays = set(date)</b>
<b>bag:</b> Multimengen	<b>birthdays_of_a_group = bag(date)</b>



Weitere Konzepte

□ Strukturierte Attributwerte im ER-Modell (alternative Notation)



Weitere Konzepte

□ Abgeleitete Attributwerte im ER-Modell

