

# Wegweisende Arbeiten in der Softwaretechnik

## Peter P. Chen Entity Relationship Modellierung



Problemseminarvortrag im SS 2004

von

Madlen Hartmann

# Übersicht

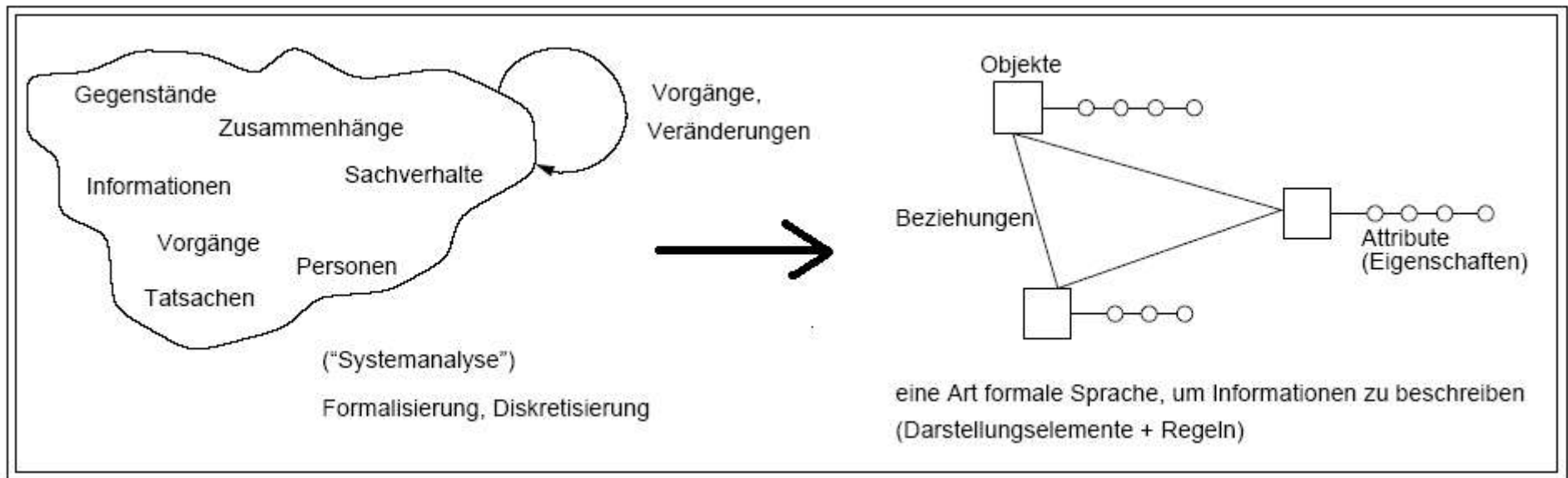
---

- 1. Einführung**
- 2. Hintergründe**
- 3. Wichtige Erkenntnisse für die Entwicklung des ERM**
- 4. Entity Relationship - Modell**
- 5. Zusammenfassung und Ausblick**

# 1. Einführung

**Datenmodell:** Abstraktion zur Beschreibung von Problemen

**Abstraktion:** Prozess des Auswählens bzw. Weglassens von Eigenschaften zur Problembeschreibung  
Wirklichkeitsausschnitt → Informationsmodell



**Entity Relationship Modell:** wichtiger Schritt im Design von Informationssystemen

# Übersicht

---

1. Einführung

**2. Hintergründe**

**Hierarchiemodell**

**Netzwerkmodell**

**Relationales Datenmodell**

3. Wichtige Erkenntnisse für die Entwicklung des ERM

4. Entity Relationship - Modell

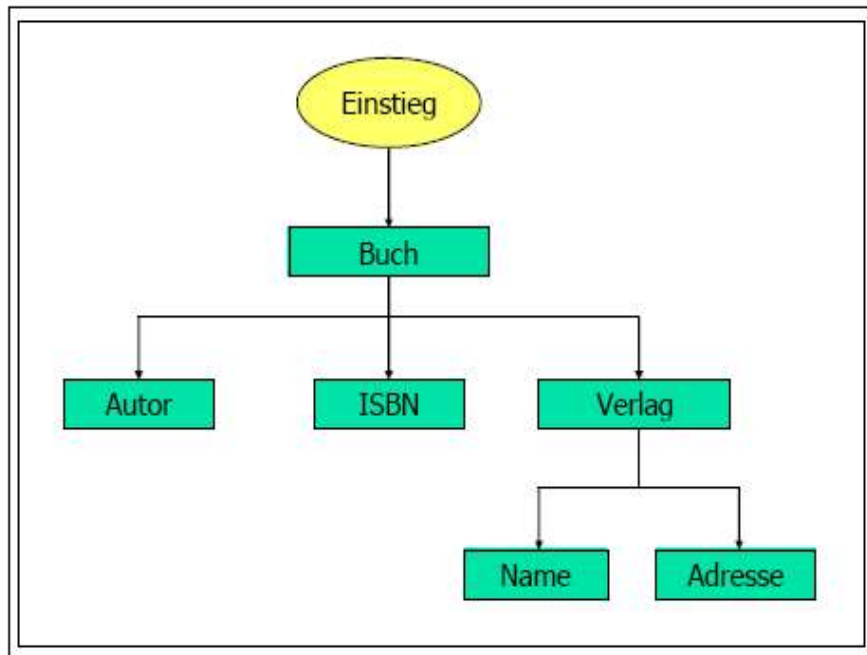
5. Zusammenfassung und Ausblick

## 2. Hintergründe - Hierarchiemodell

**A**nfang der 70er Jahre: verschiedene Datenmodelle,  
jedoch meist nur Nutzung von File-Systemen

### Datenmodelle (1)

- **H**ierarchiemodell: voneinander abhängende Tabellen bzw. Datensätze



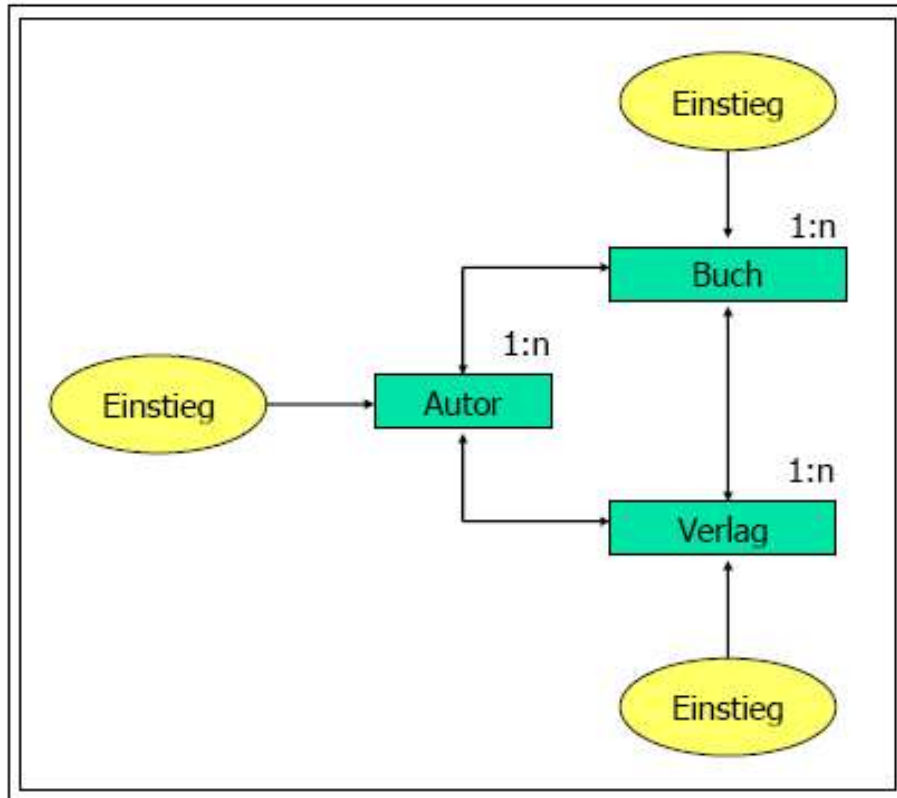
- Darstellung durch Bäume
- Wurzelobjekt und untergeordnete Objekte
- Beziehungen durch Verbindungslinien gekennzeichnet
- Einstieg nur über Wurzelobjekt
- schnelle Suche in Richtung der Hierarchie
- langsame Suche entgegen der Hierarchie

- Vorteile: effiziente, computergerechte Datenorganisation
- Nachteile: geringe Flexibilität (da Monohierarchie)  
Navigation nur bei Kenntnis des kompletten Baumes

## 2. Hintergründe - Netzwerkmodell

### Datenmodelle (2)

- **Netzwerkmodell:** polyhierarchische Darstellung der Informationsstruktur mit Netzwerken

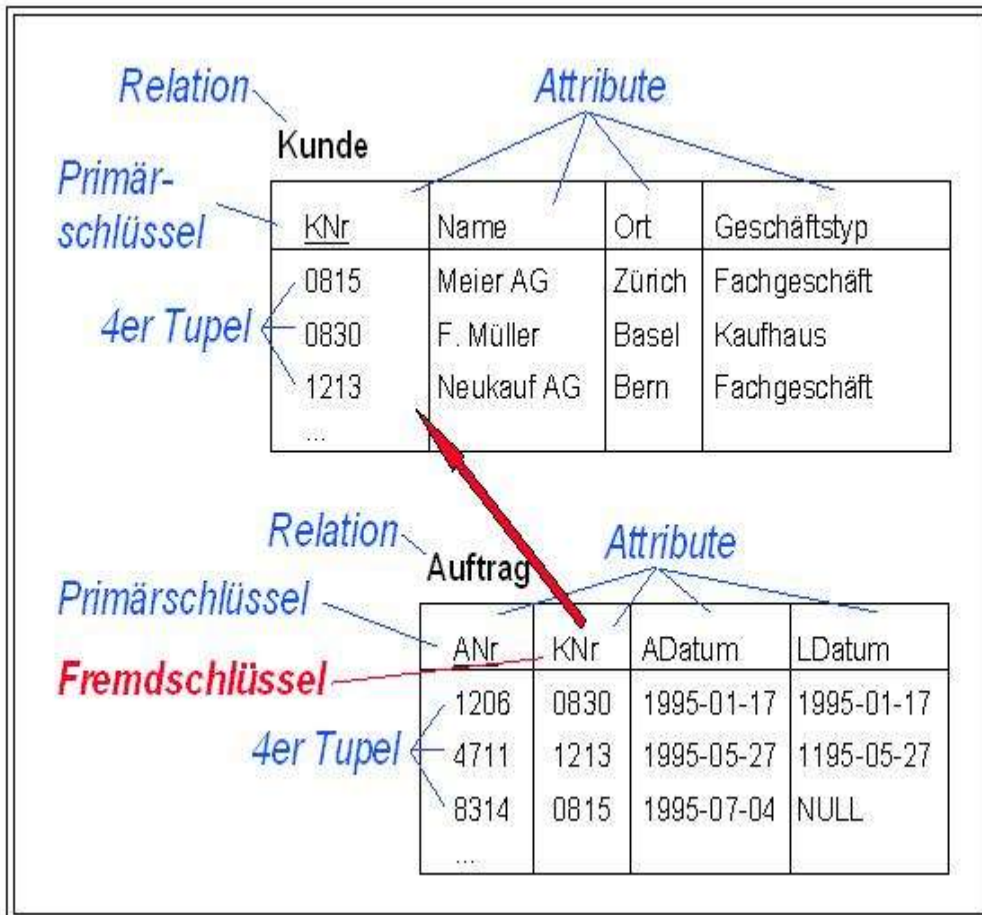


- keine Wurzelemente
- jeder Objekttyp kann Einstiegspunkt sein
- Vorteile: redundanzfreie Daten  
hohe Flexibilität

# 2. Hintergründe – Relationales Datenmodell

## Datenmodelle (3)

- **Relationales Datenmodell:**      Speicherung der Daten in Tabellen



- Zurückführung der Datenstrukturen des zu modellierenden Sachverhaltes auf Relationen
- Verwaltung der Daten in Form von Tupeln
- Zugriff auf jeden Wert durch Kombination aus Tabellennamen, Primärschlüssel und Spaltenname

## 2. Hintergründe – Definitionen

- **Definitionen**

- **R**elation:

- jede Tabelle ist eine Relation
    - gekennzeichnet durch Namen sowie Attribute, die miteinander in Beziehung stehen

- **A**tttribut:

- Eigenschaft einer Relation
    - Spalten der Relation

- **T**upel:

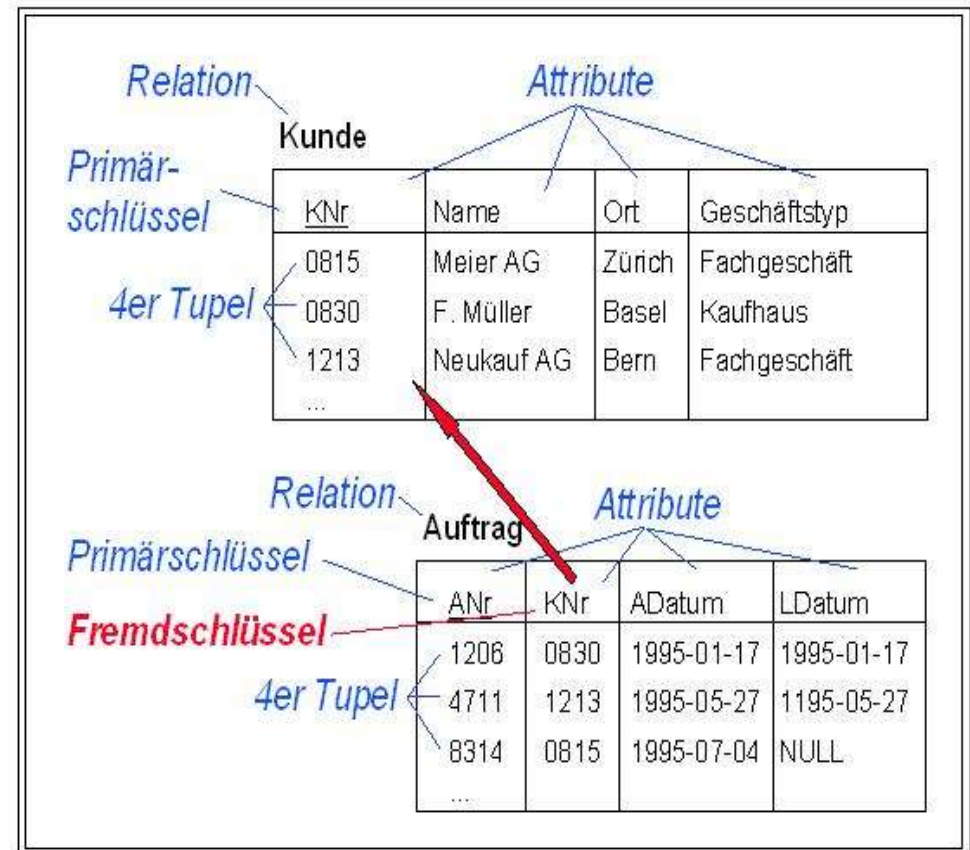
- Element der Relation
    - Reihenfolge in Tabelle beliebig

- **P**rimärschlüssel:

- eindeutige Identifikation eines Tupels
    - bestehend aus einem oder mehreren Attributen

- **F**remdschlüssel:

- Herstellung von Beziehungen zwischen Relationen
    - Attribut oder Attributkombination



# Übersicht

---

1. Einführung

2. Hintergründe

**3. Wichtige Erkenntnisse für die Entwicklung des ERM**

**Chinesische Schriftzeichen**

**Ägyptische Hieroglyphen**

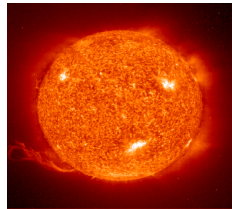
4. Entity Relationship - Modell

5. Zusammenfassung und Ausblick

# 3. Einflüsse auf die Entwicklung des ER-Modells (1)

## Chinesische Schriftzeichen

- Konzepte in Entwicklung und Evolution der chinesischen Schriftzeichen beziehen sich nah auf Modellierung von Dingen der realen Welt



<u>ursprüngliche Form</u>	<u>derzeitige Form</u>	<u>Bedeutung</u>
		Sonne
		Mond
		Person











- Komposition verschiedener Schriftzeichen zu neuem Zeichen möglich

	(Sonne)	+		(Mond)	=		(hell / Helligkeit durch Licht)
--	---------	---	--	--------	---	--	---------------------------------

# 3. Einflüsse auf die Entwicklung des ER-Modells (2)

## Ägyptische Hieroglyphen

- ähnlich wie chinesische Schriftzeichen

<u>Hieroglyph</u>	<u>Bedeutung</u>	<u>Hieroglyph</u>	<u>Bedeutung</u>
(a) 	langer Arm	(f) 	Mann
(b) 	Mund	(g) 	Frau
(c) 	Schlange	(h) 	Sonne
(d) 	Eule	(i) 	Haus
(e) 	Sieb	(j) 	Wasser

## Schlussfolgerungen aus Ähnlichkeit der Schriftzeichen

- trotz fehlender Kommunikation: Entwicklung ähnlicher Schriftzeichen
  - Menschen verfolgen ähnliche Gedanken
  - Darstellung von Dingen der realen Welt unterscheidet sich nicht
- Modellierung möglichst einfach und gut nachvollziehbar

1. Einführung

2. Hintergründe

3. Wichtige Erkenntnisse für die Entwicklung des ERM

**4. Entity Relationship - Modell**

**Definitionen**

**ER – Diagramm**

**Tabellarische Darstellung**

**Wahrung der Datenintegrität**

**Wahrung der Datenkonsistenz**

5. Zusammenfassung und Ausblick

# 4. Entity Relationship Modell (1976)

---

## Allgemein

- Formalisierung von Beziehungen zwischen Daten
- Standardmodellierungstechnik im Datenbank - Bereich
- zahlreiche Erweiterungen und grafische Darstellungsformen

## Darstellung des Betrachtungsbereiches in 3 Schritten

- Ermittlung von Entities und Relationships und Zuordnung von Attributen
- Klassifikation der Entities und Relationships zu Entitymengen und Relationshipmengen
- Darstellung des Modells als ER-Diagramm

Anschließend Umsetzung des ER-Diagramms in relationalen (tabellarischen) Entwurf

# 4.1 Einführung wichtiger Begriffe (1)

- Welt besteht aus abgrenzbaren Objekten (= Entities), zwischen denen Beziehungen (= Relationships) bestehen
- oft verschiedene Modellierungsmöglichkeiten eines Sachverhaltes

## Entity und Entymenge

- Zusammenfassung von **Entities**  $e_i$  mit gleichen Eigenschaften zu **Entymenge**  $E_j$
- Datenbank enthält endlich viele Entymengen  $E_1, E_2, E_3 \dots E_n$
- *Beispiele:*

<i>Entities</i>	<i>Informatik, Biologie, Peter Chen</i>
<i>Entymengen</i>	<i>Studienfach, Forscher</i>
- schwache Entymenge = Entymenge mit Existenzabhängigkeit zu anderer

## Relationship, Rolle und Relationshipmenge

- Zusammenfassung gleichartiger **Relationships**  $r_i$  zwischen Entities zu **Relationshipmengen**  $R_j$
- *Beispiel:* *Relationshipmenge Vorlesung*
- R ist mathematische Relation zwischen Entities:  $R \subseteq E_1 \times E_2 \times \dots \times E_n$   
wobei  $R = [r_1, r_2, \dots, r_i]$  mit  $r_i = \{[e_1, e_2, \dots, e_n] \mid e_1 \in E_1, e_2 \in E_2, \dots, e_n \in E_n\}$
- Festlegung von **Rollen** für Entities in Relationship möglich  
*Beispiel: Ehemann und Ehefrau – Rollen im Rs „HEIRATEN“*

## 4.1 Einführung wichtiger Begriffe (2)

### Attribut, Wert und Wertemenge

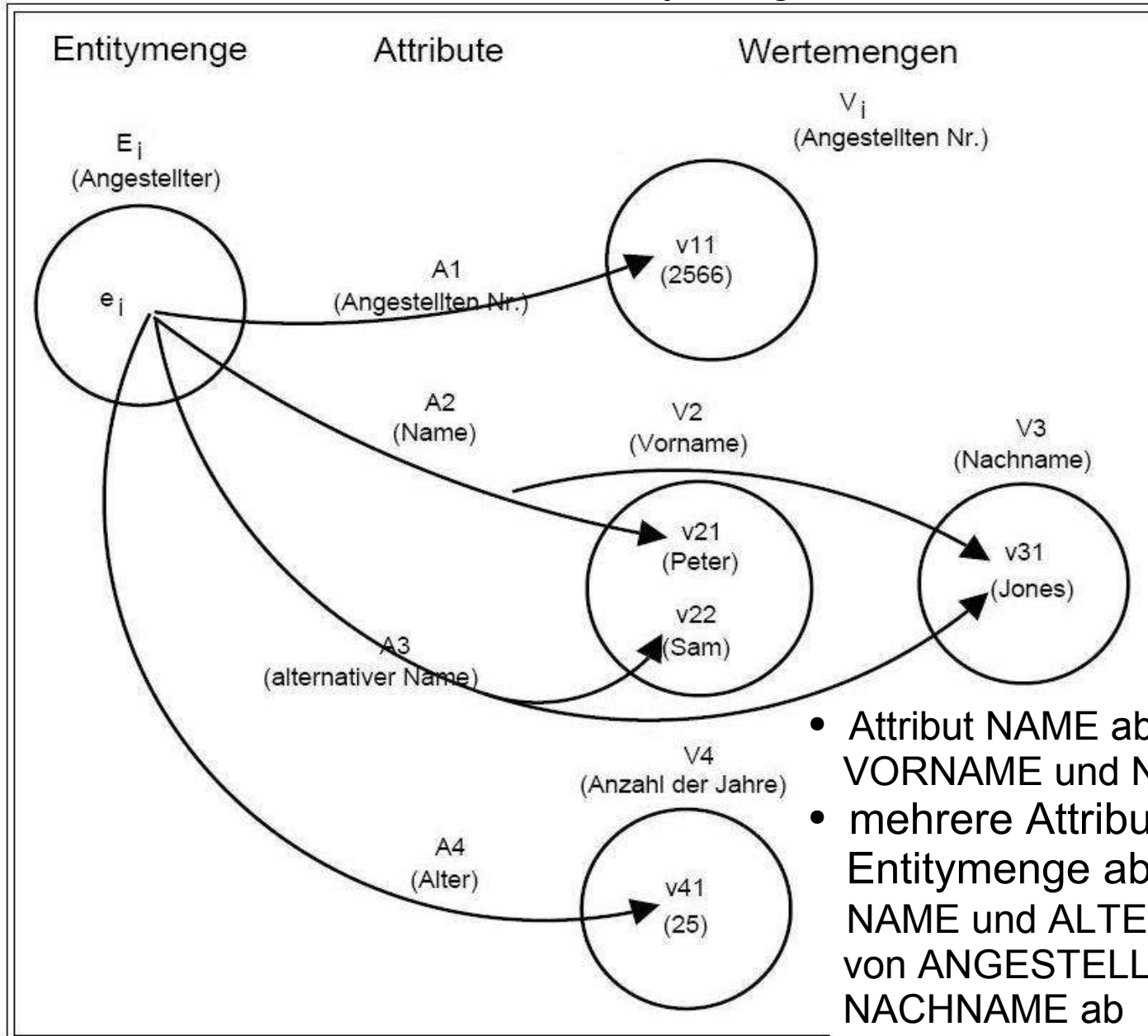
- Eigenschaften von Entities werden mit **Attribut-Werte** Paaren ausgedrückt
- Werte werden in verschiedene Mengen eingeteilt (=Attribute)
  - „rot“ ist ein Wert, „Farbe“ die zugehörige Wertemenge
- Wert einer Menge kann äquivalent zu Wert einer anderen Menge sein
  - Wert „1“ der Menge „Byte“ äquivalent zu Wert „8“ der Menge „Bit“
- Definition eines Attributes  $a_i$  als Abbildungsfunktion
  - von Entitymenge in **Wertemenge**  $f : E_i \rightarrow V_i$
  - von Relationshipmenge in **kartesisches Produkt von Wertemengen**  
$$f : R_i \rightarrow (V_{i1} \times V_{i2} \times \dots \times V_{in})$$

### Primärschlüssel

- Schlüsselkandidaten sind Attribute zur eindeutigen Identifikation von Entities
- bei Existenz mehrerer Schlüsselkandidaten wird semantisch bedeutungsvoller ausgewählt
  - Primärschlüssel
- jeder Wert des Primärschlüssels nur einmal vorhanden

# Beispiel

- **A**tribute definiert auf einer Entitymenge



- Attribut NAME abgebildet in Wertemengen VORNAME und NACHNAME
- mehrere Attribute können von einer Entitymenge abbilden:  
NAME und ALTERNATIVER NAME bilden von ANGESTELLTER in VORNAME und NACHNAME ab

## 4.2 ER–Diagramm: Symbole (1)

- grafische Dokumentation der Entities und Relationships

### Symbole (1)

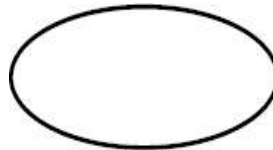
- Entity / Entitymenge



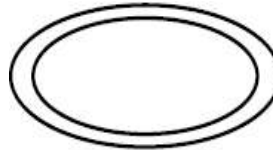
- schwaches Entity / Entitymenge



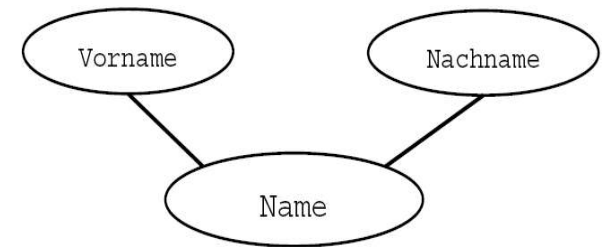
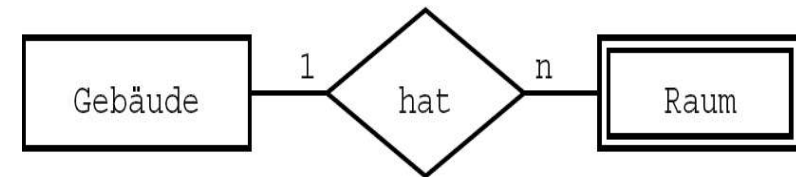
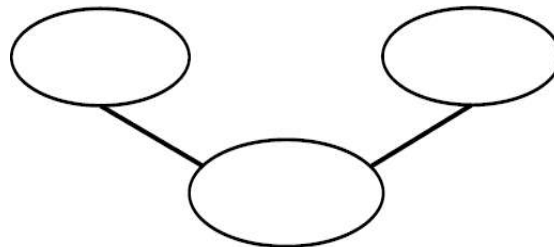
- einfaches/einwertiges Attribut



- mehrwertiges Attribut



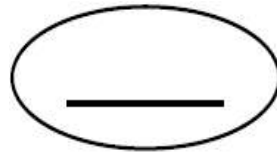
- zusammengesetztes Attribut



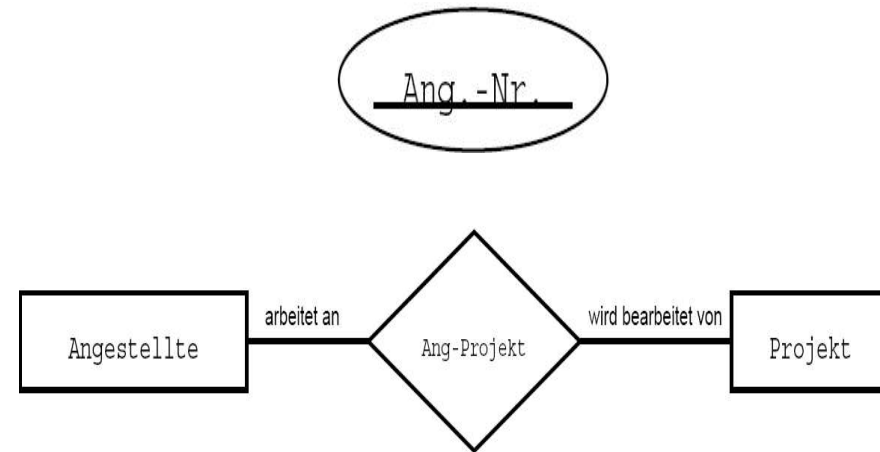
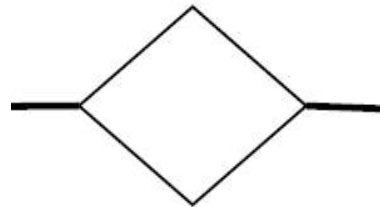
## 4.2 ER-Diagramm: Symbole (2)

### Symbole (2)

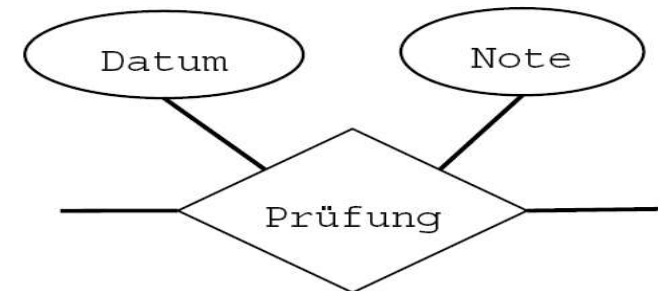
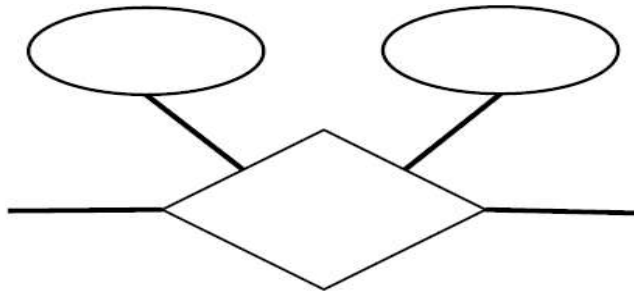
- Primärschlüssel



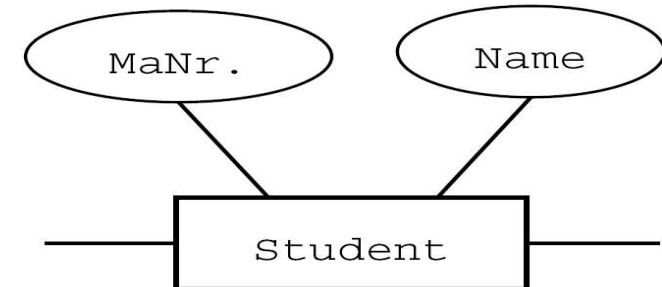
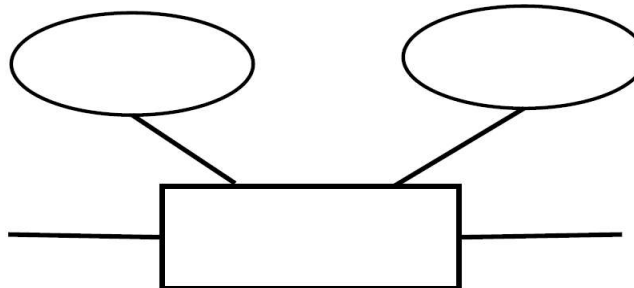
- Relationshipmenge



- Attribute von Relationships bzw. Relationshipmengen



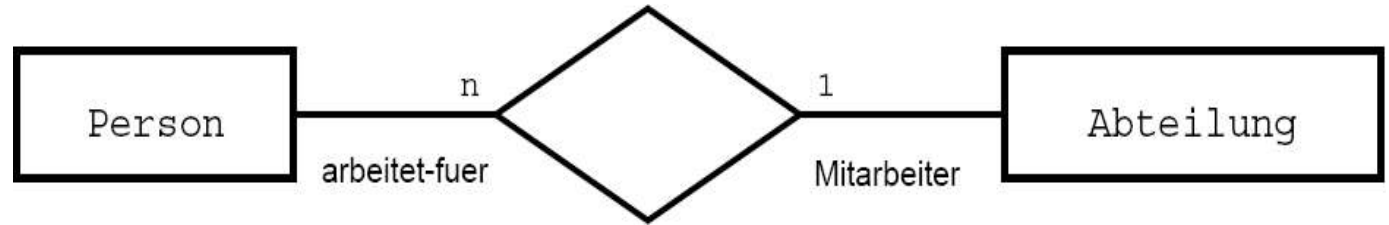
- Attribute von Entities bzw. Entitymengen



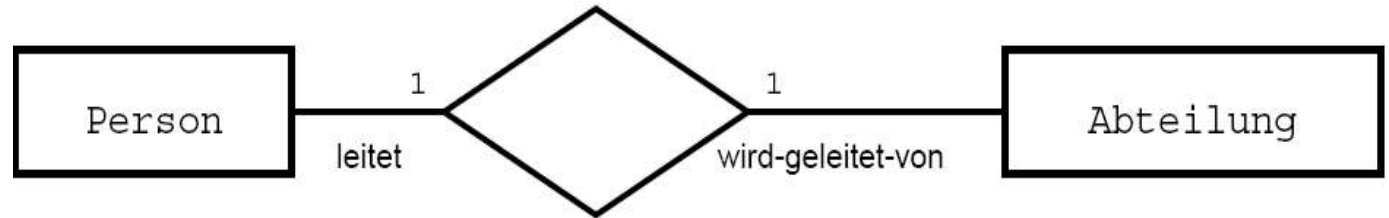
# 4.2 ER-Diagramm: Kardinalitäten

## Kardinalitäten

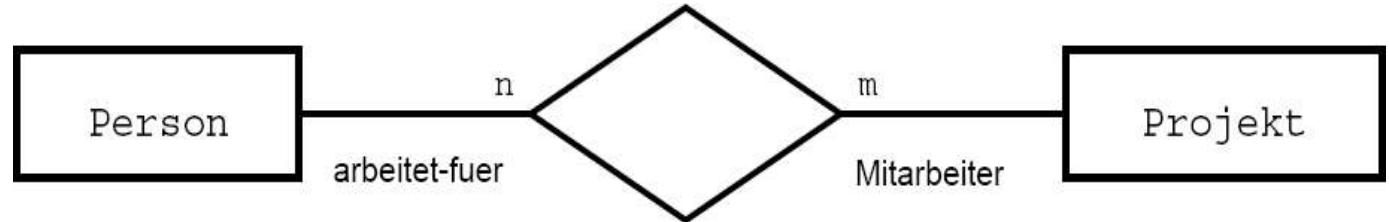
- n:1 – Beziehung



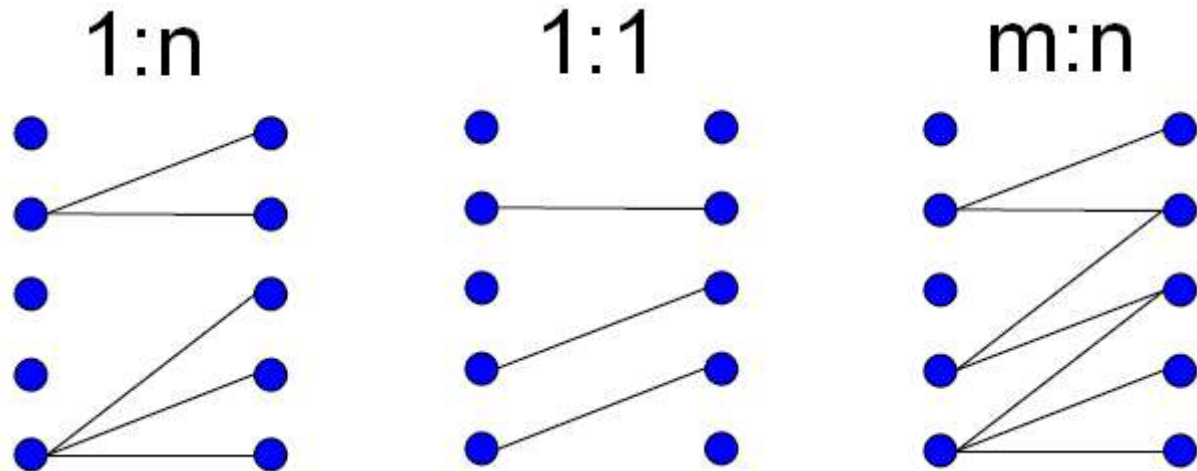
- 1:1 – Beziehung



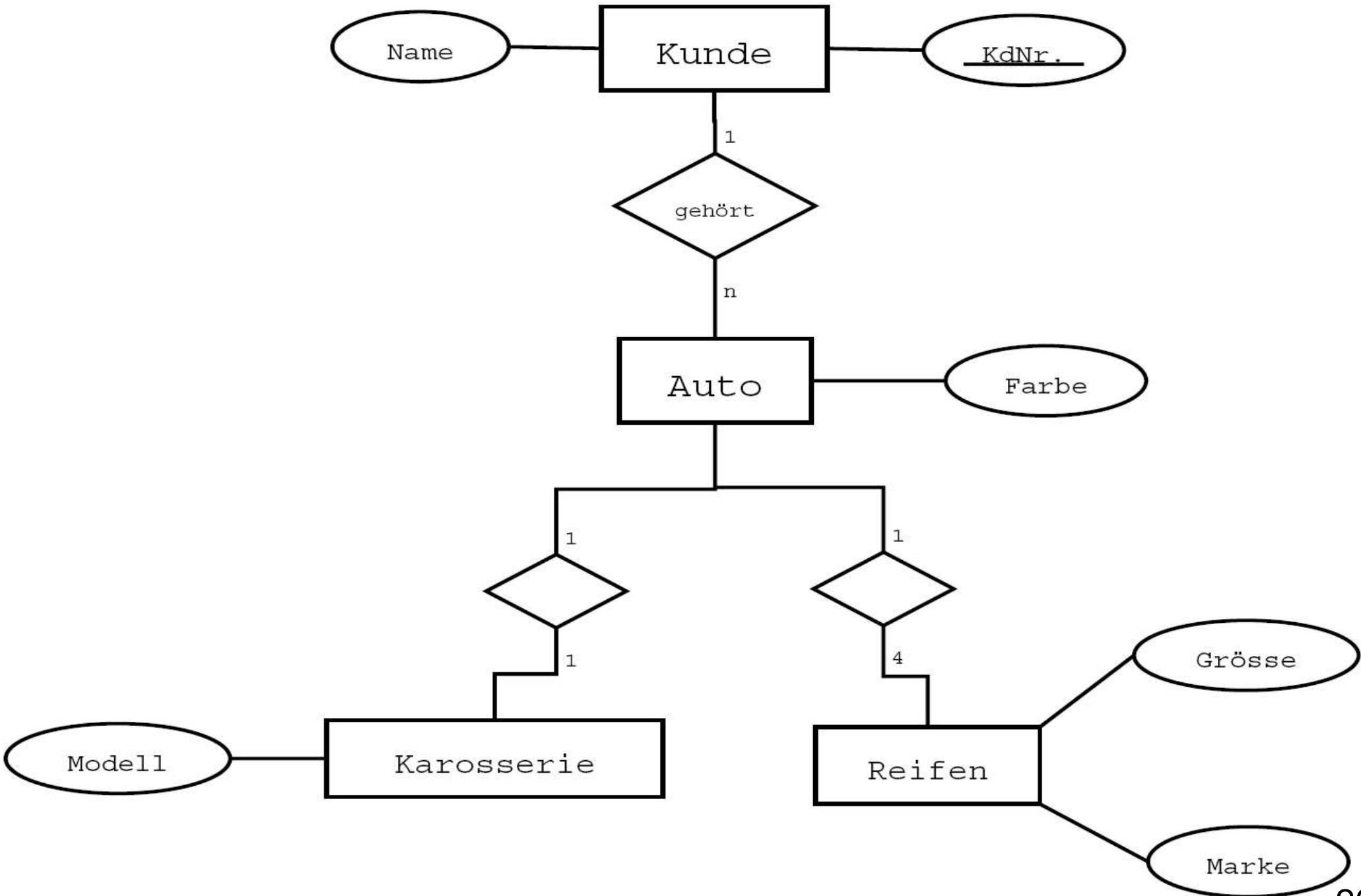
- n:m – Beziehung



## Darstellung als Graphen



## 4.2 ER-Diagramm: Beispiel



## 4.3 Tabellarische Darstellung (1)

---

### Darstellung von Entities und Entitymengen

$E_1$
$a_1$
$a_2$
$a_3$

- Entitymenge  $E_1$  , Entities der Entitymenge  $a_1(E_1), a_2(E_1), a_3(E_1)$
- Zuordnung der Entites zu den Entitymengen

## 4.3 Tabellarische Darstellung (2)

### Darstellung von Relationships

- binärer Relationshiptyp

$R$	$E_1$	$E_2$
	$a_1$	$b_1$
	$a_2$	$b_2$
	$a_3$	$a_3$
	$a_4$	$a_4$

- Relationship  $R$ , Entitymengen  $E_1, E_2$  mit Entities  $a_1, a_2, a_3, a_4$  und  $b_1, b_2$

Arbeitet_fuer	Angestellter	Projekt
	Kunze	DB-1
	Heine	DB-2
	Schulze	DB-1
	Bauer	DB-2

- ternärer Relationshiptyp

Liefert_fuer	Lieferant	Teil	Produkt
	Firma_X	Felge_c	Villiger_8
	Firma_Y	Lenker_a	Kettler_b
	Firma_Z	felge_b	BMW_z3

## 4.3 Tabellarische Darstellung (3)

### Darstellung von Entityattributen

$E$	$A_1$	$A_2$	$A_3$
$e_1$	$w_1$	$w_5$	$v_3$
$e_2$	$w_2$	$w_6$	$v_1$
$e_3$	$w_3$	$w_1$	$v_2$
$e_4$	$w_4$	$w_2$	$v_1$

- Entitymenge  $E$ , Attribute  $A_1, A_2, A_3$  mit Werten  $w_1, \dots, w_6$  bzw.  $v_1, \dots, v_3$  für Entities  $e_1, \dots, e_4$

Fahrrad	Farbe	Rahmengröße	Radgröße
Fahrrad_1	Rot	50cm	24"
Fahrrad_2	Blau	60cm	28"
pauls_Fahrrad	Grün	58cm	28"
sams_Fahrrad	Pink	64cm	28"

## 4.3 Tabellarische Darstellung (4)

Darstellung von Primärschlüsselattributen

$E$	$\underline{A_1}$	$A_2$	$A_3$
	$w_1$	$w_5$	$v_3$
	$w_2$	$w_6$	$v_1$
	$w_3$	$w_1$	$v_2$
	$w_4$	$w_2$	$v_1$

- eindeutige Identifizierung von Entities
- Bezeichnung der Entities nicht mehr nötig

Mitarbeiter	<u>Pnr</u>	Name	Abteilung
	1234	Meier	12
	1256	Müller	15
	1278	Meier	13
	1290	Kurze	14

## 4.3 Tabellarische Darstellung (5)

### Darstellung von Relationshipattributen

$R$	$E_1$	$E_2$	$R-A_1$	$R-A_2$	...	$R-A_n$
	$e_{11}$	$e_{21}$	$w_{11}$	$w_{21}$	...	$w_{n1}$
	$e_{12}$	$e_{22}$	$w_{12}$	$w_{22}$	...	$w_{n2}$
	$e_{13}$	$e_{23}$	$w_{13}$	$w_{23}$	...	$w_{n3}$
	$e_{14}$	$e_{24}$	$w_{14}$	$w_{24}$	---	$w_{n4}$

- Doppellinie zur Abgrenzung der Relationshipattribute von den Entities

<i>Arbeitet_an</i>	<i>Mitarbeiter</i>	<i>Projekt</i>	<i>Zeitungfang</i>	<i>Ort</i>
	Hans	DBMS	Vollzeit	Leipzig
	Paul	DBMS	Teilzeit	Leipzig
	Karl	DBMS2	Teilzeit	Berlin
	Ursula	DBMS	Vollzeit	Leipzig

# 4.4 Wahrung der Datenintegrität

---

- wichtig bei DB-Entwurf: Wahrung der Konsistenz und Integrität von Daten

## Datenintegrität

- logische Korrektheit, Gültigkeit und Genauigkeit von Daten in Datenbanken
  - korrekte Beziehungen von Datenobjekten untereinander
  - korrekte Wertebereiche von Daten
    - Einschränkungen auf zulässige Werte der Wertmengen.  
*z.B.: negative Altersangabe nicht zulässig*
    - Einschränkungen auf erlaubte Werte der Wertmengen.  
*z.B.: Einschränkung des Alters von Angestellten (20 bis 65)*
- keine redundanten Daten gespeichert
- spezifische Integritätsformen
  - Entitätenintegrität
    - Definition von PS, die nur einmal in Tabelle vorkommen dürfen
  - referentielle Integrität
    - Fremdschlüssel einer Tabelle muss in anderer Tabelle PS sein

# 4.5 Wahrung der Datenkonsistenz

---

## Datenkonsistenz

- auch Transaktionskonsistenz
- konsistente Daten sind in sich stimmig und widerspruchsfrei
- Gewährleistung, dass alle Manipulationsschritte einer Transaktion entweder ganz oder gar nicht durchgeführt werden
- Manipulationsschritte
  - Einfügen
    - Einfügen eines Entities ohne Probleme
    - Einfügen eines Relationship erzwingt Überprüfung der in Verbindung stehenden Entities
    - Einfügen von Attributen erzwingt die Überprüfung der Zulässigkeit der Werte
  - Löschen
    - Löschen eines Entity bedingt das Löschen aller abhängenden Entities, Relationships an denen das Entity beteiligt ist und Attribute des gelöschten Entity
    - Löschen eines Relationships bedingt das Löschen der Attribute des Relationship
  - Aktualisieren
    - Änderungen von Werten bedingt das Überprüfen der Zulässigkeit der geänderten Werte

# Übersicht

---

1. Einführung

2. Hintergründe

3. Wichtige Erkenntnisse für die Entwicklung des ERM

4. Entity Relationship - Modell

**5. Zusammenfassung und Ausblick**

**Erweitertes ER-Modell (EERM)**

**UML**

**Von ERM zu UML**

# Zusammenfassung und Ausblick

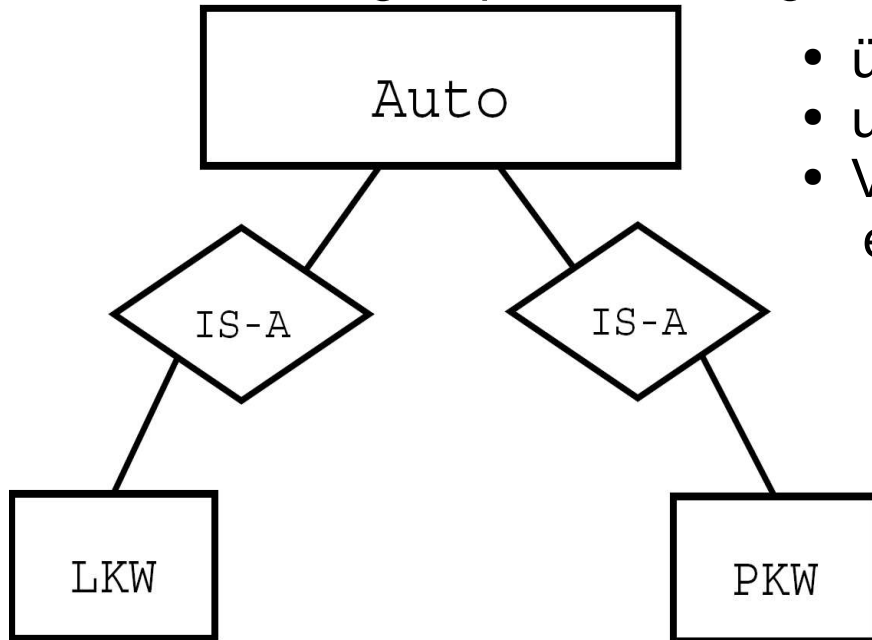
---

- ER-Modell heute Standard in DB-Modellierung
- Anfang eines konzeptionellen Entwurfs einer DB
  - Erfassung der grundlegenden Objekte und Beziehungen zwischen diesen
- einfacher Zeichenvorrat
  - ermöglicht Kommunikation mit Nicht-Fachleuten
  - schafft allgemeine Verständnisbasis zwischen Auftraggeber und Entwickler
- Überführung des ER-Modells in relationales DB-Modell
  
- verschiedene **W**eiterentwicklungen des ER-Modells
  - das Extended Entity Relationshipmodel (EER) von Atzeni und Chen,
    - beinhaltet Aggregation sowie Generalisierung/Spezialisierung
  - das Entity-Category-Relationship-Modell (ECR) von Elmasri,
  - das Strukturierte Entity-Relationship-Modell (SERM) von Sinz,
  - das ER+ Model von Hull und King,
  - das IDEFIX der US Air Force, sowie
  - das Semantically Enriched Extended Entity Relationship Model (E<sup>3</sup>R) von Jeckle

# Erweitertes ER-Modell

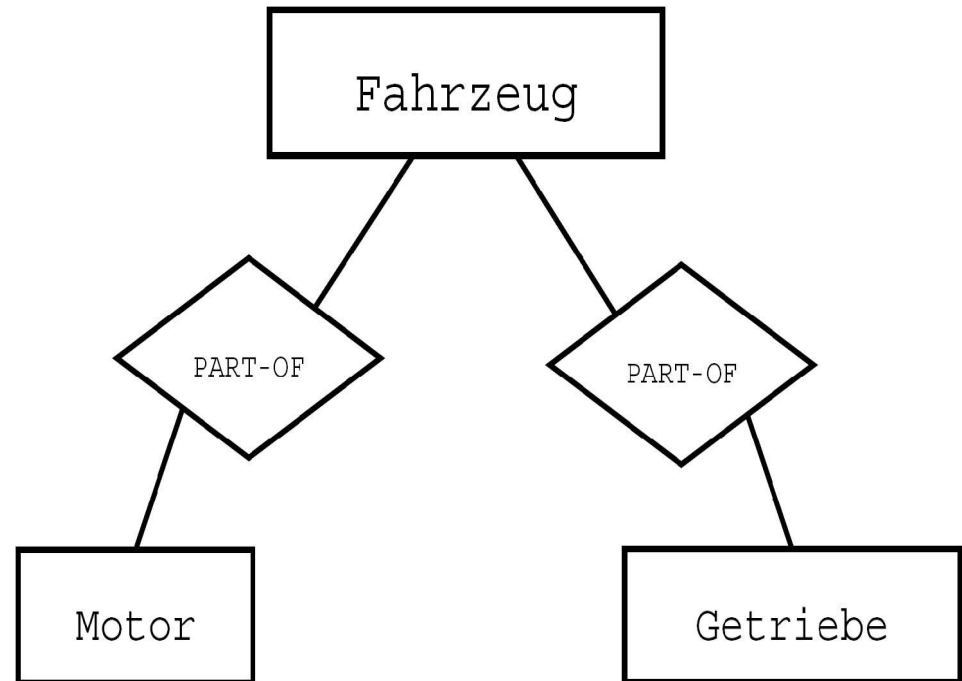
- ER-Modell wurde von Atzeni und Chen erweitert um
  - Generalisierung/Spezialisierung
  - Aggregation

## Generalisierung / Spezialisierung



- übergeordnete Menge: Generalisierungstyp
- untergeordnete Menge: Spezialisierungstyp
- Vererbungskonzept: alle Spezialisierungstypen erben Attribute von Generalisierungstyp

## Aggregation



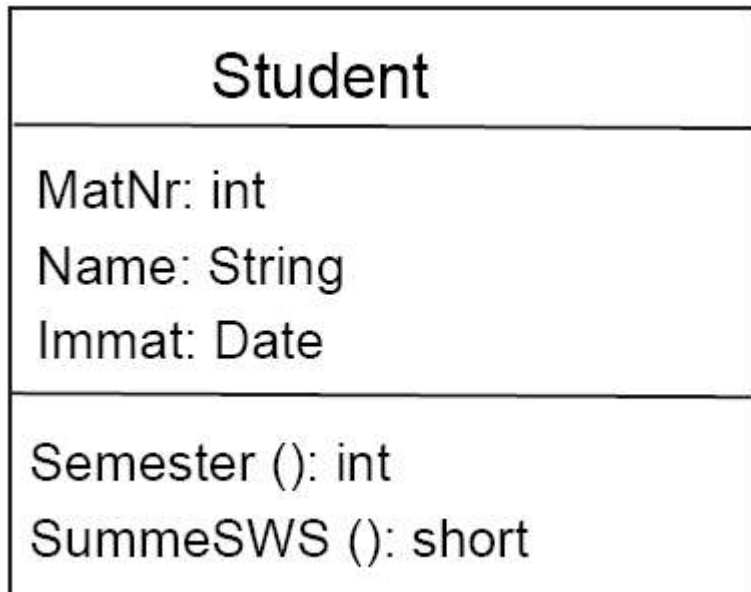
- Entitätsmenge ist aus einzelnen Entitäten anderer Entitätsmengen zusammengesetzt

# Vom ERM zu UML

- Ansätze zu objektorientierter Modellierung und Entwurf zu Beginn der 90er Jahre
  - OMT (Object Modelling Technique) von Rumbaugh
- Vereinheitlichung zu UML (Unified Modeling Language) durch Booch, Jacobson, R.
  - Erweiterung der Grundkonzepte der Objektorientierung (Klassen, Vererbung ...) um Konzepte des EERM





## UML

- Klasse: Definition der grundlegenden Eigenschaften von Methoden sowie Attribute des Objekts
  - UML - Klassendiagramm



# Vom ERM zu UML (2)

---

- Assoziation von ER-Modell übernommen (Relationships)  
UML-Diagramm: einfache Linie 
- Aggregation von EER-Modell übernommen („PART-OF“-Beziehung)  
UML-Diagramm: unausgefüllte Raute 
- Komposition: Form der Aggregation, Teilobjekte gehören nur zu einem Ganzen  
UML-Diagramm: gefüllte Raute 
- Vererbung  
UML-Diagramm: unausgefüllter Pfeil 

Einfachvererbung: aus EER-Modell übernommen („IS-A“-Beziehung)  
—▶ eine Oberklasse vererbt mehreren Unterklassen Attribute und Methoden

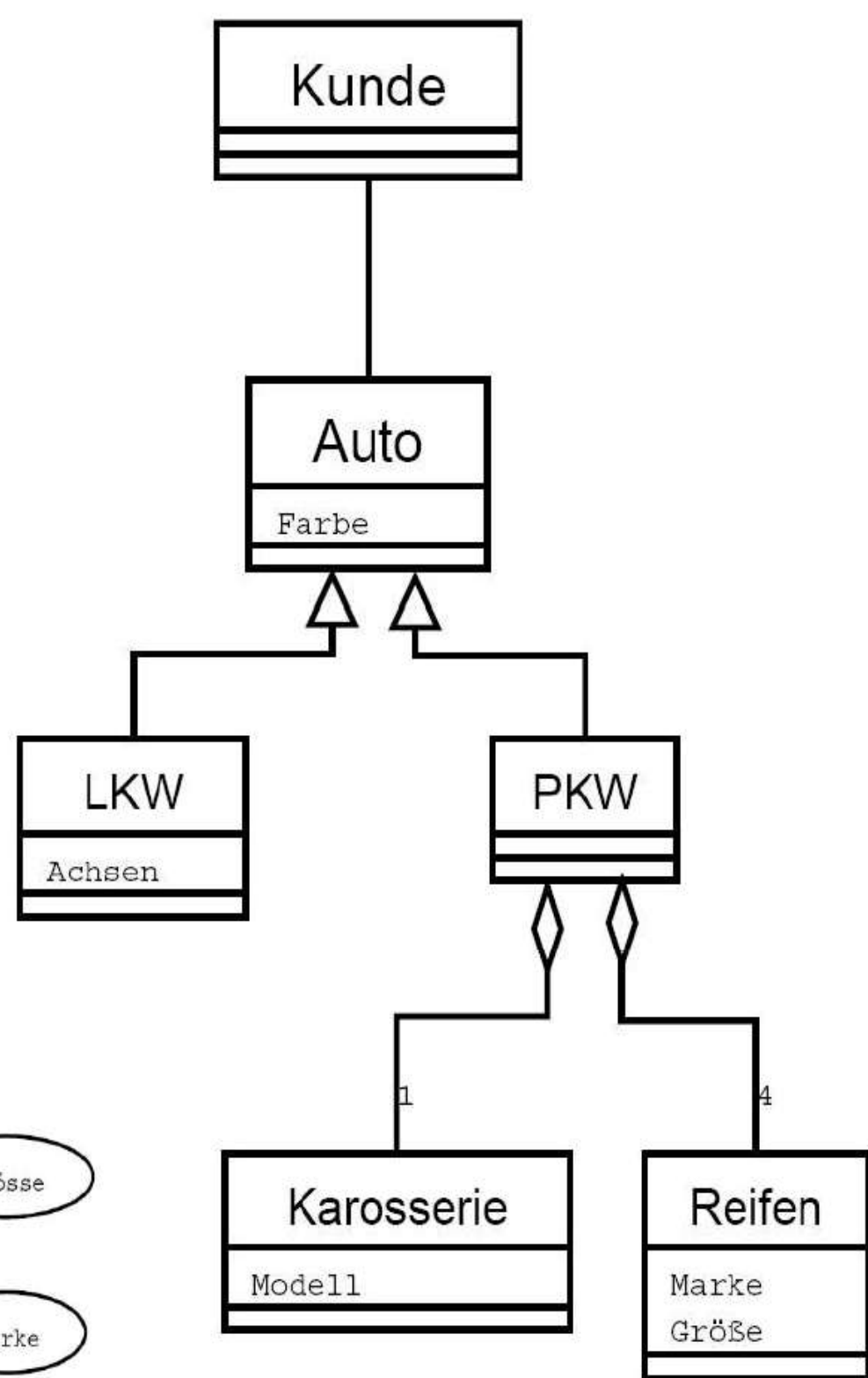
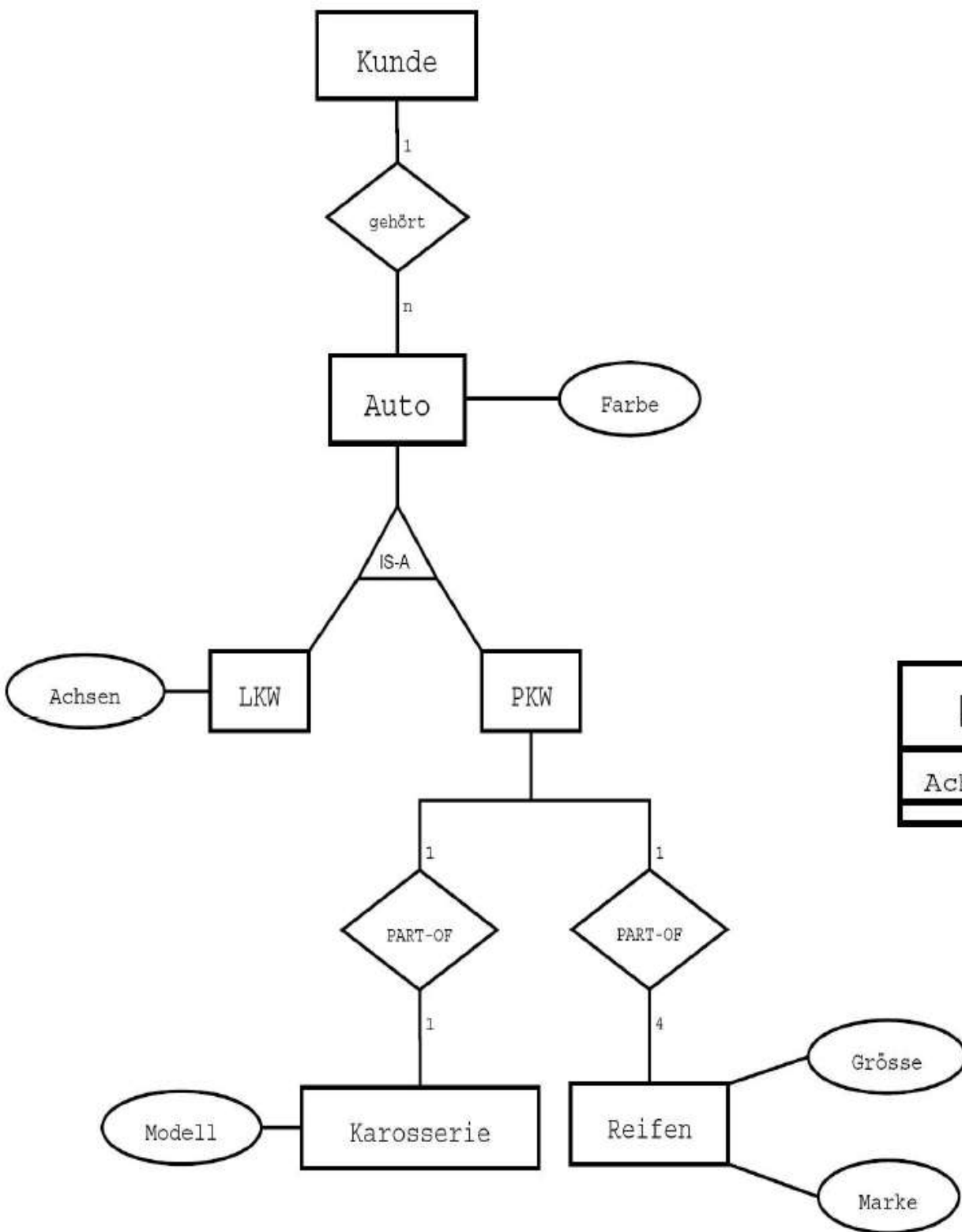
Mehrfachvererbung: nicht im ER-Modell  
—▶ mehrere Oberklassen vererben einer Unterklasse Attribute und Methoden

# Vom ERM zu UML - Beispiel

---

## Überführung ER - Modell in UML – Diagramm

- Umwandlung aller Entities und Entitymengen in Klassen
- Attribute in Klassendefinition aufnehmen
- Darstellung der Relationships als Assoziation in UML – Notation
- Umwandlung der Generalisierungen / Spezialisierungen in UML – Notation
- Umwandlung der Aggregationen in UML - Notation



ENDE

ER-Modell

# Literatur

---

- Broy, Manfred; Denert, Ernst:  
Software Pioneers - Contributions to Software Engineering. Springer, 2002
- Prof. Dr. E. Rahm, Vorlesung: „Datenbankysteme 1“ (Kapitel 2), Universität Leipzig  
(gehalten im Sommersemester 2003)
- Peter P. Chen, Bernhard Thalheim ...:  
Conceptual Modelling, Springer Verlag 1999
- verschiedene Internetseiten, u.a.:
  - [www-inf.fh-reutlingen.de/dbweb/content/DBWI/VORLES/KAP21.pdf](http://www-inf.fh-reutlingen.de/dbweb/content/DBWI/VORLES/KAP21.pdf)
  - [www.informatik.uni-stuttgart.de/ipvr/as/lehre/skripte/DBISWS0203/kapitel03.pdf](http://www.informatik.uni-stuttgart.de/ipvr/as/lehre/skripte/DBISWS0203/kapitel03.pdf)