

ISDA 02b (Tutorien)

EER Modellierung

Prof. Dr. Volker Markl

Folienmaterial von Prof. Dr. Felix Naumann
und Dr. Ralf-D. Kutsche



Fachgebiet Datenbanksysteme und Informationsmanagement
Technische Universität Berlin

<http://www.dima.tu-berlin.de/>

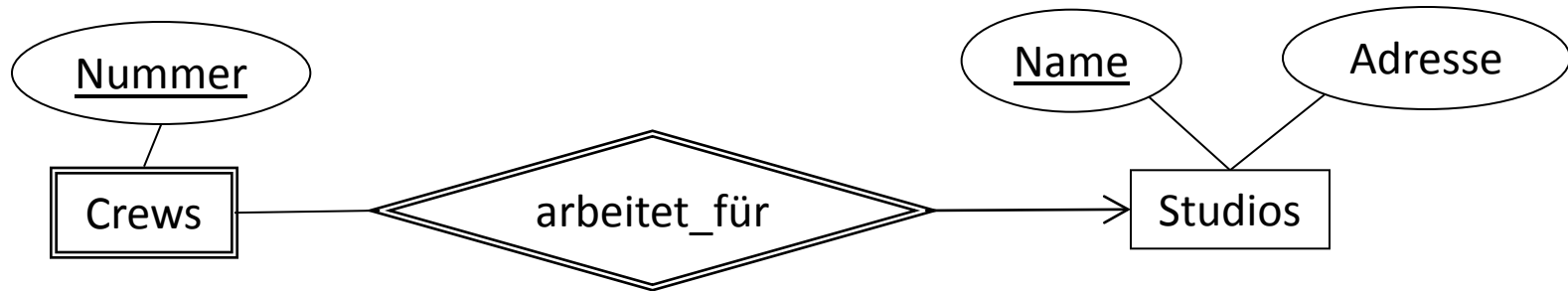
- Motivation und Einbettung
- Begriffe und Definitionen
- ER-Diagramme
- Modellierung von Nebenbedingungen
- **Schwache Entitytypen**
- Erweitertes ER-Modell
- Designprinzipien
- Sichtintegration

Kapitel 4 des Lehrbuchs

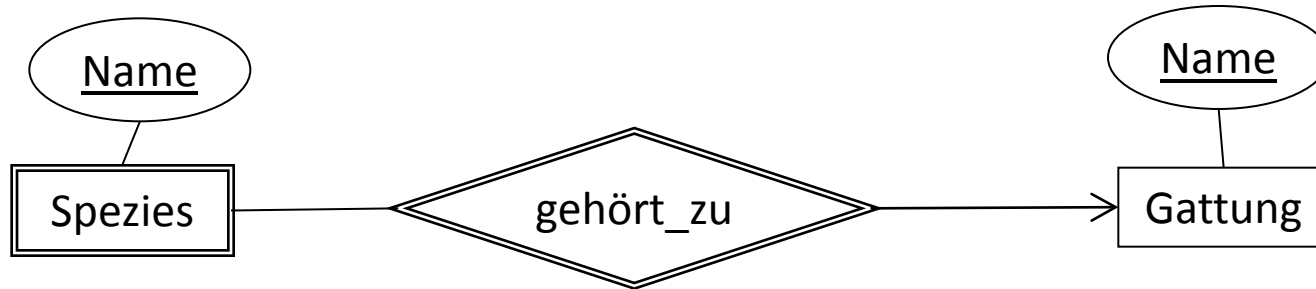
In Vorlesung behandelt

Tutorien

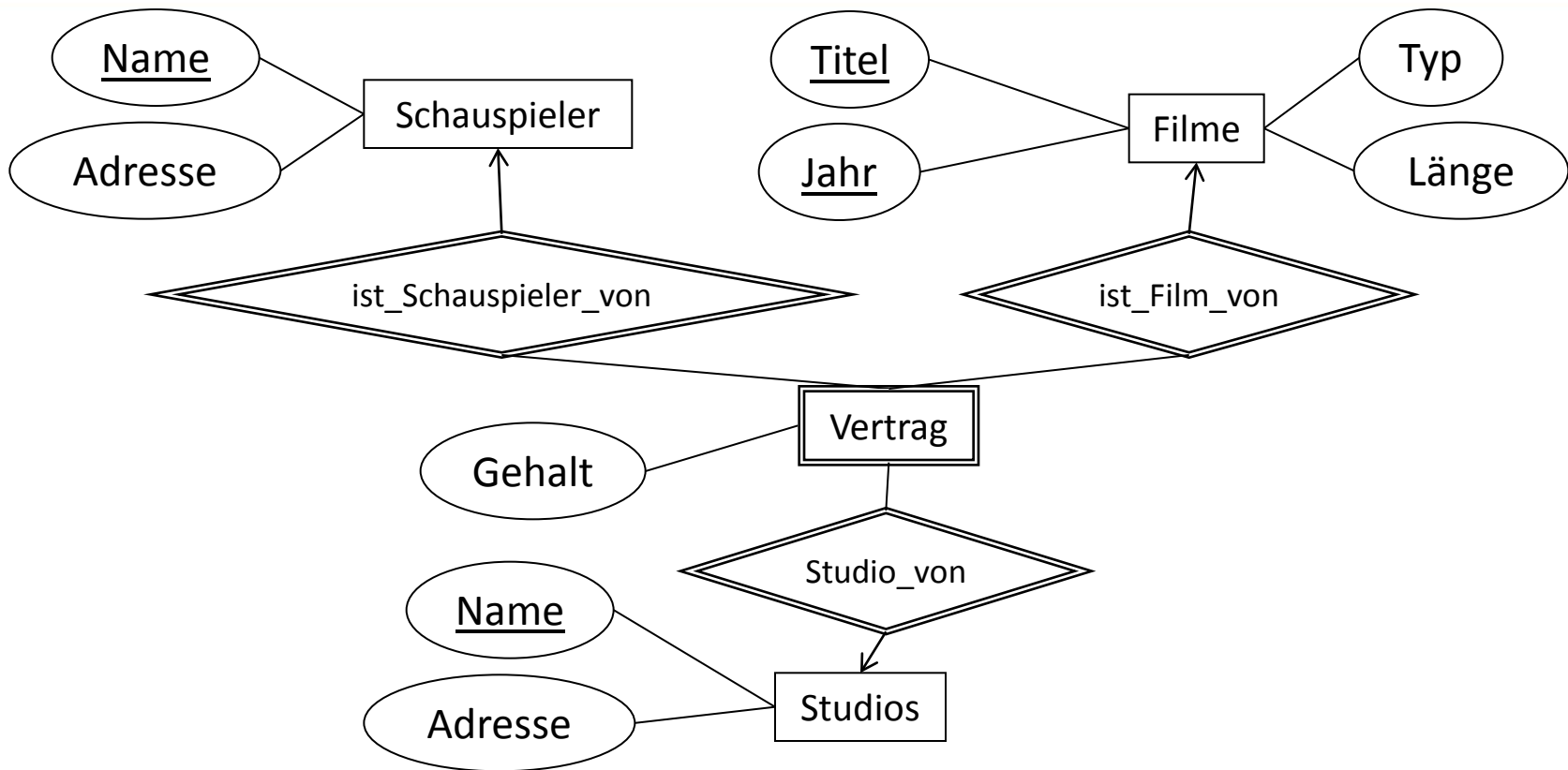
- In bestimmten Situationen können Entities nicht allein anhand ihrer Attribute identifiziert werden:
 1. Falls sie in eine nicht-IST-Hierarchie fallen.
 2. Entities, die zur Eliminierung n-ärer Relationships erschaffen wurden.
- Ein Entitytyp ist schwach wenn es zur eindeutigen Identifizierung eines Entities nötig ist, eine oder mehr n:1 Relationships zu folgen und den Schlüssel der verwandten Entities hinzuzunehmen.



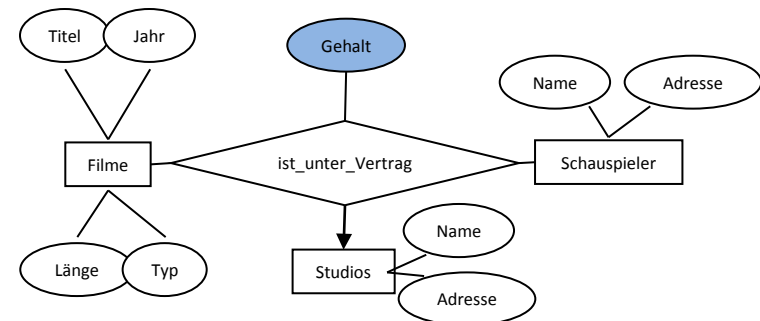
- Ein Studio beschäftigt mehrere Filmcrews.
- Filmcrews werden mit einer Nummer versehen.
- Verschiedene Studios könnten eigene Crews mit gleichen Nummern beschäftigen.
 - Nummer ist also kein Schlüssel
- Nimmt man den Schlüssel der Studios hinzu ist eine eindeutige Identifizierung möglich.



- Eine Spezies ist definiert durch den Namen der Gattung und des Spezies.
 - Gattung: homo
 - Spezies: homo sapiens



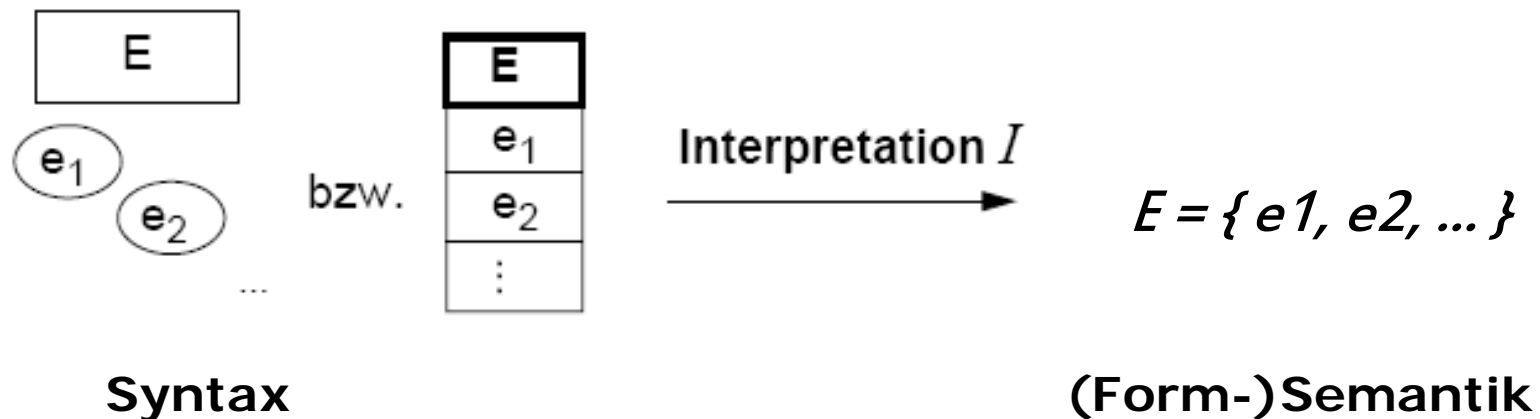
- Fall 2: Auflösung einer ternären Relationship
- Vertrag hat kein Attribut, das Teil des Schlüssels ist.



- Falls E ein schwacher Entitytyp ist, besteht sein Schlüssel aus...
 - ... null oder mehr eigenen Attributen
 - ... und den Schlüsselattributen von Entitytypen, die über bestimmte n:1 Relationshiptypen, den „unterstützenden Relationshiptypen“ erreicht werden können.
 - supporting relationships
- Unterstützende Relationshiptypen
 - n:1 vom schwachen Entitytypen zu einem anderen Entitytypen
 - Es muss referentielle Integrität gelten.
 - Falls referenzierter Entitytyp wiederum schwach ist, werden (rekursiv) weitere Schlüsselattribute übernommen.

- Mathematische Formalisierung, d.h. (Form-) Semantik oder Interpretation der Konzepte der E/R-Modellierung als auf eine sichere Grundlage unserer (graphischen) Modellierungstechnik mit den Konzepten:
 - Entity bzw. Entity-Typ,
 - Relationship bzw. Relationship-Typ
 - Attribut und Schlüsselattributsowie
 - Integritätsbedingungen
- Hierzu reichen im wesentlichen die allseits bekannten Grundbegriffe Menge, Relation, Abbildung sowie deren Eigenschaften und etwas Prädikatenlogik aus.

- Ein Entity-Typ E wird mathematisch interpretiert als eine Menge E von Elementen $e1, e2, \dots$, die ihrerseits die Interpretation der zu E gehörigen Entities $e1, e2, \dots$ sind, d.h. der in einem modellierten Diskursbereich relevanten Informationselemente.



- Ein n-ärer Relationship-Typ R zwischen den Entity-Typen E_1, E_2, \dots, E_n — kurz geschrieben als: $R(E_1, E_2, \dots, E_n)$ — wird interpretiert als

n-stellige Relation $R \subseteq E_1 \times E_2 \times \dots \times E_n$,

wobei die E_i die jeweiligen Interpretationen der E_i ($i=1, \dots, n$) seien.

- Eine einzelne Relationship r des Typs $R(E_1, E_2, \dots, E_n)$ ist demzufolge zu interpretieren als ein

n-Tupel $(e_1, e_2, \dots, e_n) \in R$,

wobei die e_i hier natürlich die Interpretationen der jeweiligen Entities e_i sind.

- Ein Entity-Attribut a zu einem Entity-Typ E wird interpretiert als Abbildung $a : E \rightarrow \text{dom}(a)$,

wobei $\text{dom}(a)$ den Wertebereich des Attributs a bezeichne und E die *Interpretation* von E sei.

Typische Wertebereiche sind Zahlenräume wie die natürlichen Zahlen \mathbb{N} , die ganzen Zahlen \mathbb{Z} oder Teilmengen davon sowie die Menge aller über einem Alphabet Σ bildbaren Zeichenketten Σ^* (bzw. nicht-leeren Zeichenketten Σ^+) oder Teilmengen davon.

- Ein Relationship-Attribut a zu einem Relationship-Typ $R(E_1, E_2, \dots, E_n)$ wird analog zu den Entity-Attributen interpretiert als

Abbildung $a : E_1 \times E_2 \times \dots \times E_n \rightarrow \text{dom}(a)$.

- Ist a ein Schlüsselattribut zum Entity-Typ E , so ist seine Interpretation a eine *injektive* Abbildung, d.h. zwei verschiedene Entities besitzen stets auch verschiedene Attributwerte, in Formeln:

$$\forall e_1, e_2 \in E : e_1 \neq e_2 \Rightarrow a(e_1) \neq a(e_2) .$$

- Zu den Integritätsbedingungen werden logische Formeln angegeben, deren klassische Interpretation in der Prädikatenlogik gerade die Semantik des entsprechenden Konzepts angibt.

- Kardinalitäten

- 1:n – Kardinalität, 1:1 – Kardinalität

Die Integritätsbedingung 1:n für einen Relationship-Typ $R(E1, E2)$ bedeutet, dass folgende Formel erfüllt sein muss:

$$\forall e1, e2 \in E1, e \in E2 : (e1, e) \in R \wedge (e2, e) \in R \Rightarrow e1 = e2$$

Der Fall n:1 ist hierzu genau symmetrisch, der Fall 1:1 erfordert das gleichzeitige Erfülltsein beider Bedingungen, d.h. die logische Konjunktion beider Formeln.

- Totalität

Die Formalisierung der Integritätsbedingung links-total eines Relationship-Typs $R(E1, E2)$ wird durch folgende Formel geleistet (die Fälle rechts-total bzw. bi-total analog):

$$\forall e1 \in E1 \exists e2 \in E2 : (e1, e2) \in R$$

- Motivation und Einbettung
- Begriffe und Definitionen
- ER-Diagramme
- Modellierung von Nebenbedingungen
- Schwache Entitytypen
- **Erweitertes ER-Modell**
- Designprinzipien
- Sichtintegration

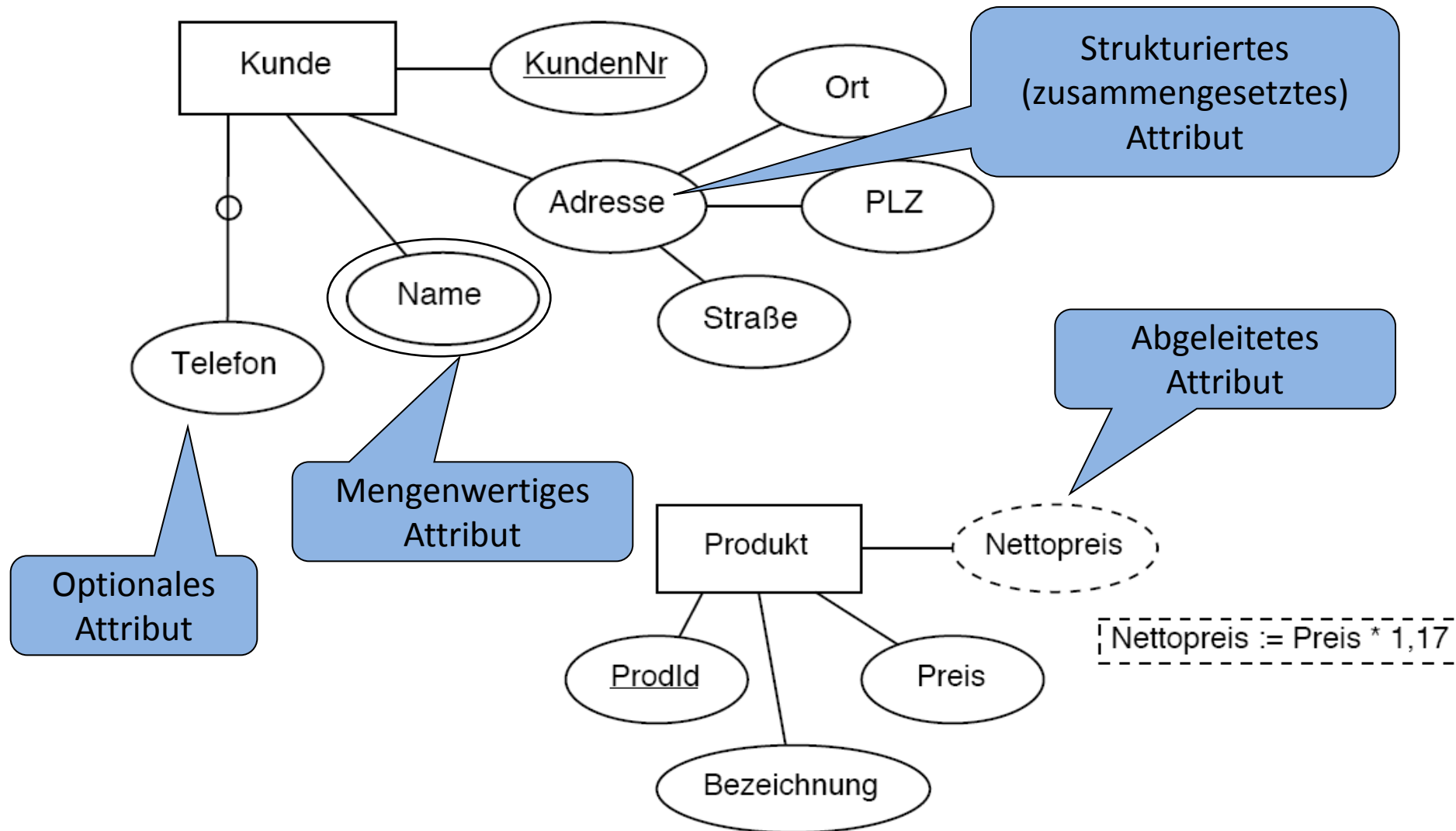


- Optionales Attribut
 - Attributwert nicht für jede Entität vorhanden

- Abgeleitetes Attribut
 - Wert wird anhand einer Berechnungsvorschrift aus nicht-abgeleiteten Attributen errechnet.

- Mengenwertiges Attribut
 - Enthält Menge von Werten

- Strukturiertes Attribut
 - Wird durch weitere Attribute beschrieben
 - Wert des strukturierten Attributs entspricht Verkettung der Unterattribute.



■ Generalisierung / Spezialisierung

Das Abstraktionskonzept der Generalisierung ermöglicht es, eine Menge "speziellerer", (oder: "zu generalisierender") Entity-Typen zu einem "allgemeineren" (oder: "generalisierten") Entity-Typ zu abstrahieren.

■ Aggregation

Ziel der Aggregationsabstraktion ist es, 'komplexe Strukturen' in ihrem Aufbau aus Einzelkomponenten (Teilen) zu beschreiben. Die korrespondierende vertikale Beziehung wird demzufolge als „Ist-Teil-von“ - Relationship-Typ (part-of) bezeichnet.

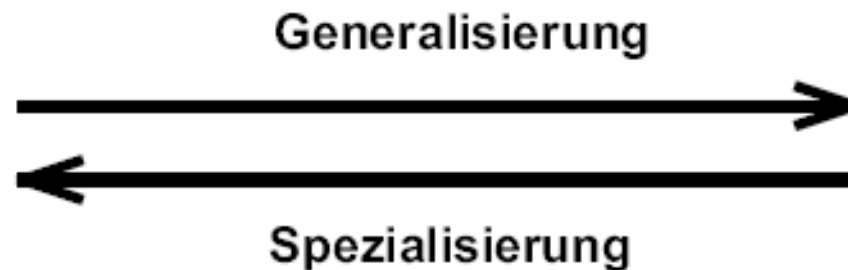
- Spezialisierung
 - entspricht der IST-EIN-Beziehung
 - Z.B. Ein Drache (Spezialisierung) IST-EIN Produkt)

- Generalisierung
 - Entities in einen allgemeineren Kontext
 - Z.B. Drache oder Windspiel generalisiert zu Produkt (Generalisierung)

- Partitionierung
 - mehrere disjunkte Entity-Typen (disjoint vs. overlapping)
 - Spezialfall der Spezialisierung
 - Partitionierung von Produkten in Zubehör und Drachen

- Vollständige Taxonomie
 - Übersicht über alle Spezialisierungen (total vs. partial)

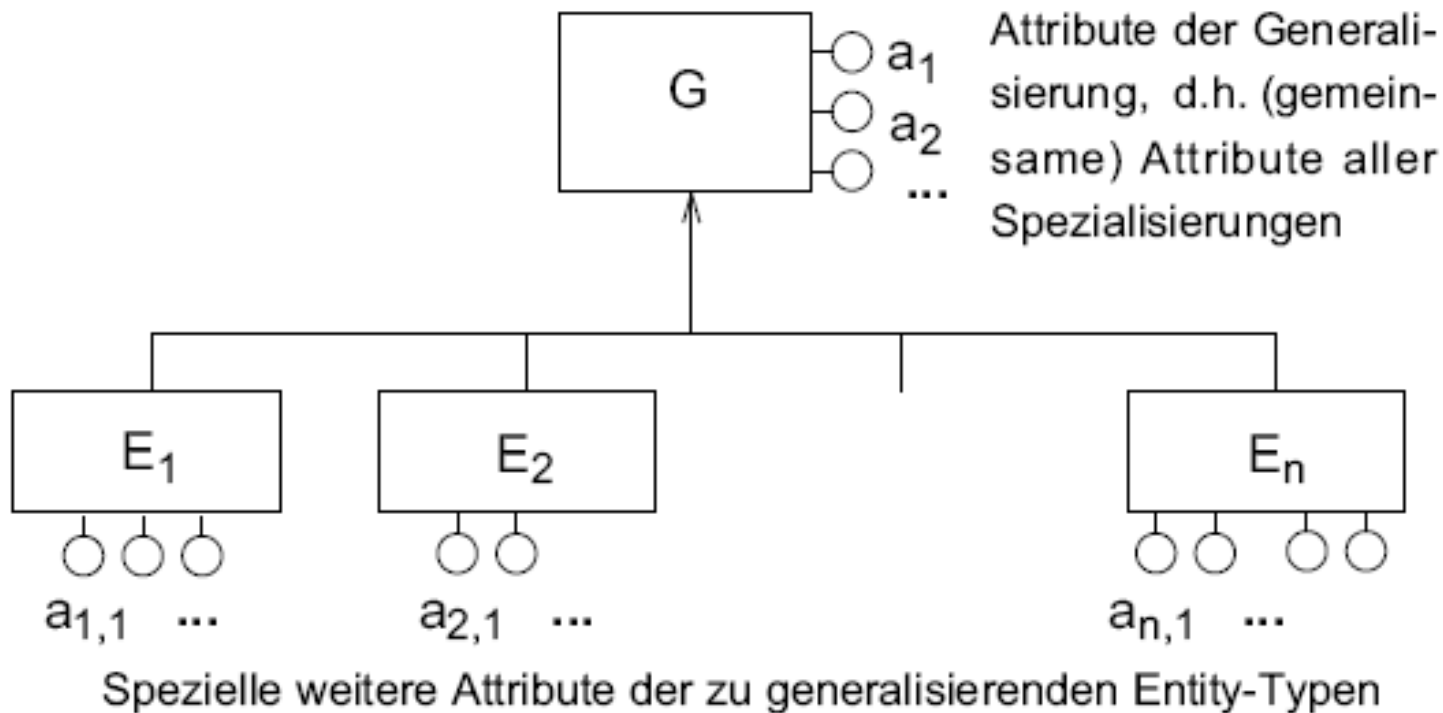
■ Konkretes Beispiel:



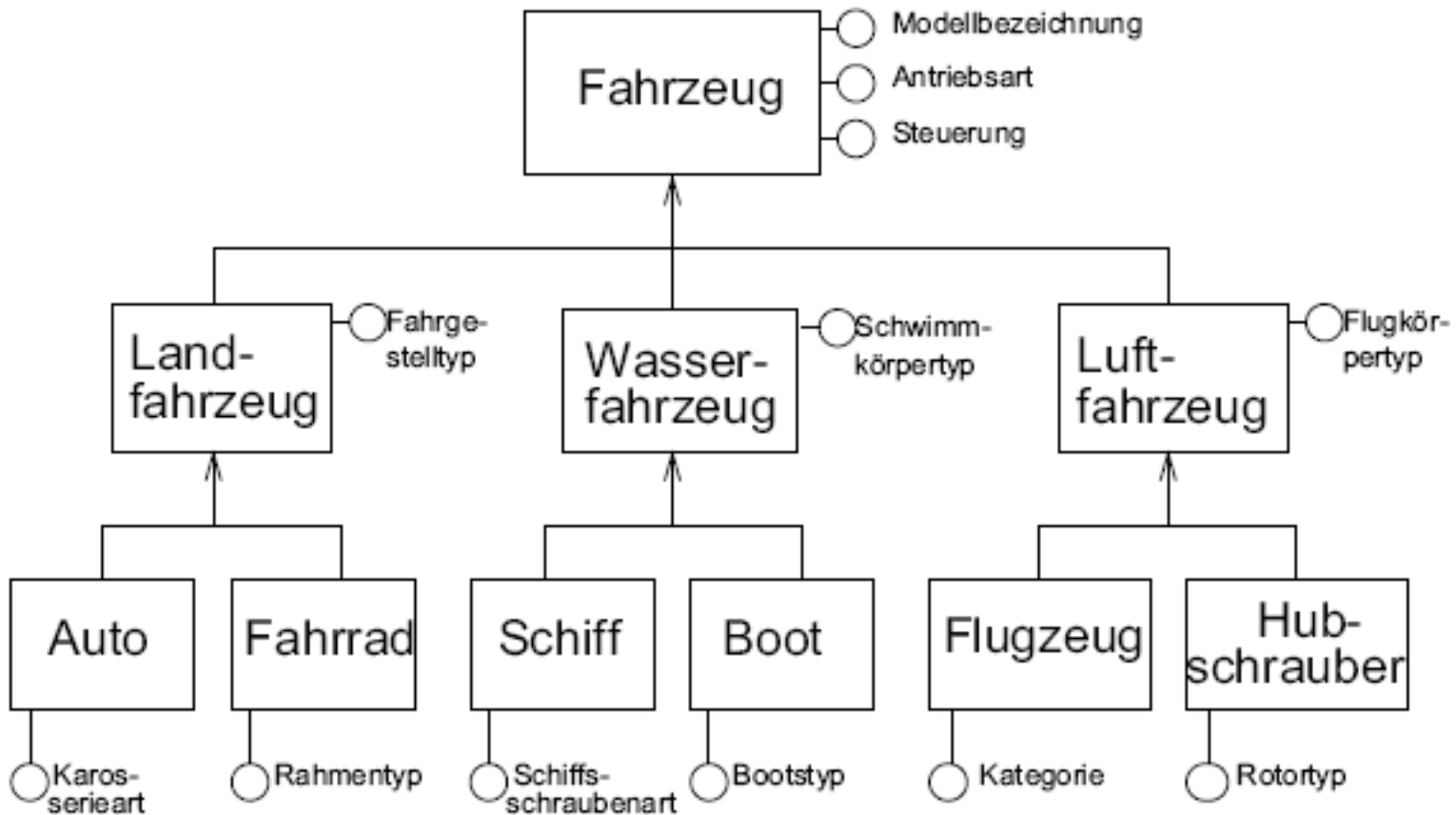
jetzt:

- Erweiterte Entity-Relationship-Type-Diagramme

hier: **Generalisierung / Spezialisierung**



- Konkretes Beispiel:
Darstellung des generalisierten Entity-Typs „Fahrzeug“



- Die Generalisierungs-/Spezialisierungs-Beziehung kann auf unterschiedliche Weise präzisiert werden:
- **total vs. partiell**
 - total, d.h. die Spezialisierungen schöpfen den Bereich der Generalisierung vollständig aus — es gibt also kein Entity des generalisierten Entity-Typs, das nicht schon seinerseits einer der vorgegebenen Spezialisierung angehört;
 - partiell, d.h. die angegebenen Spezialisierungen schöpfen den Bereich der Generalisierung nicht vollständig aus.
- **ausschließend vs. überlappend**
 - ausschließend (disjunkt), d.h. ein Entity des generalisierten Entity-Typs kann nicht zugleich mehreren Spezialisierungen angehören, sondern ausschließlich einer;
 - überlappend, d.h. ein Entity des generalisierten Entity-Typs kann zugleich mehreren Spezialisierungen angehören.

- total, überlappend:

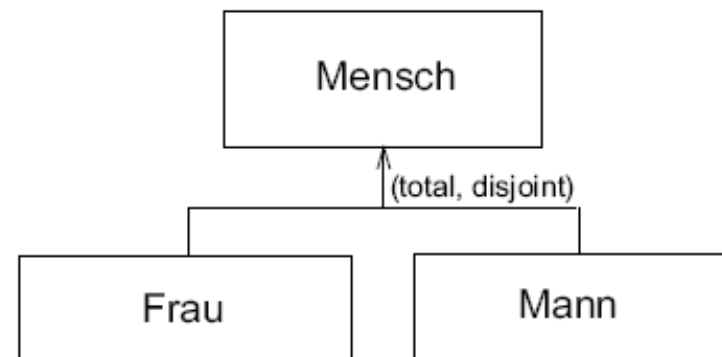


Es gibt keine Fahrzeuge als solche, die sich nicht auf dem Lande, zu Wasser oder in der Luft bewegen (total).

Allerdings gibt es Amphibienfahrzeuge, die sich sowohl auf dem Land als auch im Wasser bewegen können, und Wasserflugzeuge, die sowohl schwimmen als auch fliegen können (überlappend).

- total, ausschließend:

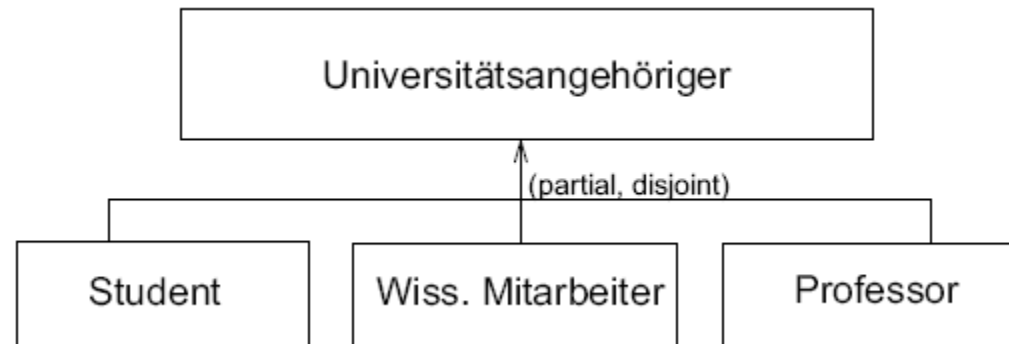
Jeder (total) Mensch ist entweder weiblich oder männlich (ausschließend).



■ partiell, ausschließend:

Die Statusgruppen sind (per Gesetz) disjunkt (ausschließend).

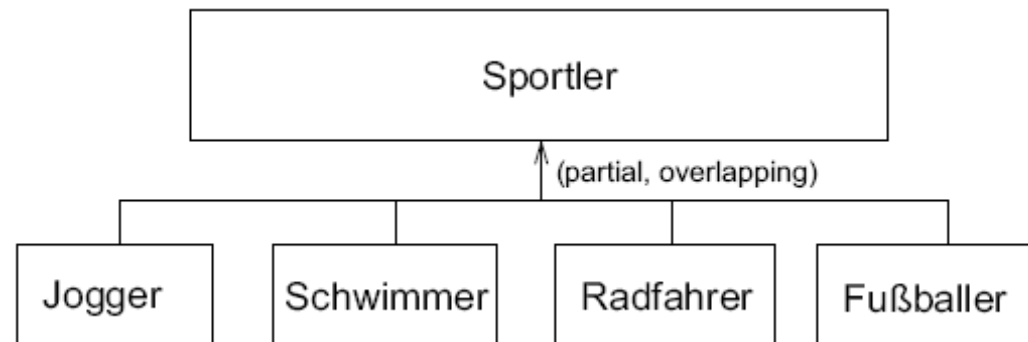
Es gibt weitere Spezialisierungen von Universitätsangehöriger, die in obigem Modell nicht erfasst sind, z.B. sonstige Dienstkräfte, Tutoren etc. (partiell).



■ partiell, überlappend:

Es gibt zahlreiche, in diesem Modell nicht als Spezialisierung erfasste Sportler, z.B. Turner, Handballer etc. (partiell).

Ein Jogger kann durchaus zugleich auch Schwimmer und Radfahrer sein, d.h. die Spezialisierungen sind nicht-disjunkt (überlappend).



- Ein durch die Generalisierungsabstraktion gebildeter abstrakter Entity-Typ wird (wie einfache Entity-Typen) durch eine Menge interpretiert. Für den generalisierten Entity-Typ und seine Spezialisierungen gelten folgende Teilmengenbeziehungen:

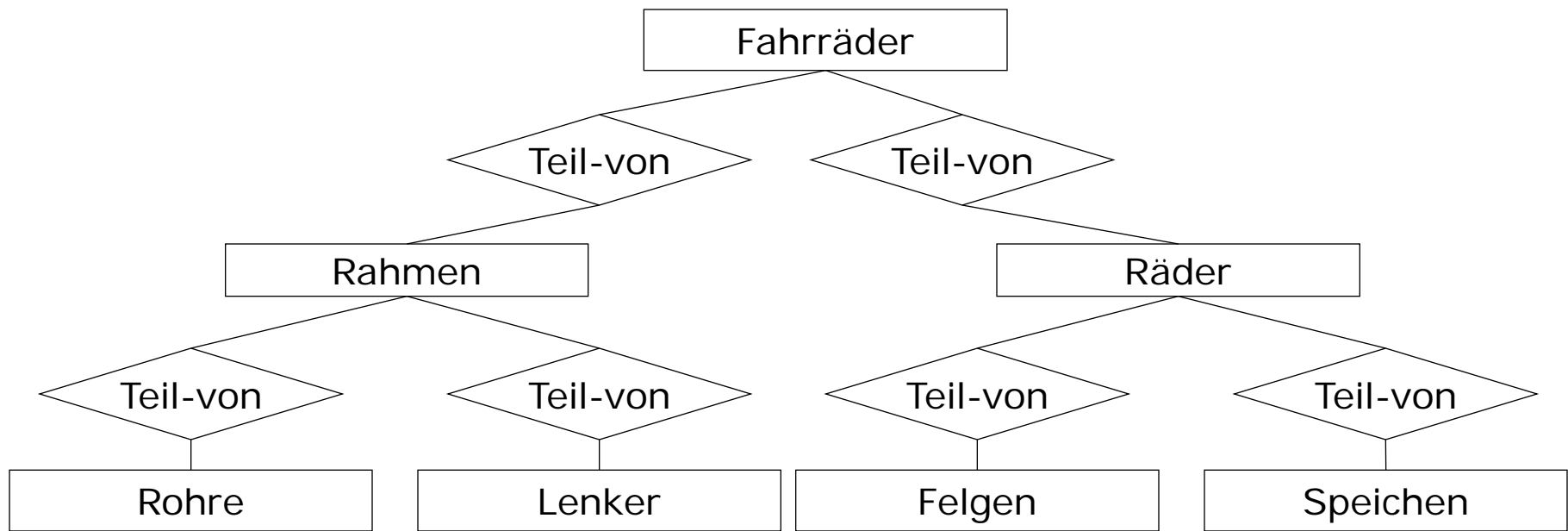
Sei G ein durch Generalisierung über den konstituierenden Entity-Typen E_1, E_2, \dots, E_n gebildeter abstrakter Entity-Typ und seien E_1, E_2, \dots, E_n und G die jeweiligen Interpretationen.

Dann gilt: $E_1 \subseteq G, E_2 \subseteq G, \dots, E_n \subseteq G$

Seien $A_i = \{ a(i,1), a(i,2), \dots, a(i,k) \}$ die Attribute des Entity-Typs E_i , A die des Entity-Typs G .

Dann gilt: $A_i \supset A$ („echte“ Obermenge)

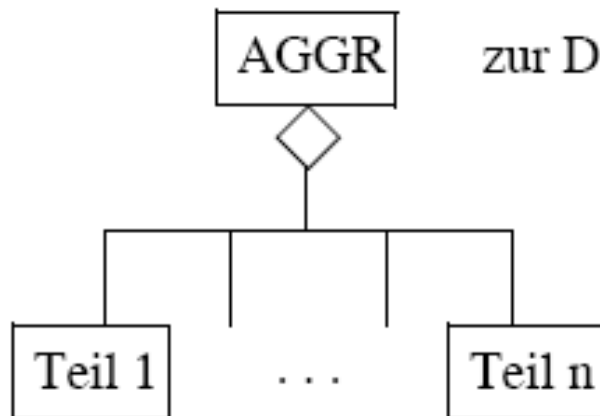
- Generalisierung (IST): Gleichartige Entitytypen
- Aggregation: Unterschiedliche Entitytypen
 - „Teil-von“ (part-of) bzw. (Leserichtung!): „Besteht-aus“)
 - Entity aus einzelnen Instanzen anderer Entity-Typen zusammengesetzt.



jetzt:

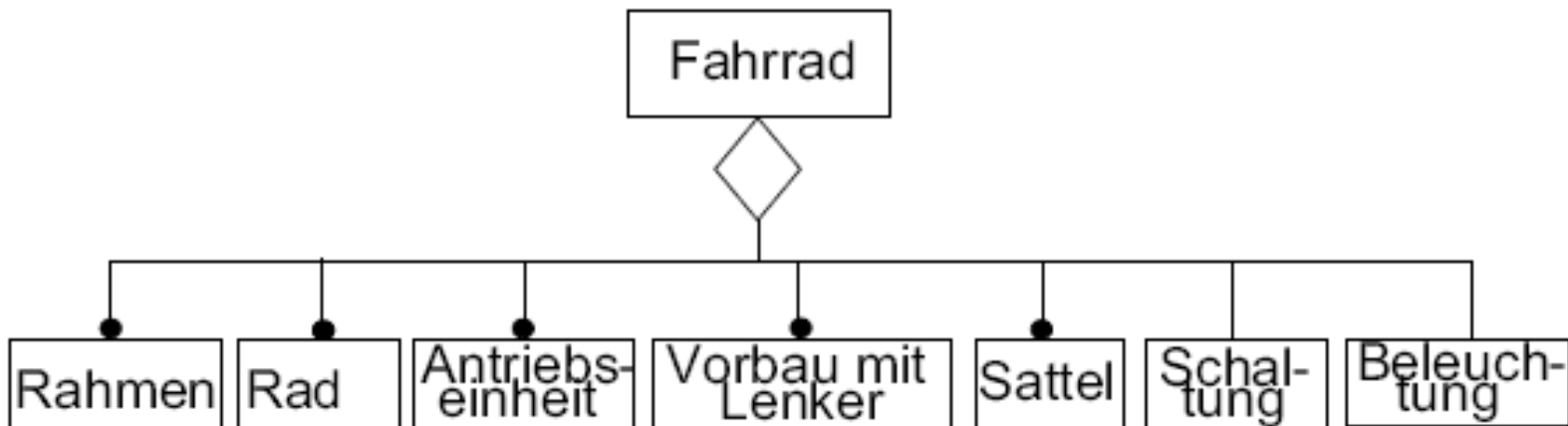
- Erweiterte Entity-Relationship-Typ-Diagramme

hier: **Aggregation**



zur Darstellung von aggregierten Entity-Typen
(Aggregationshierarchie)

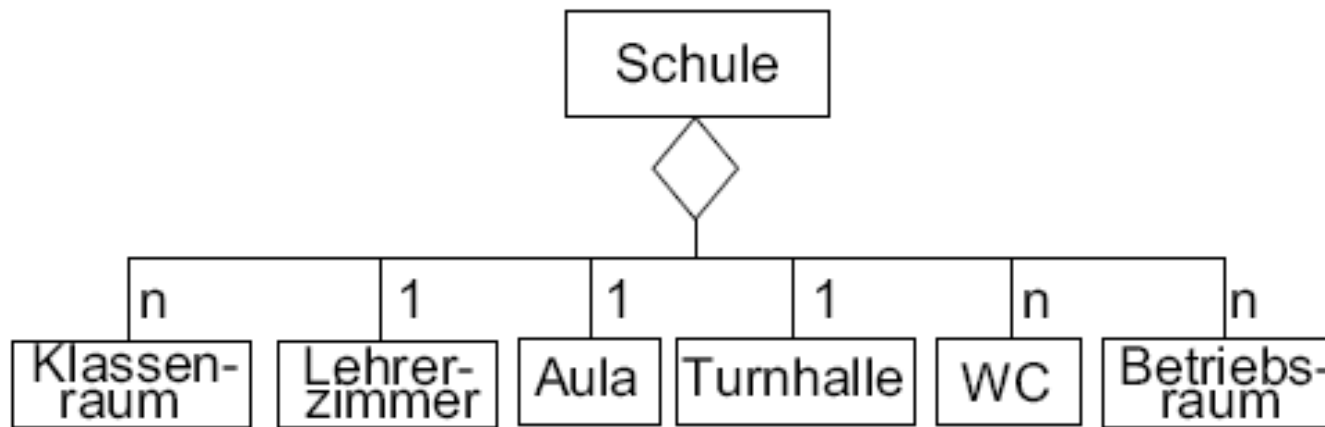
- **Beispiel: Totalität bei der Aggregationsabstraktion**
- Ohne Rahmen, Räder, Antriebseinheit, den Lenkervorbau und einen Sattel wollen wir ein Fahrrad nicht Fahrrad nennen (d.h. obligate Komponenten, also: Totalität der „Ist-Teil-von“-Beziehung)



- Eine Schaltung, Beleuchtung u.v.a.m. (wie Klingel, Gepäckträger, ...) sind zwar nützlich, z. T. auch in der Straßenverkehrsordnung vorgeschrieben, aber nicht zwingend an jedem Fahrrad.

■ Beispiel: Kardinalitäten bei der Aggregationsabstraktion

- Eine Schule besteht aus mehreren Klassenräumen, Betriebsräumen und Toiletten.
- Dagegen wollen wir nicht mehr als ein Lehrerzimmer, eine Aula und eine Turnhalle erlauben.



- (Zusätzlich könnte man hier über die Totalität noch die Existenz bestimmter Komponenten erzwingen, etwa der Klassenräume!)

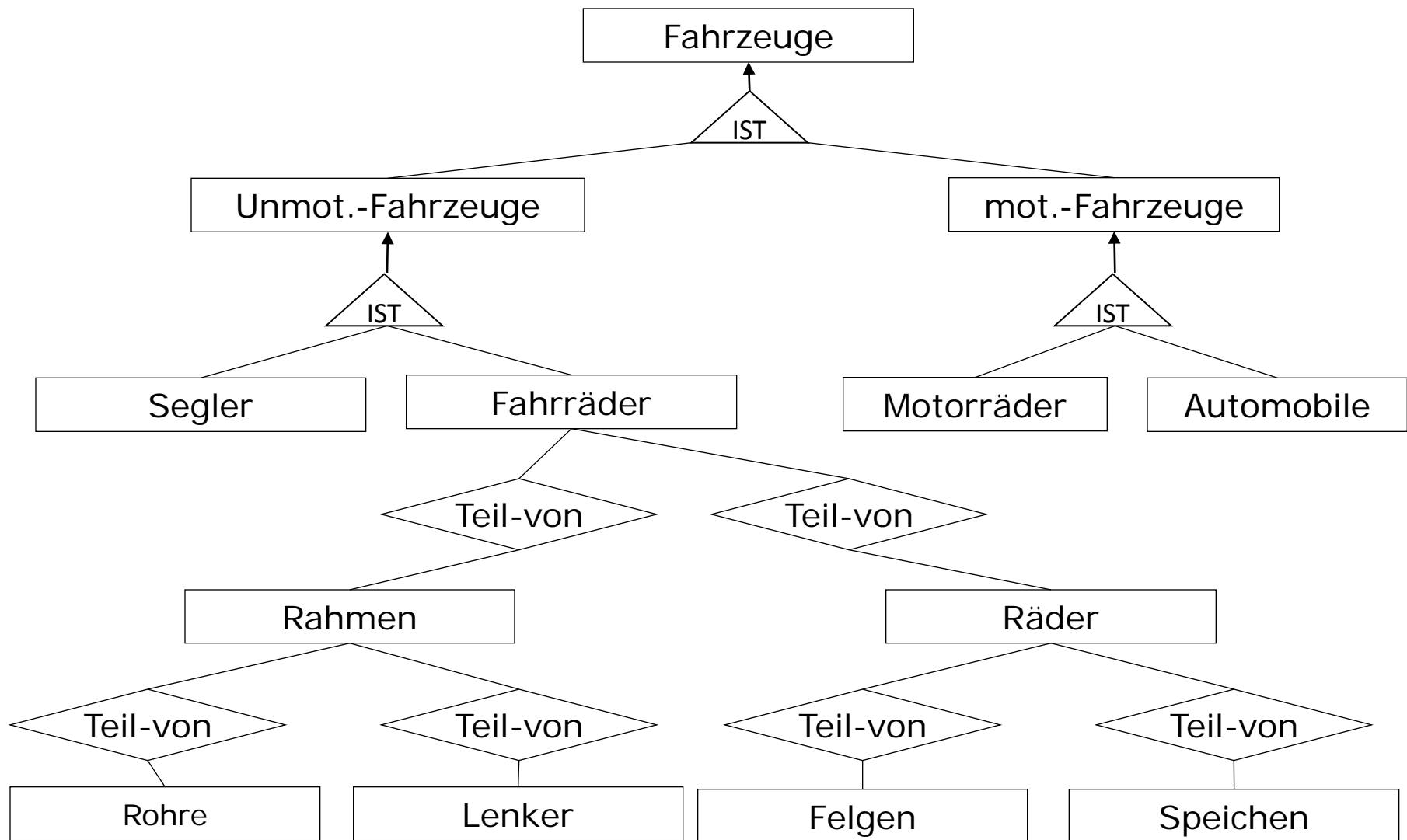
- Bei der Abstraktion durch Generalisierung wie bei der durch Aggregation handelt es sich um die vertikale Strukturierung eines Informationsmodells.
- Dies geschieht mit der Absicht, strukturelle Aussagen über das Modell auf unterschiedlichen Granularitätsniveaus treffen zu können.
- Häufig stehen Generalisierungs- und Aggregationshierarchien in einem engen Zusammenhang und werden deshalb auch oft gemeinsam (kombiniert) dargestellt.

- Aus den vorangegangenen Betrachtungen wird deutlich, dass sich die Aggregation mathematisch als (kartesische) Produktbildung darstellt — ähnlich zur Semantik von Relationship-Typen.
- Unterschiedlich behandelt werden müssen lediglich mengenwertige vs. einfache Aggregationen, da sich hier einfache Tupel von solchen unterscheiden, bei denen einzelne Komponenten mehrere Elemente enthalten können.
- Sei A durch Aggregation über den Entity-Typen $E_1, \dots, E_k, E_{k+1}, \dots, E_n$ entstandener abstrakter Entity-Typ, wobei o.B.d.A. die E_i ($1 \leq i \leq k$) einfach und die E_j ($k+1 \leq j \leq n$) mengenwertig aggregiert werden.
- Dann ist

$$A := E_1 \times \dots \times E_k \times \wp(E_{k+1}) \times \dots \times \wp(E_n)$$

die Menge aller möglichen Aggregate auf der Seite der Semantik, die über den Interpretationen $E_1 \dots E_n$ der Entity-Typen E_1, \dots, E_n gebildet werden können.

Bem. $\wp(E)$ bezeichnet hier wie üblich die Potenzmenge von E , d.h. die Menge aller Teilmengen von E .



- Komplexe Objekte
 - Teilmenge
 - Spezialfall einer Spezialisierung: Nur ein Subelement
 - Sammlung oder Assoziation
 - Mengenbildung
 - Team als Gruppe von Personen

- Beziehungen höheren Typs
 - Spezialisierung und Generalisierung auch für Beziehungstypen.
 - Beispiel: Beziehung `bestellt` zu `bestelltPerExpress` spezialisiert.
 - Beziehungen zwischen Beziehungsinstanzen: Beziehungen zweiter und höherer Ordnung

- Motivation und Einbettung
- Begriffe und Definitionen
- ER-Diagramme
- Modellierung von Nebenbedingungen
- Schwache Entitytypen
- Erweitertes ER-Modell
- **Designprinzipien**
- Sichtintegration



- Vorgehensmodell beim E/R- bzw. EER-Entwurf
 - Informationsanalyse und E/R-Modellierung in 3 Phasen:
 - frühe, mittlere und späte Entwurfsphase
 - danach dann Datenbankentwurf (Schema der DB, ggf. Normalisierung) und Umsetzung in eine Implementierung
- Konkrete Grundprinzipien beim E/R-Entwurf
 - Treue zur Anwendung
 - Vermeidung von Redundanz
 - Einfachheit
 - Überlegte Entscheidung zwischen Entity-Typen und Attributen
 - Sparsamer Einsatz von Relationships
 - Sparsamer Einsatz von schwachen Entitytypen

- Folgende Fragen sind dafür zu beantworten:
 - Welche Arbeitsschritte führen eigentlich zum E/R-Entwurf?
 - Worauf muss besonders geachtet werden?
 - Wie geht man überhaupt an die Information eines Diskursbereichs heran?
 - Wie skaliert man Informationsmodelle (Vergröberung, Verfeinerung)?
 - u.v.a.m.

- Wichtige Prinzipien:
 - Entwurfsprozess spiralförmig in mehreren Phasen
 - Zu Beginn des Entwurfs die Informationsanalyse
 - Diskussion früher Entwürfe der E/R-Diagramme (graphisch!) zum 'feedback' mit dem Auftraggeber/ Kunden

- ... beinhaltet im wesentlichen die grundsätzliche Identifikation der relevanten Information des Diskursbereichs (oft durch Analyse von Texten wie Anforderungsdefinitionen fachliche Themenbeschreibungen und Ist- wie Soll-Konzepten informationstechnischer Lösungen)
- **Analyse der relevanten Begriffe des Diskursbereichs,**
- dabei im einzelnen:
 - Klärung/Präzisierung der **generellen Zielsetzung** der geplanten Systementwicklung und, davon ausgehend, der Informationsanalyse und -modellierung
 - **erste Identifikation von den wichtigen Begrifflichkeiten** ("Dingen") des Diskursbereichs (Entity-Typen) und ihren Eigenschaften und Beziehungen zueinander
 - **Elimination von (textuellen) Unklarheiten,** Doppeldeutigkeiten usw. (Synonyme, Homonyme) ...
 - **Ausgrenzung der irrelevanten Begriffe** (Begründung!)

... daraus:

- Erstellung eines **Glossars** der konzeptuell relevanten Begriffe
d.h. eine strukturierte, stichwortartige textuelle Beschreibung der wesentlichen Begriffe

... und unmittelbar dabei:

- vorläufige Einteilung dieser Begriffe in **Entity-Typen, Attribute und Relationship-Typen**

... und ebenfalls parallel dazu:

- Erstellung des **ersten Entwurfs eines E/R-Typ-Diagramms**
wobei oft schon gewisse Unstimmigkeiten des Glossars oder der Einteilung Entity-Typen vs. Attribute auffallen und zu einer Überarbeitung Anlass geben.

... sowie begleitend (über alle Phasen der Modellierung hinweg!):

- **Protokollierung** aller wichtigen Diskussionspunkte / Entwurfsentscheidungen

... beinhaltet die weitere Untersuchung des Diskursbereiches und des bislang entwickelten Modells in Bezug auf:

- **Präzisierung** (und ggf. Revision) der zuvor entworfenen Begrifflichkeit durch Analyse weiterer Eigenschaften und Beziehungen

und damit zusammenhängend

- Analyse und Darstellung von **Integritätsbedingungen** in der in Abschnitt 3.2 eingeführten Form

sowie

- Untersuchung des Modells auf mögliche **Abstraktionen** durch Verwendung geeigneter Abstraktions- und Modularisierungskonzepte

- ... setzt den in den vorhergehenden Phasen begonnenen Prozess konsequent fort und liefert als Ergebnis das komplette E/R-Modell mit allen Begleitdokumenten, wobei nunmehr (erstmalig) eine ausreichende Präzisierung der relevanten Begrifflichkeit im Diskursbereich für einen möglichen Datenbank-Entwurf gegeben ist:
- durch die bereits vorliegenden oder neu erstellen textuellen Beschreibungen / **Anforderungsdefinitionen**,
- durch das als graphische Repräsentation besonders wichtige **E/R-Typ-Diagramm** mit nochmals verfeinerten Integritätsbedingungen (ggf. durch Erweiterung der Notation und textuelle Ergänzungen) und nochmals verfeinerten und ggf. typisierten Attributen,
- das **Begriffsglossar**, das seinerseits ergänzt wird durch textuelle Beschreibungen beispielsweise von funktionalen und dynamischen Zusammenhängen oder durch erläuternde Instanz-Beispiele,
- sowie das (oftmals separat geführte, z.T. auch in das Glossar integrierte) **Protokoll der Entwurfsentscheidungen**

- Treue zur Anwendung
- Vermeidung von Redundanz
- Einfachheit
- Überlegte Entscheidung zwischen Entity-Typen und Attributen
- Sparsamer Einsatz von Relationships
- Sparsamer Einsatz von schwachen Entitytypen

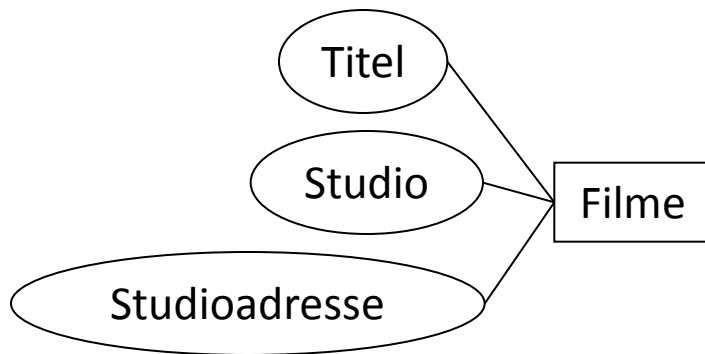
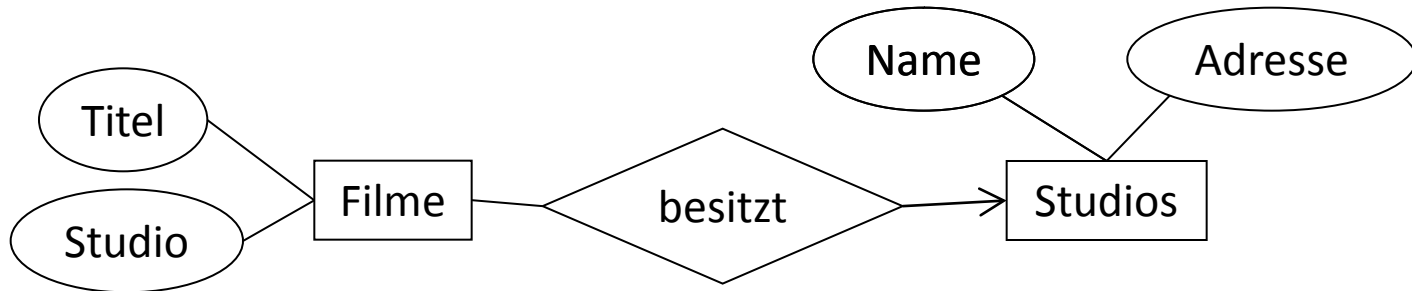
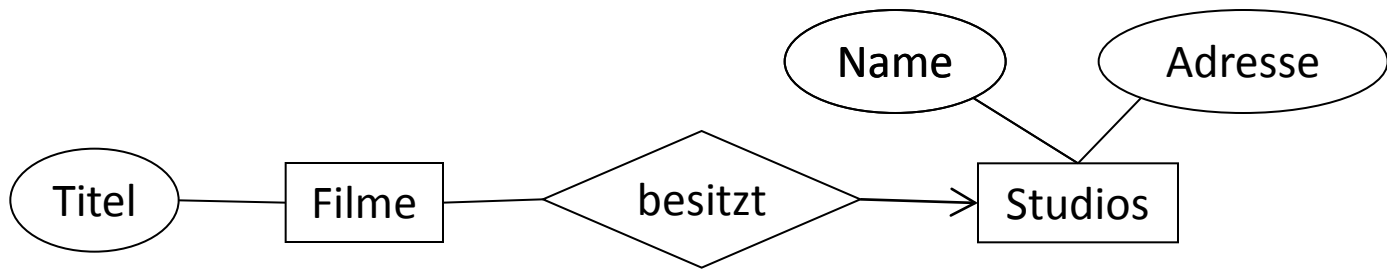
- Entitytypen und Attribute sollten Realität widerspiegeln.
 - Filme haben keine Zylinderkopfanzahl
- Relationstypen sollen Verhältnisse der Realität widerspiegeln.
 - Schauspieler und Filme stehen in einer m:n Beziehung
 - n:1, 1:n oder 1:1 wären inkorrekte Wiedergaben der Realität
- Schwierigerer Fall: Kurs und Lehrer – je nach Semantik
 - Ein Kurs kann nur von einem (verantwortlichen) Lehrer gegeben werden.
 - Lehrer geben im Team einen Kurs.
 - Nicht aktuelle Kursvergabe sondern auch Historie

n:1

m:n

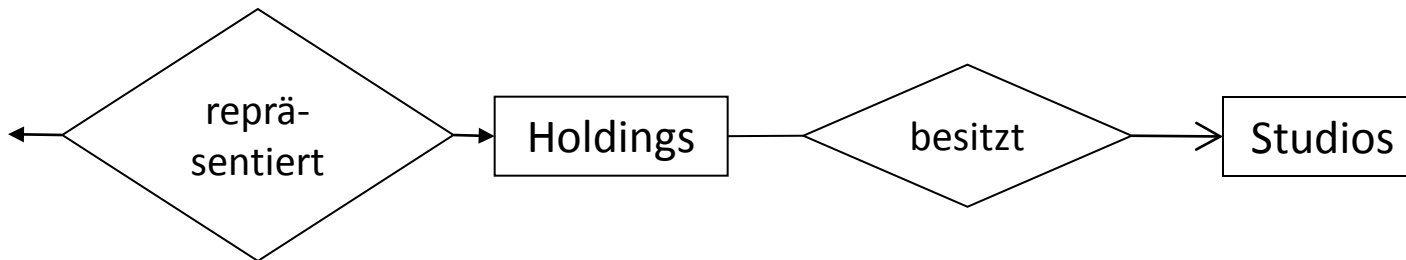
m:n

- Redundanz tritt auf, wenn der gleiche Sachverhalt auf mehr als eine Weise ausgedrückt wird.
- Redundanz verschwendet Platz.
 - Auf dem Papier
 - Auf der Festplatte
- Redundanz fördert Inkonsistenz.
 - Veränderung eines Sachverhalts wird nur an einer Stelle repräsentiert.

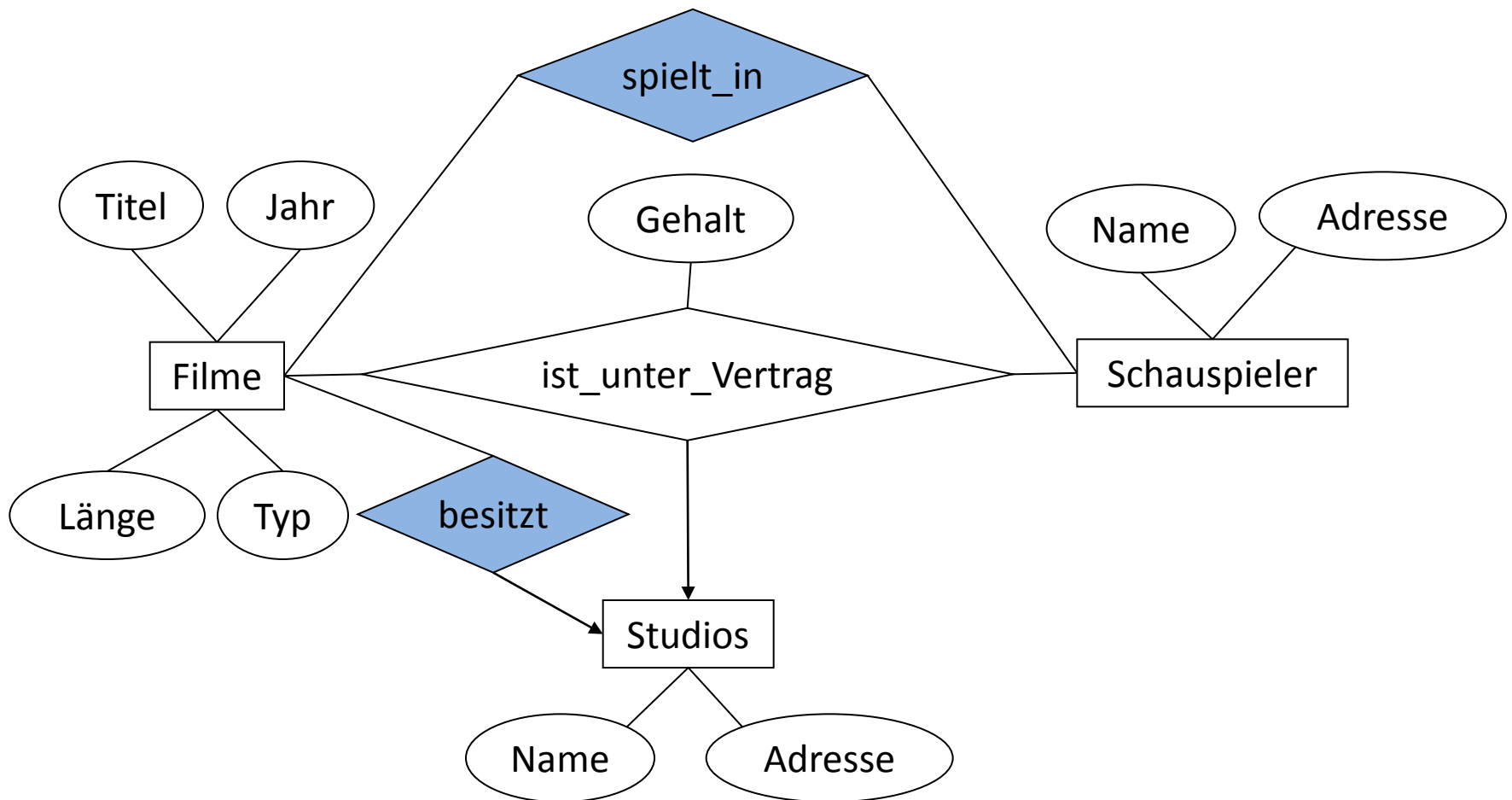


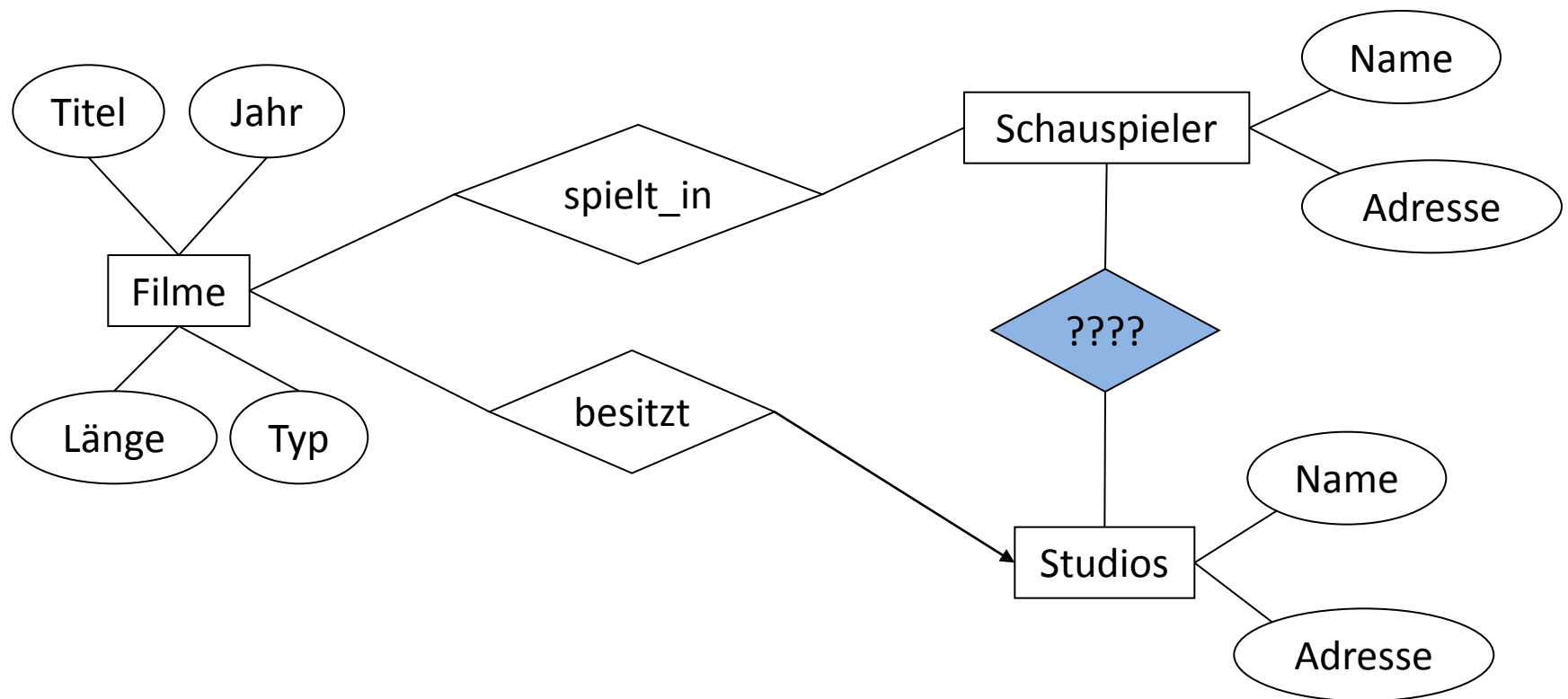
- Studioadresse wird für jeden Film wiederholt
- Studioadresse geht verloren wenn zwischenzeitlich kein Film existiert

- KISS: Keep It Simple, St...
- Unnötige Verwendung von Entitytypen vermeiden.
- Ein Film wird von einer Holding repräsentiert

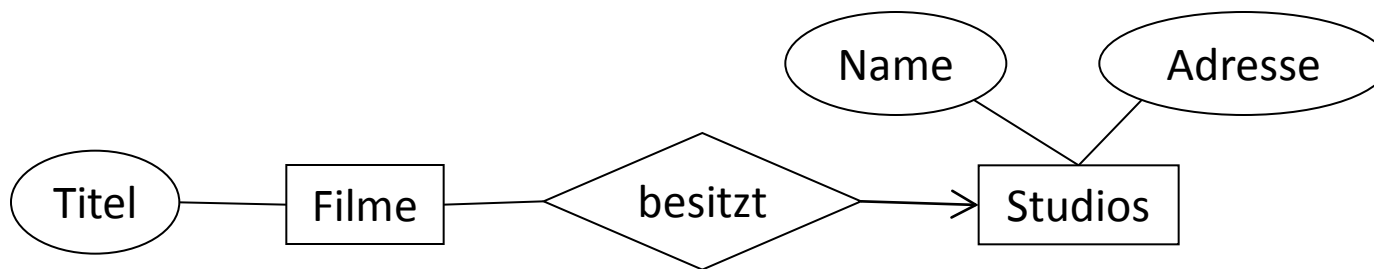


- Nicht jede mögliche Beziehung sollte abgebildet werden.
 - Vermeidung von Redundanz, wenn manche Beziehungen abgeleitet werden können.
 - Änderungen auf der Datenbank werden komplex
 - Eine Änderung eines Entities verursacht viele Änderungen in den Relationships.
 - Fehlergefahr
 - Vermehrter Aufwand

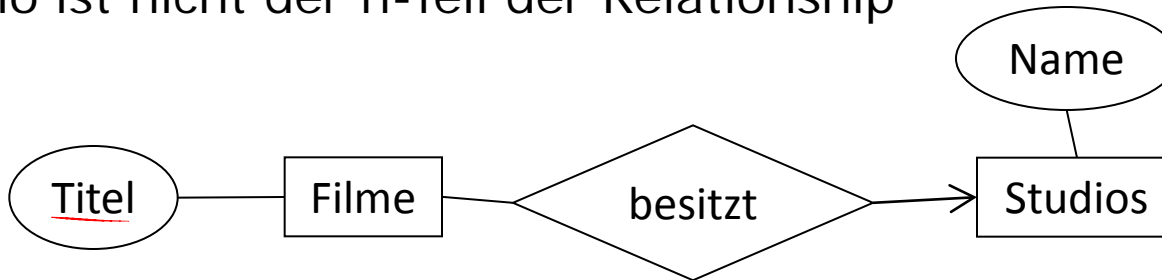




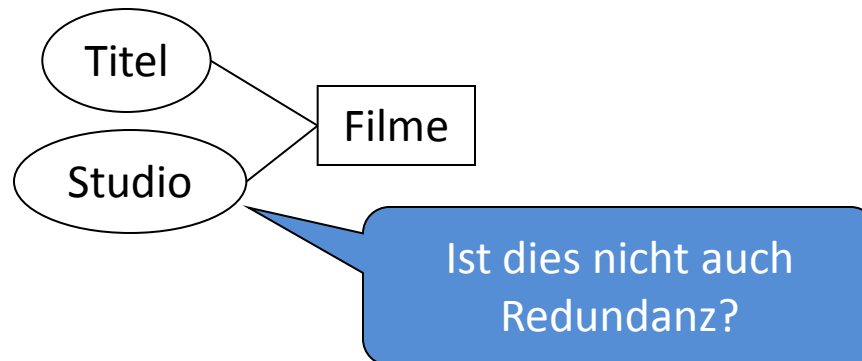
- Attribute sind einfacher zu implementieren als Entities und Relationships.
- Ein Entitytyp ist gerechtfertigt falls...
 - ... er mehr als nur den Namen eines Objekts darstellt,
 - ... oder er der n-Teil einer 1:n Relationship ist.



- Studio ist nur ein Name
- Studio ist nicht der n-Teil der Relationship



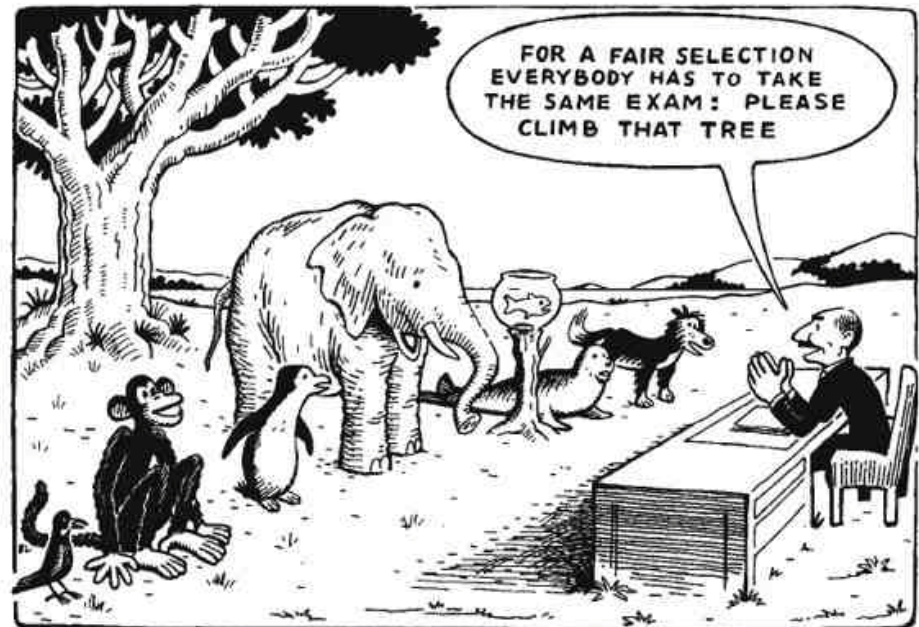
- Besser



- Man scheut sich oft einen Schlüssel zu deklarieren.
- Die Folge: Man schwächt ein Entitytyp und macht alle seine Relationships zu unterstützenden Relationships.
- In der Realität werden sehr oft künstliche IDs verwendet.
 - ISBN, SNN, VIN, etc.
- Grund für das Fehlen eines solchen Schlüssels: Es gibt keine entsprechende Autorität, die einen solchen Schlüssel vergeben könnte.
 - Bsp: Es ist unwahrscheinlich, dass jedem Fußballer der Welt eine eindeutige ID zugewiesen wird.

- Bitte erstellen Sie eine Multiple Choice Aufgabe zum Thema Schwache Entity Typen, EER, Designprinzipien oder Sichtintegration
 - Formulieren Sie eine Frage und 3 Antworten (A, B, C)
 - Davon sollte mindestens eine Antwort richtig und mindestens eine Antwort falsch sein
- Geben Sie die Aufgabe an Ihren rechten Nachbarn. Diskutieren Sie gemeinsam und markieren Sie die richtigen Lösungen
- Geben Sie am Ende der Vorlesung Ihre Aufgabe bei mir ab

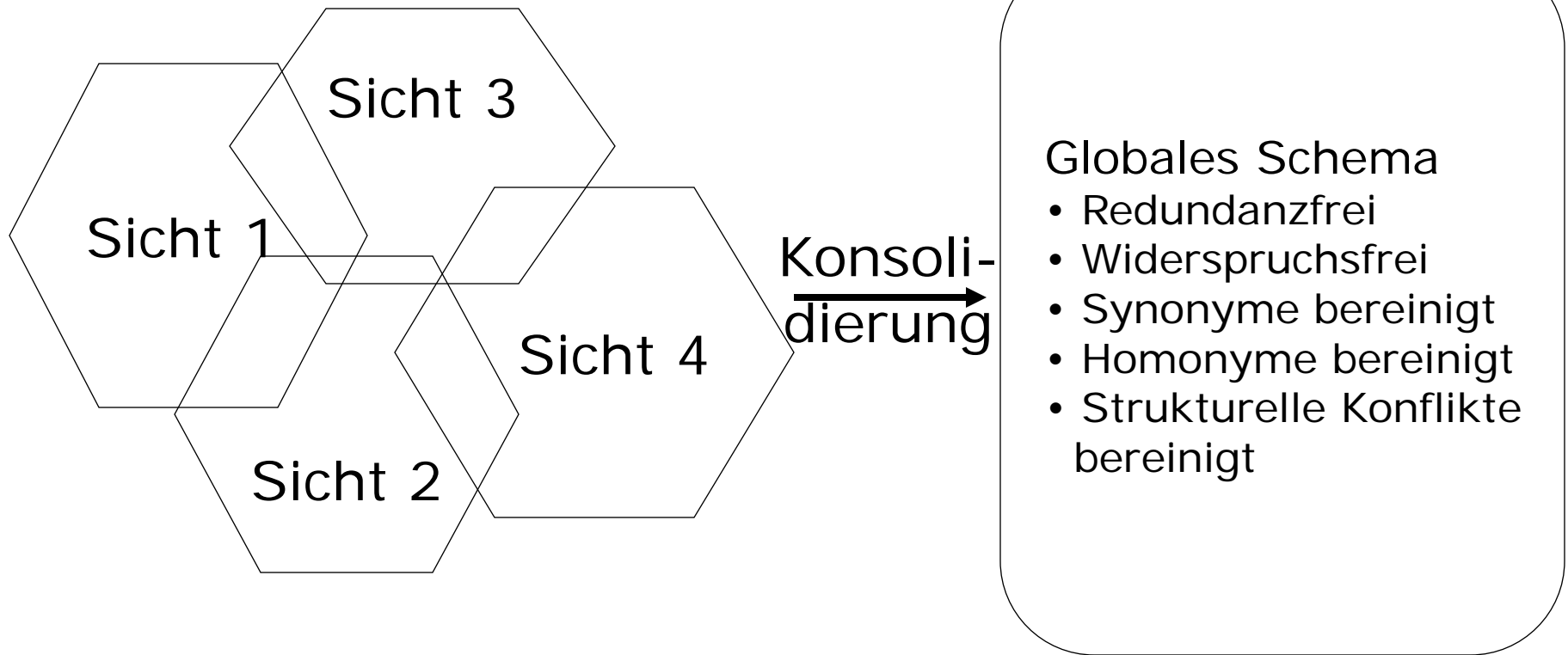
5 min

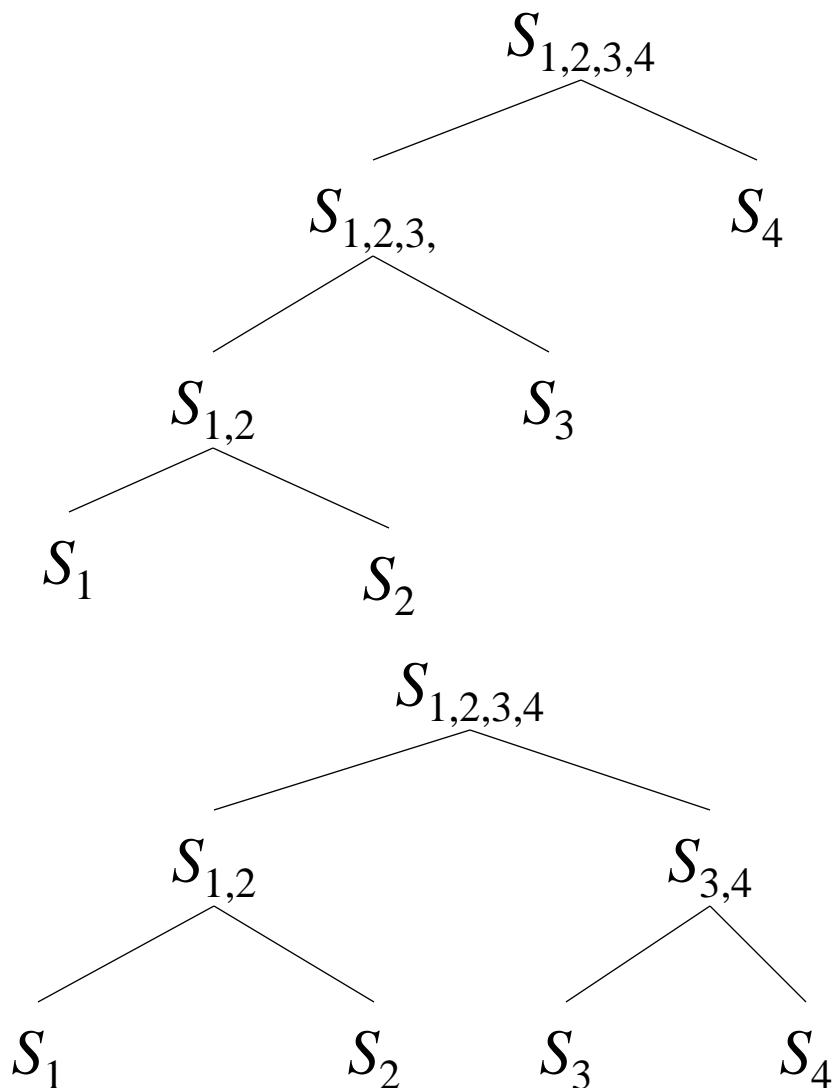


- Motivation und Einbettung
- Begriffe und Definitionen
- ER-Diagramme
- Modellierung von Nebenbedingungen
- Schwache Entitytypen
- Erweitertes ER-Modell
- Designprinzipien
- **Sichtintegration**



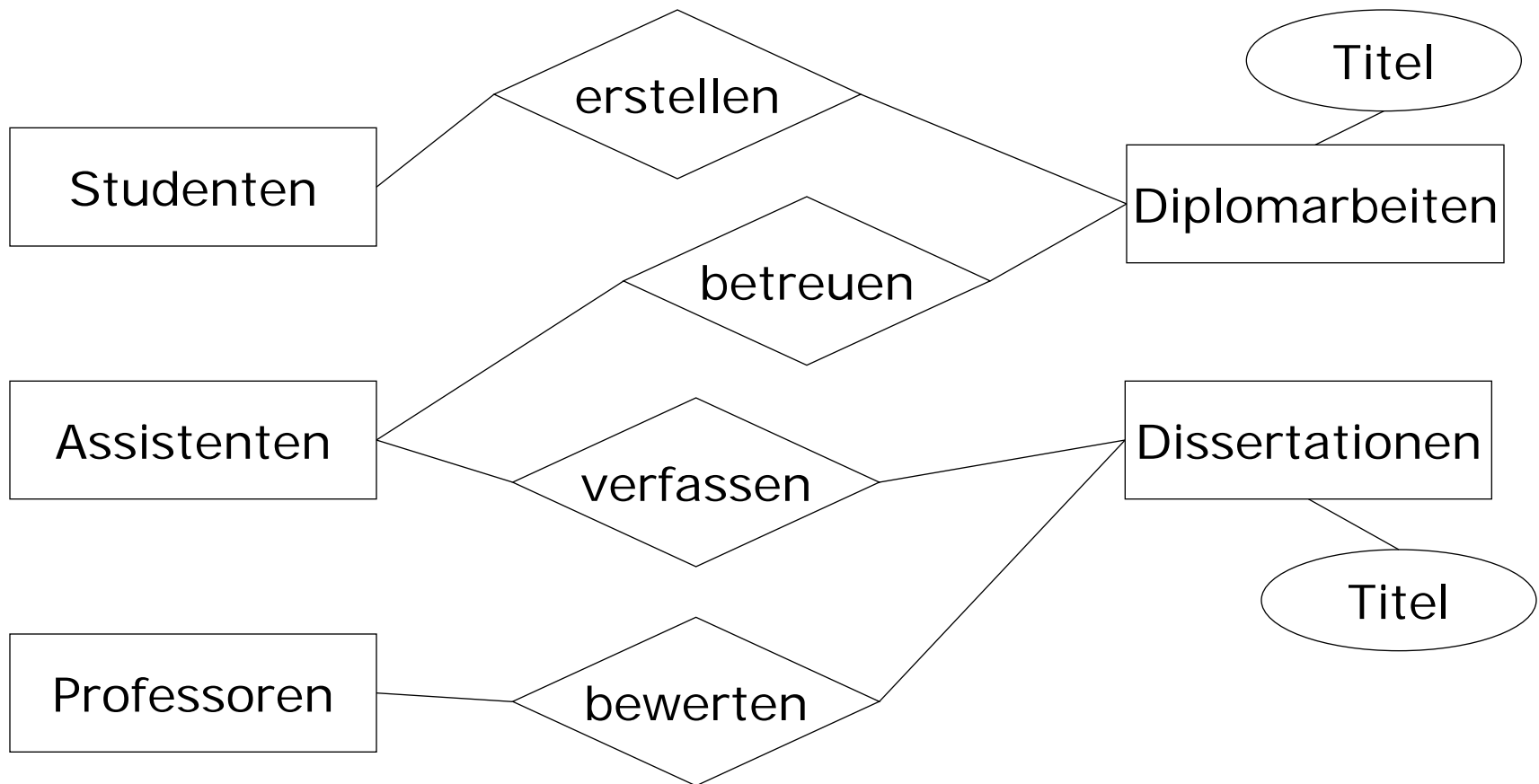
Folien nach Alfons Kemper
(TU München)



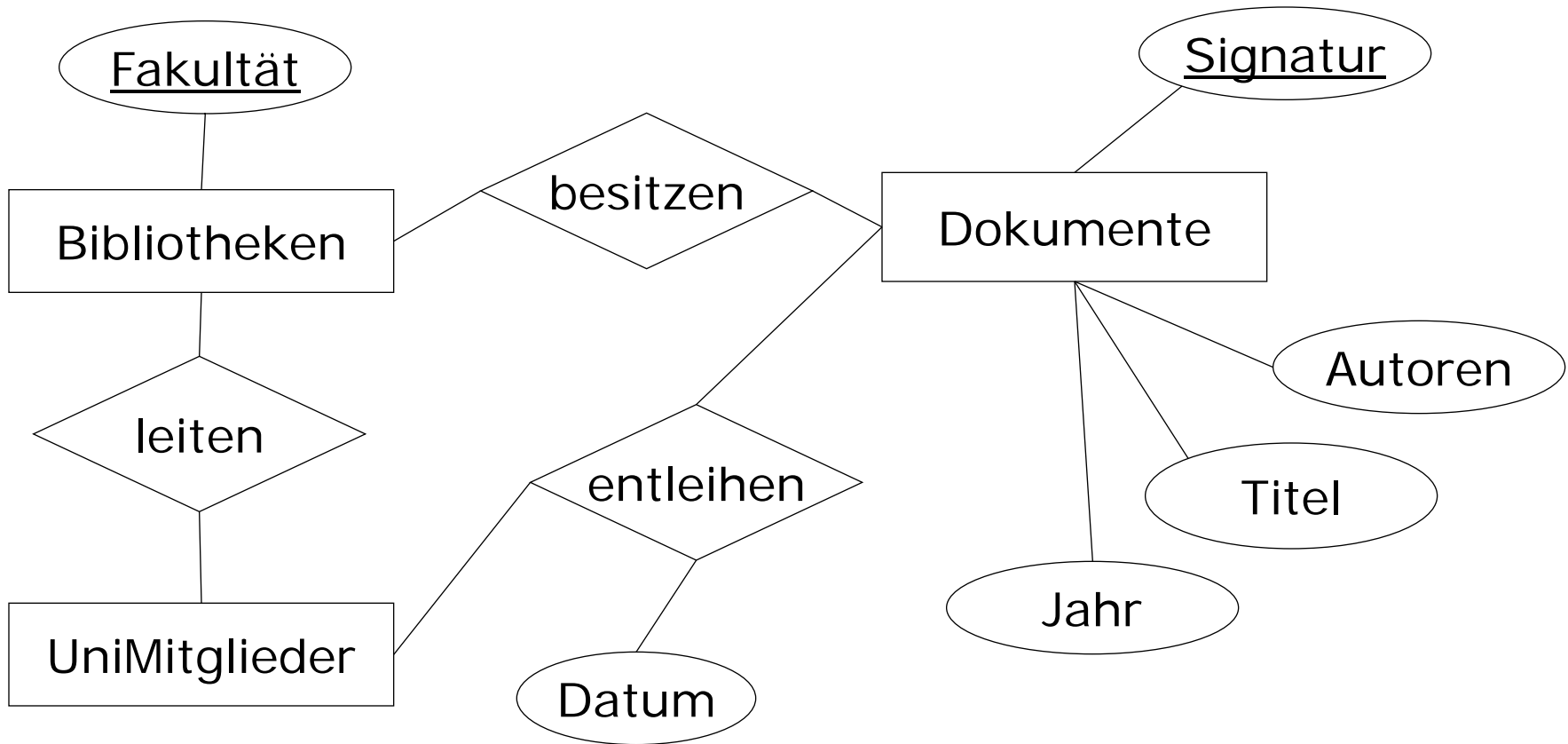


- Mögliche Konsolidierungsbäume zur Herleitung des globalen Schemas $S_{1,2,3,4}$ aus 4 Teilschemata $S_1, S_2, S_3,$ und S_4
 - Maximal hoher Konsolidierungsbaum
 - „links-tief“ (left-deep)
 - Minimal hoher Konsolidierungsbaum
 - Balanciert
 - Hybrides Vorgehen
- Beide Vorgehensweisen haben Vor- und Nachteile.

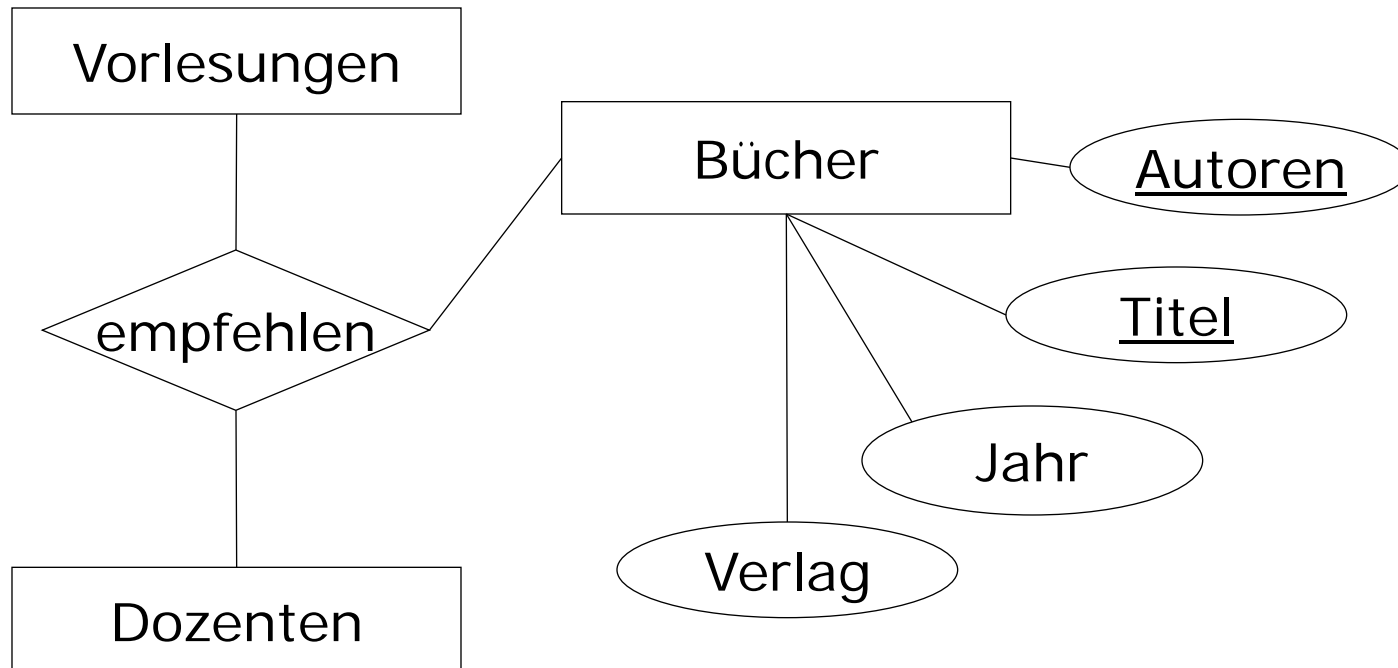
■ Sicht 1: Erstellung von Dokumenten als Prüfungsleistung



- Sicht 2: Bibliotheksverwaltung



■ Sicht 3: Buchempfehlungen für Vorlesungen



- Die Begriffe Dozenten und Professoren sind synonym.
- Der Entitytyp UniMitglieder ist eine Generalisierung von Studenten, Professoren und Assistenten.
- Fakultätsbibliotheken werden sicherlich von Angestellten (und nicht von Studenten) geleitet. Insofern ist die in Sicht 2 festgelegte Relationship „leiten“ revisionsbedürftig, sobald wir im globalen Schema ohnehin eine Spezialisierung von UniMitglieder in Studenten und Angestellte vornehmen.
- Dissertationen, Diplomarbeiten und Bücher sind Spezialisierungen von Dokumenten, die in den Bibliotheken verwaltet werden.
- Wir können davon ausgehen, dass alle an der Universität erstellten Diplomarbeiten und Dissertationen in Bibliotheken verwaltet werden.
- Die in Sicht 1 festgelegten Relationships „erstellen“ und „verfassen“ modellieren denselben Sachverhalt wie das Attribut Autoren von Büchern in Sicht 3.
- Alle in einer Bibliothek verwalteten Dokumente werden durch die Signatur identifiziert.