



**Hasso  
Plattner  
Institut**

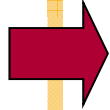
IT Systems Engineering | Universität Potsdam

Datenbanksysteme I  
ER Modellierung

27.4.2009

Felix Naumann

2



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



# Motivation

3

- Datenbankdesign als Skizze mittels ER-Modellierung
  - Welche Daten? Welche Beziehungen?
  - Nicht wie Daten erzeugt oder verändert werden.
- Skizzen heißen „Entity-Relationship-Diagramme“
  - ER-Diagramme
- In 1 Woche: Überführung von ER-Diagrammen in das relationale Modell
  - Relationen, Attribute, Integritätsbedingungen
- In 2 Wochen: Überführung von Relationenschemata in SQL Ausdrücke
  - **CREATE TABLE ...**

# Entwurfsaufgabe

4

- Datenhaltung für **mehrere** Anwendungssysteme und **mehrere** Jahre
- daher: besondere Bedeutung
- Anforderungen an Entwurf
  - Anwendungsdaten jeder Anwendung sollen aus Daten der Datenbank ableitbar sein.
    - ◇ Möglichst effizient
  - Nur „vernünftige“ (wirklich benötigte) Daten sollen gespeichert werden.
  - Nicht-redundante Speicherung

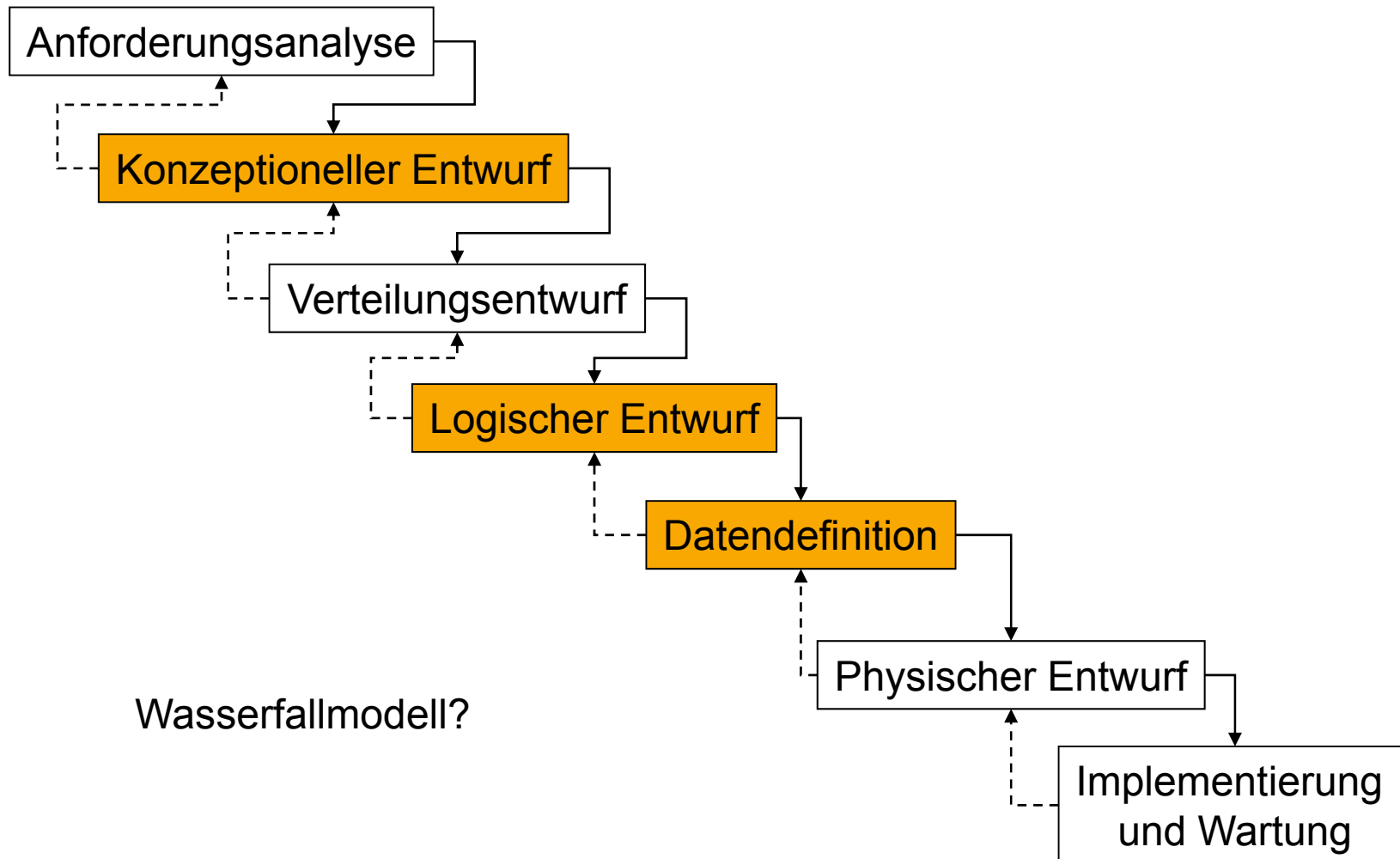
# Entwurfsprozess

5

- Abfolge von Entwurfsdokumenten
  - Von abstrakter Beschreibung
  - Bis tatsächlichen Realisierung in einem DBMS
  - Verschiedene Beschreibungsformalisten
    - ◇ ER, Relationenmodell, SQL DDL, usw.
- In jedem Schritt
  - Informationserhaltung
  - Konsistenzerhaltung

# Phasenmodell für den Datenbankentwurf

6



# Anforderungsanalyse

7

- Vorgehensweise
  - Sammlung des Informationsbedarfs in den Fachabteilungen
- Ergebnis
  - informale Beschreibung des Fachproblems
    - ◇ Texte, tabellarische Aufstellungen, Formblätter, ...
  - Trennen der Information über Daten (Datenanalyse) von den Information über Funktionen (Funktionsanalyse)
- „Klassischer“ DB-Entwurf
  - nur Datenanalyse und Folgeschritte
- Funktionsentwurf
  - siehe Methoden des Software Engineering

# Konzeptioneller Entwurf

8

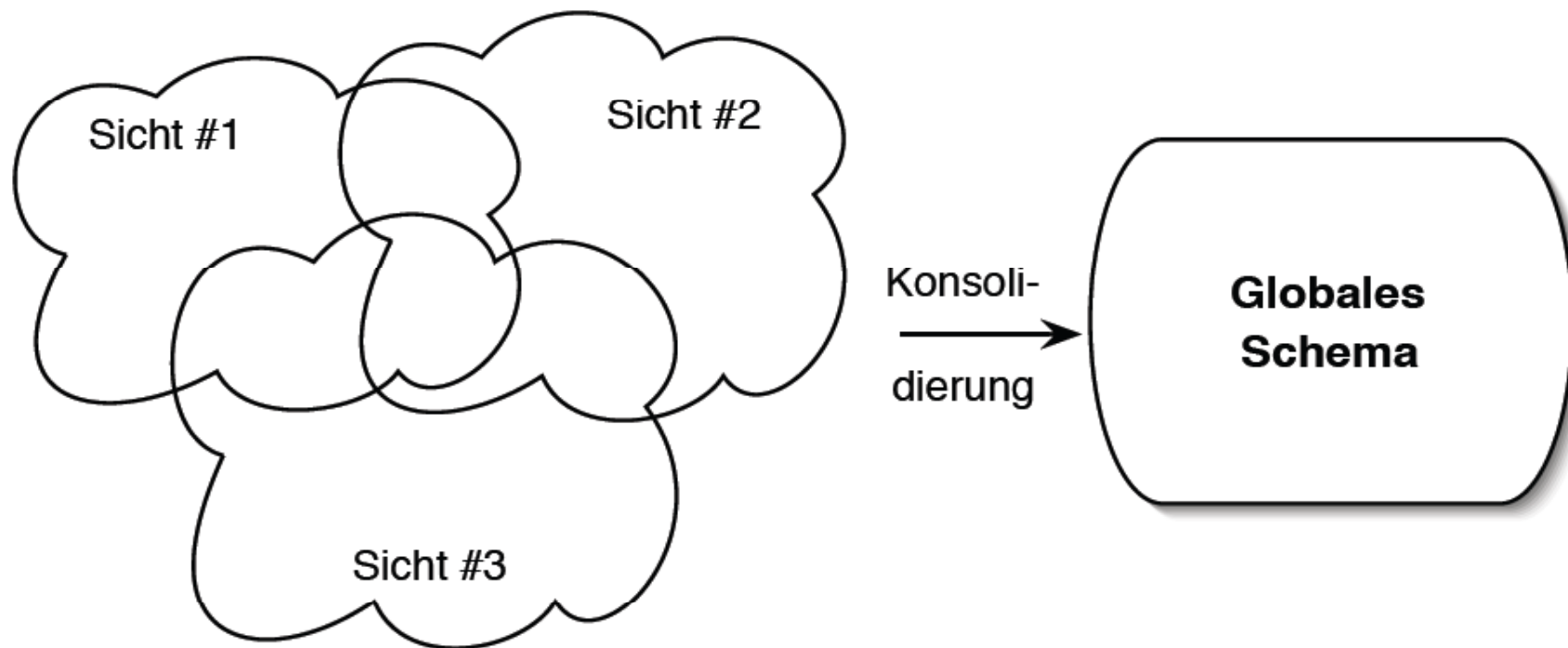
- Erste formale Beschreibung des Fachproblems
  - UoD: Universe of Discourse (Diskurswelt)
- Sprachmittel: semantisches Datenmodell
  - ER
- Vorgehensweise
  - Modellierung von Sichten z.B. für verschiedene Fachabteilungen
  - Analyse der vorliegenden Sichten in Bezug auf Konflikte
    - ◇ Namenskonflikte (Synonyme, Homonyme)
    - ◇ Typkonflikte
    - ◇ Bedingungskonflikte
    - ◇ Strukturkonflikte
  - Integration der Sichten in ein Gesamtschema
- Ergebnis
  - konzeptionelles Gesamtschema, z.B.(E)ER-Diagramm



# Sichtenintegration

9

- Analyse der vorliegenden Sichten in Bezug auf Konflikte
- Integration der Sichten in ein Gesamtschema



# Integrationskonflikte

10

- Namenskonflikte
  - Homonyme: Schloss, Kunde
  - Synonyme: Auto, KFZ, Fahrzeug
- Typkonflikte
  - verschiedene Strukturen für das gleiche Element
- Wertebereichskonflikte
  - verschiedene Wertebereiche für ein Element
- Bedingungskonflikte
  - z.B. verschiedene Schlüssel für ein Element
- Strukturkonflikte
  - gleicher Sachverhalt durch unterschiedliche Konstrukte ausgedrückt

# Verteilungsentwurf (Partitionierung)

11

- Sollen Daten auf mehreren Rechnern verteilt vorliegen, muss Art und Weise der verteilten Speicherung festgelegt werden.
  
- z.B. bei einer Relation
  - **KUNDE (KNr, Name, Adresse, PLZ, Konto)**
  - horizontale Verteilung
    - ◇ **KUNDE\_1 (KNr, Name, Adresse, PLZ, Konto)**  
where **PLZ < 50.000**
    - ◇ **KUNDE\_2 (KNr, Name, Adresse, PLZ, Konto)**  
where **PLZ >= 50.000**
  - vertikale Verteilung (Verbindung über KNr Attribut)
    - ◇ **KUNDE\_Adr (KNr, Name, Adresse, PLZ)**
    - ◇ **KUNDE\_Konto (KNr, Konto)**

# Logischer Entwurf

12

- Sprachmittel: Datenmodell des ausgewählten „Realisierungs“-DBMS
  - z.B. DB2, Oracle, ... => relationales Modell
  - Tamino => XML
- Vorgehensweise:
  - (automatische) Transformation des konzeptionellen Schemas
    - ◇ z.B. ER in relationales Modell
  - Verbesserung des relationalen Schemas anhand von Gütekriterien
    - ◇ Normalisierung, Redundanzvermeidung, ...
- Ergebnis: logisches Schema, z.B. Sammlung von Relationenschemata
- Siehe nächsten Foliensatz

# Datendefinition

13

- Umsetzung des logischen Schemas in ein konkretes Schema
- Sprachmittel:
  - DDL und DML eines DBMS
    - ◇ z.B. Oracle, DB2, SQL Server
  - Datenbankdeklaration in der DDL des DBMS
  - Realisierung der Integritätssicherung
  - Definition der Benutzersichten

**CREATE TABLE ...**

**CREATE VIEW ...**

# Physischer Entwurf

14

- Ergänzen des physischen Entwurfs um Zugriffsunterstützung zur Effizienzverbesserung
  - z.B. Definition von Indizes
  - **CREATE INDEX ...**
- Index
  - Zugriffspfad: Datenstruktur für zusätzlichen, schlüsselbasierten Zugriff auf Tupel (<Schlüsselattributwert, Tupeladresse>)
  - Meist als B\*-Baum realisiert
- Beispiel
  - Tabelle mit 10 GB Daten, Festplattentransferrate ca. 10 MB/s
  - Operation: Suchen eines Tupels (Selektion)
  - Implementierung: sequentielles Durchsuchen
  - Aufwand:  $10.240/10 = 1.024 \text{ sec.} = 17 \text{ min.}$

**Nicht in dieser Vorlesung**

# Implementierung und Wartung

15

- **Wartung des DBMS**
  - Parameter, Festplatten, etc.
- **Database Tuning**
  - Weitere Optimierung der physischen Ebene
- **Anpassung an neue Anforderungen**
- **Anpassung an neue Systemplattformen**
- **Portierung auf neue Datenbankmanagementsysteme**
  
- **Kostenaufwändigste Phase**
- **Software Engineering**

**Nicht in dieser Vorlesung**

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





# Das Entity-Relationship-Modell

17

- Nach Peter P. Chen 1976
  - The entity-relationship model – towards a unified view of data.  
ACM TODS
- Standardmodell in der frühen Entwurfsphase

## The Entity-Relationship Model—Toward a Unified View of Data

PETER PIN-SHAN CHEN

Massachusetts Institute of Technology

---

A data model, called the entity-relationship model, is proposed. This model incorporates some of the important semantic information about the real world. A special diagrammatic technique is introduced as a tool for database design. An example of database design and description using the model and the diagrammatic technique is given. Some implications for data integrity, information retrieval, and data manipulation are discussed.

The entity-relationship model can be used as a basis for unification of different views of data: the network model, the relational model, and the entity set model. Semantic ambiguities in these models are analyzed. Possible ways to derive their views of data from the entity-relationship model are presented.

Key Words and Phrases: database design, logical view of data, semantics of data, data models, entity-relationship model, relational model, Data Base Task Group, network model, entity set model, data definition and manipulation, data integrity and consistency

CR Categories: 3.50, 3.70, 4.33, 4.34

---



# Begriffe

18

- Entity
  - Ein Ding / Objekt der realen oder der Vorstellungswelt
  - Nicht direkt darstellbar, sondern nur über Eigenschaften beobachtbar
- Entitytyp (*entity set*)
  - Eine Klasse für gleichartige Objekte
- Relationship
  - Beschreibt Beziehungen zwischen Entities
  - Meist binär
- Relationshiptyp
  - Eine Klasse für gleichartige Beziehungen
- Attribut
  - repräsentiert eine Eigenschaft von Entities oder von Relationships
  - Zunächst nur primitive Datenwerte (String, Integer, ...) und Operationen darauf
    - ◇ Später auch komplexe Attribute

# Entity-Relationship-Diagramm

19

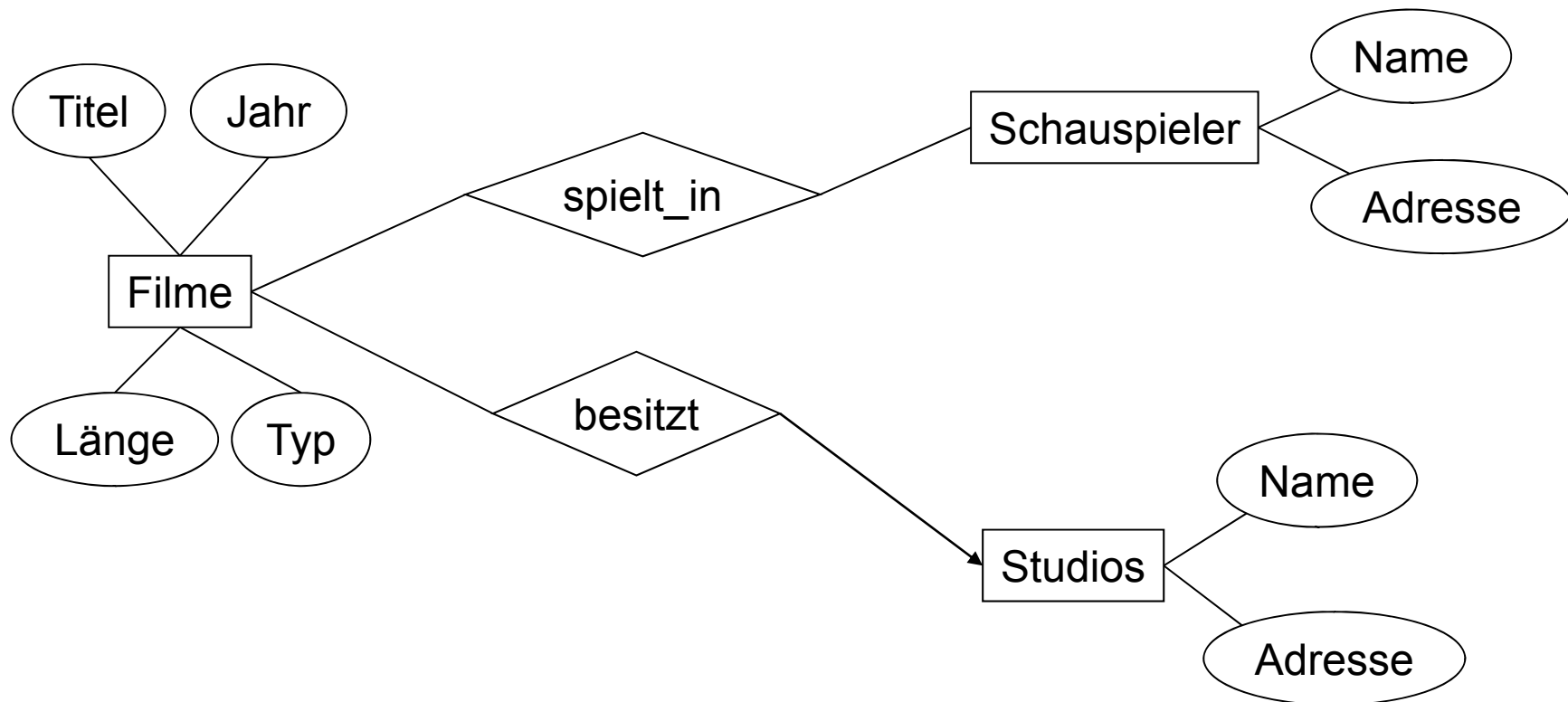
- Entitytyp: Rechteck
- Relationshiptyp: Raute
- Attribut: Oval
  
- Kanten verbinden Entitytypen mit Attributen
- Kanten verbinden Entitytypen mit Relationshiptypen

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



# Entity-Relationship-Diagramm

21



# Instanz eines ER-Diagramms

22

- ER-Diagramme beschreiben Datenbanken, die eine Instanz haben (werden).
- Der „Wert“ eines Entitytypen ist die (endliche) Menge der zugehörigen Entities.
- Jeder Entity hat bestimmte Werte für seine Attribute.
- Die Instanz eines n-ären Relationshiptypen ist eine Menge von Listen der Länge n.
  
- Dies alles ist nur abstrakte Denkhilfe.
  - Modellierung im relationalen Modell unterscheidet sich.
  - Speicherung in einem DBMS unterscheidet sich.

# Instanz eines ER-Diagramms

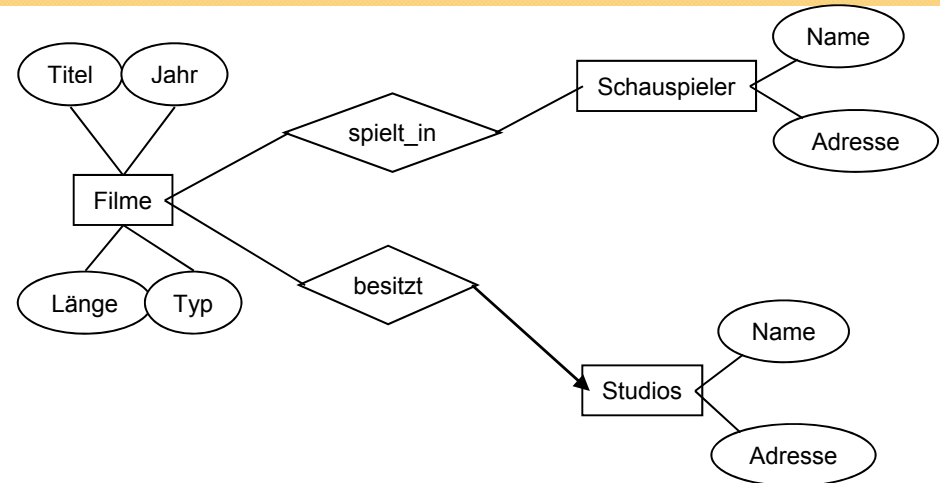
23

Filme

Titel	Jahr	Länge	Typ
Basic Instinct	1992	127	Farbe
Total Recall	1990	113	Farbe
Dead Man	1995	121	s/w

Schauspieler

Name	Adresse
Sharon Stone	Hollywood
Arnold Schwarzenegger	Sacramento
Johnny Depp	Paris



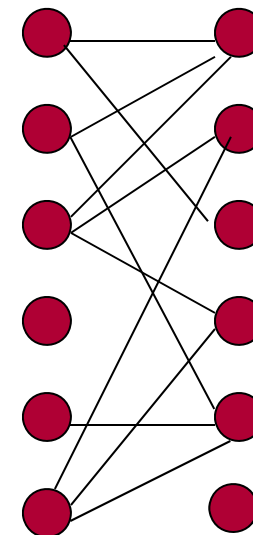
spielt\_in

Name	Titel
Sharon Stone	Total Recall
Sharon Stone	Basic Instinct
Arnold Schwarzenegger	Total Recall
Johnny Depp	Dead Man

# Kardinalitäten von Relationships

24

- Allgemein: Eine binäre Relationship kann beliebig viele Entities des einen Typen mit beliebig vielen des anderen Typen verbinden.
  - Ein Schauspieler kann in mehreren Filmen spielen.
  - In einem Film spielen mehrere Schauspieler.
- „Kann“: D.h. nicht jeder Entity muss mit einem anderen verbunden sein.
  - In einem Animationsfilm spielen keine Schauspieler.
- m:n Beziehung
- Einschränkungen („Spezialfälle“)
  - 1:n Beziehung
  - 1:1 Beziehung

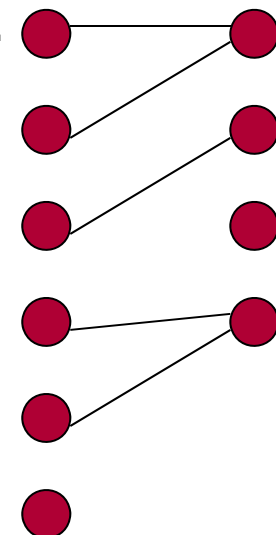




# 1:n Relationships

25

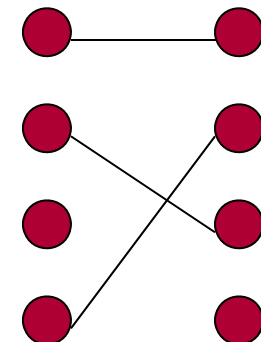
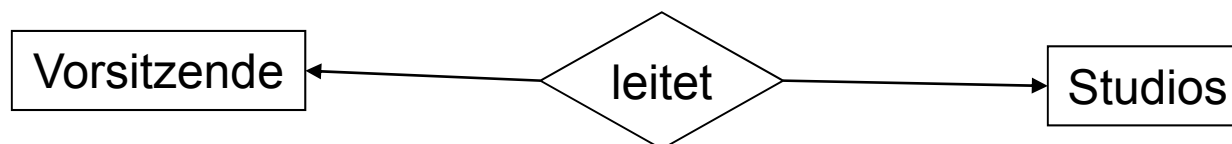
- Ein Entity vom Typ E kann mit beliebig vielen Entities des Typs F verbunden sein.
- Ein Entity vom Typ F kann mit höchstens einem Entity des Typs E verbunden sein.
- Beispiel
  - Ein Studio kann die Rechte an mehreren Filmen besitzen.
  - Ein Film kann nur von einem Studio besessen werden.
- „Kann“: D.h. nicht jeder Entity muss mit einem anderen verbunden sein.
  - Ein neues Studio besitzt noch keinen Film.
- Darstellung mittels eines Pfeils zur „1er“ Seite.



# 1:1 Relationships

26

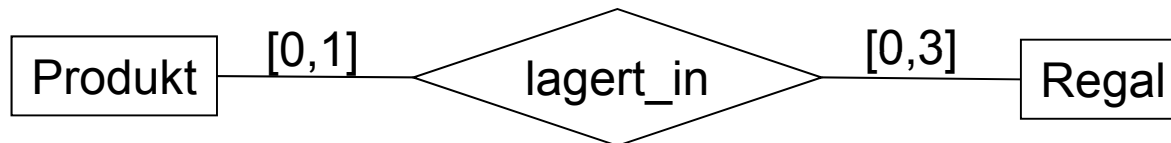
- Ein Entity vom Typ E kann mit höchstens einem Entity des Typs F verbunden sein.
- Ein Entity vom Typ F kann mit höchstens einem Entity des Typs E verbunden sein.
- Beispiel
  - Ein Studio kann nur von einem Vorsitzenden geleitet werden.
  - Ein Vorsitzender kann nur ein Studio leiten.
- „Kann“: D.h. nicht jeder Entity muss mit einem anderen verbunden sein.
  - Ein Studio kann (vorübergehend) keinen Vorsitzenden haben.



# Kardinalität: Zwei andere Notationen

27

- [min,max]-Notation (Lookup-Semantik)
  - Schränkt die möglichen Teilnahmen von Instanzen der beteiligten Entitytypen an der Beziehung ein.
    - ◇ Wie oft steht ein Entitytyp in Beziehung?




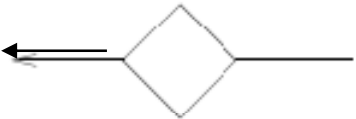

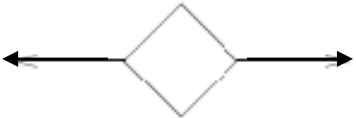
- ◇ (Ausverkaufte) Produkte sind keinem Regal zugeordnet.
- ◇ Ein Produkt kann höchstens in einem Regal gelagert werden.
- ◇ Ein Regal kann leer sein und maximal 3 Produkte Lagern.

- Numerische Notation (Partizipationssemantik)



# Kardinalität: Andere Notationen

28

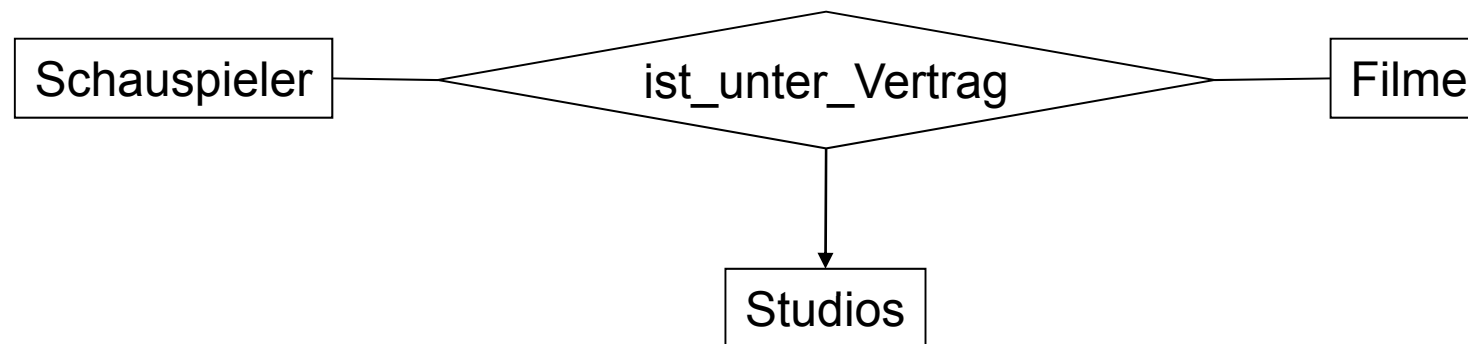
Beziehungsm.	(min1, max1)	(min2, max2)	grafische Notation
many-to-many	(0,*)	(0,*)	
one-to-many	(0,*)	(0,1)	
many-to-one	(0,1)	(0,*)	
one-to-one	(0,1)	(0,1)	

Nach Ingo Schmitt (TU Cottbus)

# n-äre Relationships

29

- Relationships zwischen mehr als zwei Entities
- Ein Schauspieler steht bei einem Studio für eine bestimmten Film unter Vertrag



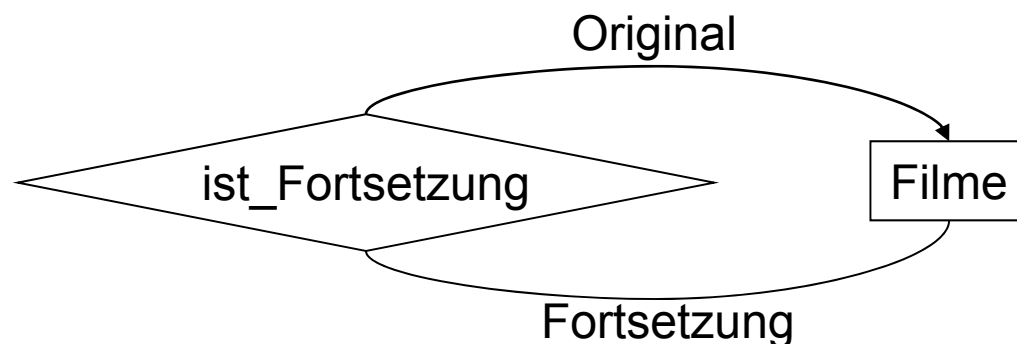
- Instanz der Relationship kann man als Tripel darstellen.
- Kardinalitäten: Jede Kombination von Schauspieler und Film kann nur mit einem Studio in Beziehung stehen.

# Rollen von Relationships

30

Entitytypen können mehr als einmal in einer Relationship auftauchen.

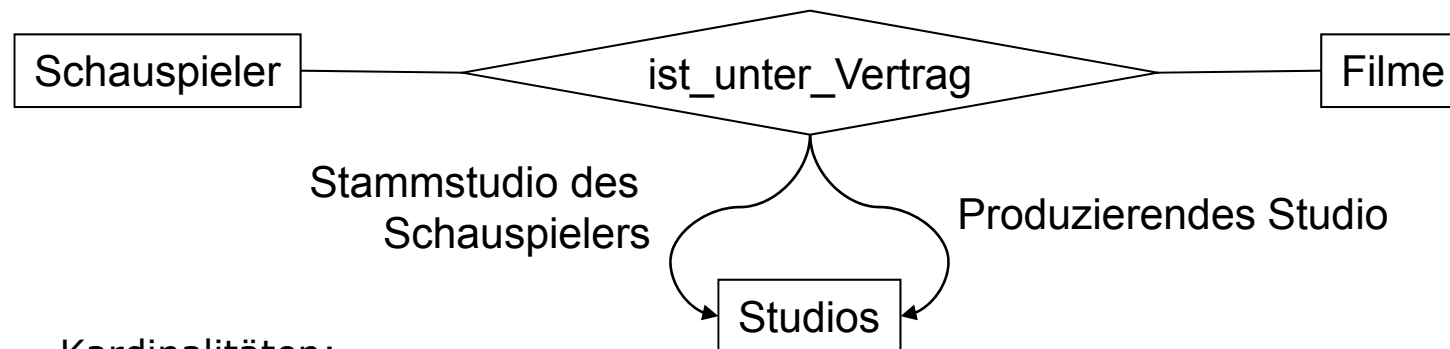
- Entsprechend mehrere Kanten
- Jede Kante entspricht einer anderen **Rolle**.
- Die Kanten werden mit den entsprechenden Rollen annotiert.
- Bsp: Fortsetzungen von Filmen
  - Beverly Hills Cop (& II, III, ...)
  - Ein Film kann viele Fortsetzungen haben.
  - Eine Fortsetzung hat aber nur ein Original.



# Rollen von Relationships

31

- Stammstudio eines Schauspielers erlaubt einem anderen Studio den Schauspieler für einen bestimmten Film auszuleihen



- Kardinalitäten:
  - Gegeben Schauspieler, Film und produzierendes Studio, ist das ausleihenden Stammstudio eindeutig (höchstens ein Studio ist das ausleihende Stammstudio).
  - Gegeben Schauspieler, Film und Stammstudio ist das produzierende Studio eindeutig (höchstens ein Studio kann einen Film produzieren).
  - Gegeben Schauspieler, Stammstudio und produzierendes Studio könnte es mehrere Filme geben, die in dieser Konstellation gedreht werden.
  - Gegeben Film, Stammstudio und produzierendes Studio könnte es mehrere Schauspieler geben, die in dieser Konstellation ausgeliehen werden.

# Konvertierung in binäre Relationships

32

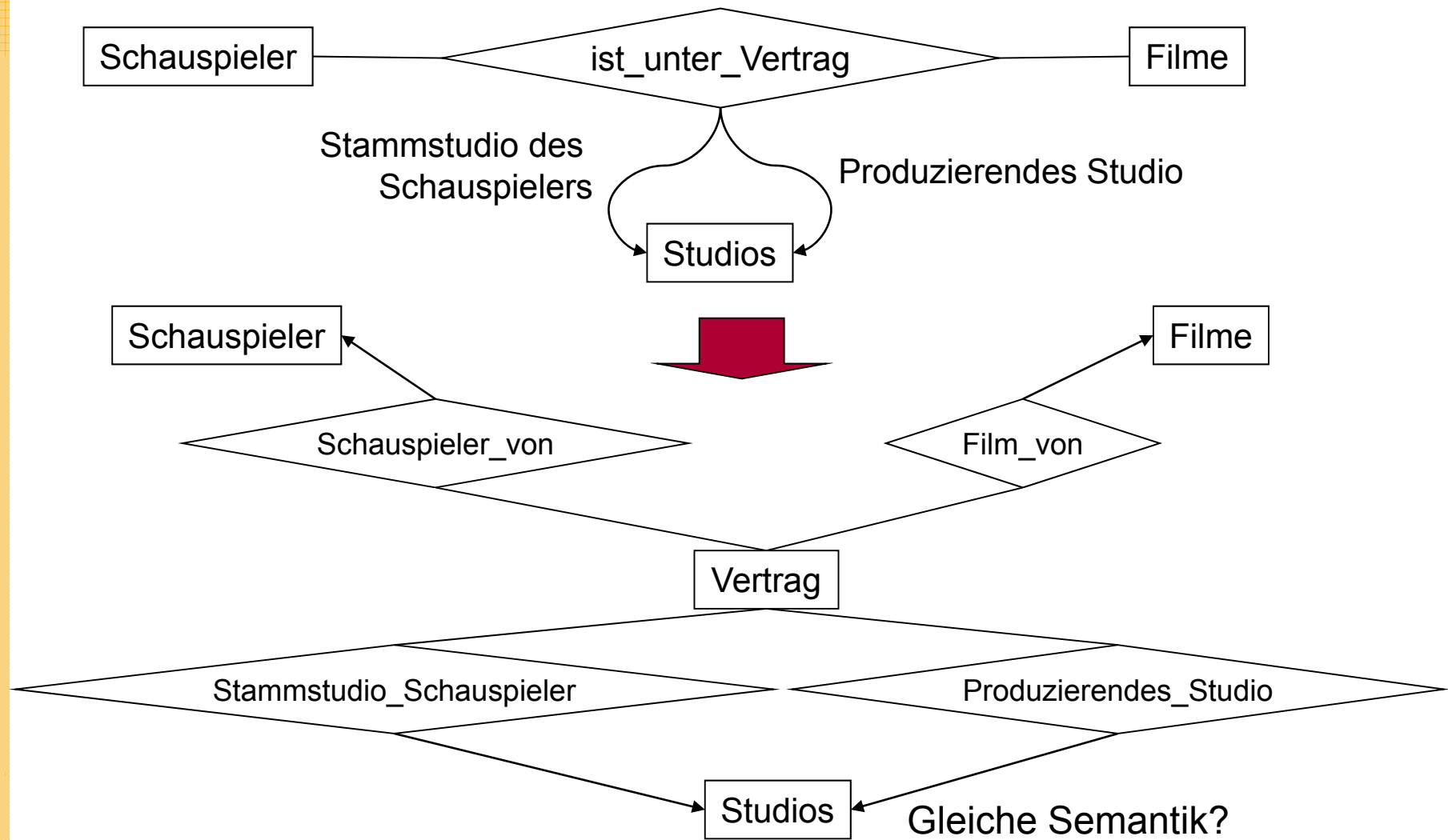
## Umwandlung n-ärer Relationships in binäre Relationships

- Erstellung eines neuen, verbindenden Entitytyps
- Neue n:1 Relationships zwischen dem neuen Entitytyp und den alten Entitytypen
- Falls ein Entitytyp mehrere Rollen spielt, entsteht pro Rolle ein Relationshiptyp.
- Attribute des Relationshiptyps werden an den neuen Entitytyp angehängt.



# Konvertierung in binäre Relationships

33



Gleiche Semantik?  
Schlüssel (nächster Foliensatz)

# Attribute an Relationships

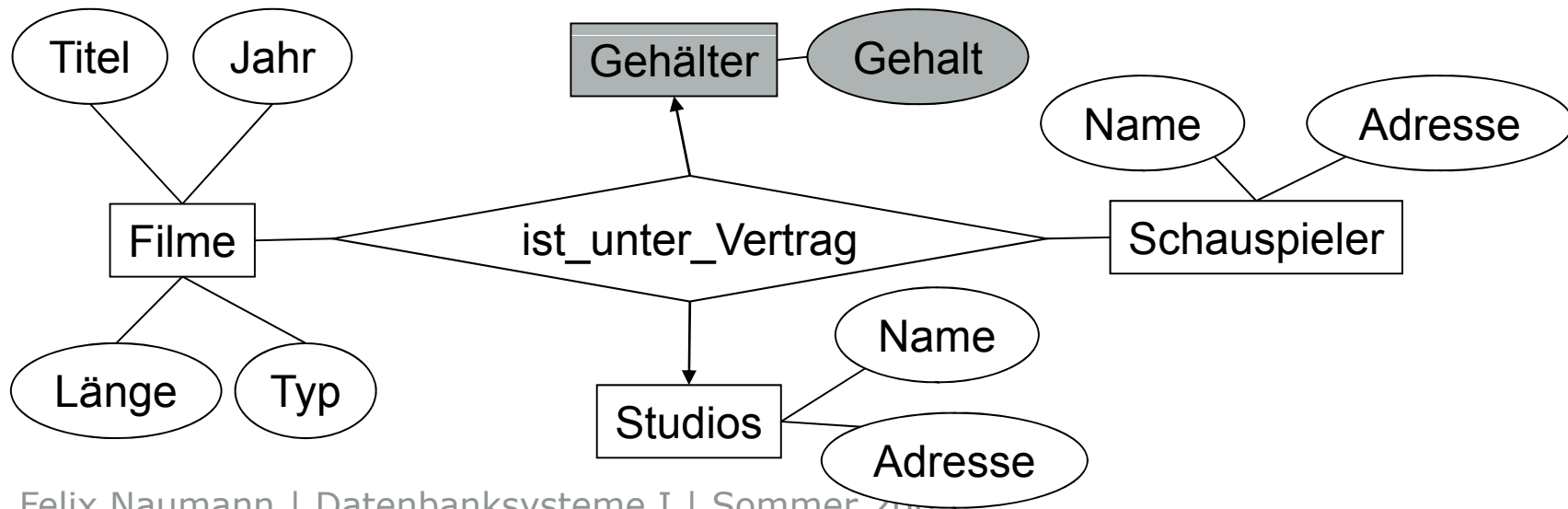
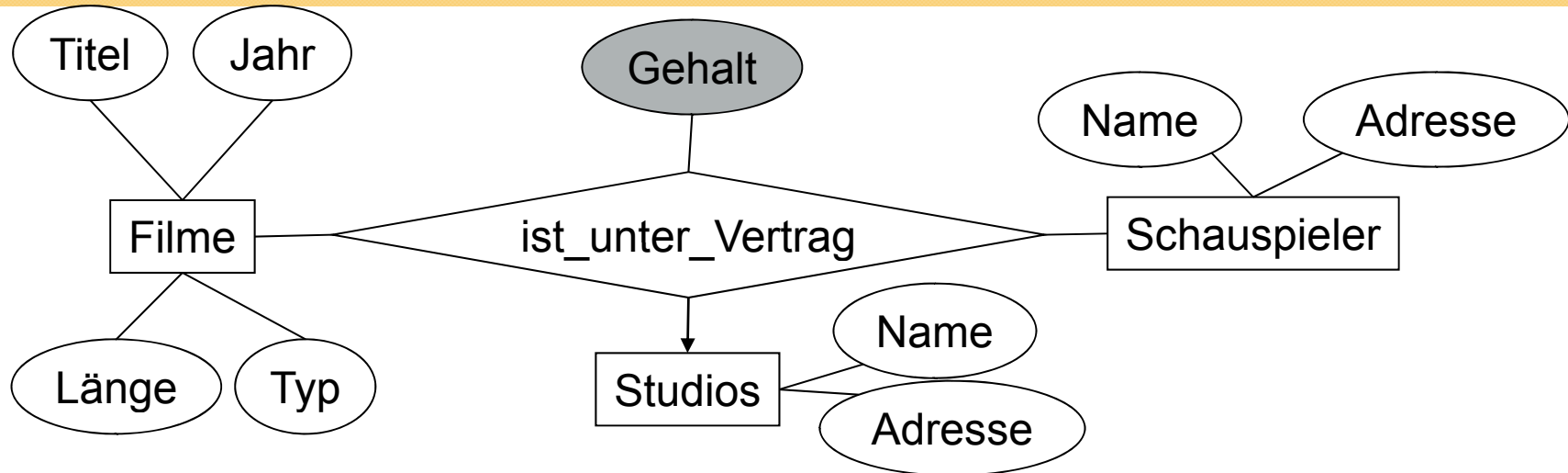
34

In manchen Fällen ist es hilfreich, Relationships Attribute zuzuordnen

- Bsp: In dem Drehvertrag wird ein Gehalt festgestellt.
  - Zuordnung zu Schauspieler? Er könnte für verschiedene Filme unterschiedliche Gehälter bekommen.
  - Zuordnung zum Film? Verschiedene Schauspieler könnten unterschiedliche Gehälter bekommen.
  - Zuordnung zum Studio? Es könnte verschiedenen Schauspielern unterschiedliche Gehälter zahlen.

# Attribute an Relationships

35



# Die IST-Beziehung

36

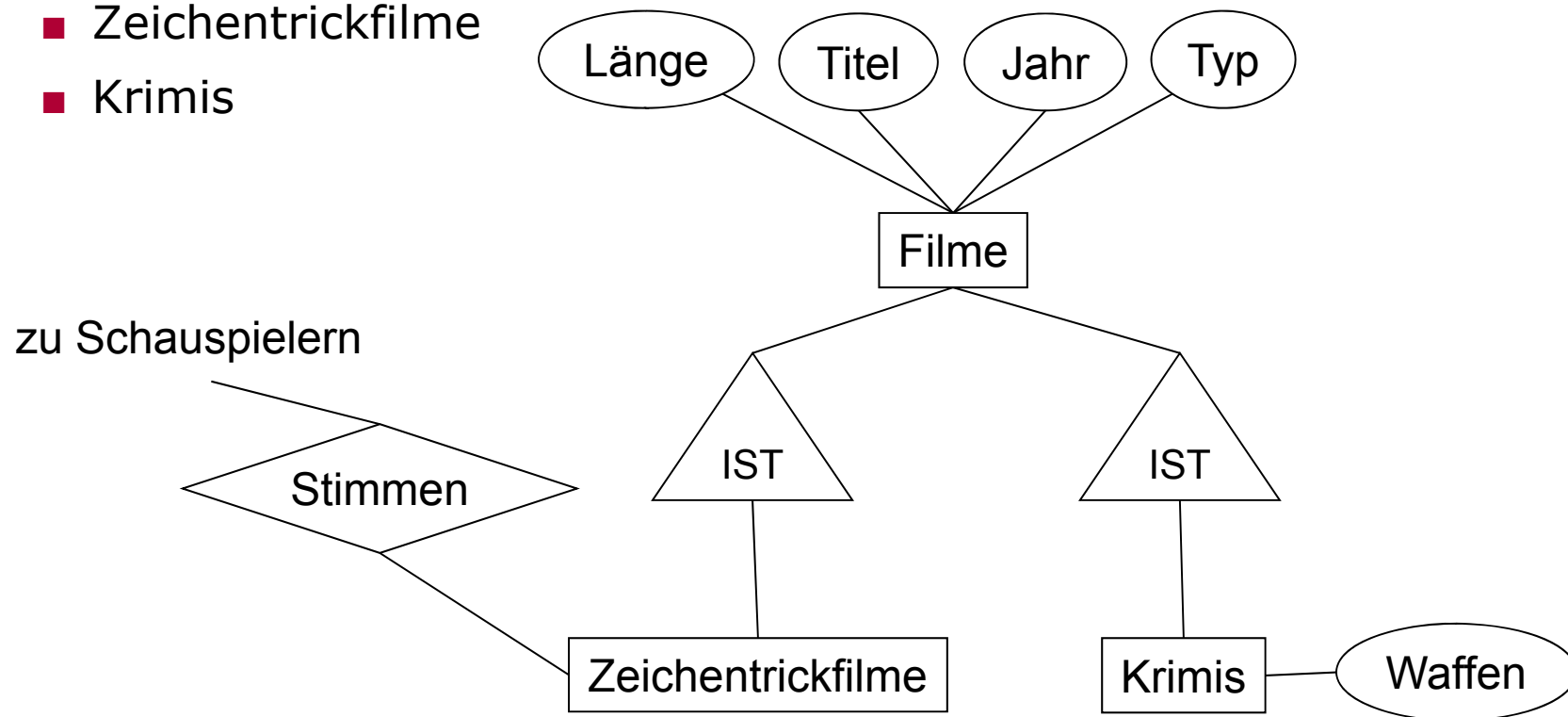
- Subklasse
  - Spezialfall / Spezialisierung
  - Weniger Entities
  - Mehr Attribute
  - Eventl. mehr Relationships
- Besonderer Relationshiptyp
  - IST (is-a)
  - Darstellung durch Dreieck
    - ◇ Spitze zeigt zur Superklasse
  - Immer 1:1
    - ◇ Trotzdem keine Pfeile

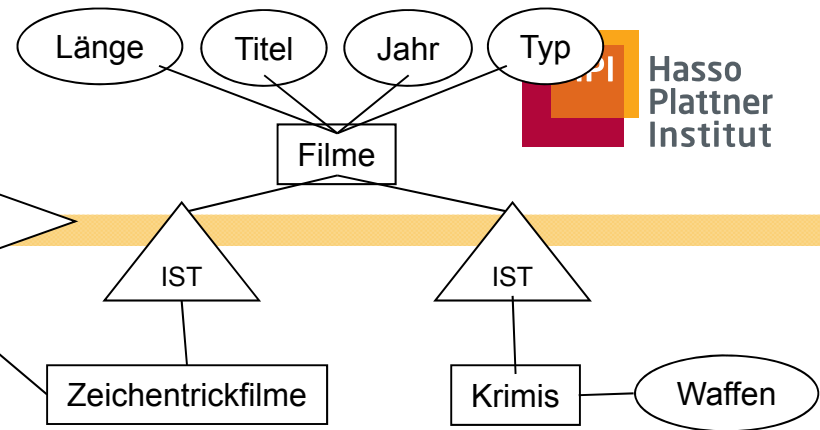
# Die IST-Beziehung

37

## Subklassen zu Filmen

- Zeichentrickfilme
- Krimis

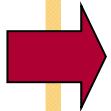




- IST-Beziehungen nur als Bäume
  - Keine Mehrfachvererbung
- Ein Entity kann aus mehreren Komponenten des IST-Baumes bestehen.
  - „Krieg der Sterne“ hat vier Attribute.
  - „Cinderella“ hat vier Attribute und „Stimmen“-Relationships.
  - „Der dritte Mann“ hat vier Attribute und zusätzlich das Attribut „Waffen“.
  - „Roger Rabbit“ hat vier Attribute, zusätzlich das Attribut „Waffen“ und „Stimmen“-Relationships.
- Anders als objekt-orientierte Modelle
  - In OO sind Objekte immer in genau einer Klasse; Subklassen erben von Superklasse(n).
  - In ER sind Entities in allen Subklassen repräsentiert, in die sie gehören.
  - In ER ist ein Entity in einer Subklasse auch automatisch in den Superklassen repräsentiert.



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



# Nebenbedingungen (Constraints)

40

- Schlüssel
  - Ein oder mehrere Attribute
  - Werte identifizieren eindeutig ein Entity.
- Referentielle Integrität
  - Existenz des referenzierten Entities
  - Entspricht „dangling pointer“
- Domänen
  - Einschränkung des Wertebereichs
- Allgemeine Nebenbedingungen (*assertions*)
  - Z.B. nicht mehr als 10 Schauspieler pro Film
- Nebenbedingungen sind Teil des Schemas. Sie leiten sich nicht aus den Daten ab!



# Schlüssel

41

- Ein Schlüssel ist eine (minimale) Menge von Attributen eines Entitytyps, für die gilt, dass keine zwei Entities gleiche Werte in allen Schlüsselattributen haben.
  - Einige Attributwerte können übereinstimmen.
  - Oft nur ein Attribut
- Für jeden Entitytyp muss ein Schlüssel angegeben werden.
- Es kann mehr als einen Schlüssel für einen Entitytyp geben.
  - Üblich: Primärschlüssel auswählen
- Bei IST-Beziehungen muss die Wurzel-Superklasse sämtliche Schlüsselattribute enthalten.
  
- Darstellung durch Unterstreichen der Attributnamen

# Schlüssel

42

- Filme
  - „Titel“ als Schlüssel?
    - ◇ King Kong
  - Titel und Jahr als Schlüssel?
    - ◇ Eventuell kann ein Film dann nicht gespeichert werden.
    - ◇ In IMDB: 275 Doppelte
- Schauspieler
  - „Name“ oder „Name“ und „Adresse“
- Studio
  - Name
- In der Praxis: Modellierung eines speziellen, numerischen Schlüsselattributs.
  - ISBN, SSN, Imma-Nummer,...

TITLE	DATE	3
Shogun	1980	2
Shriners Hospit...	2000	2
Sinatra: The Cl...	2002	2
Sirens	1995	2
Skag	1980	2
Skin Deep	2000	2
Skin Deep	2003	2
Skyport	1960	2
Something Is O...	1988	2
Space Invaders	1999	2
Sparks	1998	2
Special Delivery	2003	2
Spring	2002	2
Srečna porodica	1979	2
Sternbergs - Är...	1999	2
Still Life	2004	2
Straight Up	1997	2
Strange Days	1998	2
Strange World	1999	2
Striker	1976	2
Studio 5-B	1989	2
Summerland	2004	2
Sunset Beat	1990	2
Supercarrier	1988	2
Sweet Heart	1996	2
Take Five	1987	2
Talk to Me	2000	2
Tarzan: The E...	1996	2
Taskeholdet	1997	2
Temptation	1968	2
Tenshi no tama...	1985	2
Tenspeed and ...	1980	2
Test, The	2001	2
Top of the Hill	1989	2

# Wo ist der Schlüssel?

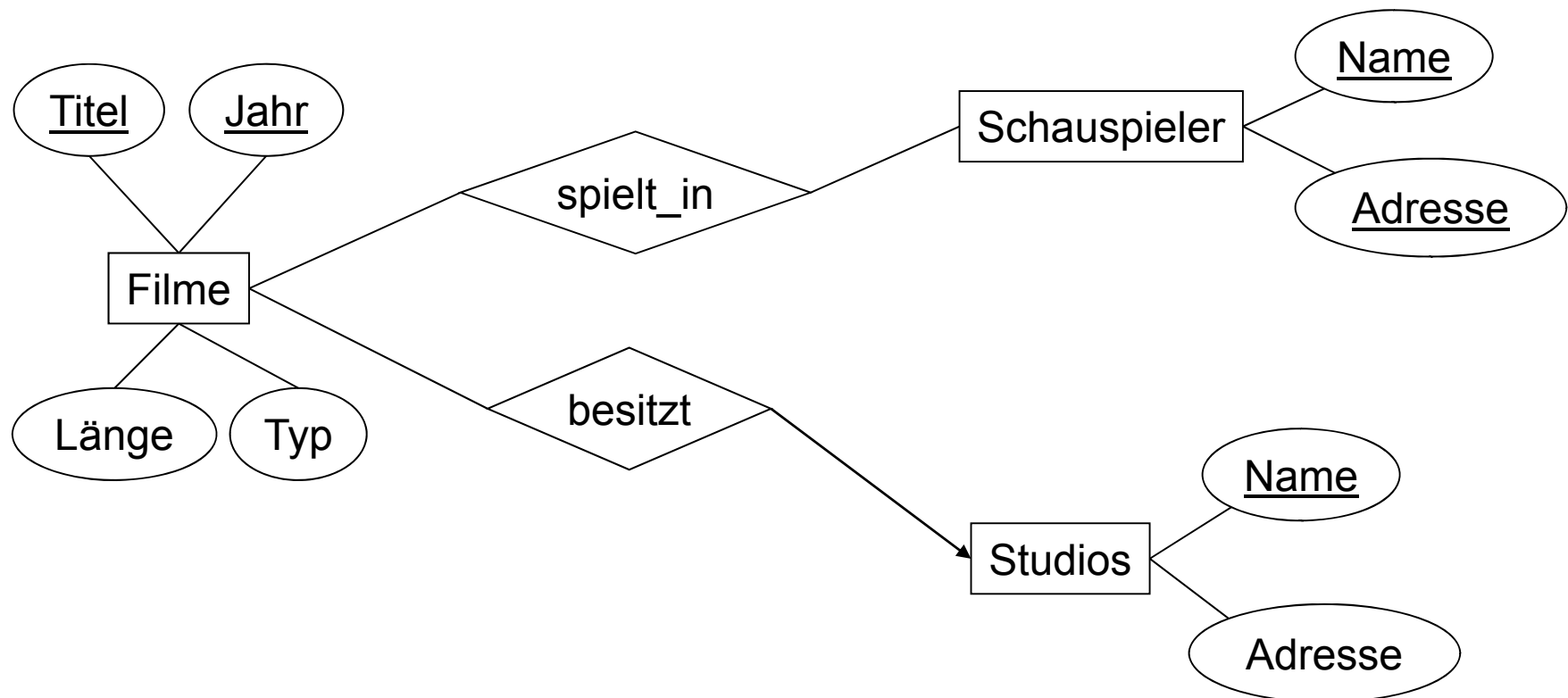
43

The screenshot shows a Mozilla Firefox browser window displaying the IMDb page for the movie "Star Wars (1977)". The browser's address bar shows the URL "http://www.imdb.com/title/tt0076759/". The IMDb page features a search bar on the left with "All" selected and a "go" button. Below the search bar, it shows "Result: 1 of 108" and navigation links. The main content area displays the movie title "Star Wars (1977)" with a small image of the movie's cover. Below the title, it lists the director as "George Lucas" and the writing credits as "George Lucas (written by)". There are buttons for "Add to MyMovies", "Photo Gallery", and "IMDbPro Details". A "Photo Gallery" section shows a row of four movie stills, with a "more" link to the right. The genre is listed as "Action / Adventure / Fantasy / Sci-Fi (more)" and the tagline is "A long time ago in a galaxy far, far away... (more)". On the right side of the page, there is a "SHOP STAR WARS" section with various media options like DVD, VHS, and CD for different countries, and an Amazon.com link.

# Schlüssel

44

- In ER-Diagrammen kann nur ein Schlüssel notiert werden.
  - Der sich über mehrere Attribute eines Entitytyps erstrecken kann.



# Referentielle Integrität

45

- Schlüssel: Höchstens ein bestimmter Wert für ein Attribut
  - Bzw. höchstens eine Wertekombination bei mehreren Attributen im Schlüssel
- Referentielle Integrität: Genau ein bestimmter Wert
  - Bsp. n:1 Relationship zwischen „Filme“ und „Studios“
    - ◇ Ein Film kann zu höchstens einem Studio gehören.
    - ◇ Aber ein Film muss zu keinem Studio gehören.
    - ◇ Auch wenn ein Film zu einem Studio gehört, muss dieses nicht in der DB repräsentiert sein.
    - ◇ Referentielle Integrität erzwingt die Existenz und Repräsentation des Studios
- „Erzwingen“
  - Bei Einfügen/Ändern eines Films muss entsprechendes Studio vorhanden sein.
  - Ein Studio darf nicht gelöscht werden, solange es noch Filme besitzt.
  - Oder: Wenn ein Studio gelöscht wird, werden auch alle entsprechenden Filme gelöscht.
  - Verschiedene Einstellungen im DBMS

# Referentielle Integrität

46



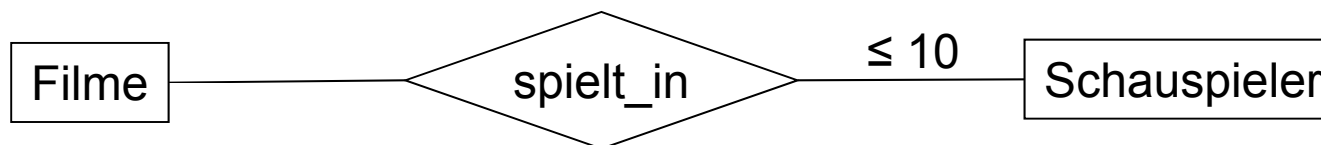
- Jeder Film muss von genau einem Studio besessen werden.
- Jeder Vorsitzende muss genau ein Studio leiten.
- Jedes Studio wird von höchstens einem Vorsitzenden geleitet.
  - Eventuell aber von keinem
- Ein Studio kann mehrere Filme besitzen.

# Weitere Nebenbedingungen

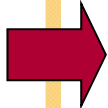
47

## Ohne formale Notation im ER-Diagramm

- Datentyp
  - Integer, String, ...
- Wertebereich / Domäne
  - $\leq 100$ , {Krimi,Doku,Zeichentrick}
- Länge eines Attributes
  - Stringlänge  $< 25$
- Kardinalität von Relationships
  - Höchstens 10 Schauspieler pro Film
  - Oder mittels min/max Notation



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





# Motivation

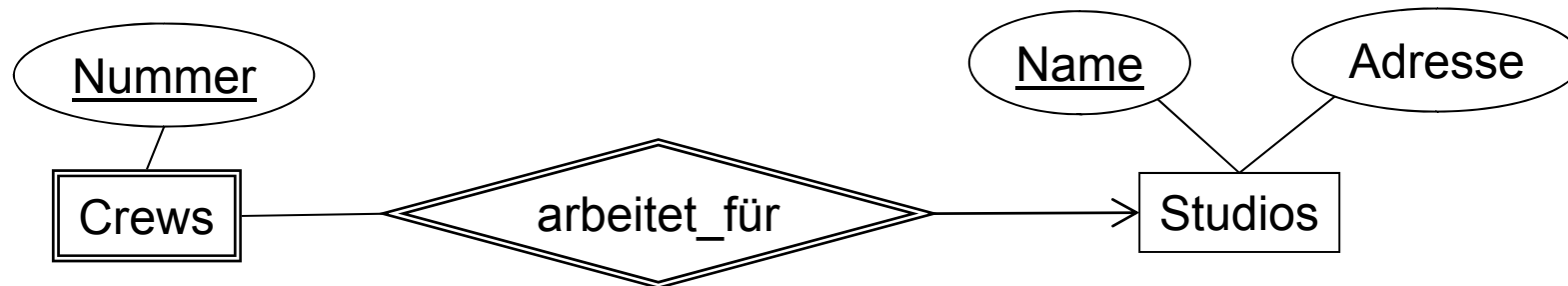
49

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.

## Beispiel

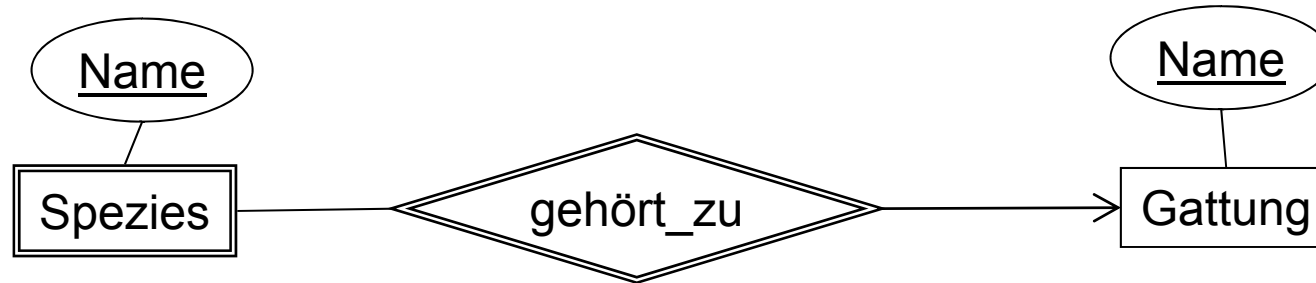
50



- 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.

# Beispiel

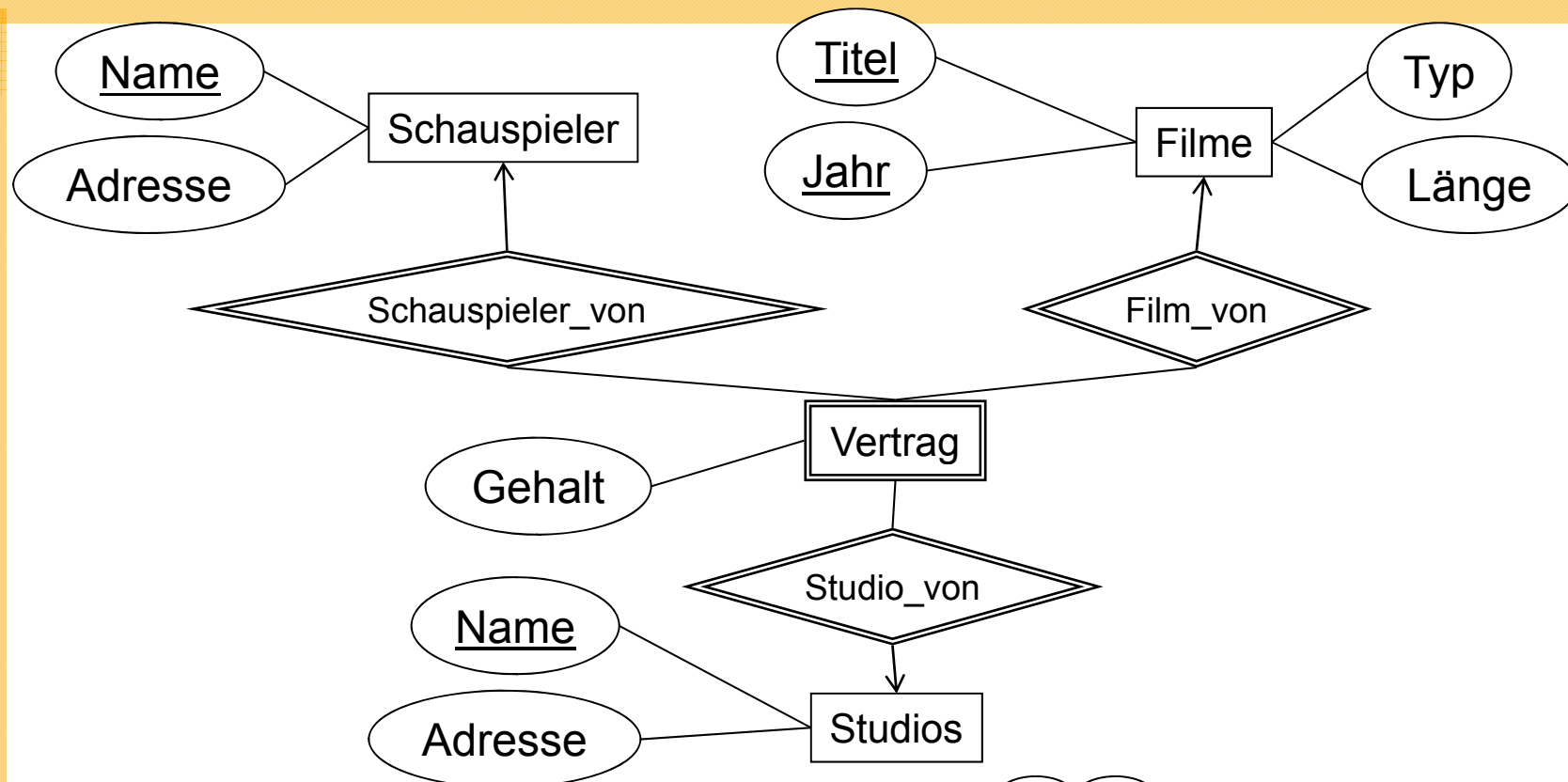
51



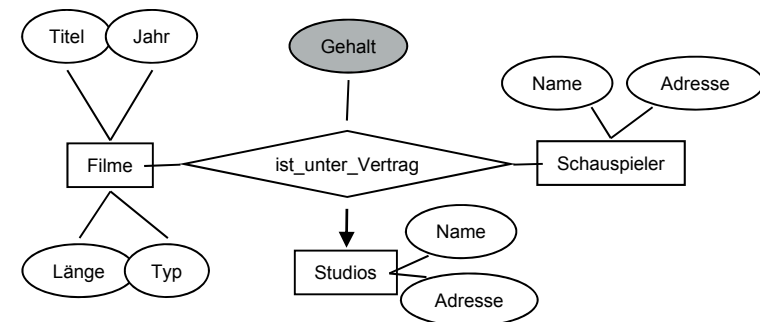
- Eine Spezies ist definiert durch den Namen der Gattung und des Spezies.
  - Gattung: homo
  - Spezies: homo sapiens

# Beispiel

52



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

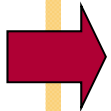


# Schlüssel schwacher Entitytypen

53

- 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.

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



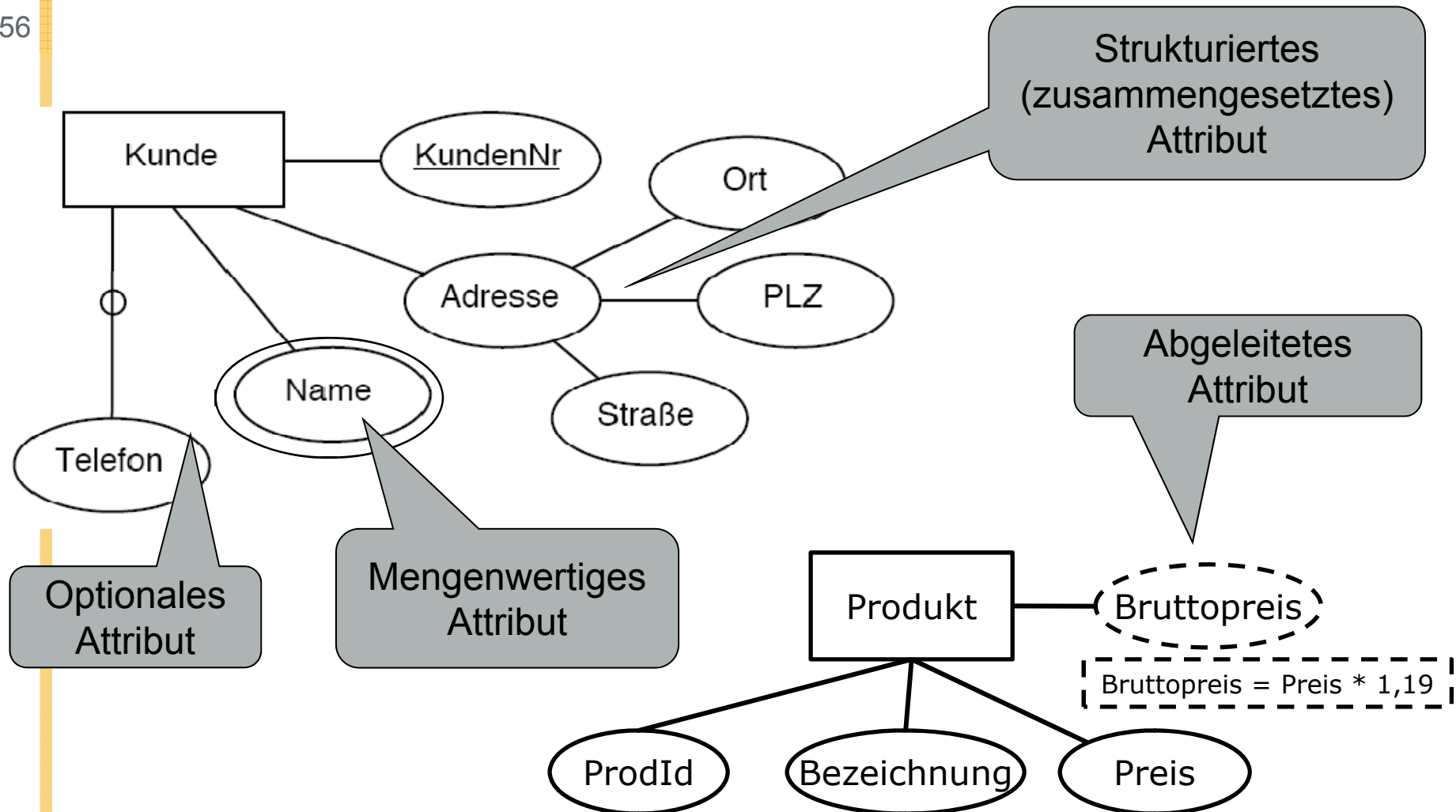
## Weitere Attribute-Arten

55

- **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.

# Weitere Attribut-Arten

56





# Spezialisierung und Generalisierung

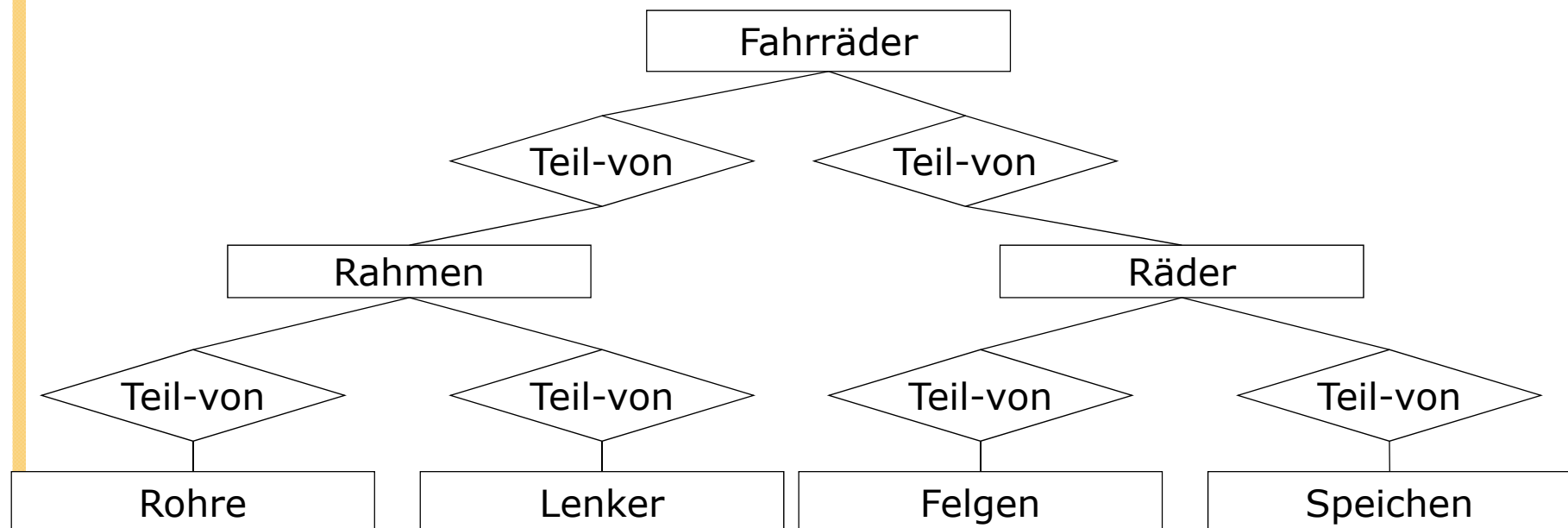
57

- Spezialisierung
  - entspricht IST-Beziehung
  - Drachen sind Spezialisierung von Produkt
- Generalisierung
  - Entities in einen allgemeineren Kontext
  - Drachen oder Windspiel als Produkt
- Partitionierung
  - mehrere disjunkte Entity-Typen
  - Spezialfall der Spezialisierung
  - Partitionierung von Produkten in Zubehör und Drachen

# Aggregation

58

- Generalisierung (IST): Gleichartige Entitytypen (*is-a*)
- Aggregation: Unterschiedliche Entitytypen
  - „Teil-von“ (*part-of*)
  - Entity aus einzelnen Instanzen anderer Entity-Typen zusammengesetzt.



## (Weitere Erweiterungen)

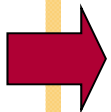
60

- 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

# Überblick

61

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



# Grundprinzipien

62

- Treue zur Anwendung
- Vermeidung von Redundanz
- Einfachheit
- Sparsamer Einsatz von Relationships
- Sparsamer Einsatz von Attributen
- Sparsamer Einsatz von schwachen Entitytypen

# Anwendungstreue

63

- Entitytypen und Attribute sollten Realität widerspiegeln.
  - Filme haben keine Zylinderkopfanzahl
- Relationshiptypen 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

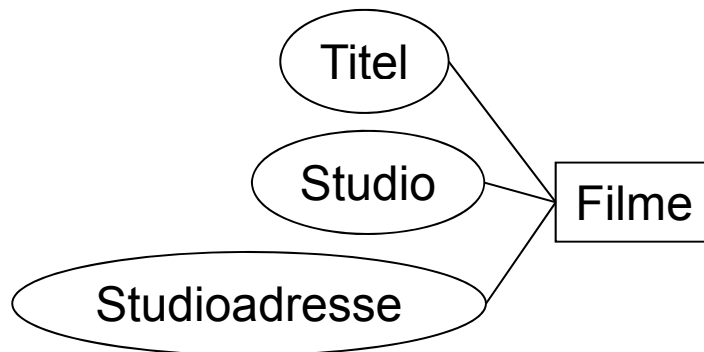
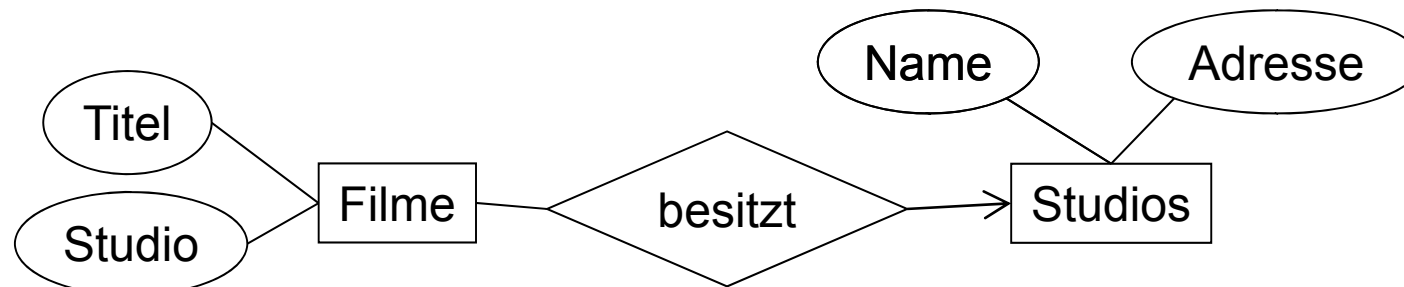
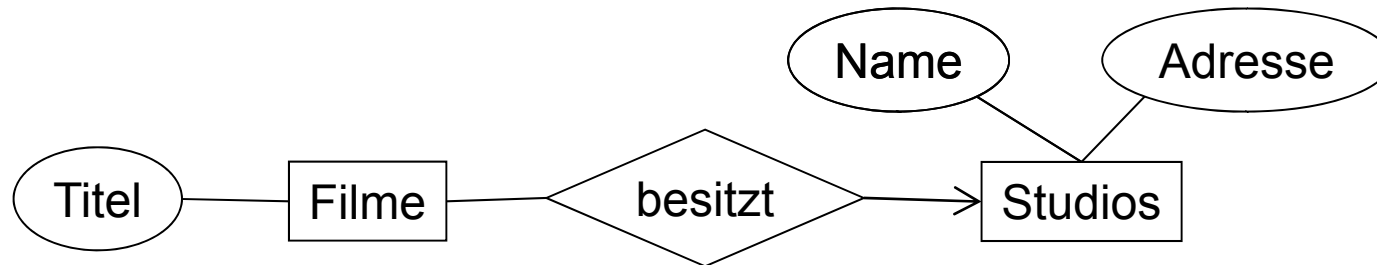
# Redundanz

64

- 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.

# Redundanz

65



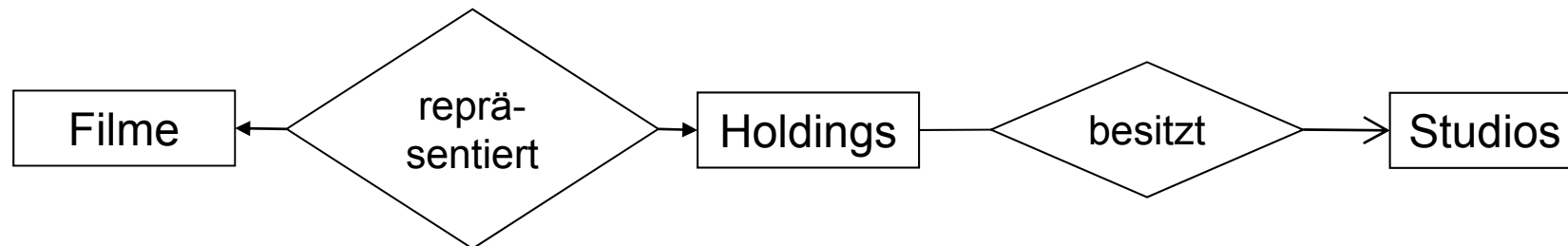
- Studioadresse wird für jeden Film wiederholt
- Studioadresse geht verloren wenn zwischenzeitlich kein Film existiert



# Einfachheit

66

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



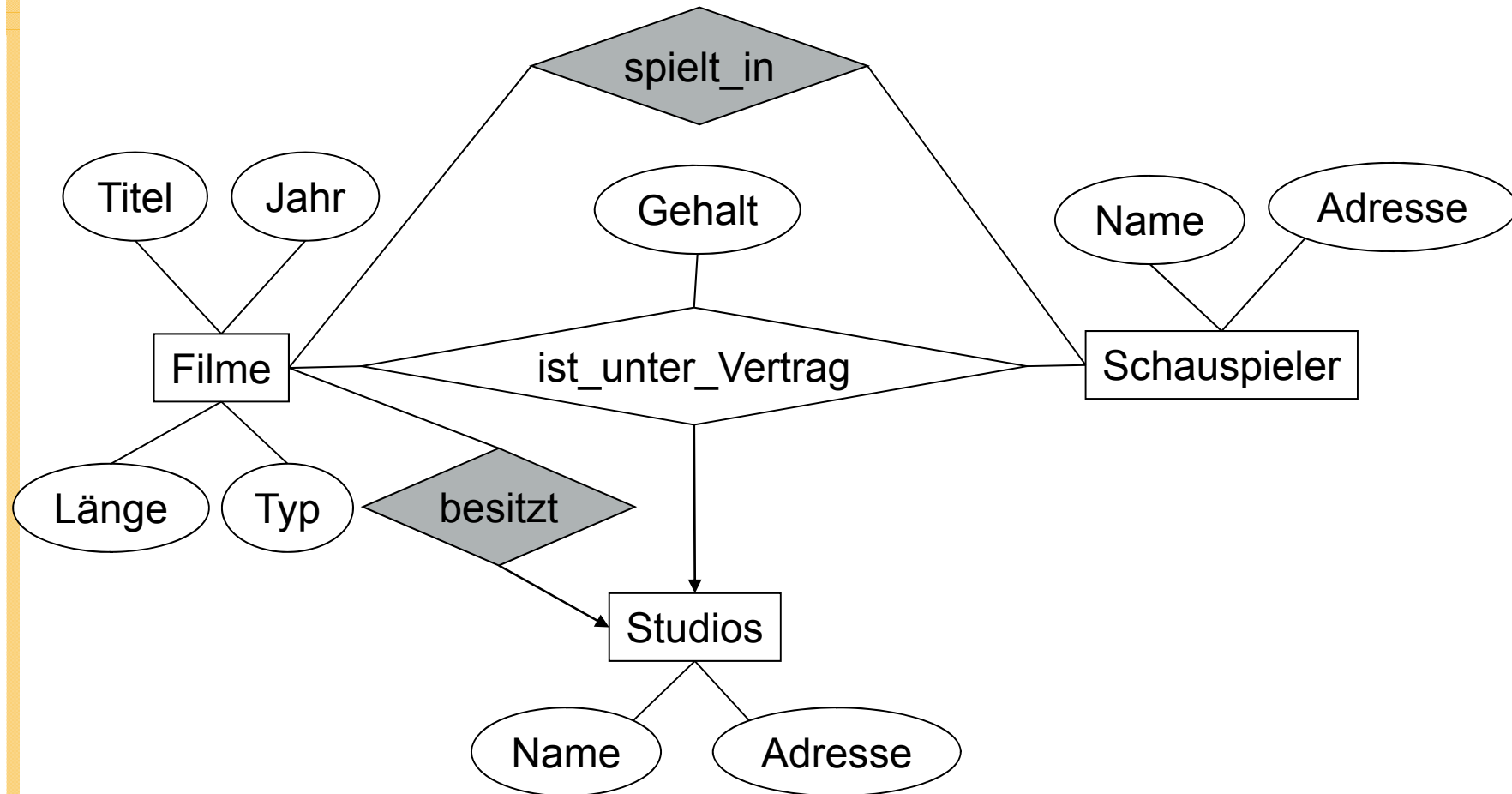
# Relationships

67

- 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

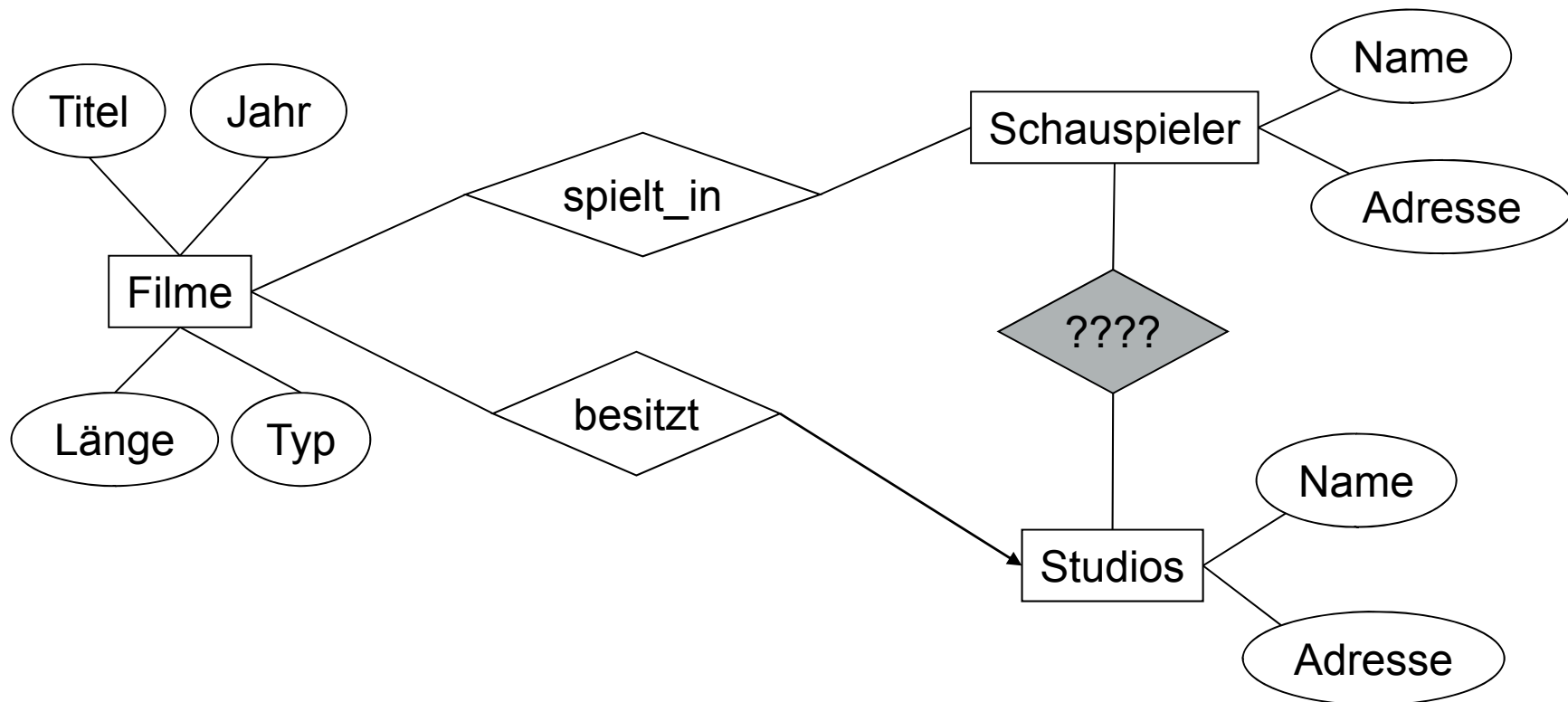
# Relationships

68



# Relationships

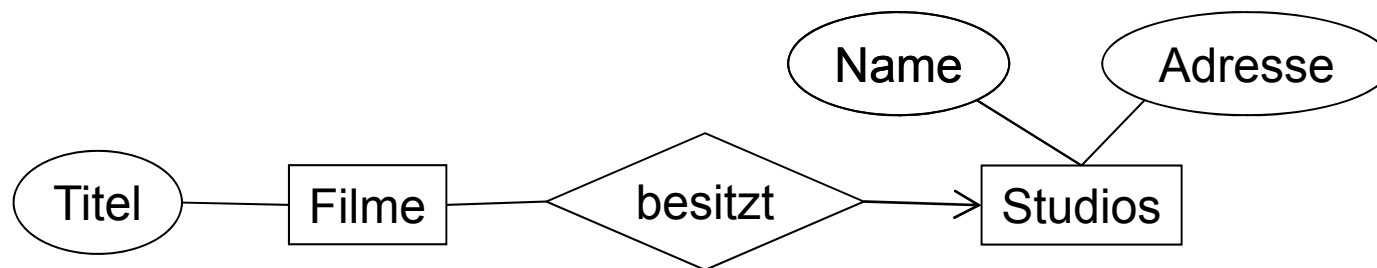
69



# Attribut vs. Element

70

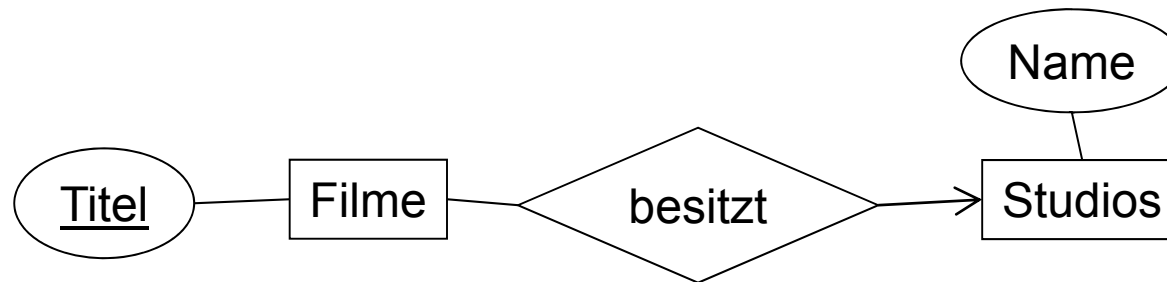
- Attribute sind einfacher zu implementieren als Entities und Relationships.
- Ein Entitytyp ist gerechtfertigt falls...
  - ... er mehr als nur den Namen eines Objekts darstellt,
  - ... oder er der n-Teil einer 1:n Relationship ist.



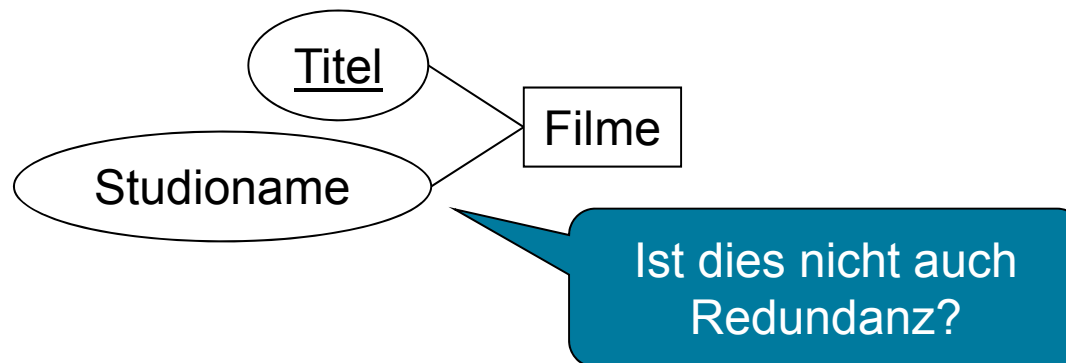
# Attribut vs. Element

71

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



- Besser

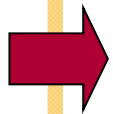


# Schwache Entitytypen

72

- 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.

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



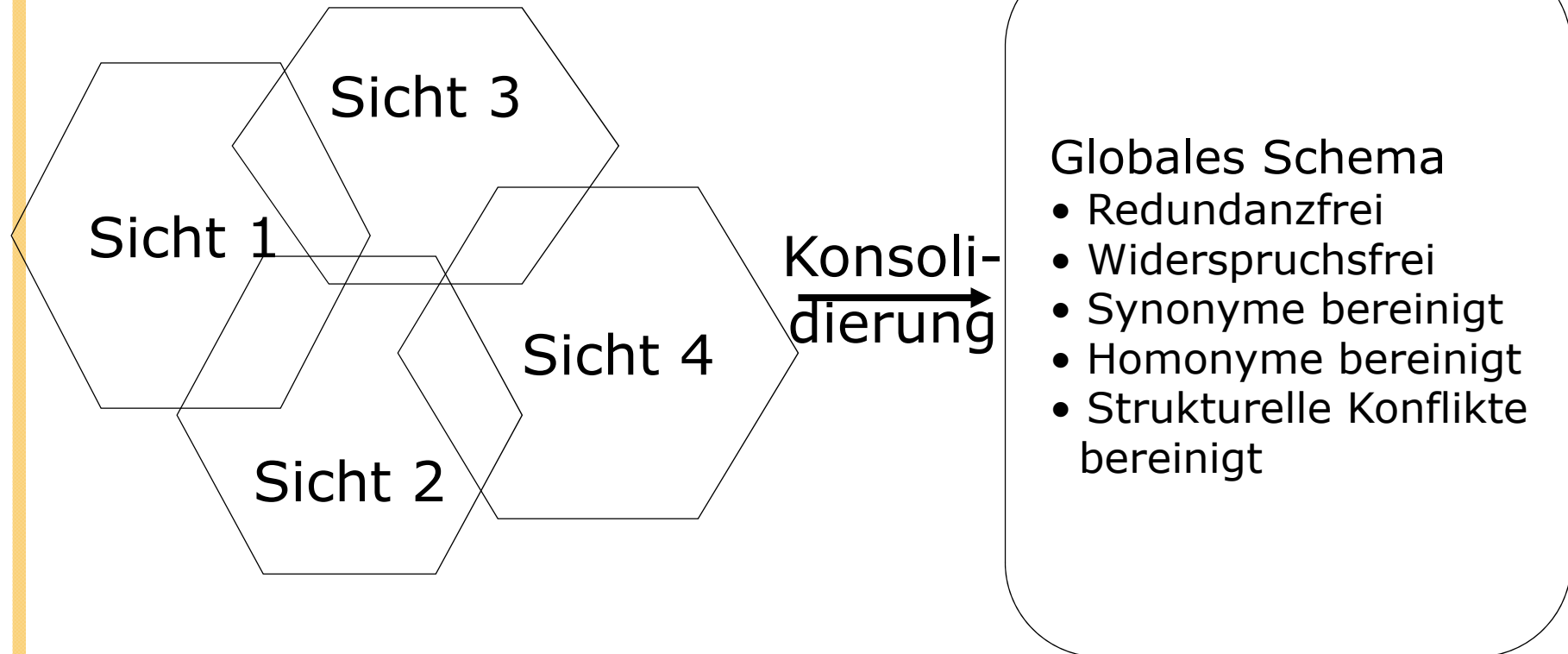
Folien nach Alfons  
Kemper (TU München)





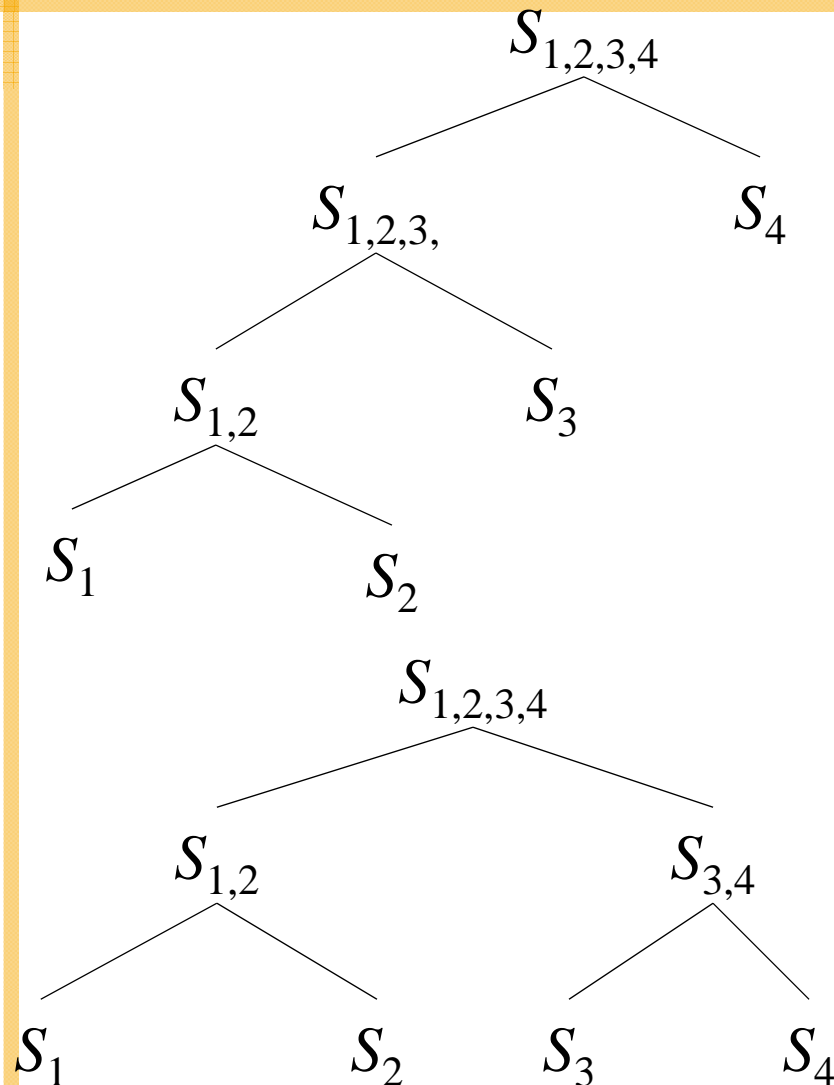
# Konsolidierung von Teilschemata oder Sichtenintegration

74



# Möglicher Konsolidierungsbaum

75

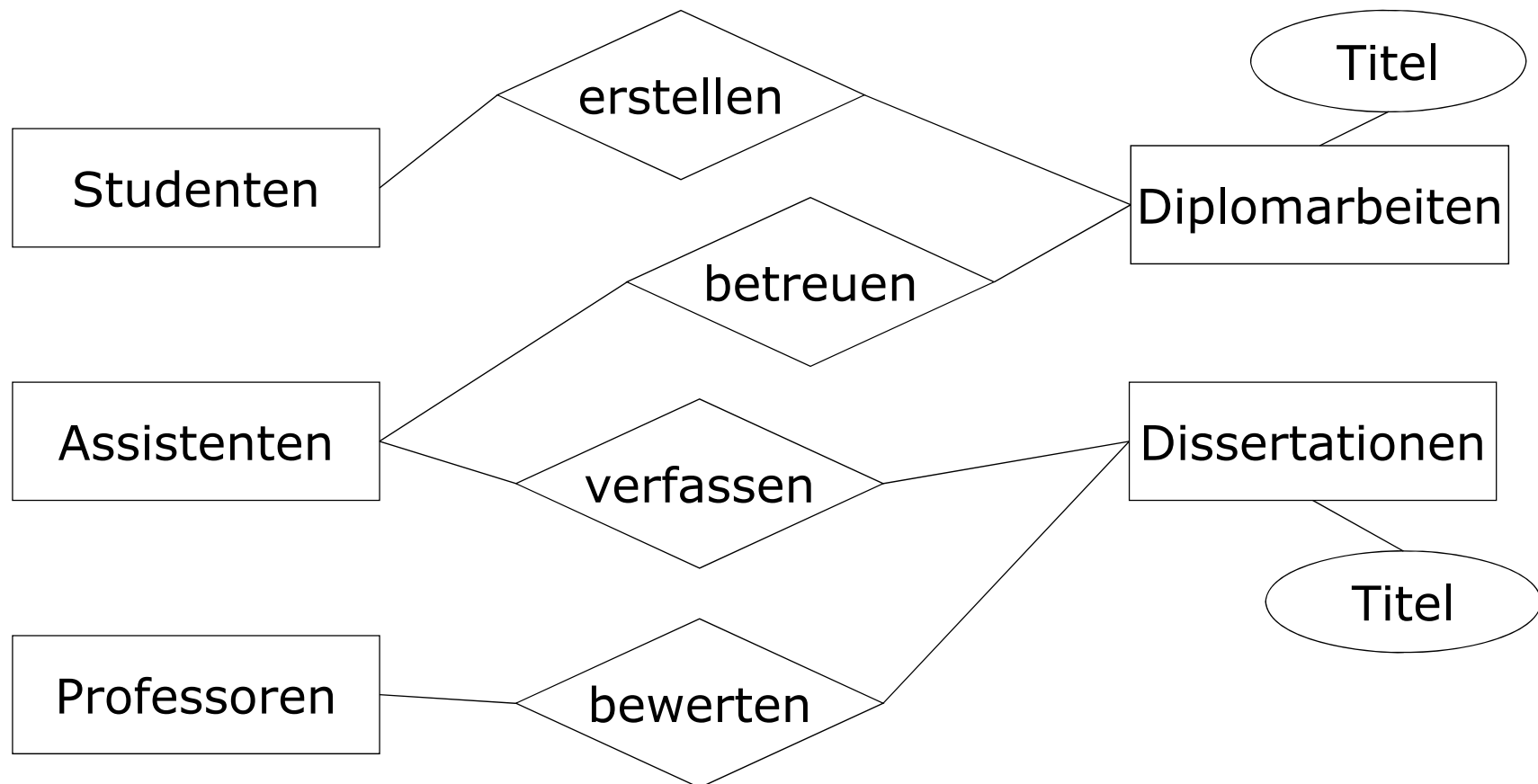


- Mögliche Konsolidierungsbäume zur Herleitung des globalen Schemas  $S_{1,2,3,4}$  aus 4 Teilschemata  $S_1$ ,  $S_2$ ,  $S_3$ , und  $S_4$ 
  - Maximal hoher Konsolidierungsbaum
    - ◇ „links-tief“ (left-deep)
  - Minimal hoher Konsolidierungsbaum
    - ◇ Balanciert
  - Hybrides Vorgehen
- Beide Vorgehensweisen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

# Drei Sichten einer Universitäts-Datenbank

76

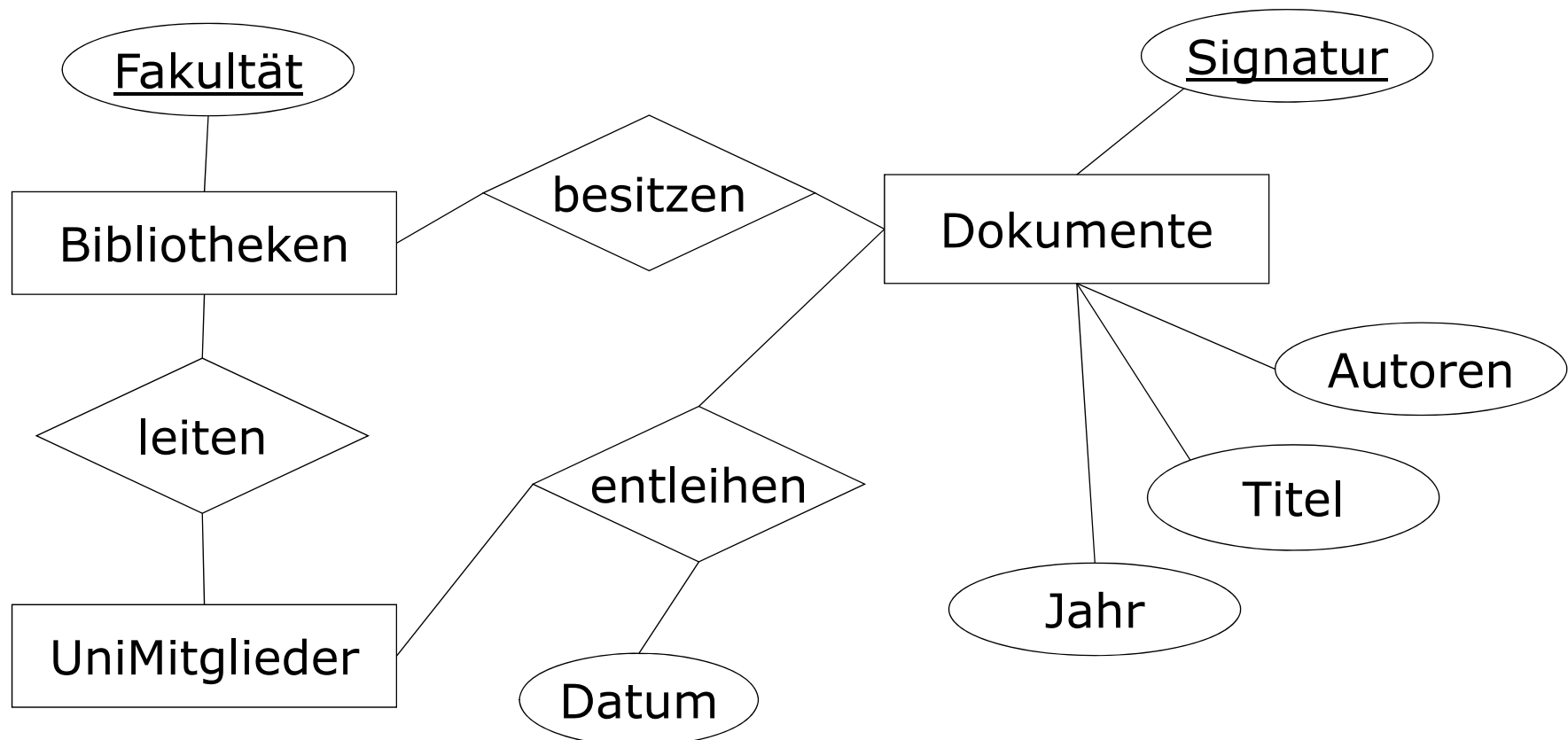
## Sicht 1: Erstellung von Dokumenten als Prüfungsleistung



# Drei Sichten einer Universitäts-Datenbank

77

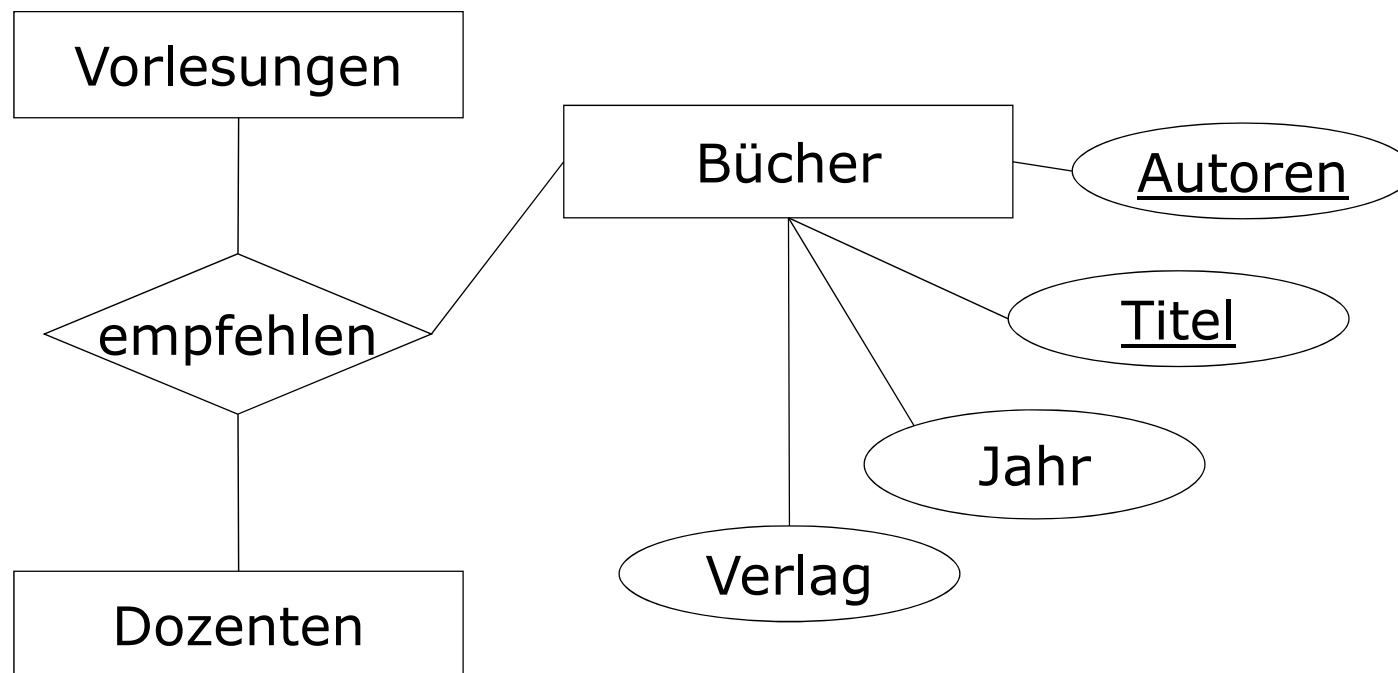
## Sicht 2: Bibliotheksverwaltung



# Drei Sichten einer Universitäts-Datenbank

78

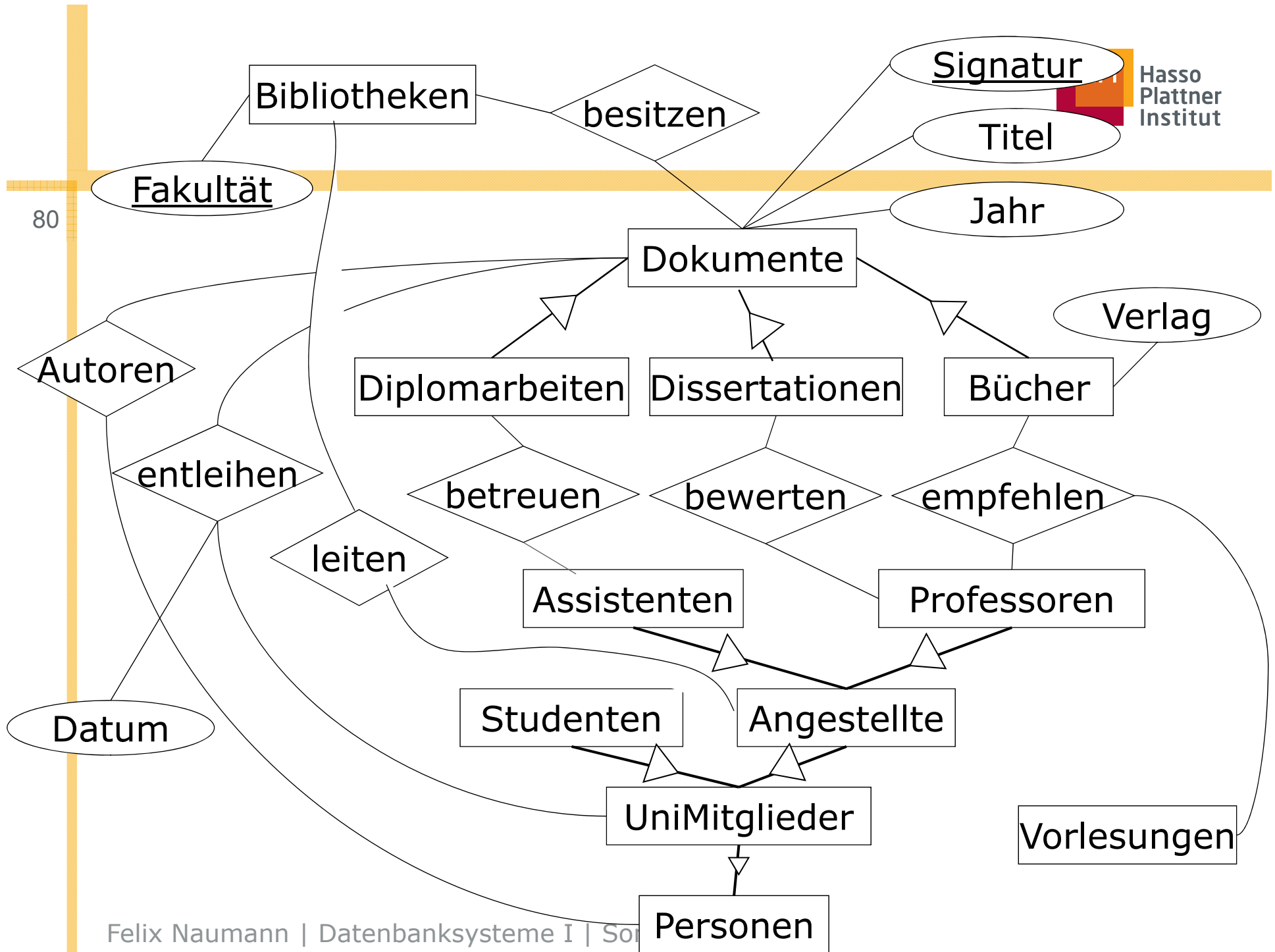
## Sicht 3: Buchempfehlungen für Vorlesungen



# Beobachtungen

79

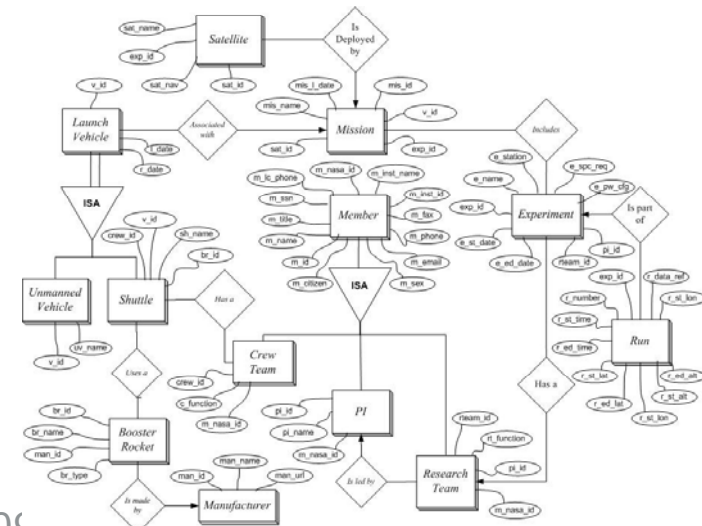
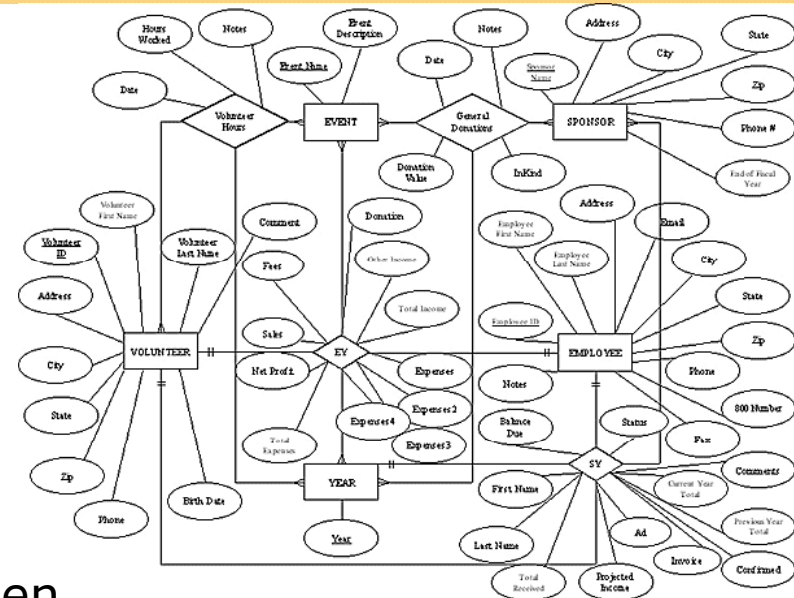
- Die Begriffe Dozenten und Professoren sind synonym.
- Der Entitytyp UniMitglieder ist eine Generalisierung von Studenten, Professoren und Assistenten.
- Fakultätsbibliotheken werden sicherlich von Angestellten (und nicht von Studenten) geleitet. Insofern ist die in Sicht 2 festgelegte Relationship „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.



# Zusammenfassung

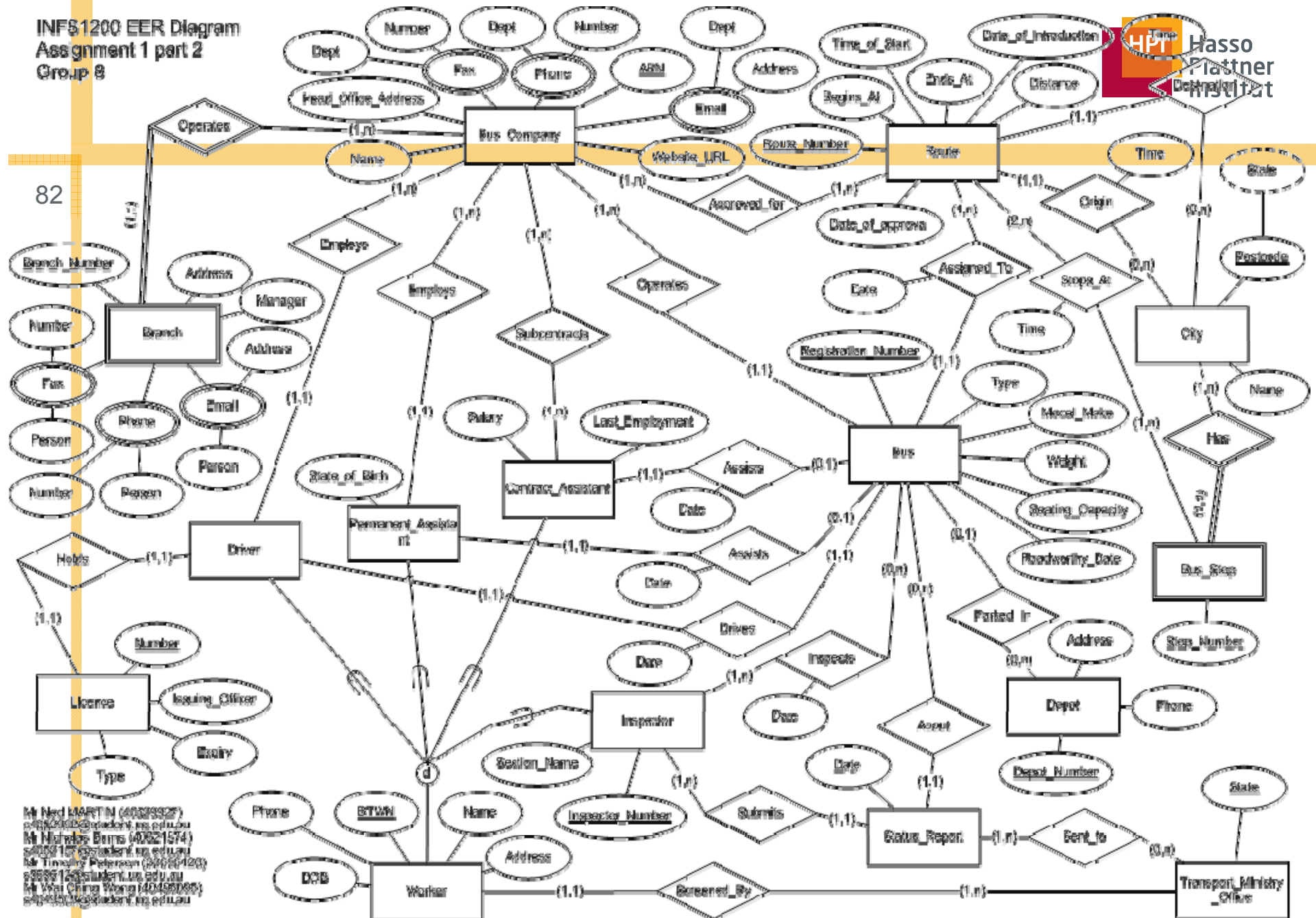
81

- Motivation und Einbettung
  - Entwurfsprozess
- Begriffe und Definitionen
  - Entities und Relationships
- ER-Diagramme
  - Kästen, Rauten und Ovale
- Modellierung von Nebenbedingungen
  - Pfeile und Striche
- Schwache Entitytypen
- Erweitertes ER-Modell
- Designprinzipien
- Sichtintegration



Tipp: <http://mediatedcultures.net/ksudigg/?p=120>





Mr Ned MARTIN (40379927)  
e40379927@student.mq.edu.au  
Mr Nicholas BIRMS (40321574)  
e40321574@student.mq.edu.au  
Mr Timothy PETERSON (40366420)  
e40366420@student.mq.edu.au  
Mr Wei CHANG WONG (40456090)  
e40456090@student.mq.edu.au