



ER Modellierung

Medieninformatik Bachelor Modul 9: Datenbanksysteme



Aufgabe 1 Anfragen & Modellierung“

Denken Sie mal darüber nach, welche Anfragen Sie an die AOL Daten stellen möchten. Bitte Sie bitte ein logisches und physisches Schema zur Beantwortung dieser Anfragen.

Aufgabe 2 „SQL und Abfrageausführung“

Bitte formulieren Sie für Ihre Analyseideen aus 1.) die SQL Anfragen. Sie verstehen auch Möglichkeiten der Abfrageausführung bzw. Optimierung.

Aufgabe 3 „Datenintegration“

Zur Ausführung der Ausführung fehlen Ihnen noch externe Daten, z.B. aus dem Internet Archive, DMOZ oder Freebase.org. Bitte ergänzen Sie Ihr Schema und die Datenbasis.

Aufgabe 4 „Analyse, Erkenntnisgewinn und Wert“

Stellen Sie in 5 Minuten die wichtigsten Erkenntnisse aus den Daten vor. Bewerten Sie den Erkenntnisgewinn, z.B. gegenüber Ihren Kommilitonen oder der Literatur! Welche Erkenntnisse hätten einen kommerziellen Wert?

Was sind Datenbanken?

- Motivation, Historie, Datenunabhängigkeit, Einsatzgebiete

Datenbankentwurf im ER-Modell & Relationaler Datenbankentwurf

- Entities, Relationships, Kardinalitäten, Diagramme

- Relationales Modell, ER -> Relational, Normalformen, Transformationseigenschaften

Relationale Algebra & SQL

- Kriterien für Anfragesprachen, Operatoren, Transformationen

- SQL DDL, SQL DML, SELECT ... FROM ... WHERE ...

Datenintegration & Transaktionsverwaltung

- JDBC, Cursor, ETL

- Mehrbenutzerbetrieb, Serialisierbarkeit, Sperrprotokolle, Fehlerbehandlung, Isolationsebenen in SQL

Ausblick

- Map/Reduce, HDFS, Hive ...

- Wert von Daten

- Datenbankdesign als „Skizze“ (korrekt: Informationsmodell) mittels ER-Modellierung
 - Welche Daten? Welche Beziehungen?
 - Nicht wie Daten erzeugt oder verändert werden.
- Informationsmodelle für relationale Datenbanken meist als „Entity-Relationship-Diagramme“
 - ER-Diagramme
- Nächste Woche: Überführung von ER-Diagrammen in das relationale Modell
 - Relationen, Attribute, Integritätsbedingungen
- In 2 Wochen: Überführung von Relationenschemata in SQL Ausdrücke
 - **CREATE TABLE ...**

■ Grundlagen der abstrakten Datenmodellierung

- **Einleitung: Was sind Modelle und wozu braucht man sie?**

- Grundlegende Elemente des ERM

- Modellierung von Nebenbedingungen

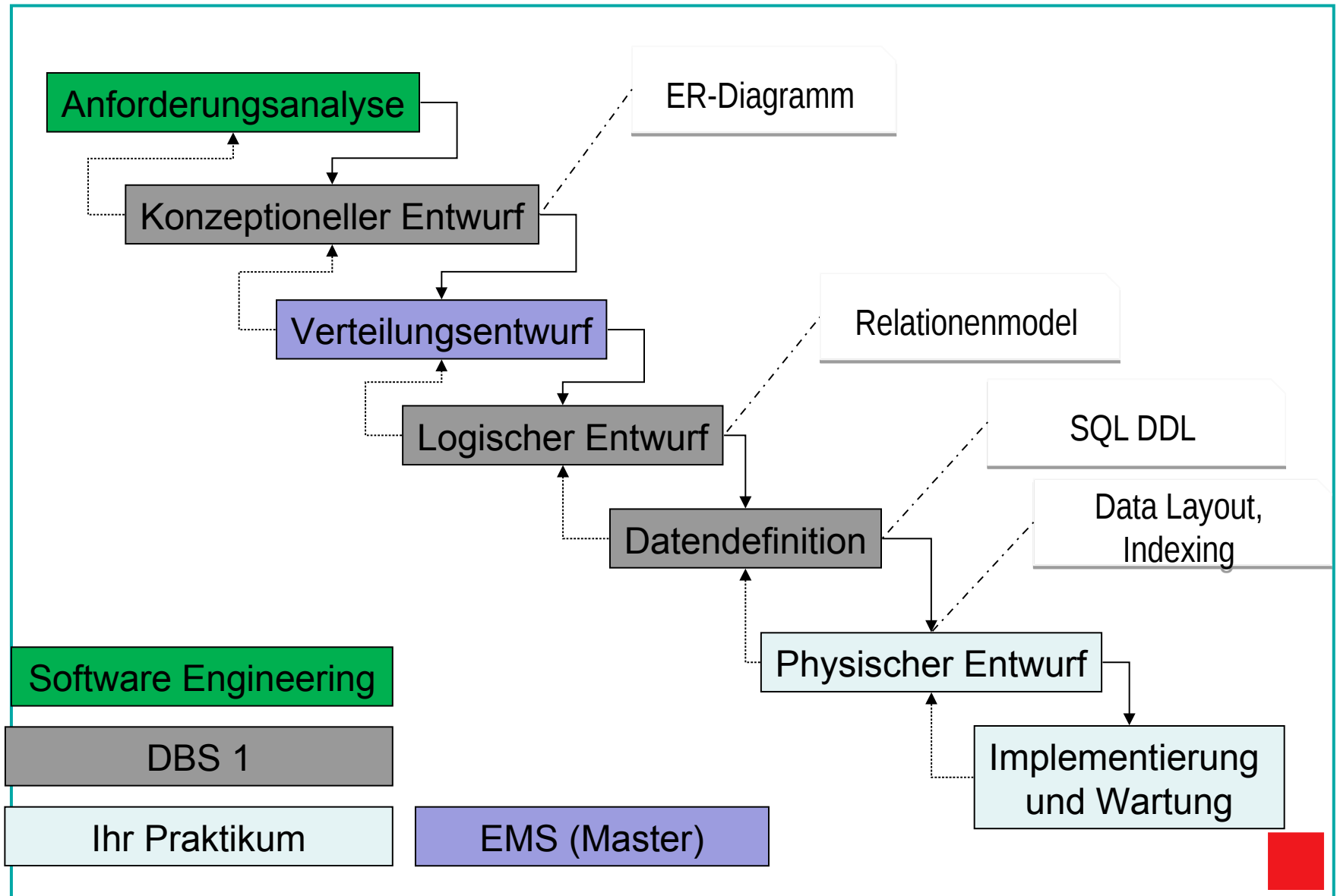
- Exkurs: Erweiterungen des ERM

- In der Übung: Vorgehen und Beispiel



Was ist ein Modell?

- Ein Modell ist eine **abstrakte Darstellung der Wirklichkeit**
- Ein Modell entspricht damit nie exakt der Realität
- Modellieren bedeutet immer **Hervorheben** und **Weglassen**
 - Hervorheben des Wesentlichen
 - Weglassen unwichtiger Details
- Was aber ist wesentlich, was unwichtig?
Keine allgemeine Beantwortung möglich, sondern abhängig davon,
 - was für Ziele mit dem Modell verfolgt werden und
 - wer die Leser des Modells sind (Wissensstand, Sichtweise, Versiertheit, Entscheidungsbefugnis)
- Modelle werden erstellt, um existierende oder zukünftige IT- und/oder Geschäftssysteme (z.B. Prozesse, Organisationen, Strukturen) besser verstehen zu können



- 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



- Erste formale Beschreibung des Fachproblems
 - UoD: Universe of Discourse (Diskursbereich)
- 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





- 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)



- Sprachmittel: Datenmodell des ausgewählten „Realisierungs“-DBMS
 - z.B. DB2, Informix => relationales Modell
 - z.B. 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: Relationaler Entwurf



- 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 ...



- 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: $10.240/10 = 1.024 \text{ sec.} = 17 \text{ min.}$

Wird nicht in der VL behandelt



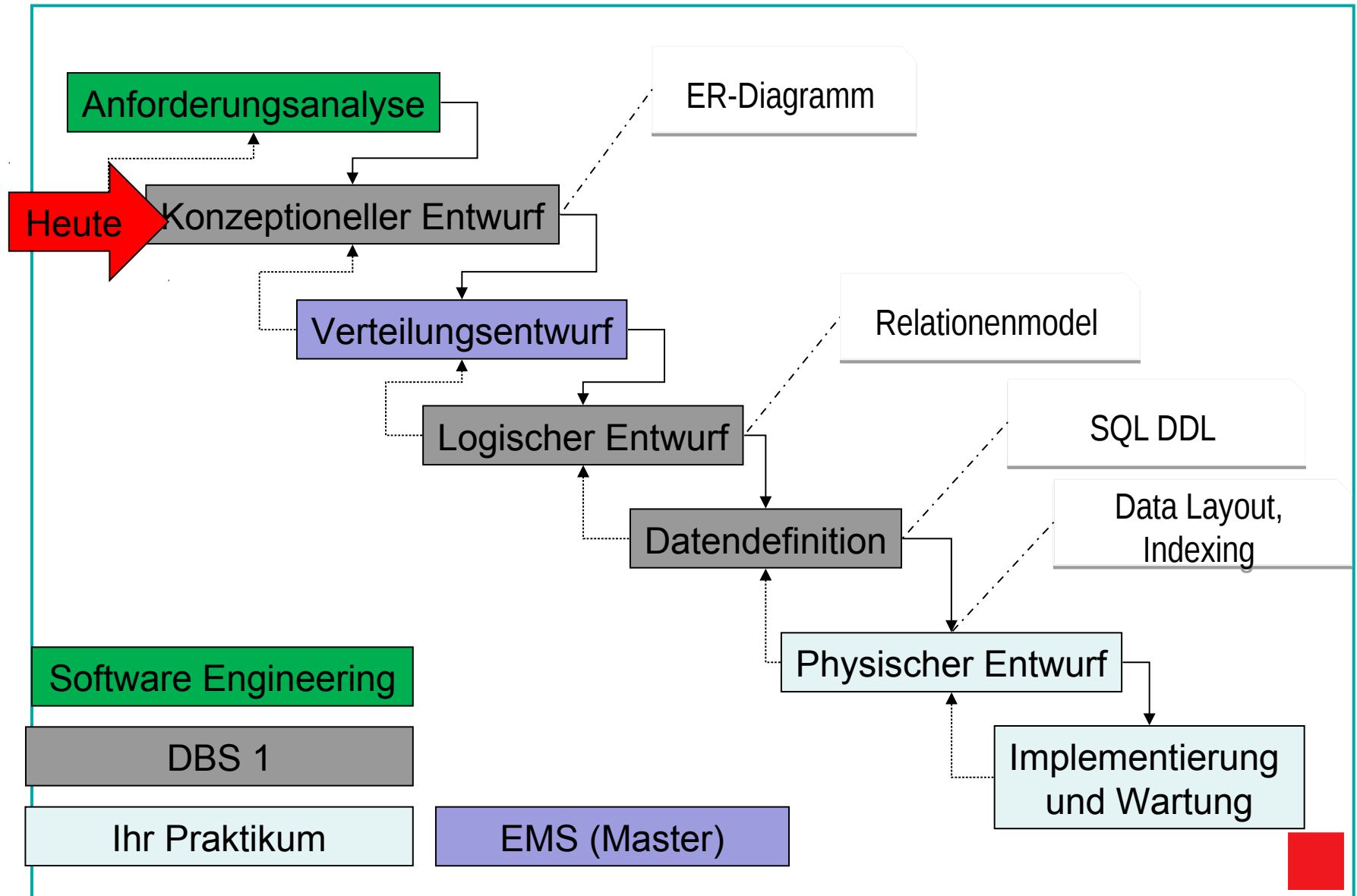


- Wartung des DBMS
 - Parameter, Festplatten, etc.
- Weitere Optimierung der physischen Ebene
- Anpassung an neue Anforderungen
- Anpassung an neue Systemplattformen
- Portierung auf neue Datenbankmanagementsysteme

- Kostenaufwändigste Phase
- Software Engineering

Wird nicht in der Vorlesung behandelt





■ Grundlagen der abstrakten Datenmodellierung

■ Einleitung: Was sind Modelle und wozu braucht man sie?

■ **Grundlegende Elemente des ERM**

■ Modellierung von Nebenbedingungen

■ Exkurs: Erweiterungen des ERM

■ In der Übung: Vorgehen und Beispiel



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

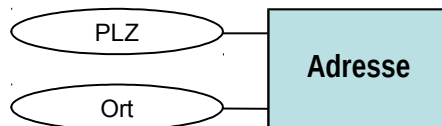
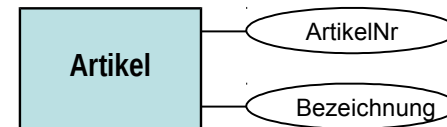
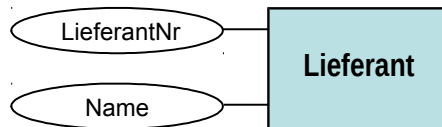


Modellelemente

ERM	Erläuterung
Entität (Entity)	Abgrenzbarer Gegenstand / Begriff / „Ding“ / Person / Ereignis in der Realsphäre.
Entitätentyp (Entity Type)	Gleichartige Entitäten werden klassifiziert. Sie werden zu einem Entitätentyp bzw. einer Klasse zusammengefasst
Beziehung (Relationship)	Eine Beziehung stellt die logische Verknüpfung von zwei oder mehr Entitäten/Objekten.
Beziehungstyp (Relationship Type)	Gleichartige Beziehungen werden klassifiziert. Sie werden zu einem Beziehungstypen zusammengefasst Anmerkung: In der UML spricht man ebenfalls von einer (Klassen-)Assoziation
Attribut (Attribute)	Attribute werden Objekten und Beziehungen als deren Eigenschaften zugeordnet und erlauben deren Charakterisierung und Identifizierung. Die Ausprägungen der Attribute sind Werte. Alle (gültigen) Werte ergeben zusammen einen Wertebereich.
Kardinalität (Cardinality)	Die Kardinalität (auch Multiplizität) beschreibt die mögliche Anzahl (Min/Max) der an einer Beziehung beteiligten Entitäten

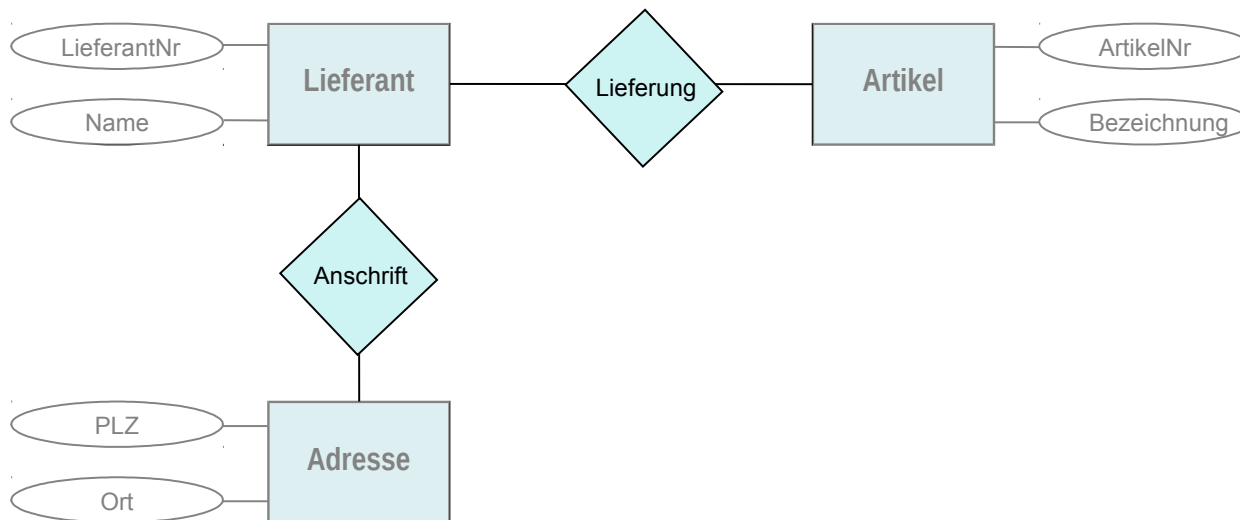
ERM: Darstellung von Entitätentypen und Attributen

- Entitäten werden als Rechtecke dargestellt, die in ihrer Mitte die Bezeichnung tragen
- Attribute werden als Ovale mit einer Linie mit dem entsprechenden Rechteck verbunden



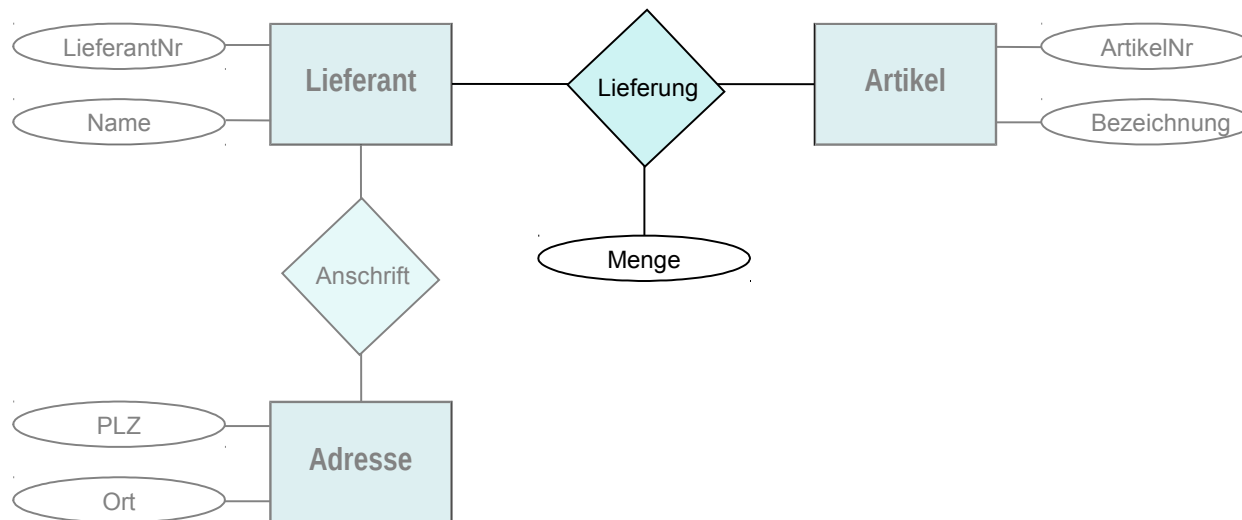
ERM: Darstellung von Beziehungstypen

- Beziehungen zwischen zwei Entitätentypen werden über eine Linie zwischen diesen Entitäten symbolisiert
- Die Linie wird von einer Raute dekoriert in die optional eine klassifizierende Bezeichnung für den Beziehungstyp eingetragen werden kann

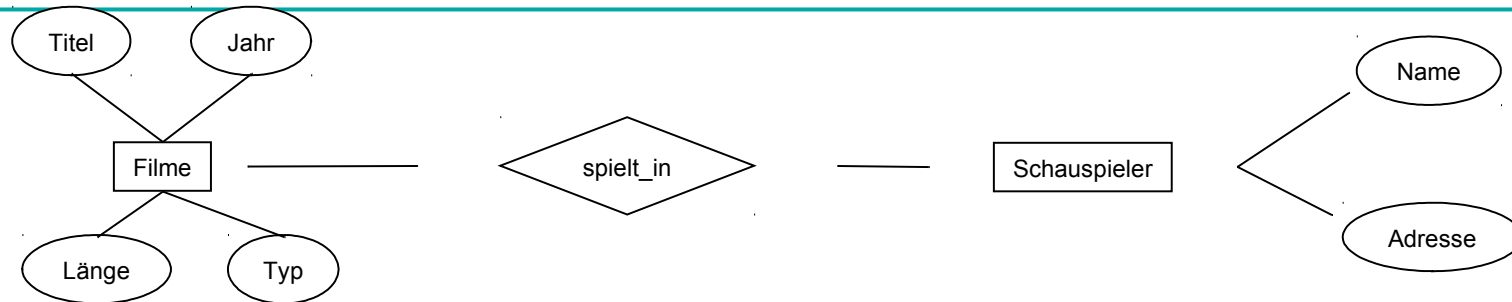


ERM: Darstellung von attribuierten Beziehungstypen

- Weist ein Beziehungstyp zusätzliche Eigenschaften (als nur das Herstellen der Beziehung) auf, so können diese durch Attribute notiert werden
- Attribute werden (auch hier) als Ovale mit einer Linie mit der entsprechenden Raute verbunden



Beispiel: ER-Diagramm und Tabellen



Filme

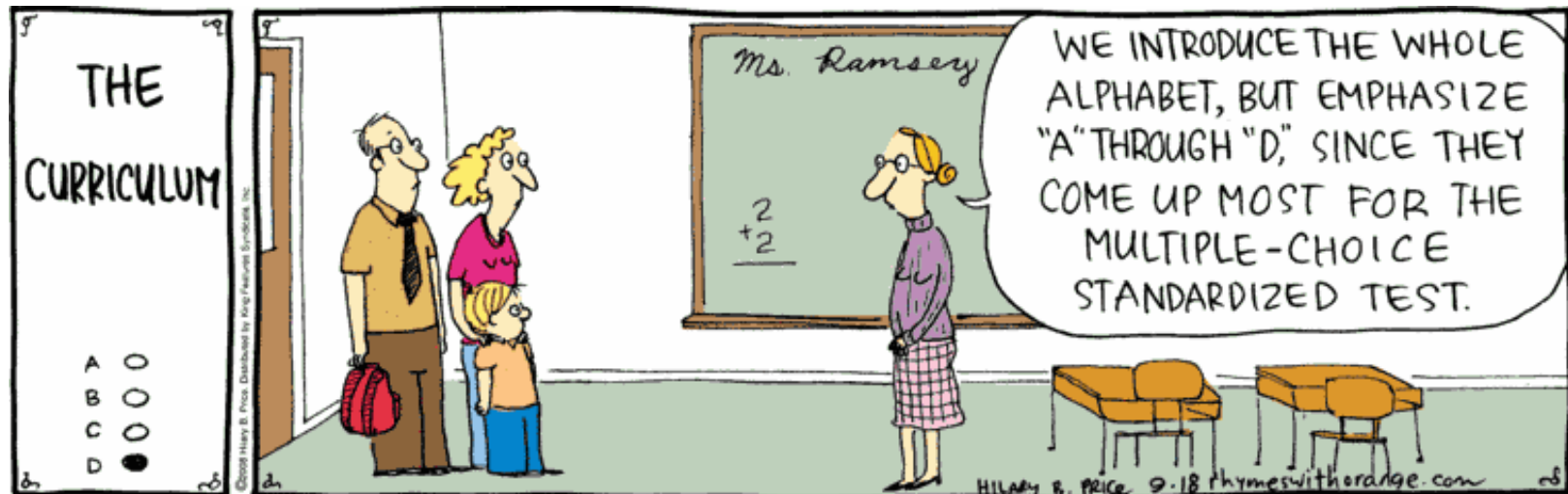
Titel	Jahr	Länge	Typ
Basic Instinct	1992	127	Farbe
Total Recall	1990	113	Farbe
Dead Man	1995	112	s/w

Schauspieler

Name	Adresse
Sharon Stone	Hollywood
Arnold Schwarzenegger	Sacramento
Johnny Depp	Paris

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

- Bitte erstellen Sie eine Multiple Choice Aufgabe zum Thema E/R Modellierung
 - 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 Aufgaben an Ihren rechten Nachbarn. Diskutieren Sie gemeinsam und markieren Sie die richtigen Lösungen
- Geben Sie am Ende der Vorlesung Ihre Aufgabe bei mir ab



■ Grundlagen der abstrakten Datenmodellierung

- Einleitung: Was sind Modelle und wozu braucht man sie?
- Grundlegende Elemente des ERM
- **Modellierung von Nebenbedingungen**
- Exkurs: Erweiterungen des ERM
- In der Übung: Vorgehen und Beispiel



Schlüssel

- Ein oder mehrere Attribute

- Werte identifizieren eindeutig ein Entity.

Kardinalitäten und Totalitäten: Referentielle Integrität

- Existenz des referenzierten Entities

- Entspricht „dangling pointer“

Domänen

- Einschränkung des Wertebereichs

Allgemeine Nebenbedingungen (Assertions)

- Z.B. nicht mehr als 10 Artikel pro Lieferung

Nebenbedingungen sind Teil des Schemas. Sie leiten sich nicht aus den Daten ab!

- 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
- Darstellung durch Unterstreichen der Attributnamen



Schlüssel

Artikel / Produkt

„Produktname“ als Schlüssel?

Den Film „King Kong“ gibt es mehrfach

Titel und Jahr als Schlüssel?

Eventuell kann ein Film dann nicht gespeichert werden

In IMDB: 275 Doppelte

Lieferant

„Name“ oder „Name“ und „Adresse“

Projekt

Name

In der Praxis: Modellierung eines speziellen, numerischen

ISBN, SSN, Imma-Nummer,...URL + DATUM, ...

Steuerzentrale - DBZCOP 1.1

Steuerzentrale Ausgewählt Editieren Sicht Tools

Objektsicht Befehlseditor 1 X

Befehle Abfrageergebnisse Zugriffsplan

Diese Ergebnisse werden mit Hilfe von Aktualisierungen u...

TITLE	DATE	3
Shadow Chasers	1988	2
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
Tæskeholdet	1997	2
Temptation	1968	2
Tenshi no tama...	1985	2
Tenspeed and ...	1980	2
Test, The	2001	2
Top of the Hill	1989	2

Festschreiben Rückgängig machen

Wo ist der Schlüssel?



Star Wars (1977) - Mozilla Firefox

http://www.imdb.com/title/tt0076759/

Star Wars (1977)

IMDb

Earth's Biggest Movie Database™

Home | Top Movies | Photos | Independent Film | Browse | Help

Search the IMDb

All

go

Result: 1 of 108

[< x > ?]

[More searches](#) | [Tips](#)

[IMDbPro.com free trial](#)

Showing page 1 of 39

Overview

- main details
- combined details
- full cast and crew
- company credits

Awards & Reviews

- user comments
- external reviews
- newsgroup reviews
- awards & nominations
- user ratings
- recommendations
- message board

Plot & Quotes

- plot summary
- plot keywords
- Amazon.com summary
- memorable quotes

Star Wars (1977)

Directed by [George Lucas](#)

Writing credits [George Lucas](#) (written by)

[Add to MyMovies](#) [Photo Gallery](#) [IMDbPro Details](#)

Photo Gallery (35 photos)

[more](#)

Genre: [Action](#) / [Adventure](#) / [Fantasy](#) / [Sci-Fi \(more\)](#)

Tagline: A long time ago in a galaxy far, far away... [\(more\)](#)

SHOP STAR WARS

[DVD](#) [VHS](#) [CD](#)

[Memorabilia](#)

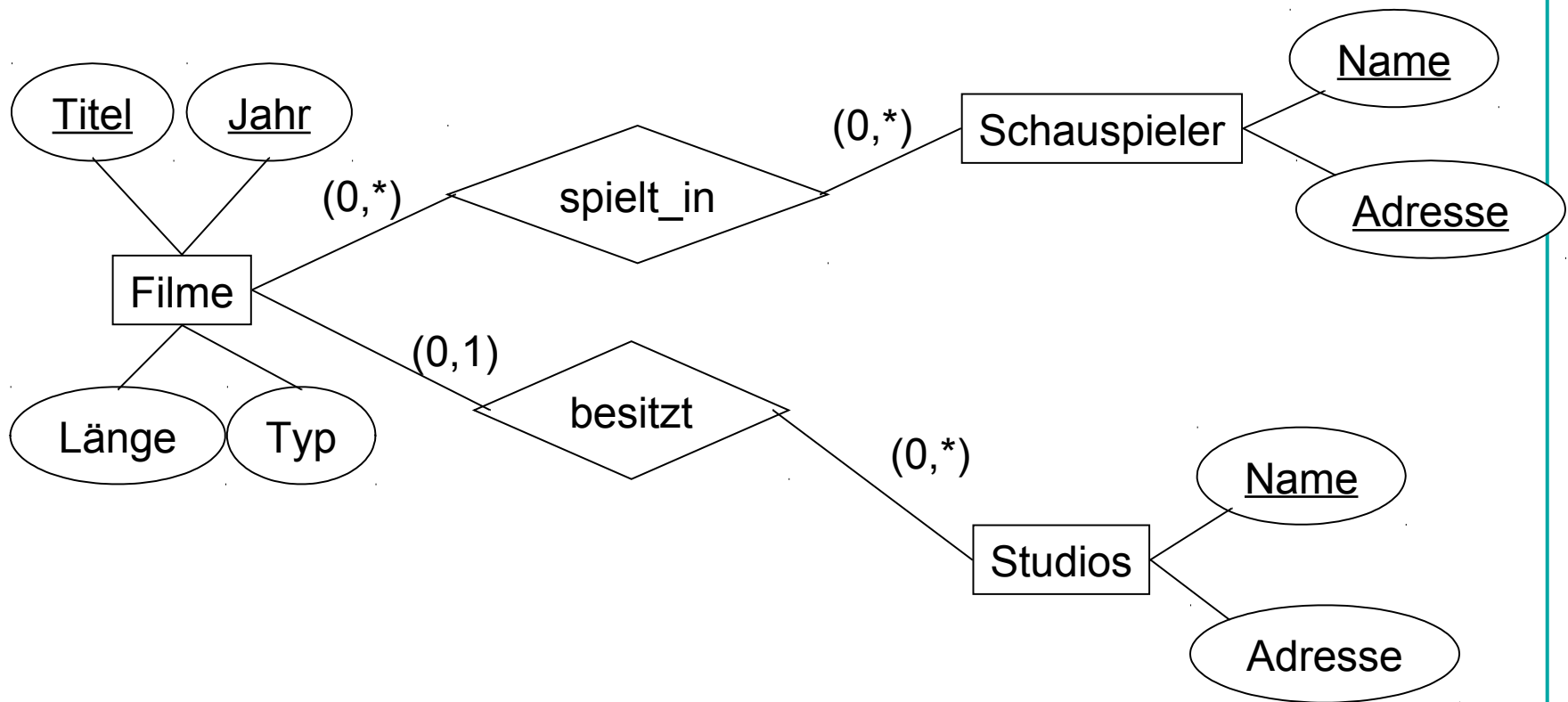
[Books](#) | [All Products](#)

[amazon.com](#)

Fertig

2.513s

Adblock



Allgemein: Ein binärer 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.

Nicht jeder Entity muss mit einem anderen verbunden sein.

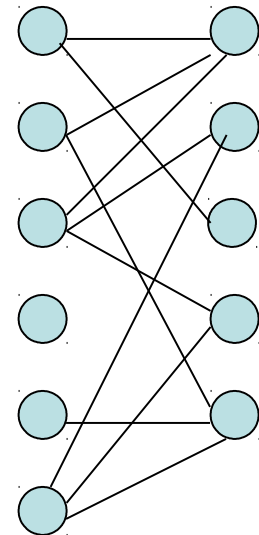
In einem Animationsfilm spielen keine Schauspieler.

m:n Beziehung

Einschränkungen („Spezialfälle“)

1:n Beziehung

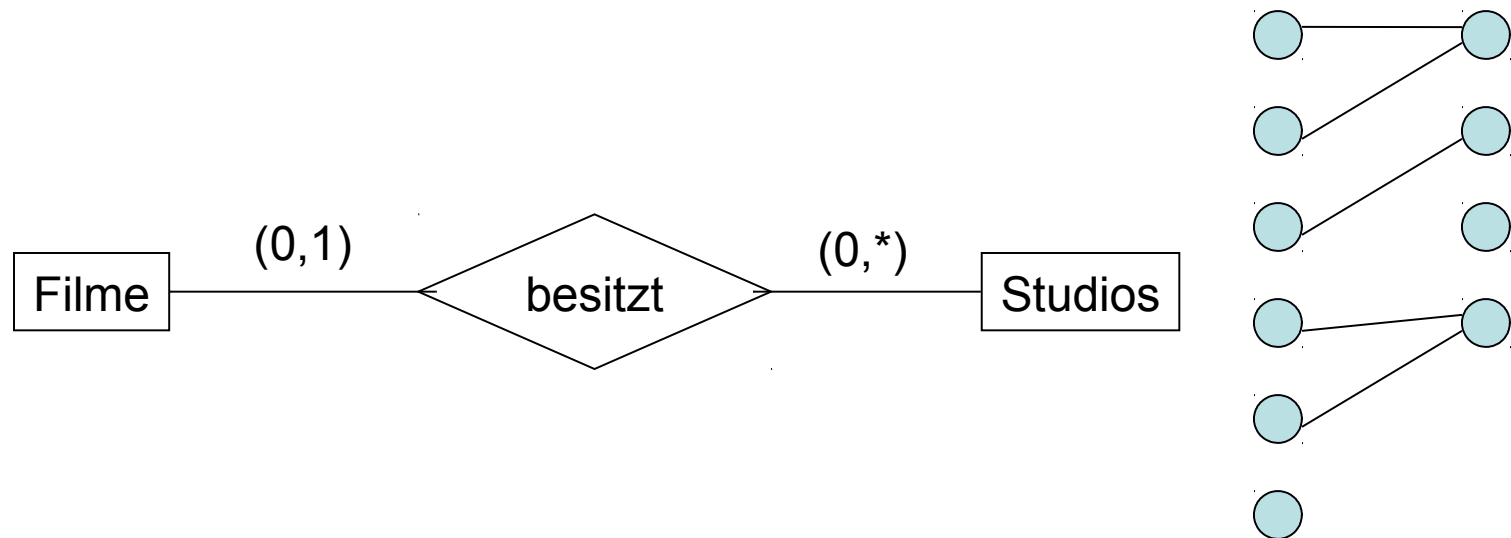
1:1 Beziehung



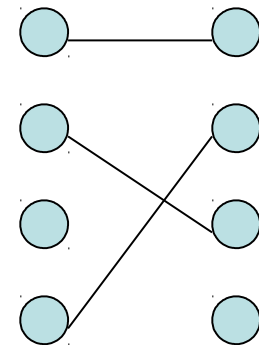
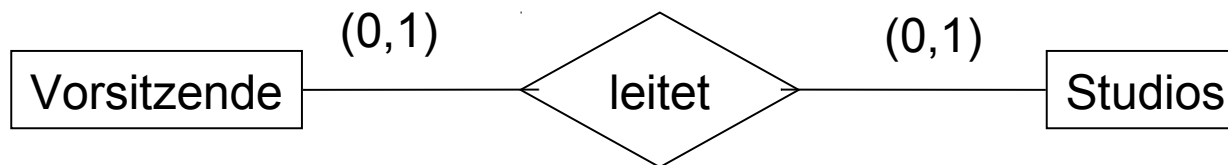
- Ein Entity vom Typ E kann mit beliebig vielen Entities des Typs F verbunden sein.
- Ein Entity vom Typ F kann mit höchstens einem Entity des Typs E verbunden sein.

Beispiel

- Ein Studio *kann* die Rechte an mehreren Filmen besitzen.
- Ein Film *kann* nur von einem Studio besessen werden.



- 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.



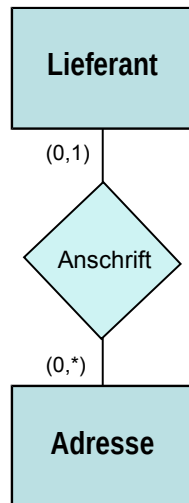
Existenzabhängigkeit (Totalität/Optionalität) von Beziehungstypen

■ Aussage zum Minimum kann sein:

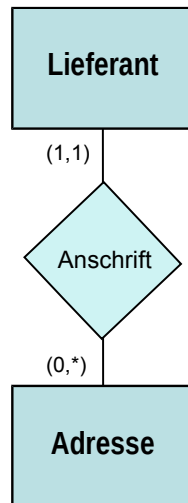
■ 0 – optionale Beziehung

■ 1 – existenzabhängige Beziehung

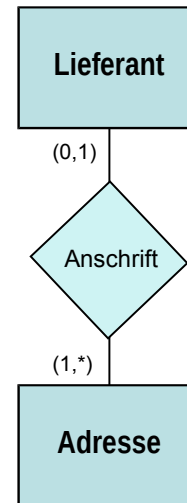
■ Existenzabhängigkeiten haben Auswirkungen auf die Einfüge-, Änderungs- und Löschoperationen Beispiel: Löschen von Lieferant / Adresse



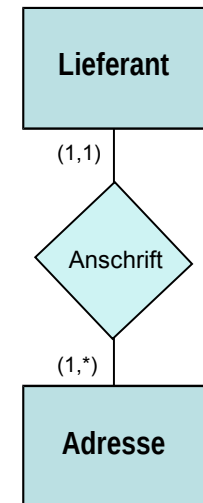
Lieferant und Adresse sind unabhängig



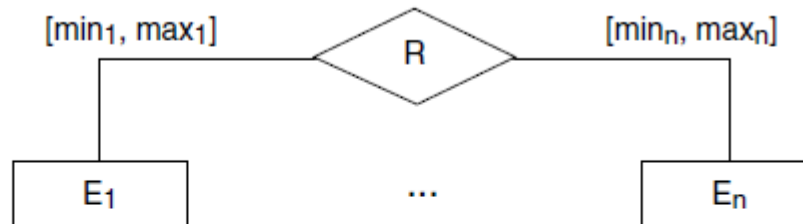
Wird Adresse gelöscht, muss Lieferant(en) mitgelöscht werden.



Wird Lieferant gelöscht, muss Adresse mitgelöscht werden, wenn nicht weiterer Lieferant zur Adresse existiert.



Lieferant und Adresse sind existenzabhängig



Schreibweise:

$$R(E_1[\min_1, \max_1], \dots, E_i[\min_i, \max_i], \dots, E_n[\min_n, \max_n])$$

verantwortlich_für (Mitarbeiter[0,*], Rechner[1,1])

„Es gibt Mitarbeiter, die für keinen Rechner oder die für alle Rechner verantwortlich sind. Jedem Rechner ist genau ein Mitarbeiter zugeordnet, der verantwortlich für den Rechner ist.“



Ohne formale Notation im ER-Diagramm

Datentyp

Integer, String, ...

Wertebereich / Domäne

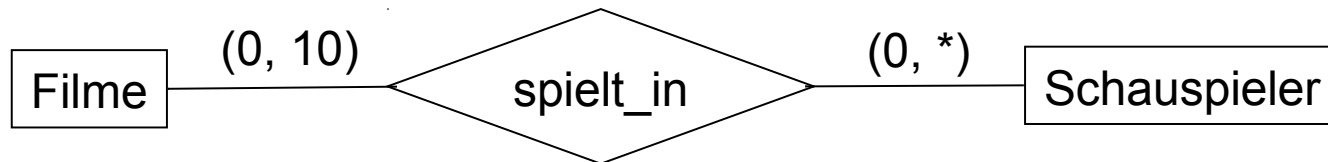
≤ 100 , {Krimi,Doku,Zeichentrick}

Länge eines Attributes

Stringlänge < 25

Kardinalität von Relationships

Höchstens 10 Schauspieler pro Film





Notation von Kardinalitäten (Multiplizitäten)

(Min,Max)	Erläuterung
(0,1)	Vertreter des Entitätentyps / der Klasse stehen mit keinem oder maximal einem anderen Vertreter des anderen Entitätentyps / der anderen Klasse in Beziehung.
(1,1)	Vertreter des Entitätentyps / der Klasse stehen mit genau einem anderen Vertreter des anderen Entitätentyps / der anderen Klasse in Beziehung.
(0,*)	Vertreter des Entitätentyps / der Klasse stehen mit keinem oder einer (unbekannten) beliebigen Anzahl von Vertreter des anderen Entitätentyps / der anderen Klasse in Beziehung.
(1,*)	Vertreter des Entitätentyps / der Klasse stehen mit mindestens einem oder einer (unbekannten) beliebigen Anzahl von Vertreter des anderen Entitätentyps / der anderen Klasse in Beziehung.
(2,5) oder (m,n)	Dies ist ein stellvertretendes Beispiel für beliebige andere Varianten, wo die genaue Mindest- und Maximalanzahl der möglichen beteiligten Vertreter spezifiziert werden kann. (Mindestens 2 und maximal 5 Vertreter)





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)	



Die (min,max)-Notation zählt die Ausprägung von Beziehungen, während die Chen Notationen Entitätstypausprägungen zählen.

(min,max) [Entity 1]	Chen-Notation	(min,max) [Entity 2]
(0,1)	1:1	(0,1)
(0,N)	1:N +Totalität	(0,1)
(0,N)	1:N +Totalität	(1:1
(0,N)	M:N	(0,N)
(1,1)	Totalität +1:1	(0,1)
(1,N)	Totalität +1:N	(0,1)
(1,1)	Totalität +1:1 +Totalität	(1,1)
(1,N)	Totalität+1:N +Totalität	(1,1)
(1,N)	Totalität +M:N	(0,N)
(1,N)	Totalität +M:N +Totalität	(1,N)

Sie dürfen in Ihren
Arbeiten und Klausuren
**eine von beiden
Notationen** dann
einheitlich verwenden.

<http://de.wikipedia.org/wiki/Min-Max-Notation>



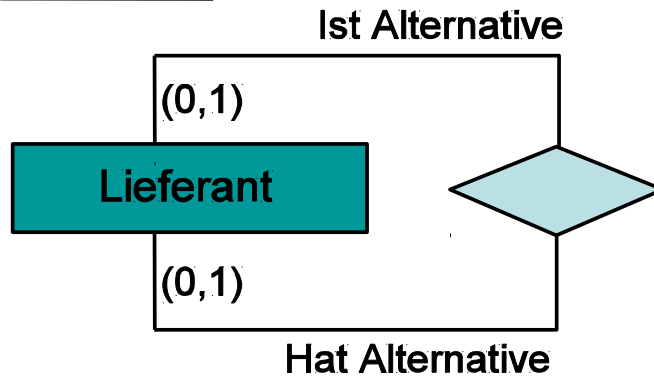
Arten von Beziehungstypen

- Nach der Anzahl der beteiligten Entitätentypen (auch Grad der Beziehung) werden unterschieden:
 - Binäre Beziehungstypen (2 Entitätentypen)
 - Siehe bisherige Beispiele
 - Rekursive / unäre Beziehungstypen (1 Entitätentyp)
 - An einem rekursiven Beziehungstyp nimmt derselbe Entitätentyp mehrfach teil. Eine Entität nimmt dabei verschiedene Rollen ein, Bsp.:
 - Abteilung ist übergeordnete Abteilung zu anderer (untergeordneter) Abt.
 - Abteilung ist untergeordnete Abteilung zu anderer (übergeordneter) Abt.
 - N-äre Beziehungstypen (n Entitätentypen)
 - An einem n-ären Beziehungstyp nehmen n Entitätentypen teil.

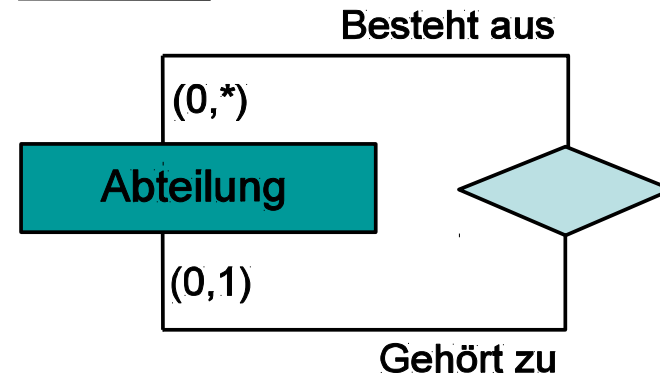


Rekursiver Beziehungstyp

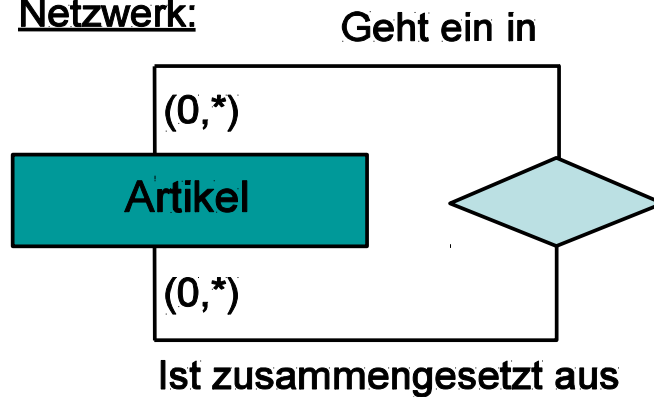
Alternative :



Hierarchie:



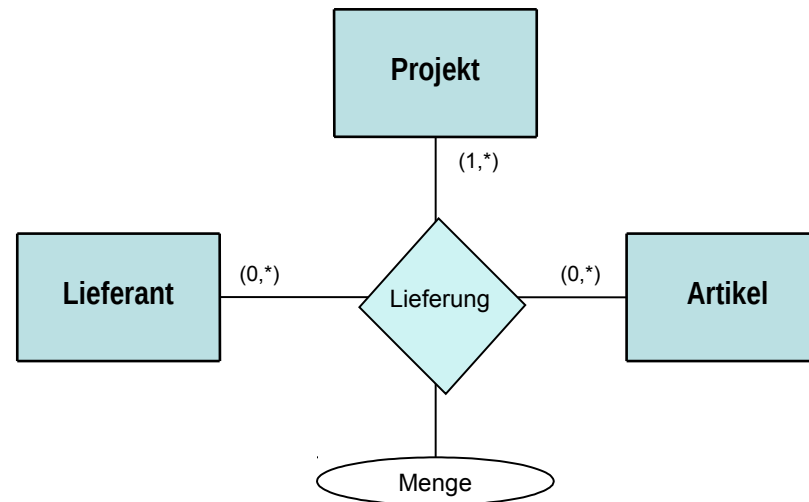
Netzwerk:





n-ärer Beziehungstyp

- Beispiel: Artikel werden für Projekte in einer bestimmten Menge benötigt und können von Lieferanten geliefert werden:
 - Ein Artikel muß nicht für ein Projekt geliefert werden (Eigenproduktion), kann aber für viele Projekte geliefert werden: $(0, *)$.
 - Ein Projekt muss mindestens einen Artikel verwenden: $(1, *)$.
 - Ein Lieferant kann Artikel liefern.

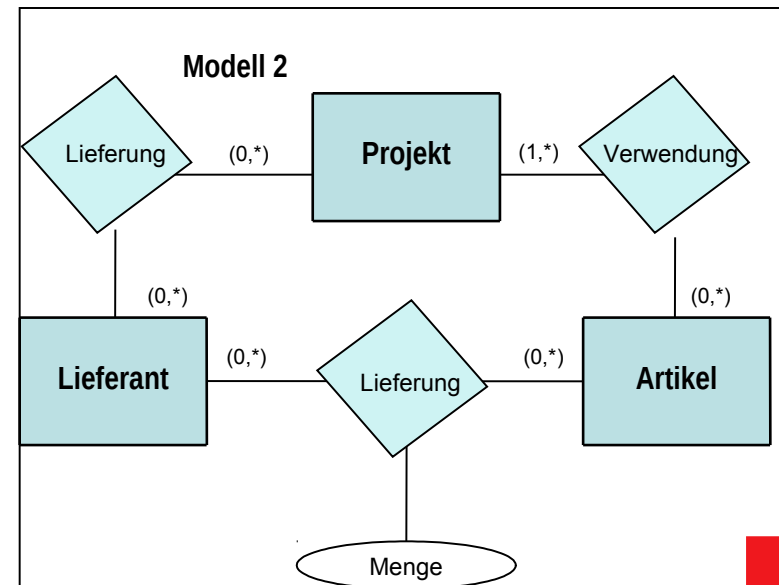
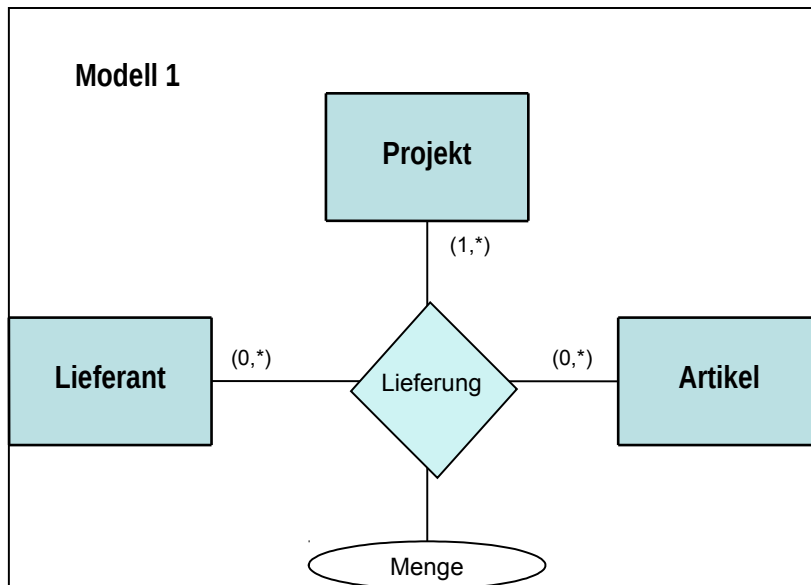




n-ärer Beziehungstyp

Ein n-ärer Beziehungstyp kann nicht immer verlustfrei in binäre Beziehungstypen aufgebrochen werden!

Modell 1 repräsentiert nicht Modell 2 ! (Bei Modell 2 können falsche Beziehungen aus der Kombination der Beziehungstypen abgeleitet werden.)



■ Grundlagen der abstrakten Datenmodellierung

- Einleitung: Was sind Modelle und wozu braucht man sie?
- Grundlegende Elemente des ERM
- Modellierung von Nebenbedingungen
- **Exkurs: Erweiterungen des ERM**
- In der Übung: Vorgehen und Beispiel





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.

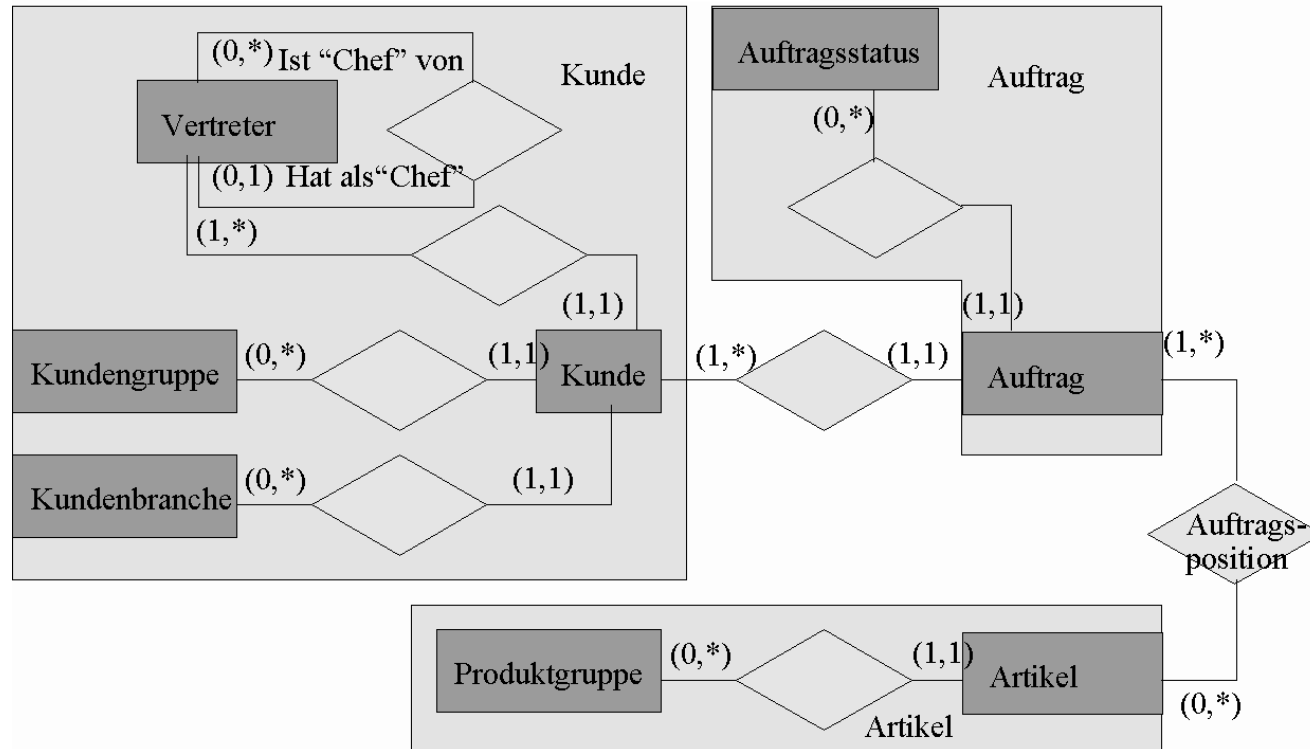
Generalisierung / Spezialisierung

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



Aggregation/ Komposition

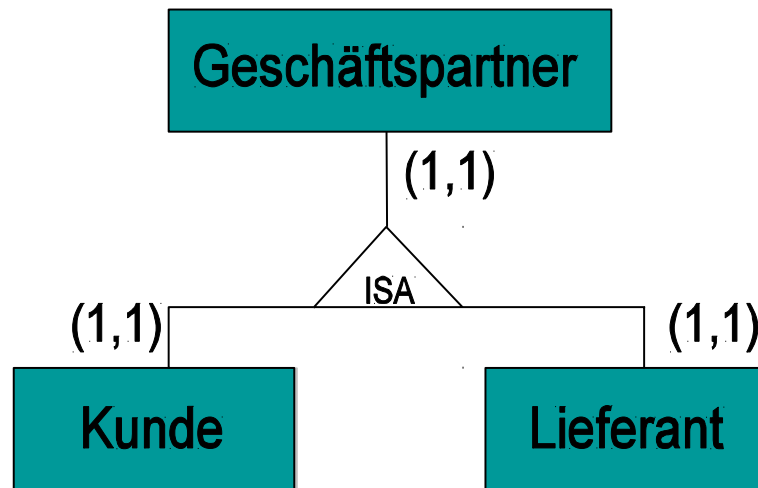
- Zusammenfassung verschiedener Elemente eines Schemas zu einem zusammengesetzten Element





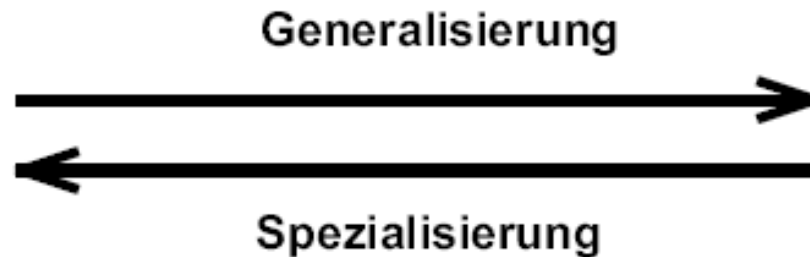
Abstrahierungskonzept Generalisierung

- Zusammenfassung ähnlicher Elemente eines Schemas zu einem generischen Element





Beispiel Generalisierung/Spezialisierung

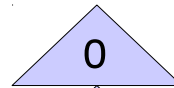


Abstrahierungskonzept Generalisierung

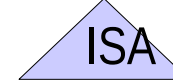
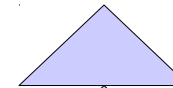
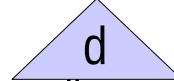
■ Mengenbeziehungen von Subtypen (nach Zehnder)

- 1. Betrachtung: Überschneidung der Entitäten der Subtypen?
(Existieren Entitäten, die 2 Subtypen zugeordnet werden können?)

- Überschneidung (ja)



- Disjunktheit (nein)



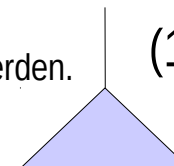
- 2. Betrachtung: Vollständige Überdeckung der Vereinigungsmenge der Subtypen über die Menge der Supertypen?

(Existieren Entitäten, die keinem Subtyp zugeordnet werden können?)

- Vollständige Überdeckung

Alle Entitäten können einem Subtyp zugeordnet werden.

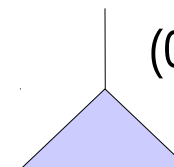
(1,...)



- Unvollständige Überdeckung

Nicht alle Entitäten können einem Subtyp zugeordnet werden.

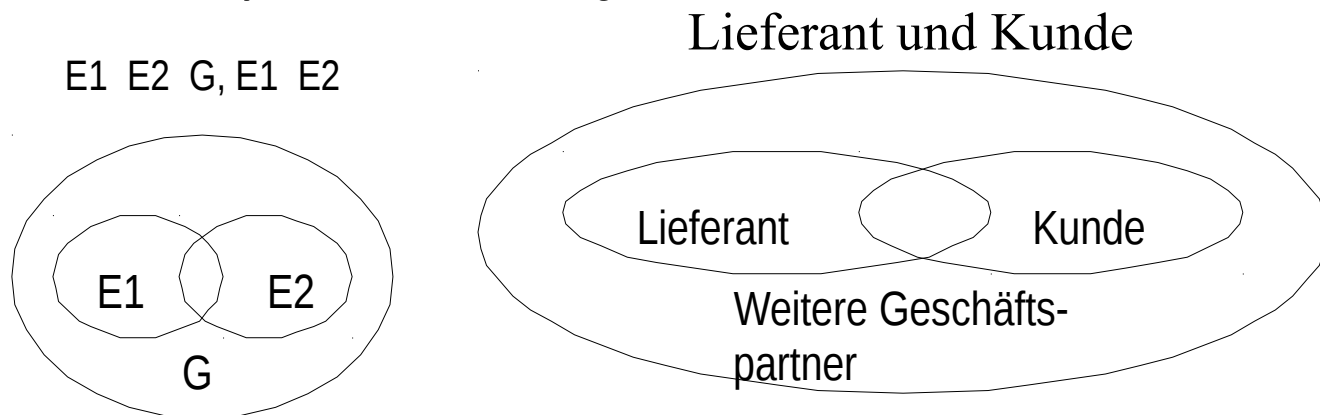
(0,...)



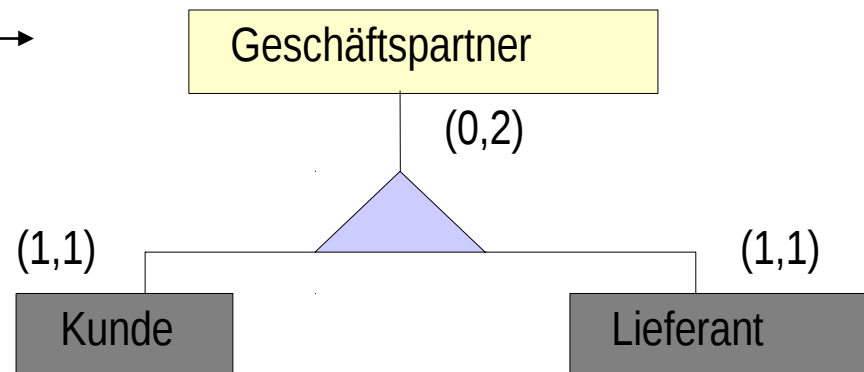
Abstrahierungskonzept Generalisierung

■ Mengenbeziehungen von Subtypen (nach Zehnder)

■ Nicht disjunkt, unvollständig



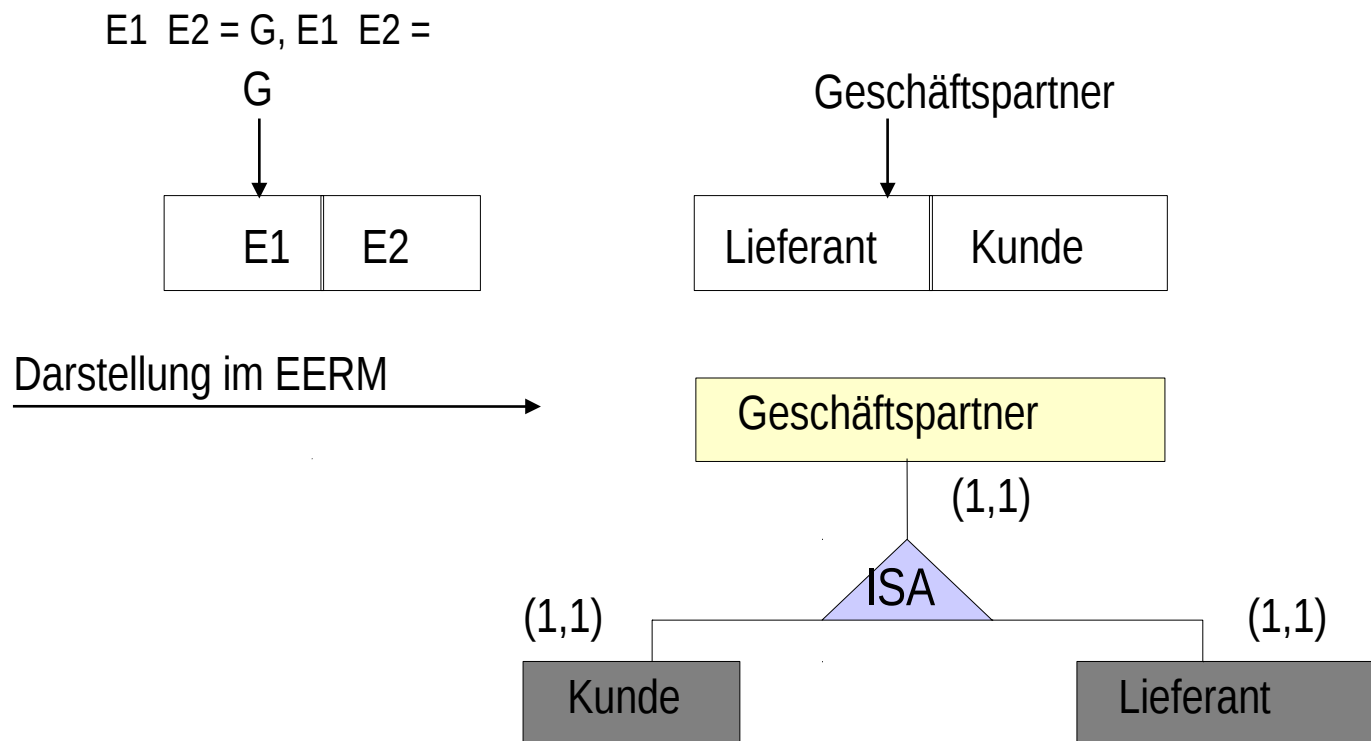
Darstellung im EERM



Abstrahierungskonzept Generalisierung

■ Mengenbeziehungen von Subtypen (nach Zehnder)

■ disjunkt, vollständig



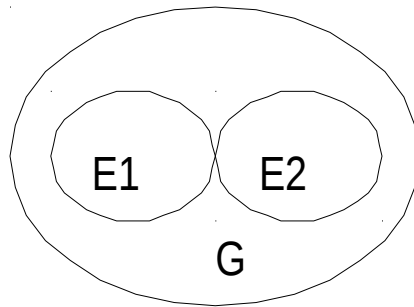


Abstrahierungskonzept Generalisierung

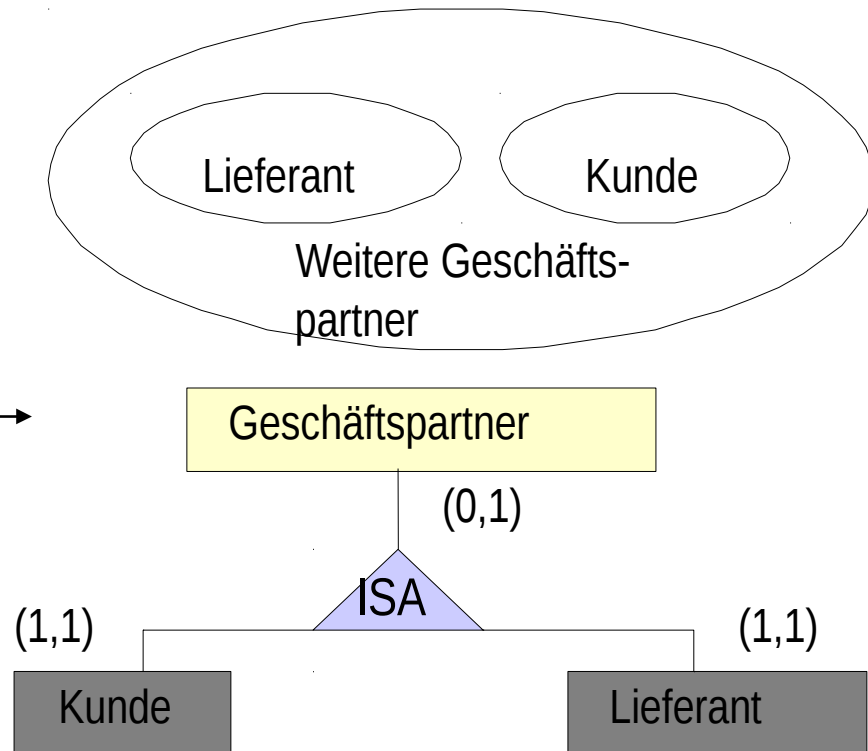
■ Mengenbeziehungen von Subtypen (nach Zehnder)

■ disjunkt, unvollständig

$E1 \cap E2 = \emptyset, E1 \cup E2 \subsetneq G$



Darstellung im EERM

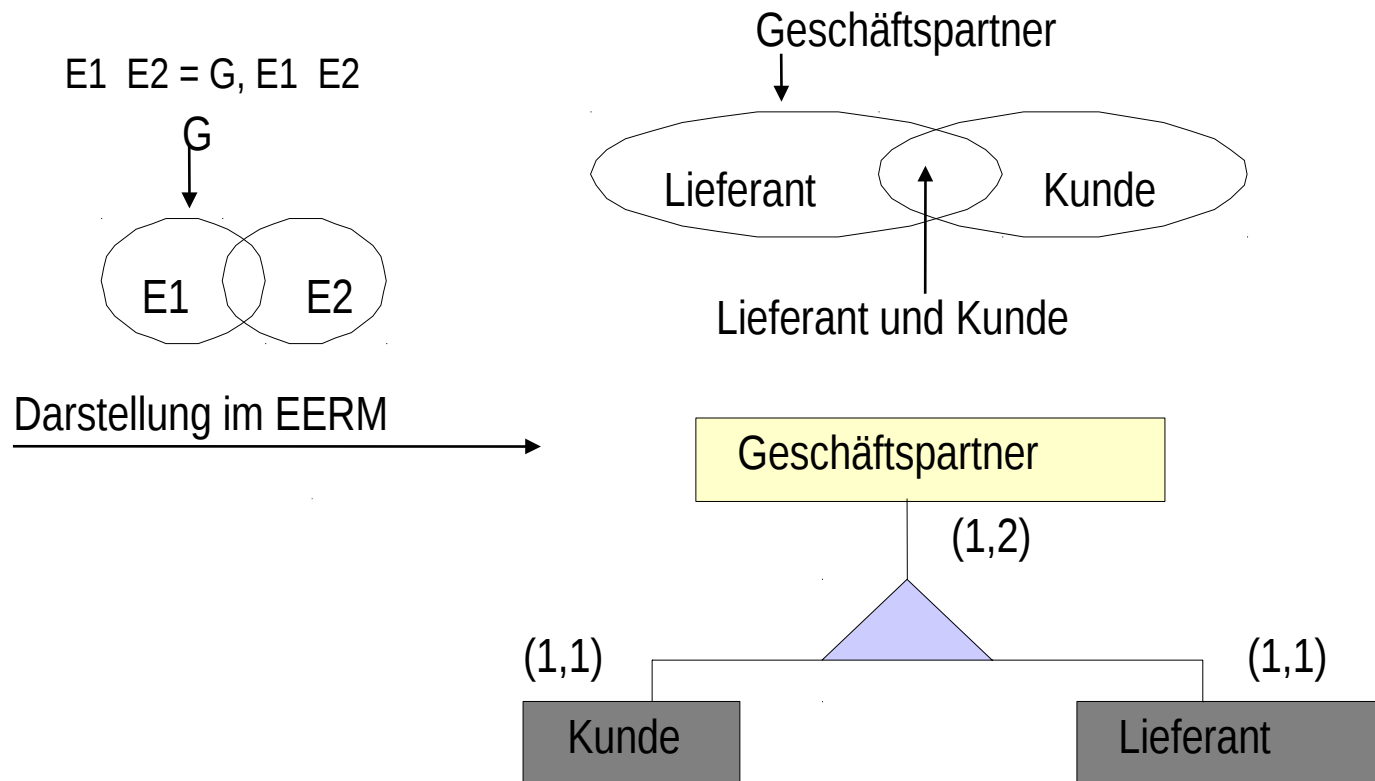




Abstrahierungskonzept Generalisierung

■ Mengenbeziehungen von Subtypen (nach Zehnder)

■ Nicht disjunkt, vollständig





Zusammenfassung: Generalisierung und Spezialisierung

■ Spezialisierung

- entspricht der IST-EIN-Beziehung
- Z.B. Ein Drache (Spezialisierung) IST-EIN Produkt
- Partitionierung (Spezialfall der Spezialisierung)
 - mehrere disjunkte Entity-Typen (disjoint vs. overlapping)
 - Partitionierung von Produkten in Zubehör und Drachen

■ Generalisierung

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

■ Vollständige Taxonomie

- Übersicht über alle Spezialisierungen (total vs. partial)



- Motivation und Einbettung
- Begriffe und Definitionen
- ER-Diagramme
- Modellierung von Nebenbedingungen
- Erweitertes ER-Modell
- (Designprinzipien)

In der nächsten Veranstaltung:

- Relationaler Entwurf
 - Das Relationale Modell
 - Von ER-Diagrammen zu Relationenschemata
 - Konvertierung von Spezialisierung
 - Funktionale Abhängigkeiten (FDs)
 - Normalformen



Aufgabe 1 Anfragen & Modellierung“

Denken Sie mal darüber nach, welche Anfragen Sie an die AOL Daten stellen möchten. Bitte modellieren Sie ein logisches und physisches Schema zur Beantwortung dieser Anfragen.

Aufgabe 2 „SQL und Abfrageausführung“

Bitte formulieren Sie für Ihre Analyseideen aus 1.) die SQL Anfragen. Sie verstehen auch Möglichkeiten der Abfrageausführung bzw. Optimierung.

Aufgabe 3 „Datenintegration“

Zur Ausführung der Ausführung fehlen Ihnen noch externe Daten, z.B. aus dem Internet Archive, DMOZ oder Freebase.org. Bitte ergänzen Sie Ihr Schema und die Datenbasis.

Aufgabe 4 „Analyse, Erkenntnisgewinn und Wert“

Stellen Sie in 10 Minuten die wichtigsten Erkenntnisse aus den Daten vor. Bewerten Sie den Erkenntnisgewinn, z.B. gegenüber Ihren Kommilitonen oder der Literatur! Welche Erkenntnisse hätten einen kommerziellen Wert?

Was sind Datenbanken?

- Motivation, Historie, Datenunabhängigkeit, Einsatzgebiete

Datenbankentwurf im ER-Modell & Relationaler Datenbankentwurf

- Entities, Relationships, Kardinalitäten, Diagramme

- Relationales Modell, ER -> Relational, Normalformen, Transformationseigenschaften

Relationale Algebra & SQL

- Kriterien für Anfragesprachen, Operatoren, Transformationen

- SQL DDL, SQL DML, SELECT ... FROM ... WHERE ...

Datenintegration & Transaktionsverwaltung

- JDBC, Cursor, ETL

- Mehrbenutzerbetrieb, Serialisierbarkeit, Sperrprotokolle, Fehlerbehandlung, Isolationsebenen in SQL

Ausblick

- Map/Reduce, HDFS, Hive ...

- Wert von Daten

■ Grundlagen der abstrakten Datenmodellierung

- Einleitung: Was sind Modelle und wozu braucht man sie?
- Grundlegende Elemente des ERM
- Modellierung von Nebenbedingungen
- Erweiterungen des ERM
- **In der Übung: Vorgehen und Beispiel**





DESIGNPRINZIPIEN



- 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
- 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!)



- Erstellung eines Glossars der konzeptuell relevanten Begriffe
 - d.h. eine strukturierte, stichwortartige textuelle Beschreibung der wesentlichen Begriffe
- ... parallel dazu
 - vorläufige Einteilung dieser Begriffe in Entity-Typen, Attribute und Relationship-Typen
 - 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** sowie
- **Untersuchung des Modells auf mögliche Abstraktionen** durch Verwendung geeigneter Abstraktions- und Modularisierungskonzepte





- ... liefert als Ergebnis das komplette E/R-Modell mit allen Begleitdokumenten.
- Erstmalig erfolgt eine ausreichende Präzisierung der relevanten Begrifflichkeit im Diskursbereich für einen möglichen Datenbank-Entwurf:
 - durch vorliegende oder neu erstellte textuelle Beschreibungen / Anforderungsdefinitionen,
 - durch E/R-Typ-Diagramm mit Integritätsbedingungen
 - das Begriffsglossar, inklusive funktionalen und dynamischen Zusammenhängen und 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





BEISPIEL



Beschreibung eines Ausschnitts einer Anwendungsdomäne

- Ein Unternehmen verkauft **Artikel** über ein Netz von Vertretern.
- **Vertreter** sind **hierarchisch** jeweils einem Hauptvertreter unterstellt (**Hauptvertreter** müssen nicht unterstellt sein.).
- Jeder Vertreter betreut einen ihm zugewiesenen Kreis von **Kunden**. Ein Kunde wird genau von einem Vertreter betreut.
- Kunden gehören zu einer **Branche** (z.B. EDV, Chemie, etc.)
- und werden in Abhängigkeit vom Auftragsvolumen einer **Kundengruppe** zugeordnet (Großabnehmer, Normalkunde, etc.).
- Von den Kunden werden **Aufträge** entgegengenommen, die jeweils ein bis mehrere Artikel des Sortiments des Unternehmens in bestimmten Mengen umfassen.
- Einzelne **Auftragspositionen** können mit Teillieferungen beliefert werden, so dass die noch zu liefernde Restmenge jederzeit ablesbar sein muss. Pro Position werden ein **Liefertermin** und ein positionsspezifischer Preis vereinbart.
- Das **Artikelsortiment** des Unternehmens ist in verschiedene **Produktgruppen** unterteilt (Fahrzeuge, Ersatzteile, etc.).
- Die Kundenaufträge werden je nach Bearbeitungsstand in 4 Status-Kategorien eingeordnet (erfasst, bestätigt, geliefert, fakturiert). Jeder Auftrag kann zu einem Zeitpunkt nur genau einen **Status** annehmen.



Datenmodell als ER-Diagramm

