






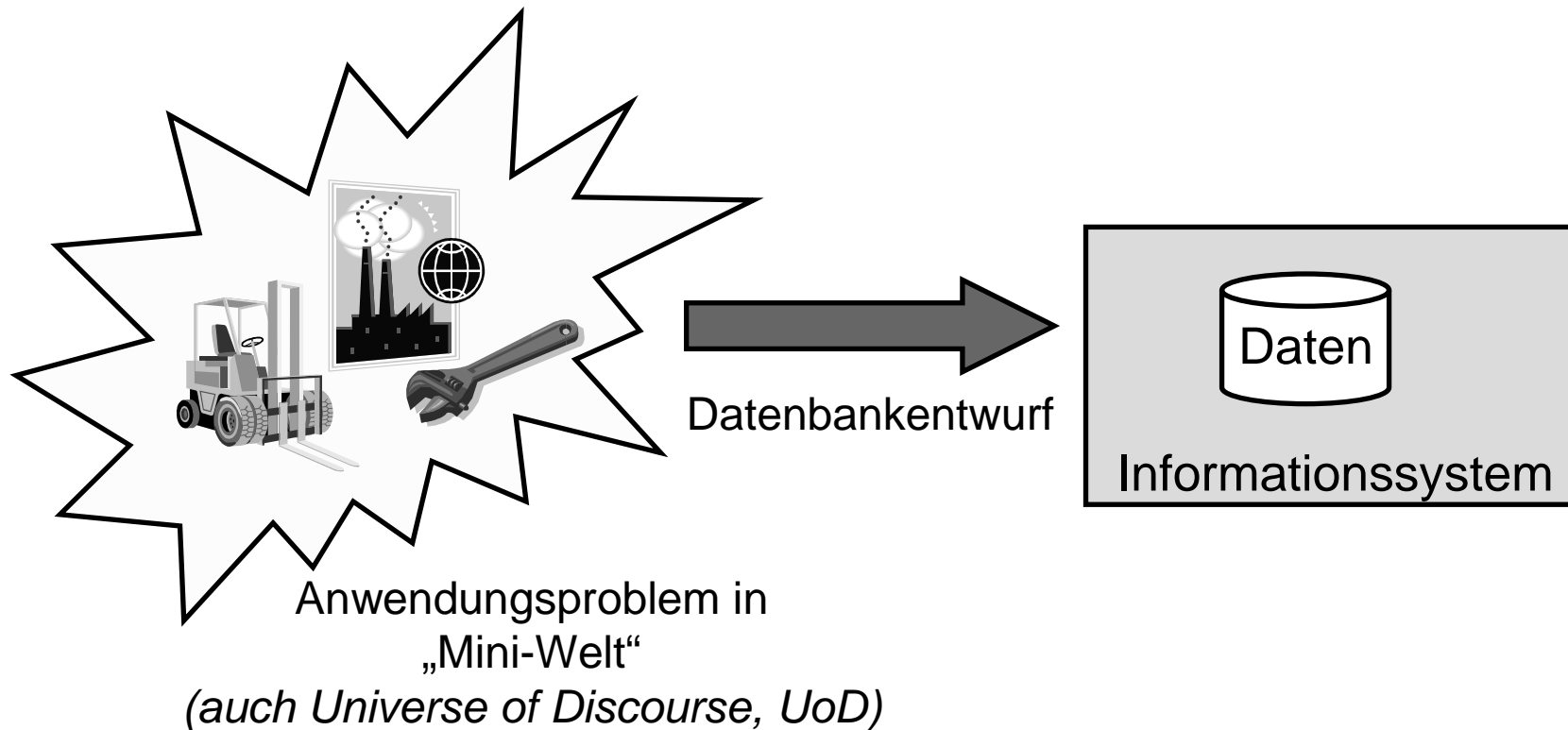
6 DB-Entwurf

6.0	Vorgehensmodell	2	
6.1	Anforderungsanalyse	6	
6.2	Konzeptueller Entwurf	26	
6.3	Logischer Entwurf	30	
6.4	Physischer Entwurf	65	

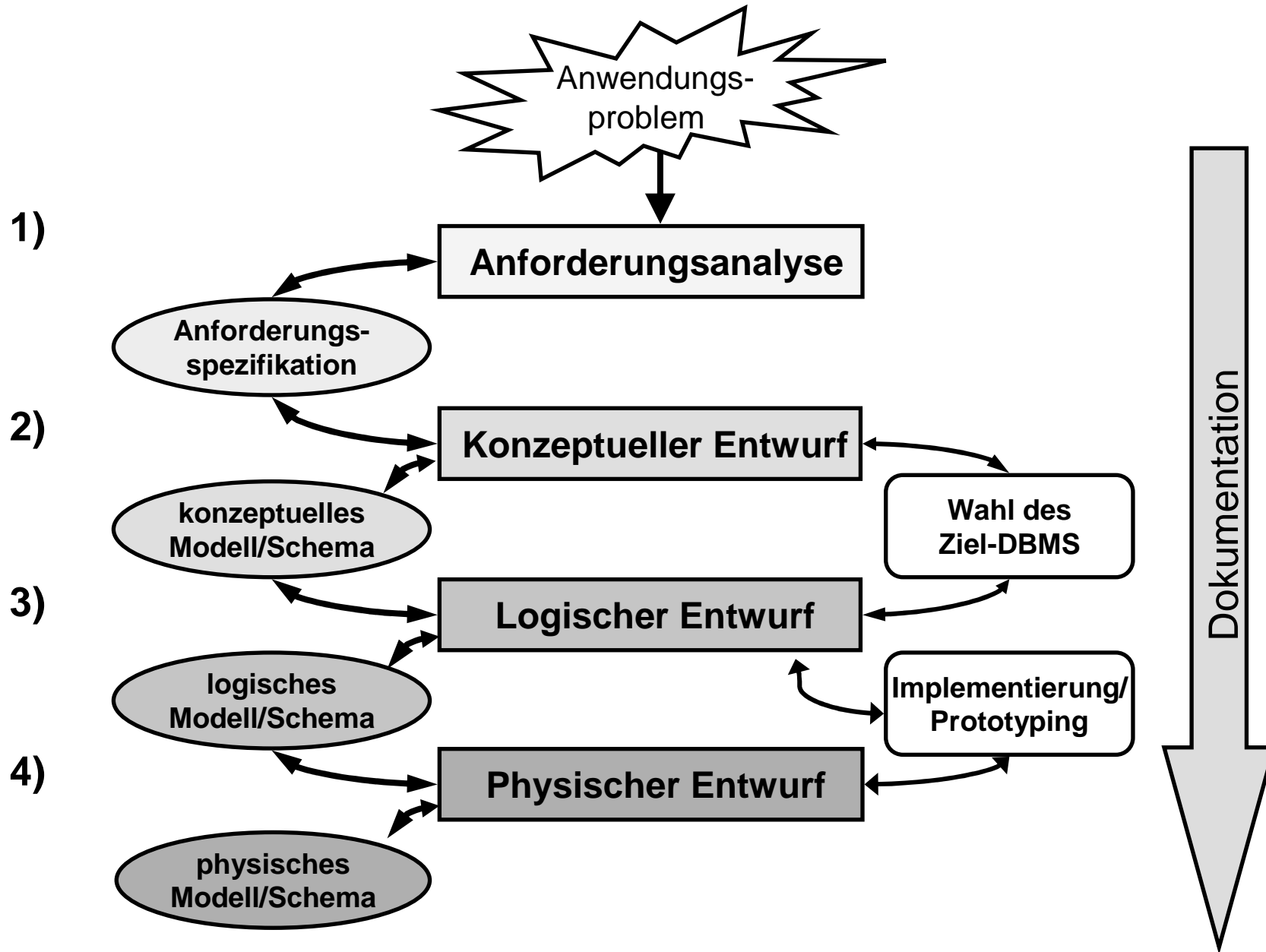
6.0 Vorgehensmodell (1/3)

Ziele des Datenbankentwurfs

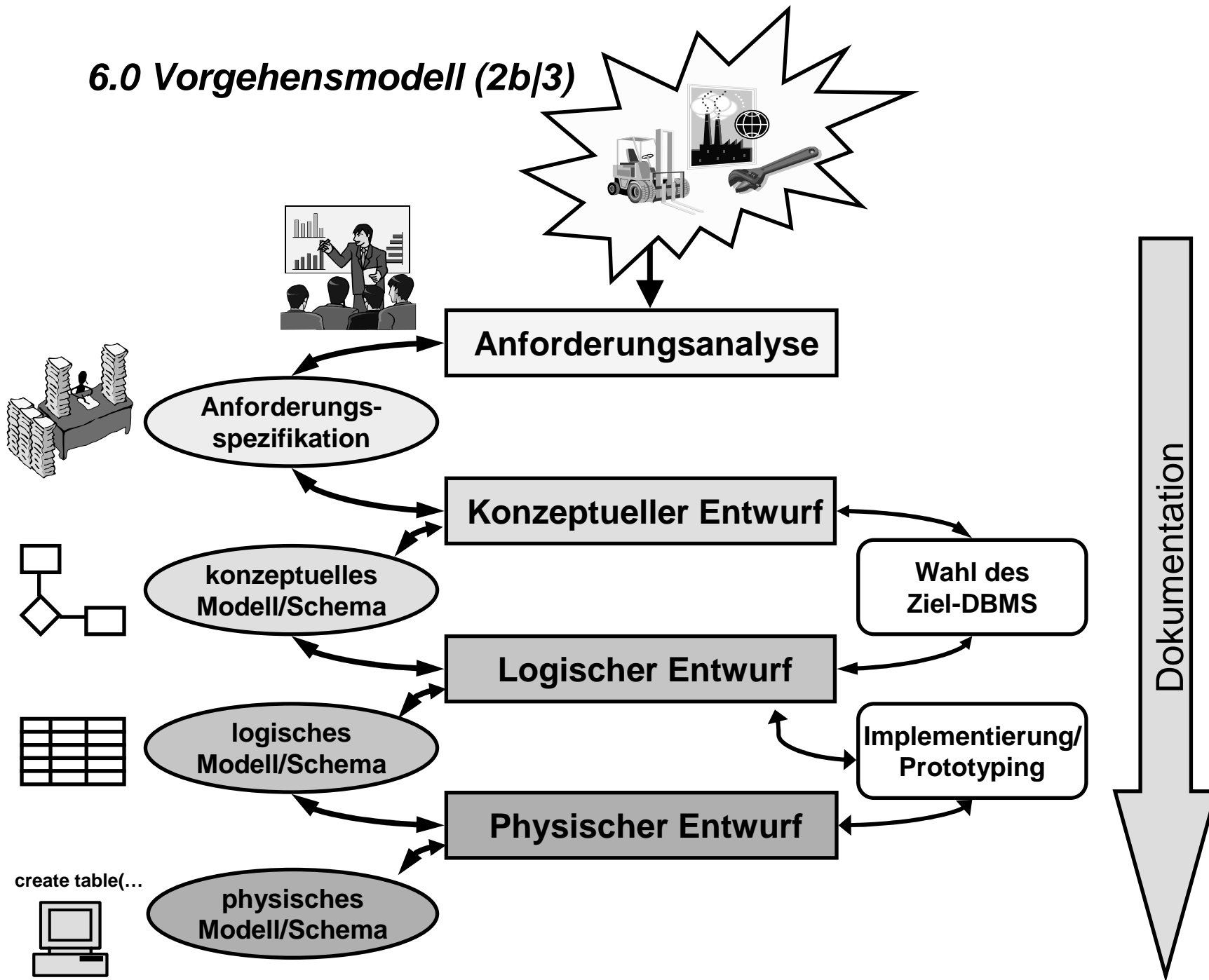
- Abbildung eines Ausschnitts der realen Welt auf ein Datenbankschema
- Dabei: Systematisches Vorgehen
- Ingenieursmäßiger Ansatz, wie im Software Engineering



6.0 Vorgehensmodell (2a/3)



6.0 Vorgehensmodell (2b/3)



6.0 Vorgehensmodell (3/3)

Anmerkungen zum Vorgehensmodell

- Vorgehen muss nicht streng hierarchisch sein
- Zyklen sind möglich
- Dokumentation ist Querschnittsfunktion



6 DB-Entwurf

6.0	Vorgehensmodell	2
6.1	Anforderungsanalyse.....	6
6.2	Konzeptueller Entwurf	26
6.3	Logischer Entwurf.....	30
6.4	Physischer Entwurf	65

6.1 Anforderungsanalyse (1/2)

1) Anforderungsanalyse

- Ziel: Sammlung & Dokumentation der Anforderungen aller potenziellen Benutzer
 - Was für Daten sollen gespeichert werden?
 - Wie sollen sie gespeichert werden?
- 2 Arten von Anforderungen:
 - Informationsanforderungen
 - statische Informationen, die das DBMS später benutzen wird
 - z.B. Angaben über Daten, Typen, charakterisierende Eigenschaften (Attribute), Wertebereiche, Identifikatoren („Schlüssel“), Konsistenzbedingungen („Integritätsbed.“)
 - Bearbeitungsanforderungen
 - Spezifikation dynamischer Aktivitäten & Prozesse für die DB
 - Welche Prozesse werden wie häufig durchgeführt?
 - Abhängigkeiten zw. Prozessen / Datenvolumen
 - Verfügbarkeit / Sicherheit
 - z.B. Anfragen / Updates, Report-Erzeugung

6.1 Anforderungsanalyse (2/2)

Anforderungsanalyse

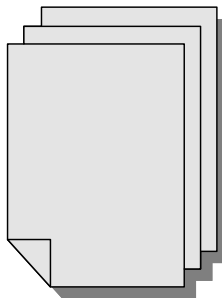
Konzeptueller Entwurf

Logischer Entwurf

Physischer Entwurf

- Typische Aktivitäten:
 - Benutzergruppen und Anwendungsbereiche identifizieren
 - Sichtung existierender Dokumentation
 - Feststellung von Herkunft, Ziel, Volumen der Daten
 - Interviews / Fragebögen

- Ergebnis der Phase:
 - Anforderungsspezifikation



Anforderungsanalyse
Konzeptueller Entwurf
Logischer Entwurf
Physischer Entwurf

6.1 Beispiel Anforderungsanalyse (1/17)

(Methode nach DeAntonellis/ Demo 1983, vgl. auch Mayr/ Dittrich/ Lockemann 1987)

Prinzip der Vorgehensweise^[1]:

natürlichsprachl. Beschreibungen

schrittweise Formalisierung ↓

formalisierte Verzeichnisse

[1] Alternative Vorgehensweise: Use Case Analyse (vgl. Objektorientierte Softwareentwicklung mit der Unified Modeling Language (UML))

6.1 Beispiel Anforderungsanalyse (2/17)

Anforderungsanalyse
Konzeptueller Entwurf
Logischer Entwurf
Physischer Entwurf

Vorgehensweise im einzelnen:

Anwendungsumwelt, Vertrag

Anforderungsermittlung in natürlicher Sprache:

- Identifikation Organisationseinheit(en)
- Identifikation zu unterstützender Aufgaben
- Erstellung eines Anforderungssammelplans
- Sammlung der Anforderungen



Anforderungssammelblätter

Filterung: Synonyme/Homonyme ersetzen

Korrektur bzgl. Einfachheit und Verständlichkeit



6.1 Beispiel Anforderungsanalyse (3/17)

Anforderungsanalyse
Konzeptueller Entwurf
Logischer Entwurf
Physischer Entwurf



gefilterte Anforderungssammelblätter

Satzklassifikation: Aufteilung nach
Objekten, Operationen, Ereignissen



Objekt-, Operations-, Ereignis-Sammelblätter



**Objekt-
verzeichnis/
Glossar**



**Operations-
verzeichnis/
Glossar**



Übertragung

**Ereignis-
verzeichnis/
Glossar**

6.1 Beispiel Anforderungsanalyse (4/17)

*Im Folgenden: Beispiele für Dokumente aus der
Anforderungsanalyse einer
Konferenzorganisation*

- a) *Querreferenz Organisationseinheit – Aufgabe*
- b) *Querreferenz Aufgabe – Informationslieferant*
- c) *Anforderungssammelblatt*
- d) *Anforderungssammelblatt (gefiltert)*
- e) *Objekt-Sammelblatt*
- f) *Operations-Sammelblatt*
- g) *Ereignis-Sammelblatt*
- h) *Objekt-Glossar*
- i) *Operations-Glossar*
- j) *Ereignis-Glossar*

Anforderungsanalyse

Konzeptueller Entwurf

Logischer Entwurf

Physischer Entwurf

AIFBO

WS05/06

6.1 Beispiel Anforderungsanalyse (5/17)

Anforderungsanalyse
Konzeptueller Entwurf
Logischer Entwurf
Physischer Entwurf

a)

Querreferenz Organisationseinheit -- Aufgabe		
Aufgaben	Programm	Organisation
...		
Programm_zusammenstellen	×	
Tagungsband_zusammenstellen	×	
Einladen		×
Anmeldungen_verwalten		×
Terminplanung	×	×
Programmkomitee_bilden	×	×
Lokales_Organisationskomitee_bilden		×
Tagungssekretariat_belegen		×
Hotelzimmer_belegen		×
Finanzplanung		×
...		

6.1 Beispiel Anforderungsanalyse (6/17)

Anforderungsanalyse

Konzeptueller Entwurf

Logischer Entwurf

Physischer Entwurf

- b)** Als Informationslieferanten kommen beispielsweise in Frage
- Leiter und Mitglieder der/ von Organisations- und Programm-Komitees:
 - LO_1, \dots, LO_j ;
 - MO_1, \dots, MO_j ;
 - LP_1, \dots, LP_k ;
 - MP_1, \dots, MP_l
 - Mitglieder der Tagungssekretariate vergangener Konferenzen:
 - MT_1, \dots, MT_m
 - Vertreter der die Konferenz veranstaltenden Gesellschaften:
 - KG_1, \dots, KG_n
 - evtl. Teilnehmer an vergangenen Konferenzen:
 - TK_1, \dots, TK_p
 - Vertreter von Reisebüro, Bank, ...
 - VR, VB, ...

6.1 Beispiel Anforderungsanalyse (7/17)

Anforderungsanalyse
Konzeptueller Entwurf
Logischer Entwurf
Physischer Entwurf

b)

Querreferenz Aufgabe -- Informationslieferant						
	Programm- _zusam- menstellen	Tagungs- band_ zusammen- stellen	Einla- den	Anmel- dungen_ verwal- ten	Termin- planung	
LO ₁ , ..., LO _i			×		×	
MO _{1,...} , MO _j				×	×	
LP ₁ , ..., LP _k	×	×			×	
MP ₁ , ..., MP _l	×	×			×	
MT _{1,...} , MT _m			×	×	×	
KG _{1,...} , KG _n			×		×	
TK ₁ , ..., TK _p					×	
...						

6.1 Beispiel Anforderungsanalyse (8/17)

Anforderungsanalyse
Konzeptueller Entwurf
Logischer Entwurf
Physischer Entwurf

c)

Anforderungs-Sammelblatt	
Organisationseinheit: Organisation	Seite: 1/1
Aufgabe: Einladen	Version: 1
Informationslieferant: LO ₄	Datum: 28.11.2004
<p><i>Eine Einladung zur Teilnahme an der Tagung wird in Form einer Vorankündigung an alle Mitglieder der die Tagung veranstaltenden Organisation (z.B. GI, IFIP, ACM, IEEE, etc.) versandt. Dies erfolgt ca. 1 Jahr vor Stattfinden der Tagung. Möglicherweise treten weitere Organisationen als Mitveranstalter auf. Dann sind die Einladungen auch an deren Mitglieder zu verschicken. Zusätzliche potentielle Interessenten an der Konferenz können durch die Mitglieder des Organisationskomitees und des Programmkomitees benannt werden. Das Tagungsprogramm wird zusammen mit einer förmlichen Einladung an den o.g. Personenkreis geschickt. Hinzukommen die Personen, die Manuskripte für die Konferenz eingereicht haben, und Personen, die durch Anschreiben an ein Mitglied des Organisationskomitees Interesse an der Tagung bekundet haben. Es ist zu vermeiden, dass dieselbe Person mehrere Einladungen erhält. Auch sollen Personen, die sich bereits angemeldet haben, keine weitere Einladung mehr erhalten.</i></p>	

6.1 Beispiel Anforderungsanalyse (9/17)

Anforderungsanalyse

Konzeptueller Entwurf

Logischer Entwurf

Physischer Entwurf

d)

Anforderungs-Sammelblatt	
Organisationseinheit: Organisation	Seite: 1/1
Aufgabe: Einladen	Version: 2 (Vers.1 gefiltert)
Informationslieferant: LO ₄	Datum: 28.11.2004
<p><i>Vorankündigung wird ca. 1 Jahr vor Konferenzbeginn an Interessent verschickt. Interessent ist Mitglied in der die Konferenz veranstaltenden Gesellschaft (GI, IFIP, ACM, IEEE, AFCET, etc.). Falls weitere Gesellschaften als Mitveranstalter auftreten, dann sind auch deren Mitglieder Interessent. Organisationskomitee- und Programmkomitee-Mitglieder benennen weitere Interessenten. Konferenzprogramm wird gedruckt und dann zusammen mit Einladung an Interessent geschickt. Interessent sind auch Personen, die Manuskripte eingereicht haben, und Personen, die durch Anschreiben an Mitglied des Organisationskomitees Interesse an Konferenz bekundet haben. Kein Interessent erhält mehrere Einladungen.</i></p>	

AIFBO

WS05/06

6.1 Beispiel Anforderungsanalyse (10/17)

Anforderungsanalyse

Konzeptueller Entwurf

Logischer Entwurf

Physischer Entwurf

e)

Objekt-Sammelblatt

Organisationseinheit: Programm

Seite: 1/1 **Version:** 1

Ersteller: vpa

Datum: 21.12.2004

D1. Sitzungsleiter ist Mitglied des Programmkomitees.

D2. Sitzung besteht aus drei eingereichten und einem eingeladenen Vortrag.

D3. Konferenzprogramm besteht aus Menge von Sitzungen.

D4. Sitzung wird Sitzungsleiter zugeordnet.

D5. Sitzung wird Datum und Uhrzeit zugeordnet.

D6. Sitzung wird Hörsaal zugeordnet.

...

Vgl. auch Objekt-Glossar 

6.1 Beispiel Anforderungsanalyse (11/17)

Anforderungsanalyse

Konzeptueller Entwurf

Logischer Entwurf

Physischer Entwurf

AIFBO

WS05/06

f)

Operations-Sammelblatt	
Organisationseinheit: Programm Ersteller: vpa	Seite: 1, Version: 1 Datum: 21.12.2004
O1. <u>Bestimme</u> Mitglied des Programmkomitees. O2. <u>Versicke</u> Call-for-papers. O3. <u>Lehne</u> Manuskripte <u>ab</u> . O4. <u>Bilde</u> Konferenzprogramm. O5. <u>Richte</u> Sitzungen <u>ein</u> . O6. <u>Lege</u> Datum und Uhrzeit der Sitzungen <u>fest</u> . O7. <u>Bestimme</u> Sitzungsleiter. O8. <u>Schicke</u> Zusagen an Autoren.	

Vgl. auch Operations-Glossar 

6.1 Beispiel Anforderungsanalyse (12/17)

Anforderungsanalyse

Konzeptueller Entwurf

Logischer Entwurf

Physischer Entwurf

g)

Ereignis-Sammelblatt	
Organisationseinheit: Programm	Seite: 1, Version: 1
Ersteller: vpa	Datum: 21.11.2004
E1. ca. 1 Jahr vor Konferenzbeginn: <ul style="list-style-type: none">– verschicke Call for papers.	
E2. Entscheidung Konferenz wird durchgeführt: <ul style="list-style-type: none">– stelle Programmkomitee zusammen.	
E3. Konferenzprogramm festgelegt: <ul style="list-style-type: none">– schicke Zusagen an Autoren.	
E4. Konferenzprogramm festgelegt: <ul style="list-style-type: none">– schicke Absagen an Autoren.	

Vgl. auch Ereignis-Glossar



6.1 Beispiel Anforderungsanalyse (13/17)

Anforderungsanalyse
Konzeptueller Entwurf
Logischer Entwurf
Physischer Entwurf

h)

Objekt-Glossar (30.11.2004)							
Organisationseinheit: Programm		Ersteller: vpa			Seite 1	Version 1	
Id	Bezeichnung	Beschreibung	Wertebereich	# Ex.	Ref.	Syn.	U.Menge von
D001	Sitzungsleiter		String (30)	20	D1		D010
D002	Sitzung	Gruppe von Vorträgen		20	D2, D4	✓	
D003	Konferenz- programm	Menge von Sitzungen		1	D3	✓	
...							

◀ **Objekt-Sammelblatt**

6.1 Beispiel Anforderungsanalyse (14/17)

Anforderungsanalyse
Konzeptueller Entwurf
Logischer Entwurf
Physischer Entwurf

i)

Operations-Glossar (30.12.2004)							
Organisations- einheit: Programm		Ersteller: vpa			Seite 1	Version 1	
Id.	Bez.	Typ	Beschr.	Ref.	Art	Häuf.	Bez. Objekt
Op001	Verschicke_ Call-for- papers			O2		1	D015
...	...						



Vgl. auch Operations-
Sammelblatt

(in Objekt-Sammelblatt)

6.1 Beispiel Anforderungsanalyse (15/17)

Anforderungsanalyse
Konzeptueller Entwurf
Logischer Entwurf
Physischer Entwurf

j)

Ereignis-Glossar (14.12.2004)							
Organisations-einheit: Programm		Ersteller: vpa			Seite 1	Version 1	
Ereignis-Nr.:E001		Referenz: E3					
Bedingungen					Operation		
Id.	Synt. Typ	Beschr.	Sem-antik	Prag-matik	Op.-id.	Beschreibung	Op.-Typ
B001	EL	Programm-festgelegt	when	TRANS	Op 002	Schicke_Zusage _an_Autor	
...							

◀ vgl. Ereignis-Sammelblatt

◀ vgl. Operations-Glossar

Kürzel:

EL: Bedingung ist elementar

TRANS: Beendigung des vorhergehenden Schrittes einer Transaktion

6.1 Beispiel Anforderungsanalyse (16/17)

Anmerkungen

- *Prinzipiell sind verschiedene Vorgehensweisen denkbar*
- *Wichtig dabei:*
 - *Strukturiertes & systematisches Vorgehen*
 - *Kommunikation mit Endanwendern/Kunden*
- *Ergebnis der Anforderungsanalyse sollte die Zusammenhänge **präzise, widerspruchsfrei und vollständig** erfassen*
 - *Zentrale Bedeutung, da hier die zu modellierenden Datenelemente und Abläufe festgelegt werden*
 - *Fehler aus dieser Phase können hohe Kosten verursachen, wenn die späteren Phasen schon begonnen wurden*
 - *Informelle Vorgehensweise erlaubt u.U. nur beschränkt das systematische und automatische Aufdecken von Inkonsistenzen*
- *Erstellte Dokumente dienen als Input für die nächste Phase (konzeptueller Entwurf)*

Anforderungsanalyse

Konzeptueller Entwurf

Logischer Entwurf

Physischer Entwurf

AIFBO

WS05/06

6.1 Beispiel Anforderungsanalyse (17/17)

Anforderungsanalyse
Konzeptueller Entwurf
Logischer Entwurf
Physischer Entwurf

Anmerkungen

- **Werkzeugunterstützung** in der Anforderungsanalyse insb. wichtig für:
 - Systematische Erfassung einer großen Anzahl von Anforderungen (Requirements)
 - Kategorisierung
 - Analyse, z.B. Verfolgung von Abhängigkeiten (Requirements Tracing)
 - Konfigurationsmanagement (Welche Anforderungen gehören zu welcher gewünschten Konfiguration?)
- Viele Werkzeuge diverser Hersteller vorhanden, z.B (nicht vollständig):

Accept Planner	DocuBurst	Objectiver	RequireIT	SpeeDEV
AnalystPro	Doors	OnYourMark	RequisitePro	Team-Trace
Caliber	Focal Point	Open Process	RETH	Truereq
CARE	IRqA	Framework	Rhapsody	Vital-Link
Catalyze	Jalsoft	RDT	ScenarioPlus	Volere
Cradle	Leap SE	Reconcile	Serena RTM	WIBNI
CRW	MKS	Reqtify	Slate	XTie



6 DB-Entwurf

6.0	Vorgehensmodell	2
6.1	Anforderungsanalyse	6
6.2	Konzeptueller Entwurf	26
6.3	Logischer Entwurf.....	30
6.4	Physischer Entwurf	65

6.2 Konzeptueller Entwurf (1/2)

2) Konzeptueller Entwurf

- Ziel: Konzeptuelle Globalsicht der Anwendung, Zielsystem-unabhängiges Datenbankschema
- Eingabe: Anforderungsspezifikation
 - wird modelliert und
 - formalisiert

} z.B. mit Entity-Relationship-Modell
- Strategien dazu:
 - **top-down** (Schrittweise Verfeinerung)
 - **bottom-up** (Schrittweise Verallgemeinerung)

 - **zentralisiert** (globalsicht-orientiert)
 - **dezentralisiert** (einzelsicht-orientiert)

6.2 Konzeptueller Entwurf (2/2)

Anforderungsanalyse

Konzeptueller Entwurf

Logischer Entwurf

Physischer Entwurf

- **Beispiel:**

Einzelansicht-orientiertes Bottom-up-Vorgehen

- *Beschreibungen einzelner Benutzer(gruppen) werden in „Sichten“ überführt, die ein abstraktes Datenmodell benutzen (z.B. ERM)*
- *Integration der einzelnen Sichten in eine konzeptuelle Globalsicht der Datenbank in 2 Schritten:*
 - *Analyse und Behebung der Inkonsistenzen/Redundanzen der einzelnen Sichten*
 - *Durch Misch-Vorgang wird Globalsicht (z.B. als ERM) erstellt, die ggf. Abhängigkeiten der Sichten untereinander berücksichtigt*

6.2 Abschluss des konzeptuellen Entwurfs

- Nach Abschluss des konzeptuellen Entwurfs:
Wahl des Ziel-DBMS

Mögliche Einflussfaktoren:

- Zu verwendendes Datenmodell
 - Verfügbare Datentypen
 - Benchmark-Ergebnisse
 - Vorhandene IT-Landschaft
 - Kosten (für Beschaffung, Wartung, Migration, Personal, etc.)
-
- Ergebnis der Phase:
 - konzeptuelles Modell/Schema (unabhängig vom Ziel-DBMS)
 - Ziel-DBMS ist gewählt

Anforderungsanalyse

Konzeptueller Entwurf

Logischer Entwurf

Physischer Entwurf

AIFB

WS05/06



6 DB-Entwurf

6.0	Vorgehensmodell	2
6.1	Anforderungsanalyse	6
6.2	Konzeptueller Entwurf	26
6.3	Logischer Entwurf.....	30
	Übertragung des ER-Modells ins relationale Datenmodell.....	31
	Sichten	61
6.4	Physischer Entwurf	65

6.3 Logischer Entwurf (1/1)

3) Logischer Entwurf

- Ziel: Transformation des konzeptuellen Modells in das Datenmodell des DBMS, z.B. in das
 - relationale Datenmodell
 - objektorientierte Datenmodell
- Techniken:
 - Transformationsregeln:
z.B. Entity-Relationship-Modell systematisch in das relationale Modell überführen; (Details gleich)
 - Normalisierung
- Ergebnis: logisches Datenmodell bzw. Schema

Anforderungsanalyse

Konzeptueller Entwurf

Logischer Entwurf

Physischer Entwurf

AIFB

WS05/06



6 DB-Entwurf

6.0	Vorgehensmodell	2
6.1	Anforderungsanalyse	6
6.2	Konzeptueller Entwurf	26
6.3	Logischer Entwurf.....	30
	Übertragung des ER-Modells ins relationale Datenmodell.....	32
	Sichten	61
6.4	Physischer Entwurf.....	65

6.3 Übertragung ER-Modell ins relationale Datenmodell

Problem:

Ein Datenschema, das in Form eines ER-Diagramms vorliegt, soll in ein relationales Datenschema übertragen werden.

- ER-Datenschema \Rightarrow relationales Datenschema

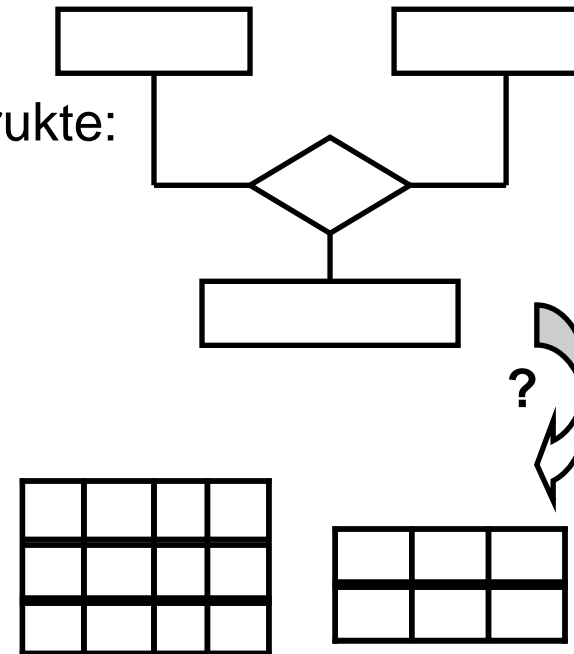
- Im ER-Modell:
zwei verschiedene Modellierungskonstrukte:
Entity-Typen und **Beziehungs-Typen**

- Im relationalen Modell
nur ein Konstrukt: **Relation**

Abbildung/Transformation notwendig:

Entity-Typen, Beziehungstypen \rightarrow Relationsschemata, Fremdschlüssel

Anforderungsanalyse
Konzeptueller Entwurf
Logischer Entwurf
Physischer Entwurf



6.3 Übertragung ER-Modell ins relationale Datenmodell

Vereinbarung für Schreibweise (Wdh.):

Es seien: A, B Attributmengen, a_1, a_2, \dots Attribute

Dann schreiben wir:

$a_1 a_2 \dots a_n$ für $\{a_1, a_2, \dots, a_n\}$

AB für $A \cup B$

aB für $\{a\} \cup B$

usw.



6 DB-Entwurf

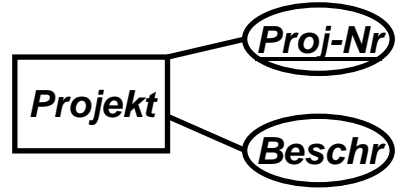
6.0	Vorgehensmodell	2
6.1	Anforderungsanalyse	6
6.2	Konzeptueller Entwurf	26
6.3	Logischer Entwurf.....	30
	Übertragung des ER-Modells ins relationale Datenmodell.....	32
	Übertragung von Entity-Typen	36
	Übertragung von Generalisierungen.....	40
	Übertragung von Beziehungs-Typen.....	47
	Sichten	61
6.4	Physischer Entwurf.....	65

Anforderungsanalyse
Konzeptueller Entwurf
Logischer Entwurf
Physischer Entwurf

6.3 Übertragung von Entity-Typen (1|4)

- Entity-Typ $E : \langle A \rangle$ mit
 - Primärschlüssel $P \subseteq A$
 - **Relationenschema** $r_E: R_E$ mit
 - $R_E = (A \mid \text{Primärschlüssel } P)$
 - (gilt nur, wenn Entity-Typ weder Weak Entity-Typ noch Unterentity-Typ ist !)

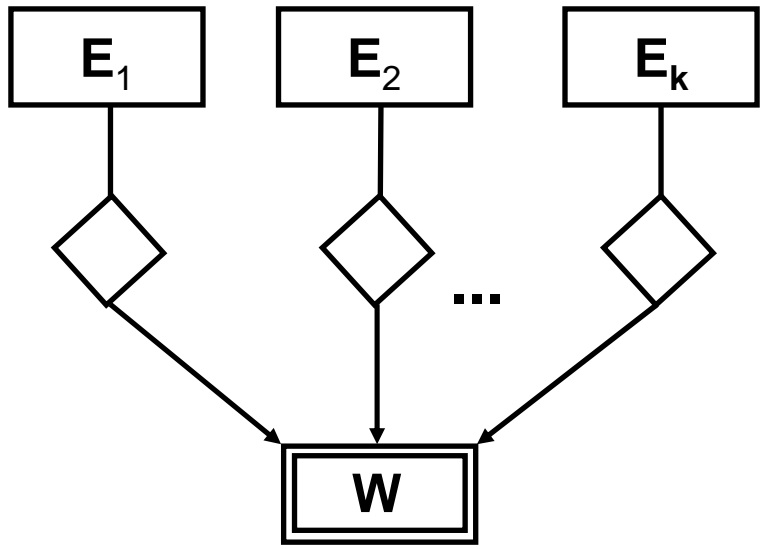
Beispiel:



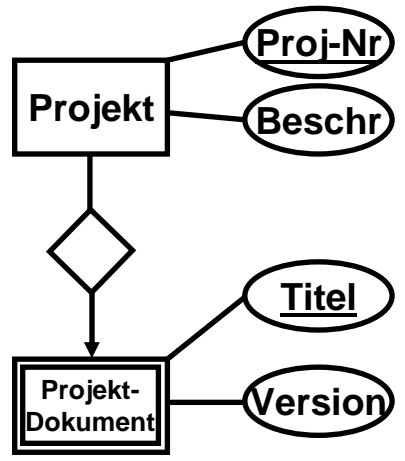
Projekt → *projekt*(*Proj-Nr*, *Beschr*)

6.3 Übertragung von Entity-Typen (2|4)

- **Weak Entity-Typ $W:\langle A \rangle$** , mit
 - Schlüsselattributen $S \subseteq A$
 - Identifikationsabhängigkeit von Entity-Typen $E_1:\langle A_1 \rangle, \dots, E_k:\langle A_k \rangle$ mit den jeweiligen Primärschlüsseln P_1, \dots, P_k



Beispiel: Ein best. WE nur mit dem übergeordneten Entity identifizierbar; hat keinen selbständigen Schlüssel



6.3 Übertragung von Entity-Typen (3|4)

$W \rightarrow$

Relationenschema $r_W: R_W$ mit

$$R_W = (A P_1 \dots P_k \mid \Sigma_W),$$

– $\Sigma_W =$
 „ $S P_1 \dots P_k$ ist der Primärschlüssel “

– Fremdschlüssel:

$$r_W.P_1 \subseteq r_1.P_1,$$

... ,

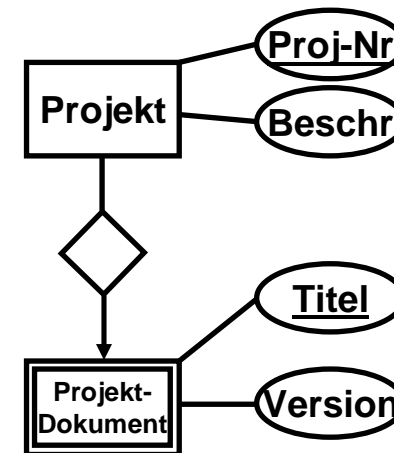
$$r_W.P_k \subseteq r_1.P_k$$

Beispiel (fortgesetzt):

Projekt \rightarrow $projekt(\underline{Proj-Nr}, Beschr)$

Projekt-Dokument \rightarrow $pdokument(\underline{Titel}, \underline{Proj-Nr}, Version),$
 $pdokument.Proj-Nr \subseteq projekt.Proj-Nr$

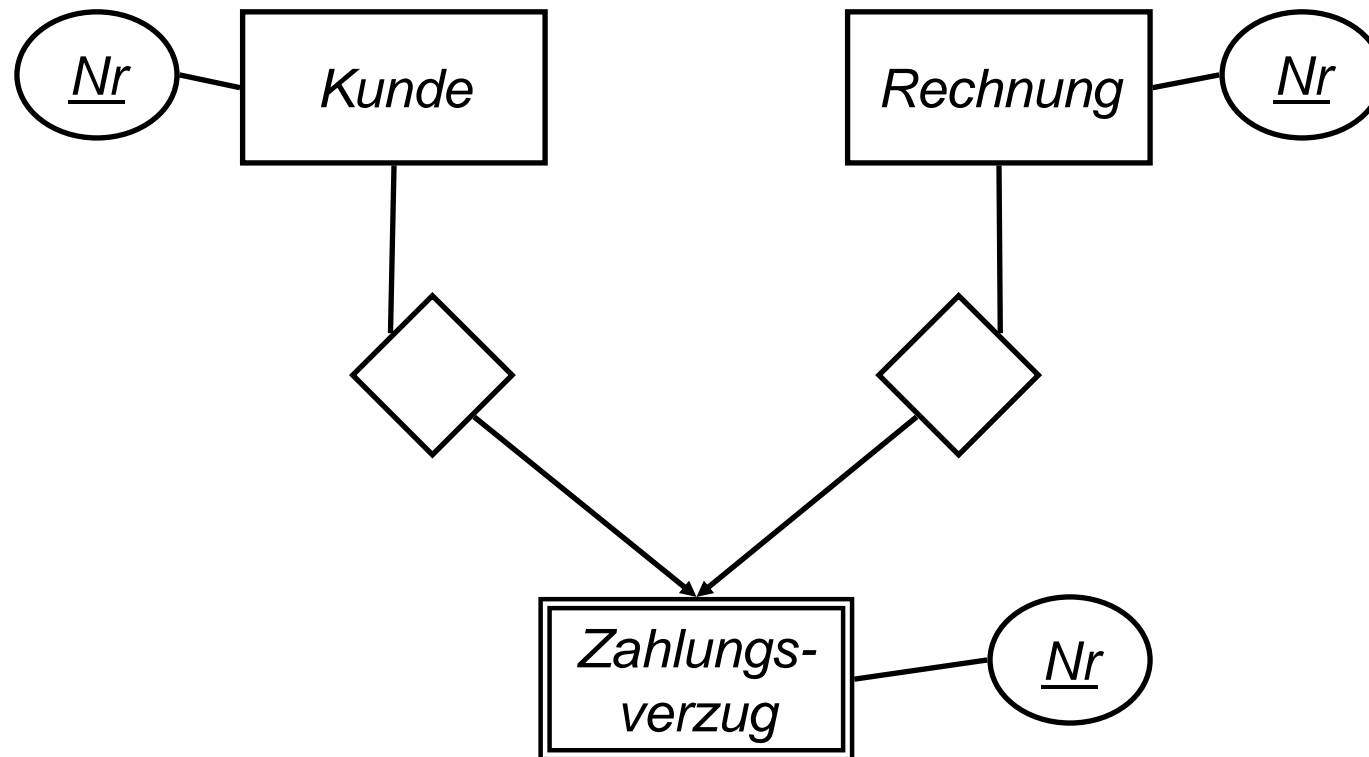
Anforderungsanalyse
Konzeptueller Entwurf
Logischer Entwurf
Physischer Entwurf



6.3 Übertragung von Entity-Typen (4|4)

Anforderungsanalyse
Konzeptueller Entwurf
Logischer Entwurf
Physischer Entwurf

Beispiel:



Zahlungsverzug → $zverzug: (\underline{KdNr}, \underline{ReNr}, \underline{Nr}, \dots)$,
 $zverzug.KdNr \subseteq kunde.Nr$, $zverzug.ReNr \subseteq rechnung.Nr$



6 DB-Entwurf

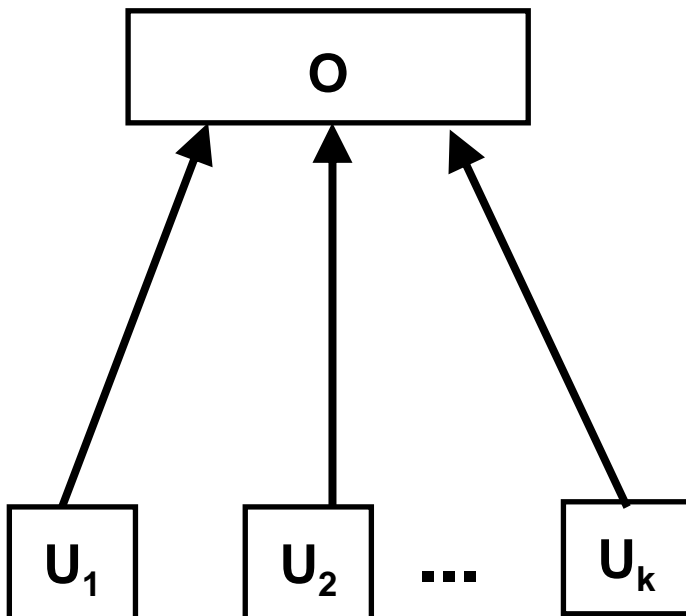
6.0	Vorgehensmodell	2
6.1	Anforderungsanalyse	6
6.2	Konzeptueller Entwurf	26
6.3	Logischer Entwurf.....	30
	Übertragung des ER-Modells ins relationale Datenmodell.....	32
	Übertragung von Entity-Typen	36
	Übertragung von Generalisierungen.....	40
	Übertragung von Beziehungs-Typen.....	47
	Sichten	61
6.4	Physischer Entwurf.....	65

6.3 Übertragung von Generalisierungen (1|4)

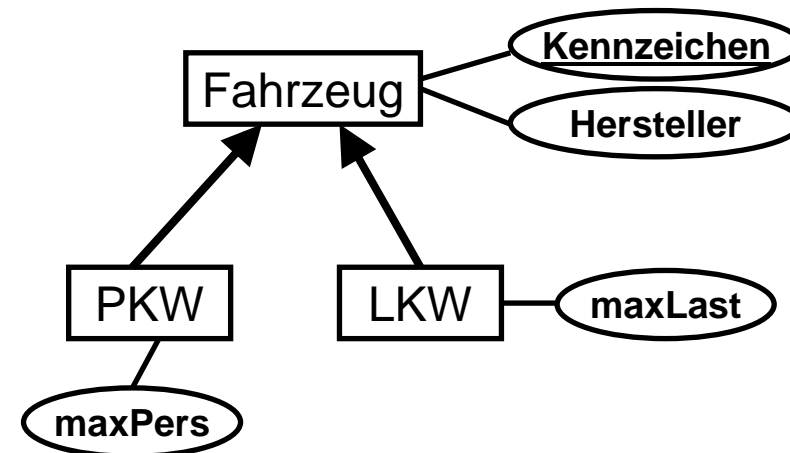
Wir betrachten nur **Einfachgeneralisierungen**

Unterentity-Typen: $U_1:\langle A_1 \rangle, U_2:\langle A_2 \rangle, \dots, U_k:\langle A_k \rangle$

Oberentity-Typ: $O:\langle A \rangle$ mit Primärschlüssel $P \subseteq A$



Beispiel:



6.3 Übertragung von Generalisierungen (2|4)

Vier Alternativen:

1. Oberentity-Typ O

→ **Relationsschema** $r_O:R_O$ mit

$R_O = (A \mid \text{Primärschlüssel } P)$ (wie zuvor).

Jeder **Unterentity-Typ** U_i

→ **Relationsschema** $r_i:R_i$ mit

$R_i = (A_i \mid \text{Primärschlüssel } P)$

– Fremdschlüssel: $r_i.P \subseteq r_O.P$

Beispiel: Fahrzeug → *fahrzeug(Kennzeichen, Hersteller)*

PKW → *pkw(Kennzeichen, maxPers),*
pkw.Kennzeichen \subseteq *fahrzeug.Kennzeichen*

LKW → *lkw(Kennzeichen, maxLast),*
lkw.Kennzeichen \subseteq *fahrzeug.Kennzeichen*

Anforderungsanalyse

Konzeptueller Entwurf

Logischer Entwurf

Physischer Entwurf

AIFBO

WS05/06

Anforderungsanalyse
Konzeptueller Entwurf
Logischer Entwurf
Physischer Entwurf

6.3 Übertragung von Generalisierungen (3|4)

- 2. Für den **Oberentity**-Typ wird kein eigenes Schema erstellt:
 Jeder **Unterentity**-Typ U_i
 → **Relationsschema** $r_i:R_i$ mit
 $R_i = (A A_i | \text{Primärschlüssel } P)$

! nur möglich, wenn U_1, \dots, U_k vollst. Spezialisierung von O

Beispiel: (Annahme, es gibt nur PKW und LKW als Fahrz.)

- PKW* → *pkw(Kennzeichen, maxPers, Hersteller),*
- LKW* → *lkw(Kennzeichen, maxLast, Hersteller),*

Anforderungsanalyse
Konzeptueller Entwurf
Logischer Entwurf
Physischer Entwurf

6.3 Übertragung von Generalisierungen (3|4)

3. Für $\mathbf{O}, \mathbf{U}_1, \mathbf{U}_2, \dots, \mathbf{U}_k$
 wird ein einziges **Relationenschema $r: R$** erstellt mit

$$R = (A \ A_1 \ \dots \ A_k \ t \mid \text{Primärschlüssel } P)$$

- t ist ein zusätzliches Attribut, zur Angabe des **Untertyps**:
mögliche Werte 1, ..., k

! Nullwerte notwendig

Beispiel:

*Fahrzeug, PKW, LKW →
 fahrzeug(Kennzeichen, Hersteller, maxPers, maxLast, t)*

Anforderungsanalyse
Konzeptueller Entwurf
Logischer Entwurf
Physischer Entwurf

6.3 Übertragung von Generalisierungen (4|4)

4. Für O, U_1, U_2, \dots, U_k
wird ein einziges **Relationenschema** $r: R$ erstellt mit
 $R = (A A_1 \dots A_k t_1, t_2, \dots, t_k \mid \text{Primärschlüssel } P)$
- t_1, t_2, \dots, t_k sind **zusätzliche Attribute** vom Typ Boolean zur Angabe ob ein Entity zu einem bestimmten **Untertyp** gehört (notwendig wenn **U1, U2, ..., Uk** nicht disjunkt sind)
- ! Nullwerte notwendig

Beispiel:

*Fahrzeug, PKW, LKW →
fahrzeug(Kennzeichen, Hersteller, maxPers, maxLast, t1, t2)*

6.3 Übertragung von Generalisierungen (4|4)

Anmerkung:

Alternative (1) ist immer verwendbar und sollte, sofern keine sonstigen Gründe dagegen sprechen, nach Möglichkeit verwendet werden!

Anforderungsanalyse

Konzeptueller Entwurf

Logischer Entwurf

Physischer Entwurf

AIFB

WS05/06



6 DB-Entwurf

6.0	Vorgehensmodell	2
6.1	Anforderungsanalyse	6
6.2	Konzeptueller Entwurf	26
6.3	Logischer Entwurf.....	30
	Übertragung des ER-Modells ins relationale Datenmodell.....	32
	Übertragung von Entity-Typen	36
	Übertragung von Generalisierungen.....	40
	Übertragung von Beziehungs-Typen.....	47
	Sichten	61
6.4	Physischer Entwurf.....	65

6.3 Übertragung von Beziehungstypen (1|13)

1:1 - Beziehungen

- Ein 1:1-Relationship-Typ wird i.A. nicht zu einer eigenen Relation.
- Information wird an eine der beiden (den betroffenen Entity-Typen entsprechenden) Relationen "angehängt".

Anforderungsanalyse

Konzeptueller Entwurf

Logischer Entwurf

Physischer Entwurf

AIFBO

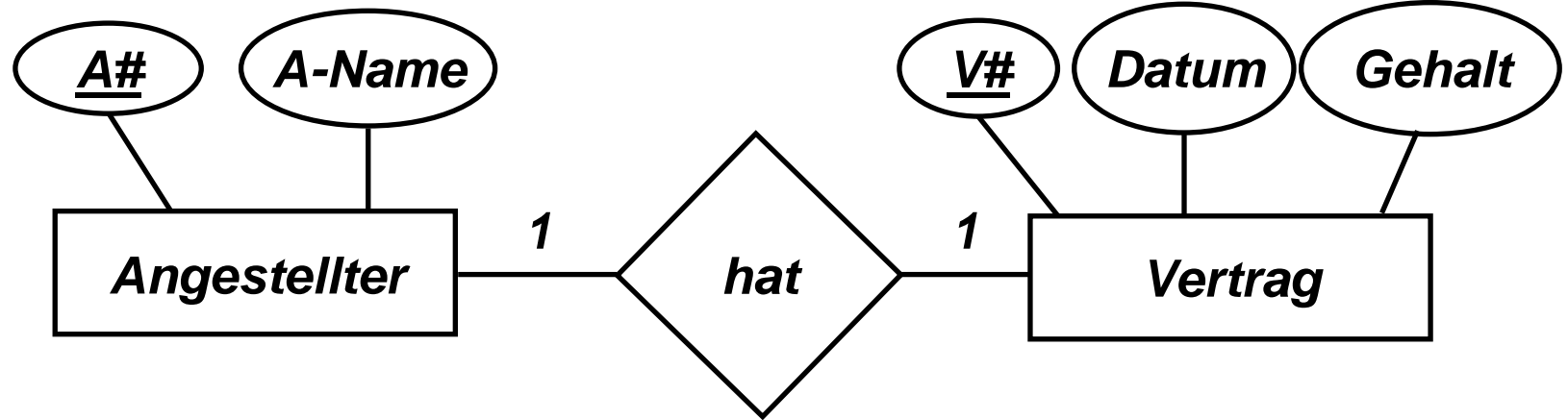
WS05/06

Anforderungsanalyse
Konzeptueller Entwurf
Logischer Entwurf
Physischer Entwurf

6.3 Übertragung von Beziehungstypen (2|13)

1:1 - Beziehungen

Beispiel:



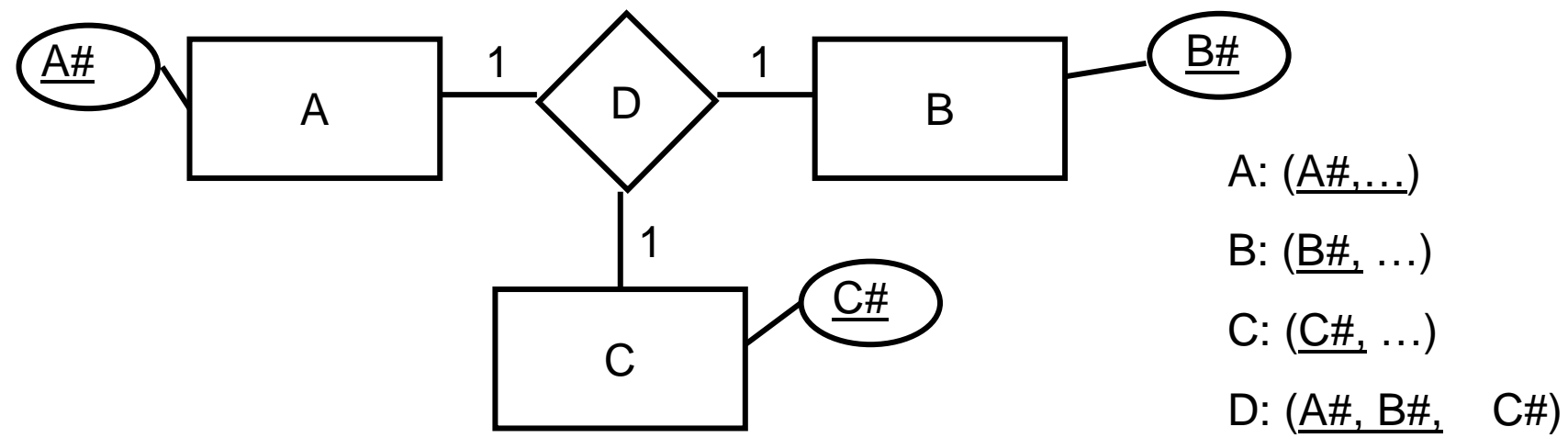
→ *angestellter*(A#, A-Name, V#)
vertrag(V#, Datum, Gehalt)

oder:
angestellter(A#, A-Name)
vertrag(V#, Datum, Gehalt, A#)

6.3 Übertragung von Beziehungstypen (3|13)

1:1:1 - Beziehungen

Aber: Geht das so auch bei 1:1:1-Beziehungen oder muss D eine eigenständige Relation werden?



Antwort: D muss eigenständige Relation werden
 Begründung anhand eines Beispiels für D:

A#	B#	C#
a	b	c
a	b'	c'
a	b''	c''

Notation besagt ja, dass eine Attributkombination a, b für ein c in der Spalte C# nur einmal vorkommen darf, jedoch kann ein Attribut a mehrmals in der Spalte A# vorkommen => Problem, wenn man in Relation A nur 2 Fremdschlüsselattribute genommen hätte, statt Relation D.

6.3 Übertragung von Beziehungstypen (4|13)

Anforderungsanalyse

Konzeptueller Entwurf

Logischer Entwurf

Physischer Entwurf

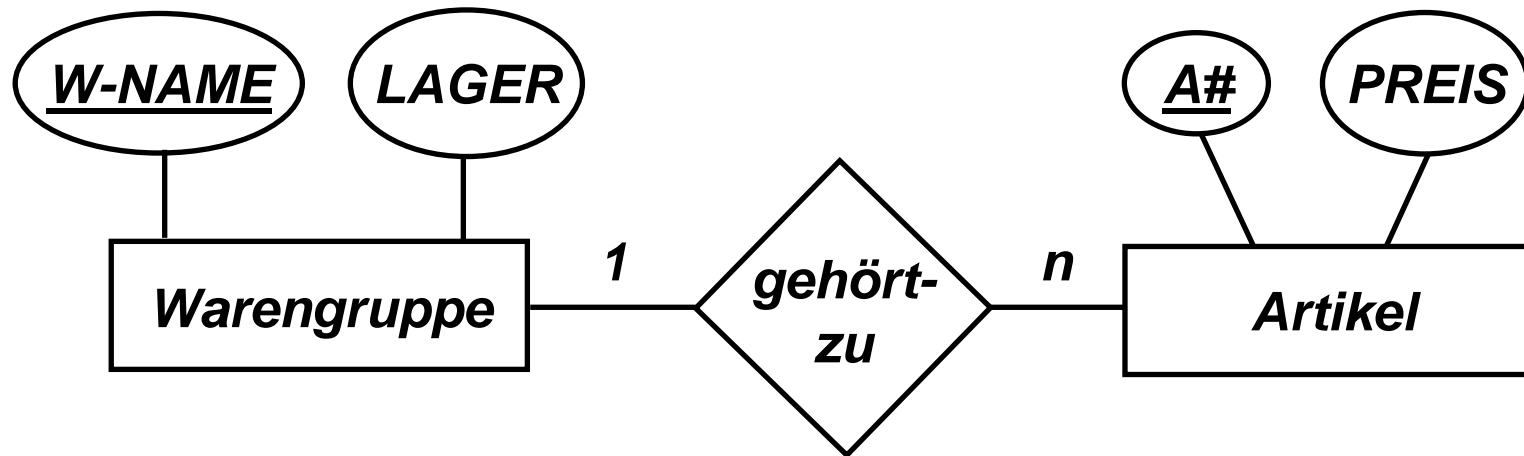
1:n - Beziehungen

- Ein 1:n-Relationship-Typ wird i.A. nicht zu einer eigenen Relation.
- Information wird an die Relation "angehängt", die dem Entity-Typ an der mit **n** beschrifteten Kante entspricht (z.B. Fremdschlüssel oder Attribute des Relationship-Typs).
- ggf. NULL-Werte notwendig

6.3 Übertragung von Beziehungstypen (5|13)

1:n – Beziehungen

Beispiel: „Artikel gehört zu Warengruppe“



→ *artikel*(A#, Preis, *W-Name*)
warengruppe(W-Name, Lager)

Anforderungsanalyse
Konzeptueller Entwurf
Logischer Entwurf
Physischer Entwurf

6.3 Übertragung von Beziehungstypen (6|13)

Anforderungsanalyse

Konzeptueller Entwurf

Logischer Entwurf

Physischer Entwurf

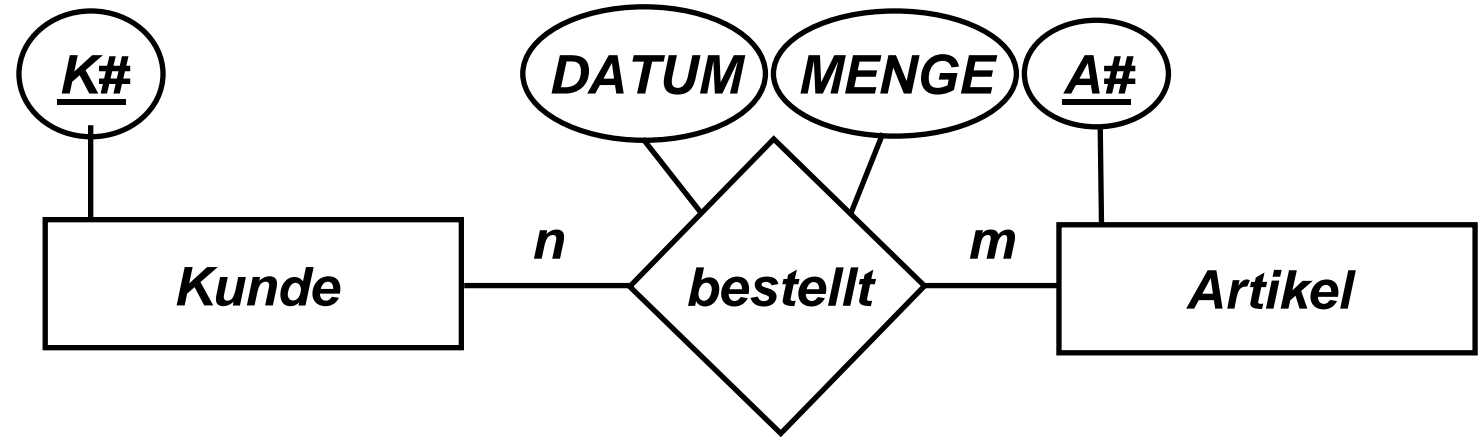
n:m - Beziehungen

- Aus n:m-Relationship-Typ wird eine zusätzliche Relation.
- Relation enthält die (Primär-)Schlüssel X und Y der beteiligten Entity-Typen als Attribute (Fremdschlüsselbeziehung) und zusätzlich die Attribute des Relationship-Typs.
- Neuer Schlüssel ergibt sich aus Vereinigung $X \cup Y$

6.3 Übertragung von Beziehungstypen (7|13)

n:m - Beziehungen

Beispiel:

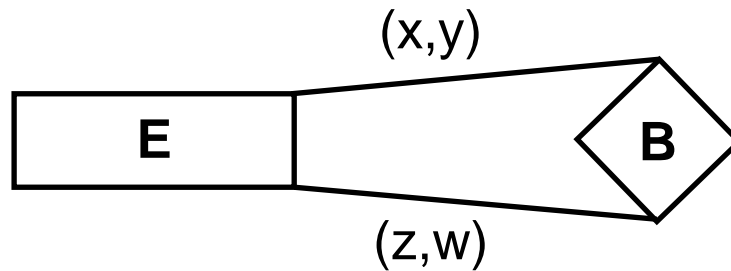


→ *bestellt*(K#, A#, Datum, Menge)

bestellt:	<u>K#</u>	<u>A#</u>	DATUM	MENGE

6.3 Übertragung von Beziehungstypen (8|13)

Rekursive Beziehungen



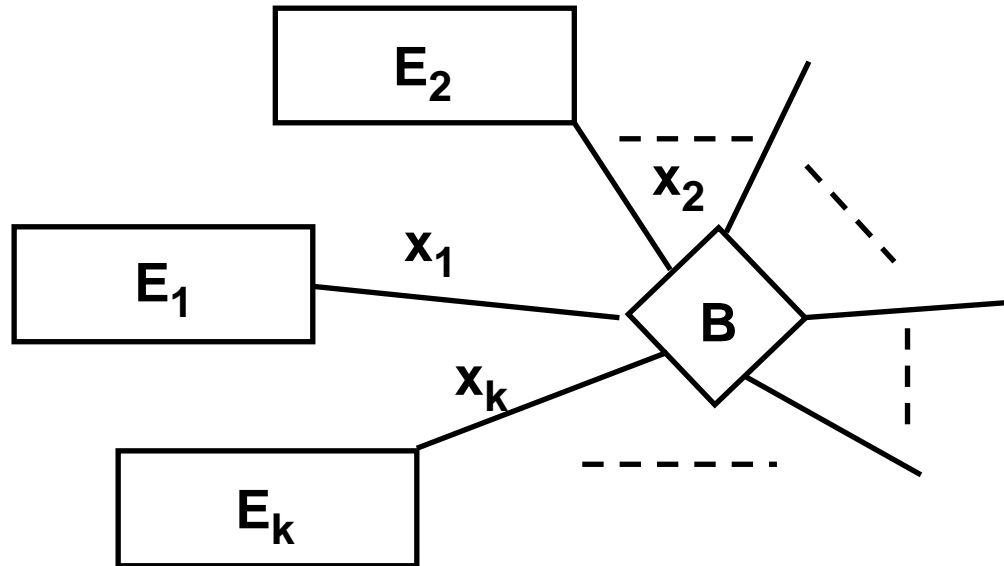
analog zu normalen 1:1 -, 1:n - und n:m - Beziehungen

(die Namen hinzugefügter Attribute werden um den Rollennamen ergänzt)

Anforderungsanalyse
Konzeptueller Entwurf
Logischer Entwurf
Physischer Entwurf

6.3 Übertragung von Beziehungstypen (9|13)

Beziehungen vom Grad > 2



B wird eigenes Relationenschema $r_B: R_B$:

B: $\langle E_1, \dots, E_k / A_z \rangle$,

Relationen für die Entity-Typen:

$r_1: R_1$ mit $R_1 = (A_1 | \Sigma_1)$, (P_1 ist Pr.schlüssel) $\in \Sigma_1$, ...,

$r_k: R_k$ mit $R_k = (A_k | \Sigma_k)$, (P_k ist Pr.schlüssel) $\in \Sigma_k$

→

Anforderungsanalyse

Konzeptueller Entwurf

Logischer Entwurf

Physischer Entwurf

AIFBO

WS05/06

Anforderungsanalyse
Konzeptueller Entwurf
Logischer Entwurf
Physischer Entwurf

6.3 Übertragung von Beziehungstypen (10|13)

Beziehungen vom Grad > 2

Relation für die Beziehung:

$r_B : R_B$,

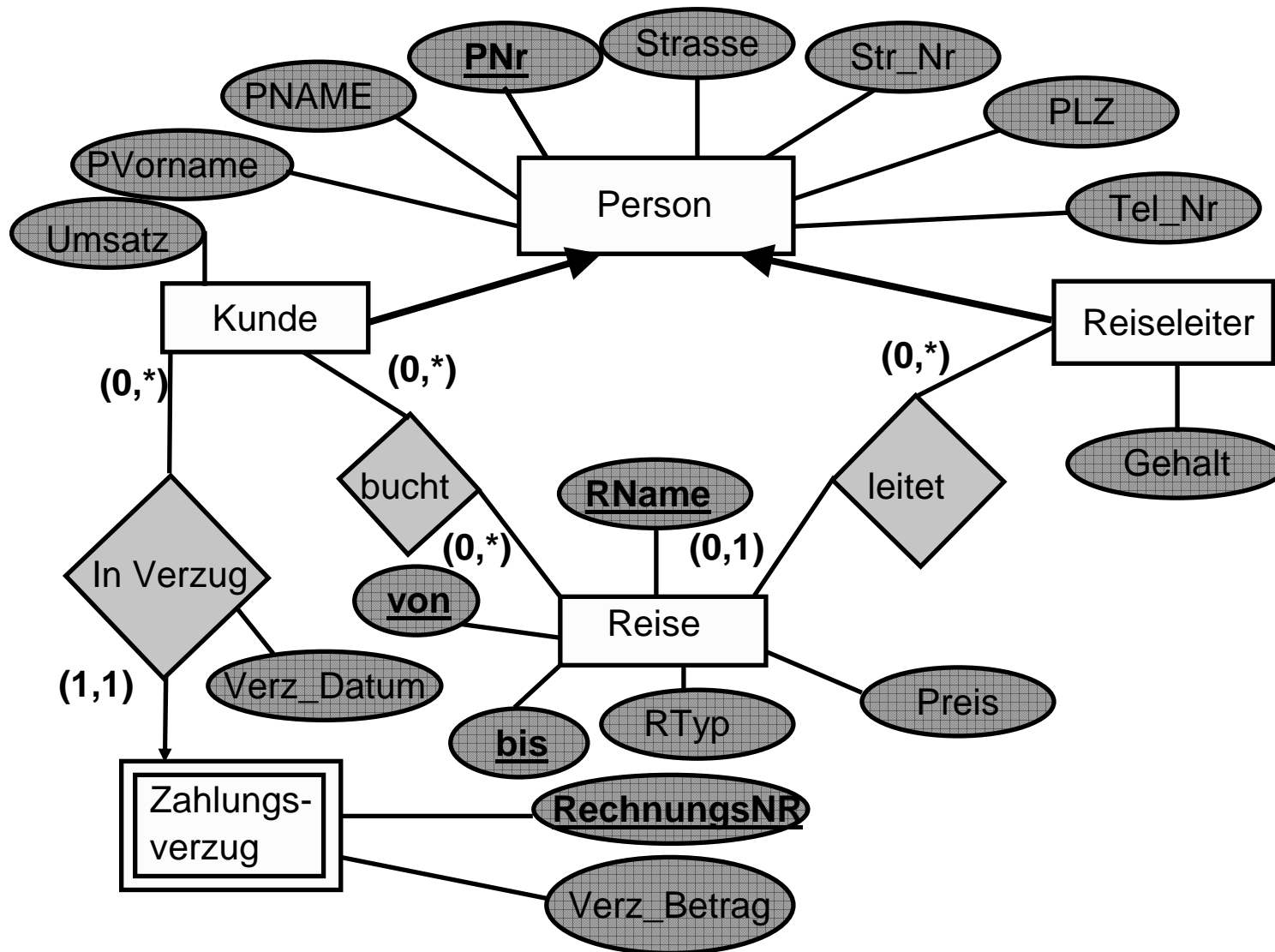
- $r_B : R_B$ mit $R_B = (P_1 \dots P_k A_z | \Sigma_B)$,
- Fremdschlüssel:
 $r_B.P_1 \subseteq r_1.P_1, \dots, r_B.P_k \subseteq r_k.P_k$
- Falls 1 die Mindestkardinalität von E_j bzgl. B ist
zusätzlich $r_j.P_j \subseteq r_B.P_j$

Σ_B : Schlüssel ergibt sich aus der Vereinigung der Primärschlüssel P_j der Relationsschemata, für welche die entsprechenden Entity-Typen (gemeinsam) einen Schlüssel von B darstellen

Anforderungsanalyse
Konzeptueller Entwurf
Logischer Entwurf
Physischer Entwurf

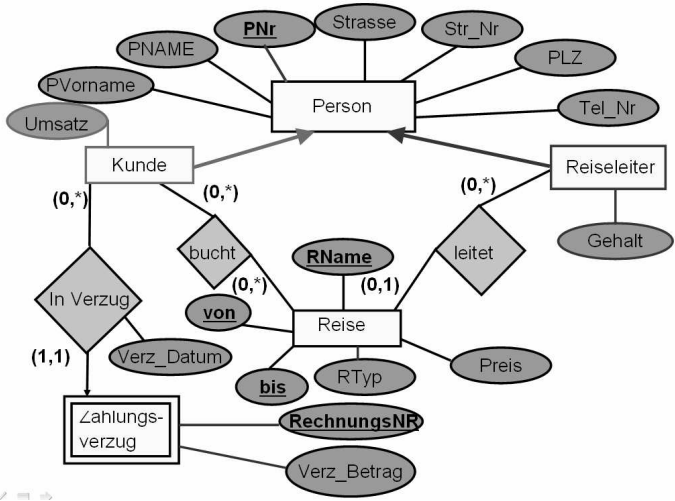
6.3 Übertragung von Beziehungstypen (11|13)

Beispiel: Datenmodell eines Reiseveranstalters



Anforderungsanalyse
Konzeptueller Entwurf
Logischer Entwurf
Physischer Entwurf

6.3 Übertragung von Beziehungstypen (12|13)



„Normale“ Entity-Typen:

- **Person** → person: (PNr, PVorname, PName, Strasse, Str_Nr, PLZ, Tel_Nr)
- **Reise** → reise: (RName, von, bis, Preis, RTyp)

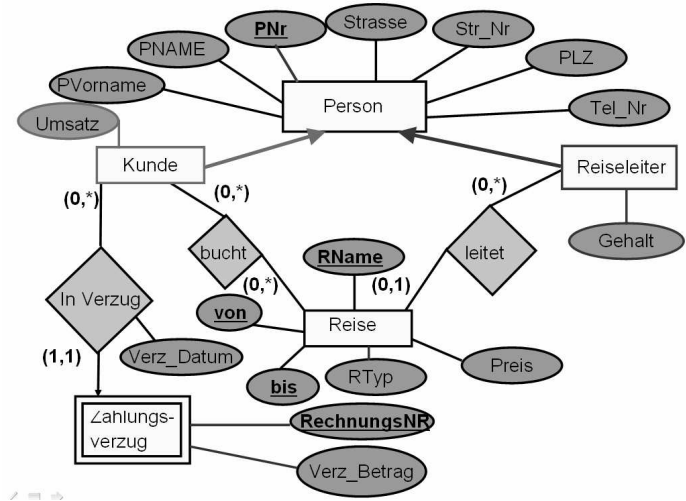
Weak Entity-Typen und Unterentity-Typen:

- **Kunde** → kunde: (PNr, Umsatz),
kunde.PNr ⊆ person.PNr
- **Reiseleiter** → rleiter: (PNr, Gehalt),
rleiter.PNr ⊆ person.PNr
- **Zahlungsverzug** → zverzug: (PNr, RechnungsNr,
Verz_Datum, Verz_Betrag),
zverzug.PNr ⊆ kunde.PNr

NULL-Werte für Fremdschlüssel sind nicht erlaubt.

Anforderungsanalyse
Konzeptueller Entwurf
Logischer Entwurf
Physischer Entwurf

6.3 Übertragung von Beziehungstypen (13|13)



1:n- Beziehungen

- „in Verzug“ wurde schon durch den **Weak Entity-Typ** zverzug realisiert
- „leitet“: Ergänzen von **Reise** um die Personenummer des Reiseleiters
 → reise: (RName, von, bis, Preis, RTyp, PNr)
 reise.PNr \subseteq rleiter.PNr,
 NULL-Werte sind erlaubt

m:n-Beziehungen

- „bucht“ → buchung: (PNr, RName, von, bis),
 buchung.PNr \subseteq kunde.PNr
 buchung.{RName, von, bis} \subseteq reise.{RName, von, bis}



6 DB-Entwurf

6.0	Vorgehensmodell	2
6.1	Anforderungsanalyse	6
6.2	Konzeptueller Entwurf	26
6.3	Logischer Entwurf.....	30
	Übertragung des ER-Modells ins relationale Datenmodell.....	32
	Übertragung von Entity-Typen	36
	Übertragung von Generalisierungen.....	40
	Übertragung von Beziehungs-Typen.....	47
	Sichten	61
6.4	Physischer Entwurf.....	65

6.3 Sichten (Views) (1/3)

Sichten

Externe Ebene

- Für Anwender / Benutzer:
„Sichten“ („Views“) über Basisrelationen
- nicht „materialisiert“,
d.h. sie werden bei jedem Aufruf neu ausgewertet
- Definition einer Sicht durch Ausdruck der Art
sichtname: Operationsteil
↑
Rel'operationen auf Basisrelationen;
„Anfrage“ (engl. query) an DB
- In der Praxis: Sichtendefinition mit SQL („CREATE VIEW...“)
- Sicherheitsaspekt: Sichten sollen bestimmten Benutzergruppen
nur bestimmte Daten anzeigen

6.3 Sichten (Views) (2/3)

Anforderungsanalyse

Konzeptueller Entwurf

Logischer Entwurf

Physischer Entwurf

AIFBO

WS05/06

Beispiel:

Sicht (in relationaler Algebra):

$abt-30: \pi_{[ANG-NR, NAME, WOHNORT]}(\sigma_{[ABT-NR=30]} \text{angestellte})$

↑

*Operationsteil hier: Ausdruck der **Relationenalgebra***

Danach:

- Behandlung von abt-30 wie „normale“ Relation (Bezüglich Anwendung von Operationen)
- hier auch möglich:
Update / Insert / Delete mit abt-30 mit Auswirkungen auf angestellte !
(ggf. mit NULL ergänzen)

6.3 Sichten (Views) (3/3)

Anforderungsanalyse

Konzeptueller Entwurf

Logischer Entwurf

Physischer Entwurf

AIFBO

WS05/06

Beispiel:

Sicht

proj-spezial:

$\pi_{[ANG-NR,NAME,WOHNORT]}\sigma_{[P-NR=770114]}$ (**angestellte * ang-pro**)

z.B.: Sicht der Leitung von Projekt 770114

INSERT / UPDATE / DELETE auf proj-spezial

- nicht erlaubt
- mehrere Basisrelationen betroffen /
Wirkung nicht überschaubar



6 DB-Entwurf

6.0	Vorgehensmodell	2
6.1	Anforderungsanalyse	6
6.2	Konzeptueller Entwurf	26
6.3	Logischer Entwurf.....	30
6.4	Physischer Entwurf.....	65

6.4 Physischer Entwurf (1/11)

4) Physischer Entwurf

- Ziel: Definition des internen Schemas des DBMS, sowie der Systemparameter
- Dabei wichtig: Minimierung der Zugriffszeiten auf Sekundärspeicher
- Zu berücksichtigende Aspekte:
 - Genaue Datentypen der Attribute
 - Block- bzw. Seitenzuweisung auf Festplatte
 - Indexstrukturen für den effizienten Datenzugriff
 - Evtl. Denormalisierung, um bei häufig zu verbindenden Relationen Zugriffe/Rechenzeit zu sparen
- Implementierung der Schemata unter Verwendung von DDL des gewählten Systems
 - z.B. SQL, grafische Dialoge, etc.
- Evtl. entwurfsbegleitend: Prototyping mit Testdaten-Bestand
- Ergebnis der Phase: Physisches Modell bzw. Schema der Datenbank

6.4 Physischer Entwurf (2/11)

Relationale Datenbank

- **Typ-Ebene:** „Schema der relationalen DB“
besteht aus:
 - Menge von Relationsschemata $\{r_i: (A_i | \Sigma_i) \mid i=1, \dots, n\}$
(mit intrarelationalen Integritätsbedingungen Σ_i)
 - relationenübergreifende (interrelationale)
Integritätsbedingungen: Σ_{DB}
- **Ausprägungsebene:**
 - Zu jedem Relationsschema **eine** (1NF-) Relation vom entsprechenden Typ.
 - Alle Relationen erfüllen in ihrer Gesamtheit alle Integritätsbedingungen aus Σ_{DB} .

6.4 Physischer Entwurf (3/11)

Beispiel

Input aus dem logischen Entwurf:

- person: (PNr, PVname, PNname, Strasse, Str_Nr, PLZ, Tel_Nr)
- rleiter: (PNr, Gehalt),
rleiter.PNr \subseteq person.PNr
- kunde: (PNr, Umsatz),
kunde.PNr \subseteq person.PNr
- reise: (RName, von, bis, Preis, RTyp, PNr)
reise.PNr \subseteq rleiter.PNr
- zverzug: (PNr, RechnungsNr, Verz_Datum, Verz_Betrag),
zverzug.PNr \subseteq kunde.PNr
- buchung: (PNr, RName, von, bis),
buchung.PNr \subseteq kunde.PNr
buchung.{RName, von, bis} \subseteq reise.{RName, von, bis}

6.4 Physischer Entwurf (4/11)

Beispiel

Input aus dem logischen Entwurf:

- person: (PNR, PVname, PNname, Strasse, Str_Nr, PLZ, Tel_Nr)

Beispiel zur graphischen Eingabe in MS ACCESS

The screenshot shows the 'person : Tabelle' window in MS Access. The top part is a table with columns: Feldname, Felddatentyp, and Beschreibung. The bottom part is the 'Feldeigenschaften' (Field Properties) window, which is currently showing the 'Allgemein' (General) tab. The 'PLZ' field is selected in the table, and its properties are shown in the 'Feldeigenschaften' window.

Feldname	Felddatentyp	Beschreibung
PNR	AutoWert	Primärschlüssel
PVname	Text	Vorname
PNname	Text	Nachname
Strasse	Text	
Str_Nr	Zahl	
PLZ	Zahl	Postleitzahl
Tel_Nr	Text	+, (), /, - erlaubt

Feldeigenschaften	
Allgemein	Nachschlagen
Feldgröße	Dezimal
Format	
Genauigkeit	5
Dezimalstellen	0
Dezimalstellenanzeige	0
Eingabeformat	
Beschriftung	
Standardwert	0
Gültigkeitsregel	
Gültigkeitsmeldung	
Eingabe erforderlich	Nein
Indiziert	Nein
Smarttags	

Ein Feldname kann bis zu 64 Zeichen lang sein, einschließlich Leerzeichen. Drücken Sie F1, um Hilfe zu Feldnamen zu erhalten.

Feldeigenschaften

6.4 Physischer Entwurf (5/11)

Beispiel

Input aus dem logischen Entwurf:

- rleiter: (PNr, Gehalt),

Feldname	Felddatentyp	Beschreibung
PNr	Zahl	
Gehalt	Währung	

Feldeigenschaften

Allgemein | Nachschlagen

Format	Euro
Dezimalstellenanzeige	Automatisch
Eingabeformat	
Beschriftung	
Standardwert	0
Gültigkeitsregel	
Gültigkeitsmeldung	
Eingabe erforderlich	Nein
Indiziert	Nein
Smarttags	

Auch möglich:

Definition mit SQL-DDL:

```
Create table rleiter (  
    PNR Integer not null, } A  
    Gehalt Currency, }  
    CONSTRAINT C1 PRIMARY } Σ  
    KEY (PNR)  
);
```

Ähnlich unserer Notation $r: (A | \Sigma)$

Anforderungsanalyse

Konzeptueller Entwurf

Logischer Entwurf

Physischer Entwurf

AIFBO

WS05/06

6.4 Physischer Entwurf (6/11)

Beispiel

Input aus dem logischen Entwurf:

- kunde: (PNr, Umsatz)

The screenshot shows a window titled "kunde : Tabelle" with a table structure and a properties panel. The table has the following fields:

Feldname	Felldatentyp	Beschreibung
PNr	Zahl	
Umsatz	Währung	

The "Umsatz" field is selected, and its properties are shown in the "Feldeigenschaften" panel:

Feldeigenschaften	
Allgemein	Nachschlagen
Format	Euro
Dezimalstellenanzeige	2
Eingabeformat	
Beschriftung	
Standardwert	0
Gültigkeitsregel	
Gültigkeitsmeldung	
Eingabe erforderlich	Nein
Indiziert	Nein
Smarttags	

Anforderungsanalyse

Konzeptueller Entwurf

Logischer Entwurf

Physischer Entwurf

AIFBO

WS05/06

6.4 Physischer Entwurf (7/11)

Beispiel

Input aus dem logischen Entwurf:

- reise: (RName, von, bis, Preis, RTyp, PNr)

Feldname	Felddatentyp	Beschreibung
RName	Text	
von	Datum/Uhrzeit	
bis	Datum/Uhrzeit	
Preis	Währung	
RTyp	Text	
PNr	Zahl	

Feldeigenschaften

Allgemein | Nachschlagen

Format	Datum, kurz
Eingabeformat	
Beschriftung	
Standardwert	
Gültigkeitsregel	
Gültigkeitsmeldung	
Eingabe erforderlich	Nein
Indiziert	Nein
IME-Modus	Keine Kontrolle
IME-Satzmodus	Keine
Smarttags	

Anforderungsanalyse

Konzeptueller Entwurf

Logischer Entwurf

Physischer Entwurf

AIFBO

WS05/06

6.4 Physischer Entwurf (8/11)

Beispiel

Input aus dem logischen Entwurf:

- zverzug: (PNr, RechnungsNr, Verz_Datum, Verz_Betrag)

The screenshot shows a window titled "zverzug : Tabelle" with a table of field definitions and a properties panel below it.

	Feldname	Felddatentyp	Beschreibung
<input type="checkbox"/>	PNr	Zahl	
<input type="checkbox"/>	RechnungsNr	Zahl	
<input type="checkbox"/>	Verz_Datum	Datum/Uhrzeit	
<input checked="" type="checkbox"/>	Verz_Betrag	Währung	

Feldeigenschaften

Allgemein Nachschlagen

Format	Euro
Dezimalstellenanzeige	Automatisch
Eingabeformat	
Beschriftung	
Standardwert	0
Gültigkeitsregel	
Gültigkeitsmeldung	
Eingabe erforderlich	Nein
Indiziert	Nein
Smarttags	

6.4 Physischer Entwurf (9/11)

Beispiel

Input aus dem logischen Entwurf:

- buchung: (PNr, RName, von, bis)

Feldname	Felddatentyp	Beschreibung
PNr	Zahl	
RName	Text	
von	Datum/Uhrzeit	
bis	Datum/Uhrzeit	

Feldeigenschaften

Allgemein | Nachschlagen

Feldgröße	50
Format	
Eingabeformat	
Beschriftung	
Standardwert	
Gültigkeitsregel	
Gültigkeitsmeldung	
Eingabe erforderlich	Nein
Leere Zeichenfolge	Ja
Indiziert	Ja (Duplikate möglich)
Unicode-Kompression	Ja
IME-Modus	Keine Kontrolle
IME-Satzmodus	Keine
Smarttags	

Anforderungsanalyse

Konzeptueller Entwurf

Logischer Entwurf

Physischer Entwurf

AIFBO

WS05/06

Anforderungsanalyse
Konzeptueller Entwurf
Logischer Entwurf
Physischer Entwurf

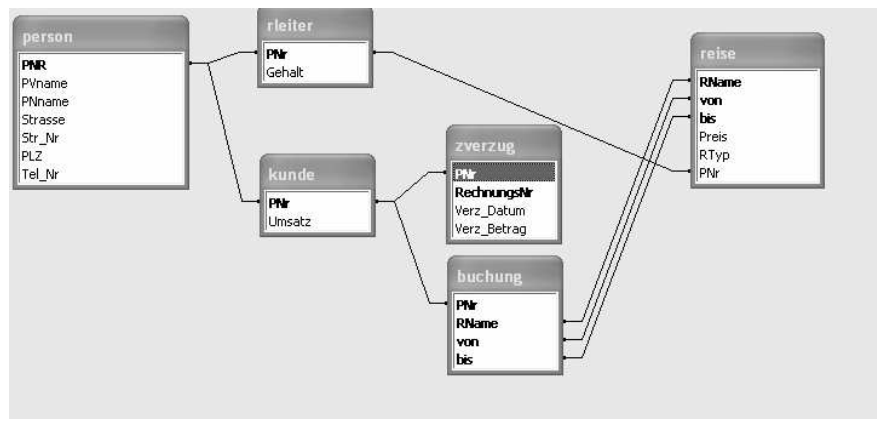
6.4 Physischer Entwurf (10/11)

Beispiel

Relationenübergreifende (interrelationale) Integritätsbedingungen:

- rleiter.PNr \subseteq person.PNr
- kunde.PNr \subseteq person.PNr
- reise.PNr \subseteq rleiter.PNr
- zverzug.PNr \subseteq kunde.PNr
- buchung.PNr \subseteq kunde.PNr
- buchung.{RName, von, bis} \subseteq reise.{R_Name, von, bis}

Graphisch definierbar (MS ACCESS) oder direkt in SQL, z.B. (MS ACCESS):



```

CREATE TABLE rleiter (
    PNr Integer not null,
    Gehalt Currency,
    CONSTRAINT C1 PRIMARY KEY(PNr),
    CONSTRAINT C2 FOREIGN KEY(PNr)
        REFERENCES person(PNr)
)
    
```

6.4 Physischer Entwurf (11/11)

Anmerkungen:

- alle Relationen gleichberechtigt
- alle Operationen auf jede Relation anwendbar (→ keine „Einstiegspunkte“ !)
- Relationsschema:
kann geändert / gelöscht werden (wenn auch nicht „regelmäßig“)
- Zusätzlich Angaben über:
 - Zugriffsberechtigung
 - Integritätssicherung

(Thema der Vorlesung DBS)

Anforderungsanalyse

Konzeptueller Entwurf

Logischer Entwurf

Physischer Entwurf

AIFBO

WS05/06

Literatur

- G. Schlageter, W. Stucky. **Datenbanksysteme: Konzepte und Modelle**. B. G. Teubner, Stuttgart 1983 (2. Auflage)
Download:
<http://www.aifb.uni-karlsruhe.de/Lehre/Winter2005-06/AI1/Literatur.html>
(Passwort wird nur in der Vorlesung bekannt gegeben)
- R. Elmasri, S. B. Navathe. **Fundamentals of Database Systems**. Pearson Education, 4. Aufl., 2004, ISBN 0321204484
- H. Garcia-Molina, J.D. Ullman, J. Widom. **Database Systems. The Complete Book**. Prentice Hall, 2002, ISBN 0130319953
- G. Vossen. **Datenbankmodelle, Datenbanksprachen und Datenbankmanagementsysteme**. Oldenbourg, München, 2000, ISBN: 3486253395.
- S. M. Lang, P. C. Lockemann. **Datenbankeinsatz**. Springer, 1995, ISBN: 3-540-58558-3