



8. Logischer DB-Entwurf

Inhalt

Grundlagen

Funktionale Abhängigkeiten

Zerlegung von Relationen

Normalformenlehre



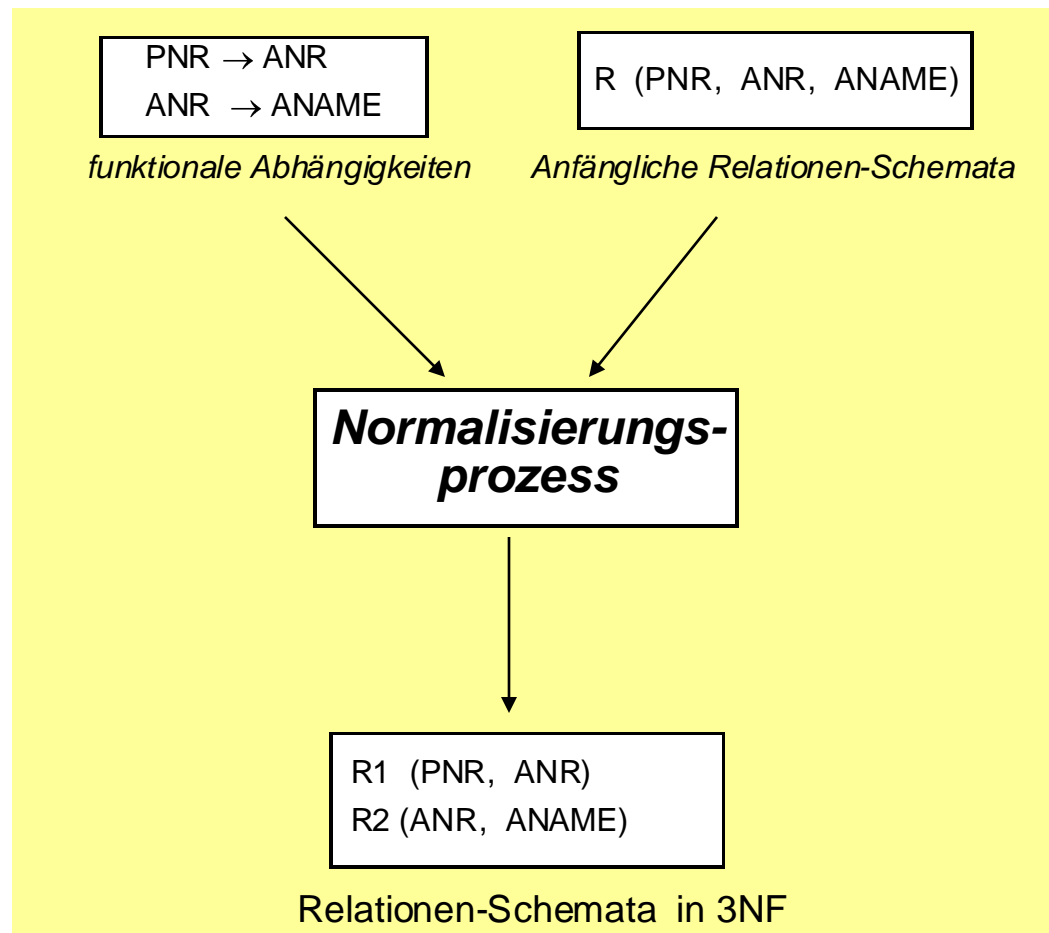
Grundlagen (1)

- ZIEL
 - Theoretische Grundlage für den Entwurf eines „guten“ relationalen DB-Schemas (→ Entwurfstheorie, Normalisierungslehre)
- GÜTE
 - leichte Handhabbarkeit, Verständlichkeit, Natürlichkeit, Übersichtlichkeit, ...
 - Entwurfstheorie präzisiert/formalisiert „Güte“ z. T.
- Beispiele

KunterBunt	(<u>A1</u> , A2, A3, ..., A300)
ABTMGR	(<u>ANR</u> , ANAME, BUDGET, MNR, PNAME, TITEL, SEIT_JAHR)
- Was macht einen schlechten DB-Schema-Entwurf aus?
 - implizite Darstellung von Informationen
 - Redundanzen, potentielle Inkonsistenz (Änderungsanomalien)
 - Einfügeanomalien, Löschanomalien
 - ...

Grundlagen (2)

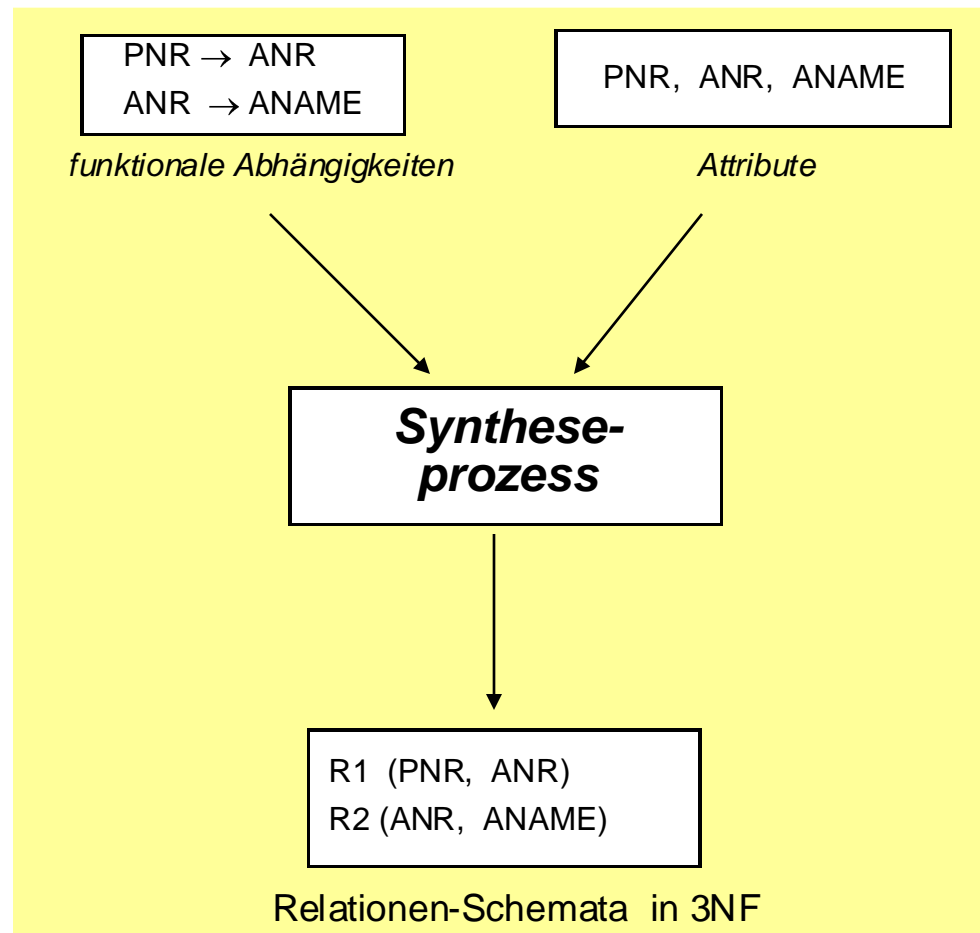
- **Normalisierung** von Relationen
hilft einen gegebenen Entwurf zu verbessern



Grundlagen (3)

- **Synthese** von Relationen
zielt auf die Konstruktion eines „optimalen“ DB-Schemas ab

*wird hier nicht
näher behandelt*





Grundlagen (4)

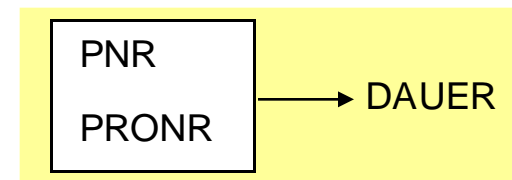
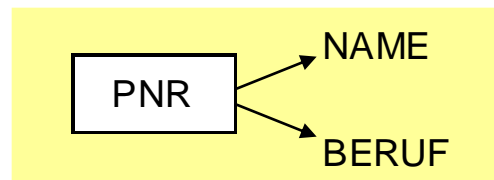
- Funktionale Abhängigkeiten

- **Konventionen**

- \mathcal{R}, S Relationenschemata (Relationenname, Attribute)
 - R, S Relationen der Relationenschemata \mathcal{R}, S
 - A, B, C, \dots einfache Attribute
 - $\mathcal{A} = \{A_1, \dots, A_n\}$ vollständige Attributmenge eines Relationenschemas
 - W, X, Y, Z, \dots Mengen von Attributen
 - a, b, c Werte einfacher Attribute
 - w, x, y, z Werte(kombinationen) von W, X, Y, Z

Grundlagen (5)

- Funktionale Abhängigkeiten (Forts.)
 - **Funktionale Abhängigkeit (FA)** (engl. *functional dependency*)
 - Die FA $X \rightarrow Y$ (X bestimmt Y funktional) gilt, wenn für alle R von \mathcal{R} gilt:
Zwei Tupel, deren Komponenten in X übereinstimmen,
stimmen auch in Y überein: $\forall t \in R, \forall u \in R: (t.X = u.X) \Rightarrow (t.Y = u.Y)$
 - *alternativ*: Die Relation R erfüllt die FA $X \rightarrow Y$, wenn für jeden X-Wert x $\pi_Y(\sigma_{X=x}(R))$ höchstens ein Tupel hat.
 - **Notation**
 - $\{PNR\} \rightarrow \{NAME, BERUF\}$: verkürzt $PNR \rightarrow NAME, BERUF$
 - $\{PNR, PRONR\} \rightarrow \{DAUER\}$: verkürzt $PNR, PRONR \rightarrow DAUER$



Grundlagen (6)

- Funktionale Abhängigkeiten (Forts.)
 - **Beispiel:** Gegeben sei die Relation R des Relationenschemas \mathcal{R} mit $\mathcal{A} = \{A, B, C, D\}$:

A	B	C	D
a ₁	b ₁	c ₁	d ₁
a ₁	b ₁	c ₁	d ₂
a ₂	b ₂	c ₃	d ₂
a ₃	b ₂	c ₄	d ₃
a ₄	b ₂	c ₄	d ₃

R erfüllt die FAs:

$A \rightarrow B; A \rightarrow C; C \rightarrow B; A, B \rightarrow C; \dots$



Grundlagen (7)

- Funktionale Abhängigkeiten (Forts.)
 - **Triviale funktionale Abhängigkeit**
 - Funktionale Abhängigkeiten, die von jeder Relationenausprägung automatisch immer erfüllt sind, nennt man triviale FAs.
 - Nur FAs der Art $X \rightarrow Y$ mit $Y \subseteq X$ sind trivial.
 - Es gilt $\mathcal{R} \rightarrow \mathcal{R}$
 - **Achtung:**
 - FAs lassen sich nicht durch Analyse einer Relation R gewinnen. Sie sind vom Entwerfer festzulegen.
 - FAs beschreiben semantische Integritätsbedingungen bezüglich der Attribute eines Relationenschemas, die jederzeit erfüllt sein müssen.



Grundlagen (8)

- Schlüssel

- **Superschlüssel**

- Im Relationenschema \mathcal{R} ist $X \subseteq \mathcal{R}$ ein Superschlüssel, falls gilt: $X \rightarrow \mathcal{R}$
- Falls X Schlüsselkandidat von \mathcal{R} , dann gilt für alle Y aus \mathcal{R} : $X \rightarrow Y$
- Wir benötigen das Konzept der vollen funktionalen Abhängigkeit, um Schlüssel (-kandidaten) von Superschlüsseln abzugrenzen.

- **Volle funktionale Abhängigkeit**

- Y ist voll funktional abhängig (\Rightarrow) von X , wenn gilt
 - $X \rightarrow Y$,
 - X ist „minimal“, d. h. $\forall A_i \in X : \neg(X - \{A_i\} \rightarrow Y)$
(Y ist funktional abhängig von X , aber nicht funktional abhängig von einer echten Teilmenge von X ;
falls $X \Rightarrow \mathcal{R}$ gilt, ist X Schlüsselkandidat von \mathcal{R})



Grundlagen (9)

- Schlüssel

- **Beispiel:**

Name	BLand	EW	VW
Kaiserslautern	Rlp	100000	0631
Mainz	Rlp	250000	06131
Frankfurt	Bdg	90000	0335
Frankfurt	Hes	700000	069
...			

Superschlüssel: N, B, E, V; N, B, E; N, B, V; N, B; N, V; ...

Schlüsselkandidaten: N, B; N, V.



Zerlegung von Relationen (1)

- **Anomalien**

- Änderungsanomalien

- erhöhter Speicherplatzbedarf wegen redundant gespeicherter Information
- gleichzeitige Aktualisierung aller redundanten Einträge erforderlich!
- Leistungseinbußen, da mehrere redundante Einträge geändert werden müssen

- Einfüge- und Löschanomalien

- Vermischung von Informationen zweier Entitytypen führt auf Probleme, wenn Information eingetragen/gelöscht werden soll, die nur zu einem der Entitytypen gehört
- Erzeugen vieler NULL-Werte oder Verlust von Information

- **Anomalien** sind darauf zurückzuführen, dass „**nicht zusammen passende**“ Informationen vermischt werden



Zerlegung von Relationen (2)

- **Grundlegende Korrektheitskriterien** für eine Zerlegung oder Normalisierung von Relationenschemata
 - **Verlustlosigkeit:** Die in der ursprünglichen Ausprägung R des Schemas \mathcal{R} enthaltenen Informationen müssen aus den Ausprägungen R_1, \dots, R_n der neuen Relationenschemata $\mathcal{R}_1, \dots, \mathcal{R}_n$ rekonstruierbar sein.
 - **Abhängigkeitsbewahrung:** Die für \mathcal{R} geltenden funktionalen Abhängigkeiten müssen auf die Schemata $\mathcal{R}_1, \dots, \mathcal{R}_n$ übertragbar sein.
- **Gültige Zerlegung:** $\mathcal{R} = \mathcal{R}_1 \cup \mathcal{R}_2$, d.h., alle Attribute aus \mathcal{R} bleiben erhalten

- **Verlustlose Zerlegung**

$$R_1 := \Pi_{\mathcal{R}_1}(R)$$

$$R_2 := \Pi_{\mathcal{R}_2}(R)$$

wenn für jede mögliche (gültige) Ausprägung R von \mathcal{R} gilt $R = R_1 \bowtie R_2$



Zerlegung von Relationen (3)

- **Beispiel 1:**

- FBSTUDENT (MATNR, NAME, FBNR, FBADR)
mit $\text{MATNR} \rightarrow \text{NAME, FBNR, FBADR}$
 $\text{FBNR} \rightarrow \text{FBADR}$
- $\text{STUDENT} := \Pi_{\text{MATNR, NAME, FBNR}} (\text{FBSTUDENT})$
 $\text{FB} := \Pi_{\text{FBNR, FBADR}} (\text{FBSTUDENT})$
- $\text{FBSTUDENT} = \text{STUDENT} \bowtie \text{FB}$

Zerlegung von Relationen (4)

- Eine verlustlose Zerlegung von FBSTUDENT

FBSTUDENT			
MATRNR	NAME	FBNR	FBADR
1234	Mueller	5	x
5678	Maier	1	x
9000	Schmidt	5	x
0007	Maier	2	y

STUDENT		
MATRNR	NAME	FBNR
1234	Mueller	5
5678	Maier	1
9000	Schmidt	5
0007	Maier	2

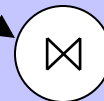
FB	
FBNR	FBADR
5	x
1	x
2	y

Zerlegung von Relationen (5)

- Eine verlustlose Zerlegung von FBSTUDENT (Forts.)

STUDENT		
MATRNR	NAME	FBNR
1234	Mueller	5
5678	Maier	1
9000	Schmidt	5
0007	Maier	2

FB	
FBNR	FBADR
5	x
1	x
2	y



FBSTUDENT			
MATRNR	NAME	FBNR	FBADR
1234	Mueller	5	x
5678	Maier	1	x
9000	Schmidt	5	x
0007	Maier	2	y

=

Zerlegung von Relationen (6)

- Beispiel 2

BIERTRINKER		
KNEIPE	GAST	BIER
Red Devil	Ernie	Guinness
Red Devil	Grouch	Bud
Bierhalle	Ernie	Bud

$\Pi_{KNEIPE, GAST}$

BESUCHT	
KNEIPE	GAST
Red Devil	Ernie
Red Devil	Grouch
Bierhalle	Ernie

$\Pi_{GAST, BIER}$

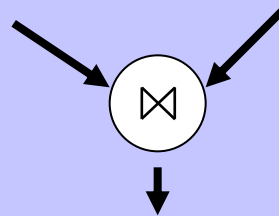
TRINKT	
GAST	BIER
Ernie	Guinness
Grouch	Bud
Ernie	Bud

Zerlegung von Relationen (7)

- Beispiel 2 (Forts.)

BESUCHT	
KNEIPE	GAST
Red Devil	Ernie
Red Devil	Grouch
Bierhalle	Ernie

TRINKT	
GAST	BIER
Ernie	Guinness
Grouch	Bud
Ernie	Bud



BESUCHT-TRINKT		
KNEIPE	GAST	BIER
Red Devil	Ernie	Guinness
Red Devil	Ernie	Bud
Red Devil	Grouch	Bud
Bierhalle	Ernie	Guinness
Bierhalle	Ernie	Bud

≠



Zerlegung von Relationen (8)

- **Beispiel 2 (Forts.): Warum verlustbehaftet?**
 - KNEIPE, GAST \rightarrow BIER ist die einzige nicht-triviale FA von BIERTRINKER
 - {KNEIPE, GAST} ist Schlüssel
 - Gewählte Zerlegung teilt Schlüssel auf



Zerlegung von Relationen (9)

- **Beispiel für Abhängigkeitsverlust**

- PLZverzeichnis (Straße, Ort, BLand, PLZ)
 - Orte werden durch ihren Namen (Ort) und das Bundesland (BLand) eindeutig identifiziert
 - PLZ-Gebiete gehen nicht über Ortsgrenzen und Orte nicht über BLand-Grenzen hinweg
 - Innerhalb einer Straße ändert sich die PLZ nicht
 - FAs
 - $PLZ \rightarrow ORT, BLand$
 - $Stra\beta e, Ort, BLand \rightarrow PLZ$

Straße, Ort, BLand → PLZ
„Frankfurt, Bdg, Goethestr, 15235“

Zerlegung von Relationen (10)

- **Beispiel für Abhängigkeitsverlust (Forts.)**

PLZverzeichnis			
ORT	BLand	Straße	PLZ
Frankfurt	Hes	Goethestr	60313
Frankfurt	Hes	Schillerstr	60505
Frankfurt	Bdg	Goethestr	15234

$\Pi_{\text{PLZ, Straße}}$

Straßen	
PLZ	Straße
60313	Goethestr
60505	Schillerstr
15234	Goethestr

$\Pi_{\text{Ort, BLand, PLZ}}$

Orte		
ORT	BLand	PLZ
Frankfurt	Hes	60313
Frankfurt	Hes	60505
Frankfurt	Bdg	15234



Zerlegung von Relationen (11)

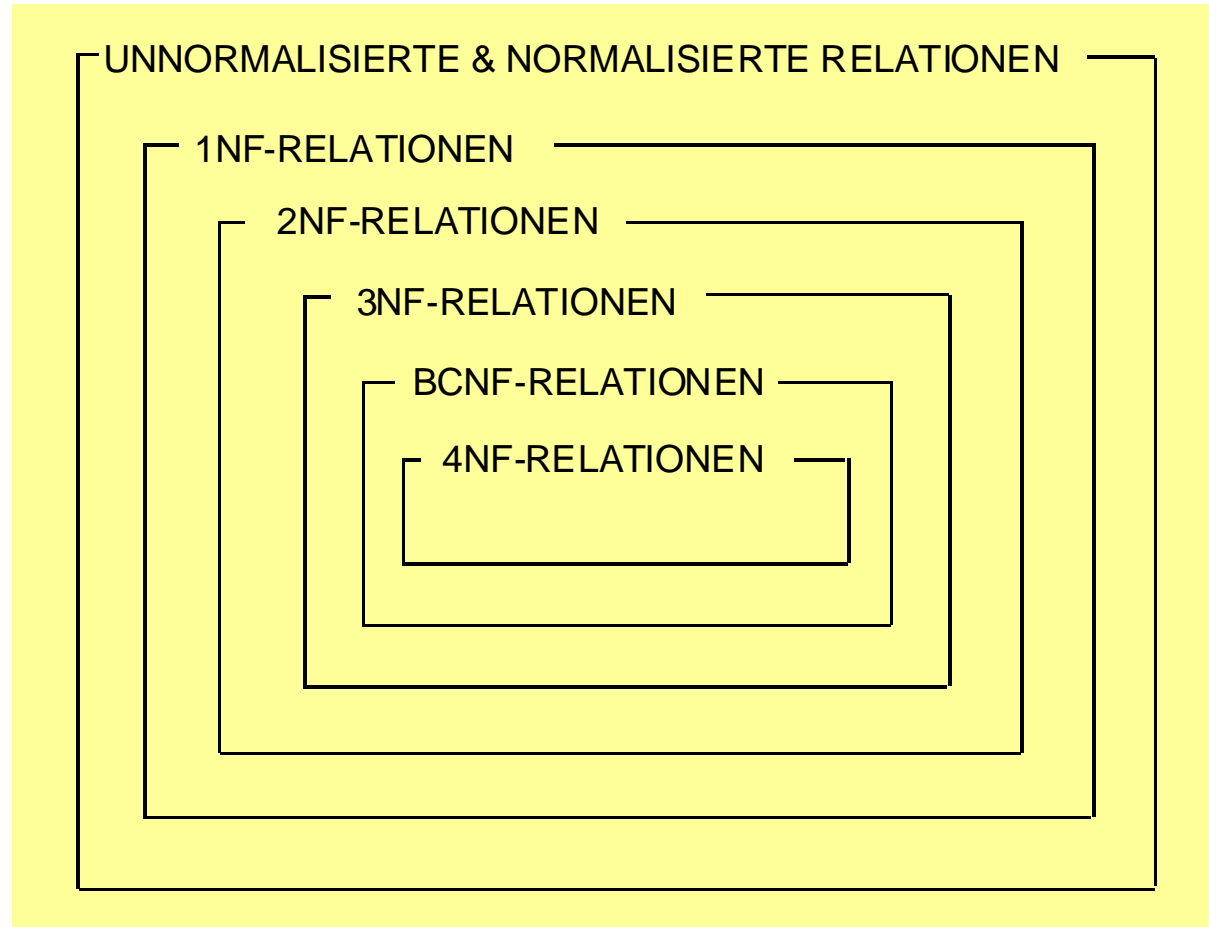
- **Beispiel für Abhängigkeitsverlust (Forts.)**

- Die FA
Straße, Ort, BLand → PLZ
ist im zerlegten Schema nicht mehr enthalten
- Einfügen eines Eintrags:
 - „Frankfurt, Bdg, Goethestr, 15235“
führt auf Verletzung dieser FA



Normalisierung (1)

- Übersicht





Normalisierung (2)

- Zerlegung eines Relationenschemas \mathcal{R} in höhere Normalformen
 - Beseitigung von Anomalien bei Änderungsoperationen
 - fortgesetzte Anwendung der Projektion im Zerlegungsprozess
 - bessere „Lesbarkeit“ der aus \mathcal{R} gewonnenen Relationen
 - Erhaltung aller nicht-redundanter Funktionalabhängigkeiten von \mathcal{R}
(sie bestimmen den Informationsgehalt von \mathcal{R})
 - Verlustlosigkeit der Zerlegung ist in allen Normalformen gewährleistet
 - Abhängigkeitserhaltung kann nur bei Zerlegungen bis zur 3NF garantiert werden



Normalisierung (3)

- Unnormalisierte Relation
 - Non-First Normal-Form (NF²)
 - Relation enthält „Attribute“, die wiederum Relationen sind
 - Darstellung von komplexen Objekten (hierarchische Sichten)
 - Beispiel:

PRÜFUNGSGESCHEHEN					
(<u>PNR</u> ,	PNAME,	FACH,	STUDENT	(MATNR, NAME, ...))
1	Härder	DBS			1234 Müller
					5678 Maier
					9000 Schmitt
2	Schock	FA			5678 Maier
					007 Coy

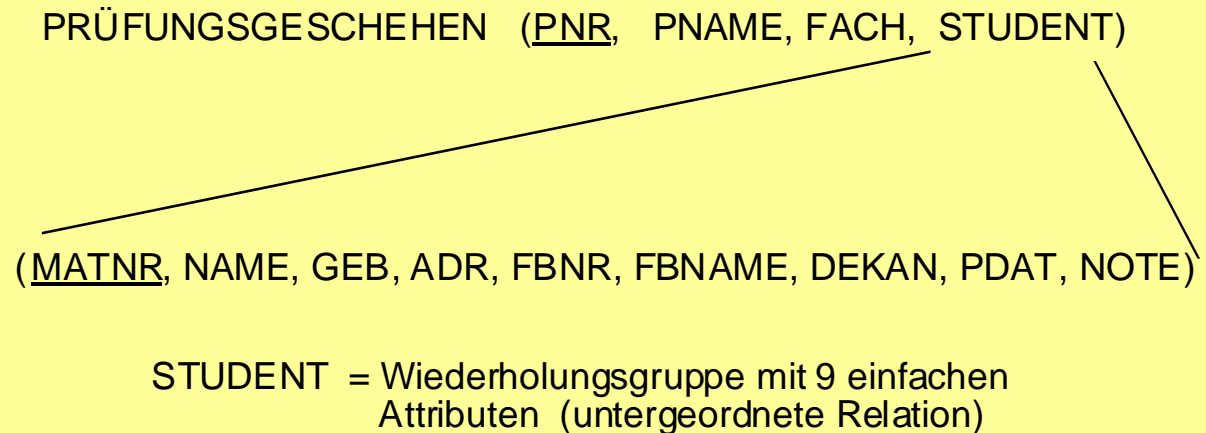


Normalisierung (4)

- Vorteile von NF^2
 - Clusterbildung
 - Effiziente Verarbeitung in einem hierarchisch strukturierten Objekt entlang der Vorzugsrichtung
- Nachteile von NF^2
 - Unsymmetrie (nur eine Richtung der Beziehung)
 - implizite Darstellung von Information
 - Redundanzen bei (n:m)-Beziehungen
 - Anomalien bei Aktualisierung
 - Definiertheit des Vaters
- Normalisierung
 - „Herunterkopieren“ von Werten führt hohen Grad an Redundanz ein
 - aber: Erhaltung ihres Informationsgehaltes

Normalisierung (5)

- Beispiel einer nicht normalisierten Relation



- Normalisierung (Überführung in 1NF):
 - Starte mit der übergeordneten Relation (Vaterrelation)
 - Nimm ihren Primärschlüssel und erweitere jede unmittelbar untergeordnete Relation damit zu einer selbständigen Relation
 - Streiche alle nicht-einfachen Attribute (untergeordnete Relationen) aus der Vaterrelation
 - Wiederhole diesen Prozess ggf. rekursiv



Normalisierung (6)

- Grundlegende Regeln bei der Überführung in 1NF
 - Nicht-einfache Attribute bilden neue Relationen
 - Primärschlüssel der übergeordneten wird an untergeordnete Relation angehängt ('copy down the key')
- Relationenschema in 1NF (zu obigem Beispiel):
PRÜFER (PNR, PNAME, FACH)
PRÜFUNG (PNR, MATNR, NAME, GEB, ADR, FBNR, FBNAME, DEKAN, PDAT, NOTE)
- Beobachtung
 - 1NF verursacht immer noch viele Änderungsanomalien, da verschiedene Entity-Mengen in einer Relation gespeichert werden können bzw. aufgrund von Redundanz innerhalb einer Relation (Beispiel: PRÜFUNG)
 - 2NF vermeidet einige der Anomalien dadurch, dass nicht voll funktional (partiell) abhängige Attribute eliminiert werden
 - Lösung: Separierung verschiedener Entity-Mengen in eigene Relationen



Normalisierung (7)

- Definition

- Ein **Primärattribut** (Schlüsselattribut) eines Relationenschemas ist ein Attribut, das zu mindestens einem Schlüsselkandidaten des Schemas gehört.
- Ein Relationenschema \mathcal{R} ist in **2NF**, wenn es in 1NF ist und jedes Nicht-Primärattribut von \mathcal{R} voll funktional von jedem Schlüsselkandidaten in \mathcal{R} abhängt.

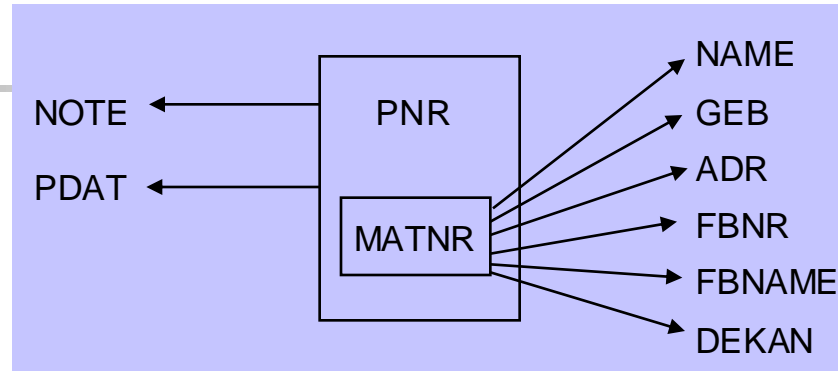
- Überführung in 2NF

- Bestimme funktionale Abhängigkeiten zwischen Nicht-Primärattributen und Schlüsselkandidaten
- Eliminiere partiell abhängige Attribute und fasse sie in eigener Relation zusammen (unter Hinzunahme der zugehörigen Primärattribute)

Normalisierung (8)

■ Beispiel

- Volle funktionale Abhängigkeiten in PRÜFUNG
- Relationen in 2NF



PRÜFUNG'

<u>PNR</u>	<u>MATNR</u>	PDAT	NOTE
1234	123 766	221001	4
1234	654 711	140200	3
3678	196 481	210999	2
3678	123 766	020301	4
8223	226 302	120701	1

PRÜFER

<u>PNR</u>	PNAME	FACH
1234	Schock	FA
3678	Härder	DBS
8223	Franke	FM

STUDENT'

<u>MATNR</u>	NAME	GEB	ADR	FBNR	FBNAME	DEKAN
123 766	Coy	050578	XX	FB1	Mathematik	Freden
654 711	Abel	211176	XY	FB9	Informatik	Avenhaus
196 481	Maier	010179	YX	FB9	Informatik	Avenhaus
226 302	Schulz	310778	YY	FB1	Mathematik	Freden



Normalisierung (9)

- Beobachtung
 - Änderungsanomalien in 2NF sind immer noch möglich aufgrund von *transitiven Abhängigkeiten*.
 - Beispiel: Vermischung von Fachbereichs- und Studentendaten in Student'
- Definition
 - Eine Attributmenge Z von Relationenschema \mathcal{R} ist transitiv abhängig von einer Attributmenge X in \mathcal{R} , wenn gilt:
 - X und Z sind disjunkt
 - Es existiert eine Attributmenge Y in \mathcal{R} , so dass gilt:
 $X \rightarrow Y, Y \rightarrow Z, Y \not\rightarrow X, Z \not\subseteq Y$
(keine strikte Transitivität gefordert, d.h., $Z \rightarrow Y$ erlaubt)
 - Ein Relationenschema \mathcal{R} befindet sich in **3NF**, wenn es sich in 2NF befindet und jedes Nicht-Primärattribut von \mathcal{R} von keinem Schlüsselkandidaten von \mathcal{R} transitiv abhängig ist.

Normalisierung (10)

■ Beispiel

PRÜFUNG'

<u>PNR</u>	<u>MATNR</u>	PDAT	NOTE
1234	123 766	221001	4
1234	654 711	140200	3
3678	196 481	210999	2
3678	123 766	020301	4
8223	226 302	120701	1

STUDENT''

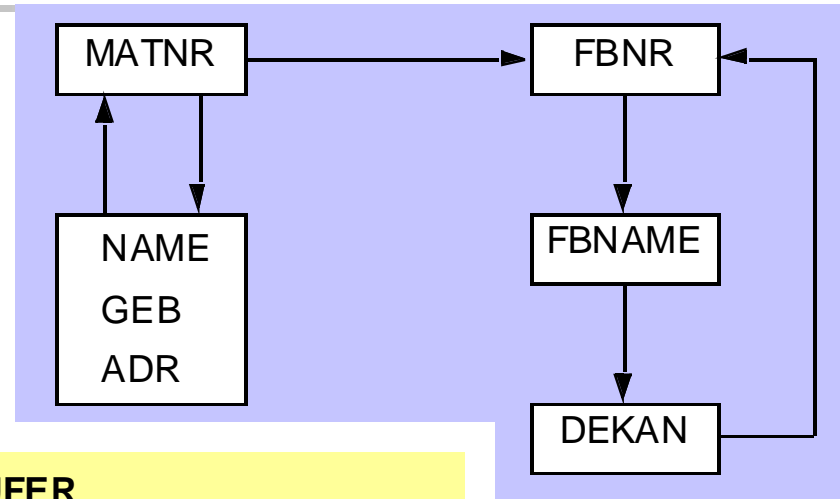
<u>MATNR</u>	NAME	GEB	ADR	FBNR
123 766	Coy	050578	XX	FB1
654 711	Abel	211176	XY	FB9
196 481	Maier	010179	YX	FB9
226 302	Schulz	310778	YY	FB1

PRÜFER

<u>PNR</u>	PNAME	FACH
1234	Schock	FA
3678	Härder	DBS
8223	Franke	FM

FACHBEREICH

<u>FBNR</u>	FBNAME	DEKAN
FB1	Mathematik	Freeden
FB9	Informatik	Avenhaus
FB2	Physik	Jodl

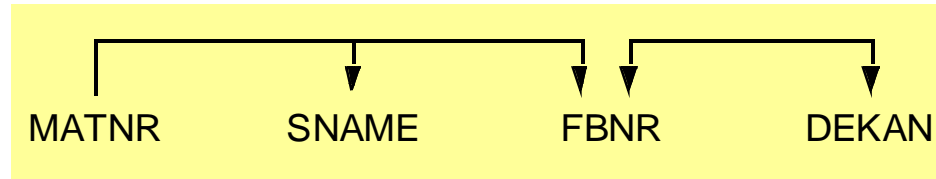


Normalisierung (11)

- Weitere Beispiele zu 3NF

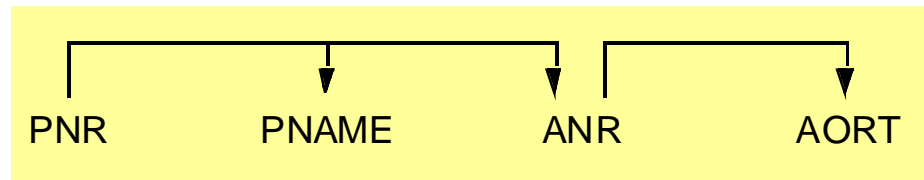
- Beispiel 1

- Transitive Abhängigkeit
- Zerlegung in 3NF:
R1 (MATNR, SNAME, FBNR)
R2 (FBNR, DEKAN)



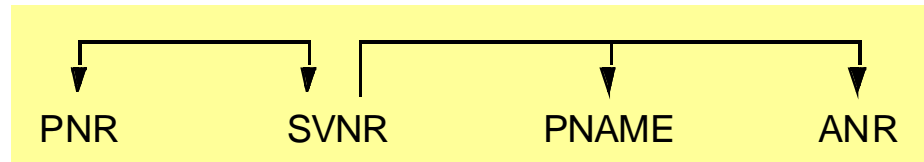
- Beispiel 2

- strikte transitive Abhängigkeit
- Zerlegung in 3NF:
R1 (PNR, PNAME, ANR)
R2 (ANR, AORT)



- Beispiel 3

- Keine transitive Abhängigkeit
- Keine Zerlegung
R1 (PNR, SVN, PNAME, ANR)



Normalisierung (12)

- Definition der 3NF hat gewisse Schwächen
 - bei Relationen mit mehreren Schlüsselkandidaten, wobei
 - Schlüsselkandidaten zusammengesetzt sind und
 - sich überlappen

- Beispiel

PRÜFUNG (PNR, MATNR, FACH, NOTE)
PRIMARY KEY (PNR, MATNR)
UNIQUE (FACH, MATNR)

- es bestehe eine (1:1)-Beziehung zwischen PNR und FACH
- einziges Nicht-Primärattribut: NOTE
- PRÜFUNG ist in 3NF

z. B. bei FACH

PRÜFUNG	(PNR ,	MATNR ,	FACH ,	NOTE)
	4	4711	BS	1
	4	1007	BS	2
	4	1234	BS	2
	5	4711	RO	3

- **Ziel**

- Ausschluss/Beseitigung der Anomalien in den Primärattributen



Normalisierung (13)

- Definition

- Ein Attribut (oder eine Gruppe von Attributen), von dem andere voll funktional abhängen, heißt Determinant.
- Ein Relationenschema \mathcal{R} ist in **BCNF** (Boyce-Codd-Normalform), wenn jeder Determinant ein Schlüsselkandidat von \mathcal{R} ist.
- Genauer: ein Relationenschema ist in **BCNF**, falls gilt: Wenn eine Sammlung von Attributen Y (voll funktional) abhängt von einer disjunkten Sammlung von Attributen X, dann hängt jede andere Sammlung von Attributen Z auch von X (voll funktional) ab,
 - d. h. für alle X, Y, Z mit X und Y disjunkt gilt:
 $X \rightarrow Y$ impliziert $X \rightarrow Z$



Normalisierung (14)

- Beispiel
 - Zerlegung von Prüfung
 - PRÜF1 (PNR, MATNR, NOTE), FBEZ (PNR, FACH) oder
 - PRÜF2 (MATNR, FACH, NOTE), FBEZ (PNR, FACH)
 - Beide Zerlegungen führen auf BCNF-Relationen
 - Änderungsanomalie ist verschwunden
 - alle funktionalen Abhängigkeiten sind erhalten

Normalisierung (15)

- Ist BCNF-Zerlegung immer sinnvoll?
- Beispiel

STUDENT, FACH \rightarrow PRÜFER; PRÜFER \rightarrow FACH

SFP (STUDENT FACH PRÜFER)

		Sloppy	DBS	Härder
		Hazy	DBS	Ritter
Sloppy	BS	Nehmer		

- jeder Prüfer prüft nur ein Fach (aber ein Fach kann von mehreren geprüft werden)
- jeder Student legt in einem bestimmten Fach nur eine Prüfung ab
- BCNF-Zerlegung: SP (STUDENT, PRÜFER), PF (PRÜFER, FACH)
- Neue Probleme
 - STUDENT, FACH \rightarrow PRÜFER ist nun „extern“, Konsistenzprüfung?
 - BCNF hier zu streng, um bei der Zerlegung alle funktionalen Abhängigkeiten zu bewahren (*key breaking dependency*)

Normalisierung (16)

■ Mehrwertige Abhängigkeiten (MWA)

- Eine FA drückt aus, dass zu einem Wert ‚des bestimmenden Attributs‘ jeweils (höchstens) ein Wert ‚des abhängigen Attributes‘ gehört
- MWAs sind Generalisierungen von FAs; es geht jeweils um Mengen von Werten ‚des abhängigen Attributes‘
- MWA entstehen durch zwei (oder mehr) unabhängige Attribute im Schlüssel einer Relation (all-key relation): z.B. Fähigkeiten - Kinder

■ Beispiel:	PNR	FÄHIGKEIT	KIND
	123	Englisch	Nadine
	123	Englisch	Philip
	123	Englisch	Tobias
	123	Progr.	Nadine
	123	Progr.	Philip
123	Progr.	Tobias	

Änderungsanomalien obwohl in BCNF!



Normalisierung (17)

- Definition

- X, Y, Z seien Attributmengen des Relationenschemas \mathcal{R} .

- Die *mehrwertige Abhängigkeit (MWA)*

$$X \twoheadrightarrow Y$$

gilt in \mathcal{R} genau dann, wenn die Menge der Y-Werte, die zu einem (X-Wert, Z-Wert)-Paar gehören, nur vom X-Wert bestimmt sind (d. h. unabhängig vom Z-Wert sind).

- MWA im Beispiel: $\text{PNR} \twoheadrightarrow \text{FÄHIGKEIT}$, $\text{PNR} \twoheadrightarrow \text{KIND}$

- $X \twoheadrightarrow Y$ impliziert $X \twoheadrightarrow Z$

Schreibweise: $X \twoheadrightarrow Y \mid Z$, z.B. $\text{PNR} \twoheadrightarrow \text{FÄHIGKEIT} \mid \text{KIND}$

- Jede FA ist auch eine MWA, nicht umgekehrt

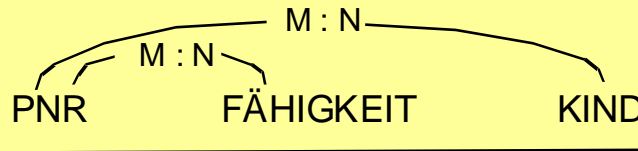


Normalisierung (18)

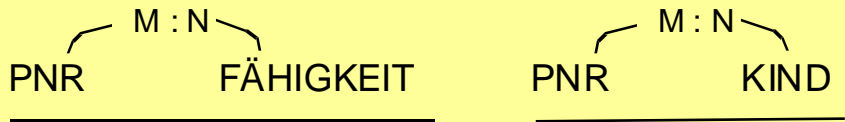
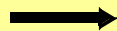
- 4NF
 - behandelt Probleme mit mehrwertigen Abhängigkeiten
 - Schlüssel darf nicht 2 oder mehr unabhängige mehrwertige Fakten enthalten
- Definition
 - Ein Relationenschema \mathcal{R} ist in **4NF**, wenn es in BCNF ist und jede MWA in \mathcal{R} eine FA ist.
- Überführung in 4NF
 - Zerlege Relationenschema mit MWA $X \twoheadrightarrow Y \mid Z$ in zwei neue Relationenschemata mit den Attributen X, Y und X, Z .

Normalisierung (19)

- Beispiel



ZERLEGUNG:



andere Attribute sind erlaubt:



PNR	FÄHIGK.	PNR	KIND
123	Englisch	123	Nadine
123	Prog.	123	Philip
		123	Tobias



Normalisierung (20)

- Abhängigkeit bei mehrwertigen Fakten
 - Wenn Abhängigkeit besteht, muss sie durch die Wertekombinationen ausgedrückt werden
 - **Beispiel:**
 - (M:N)-Beziehung zwischen: PNR - PROJEKT, PNR - FÄHIGKEIT
 - zusätzliche (M:N)-Beziehung zwischen PROJEKT und FÄHIGKEIT, d. h., Projektmitarbeit erfordert bestimmte Fähigkeiten

- **Gültige 4NF:** R (PNR, PROJEKT, FÄHIGKEIT)

123	P1	Progr.
123	P2	Progr.
123	P2	Englisch

- Zerlegung von R in zwei Projektionen R1 (PNR, PROJEKT) und R2 (PNR, FÄHIGKEIT) führt zu 'Verlust' von Information, da Join-Bildung auf den Projektionen vorher nicht existente Tupel generieren kann (*connection trap*).



Normalisierung (22)

- Normalformenlehre nach E.F. Codd
 - **1NF:** Ein Relationenschema \mathcal{R} ist in 1NF genau dann, wenn alle seine Wertebereiche nur atomare Werte besitzen.
 - **2NF:** Ein Relationenschema \mathcal{R} ist in 2NF, wenn es in 1NF ist und jedes Nicht-Primärattribut von \mathcal{R} voll funktional von jedem Schlüsselkandidaten von \mathcal{R} abhängt.
 - **3NF:** Ein Relationenschema \mathcal{R} ist in 3NF, wenn es in 2NF ist und jedes Nicht-Primärattribut von keinem Schlüsselkandidaten von \mathcal{R} transitiv abhängig ist.



Normalisierung (23)

- Normalformenlehre nach E.F. Codd (Forts.)
 - **3NF (BCNF):** Ein Relationenschema \mathcal{R} ist in BCNF, falls gilt:
Wenn eine Sammlung von Attributen Y (voll funktional) abhängt von einer disjunkten Sammlung von Attributen X , dann hängt jede andere Sammlung von Attributen Z auch von X (voll funktional) ab,
d.h. für alle X, Y, Z mit X und Y disjunkt gilt: $X \rightarrow Y$ impliziert $X \rightarrow Z$
 - **Alternative Definition der BCNF:**
Ein normalisiertes Relationenschema \mathcal{R} ist in 3NF (BCNF), wenn jeder Determinant in \mathcal{R} ein Schlüsselkandidat von R ist.
 - **4NF:** Ein Relationenschema \mathcal{R} ist in 4NF, wenn es in BCNF ist und jede MWA auf \mathcal{R} eine FA ist.



Zusammenfassung (1)

- Festlegung aller funktionalen Abhängigkeiten
 - unterstützt präzises Denken beim Entwurf
 - erlaubt Integritätskontrollen durch das DBS
- ZIEL
 - klare und natürliche Zuordnung von Objekt und Datenstruktur
 - durch einen Satztyp (Relation) wird genau ein Objekttyp beschrieben
- Normalisierung von Relationen
 - lokales Verfahren auf existierenden Datenstrukturen
 - schrittweise Eliminierung von Änderungsanomalien



Zusammenfassung (2)

- Synthese von Relationen (*nicht im Detail behandelt*)
 - globales Verfahren liefert 3NF-Relationen
 - ggf. Überprüfung von überlappenden Schlüsselkandidaten, mehrwertigen Abhängigkeiten und Join-Abhängigkeiten (BCNF-, 4NF-Zerlegung, etc.)
- Weitere Probleme
 - Definition aller relevanten FAs bei sehr vielen Attributen schwierig
 - Entwurfs-Algorithmen liefern i. allg. mehrere minimale Überdeckungen
 - Bei Überführung von 3NF in BCNF können FAs verloren gehen
- Überarbeitung des DB-Schemas
 - Stabilitätsgesichtspunkte/Änderungshäufigkeiten können schwächere Normalformen erzwingen
 - Berücksichtigung von Abstraktionskonzepten
 - Der Entwerfer, und nicht die Methode, bestimmt den Entwurf