

Teil V

Relationaler Entwurf

# Relationaler Entwurf

- 1 Zielmodell des logischen Entwurfs
- 2 Relationaler DB-Entwurf
- 3 Normalformen
- 4 Transformationseigenschaften
- 5 Weitere Abhängigkeiten

## Lernziele für heute ...

- Kenntnisse zur Verfeinerung des relationalen Entwurfs
- Verständnis der Normalformen
- Methodik und Verfahren zur Normalisierung



# Relationenmodell

<b>WEINE</b>	<b>WeinID</b>	<b>Name</b>	<b>Farbe</b>	<b>Jahrgang</b>	<b>Weingut</b>
	1042	La Rose Grand Cru	Rot	1998	Château La Rose
	2168	Creek Shiraz	Rot	2003	Creek
	3456	Zinfandel	Rot	2004	Helena
	2171	Pinot Noir	Rot	2001	Creek
	3478	Pinot Noir	Rot	1999	Helena
	4711	Riesling Reserve	Weiß	1999	Müller
	4961	Chardonnay	Weiß	2002	Bighorn

<b>ERZEUGER</b>	<b>Weingut</b>	<b>Anbaugebiet</b>	<b>Region</b>
	Creek	Barossa Valley	South Australia
	Helena	Napa Valley	Kalifornien
	Château La Rose	Saint-Emilion	Bordeaux
	Château La Pointe	Pomerol	Bordeaux
	Müller	Rheingau	Hessen
	Bighorn	Napa Valley	Kalifornien

# Begriffe des Relationenmodells

<b>Begriff</b>	<b>Informale Bedeutung</b>
Attribut	Spalte einer Tabelle
Wertebereich	mögliche Werte eines Attributs (auch Domäne)
Attributwert	Element eines Wertebereichs
Relationenschema	Menge von Attributen
Relation	Menge von Zeilen einer Tabelle
Tupel	Zeile einer Tabelle
Datenbankschema	Menge von Relationenschemata
Datenbank	Menge von Relationen (Basisrelationen)

# Begriffe des Relationenmodells /2

<b>Begriff</b>	<b>Informale Bedeutung</b>
Schlüssel	minimale Menge von Attributen, deren Werte ein Tupel einer Tabelle eindeutig identifizieren
Primärschlüssel	ein beim Datenbankentwurf ausgezeichneter Schlüssel
Fremdschlüssel	Attributmenge, die in einer anderen Relation Schlüssel ist
Fremdschlüsselbedingung	alle Attributwerte des Fremdschlüssels tauchen in der anderen Relation als Werte des Schlüssels auf

# Formalisierung Relationenmodell

## ● Attribute und Domänen

- ▶  $\mathcal{U}$  nichtleere, endliche Menge: **Universum**
- ▶  $A \in \mathcal{U}$ : **Attribut**
- ▶  $\mathcal{D} = \{D_1, \dots, D_m\}$  Menge endlicher, nichtleerer Mengen: jedes  $D_i$ :  
**Wertebereich** oder **Domäne**
- ▶ total definierte Funktion  $\text{dom} : \mathcal{U} \longrightarrow \mathcal{D}$
- ▶  $\text{dom}(A)$ : **Domäne von A**  
 $w \in \text{dom}(A)$ : **Attributwert** für A

# Formalisierung Relationenmodell /2

## ● Relationenschemata und Relationen

- ▶  $R \subseteq \mathcal{U}$ : **Relationenschema**
- ▶ **Relation**  $r$  über  $R = \{A_1, \dots, A_n\}$  (kurz:  $r(R)$ ) ist endliche Menge von Abbildungen  $t : R \rightarrow \bigcup_{i=1}^m D_i$ , **Tupel** genannt
- ▶ Es gilt  $t(A) \in \text{dom}(A)$  ( $t(A)$  Restriktion von  $t$  auf  $A \in R$ )
- ▶ für  $X \subseteq R$  analog  $t(X)$  **X-Wert** von  $t$
- ▶ Menge aller Relationen über  $R$ : **REL**( $R$ ) :=  $\{r \mid r(R)\}$



# Formalisierung Relationenmodell /3

## ● Datenbankschema und Datenbank

- ▶ Menge von Relationenschemata  $S := \{R_1, \dots, R_p\}$ :  
Datenbankschema
- ▶ Datenbank über  $S$ : Menge von Relationen  $d := \{r_1, \dots, r_p\}$ , wobei  
 $r_i(R_i)$
- ▶ Datenbank  $d$  über  $S$ :  $d(S)$
- ▶ Relation  $r \in d$ : Basisrelation

# Integritätsbedingungen

- Identifizierende Attributmenge  $K := \{B_1, \dots, B_k\} \subseteq R$ :

$$\forall t_1, t_2 \in r [t_1 \neq t_2 \implies \exists B \in K : t_1(B) \neq t_2(B)]$$

- **Schlüssel**: ist minimale identifizierende Attributmenge
  - ▶ {Name, Jahrgang, Weingut} und
  - ▶ {WeinID} für WEINE
- **Primattribut**: Element eines Schlüssels
- **Primärschlüssel**: ausgezeichnete Schlüssel
- **Oberschlüssel** oder **Superkey**: jede Obermenge eines Schlüssels (= identifizierende Attributmenge)
- **Fremdschlüssel**:  $X(R_1) \rightarrow Y(R_2)$

$$\{t(X) | t \in r_1\} \subseteq \{t(Y) | t \in r_2\}$$

# Relationaler DB-Entwurf: Überblick

- Verfeinern des logischen Entwurfs
- Ziel: Vermeidung von Redundanzen durch Aufspalten von Relationenschemata, ohne gleichzeitig
  - ▶ semantische Informationen zu verlieren (Abhängigkeitstreue)
  - ▶ die Möglichkeit zur Rekonstruktion der Relationen zu verlieren (Verbundtreue)
- Redundanzvermeidung durch Normalformen (s.u.)

# Relation mit Redundanzen

WEINE	WeinID	Name	...	Weingut	Anbaugebiet	Region
	1042	La Rose Gr. Cru	...	Ch. La Rose	Saint-Emilion	Bordeaux
	2168	Creek Shiraz	...	Creek	Barossa Valley	South Australia
	3456	Zinfandel	...	Helena	Napa Valley	Kalifornien
	2171	Pinot Noir	...	Creek	Barossa Valley	South Australia
	3478	Pinot Noir	...	Helena	Napa Valley	Kalifornien
	4711	Riesling Res.	...	Müller	Rheingau	Hessen
	4961	Chardonnay	...	Bighorn	Napa Valley	Kalifornien

# Redundanzen

- Redundanzen in Basisrelationen aus mehreren Gründen unerwünscht:
  - ▶ Redundante Informationen belegen unnötigen **Speicherplatz**
  - ▶ **Änderungsoperationen** auf Basisrelationen mit Redundanzen nur schwer korrekt umsetzbar: wenn eine Information redundant vorkommt, muss eine Änderung diese Information in allen ihren Vorkommen verändern
    - ★ mit normalen relationalen Änderungsoperationen und den in relationalen Systemen vorkommenden lokalen Integritätsbedingungen (Schlüsseln) nur schwer realisierbar

# Änderungsanomalien

- Einfügen in die mit Redundanzen behaftete WEINE-Relation:

```
insert into WEINE (WeinID, Name, Farbe, Jahrgang,  
    Weingut, Anbaugebiet, Region)  
values (4711, 'Chardonnay', 'Weiß', 2004,  
    'Helena', 'Rheingau', 'Kalifornien')
```

- ▶ WeinID 4711 bereits anderem Wein zugeordnet: verletzt FD  
WeinID → Name
  - ▶ Weingut Helena war bisher im Napa Valley angesiedelt: verletzt FD  
Weingut → Anbaugebiet
  - ▶ Rheingau liegt nicht in Kalifornien: verletzt FD  
Anbaugebiet → Region
- auch **update**- und **delete**-Anomalien

# Funktionale Abhängigkeiten

- **funktionale Abhängigkeit** zwischen Attributmengen  $X$  und  $Y$  einer Relation

Wenn in jedem Tupel der Relation der Attributwert unter den  $X$ -Komponenten den Attributwert unter den  $Y$ -Komponenten festlegt.

- Unterscheiden sich zwei Tupel in den  $X$ -Attributen nicht, so haben sie auch gleiche Werte für alle  $Y$ -Attribute
- Notation für funktionale Abhängigkeit (FD, von functional dependency):  $X \rightarrow Y$
- Beispiel:  
WeinID  $\rightarrow$  Name, Weingut  
Anbaugebiet  $\rightarrow$  Region
- aber nicht: Weingut  $\rightarrow$  Name

# Schlüssel als Spezialfall

- für Beispiel auf Folie 5-11

WeinID  $\rightarrow$  Name, Farbe, Jahrgang, Weingut, Anbaugebiet, Region

- Immer: WeinID  $\rightarrow$  WeinID, dann gesamtes Schema auf rechter Seite
- Wenn linke Seite minimal: Schlüssel
- Formal: Schlüssel  $X$  liegt vor, wenn für Relationenschema  $R$  FD  $X \rightarrow R$  gilt und  $X$  minimal

**Ziel des Datenbankentwurfs:** alle gegebenen funktionalen Abhängigkeiten in „Schlüsselabhängigkeiten“ umformen, ohne dabei semantische Information zu verlieren



# Ableitung von FDs

$r$

<b>A</b>	<b>B</b>	<b>C</b>
$a_1$	$b_1$	$c_1$
$a_2$	$b_1$	$c_1$
$a_3$	$b_2$	$c_1$
$a_4$	$b_1$	$c_1$

- Tabelle genügt  $A \rightarrow B$  und  $B \rightarrow C$
- dann gilt auch  $A \rightarrow C$
- nicht ableitbar  $C \rightarrow A$  oder  $C \rightarrow B$

*Formalisierung im nächsten Abschnitt!*

# Schemaeigenschaften

- Relationenschemata, Schlüssel und Fremdschlüssel so wählen, dass
  - ① alle Anwendungsdaten aus den Basisrelationen hergeleitet werden können,
  - ② nur semantisch sinnvolle und konsistente Anwendungsdaten dargestellt werden können und
  - ③ die Anwendungsdaten möglichst nicht-redundant dargestellt werden.
- Hier: Forderung 3
  - ▶ Redundanzen innerhalb einer Relation: **Normalformen**
  - ▶ globale Redundanzen: **Minimalität**

# Normalformen

- legen Eigenschaften von Relationenschemata fest
- verbieten bestimmte Kombinationen von funktionalen Abhängigkeiten in Relationen
- sollen Redundanzen und Anomalien vermeiden

# Erste Normalform

- erlaubt nur atomare Attribute in den Relationenschemata, d.h. als Attributwerte sind Elemente von Standard-Datentypen wie **integer** oder **string** erlaubt, aber keine Konstrukturen wie **array** oder **set**
- Nicht in 1NF:

Weingut	Anbaugebiet	Region	WName
Ch. La Rose	Saint-Emilion	Bordeaux	La Rose Grand Cru
Creek	Barossa Valley	South Australia	Creek Shiraz, Pinot Noir
Helena	Napa Valley	Kalifornien	Zinfandel, Pinot Noir
Müller	Rheingau	Hessen	Riesling Reserve
Bighorn	Napa Valley	Kalifornien	Chardonnay

# Erste Normalform /2

- in erster Normalform:

Weingut	Anbaugebiet	Region	WName
Ch. La Rose	Saint-Emilion	Bordeaux	La Rose Grand Cru
Creek	Barossa Valley	South Australia	Creek Shiraz
Creek	Barossa Valley	South Australia	Pinot Noir
Helena	Napa Valley	Kalifornien	Zinfandel
Helena	Napa Valley	Kalifornien	Pinot Noir
Müller	Rheingau	Hessen	Riesling Reserve
Bighorn	Napa Valley	Kalifornien	Chardonnay

## Zweite Normalform

- **partielle Abhängigkeit** liegt vor, wenn ein Attribut funktional schon von einem **Teil** des Schlüssels abhängt

Name	Weingut	Farbe	Anbaugebiet	Region	Preis
La Rose Grand Cru	Ch. La Rose	Rot	Saint-Emilion	Bordeaux	39.00
Creek Shiraz	Creek	Rot	Barossa Valley	South Australia	7.99
Pinot Noir	Creek	Rot	Barossa Valley	South Australia	10.99
Zinfandel	Helena	Rot	Napa Valley	Kalifornien	5.99
Pinot Noir	Helena	Rot	Napa Valley	Kalifornien	19.99
Riesling Reserve	Müller	Weiß	Rheingau	Hessen	14.99
Chardonnay	Bighorn	Weiß	Napa Valley	Kalifornien	9.90

$f_1$ : Name, Weingut  $\rightarrow$  Preis

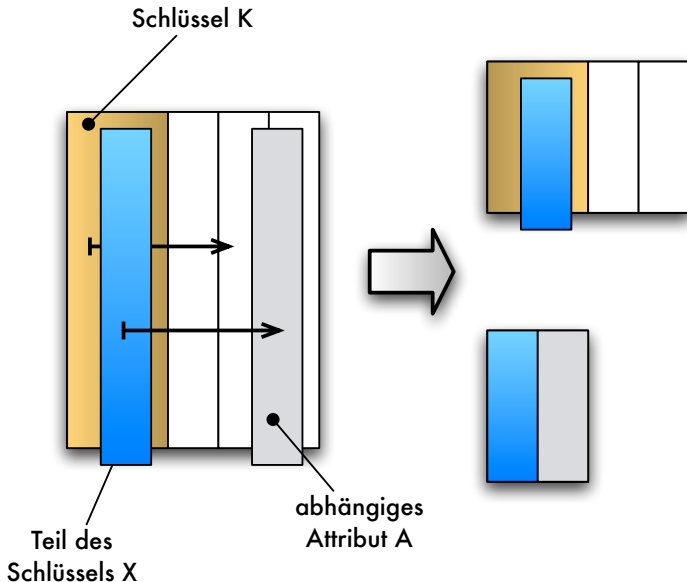
$f_2$ : Name  $\rightarrow$  Farbe

$f_3$ : Weingut  $\rightarrow$  Anbaugebiet, Region

$f_4$ : Anbaugebiet  $\rightarrow$  Region

- Zweite Normalform eliminiert derartige partielle Abhängigkeiten bei Nichtschlüsselattributen

# Eliminierung partieller Abhängigkeiten



## Zweite Normalform /2

- Beispielrelation in 2NF

R1(Name, Weingut, Preis)

R2(Name, Farbe)

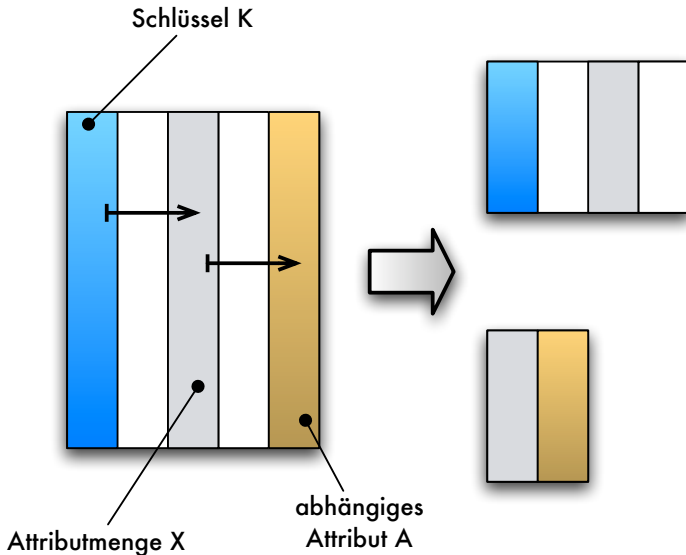
R3(Weingut, Anbaugebiet, Region)



## Dritte Normalform

- eliminiert (zusätzlich) transitive Abhängigkeiten
- etwa Weingut  $\rightarrow$  Anbaugebiet und Anbaugebiet  $\rightarrow$  Region in Relation auf Folie 5-21
- man beachte: 3NF betrachtet nur Nicht-Schlüsselattribute als Endpunkt transitiver Abhängigkeiten

# Eliminierung transitiver Abhängigkeiten



## Dritte Normalform /2

- transitive Abhängigkeit in R3, d.h. R3 verletzt 3NF
- Beispielrelation in 3NF

R3\_1(Weingut, Anbauggebiet)

R3\_2(Anbauggebiet, Region)

## Dritte Normalform: formal

- Relationenschema  $R$ ,  $X \subseteq R$  und  $F$  ist eine FD-Menge über  $R$
- $A \in R$  heißt **transitiv abhängig** von  $X$  bezüglich  $F$  genau dann, wenn es ein  $Y \subseteq R$  gibt mit  $X \rightarrow Y$ ,  $Y \not\rightarrow X$ ,  $Y \rightarrow A$ ,  $A \notin XY$
- erweitertes Relationenschema  $\mathcal{R} = (R, \mathcal{K})$  ist in **3NF** bezüglich  $F$  genau dann, wenn

$\nexists A \in R : \quad A$  ist Nicht-Primattribut in  $R$   
 $\wedge A$  transitiv abhängig von einem  $K \in \mathcal{K}$  bezüglich  $F$ .

- Nicht-Primattribut:  $A$  ist in keinem Schlüssel von  $R$  enthalten

# Boyce-Codd-Normalform

- Verschärfung der 3NF: Eliminierung transitiver Abhängigkeiten auch zwischen Primattributen

Name	Weingut	Händler	Preis
La Rose Grand Cru	Château La Rose	Weinkontor	39.90
Creek Shiraz	Creek	Wein.de	7.99
Pinot Noir	Creek	Wein.de	10.99
Zinfandel	Helena	GreatWines.com	5.99
Pinot Noir	Helena	GreatWines.com	19.99
Riesling Reserve	Müller	Weinkeller	19.99
Chardonnay	Bighorn	Wein-Dealer	9.90

- FDs:

Name, Weingut  $\rightarrow$  Preis

Weingut  $\rightarrow$  Händler

Händler  $\rightarrow$  Weingut

- Schlüsselkandidaten: { Name, Weingut } und { Name, Händler }
- in 3NF, nicht jedoch in BCNF

## Boyce-Codd-Normalform /2

- erweitertes Relationenschema  $\mathcal{R} = (R, \mathcal{K})$ , FD-Menge  $F$
- BCNF formal:

$\nexists A \in R : A$  transitiv abhängig von einem  $K \in \mathcal{K}$  bezüglich  $F$ .

- Schema in BCNF:

WEINE(Name, Weingut, Preis)

WEINHANDEL(Weingut, Händler)

- BCNF kann jedoch **Abhängigkeitstreue** verletzen, daher oft nur bis 3NF

# Minimalität

- Global Redundanzen vermeiden
- andere Kriterien (wie Normalformen) mit möglichst wenig Schemata erreichen
- Beispiel: Attributmenge  $ABC$ , FD-Menge  $\{A \rightarrow B, B \rightarrow C\}$
- Datenbankschemata in dritter Normalform:

$$S = \{(AB, \{A\}), (BC, \{B\})\}$$

$$S' = \{(AB, \{A\}), (BC, \{B\}), (AC, \{A\})\}$$

Redundanzen in  $S'$

# Schemaeigenschaften

Kennung	Schemaeigenschaft	Kurzcharakteristik
	1NF	nur atomare Attribute
	2NF	keine partielle Abhängigkeit eines Nicht-Primattributes von einem Schlüssel
<b>S1</b>	3NF	keine transitive Abhängigkeit eines Nicht-Primattributes von einem Schlüssel
	BCNF	keine transitive Abhängigkeit eines Attributes von einem Schlüssel
<b>S2</b>	Minimalität	minimale Anzahl von Relationenschemata, die die anderen Eigenschaften erfüllt



# Transformationseigenschaften

- Bei einer Zerlegung einer Relation in mehrere Relationen ist darauf zu achten, dass
  - 1 nur semantisch sinnvolle und konsistente Anwendungsdaten dargestellt (**Abhängigkeitstreue**) und
  - 2 alle Anwendungsdaten aus den Basisrelationen hergeleitet werden können (**Verbundtreue**)

# Abhängigkeitstreue

- **Abhängigkeitstreue**: eine Menge von Abhängigkeiten kann äquivalent in eine zweite Menge von Abhängigkeiten transformiert werden
- spezieller: in die Menge der Schlüsselabhängigkeiten, da diese vom Datenbanksystem effizient überprüft werden kann
  - ▶ die Menge der Abhängigkeiten soll äquivalent zu der Menge der Schlüsselbedingungen im resultierenden Datenbankschema sein
  - ▶ Äquivalenz sichert zu, dass mit den Schlüsselabhängigkeiten semantisch genau die gleichen Integritätsbedingungen ausgedrückt werden wie mit den funktionalen oder anderen Abhängigkeiten vorher

# Abhängigkeitstreue: Beispiel

- Zerlegung des Relationenschemas WEINE (Folie 5-21) in 3NF:

R1(Name, Weingut, Preis)

R2(Name, Farbe)

R3\_1(Weingut, Anbauggebiet)

R3\_2(Anbauggebiet, Region)

mit Schlüsselabhängigkeiten

Name, Weingut  $\rightarrow$  Preis

Name  $\rightarrow$  Farbe

Weingut  $\rightarrow$  Anbauggebiet

Anbauggebiet  $\rightarrow$  Region

- äquivalent zu FDs  $f_1 \dots f_4$  (Folie 5-21)  $\rightsquigarrow$  abhängigkeittreu

## Abhängigkeitstreue: Beispiel /2

- Postleitzahl-Struktur der Deutschen Post

ADRESSE(PLZ (P), Ort (O), Strasse(S), Hausnummer(H))  
und funktionalen Abhängigkeiten  $F$

$$OSH \rightarrow P, P \rightarrow O$$

- Schlüsselkandidaten: OSH und PSH  $\rightsquigarrow$  3NF
- nicht in BCNF (wegen  $PSH \rightarrow P \rightarrow O$ ): daher Zerlegung von ADRESSE
- aber: jede Zerlegung würde  $OSH \rightarrow P$  zerstören
- Menge der sich ergebenden FDs ist nicht äquivalent zu  $F$ , die Zerlegung damit nicht abhängigkeitsstreu

# Abhängigkeitstreue formal

- lokal erweitertes Datenbankschema  $S = \{(R_1, \mathcal{K}_1), \dots, (R_p, \mathcal{K}_p)\}$ ;  
ein Menge  $F$  lokaler Abhängigkeiten

$S$  charakterisiert vollständig  $F$  (oder: ist abhängigkeittreu bezüglich  $F$ ) genau dann, wenn

$$F \equiv \{K \rightarrow R \mid (R, \mathcal{K}) \in S, K \in \mathcal{K}\}$$

# Verbundtreue

- zur Erfüllung des Kriteriums der Normalformen müssen Relationenschemata teilweise in kleinere Relationenschemata zerlegt werden
- für Beschränkung auf „sinnvolle“ Zerlegungen gilt Forderung, dass die Originalrelation wieder aus den zerlegten Relationen mit dem natürlichen Verbund zurückgewonnen werden kann  
↪ **Verbundtreue**

## Verbundtreue: Beispiele

- Zerlegung des Relationenschemas  $R = ABC$  in

$$R_1 = AB \text{ und } R_2 = BC$$

- Dekomposition bei Vorliegen der Abhängigkeiten

$$F = \{A \rightarrow B, C \rightarrow B\}$$

ist nicht verbundtreu

- dagegen bei Vorliegen von

$$F' = \{A \rightarrow B, B \rightarrow C\}$$

verbundtreu

# Verbundtreue Dekomposition

- Originalrelation:

A	B	C
1	2	3
4	2	3

- Dekomposition:

A	B
1	2
4	2

B	C
2	3

- Verbund (verbundtreu):

A	B	C
1	2	3
4	2	3



# Nicht verbundtreue Dekomposition

- Originalrelation:

A	B	C
1	2	3
4	2	5

- Dekomposition:

A	B	B	C
1	2	2	3
4	2	2	5

- Verbund (nicht verbundtreu):

A	B	C
1	2	3
4	2	5
1	2	5
4	2	3

## Verbundtreue formal

Die Dekomposition einer Attributmenge  $X$  in  $X_1, \dots, X_p$  mit  $X = \bigcup_{i=1}^p X_i$  heißt **verbundtreu** ( $\pi$ - $\bowtie$ -treu, lossless) bezüglich einer Menge von Abhängigkeiten  $F$  über  $X$  genau dann, wenn

$$\forall r \in \mathbf{SAT}_X(F) : \pi_{X_1}(r) \bowtie \dots \bowtie \pi_{X_p}(r) = r$$

gilt.

- einfaches Kriterium für Verbundtreue bei Dekomposition in zwei Relationenschemata: Dekomposition von  $X$  in  $X_1$  und  $X_2$  ist verbundtreu bzgl.  $F$ , wenn  $X_1 \cap X_2 \rightarrow X_1 \in F^+$  oder  $X_1 \cap X_2 \rightarrow X_2 \in F^+$

# Transformationseigenschaften

<b>Kennung</b>	<b>Transformationseigenschaft</b>	<b>Kurzcharakteristik</b>
<b>T1</b>	Abhängigkeitstreue	alle gegebenen Abhängigkeiten sind durch Schlüssel repräsentiert
<b>T2</b>	Verbundtreue	Originalrelationen können durch den Verbund der Basisrelationen wiedergewonnen werden

# Weitere Abhängigkeiten

- Mehrwertige Abhängigkeit (kurz: MVD)
  - ▶ innerhalb einer Relation  $r$  wird einem Attributwert von  $X$  eine Menge von  $Y$ -Werten zugeordnet, unabhängig von den Werten der restlichen Attribute  $\rightsquigarrow$  Vierte Normalform
- Verbundabhängigkeit (kurz: JD)
  - ▶  $R$  kann ohne Informationsverlust in  $R_1, \dots, R_p$  aufgetrennt werden:  
 $\bowtie [R_1, \dots, R_p]$
- Inklusionsabhängigkeit (kurz: IND)
  - ▶ auf der rechten Seite einer Fremdschlüsselabhängigkeit nicht unbedingt den Primärschlüssel einer Relation

# Mehrwertige Abhängigkeiten

- Folge der 1NF
- Mehrwertige Abhängigkeiten erzeugen Redundanz:

WEIN_EMPFEHLUNG	WName	Jahrgang	Gericht
	Chardonnay	2002	Geflügel
	Chardonnay	2002	Fisch
	Chardonnay	2003	Fisch
	Chardonnay	2003	Geflügel
	Shiraz	2003	Wild
	Shiraz	2003	Lamm
	Shiraz	2004	Wild
	Shiraz	2004	Lamm

- ▶ eine (oder mehrere) Gruppe von Attributwerten ist von einem Schlüssel bestimmt, unabhängig von anderen Attributen
- ▶ hier: Menge von Jahrgängen plus Menge von Gerichten  
 $WName \twoheadrightarrow Jahrgang, WName \twoheadrightarrow Gericht$
- ▶ Resultat: Redundanz durch Bildung aller Kombinationen

# Mehrwertige Abhängigkeiten formal

- Relation  $r(R)$  mit  $X, Y \subseteq R$ ,  $Z := R - (X \cup Y)$  **genügt** der MVD  $X \twoheadrightarrow Y$  gdw.

$$\begin{aligned} \forall t_1, t_2 \in r : & & [(t_1 \neq t_2 \wedge t_1(X) = t_2(X)) \\ & \implies & \exists t_3 \in r : t_3(X) = t_1(X) \wedge t_3(Y) = t_1(Y) \wedge \\ & & t_3(Z) = t_2(Z)] \end{aligned}$$

- Relation  $r(R)$  mit  $R = XYZ$  und  $X \twoheadrightarrow Y$ :
  - ▶ wenn  $(x_1, y_1, z_1) \in r$  und  $(x_1, y_2, z_2) \in r$
  - ▶ dann auch:  $(x_1, y_1, z_2) \in r$  und  $(x_1, y_2, z_1) \in r$
- Bsp.: wegen ('Chardonnay', 2002, 'Geflügel') und ('Chardonnay', 2003, 'Fisch') müssen auch ('Chardonnay', 2002, 'Fisch') und ('Chardonnay', 2003, 'Geflügel') enthalten sein

# Mehrwertige Abhängigkeiten und 4NF

- wünschenswerte Schemaeigenschaft bei Vorliegen von MVDs:  
**vierte Normalform**
- fordert die Beseitigung derartiger Redundanzen: keine zwei MVDs zwischen Attributen einer Relation
- Beispiel von Folie 5-44 verletzt diese Forderung
- Prinzip
  - ▶ Elimination der rechten Seite einer der beiden mehrwertigen Abhängigkeiten,
  - ▶ linke Seite mit dieser rechten Seite in neue Relation kopiert

# Vierte Normalform

<b>WEIN_JAHR</b>	<b>WName</b>	<b>Jahrgang</b>
	Chardonnay	2002
	Chardonnay	2003
	Shiraz	2003
	Shiraz	2004

<b>WEIN_GERICHT</b>	<b>WName</b>	<b>Gericht</b>
	Chardonnay	Geflügel
	Chardonnay	Fisch
	Shiraz	Wild
	Shiraz	Lamm



## Vierte Normalform formal

- Relationenschema  $R$  mit  $X, Y \subseteq R$ , MVD-Menge  $M$  über  $R$
- MVD  $X \twoheadrightarrow Y$  heißt trivial genau dann, wenn  $Y \subseteq X$  oder  $X \cup Y = R$

erweitertes Relationenschema  $\mathcal{R} = (R, \mathcal{K})$  ist in **vierter Normalform** (4NF) bezüglich  $M$  genau dann, wenn für alle  $X \twoheadrightarrow Y \in M^+$  gilt:

$X \twoheadrightarrow Y$  ist trivial oder  $X \supseteq K$  für ein  $K \in \mathcal{K}$ .

## Nichttriviale MVDs

- Erweiterung der Relation WEIN\_JAHR von Folie 5-47 um Attribute Farbe und Restsüße
- MVD  $wName \twoheadrightarrow Jahrgang$  ist nicht mehr trivial
- Zerlegung:

$WEIN\_JAHR1(\underline{wName}, Jahrgang)$

$WEIN\_JAHR2(\underline{wName}, Farbe, Restsüße)$

# Zusammenfassung

- funktionale Abhängigkeiten
- Normalformen (1NF - 3NF, BCNF)
- Abhängigkeitstreue und Verbundtreue
- Entwurfsverfahren
- mehrwertige Abhängigkeiten

# Kontrollfragen

- Welches Ziel hat die Normalisierung relationaler Schemata?
- Welche Eigenschaften relationaler Schemata werden bei den Normalformen berücksichtigt?
- Was unterscheidet 3NF und BCNF?
- Was fordern Abhängigkeitstreue und Verbundtreue?

