Kap. 3 Relationenmodell mit relationaler Algebra

Kap. 3.1. Trägermenge

Seien $D_1, D_2, ..., D_k$ Domänen:

(Typen, Arten, Sorten, Wertmengen)

z.B. string integer real Boolean DateTime

… BLOB, TIFF-image, HTML-Doc,…

Def 3.1: Relation R ist eine (endliche) Untermenge des Cartesischen Produkts von $D_{1,...,} D_k$,

$R \subseteq D_1 x D_2 x \ldots x D_k$

Hinweis: Spalten nach Positionen (Reihenfolge wichtig) oder Attributnamen

$R \subseteq (A_1 : D_1, A_2 : D_2, \ldots, A_k : D_k)$

A_i eindeutig bzgl. R oder systemweit: $R. A_k$ oder A_k

Sprechweisen: *Tabelle (table) R oder Relation R*

<u>Spalten</u>	<u>Zeilen</u>
Column	Row, Tupel, tuple
feste Anzahl	variable Anzahl
Reihenfolge wichtig	Reihenf. unwichtig, Menge

Relationale Algebra:

Trägermenge: hier $\mathfrak{R}$ = Menge der (endlichen) Relationen über vorgegebener Menge von Domänen (z. B. Datentypen von SQL)

Operatoren:

einstellig: op1: $\mathfrak{R} \to \mathfrak{R}$

zweistellig: op2: $\mathfrak{R} \times \mathfrak{R} \to \mathfrak{R}$

hier: generische, parametrisierte Operatoren

Abschlußeigenschaften: $\Re \rightarrow \Re$

$\Re \times \Re \rightarrow \Re$

d. h. $op1\,(R_1) = R_2 \in \Re$

$R_3\ op2\ R_4 = R_5 \in \Re$

$\Rightarrow$ beliebige Aufschichtung von Ausdrücken der relationalen Algebra, analog zu arithmetischen Ausdrücken der algorithmischen Sprachen

$R_3\ op3\ \mathbf{(}op1\,(R_1)\ \ op2\ R_2\,)$

algebraische Gesetze anwendbar!

Beispiel:

Domänen: $\mathbf{D_1}$ = { rot, rosé, weiß}

$\mathbf{D_2}$ = { Riesling, Sylvaner, Chablis, Shiraz, Spätburgunder, …}

integer

real

$\mathbf{D_3}$ = { Mosel, Ahr, Franken, Nappa…}

Keller $\subseteq$ (Sorte: $\mathbf{D_2}$, Gebiet : $\mathbf{D_3}$, Jahr : **integer,** Farbe:$\mathbf{D_1}$, Größe :**real**)

Stelligkeit = Anzahl Attribute, hier 5

Extension von Keller

Sorte	Gebiet	Jahr	Farbe	Größe
Chablis	Nappa	1991	weiß	0.75
Riesling	Franken	1990	weiß	0.75
Riesling	Mosel	1993	weiß	1.0
Shiraz	Nappa	1992	rot	0.5
		etc.		

Hinweis: funktionale Abhängigkeit zwischen Attributen

Sorte bestimmt Farbe!

Sorte $\rightarrow$ Farbe

Notwendig:

- Relationsbezeichnung, Rel. Name
- Domänen
- Attributnamen
-

Schlüssel von Keller?

Sorte, Gebiet, Jahr, Größe

Prädikative Anfrage

$\{w: w \in$ Keller $\wedge$ w.Gebiet = “Nappa”$\}$

Kardinalität = Anzahl Tupel in R = $|R|$

|Keller| = 4

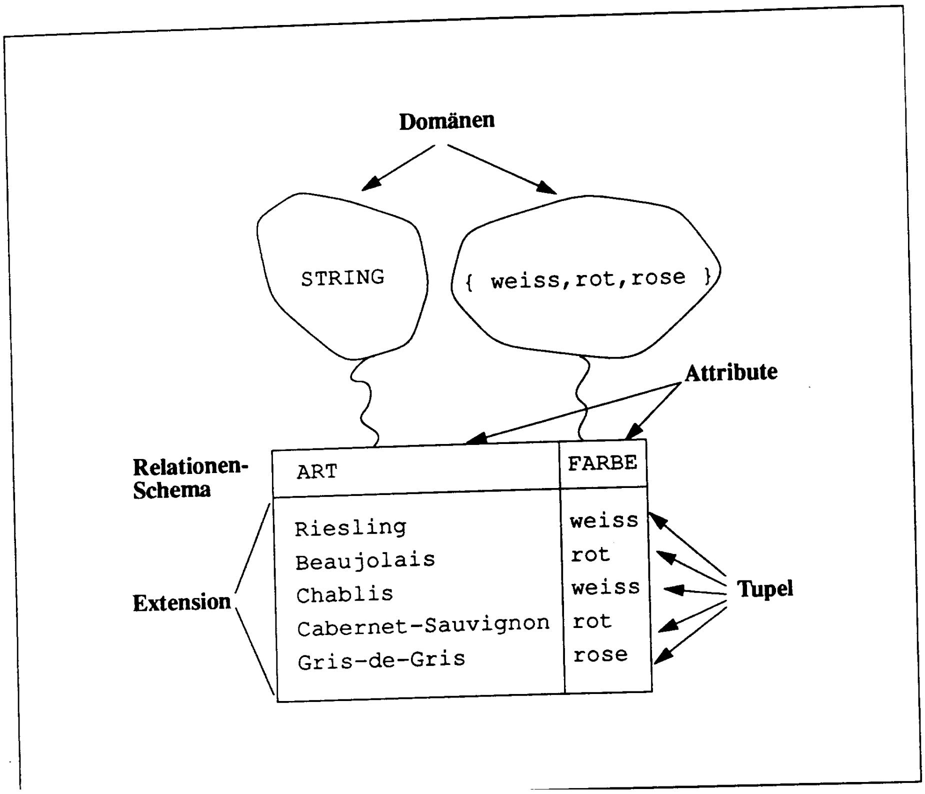
Schritte bei Datenbank-Aufbau:

1. **Domänen** : vorgegeben
2. **Relationsschema** für R : legt Struktur von R
   und seiner Tupel fest, Schema-Definition,
   z. B. Keller
3. **DB – Schema**: Menge von Relations- Schemata
4. **Relationen – Extension**: kurz "Relation"
   Extensions – Aufbau
   Getrennt, einzelne inserts
   Lade-Utilities: per bulk load, z.B. für OMNIS, VD17

   So entsteht eine DB !!

Eigenschaften:

- ***Relation ist eine Menge*** $\Rightarrow$ keine Duplikate
- ***Tupel-Identifizierung*** durch Attributwerte, Schlüssel, Schlüsselkandidaten d.h. wertorientiertes Modell
- Relation ist ***ungeordnet*** (Menge)
- ***kein first, next, eof,*** siehe cursor Konzept später!
- ***Tupel als Abbildung:*** keine Ordnung der Attribute (theoretische Eigenschaft, praktisch wird Ordnung verwendet)
- ***atomare Attributwerte:*** Verallgemeinerung durch beliebige Schachtelungen und Listen der Attribute führt
  - zu **NF^2**-DB (**n**on **f**irst **n**ormal **f**orm) oder
  - zu **oo**-DB (**o**bjekt **o**rientierte DB)



Kap. 3.2 Relationale Operationen

Relationen: R, S, T

Mengenoperationen: $\cup$ $\cap$ ****
für artvertägliche Relationen

R1 (*A1*:**string**, *A2*:**integer**, *A3*:**real**)
$\cup$ R2 (" " ")

oder R1 $\cap$ R2 R1 \ R2

Komplement nicht sinnvoll, da keine universelle Relation vorhanden, wäre nur intensional darstellbar, wäre aber meist unendlich.

Hinweis: Duplikate von Tupeln sind bei $\cup$ zu eliminieren z. B.

a	1
b	2
a	1

ist keine erlaubte Relation, “Tabelle” ungenau!

Weitere Operationen:

- Projektion π
- Selektion σ
- Catersisches Produkt x
- Equijoin ε
- Naturaljoin ɛ
- Allg. Verbund ν

Projektion π:

R =

a	1	1
b	2	4
a	1	3

streichen einer oder mehrerer Spalten,
Duplikate eliminieren!

R' =

a	1
b	2
a	1

R'' =

a	1
b	2

Schreibweise mit Attributen: A_1, A_2, A_3: $\pi_{A1,A2}$

=> π ist generische, parametrisierte Operation, d.h. unendliche Familie

Restriktion, Selektion σ

$R = t_1$ mit Tupeln $t_1, ...t_n$

t_2

: $|R| = n$

t_n

$\sigma_{P(t)} R := \{ t \mid t \in R \wedge P(t) \}$

z. B. $\sigma_{t.A1=\text{a}} (\text{R})$ **=**

a	1	1
a	1	3

Tupel-Variable *t*, läuft über *R, R* ist ***Werte-, Individuenbereich von t***
Duplikate?

$\sigma_{P(t)}$ ist generische Operation parametrisiert mit P(t),
P(t) heißt auch ***Restriktions-, Selektionsprädikat***!

Cartesisches Produkt x:

$R \times S := \{ r \times s \mid r \in R \wedge s \in S\}$

r x s ist Tupel- Konkatenation

$|R \times S| = |R| \cdot |S|$ Duplikate?

Equi- Join: $\varepsilon_{i,j}$

Cart. Produkt, aber *i* - tes Attribut von *R* mit *j*-tem Atr. von *S* matchen:

$\sigma_{R.A_i = S.A_j} (R \times S) =: R\,\varepsilon_{i,j}\,S$

Fazit:

$\Rightarrow$ relationale Operationen sind nicht minimal,

$\varepsilon = \sigma \circ x$

$\Rightarrow$ Ergebnis von $\varepsilon_{i,j}$ ist redundant,
2 gleiche Spalten, $S.A_j$ streichen!
Duplikate?

Natural-Join: 2

$R 2_{i,j} S \quad = \quad R 2_{R.A_i = S.B_j} S \quad =_{df}$

$\pi_{R.A1,\ldots R.An,\ S.B1,\ldots \mathbf{S.Bj-1},\ \mathbf{S.Bj+1},\ldots S.Bm} (R\varepsilon_{i,j} S)$

es ist üblich, Spalte in 2. Relation zu streichen.

Allgemeiner Verbund: $\nu_{P(R,S)}$

$R \nu_{P(r,s)} S = \sigma_{P(r,s)} (RxS)$

df

$= \{ r \times s \mid r \in R \wedge s \in S \wedge P(r,s) \}$

wobei *P(r,s)* über die Attribute von *R* und *S* definiert ist, z.B.

$P(r,s) \equiv r.i \;\varphi\; s.j$

mit $\varphi \in \{<, \leq, \neq, \geq, >\}$

Division: ***R/S***

$R/S = \{ t \mid \{t\} \times S \subseteq R \}$

$= \{ t \mid \forall s \in S : t \times s \in R \}$ *z.B.*

R =

f	*b*	*c*
f	*d*	*e*
g	*c*	*b*
h	*b*	*c*
h	*d*	*e*

S =

b	*c*
d	*e*

R/S =

f
h

Gesetze: (R x S) / S = R

aber i.a. (R / S) x S $\neq$ R

analog zu integer Division und Multiplikation

Semi-Join:

$\pi_{\text{Attr. von R}} (R\ \nu_{P(r,s)}\ S)$

Join Partner in *S* muß existieren,
aber Attribute von *S* nicht interessant

Duplikate?

Äußerer Verbund:

Ziel: Es sollen keine Tupel durch Verbund „verloren“ gehen, „lossless“ Joins.

Def: Sei $\perp_R$, $\perp_S$ das Tupel, das für alle Attr. von R bzw. S den Wert „undefiniert“ ***(null)*** hat.

Def: $R ⟗_{P(r,s)} S :=$

$$R\nu_{P(r.s)}\ S \cup$$
$$\{r \times \perp_S \mid \neg \exists\ s \in S : P(r,s)\} \cup \qquad (1)$$
$$\{\perp_R \times s \mid \neg \exists\ r \in R : P(r,s)\} \qquad (2)$$

Def: linker und recher äußerer Verbund

linker: $R ⟕_{P(r,s)} S :=$ $R\nu_{P(r,s)}\ S \cup \{r \times \perp_S \mid \neg \exists\ s \in S : P(r,s)\}$ (1)

rechter: $R ⟖_{P(r,s)} S :=$ $R\nu_{P(r,s)}\ S \cup \{\perp_R \times s \mid \neg \exists\ r \in R : P(r,s)\}$ (2)

Gesetze für äußere Verbünde:

$$\pi_{\text{Attr. von R}}\,(R \;⟗_{P(r,s)}\; S) \;:=\; R$$

$$\pi_{\text{Attr. von S}}\,(R \;⟗_{P(r,s)}\; S) \;:=\; S$$

$$\pi_{\text{Attr. von R}}\,(R \;⟕_{P(r,s)}\; S) \;:=\; R$$

$$\pi_{\text{Attr. von S}}\,(R \;⟖_{P(r,s)}\; S) \;:=\; S$$

3.2.7

Beispiele zum äußeren Gleichverbund

- Gleichverbund

R	A	B	C
	a_1	b_1	c_1
	a_2	b_2	c_2

$⋈$

S	C	D	E
	c_1	d_1	c_1
	c_3	d_2	c_2

=

ERG	A	B	C	D	E
	a_1	b_1	c_1	d_1	e_1

- Linker äußerer Gleichverbund

R	A	B	C
	a_1	b_1	c_1
	a_2	b_2	c_2

$⟕$

S	C	D	E
	c_1	d_1	c_1
	c_3	d_2	c_2

=

ERG	A	B	C	D	E
	a_1	b_1	c_1	d_1	e_1
	a_2	b_2	c_2	--	--

- Rechter äußerer Gleichverbund

R	A	B	C
	a_1	b_1	c_1
	a_2	b_2	c_2

$⟖$

S	C	D	E
	c_1	d_1	c_1
	c_3	d_2	c_2

=

ERG	A	B	C	D	E
	a_1	b_1	c_1	d_1	e_1
	--	--	c_3	d_2	e_2

- äußerer Gleichverbund

R	A	B	C
	a_1	b_1	c_1
	a_2	b_2	c_2

$⟗$

S	C	D	E
	c_1	d_1	c_1
	c_3	d_2	c_2

=

ERG	A	B	C	D	E
	a_1	b_1	c_1	d_1	e_1
	a_2	b_2	c_2	--	--
	--	--	c_3	d_2	e_2

4 - 54