

4. Relationenalgebra

- Einleitung
- Selektion und Projektion
- Mengenoperatoren
 - Vereinigung, Durchschnitt, Differenz
 - kartesisches Produkt
- Verbundoperationen (Join)
 - Theta-Join
 - natürlicher Verbund
 - Semi-Join
 - äußerer Verbund
- Division
- Beispielanfragen



Sprachen für das Relationenmodell

- Datenmodell = Datenobjekte + Operatoren
- im RM wird vereinheitlichte Sprache angestrebt für:
 - Anfragen (Queries) im 'Stand-Alone'-Modus
 - Datenmanipulation und Anfragen eingebettet in eine Wirtssprache
 - Datendefinition
 - Zugriffs- und Integritätskontrolle
 - Unterstützung verschiedener Benutzerklassen:
Anwendungsprogrammierer, DBA, gelegentliche Benutzer
- verschiedene Grundtypen von Sprachen
 - formale Ansätze: Relationenalgebra und Relationenkalkül
 - abbildungsorientierte Sprachen (z. B. SQL)
 - graphik-orientierte Sprachen (z. B. Query-by-Example)



Relationenalgebra

- **Algebra**: ein System, das aus einer nichtleeren Menge und einer Familie von Operationen besteht

- Relationen sind Mengen
- Operationen auf Relationen arbeiten auf einer oder mehreren Relationen als Eingabe und erzeugen eine Relation als Ausgabe (Abgeschlossenheitseigenschaft)
=> mengenorientierte Operationen

- **Operationen:**

Klassische Mengenoperationen:

- Vereinigung
- Differenz
- kartesisches Produkt
- Durchschnitt (ableitbar)

Relationenoperationen:

- Restriktion (Selektion)
- Projektion
- Verbund (Join) (ableitbar)
- Division (ableitbar)

Selektion (Restriktion)

- Auswahl von Zeilen einer Relation über Prädikate, abgekürzt σ_P

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

P = log. Formel (ohne Quantoren !) zusammengestellt aus:

- Operanden: Attributnamen oder Konstanten
- Vergleichsoperatoren $\theta \in \{ <, =, >, \leq, \neq, \geq \}$
- logische Operatoren: \vee, \wedge, \neg

- **Beispiele:**

$$\sigma_{\text{GEHALT} < \text{PROVISION}} (\text{PERS})$$

$$\sigma_{\text{BERUF} = \text{'Programmierer'} \wedge \text{ALTER} < 50} (\text{PERS})$$

- **Eigenschaften**

- $\text{grad}(\sigma_P(R)) = \text{grad}(R)$
- $\text{card}(\sigma_P(R)) \leq \text{card}(R)$

Projektion

- Auswahl der Spalten (Attribute) A_1, A_2, \dots, A_k aus einer Relation R (Grad $n \geq k$)

$$\pi_{A_1, A_2, \dots, A_k}(R) = \{p \mid \exists t \in R : p = \langle t[A_1], \dots, t[A_k] \rangle\}$$

- Beispiel:

$$\pi_{\text{NAME, GEHALT}}(\text{PERS})$$

- Eigenschaften:

- wichtig: Duplikate werden entfernt ! (Mengeneigenschaft)
- $\text{grad}(\pi_A(R)) \leq \text{grad}(R)$
- $\text{card}(\pi_A(R)) \leq \text{card}(R)$

Relationenalgebra: Beispiel-DB

ABT

<u>ANR</u>	ANAME	ORT
K51	Planung	Leipzig
K53	Einkauf	Frankfurt
K55	Vertrieb	Frankfurt

PERS

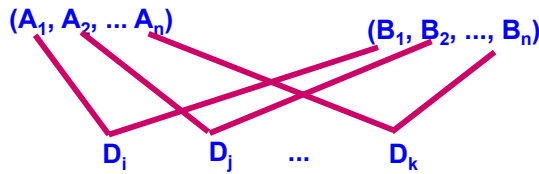
<u>PNR</u>	Name	Alter	Gehalt	ANR	MNR
406	Abel	47	50700	K55	123
123	Schulz	32	43500	K51	-
829	Müller	36	40200	K53	406
574	Schmid	28	36000	K55	123

- Finde alle Angestellten aus Abteilung K55, die mehr als 40.000 verdienen
- Finde alle Abteilungsorte
- Finde den Abteilungsnamen von Abteilung K53

Klassische Mengenoperationen

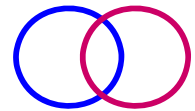
- Voraussetzung: *Vereinigungsverträglichkeit* der beteiligten Relationen:

gleicher Grad - gleiche Bereiche: $\Rightarrow W(A_i) = W(B_i) \quad : \quad i = 1, n$



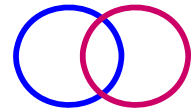
- Vereinigung: $R \cup S = \{ t \mid t \in R \vee t \in S \}$

– $\text{card}(R \cup S) \leq \text{card}(R) + \text{card}(S)$



- Differenz: $R - S = \{ t \mid t \in R \wedge t \notin S \}$

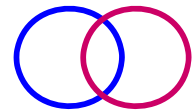
– $\text{card}(R - S) \leq \text{card}(R)$



- Durchschnitt:

$$R \cap S = R - (R - S) = \{ t \mid t \in R \wedge t \in S \}$$

– $\text{card}(R \cap S) \leq \min(\text{card}(R), \text{card}(S))$



(Erweitertes) Kartesisches Produkt

R (Grad r) und S (Grad s) beliebig

$$R \times S = \{ k \mid \exists x \in R, y \in S : k = x \mid y \}$$

Beachte: $k = x \mid y = \langle x_1, \dots, x_r, y_1, \dots, y_s \rangle$

nicht $\langle \langle x_1, \dots, x_r \rangle, \langle y_1, \dots, y_s \rangle \rangle$ wie übliches kart. Produkt

– $\text{grad}(R \times S) = \text{grad}(R) + \text{grad}(S)$; $\text{card}(R \times S) = \text{card}(R) * \text{card}(S)$

Beispiel

R		
A	B	C
a	γ	1
d	α	2
b	β	3

S		
D	E	F
b	γ	3
d	α	2

R × S

Allgemeiner Verbund (Theta-Join)

■ grob:

- kartesisches Produkt zwischen zwei Relationen R (Grad r) und S (Grad s).
- eingeschränkt durch Θ -Bedingungen zwischen Attribut A von R und Attribut B von S.

■ Θ -Verbund zwischen R und S:

$$R \bowtie_{A \Theta B} S = \sigma_{A \Theta B} (R \times S)$$

mit arithm. Vergleichsoperator $\Theta \in \{<, =, >, \leq, \neq, \geq\}$

■ Bemerkungen:

- *Gleichverbund (Equijoin)*: $\Theta = '='$:

Natürlicher Verbund (Natural Join)

■ grob: Gleichverbund über alle gleichnamigen Attribute und Projektion über die verschiedenen Attribute

■ natürlicher Verbund zwischen R und S:

gegeben: $R (A_1, A_2, \dots, A_{r-j+1}, \dots, A_r), \quad S (B_1, B_2, \dots, B_j, \dots, B_s)$

o.B.d.A.:(sonst. Umsortierung: $B_1 = A_{r-j+1}, B_2 = A_{r-j+2} \dots B_j = A_r$)

$$R \bowtie S = \pi_{A_1, \dots, A_r, B_{j+1}, \dots, B_s} \sigma_{(R.A_{r-j+1} = S.B_1) \wedge \dots \wedge (R.A_r = S.B_j)} (R \times S)$$

\bowtie Zeichen für Natural Join $\Rightarrow \Theta = '='$

- Join-Attribute sind durch Übereinstimmungsbedingung gegeben

R		
A	B	C
a ₁	b ₁	c ₁
a ₂	b ₂	c ₂



S		
C	D	E
c ₁	d ₁	e ₁
c ₃	d ₂	e ₂

=

Resultat				
A	B	C	D	E
a ₁	b ₁	c ₁	d ₁	e ₁

Join-Beispiel

ABT		
<u>ANR</u>	ANAME	ORT
K51	Planung	Leipzig
K53	Einkauf	Frankfurt
K55	Vertrieb	Frankfurt

PERS					
<u>PNR</u>	Name	Alter	Gehalt	ANR	MNR
406	Abel	47	50700	K55	123
123	Schulz	32	43500	K51	-
829	Müller	36	40200	K53	406
574	Schmid	28	36000	K55	123

- Finde alle Angestellten (PNR, ALTER, ANAME), die in einer Abteilung in Frankfurt arbeiten und älter als 30 sind.

Semi-Join

- Ergebnisbeschränkung des Gleichverbundes auf eine der beiden Eingaberelationen

$$S \bowtie R = \pi_{S\text{-Attribute}} (S \bowtie R)$$

$$S \ltimes R = \pi_{R\text{-Attribute}} (S \bowtie R)$$

S			⋈	R			=	Resultat		
A	B	C		C	D	E		A	B	C
a ₁	b ₁	c ₁		c ₁	d ₁	e ₁	a ₁	b ₁	c ₁	
a ₂	b ₂	c ₂		c ₃	d ₂	e ₂				

S			⋈	R			=	Resultat		
A	B	C		C	D	E		C	D	E
a ₁	b ₁	c ₁		c ₁	d ₁	e ₁	c ₁	d ₁	e ₁	
a ₂	b ₂	c ₂		c ₃	d ₂	e ₂				

Äußerer Verbund (Outer Join)

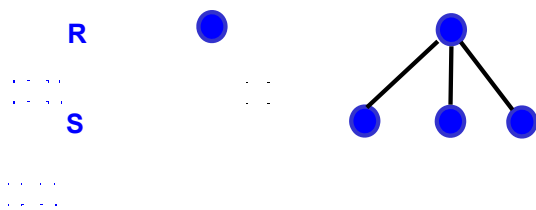
■ Ziel: Verlustfreier Verbund soll erzwungen werden

■ ein Gleichverbund zwischen R und S heißt *verlustfrei*, wenn alle Tupel von R und S am Verbund teilnehmen. Die inverse Operation Projektion erzeugt dann wieder R und S (**lossless join**).

$$R \bowtie S \text{ verlustfrei} \Leftrightarrow$$

■ bisher: $R \bowtie S$ liefert nur „vollständige Objekte“

– es sollen aber auch Teilobjekte als Ergebnis geliefert werden (z. B. komplexe Objekte)



– Trick: Einfügen künstlicher Verbundpartner

Outer Join (2)

■ Def.: Seien A die Verbundattribute, $\{\equiv\}$ der undefinierte Wert und
 $R' := R \cup ((\pi_A(S) - \pi_A(R)) \times \{\equiv\} \times \dots \times \{\equiv\})$
 $S' := S \cup ((\pi_A(R) - \pi_A(S)) \times \{\equiv\} \times \dots \times \{\equiv\})$

Äußerer Gleichverbund

$$R \bowtie_{R.A=S.A} S := R' \bowtie_{R'.A=S'.A} S'$$

Äußerer natürlicher Verbund

$$R \bowtie_{R.A=S.A} S := R' \bowtie S'$$

■ linker und rechter äußerer Gleichverbund

– nur die linke bzw. rechte Argumentrelation bleibt verlustfrei (Einfügen künstlicher Verbundpartner in rechter bzw. linker Eingaberelation)

$$R \bowtie_{R.A=S.A} S := R \bowtie_{R.A=S'.A} S'$$

Linker äußerer Gleichverbund

$$R \bowtie_{R.A=S.A} S := R' \bowtie_{R'.A=S.A} S$$

Rechter äußerer Gleichverbund

■ Verallgemeinerung auf 2 (oder mehr) Joins, z.B. $R \bowtie S \bowtie T$

– selbst isolierte Tupel können zu einem vollständigen Pfad expandiert werden

Outer Join - Beispiel



PERS

PNR	ANR ...
P1	A1
P2	A1
P3	A2
P4	-
P5	-

PERS ⋈ ABT

PNR	ANR	ANAME ...

PERS ⋈_{ABT} ABT

PNR	ANR	ANAME ...

PERS ⋈_{ABT} ABT

PNR	ANR	ANAME ...

PNR	ANR	ANAME ...

PERS ⋈_{ABT} ABT

4 - 15



Division

- Beantwortung von Fragen, bei denen eine „ganze Relation“ zur Qualifikation herangezogen wird
- Simulation des Allquantors \Rightarrow ein Tupel aus R steht mit allen Tupeln aus S in einer bestimmten Beziehung

Definition

Voraussetzung: S-Attribute \subset R-Attribute

sei R vom Grad r und S vom Grad s, $r > s$

t sei (r-s)-Tupel, u sei s-Tupel;

dann gilt: $R \div S = \{t \mid \forall u \in S : tu \in R\}$

grad (R \div S) =

card (R \div S)



Division (2)

■ Beispiel

LNR	PNR	TNR
L1	P1	T1
L1	P2	T1
L2	P1	T1
L2	P1	T2
L2	P2	T1

÷

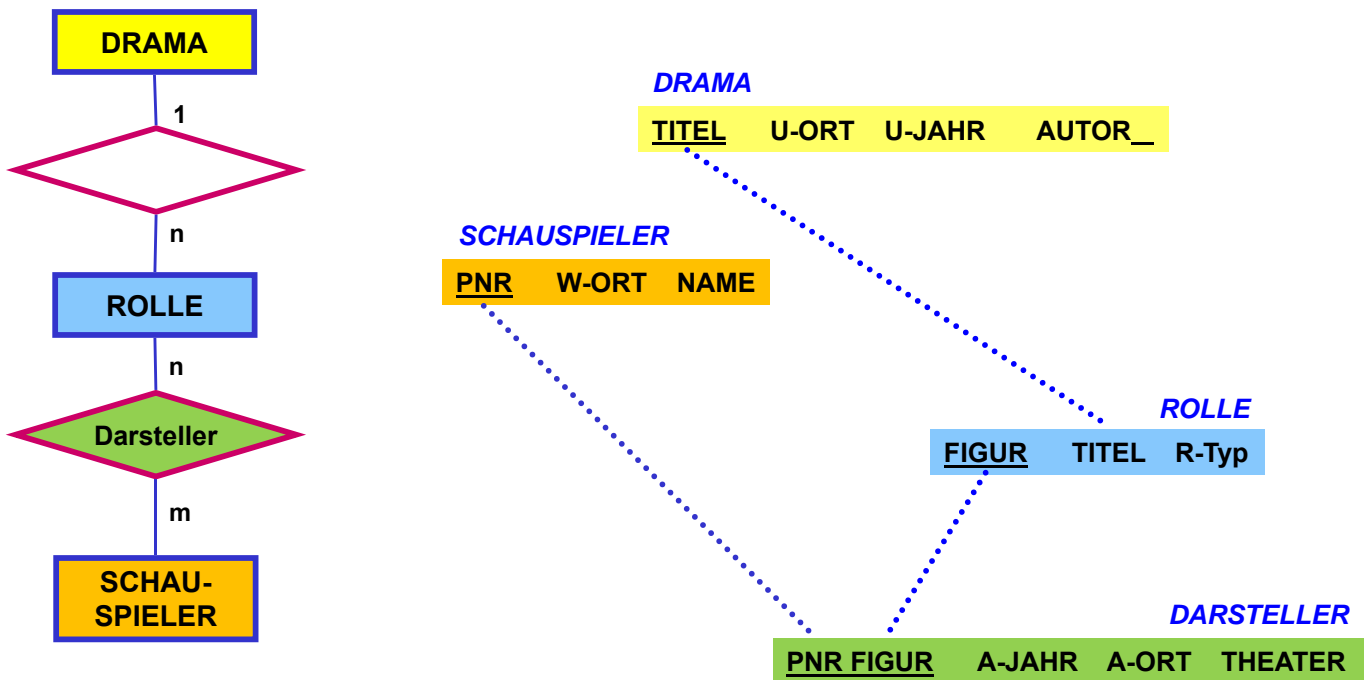
PNR	TNR
P1	T1
P1	T2
P2	T1

- welche Lieferanten beliefern alle Projekte?
- welche Lieferanten liefern alle Teile?

■ Zusammenhang zwischen Division und kartesischem Produkt: $(R \times S) \div S = R$



Beispiel-DB: Bühne



Beispielfragen

- Welche Darsteller (PNR) haben im Schauspielhaus gespielt?
- Finde alle Schauspieler (NAME, W-ORT), die einmal im 'Faust' mitgespielt haben.
- Finde alle Schauspieler (NAME), die bei in Weimar uraufgeführten Dramen an ihrem Wohnort als 'Held' mitgespielt haben
- Finde die Schauspieler (PNR), die nie gespielt haben
- Finde alle Schauspieler (NAME), die alle Rollen gespielt haben.

Zusammenfassung Relationenalgebra

- saubere mathematische Definition
- mengenorientierte Operationen
- keine Änderungsoperationen!
- für Laien nicht leicht verständlich

