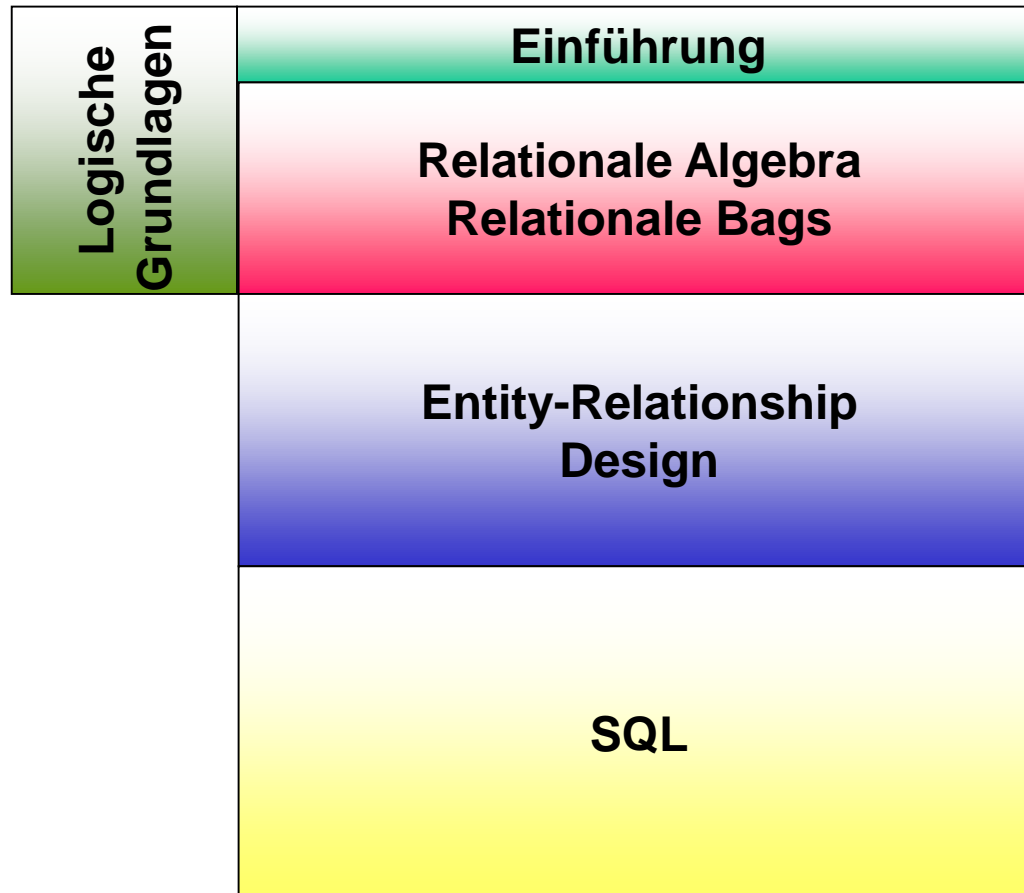


DAB1 – Datenbanken 1

Dr. Daniel Aebi (aebd@zhaw.ch)

Lektion 3: Relationale Algebra

Wo stehen wir?



← "You are here"

Rückblick

- Datenarten:
 - Strukturiert
 - Unstrukturiert
 - Semi-strukturiert
- Erfinder des relationalen Datenmodells: Codd (Turing-Award-Winner)
- Begriffe des Relationenmodelles:
 - Attribut: Name / Domäne (Wertebereich)
 - n-Tupel: Menge von Attributen (n sachlich zusammengehörige Attribute)
 - Relation: Menge von gleich strukturierten Tupeln
- Operationen (entfernend, unär):
 - σ : Selektion
 - π : Projektion

Lernziele Lektion 3

- Folgende (binäre) Operationen der relationalen Algebra verstehen:
 - \cup Vereinigung
 - \cap Durchschnitt
 - \setminus Differenz
 - \times Kreuzprodukt (kartesisches Produkt)
 - \bowtie Verbundvarianten
- Folgende (unäre) Operation der relationalen Algebra verstehen:
 - ρ Umbenennung

Bemerkungen zur Notation

- Bisher eingeführt:
 - $R(A,B,C)$: **Relationenformat** (gelegentlich auch Relationenschema oder Relationenvariable genannt)
 - $r \in R$: **Tupelvariable**, zu R gehörige Relation
 - $\text{dom}(A)$: Domäne, Wertebereich für die Werte des Attributes A
 - $\text{ext}(R) \subseteq \text{dom}(A) \times \text{dom}(B) \times \text{dom}(C)$
 - Kurzschreibweise: $\text{dom}(R) = \text{dom}(A) \times \text{dom}(B) \times \text{dom}(C)$

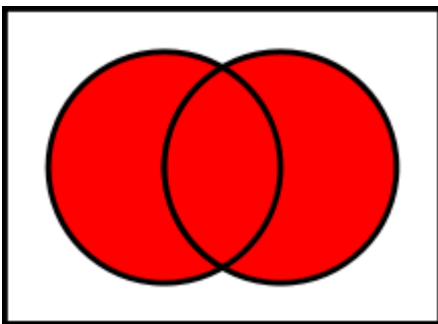
Symbole der relationalen Algebra

Übersicht der Operationen:

- σ Selektion
- π Projektion
- \cup Mengenvereinigung
- \cap Mengendurchschnitt
- \setminus Mengendifferenz
- \times Kreuzprodukt
- \bowtie Join (Verbund)
- \div Division
- ρ Umbenennung
- Linker äusserer Verbund (Symbol, oft weggelassen: \bowtie)
- Rechter äusserer Verbund (Symbol, oft weggelassen: \bowtie)
- Voller äusserer Verbund (Symbol, oft weggelassen: \bowtie)

Mengenoperator: Vereinigung, \cup

- Sammelt Elemente (Tupel) zweier Relationen unter einem gemeinsamen Format auf.
- Attributmengen (Formate) beider Relationen müssen identisch sein.
 - Namen, Typen
 - Zur Not: Umbenennung
- Ein Element ist nur einmal in $(R \cup S)$ vertreten, auch wenn es jeweils einmal in R und S auftaucht: **Duplikatentfernung**.



Mengenoperator: Vereinigung, \cup

R			
Name	Adresse	Geschlecht	Geburt
Carrie Fisher	123 Maple St., Hollywood	F	9/9/99
Mark Hamill	456 Oak Rd., Brentwood	M	8/8/88

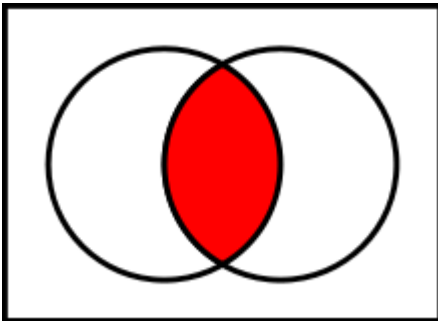
S			
Name	Adresse	Geschlecht	Geburt
Carrie Fisher	123 Maple St., Hollywood	F	9/9/99
Harrison Ford	789 Palm Dr., Beverly Hills	M	7/7/77

$R \cup S$

Name	Adresse	Geschlecht	Geburt
Carrie Fisher	123 Maple St., Hollywood	F	9/9/99
Mark Hamill	456 Oak Rd., Brentwood	M	8/8/88
Harrison Ford	789 Palm Dr., Beverly Hills	M	7/7/77

Mengenoperator: Durchschnitt, \cap

- Durchschnitt $R \cap S$ ergibt die Tupel, die in beiden Relationen gemeinsam vorkommen.
- Attributmengen (Schemas) beider Relationen müssen identisch sein.
 - Namen, Typen
 - Zur Not: Umbenennung
- Ein Element ist nur einmal in $(R \cap S)$ vertreten, auch wenn es jeweils einmal in R und S auftaucht: **Duplikatentfernung**.



Mengenoperator: Durchschnitt, \cap

R			
Name	Adresse	Geschlecht	Geburt
Carrie Fisher	123 Maple St., Hollywood	F	9/9/99
Mark Hamill	456 Oak Rd., Brentwood	M	8/8/88

S			
Name	Adresse	Geschlecht	Geburt
Carrie Fisher	123 Maple St., Hollywood	F	9/9/99
Harrison Ford	789 Palm Dr., Beverly Hills	M	7/7/77

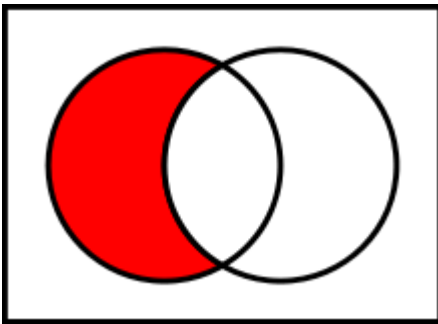
$R \cap S$

Name	Adresse	Geschlecht	Geburt
Carrie Fisher	123 Maple St., Hollywood	F	9/9/99

© F.Naumann, 2011

Mengenoperator: Differenz, \setminus oder $-$

- Differenz $R \setminus S$ (oder $R - S$) eliminiert die Tupel aus der ersten Relation, die auch in der zweiten Relation vorkommen.
- Attributmengen (Schemas) beider Relationen müssen identisch sein.
 - Namen, Typen
 - Zur Not: Umbenennung
- Achtung: $R \setminus S \neq S \setminus R$



Mengenoperator: Differenz, \ oder -

R			
Name	Adresse	Geschlecht	Geburt
Carrie Fisher	123 Maple St., Hollywood	F	9/9/99
Mark Hamill	456 Oak Rd., Brentwood	M	8/8/88

S			
Name	Adresse	Geschlecht	Geburt
Carrie Fisher	123 Maple St., Hollywood	F	9/9/99
Harrison Ford	789 Palm Dr., Beverly Hills	M	7/7/77

R - S

Name	Adresse	Geschlecht	Geburt
Mark Hamill	456 Oak Rd., Brentwood	M	8/8/88

Vereinigung, Durchschnitt, Differenz

- Da Relationen Mengen sind, sind auch die üblichen Mengenoperationen definiert
- Sind r_1 und r_2 Relationen zum Relationenformat R , so gilt:
 - **Vereinigung** (union):
$$r_1 \cup r_2 := \{t \in \text{dom}(R) \mid t \in r_1 \vee t \in r_2\}$$
 - **Durchschnitt** (intersection):
$$r_1 \cap r_2 := \{t \in \text{dom}(R) \mid t \in r_1 \wedge t \in r_2\}$$
 - **Differenz** (difference) $R \setminus S$:
$$r_1 \setminus r_2 := \{t \in \text{dom}(R) \mid t \in r_1 \wedge t \notin r_2\}$$

Vereinigungskompatibilität

r

StudentName	DateOfBirth
Max	20.3.1992
Sandra	2.5.1994
Peter	1.10.1990

s

DateOfBirth	StudentName
2.5.1991	Kevin
6.8.1991	Jessica
2.5.1994	Sandra

$$t = r \cap s = ?$$

In Algebra berechenbar (da Attributreihenfolge irrelevant)

In SQL nicht vereinigungskompatibel

(Domains stimmen nicht überein)
→ nicht definiert

Vereinigungskompatibilität

r

StudentName	Klasse
Max	It05a
Sandra	It05a
Peter	It05b

s

Klasse	StudentName
jo05	Kevin
bo05c	Jessica
it05a	Sandra

$$t = r \cap s = ?$$

In Algebra berechenbar

In SQL nicht vereinigungskompatibel

(Achtung: falls StudentName und Klasse einfach als Zeichenketten definiert sind, sind r und s in SQL sehr wohl vereinigungskompatibel!
Die Schnittmenge ist dann leer, und das Resultat wahrscheinlich ungewollt)

Vereinigungskompatibilität

StudentName	Klasse
Max	It05a
Sandra	It05a
Peter	it05b

$$t = r \setminus s = ?$$

Nicht vereinigungskompatibel

(Anzahl Attribute stimmen nicht überein)
→ nicht definiert

StudentName	Klasse	DateOfBirth	StudentsFee
Peter	It05b	1.10.1990	500.00
Sandra	It05a	2.5.1994	250.00
Max	It05a	20.3.1992	500.00

Kombinierend: (kartesisches) Produkt, \times

- Kreuzprodukt zweier Relationen R und S ist die **Menge aller Tupel**, die man erhält, wenn man **jedes Tupel aus R mit jedem Tupel aus S** „paart“.
- Format hat ein Attribut für jedes Attribut aus R und S.
- Achtung: Bei Namensgleichheit wird kein Attribut ausgelassen, stattdessen: Umbenennen oder Qualifizieren mit Formatnamen.

Kombinierend: (kartesisches) Produkt, \times

R

A	B
1	2
3	4

S

B	C	D
2	5	6
4	7	8
9	10	11

R \times S

A	R.B	S.B	C	D
1	2	2	5	6
1	2	4	7	8
1	2	9	10	11
3	4	2	5	6
3	4	4	7	8
3	4	9	10	11

Kombinierend: Verbund, join, ⋈

- Motivation: Statt im Kreuzprodukt alle Paare zu bilden, sollen nur die Tupelpaare gebildet werden, deren Tupel irgendwie **übereinstimmen**.
- Diverse Varianten vorhanden:
 - «natural» join: **Übereinstimmung in allen gemeinsamen Attributen**.
 - Gegebenenfalls Umbenennung
 - Schema: Vereinigung der beiden Attributmengen:



Kombinierend: Verbund, join, \bowtie

R	A	B
	1	2
	3	4

S	B	C	D
	2	5	6
	4	7	8
	9	10	11

R ⋈ S	A	B	C	D
	1	2	5	6
	3	4	7	8

- Mehr als ein gemeinsames Attribut: Partner müssen in **allen** Attributwerten übereinstimmen

R	A	B	C
	1	2	3
	6	7	8
	9	7	8

S	B	C	D
	2	5	6
	2	3	5
	7	8	10

R ⋈ S	A	B	C	D
	1	2	3	5
	6	7	8	10
	9	7	8	10

Beispiel: (natürlicher) Verbund

StudentName	DateOfBirth	StudentsFee	StudentName	Klasse
Peter	1.10.1980	500.00	Max	IT1A
Sandra	2.5.1984	250.00	Sandra	IT1A
Max	20.3.1982	500.00	Peter	IT1B

StudentName	DateOfBirth	StudentsFee	Klasse
Peter	1.10.1980	500.00	IT1B
Sandra	2.5.1984	250.00	IT1A
Max	20.3.1982	500.00	IT1A

Verbundvarianten: Natürlicher Verbund

- **Natürlicher Verbund** (natural join):
- Gegeben seien: $R(A_1, \dots, A_m, B_1, \dots, B_k)$ und $S(B_1, \dots, B_k, C_1, \dots, C_n)$
- Dann gilt:

$$R \bowtie S = \pi_{A_1, \dots, A_m, R.B_1, \dots, R.B_k, C_1, \dots, C_n}(\sigma_{R.B_1 = S.B_1 \wedge \dots \wedge R.B_k = S.B_k}(R \times S))$$

$R \bowtie S$											
$R - S$				$R \cap S$				$S - R$			
A_1	A_2	...	A_m	B_1	B_2	...	B_k	C_1	C_2	...	C_n

Beispiel: (natürlicher) Verbund

Aufgabe:

- Geg. Relationenformate $R(A,B,C,D)$ und $S(B,D,E)$. Attribute mit gleichen Bezeichnungen seien in beiden Formaten dieselben

Relationen:

$r = \{ \langle 1,2,3,4 \rangle, \langle 5,6,7,8 \rangle \}$ zum Format R

$s = \{ \langle 2,4,0 \rangle, \langle 2,4,1 \rangle, \langle 2,4,3 \rangle, \langle 6,7,8 \rangle \}$ zum Format S

Berechne $r \bowtie s$

Berechne $r \bowtie \pi_{B,D}(s)$

Beispiel: (natürlicher) Verbund

Lösung:

$r \bowtie s = \{ \langle 1,2,3,4,0 \rangle, \langle 1,2,3,4,1 \rangle, \langle 1,2,3,4,3 \rangle \}$, Format (A,B,C,D,E)

$r \bowtie \pi_{B,D}(s) = \{ \langle 1,2,3,4 \rangle \}$, Relation zum Format R.

Beispiel: (natürlicher) Verbund

- Geg. Relationenformate $R(A,B)$ und $S(C,D)$. Attribute mit gleichen Bezeichnungen seien in beiden Formaten dieselben.
- Relationen $r = \{ \langle 1,2 \rangle, \langle 3,4 \rangle \}$, $s = \{ \langle 1,2 \rangle, \langle 0,3 \rangle, \langle 1,5 \rangle \}$

Berechne $r \bowtie s$

- Wir haben **keine gleichnamigen** Attribute, welche gepaart werden können
→ das Resultat wird zum **Kreuzprodukt** $r \times s$:

$$\text{Res}(A, B, C, D) = \{ \langle 1, 2, 1, 2 \rangle, \langle 1, 2, 0, 3 \rangle, \langle 1, 2, 1, 5 \rangle, \\ \langle 3, 4, 1, 2 \rangle, \langle 3, 4, 0, 3 \rangle, \langle 3, 4, 1, 5 \rangle \}$$

Kombinierend: Theta-join, \bowtie_{θ}

- **Verallgemeinerung** des natürlichen Joins
- Verknüpfungsbedingung kann **frei** gestaltet werden
- Konstruktion des Ergebnisses:
 - Bilde Kreuzprodukt
 - Selektiere mittels der Joinbedingung
 - Also: $R \bowtie_{\theta} S = \sigma_{\theta}(R \times S)$
 - $\theta \in \{=, <, >, \leq, \geq, \neq\}$
- Schema: Wie beim Kreuzprodukt
- Natural Join ist ein Spezialfall des Theta-Joins

Kombinierend: theta-join, \bowtie_{θ}

R

A	B	C
1	2	3
6	7	8
9	7	8

S

B	C	D
2	5	6
2	3	5
7	8	10

R $\bowtie_{A < D}$ S

A	R.B	R.C	S.B	S.C	D
1	2	3	2	5	6
1	2	3	2	3	5
1	2	3	7	8	10
6	7	8	7	8	10
9	7	8	7	8	10

R $\bowtie_{A < D \text{ AND } R.B \neq S.B}$ S

A	R.B	R.C	S.B	S.C	D
1	2	3	7	8	10

Verbundvarianten: Allgemeiner Verbund

- **Allgemeiner Verbund** (theta join):
- Gegeben seien: $R(A_1, \dots, A_n)$ und $S(B_1, \dots, B_m)$

— Dann gilt:

$$R \bowtie_{\theta} S = \sigma_P(R \times S) \text{ (kartesisches Produkt von R und S mit Selektion)}$$

$R \bowtie_{\theta} S$							
R				S			
A_1	A_2	...	A_n	B_1	B_2	...	B_m

Qualifizierung, Namenskonflikte

- Attributnamen müssen **innerhalb** eines Relationenformates **eindeutig** sein. Bei mehreren gleich benannten Attributen in verschiedenen Relationenformaten wird der Name zusätzlich durch die Bezeichnung des Relationenformaten qualifiziert: Gegeben: $R(A,B,C)$ und $S(A,B,C)$ dann wird z.B. A aus R und A aus S wie folgt qualifiziert: $R.A$ bzw. $S.A$
- Bei mehreren gleichlautenden Relationenformaten (siehe z.B. Auto-Join) wird der sogenannte Umbenennungsoperator ρ eingesetzt:

$$\rho_{\text{neuerName}}(R)$$

(auch gebräuchlich: $R \text{ AS } \text{neuerName}$)

R enthält dadurch den Namen *neuerName*

$$\rho_{S(D,E)}(R(A,B)) \text{ (Umbenennung } R.A \rightarrow S.D, R.B \rightarrow S.E)$$

Verbundvarianten: Sonstige

- Equi-Join: Prädikat P prüft nur auf Gleichheit und die Attribute sind mit logischem „und“ verknüpft
- Auto-Join: $R_1 \bowtie R_1$ (benötigt Umbenennung)
- Cross-Join: $R_1 \times R_2$
 - R_1 und R_2 haben keine gemeinsamen Attribute
 - Entspricht kartesischem Produkt
- Semi-Join: $\pi_{r_1, \dots, r_n}(R_1 \bowtie R_2)$
 - Natural Join mit Projektion auf die Attribute von R_1

Verbundvarianten: Outer Joins

- **Outer Joins:**
 - Left outer join: $L \bowtie R$
 - Right outer join: $L \ltimes R$
 - Full outer join: $L \ltimes R$
- **Left outer join:** Alle Tupel von L mit NULL's für Attribute von R wenn nicht verknüpfbar
- **Right outer join:** Alle Tupel von R mit NULL's für Attribute von L wenn nicht verknüpfbar
- **Full outer join:** Alle Tupel von L und R mit NULL's für Attribute von L bzw. R wenn nicht verknüpfbar

- Natürlicher join:

- Left outer join:



32

Join - Beispiele

- Right outer join:

L			R			Resultat				
A	B	C	C	D	E	A	B	C	D	E
a ₁	b ₁	c ₁	c ₁	d ₁	e ₁	a ₁	b ₁	c ₁	d ₁	e ₁
a ₂	b ₂	c ₂	c ₃	d ₂	e ₂	-	-	c ₃	d ₂	e ₂

- Full outer join:

L			R			Resultat				
A	B	C	C	D	E	A	B	C	D	E
a ₁	b ₁	c ₁	c ₁	d ₁	e ₁	a ₁	b ₁	c ₁	d ₁	e ₁
a ₂	b ₂	c ₂	c ₃	d ₂	e ₂	a ₂	b ₂	c ₂	-	-
-	-	-	-	-	-	-	-	c ₃	d ₂	e ₂

Äquivalenzregeln der Relationenalgebra

$$\sigma_{\Phi}(\sigma_{\Psi}(r)) = \sigma_{\Psi}(\sigma_{\Phi}(r))$$

Kommutativität

$$\pi_A(\sigma_{\Phi}(r)) = \sigma_{\Phi}(\pi_A(r)) \text{ falls } \Phi \text{ nur Attribute aus der Menge } A \text{ referenziert}$$

$$r \bowtie s = s \bowtie r \text{ (Achtung: Relationenformat ist verschieden!)}$$

$$r \bowtie (s \bowtie t) = (r \bowtie s) \bowtie t$$

Assoziativität

$$\pi_A(\pi_C(r)) = \pi_A(r) \text{ falls } A \subseteq C$$

$$\sigma_{\Phi}(\sigma_{\Psi}(r)) = \sigma_{\Phi \wedge \Psi}(r)$$

Idempotenz

$$\pi_A(r \cup s) = \pi_A(r) \cup \pi_A(s)$$

$$\sigma_{\Phi}(r \cup s) = \sigma_{\Phi}(r) \cup \sigma_{\Phi}(s)$$

Distributivität

$$\sigma_{\Phi}(r \bowtie s) = \sigma_{\Phi}(r) \bowtie s \text{ falls } \Phi \text{ nur Attribute von } r \text{ referenziert}$$

$$\pi_{A,B}(r \bowtie s) = \pi_A(r) \bowtie \pi_B(s) \text{ falls für die Joinattribute } J \text{ gilt: } J \subseteq A \cap B$$

$$r \bowtie (s \cup t) = (r \bowtie s) \cup (r \bowtie t)$$

Bierideen!

Gegeben:

- Gast(Besucher, Restaurant)
- Sortiment(Restaurant, Biersorte)
- Vorzug(Besucher, Biersorte)



Gesucht:

- Für Besucher „Meier“ alle Restaurants, die er besucht, und die seine bevorzugte Biersorte im Sortiment haben
- Für das Restaurant „Ochsen“ alle Biersorten, die es im Sortiment haben müsste, damit jeder Gast sein Vorzugsbier bekäme
- Alle Besucher, die ein Restaurant als Gast besuchen, welches eine von ihnen bevorzugte Biersorte führt.

Lösungen

- $\pi_{\text{Restaurant}}(\sigma_{\text{Besucher}='Meier'}(g) \bowtie s \bowtie v)$
- $\pi_{\text{Biersorte}}(\sigma_{\text{Restaurant}='Ochsen'}(g) \bowtie v)$
- $\pi_{\text{Besucher}}(g \bowtie s \bowtie v)$

Und weiter..

Das nächste Mal: Relationale Algebra (Fortsetzung), «bags»

