

**Folien basierend auf
Thorsten Papenbrock**

Übung Datenbanksysteme I Relationale Algebra

Leon Bornemann
G-3.1.09, Campus III
Hasso Plattner Institut

Organisatorisches

- Stand Postgresql Installation?
- Submit-System Probleme?
 - Jeder hat die Korrektur erhalten?
- Alle haben Mails der Mailingliste erhalten?

- Raumwechsel ab nächstem Mal!
 - Gebäude F
 - Raumnummer wird rumgeschickt

- Fragen zur Abgabe/Lösung?

DBSI - Übung

Relationale Algebra

Leon Bornemann

Chart **2**

Übersicht Relationale Algebra

Unäre Operatoren

Operator	Beschreibung
π (pi)	Projektion
σ (sigma)	Selektion
δ (delta)	Duplikateliminierung
ρ (rho)	Umbenennung
τ (tau)	Sortierung
γ (gamma)	Gruppierung

Binäre Operatoren

Operator	Beschreibung
\cap	Schnittmenge
\cup	Vereinigung
$-$	Differenz (auch „\“)
\times	Kreuzprodukt
\bowtie	Natürlicher Join
\bowtie_{θ}	Theta-Join
$ \bowtie$	Left outer Join
$\bowtie $	Right outer Join
$ \bowtie $	Full outer Join
\ltimes	Semijoin

Auswendig Lernen!

DBSI - Übung
Relationale Algebra

Leon Bornemann
Chart **3**

Aufgabe: Joinvarainten zuordnen

Semijoin

R

A	B
1	2
2	2
3	3

S

B	C
2	„a“
4	„b“

A	B
1	2
2	2

Full outer

A	B	C
1	2	„a“
2	2	„a“
3	3	null

A	B	C
1	2	„a“
2	2	„a“
null	4	„b“

A	B	C
1	2	„a“
2	2	„a“
null	4	„b“
3	3	null

Outer left

Outer right

Aufgabe:
Kardinalitäten bestimmen

Gegeben:

- Relation R mit m Tupeln
- Relation S mit n Tupeln

Gesucht:

- Die minimale und maximale Anzahl von Tupeln in folgenden Ausdrücken:

Ausdruck	minimal	maximal
$R \cup S$		
$R \bowtie S$		
$\sigma_C(R) \times S$		
$\pi_L(R) - S$		

Lösung: Kardinalitäten bestimmen

Gegeben:

- Relation R mit m Tupeln
- Relation S mit n Tupeln

Gesucht:

- Die minimale und maximale Anzahl von Tupeln in folgenden Ausdrücken:

Ausdruck	minimal	maximal
$R \cup S$	$\max(m,n)$	$m + n$
$R \bowtie S$	0	$m \cdot n$
$\sigma_C(R) \times S$	0	$m \cdot n$
$\pi_L(R) - S$	0	m

Bei Multimengensemantik: $m+n$

Bei Multimengensemantik: $\max(0,m-n)$

Übersicht:
Relationale Übungsschemata

Modellnummer

Prozessorgeschwindigkeit [MHz]

Product(maker, model, release)

- Beispieltupel: ('Lenovo', 1005, '2009')

Festplattengröße [GB]

PC(model, speed, ram, hd, rd, price)

- Beispieltupel: (1005, 1000, 128, 20, '12xDVD', 1499)

Geschwindigkeit und Typ des Laufwerks

Laptop(model, speed, ram, hd, screen, price)

- Beispieltupel: (2008, 650, 64, 10, 12.1, 1249)

Bildschirm-auflösung

Printer(model, color, type, price)

- Beispieltupel: (3005, 'red', 'bubble', 200)

Typ (laser, ink-jet, bubble)

Preis [Euro]

Aufgabe:
Deutsch → relationale Algebra

Product(maker, model, release)

- Beispieltupel: (``Lenovo``, 1005, ``2009``)

PC(model, speed, ram, hd, rd, price)

- Beispieltupel: (1005, 1000, 128, 20, ``12xDVD``, 1499)

Laptop(model, speed, ram, hd, screen, price)

- Beispieltupel: (2008, 650, 64, 10, 12.1, 1249)

Printer(model, color, type, price)

- Beispieltupel: (3005, true, ``bubble``, 200)

Anfrage: „Welche PC-Modelle haben eine Geschwindigkeit von mindestens 1000 MHz?“

Lösung:
Deutsch → relationale Algebra

Product(maker, model, release)

- Beispieltupel: (``Lenovo``, 1005, ``2009``)

PC(model, speed, ram, hd, rd, price)

- Beispieltupel: (1005, 1000, 128, 20, ``12xDVD``, 1499)

Laptop(model, speed, ram, hd, screen, price)

- Beispieltupel: (2008, 650, 64, 10, 12.1, 1249)

Printer(model, color, type, price)

- Beispieltupel: (3005, true, ``bubble``, 200)

Anfrage: „Welche PC-Modelle haben eine Geschwindigkeit von mindestens 1000 MHz?“

$$\pi_{\text{model}}(\sigma_{\text{speed} \geq 1000}(\text{PC}))$$

Aufgabe:
Deutsch → relationale Algebra

Product(maker, model, release)

- Beispieltupel: (``Lenovo``, 1005, ``2009``)

PC(model, speed, ram, hd, rd, price)

- Beispieltupel: (1005, 1000, 128, 20, ``12xDVD``, 1499)

Laptop(model, speed, ram, hd, screen, price)

- Beispieltupel: (2008, 650, 64, 10, 12.1, 1249)

Printer(model, color, type, price)

- Beispieltupel: (3005, true, ``bubble``, 200)

Anfrage: „Welche Hersteller bauen Laptops mit einer Harddisk von mindestens 10 GB Größe?“

Lösung:
Deutsch → relationale Algebra

Product(maker, model, type)

- Beispieltupel: (`'Lenovo'`, 1005, `'pc'`)

PC(model, speed, ram, hd, rd, price)

- Beispieltupel: (1005, 1000, 128, 20, `'12xDVD'`, 1499)

Laptop(model, speed, ram, hd, screen, price)

- Beispieltupel: (2008, 650, 64, 10, 12.1, 1249)

Printer(model, color, type, price)

- Beispieltupel: (3005, `true`, `'bubble'`, 200)

Anfrage: „Welche Hersteller bauen Laptops mit einer Harddisk von mindestens 10 GB Größe?“

$$\pi_{\text{maker}}(\sigma_{\text{hd} \geq 10}(\text{Product} \bowtie \text{Laptop}))$$

Lösung:
Deutsch → relationale Algebra

Product(maker, model, type)

- Beispieltupel: ('Lenovo', 1005, 'pc')

PC(model, speed, ram, hd, rd, price)

- Beispieltupel: (1005, 1000, 128, 20, '12xDVD', 1499)

Laptop(model, speed, ram, hd, screen, price)

- Beispieltupel: (2008, 650, 64, 10, 12.1, 1499)

Printer(model, color, type, price)

- Beispieltupel: (3005, true, 'Laser', 200)

Anfrage: „Welche Hersteller bauen Laptops mit einer Harddisk von mindestens 10 GB Größe?“

$\delta(\pi_{\text{maker}}(\sigma_{\text{hd} \geq 10}(\text{Product} \bowtie \text{Laptop})))$

Bei Multimensenmantik
Wichtig!

Aufgabe:
Deutsch → relationale Algebra

Product(maker, model,

- Beispieltupel: ('Lenovo',

Farbe
[true, false]

Typ
[laser, ink-jet, bubble]

PC(model, speed, ram, hd, price)

- Beispieltupel: (1005, 1, 128, 20, '12x7', 1499)

Laptop(model, speed, ram, hd, screen, price)

- Beispieltupel: (2005, 650, 64, 10, 12.1, 1249)

Printer(model, color, type, price)

- Beispieltupel: (3005, true, 'bubble', 200)

Anfrage: „Finde die Modellnummern aller Farblaserdrucker.“

Lösung:
Deutsch → relationale Algebra

Product(maker, model,

- Beispieltupel: ('Lenovo',

Farbe
[true, false]

Typ
[laser, ink-jet, bubble]

PC(model, speed, ram, hd, price)

- Beispieltupel: (1005, 1, 128, 20, '12x7', 1499)

Laptop(model, speed, ram, hd, screen, price)

- Beispieltupel: (2005, 650, 64, 10, 12.1, 1249)

Printer(model, color, type, price)

- Beispieltupel: (3005, true, 'bubble', 200)

Anfrage: „Finde die Modellnummern aller Farblaserdrucker.“

$\pi_{\text{model}}(\sigma_{\text{color}=\text{true} \wedge \text{type}=\text{'laser'}}(\text{Printer}))$

Aufgabe:
Deutsch → relationale Algebra

Product(maker, model, release)

- Beispieltupel: (`'Lenovo'`, 1005, `'2009'`)

PC(model, speed, ram, hd, rd, price)

- Beispieltupel: (1005, 1000, 128, 20, `'12xDVD'`, 1499)

Laptop(model, speed, ram, hd, screen, price)

- Beispieltupel: (2008, 650, 64, 10, 12.1, 1249)

Printer(model, color, type, price)

- Beispieltupel: (3005, `true`, `'bubble'`, 200)

Anfrage: „Finde die Modellnummer und den Preis aller Produkte (jeden Typs), die von Hersteller `'Apple'` gebaut werden.“

Lösung:
Deutsch → relationale Algebra

Product(maker, model, release)

- Beispieltupel: ('Lenovo', 1005, '2009')

PC(model, speed, ram, hd, rd, price)

- Beispieltupel: (1005, 1000, 128, 20, '12xDVD', 1499)

Laptop(model, speed, ram, hd, screen, price)

- Beispieltupel: (2008, 650, 64, 10, 12.1, 1249)

Printer(model, color, type, price)

- Beispieltupel: (3005, true, 'bubble', 200)

Anfrage: „Finde die Modellnummer und den Preis aller Produkte (jeden Typs), die von Hersteller 'Apple' gebaut werden.“

$$\pi_{\text{model,price}}(\sigma_{\text{maker}='Apple'}(\text{Product} \bowtie (\pi_{\text{model,price}}(\text{PC}) \cup \pi_{\text{model,price}}(\text{Laptop}) \cup \pi_{\text{model,price}}(\text{Printer}))))$$

Warum?

Aufgabe:
Deutsch → relationale Algebra

Product(maker, model, release)

- Beispieltupel: (``Lenovo``, 1005, ``2009``)

PC(model, speed, ram, hd, rd, price)

- Beispieltupel: (1005, 1000, 128, 20, ``12xDVD``, 1499)

Laptop(model, speed, ram, hd, screen, price)

- Beispieltupel: (2008, 650, 64, 10, 12.1, 1249)

Printer(model, color, type, price)

- Beispieltupel: (3005, true, ``bubble``, 200)

Anfrage: „Finde alle Hersteller, die Laptops, aber keine PCs herstellen.“

Lösung:
Deutsch → relationale Algebra

Product(maker, model, release)

- Beispieltupel: (``Lenovo``, 1005, ``2009``)

PC(model, speed, ram, hd, rd, price)

- Beispieltupel: (1005, 1000, 128, 20, ``12xDVD``, 1499)

Laptop(model, speed, ram, hd, screen, price)

- Beispieltupel: (2008, 650, 64, 10, 12.1, 1249)

Printer(model, color, type, price)

- Beispieltupel: (3005, `true`, ``bubble``, 200)

Anfrage: „Finde alle Hersteller, die Laptops, aber keine PCs herstellen.“

$$\pi_{\text{maker}}(\text{Product} \bowtie \text{Laptop}) - \pi_{\text{maker}}(\text{Product} \bowtie \text{PC})$$

Lösung:
Deutsch → relationale Algebra

Product(maker, model, release)

- Beispieltupel: ('Lenovo', 1005, '2009')

PC(model, speed, ram, hd, rd, price)

- Beispieltupel: (1005, 1000, 128, 20, '12xDVD', 1000)

Laptop(model, speed, ram, hd, screen, price)

- Beispieltupel: (2008, 650, 64, 10, 12.1, 1249)

Printer(model, color, type, price)

- Beispieltupel: (3005, true, 'bubble', 200)

Anfrage: „Finde alle Hersteller, die Laptops, aber keine PCs herstellen.“

Multimengensemantik:
Bei Multimengensemantik
brauchen wir die
Dublikateliminierung!
(ein existierender PC reicht aus)

$$\delta(\pi_{\text{maker}}(\text{Product} \bowtie \text{Laptop})) - \delta(\pi_{\text{maker}}(\text{Product} \bowtie \text{PC}))$$

Lösung:
Deutsch → relationale Algebra

Product(maker, model, release)

- Beispieltupel: ('Lenovo', 1005, '2009')

PC(model, speed, ram, hd, rd, price)

- Beispieltupel: (1005, 1000, 128, 20, '12xDVD', 1499)

Laptop(model, speed, ram, hd, screen, price)

- Beispieltupel: (2008, 650, 64, 10, 12.1, 1249)

Printer(model, color, type, price)

- Beispieltupel: (3005, true, 'bubble', 200)

Anfrage: „Finde alle Hersteller, die mehr Laptopmodelle als PC-Modelle herstellen.“

$$\delta(\pi_{\text{maker}}(\text{Product} \bowtie \text{Laptop}) - \pi_{\text{maker}}(\text{Product} \bowtie \text{PC}))$$

Was ist die Semantik dieser Anfrage
Bei Multimengensemantik ?

Aufgabe:
Deutsch → relationale Algebra

Product(maker, model, release)

- Beispieltupel: (``Lenovo``, 1005, ``2009``)

PC(model, speed, ram, hd, rd, price)

- Beispieltupel: (1005, 1000, 128, 20, ``12xDVD``, 1499)

Laptop(model, speed, ram, hd, screen, price)

- Beispieltupel: (2008, 650, 64, 10, 12.1, 1249)

Printer(model, color, type, price)

- Beispieltupel: (3005, true, ``bubble``, 200)

Anfrage: „Finde alle Harddisk-Größen, die in mehr als zwei PCs vorkommen.“

Lösung:
Deutsch → relationale Algebra

Product(maker, model, release)

- Beispieltupel: (``Lenovo``, 1005, ``2009``)

PC(model, speed, ram, hd, rd, price)

- Beispieltupel: (1005, 1000, 128, 20, ``12xDVD``, 1499)

Laptop(model, speed, ram, hd, screen, price)

- Beispieltupel: (2008, 650, 64, 10, 12.1, 1249)

Printer(model, color, type, price)

- Beispieltupel: (3005, `true`, ``bubble``, 200)

Anfrage: „Finde alle Harddisk-Größen, die in mehr als zwei PCs vorkommen.“

$$\pi_{hd}(\sigma_{Anzahl > 2}(\gamma_{hd, count(model) \rightarrow Anzahl(PC)}))$$

Aufgabe:
Deutsch → relationale Algebra

Product(maker, model, release)

- Beispieltupel: (``Lenovo``, 1005, ``2009``)

PC(model, speed, ram, hd, rd, price)

- Beispieltupel: (1005, 1000, 128, 20, ``12xDVD``, 1499)

Laptop(model, speed, ram, hd, screen, price)

- Beispieltupel: (2008, 650, 64, 10, 12.1, 1249)

Printer(model, color, type, price)

- Beispieltupel: (3005, true, ``bubble``, 200)

Anfrage: „Finde alle Paare von PCs mit gleicher Festplatten- und Hauptspeichergröße. Vermeide dabei doppelte Paare.“

Lösung:
Deutsch → relationale Algebra

Product(maker, model, release)

- Beispieltupel: ('Lenovo', 1005, '2009')

PC(model, speed, ram, hd, rd, price)

- Beispieltupel: (1005, 1000, 128, 20, '12xDVD', 1499)

Laptop(model, speed, ram, hd, screen, price)

- Beispieltupel: (2008, 650, 64, 10, 12.1, 1249)

Printer(model, color, type, price)

- Beispieltupel: (3005, true, 'bubble', 200)

Anfrage: „Finde alle Paare von PCs mit gleicher **Festplatten-** und **Hauptspeichergröße.** **Vermeide dabei doppelte Paare.**“

$$(\rho_{PC_1}(PC)) \bowtie_{PC_1.HD=PC_2.HD \wedge PC_1.RAM=PC_2.RAM \wedge PC_1.model < PC_2.model} (\rho_{PC_2}(PC))$$

Aufgabe:

Relationale Algebra → Deutsch

$$\pi_{D.a}(\sigma_{A.b=B.b \wedge A.a=C.g \wedge B.a=D.g \wedge C.a=D.a}(\rho_{A(a,b,c,d,e,f)}(\text{Laptop}) \times \rho_{B(a,b,c,d,e,f)}(\text{PC}) \times \rho_{C(a,g,h)}(\text{Product}) \times \rho_{D(a,g,h)}(\text{Product})))$$

Product(maker, model, release)

- Beispieltupel: ('Lenovo', 1005, '2009')

PC(model, speed, ram, hd, rd, price)

- Beispieltupel: (1005, 1000, 128, 20, '12xDVD', 1499)

Laptop(model, speed, ram, hd, screen, price)

- Beispieltupel: (2008, 650, 64, 10, 12.1, 1249)

Printer(model, color, type, price)

- Beispieltupel: (3005, true, 'bubble', 200)

DBSI - Übung

Relationale Algebra

Leon Bornemann

Chart 25

Lösung:

Relationale Algebra → Deutsch

model

model

$$\pi_{D.a}(\sigma_{A.b=B.b \wedge A.a=C.g \wedge B.a=D.g \wedge C.a=D.a}(\rho_{A(a,b,c,d,e,f)}(\text{Laptop}) \times \rho_{B(a,b,c,d,e,f)}(\text{PC}) \times \rho_{C(a,g,h)}(\text{Product}) \times \rho_{D(a,g,h)}(\text{Product})))$$

maker

$$\Leftrightarrow \pi_{D.a}(\sigma_{A.b=B.b \wedge C.a=D.a}(\rho_{A(a,b,c,d,e,f)}(\text{Laptop}) \bowtie_{A.a=C.g} \rho_{C(a,g,h)}(\text{Product})) \times (\rho_{B(a,b,c,d,e,f)}(\text{PC}) \bowtie_{B.a=D.g} \rho_{D(a,g,h)}(\text{Product})))$$

speed

$$\Leftrightarrow \pi_{D.a}(\sigma_{A.b=B.b}(\rho_{A(a,b,c,d,e,f)}(\text{Laptop}) \bowtie_{C.a=D.a} \rho_{C(a,g,h)}(\text{Product})) \bowtie_{C.a=D.a} (\rho_{B(a,b,c,d,e,f)}(\text{PC}) \bowtie_{D.a=g} \rho_{D(a,g,h)}(\text{Product})))$$

maker

„Finde alle Hersteller, die mindestens einen Laptop und einen PC anbieten, deren Prozessoren dieselbe Taktfrequenz haben.“



Übung Datenbanksysteme I Relationale Algebra

Leon Bornemann
G-3.1.09, Campus III
Hasso Plattner Institut