AIFB

Übung zur Vorlesung Angewandte Informatik I
WS 2005/2006

Prof. Dr. A Oberweis, Prof. Dr. R. Studer, Dr. Hitzler
Dipl.-Wi.-Inform. Victor Pankratius, Dipl.-Ing.-El. Nenad Stojanovic

Übung 7

Relationales Datenmodell, Teil 2

- Lösungsvorschläge -

Übung 7 Aufgabe 1

Aufgabe 1

In der Vorlesung wurden algebraische Eigenschaften der relationalen Operatoren besprochen.

a) *Geben Sie für jede unten aufgeführte Eigenschaft jeweils ein Beispiel (Details vgl. Folien).*

b) *Zeigen Sie anhand eines Beispiels, dass die Reihenfolge von Projektion und Join wichtig ist (vgl. Folie 57).*

c) *Wofür könnte man die algebraischen Eigenschaften bei der Implementierung von DBMS verwenden?*

Übung 7 Aufgabe 2

Aufgabe 2

Erklären Sie die Bedeutung und die Einsatzmöglichkeiten folgender Join-Varianten. Geben Sie jeweils ein Beispiel!

- *Theta Join*
- *Equi-Join*
- *Left Semi-Join*
- *Left-Outer-Theta-Join*
- *Right-Outer-Theta-Join*
- *Full-Outer-Theta-Join*

Übung 7 Aufgabe 3

Gegeben seien folgende Relationsschemata einer relationalen Datenbank:

```
lieferant: LIEFERANT(L#, LNAME, STATUS, STADT)
teil: TEIL(T#, TNAME, FARBE, GEWICHT, STADT)
projekt: PROJEKT(P#, PNAME, STADT)
lieferung: LIEFERUNG(L#, T#, P#, MENGE)
```

Formulieren Sie die folgenden Anfragen in Relationenalgebra:

a) Liste sämtliche Lieferungen auf, deren Mengen zwischen 300 und 750 liegen.
b) Gib alle Kombinationen von Lieferanten-, Teil- und Projektnummern aus, für die gilt: Lieferant, Teil und Projekt befinden sich in derselben Stadt.
c) Gib die Nummern aller Teile aus, die von einem Lieferanten aus London geliefert wurden.
d) Gib die Nummern aller Teile aus, die für ein Projekt in London von einem Lieferanten aus London geliefert wurden.
e) Suche die Nummern von Projekten, für die von mindestens einem Lieferanten Teile geliefert wurden, der sich nicht in derselben Stadt befindet wie das Projekt.
f) Gib die Namen aller Projekte aus, für die Lieferant L1 Teile geliefert hat.
g) Suche alle Nummern von Teilen, die nicht für ein Projekt in London geliefert wurden.
h) Suche die Nummern von Lieferanten außer L2, die gleiche Teile liefern wie der Lieferant L2.

AIFB

Übung 7 Aufgabe 3

a.) $\sigma_{[\text{Menge} > 300 \wedge \text{Menge} < 750]}(\text{lieferung})$

b.) $\pi_{[L\#,\ T\#,\ P\#]}\ (\text{lieferant} * (\text{teil} * \text{projekt}))$

c.) $\pi_{[T\#]}\ (\text{lieferung} * (\sigma_{[\text{Stadt} = \text{"London“}]}\ (\text{lieferant}))$

d.) $\pi_{[T\#]}\ (\text{lieferung} * (\sigma_{[\text{Stadt} = \text{"London“}]}\ (\text{projekt}) * \sigma_{[\text{Stadt} = \text{"London“}]}\ (\text{lieferant})))$

e.) $\pi_{[P\#]}\ ((\text{projekt} *_{[\text{projekt.STADT} \neq \text{lieferant.STADT}]}\ \text{lieferant}) * \text{lieferung})$

f.) $\pi_{[\text{PNAME}]}\ (\text{projekt} * (\sigma_{\ [L\# = \text{"L1“}]}\ (\text{lieferung})))$

g.) $\pi_{[T\#]}\ (\text{teil}) \setminus \pi_{[T\#]}\ (\sigma_{[\text{STADT}=\text{"LONDON“}]}\ (\text{projekt} * \text{lieferung}))$

Die Mengen müssen vereinigungskompatibel sein, deshalb jeweils Projektionen auf T#

h.) $\underbrace{\pi_{[L\#]}\ (\text{lieferung} * \pi_{[T\#]}\ (\sigma_{[L\# = \text{"L2“}]}\ (\text{lieferung})))}_{\text{L2 hier auch dabei}} \quad \setminus \pi_{[L\#]}\ (\sigma_{[L\# = \text{"L2“}]}\ (\text{lieferung}))$

deswegen Differenz

Bemerkung:

Zu einer Aufgabenstellung sind unterschiedliche Anfragen in Relationenalgebra möglich
(vgl. algebraische Regeln zur Umformung aus der Vorlesung)

AIFB

- Es folgen die von den Tutoren erstellten Folien

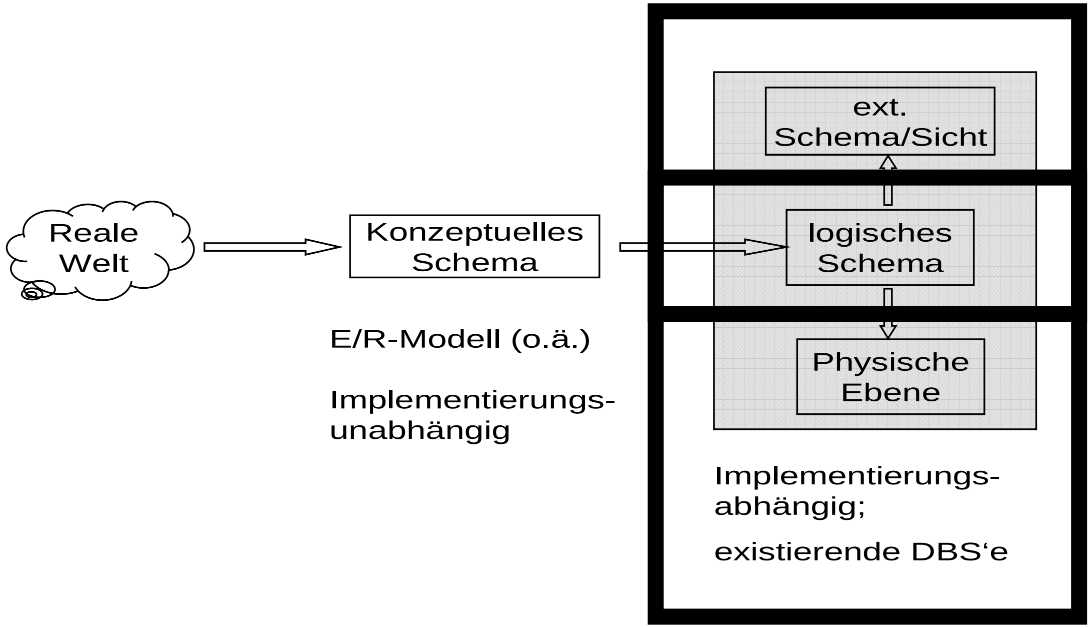
»Willkommen!«

Angewandte Informatik I, Tutorium Nr. 7
WS 2005 / 06

Henrik Simon

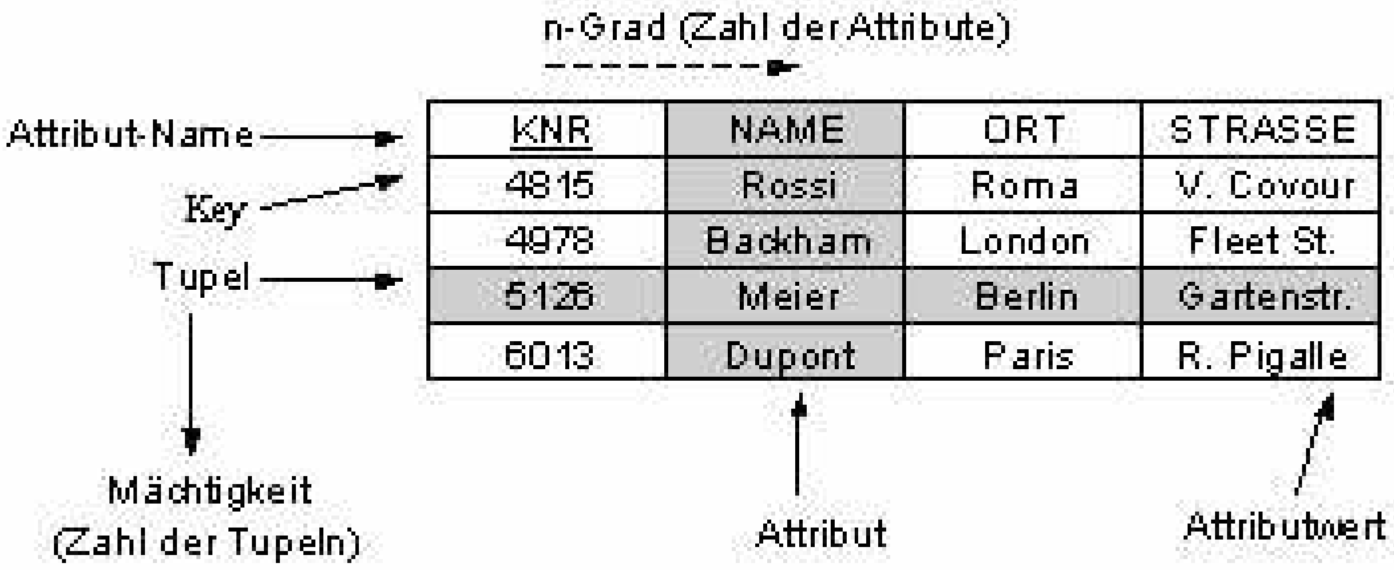
Relationales Datenmodell

Wie kann eine Datenbank konkret aufgebaut werden?



Im Folgenden realisieren wir Datenbanken mittels Tabellen.

AIFB



KNR	NAME	ORT	STRASSE
4815	Rossi	Roma	V. Covour
4978	Backham	London	Fleet St.
5126	Meier	Berlin	Gartenstr.
6013	Dupont	Paris	R. Pigalle

Relation (bzw. Tabelle):
Sammlung aller zugehörigen Tupeln.

Attribut:
Teil (Feld) einer Tupel.

Tupel:
Einzelner Satz einer Relation (Zeile) aus Attributen zusammengesetzt.

Schlüssel (Key):
Eindeutige Charakterisierung der Tupel einer Relation; aus einem oder mehreren Attributen zusammengesetzt

AIFB

Relationstyp

$$\textit{TELEFONBUCH} = (\underbrace{\textit{NAME, VORNAME, NR}}_{\textbf{Attributmenge } A} \mid \underbrace{\Sigma T}_{\text{Menge von semantischen Integritätsbedingungen}})$$

mit Σ_T*={„NR ist PS"}*

TELEFONBUCH $\subseteq$ *dom(NAME) x dom(VORNAME) x dom(NR)*
mit Einschränkung Σ_T

r: (A | Σ*) wird als Schema einer Relation bezeichnet.*

Wdh. - Gegenüberstellung der Konzepte

AIFB

Tabelle	Relationales Datenmodell	Entity Relationship Model (ERM)	Unified Modeling Language (UML)
Wertebereich (Domäne, Domain)	Wertebereich (Domäne, Domain)	Wertebereich (Domäne, Domain)	Wertebereich (Domäne, Domain)
Kopfzeile	Relationstyp / Relationsformat	Entitätstyp	Klasse
Spaltenüberschrift	Attribut	Attribut	Attribut
Inhalt	Relation	Entitätsmenge(-set)	Objektmenge, Instanzmenge, Klasse
Zeile	Tupel	Entität	Objekt, Instanz
Zelle	Attributwert	Attributwert	Attributwert

1) $\sigma_{[b1]}(\sigma_{[b2]}(r_1)) = \sigma_{[b2]}(\sigma_{[b1]}(r_1))$

Bedingungen:
b_1 : c = 2
b_2 : a = 1

r_1		
a	**b**	**c**
1	1	1
1	2	2
2	0	2

!

=

r_1		
a	**b**	**c**
1	1	1
1	2	2
2	0	2

AIFB

1) $\sigma_{[b1]}(\sigma_{[b2]}(r_1)) = \sigma_{[b2]}(\sigma_{[b1]}(r_1))$

Bedingungen:
b_1 : c = 2
b_2 : a = 1

$\sigma_{[b1]}(\sigma_{[b2]}(r_1))$		
a	**b**	**c**
1	2	2

=

$\sigma_{[b2]}(\sigma_{[b1]}(r_1))$		
a	**b**	**c**
1	2	2

Neue Tabelle besteht nur noch aus dem rot gefärbten Bereich.

AIFB

2) $\sigma_{[b1 \wedge b2]}(r_1) \quad = \quad \sigma_{[b1]}(\sigma_{[b2]}(r_1))$

Bedingungen:
b_1 : c = 2
b_2 : a = 1

r_1		
a	**b**	**c**
1	1	1
1	2	2
2	0	2

! =

r_1		
a	**b**	**c**
1	1	1
1	2	2
2	0	2

2) $\sigma_{[b1 \wedge b2]}(r_1) = \sigma_{[b1]}(\sigma_{[b2]}(r_1))$

Bedingungen:
b_1 : c = 2
b_2 : a = 1

$\sigma_{[b1 \wedge b2]}(r_1)$		
a	**b**	**c**
1	2	2

=

$\sigma_{[b1]}(\sigma_{[b2]}(r_1))$		
a	**b**	**c**
1	2	2

AIFB

3) $\sigma_{[b1 \vee b2]}(r) \quad = \quad \sigma_{[b1]}(r) \cup \sigma_{[b2]}(r)$

Bedingungen:

b_1 : c = 2
b_2 : a = 1

r_1		
a	**b**	**c**
1	1	1
1	2	2
2	0	2

$\overset{!}{=}$

r_1		
a	**b**	**c**
1	1	1
1	2	2
2	0	2

$\cup$

r_1		
a	**b**	**c**
1	1	1
1	2	2
2	0	2

Aufgabe 1a) – Eigenschaften von Selektion

3) $\sigma_{[b1 \vee b2]}(r) \quad = \quad \sigma_{[b1]}(r) \cup \sigma_{[b2]}(r)$

Bedingungen:	
b_1 :	c = 2
b_2 :	a = 1

r_1		
a	**b**	**c**
1	1	1
1	2	2
2	0	2

!

=

$\sigma_{[b1]}(r) \cup \sigma_{[b2]}(r)$		
a	**b**	**c**
1	1	1
1	2	2
2	0	2

r_1		
a	**b**	**c**
1	1	1
1	2	2
2	0	2

4) $\sigma_{[b]}(r_1 \cup r_2) \quad = \quad \sigma_{[b]}(r_1) \cup \sigma_{[b]}(r_2)$

Bedingung:
b : a = 1

a
1
1
2

$r_1 \cup r_2$		
a	**b**	**c**
1	1	1
1	2	2
2	0	2
1	2	3

$\cup$

r_2		
a	**b**	**c**
1	2	2
1	2	3

4) $\sigma_{[b]}(r_1 \cup r_2) \quad = \quad \sigma_{[b]}(r_1) \cup \sigma_{[b]}(r_2)$

Bedingung:
b : a = 1

$r_1 \cup r_2$		
a	**b**	**c**
1	1	1
1	2	2
2	0	2
1	2	3

! =

$r_1 \cup r_2$		
a	**b**	**c**
1	1	1
1	2	2
2	0	2
1	2	3

$\cup$

r_2		
a	**b**	**c**
1	2	2
1	2	3

4) $\sigma_{[b]}(r_1 \cup r_2) \quad = \quad \sigma_{[b]}(r_1) \cup \sigma_{[b]}(r_2)$

Bedingung:
b : a = 1

$r_1 \cup r_2$		
a	**b**	**c**
1	1	1
1	2	2
1	2	3

$\overset{!}{=}$

$r_1 \cup r_2$		
a	**b**	**c**
1	1	1
1	2	2
1	2	3

geg. $\mathbf{r_1}$: (A | …), $\mathbf{r_2}$: (B | …)

5) $\sigma_{[b]}(r_1 \times r_2) \quad = \quad \sigma_{[b]}(r_1) \times r_2$

falls b nur Attribute aus A

Bedingung:
b :   a = 1

r_1 x r_2					
a	**b**	**c**	**d**	**e**	**f**
1	1	1	1	2	2
1	2	2	1	2	2
2	0	2	1	2	2
1	1	1	1	2	3
1	2	2	1	2	3
2	0	2	1	2	3

r_2	
e	**f**
2	2
2	3

geg. $\mathbf{r_1}$: (A | …), $\mathbf{r_2}$: (B | …)

5) $\sigma_{[b]}(r_1 \times r_2) \quad = \quad \sigma_{[b]}(r_1) \times r_2$

falls b nur Attribute aus A

Bedingung:
b : a = 1

r_1 x r_2					
a	**b**	**c**	**d**	**e**	**f**
1	1	1	1	2	2
1	2	2	1	2	2
2	0	2	1	2	2
1	1	1	1	2	3
1	2	2	1	2	3
2	0	2	1	2	3

AIFB

geg. r_1: (A | …), r_2: (B | …)

5) $\sigma_{[b]}(r_1 \times r_2) \quad = \quad \sigma_{[b]}(r_1) \times r_2$

falls b nur Attribute aus A

Bedingung:

b : a = 1

σ[b] r_1 x r_2					
a	**b**	**c**	**d**	**e**	**f**
1	1	1	1	2	2
1	2	2	1	2	2
1	1	1	1	2	3
1	2	2	1	2	3

$\overset{!}{=}$

r_1		
a	**b**	**c**
1	1	1
1	2	2
2	0	2

Aufgabe 1a) – Eigenschaften von Selektion

geg. $\mathbf{r_1}$: (A | …), $\mathbf{r_2}$: (B | …)

5) $\sigma_{[b]}(r_1 \times r_2) \quad = \quad \sigma_{[b]}(r_1) \times r_2$

falls b nur Attribute aus A

Bedingung:
b : a = 1

σ[b] r_1 x r_2					
a	**b**	**c**	**d**	**e**	**f**
1	1	1	1	2	2
1	2	2	1	2	2
1	1	1	1	2	3
1	2	2	1	2	3

=

σ[b] (r_1) x r_2					
a	**b**	**c**	**d**	**e**	**f**
1	1	1	1	2	2
1	2	2	1	2	2
1	1	1	1	2	3
1	2	2	1	2	3

r_2	
e	**f**
2	2
2	3

AIFB

1. Fall

geg. $\mathbf{r_1}$: (A | …), $\mathbf{r_2}$: (B | …)

5) $\sigma_{[b]}(r_1 \times r_2) \quad = \quad \sigma_{[b]}(r_1) \times r_2$

falls b nur Attribute aus A

2. Fall

geg. $\mathbf{r_1}$: (A | …), $\mathbf{r_2}$: (B | …)

5) $\sigma_{[b]}(r_1 \times r_2) \quad = \quad r_1 \times \sigma_{[b]}(r_2)$

falls b nur Attribute aus B

…analog zu Fall 1

1) $r_1 * r_2 = r_2 * r_1$

geg. r_1: (A | …), r_2: (B | …), r_3: (C | …)

r_1		
a	**b**	**c**
1	1	1
1	2	2
2	0	2

*

r_2	
a	**d**
1	1
0	1
2	0

$r_1 * r_2$			
a	**b**	**c**	**d**
1	1	1	1
1	2	2	1
2	0	2	0

r_2	
a	**d**
1	1
0	1
2	0

*

r_1		
a	**b**	**c**
1	1	1
1	2	2
2	0	2

$\underline{(r_1 * r_2) * r_3} = r_1 * (r_2 * r_3) = r_1 * r_2 * r_3$

geg. r_1: (A | …), r_2: (B | …), r_3: (C | …)

r_1		
a	**b**	**c**
1	1	1
1	2	2
2	0	2

*

r_2	
a	**d**
1	1
0	1
2	0

$r_1 * r_2$			
a	**b**	**c**	**d**
1	1	1	1
1	2	2	1
2	0	2	0

$r_1 * r_2$			
a	**b**	**c**	**d**
1	1	1	1
1	2	2	1
2	0	2	0

*

r_3		
c	**d**	**e**
1	1	0
0	1	1
2	1	0
2	2	1

$r_1 * r_2 * r_3$				
a	**b**	**c**	**d**	**e**
1	1	1	1	0
1	2	2	1	0

AIFB

$(r_1 * r_2) * r_3 = \underline{r_1 * (r_2 * r_3)} = r_1 * r_2 * r_3$

geg. r_1: (A | …), r_2: (B | …), r_3: (C | …)

r_2	
a	**d**
1	1
0	1
2	0

*

r_3		
c	**d**	**e**
1	1	0
0	1	1
2	1	0
2	2	1

$r_2 * r_3$			
a	**c**	**d**	**e**
1	1	1	0
1	0	1	1
1	2	1	0
0	1	1	0
0	0	1	1
0	2	1	0

Ergebnis 1. Teil

$r_1 * r_2 * r_3$				
a	**b**	**c**	**d**	**e**
1	1	1	1	0
1	2	2	1	0

Aufgabe 1a) – Eigenschaften von *

$(r_1 * r_2) * r_3 = \underline{r_1 * (r_2 * r_3)} = r_1 * r_2 * r_3$

geg. r_1: (A | …), r_2: (B | …), r_3: (C | …)

$r_2 * r_3$			
a	**c**	**d**	**e**
1	1	1	0
1	0	1	1
1	2	1	0
0	1	1	0
0	0	1	1
0	2	1	0

*

r_1		
a	**b**	**c**
1	1	1
1	2	2
2	0	2

$r_1 * r_2 * r_3$				
a	**b**	**c**	**d**	**e**
1	1	1	1	0
1	2	2	1	0

=

Ergebnis 1. Teil

$r_1 * r_2 * r_3$				
a	**b**	**c**	**d**	**e**
1	1	1	1	0
1	2	2	1	0

AIFB

$(r_1 * r_2) * r_3 = r_1 * (r_2 * r_3) = \underline{r_1 * r_2 * r_3}$

geg. $\mathbf{r_1}$: (A | …), $\mathbf{r_2}$: (B | …), $\mathbf{r_3}$: (C | …)

…analog zum 1. Teil

$(r_1 * r_2) * r_3$				
a	**b**	**c**	**d**	**e**
1	1	1	1	0
1	2	2	1	0

=

$r_1 * (r_2 * r_3)$				
a	**b**	**c**	**d**	**e**
1	1	1	1	0
1	2	2	1	0

=

$r_1 * r_2 * r_3$				
a	**b**	**c**	**d**	**e**
1	1	1	1	0
1	2	2	1	0

AIFB

$A \cap B = \emptyset \quad \Rightarrow r_1 * r_2 = r_1 \times r_2$

geg. r_1: (A | …), r_2: (B | …), r_3: (C | …)

r_1		
a	**b**	**c**
1	1	1
1	2	2
2	0	2

$\cap$

r_2		
d	**e**	**f**
1	1	2
1	2	3

$= \emptyset$

Gibt es keine gemeinsamen Attribute in den beiden Relationen, entspricht die Operation * dem kartesischen Produkt.

$r_1 * r_2 = r_1 \times r_2$					
a	**b**	**c**	**d**	**e**	**f**
1	1	1	1	2	2
1	2	2	1	2	2
2	0	2	1	2	2
1	1	1	1	2	3
1	2	2	1	2	3
2	0	2	1	2	3

Aufgabe 1a) – Eigenschaften von *

AIFB

$r_1 = \emptyset \quad \Rightarrow r_1 * r_2 = \emptyset$

geg. r_1: (A | …), r_2: (B | …), r_3: (C | …)

r_1	
a	d

= $\emptyset$

*

r_2		
a	b	c
1	1	1
1	2	2
2	0	2

$r_1 * r_2$			
a	b	c	d

= $\emptyset$

$A = B \Rightarrow r_1 * r_2 = r_1 \cap r_2$

geg. r_1: (A | …), r_2: (B | …), r_3: (C | …)

r_1

a	b	c
1	1	1
1	2	2
2	0	2

*

r_2

a	b	c
1	1	1
1	2	2
2	0	2

⇒

$r_1 * r_2$

a	b	c
1	1	1
1	2	2
2	0	2

=

r_1

a	b	c
1	1	1
1	2	2
2	0	2

$\cap$

r_2

a	b	c
1	1	1
1	2	2
2	0	2

⇒

$r_1 \cap r_2$

a	b	c
1	1	1
1	2	2
2	0	2

$r_1 * r_1 = r_1$

geg. r_1: (A | …), r_2: (B | …), r_3: (C | …)

r_1		
a	**b**	**c**
1	1	1
1	2	2
2	0	2

*

r_1		
a	**b**	**c**
1	1	1
1	2	2
2	0	2

$r_1 * r_1$		
a	**b**	**c**
1	1	1
1	2	2
2	0	2

$$(r_1)^{21} = (r_1) * (r_1) * (r_1) * \ldots = ?$$

$\sigma_{[b]}(r_1 * r_2) = \sigma_{[b]}(r_1) * \sigma_{[b]}(r_2)$

geg. r_1: (A | …), r_2: (B | …), r_3: (C | …)

falls b nur Attribute aus A ∩ B besitzt

Bedingung:
b : a = 1

r_1		
a	**b**	**c**
1	1	1
1	2	2
2	0	2

*****

r_2	
a	**d**
1	1
0	1
2	0

$r_1 * r_2$			
a	**b**	**c**	**d**
1	1	1	1
1	2	2	1
2	0	2	0

Aufgabe 1a) – Eigenschaften von *

$\sigma_{[b]}(r_1 * r_2) = \sigma_{[b]}(r_1) * \sigma_{[b]}(r_2)$

geg. r_1: (A | …), r_2: (B | …), r_3: (C | …)

falls b nur Attribute aus A ∩ B besitzt

Bedingung:
b : a = 1

σ[b] (r_1 * r_2)			
a	b	c	d
1	1	1	1
1	2	2	1

!=

r_1		
a	b	c
1	1	1
1	2	2
2	0	2

*

r_2	
a	d
1	1
0	1
2	0

$\sigma_{[b]}(r_1 * r_2) = \sigma_{[b]}(r_1) * \sigma_{[b]}(r_2)$

geg. r_1: (A | …), r_2: (B | …), r_3: (C | …)

falls b nur Attribute aus $A \cap B$

Bedingung:
b :   a = 1

$r_1 * r_2$			
a	**b**	**c**	**d**
1	1	1	1
1	2	2	1

$\overset{!}{=}$

$\sigma[b]\ (r_1) * \sigma[b]\ (r_2)$			
a	**b**	**c**	**d**
1	1	1	1
1	2	2	1

2	0	2

r_2	
a	**d**
1	1
0	1
2	0

$\sigma_{[b]}(r_1 * r_2) = \sigma_{[b]}(r_1) * r_2$

geg. r_1: (A | …), r_2: (B | …), r_3: (C | …)

falls b nur Attribute aus A besitzt

Bedingung:
b : c = 1

r_1		
a	**b**	**c**
1	1	1
1	2	2
2	0	2

*

r_2	
a	**d**
1	1
0	1
2	0

$r_1 * r_2$			
a	**b**	**c**	**d**
1	1	1	1
1	2	2	1
2	0	2	0

AIFB

$\sigma_{[b]} (r_1 * r_2) = \sigma_{[b]} (r_1) * r_2$ geg. r_1: (A | …), r_2: (B | …), r_3: (C | …)

falls b nur Attribute aus A besitzt

Bedingung:
b : c = 1

σ[b] (r_1 * r_2)			
a	**b**	**c**	**d**
1	1	1	1

$\overset{!}{=}$

r_1		
a	**b**	**c**
1	1	1
1	2	2
2	0	2

AIFB

$\sigma_{[b]}(r_1 * r_2) = \sigma_{[b]}(r_1) * r_2$

geg. r_1: (A | …), r_2: (B | …), r_3: (C | …)

falls b nur Attribute aus A besitzt

Bedingung:
b : c = 1

σ[b] (r_1 * r_2)			
a	b	c	d
1	1	1	1

$\overset{!}{=}$

σ[b] (r_1)		
a	b	c
1	1	1

*

r_2	
a	d
1	1
0	1
2	0

AIFB

$\sigma_{[b]}(r_1 * r_2) = \sigma_{[b]}(r_1) * r_2$

geg. r_1: (A | …), r_2: (B | …), r_3: (C | …)

falls b nur Attribute aus A besitzt

Bedingung:
b : c = 1

σ[b] (r_1 * r_2)			
a	**b**	**c**	**d**
1	1	1	1

!
=

σ[b] (r_1) * r_2			
a	**b**	**c**	**d**
1	1	1	1

*

r_2	
a	**d**
1	1
0	1
2	0

$\pi_{[a,e]}(r_2 * r_3)$ und $\pi_{[a]} r_2 * \pi_{[e]} r_3$

$\pi_{[a,e]}(r_2 * r_3)$:

r_2	
a	d
1	1
0	1
2	0

*

r_3		
c	d	e
1	1	0
0	1	1
2	1	0
2	2	1

⇨

$r_2 * r_3$			
a	c	d	e
1	1	1	0
1	0	1	1
1	2	1	0
0	1	1	0
0	0	1	1
0	2	1	0

$\pi_{[a,e]}\ (r_2 * r_3)$ und $\pi_{[a]}\ r_2 * \pi_{[e]}\ r_3$

$\pi_{[a,e]}\ (r_2 * r_3)$:

r_2	
a	**d**
1	1
0	1
2	0

*

r_3		
c	**d**	**e**
1	1	0
0	1	1
2	1	0
2	2	1

$\Rightarrow$

$\pi_{[a,e]}\ (r_2 * r_3)$:	
a	**e**
1	0
1	1
0	0
0	1

AIFB

$\pi_{[a,e]} (r_2 * r_3)$ und $\pi_{[a]} r_2 * \pi_{[e]} r_3$

$\pi_{[a]} r_2 * \pi_{[e]} r_3$

r_2	
a	d
1	1
0	1
2	0

*

r_3		
c	d	e
1	1	0
0	1	1
2	1	0
2	2	1

$\pi_{[a,e]}\,(r_2 * r_3)$ und $\pi_{[a]}\, r_2 * \pi_{[e]}\, r_3$

$\pi_{[a]}\, r_2 * \pi_{[e]}\, r_3$

a
1
0
2

*

e
0
1
0
1

$\pi_{[a]}\, r_2 * \pi_{[e]}\, r_3$	
a	e
1	0
1	1
0	0
0	1
2	0
2	1

$\neq$

$\pi_{[a,e]}\,(r_2 * r_3)$	
a	e
1	0
1	1
0	0
0	1

- Algebraische Eigenschaften erleichtern das Formulieren von Anfragen in der Datenbank

- Durch algebraische Umformungen kann der Aufwand für einzelne Anfragen optimiert werden (z.B. Vermeidung des kartesischen Produktes).

Join ist die Hintereinander-Ausführung der Operationen kartesischen Produkt und Selektion.

Theta-Join

$$r_1 * [Join{-}Bedingung\ \theta]\ r_2$$
$$r_1 \bowtie [Join{-}Bedingung\ \theta]\ r_2$$

Vergleich von Attributen entsprechend der Theta-Bedingung.

AIFB

Theta-Join

Mitarbeiter		
Nachname	Vorname	Geburtsdatum
Milke	Lise	3.6.1934
Huber	Karl	16.12.1964
Trunstein	Helga	30.7.1956

* [Mitarbeiter.Nachname >= Projekte.Nachname]

Projekte		
Projekt	Nachname	Vorname
Neubau	Huber	Anna
Werbung	Trunstein	Helga
Design	Kohlmeier	Johann

⇒

Mitarbeiter. Nachname	Mitarbeiter. Vorname	Geburtsdatum	Projekt	Projekte. Nachname	Projekte. Vorname
Milke	Lise	3.6.1934	Werbung	Trunstein	Helga
Huber	Karl	16.12.1964	Neubau	Huber	Anna
Huber	Karl	16.12.1964	Werbung	Trunstein	Helga
Huber	Karl	16.12.1964	Design	Kohlmeier	Johann
Trunstein	Helga	30.7.1956	Werbung	Trunstein	Helga

Alphabetische Sortierung: „A“ ist größer als „Z“

Spezialfälle des Theta-Joins

Equi-Join

- Beim Equi-Join wird als erstes das kartesische Produkt gebildet.
- Dann erfolgt die Selektion mit der Bedingung, dass der Inhalt bestimmter Spalten identisch sein muss.
- Der Equi-Join ist ein allgemeiner Verbund mit einer Formel der Form A = B.

Spezialfälle des Theta-Joins

Natural Join

- entspricht einem Equi-Join
- zusätzlich werden gleiche Spalten (Projektion) ausgeblendet
- ist kommutativ und assoziativ

Spezialfälle des Theta-Joins

Natural Join / Equi-Join

Schüler

SNr	Vorname	Name
4711	Paul	Müller
0815	Erich	Schmidt
7472	Sven	Lehmann
1234	Olaf	Müller
2313	Jürgen	Paulsen

Kurse

SNr	KNr	Fehlstunden	Punkte
0815	03	0	12
4711	03	12	03
1234	23	3	14
0987	09	9	09

→

J_{SNr}(Schüler, Kurs)

SNr	Vorname	Name	KNr	Fehlstunden	Punkte
0815	Erich	Schmidt	03	0	12
4711	Paul	Müller	03	12	03
1234	Olaf	Müller	23	3	14

Left-Outer-Theta-Join

- Mit einem Left Join wird eine sogenannte linke Inklusionsverknüpfung erstellt.
- Linke Inklusionsverknüpfungen schließen alle Datensätze aus der ersten (linken) Tabelle ein, auch wenn keine entsprechenden Werte für Datensätze in der zweiten Tabelle existieren.

Right-Outer-Theta-Join

- Mit einem Right Join wird eine sogenannte rechte Inklusionsverknüpfung erstellt.
- Rechte Inklusionsverknüpfungen schließen alle Datensätze aus der zweiten (rechten) Tabelle ein, auch wenn keine entsprechenden Werte für Datensätze in der ersten Tabelle existieren.

Left-Outer-Theta-Join

Schüler

SNr	Vorname	Name
4711	Paul	Müller
0815	Erich	Schmidt
7472	Sven	Lehmann
1234	Olaf	Müller
2313	Jürgen	Paulsen

Kurse

SNr	KNr	Fehlstunden	Punkte
0815	03	0	12
4711	03	12	03
1234	23	3	14
0987	09	9	09

→

J_{SNr}(Schüler, Kurs)

SNr	Vorname	Name	KNr	Fehlstunden	Punkte
0815	Erich	Schmidt	03	0	12
4711	Paul	Müller	03	12	03
7472	Sven	Lehmann			
1234	Olaf	Müller	23	3	14
2313	Jürgen	Paulsen			

Right-Outer-Theta-Join

Schüler

SNr	Vorname	Name
4711	Paul	Müller
0815	Erich	Schmidt
7472	Sven	Lehmann
1234	Olaf	Müller
2313	Jürgen	Paulsen

Kurse

SNr	KNr	Fehlstunden	Punkte
0815	03	0	12
4711	03	12	03
1234	23	3	14
0987	09	9	09

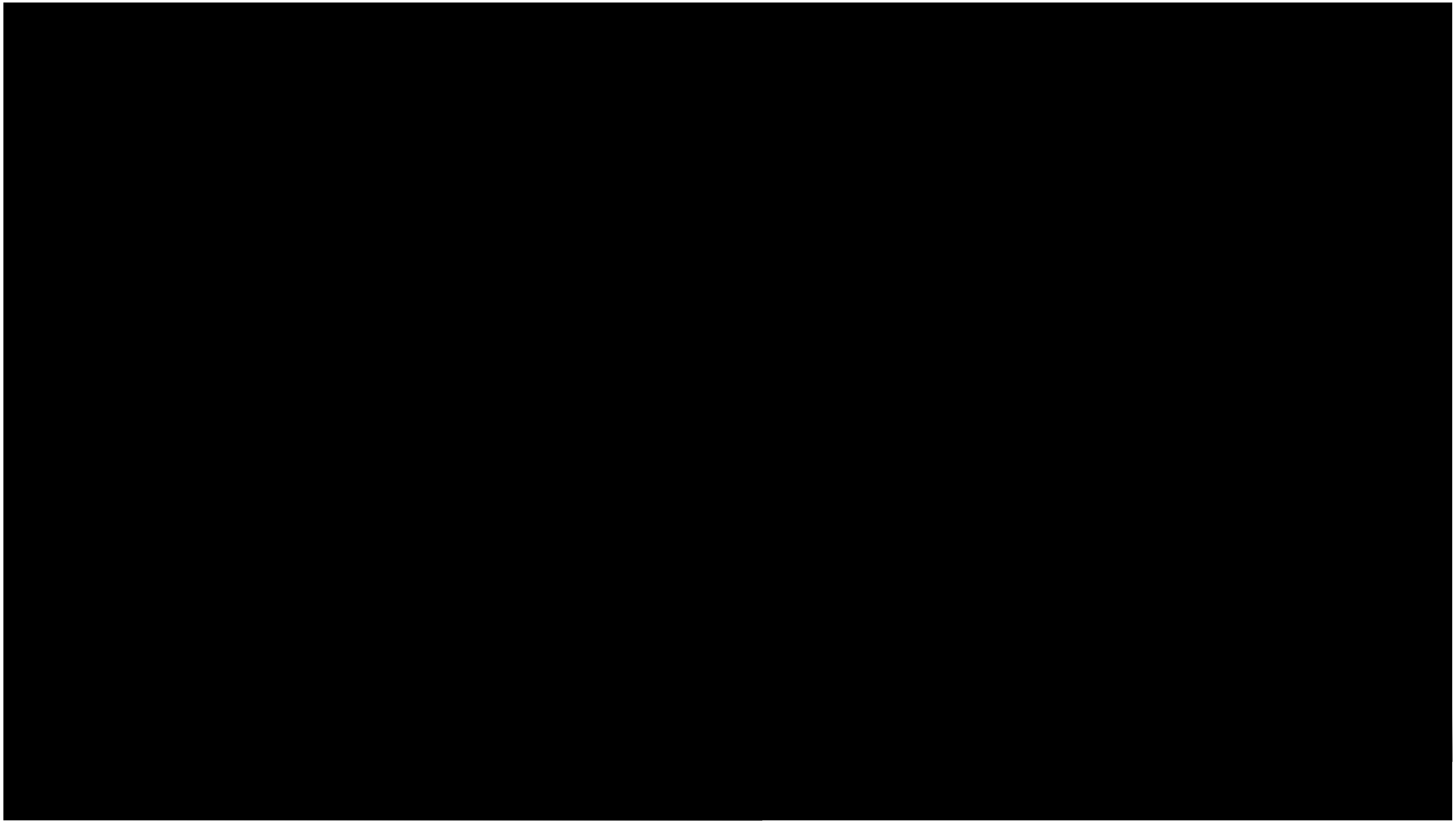
→

J_{SNr}(Schüler, Kurs)

SNr	Vorname	Name	KNr	Fehlstunden	Punkte
0815	Erich	Schmidt	03	0	12
4711	Paul	Müller	03	12	03
1234	Olaf	Müller	23	3	14
0987			09	9	09

Aufgabe 2 – Arten von Joins

Left-Semi-Join



AIFB

Full-Outer-Theta-Join

- Eine Kombination von Left Outer Join und Right Outer Join.

Schüler

SNr	Vorname	Name
4711	Paul	Müller
0815	Erich	Schmidt
7472	Sven	Lehmann
1234	Olaf	Müller
2313	Jürgen	Paulsen

Kurse

SNr	KNr	Fehlstunden	Punkte
0815	03	0	12
4711	03	12	03
1234	23	3	14
0987	09	9	09

→

J_{SNr}(Schüler, Kurs)

SNr	Vorname	Name	KNr	Fehlstunden	Punkte
0815	Erich	Schmidt	03	0	12
4711	Paul	Müller	03	12	03
7472	Sven	Lehmann			
1234	Olaf	Müller	23	3	14
2313	Jürgen	Paulsen			
0987			09	9	09

Gegeben seien folgende Relationsschemata einer relationalen Datenbank:

```
lieferant: LIEFERANT(L#, LNAME, STATUS, STADT)
teil: TEIL(T#, TNAME, FARBE, GEWICHT, STADT)
projekt: PROJEKT(P#, PNAME, STADT)
lieferung: LIEFERUNG(L#, T#, P#, MENGE)
```

Formulieren Sie die folgenden Anfragen in Relationenalgebra.

```
lieferant: LIEFERANT(L#, LNAME, STATUS, STADT)
teil: TEIL(T#, TNAME, FARBE, GEWICHT, STADT)
projekt: PROJEKT(P#, PNAME, STADT)
lieferung: LIEFERUNG(L#, T#, P#, MENGE)
```

a) Liste sämtliche Lieferungen auf, deren Mengen zwischen 300 und 750 liegen.

```
lieferant: LIEFERANT(L#, LNAME, STATUS, STADT)
teil: TEIL(T#, TNAME, FARBE, GEWICHT, STADT)
projekt: PROJEKT(P#, PNAME, STADT)
lieferung: LIEFERUNG(L#, T#, P#, MENGE)
```

b) Gib alle Kombinationen von Lieferanten-, Teil- und Projektnummern aus, für die gilt: Lieferant, Teil und Projekt befinden sich in derselben Stadt.

```
lieferant: LIEFERANT(L#, LNAME, STATUS, STADT)
teil: TEIL(T#, TNAME, FARBE, GEWICHT, STADT)
projekt: PROJEKT(P#, PNAME, STADT)
lieferung: LIEFERUNG(L#, T#, P#, MENGE)
```

c) Gib die Nummern aller Teile aus, die von einem Lieferanten aus London geliefert wurden.

```
lieferant: LIEFERANT(L#, LNAME, STATUS, STADT)
teil: TEIL(T#, TNAME, FARBE, GEWICHT, STADT)
projekt: PROJEKT(P#, PNAME, STADT)
lieferung: LIEFERUNG(L#, T#, P#, MENGE)
```

d) Gib die Nummern aller Teile aus, die für ein Projekt in London von einem Lieferanten aus London geliefert wurden.

```
lieferant: LIEFERANT(L#, LNAME, STATUS, STADT)
teil: TEIL(T#, TNAME, FARBE, GEWICHT, STADT)
projekt: PROJEKT(P#, PNAME, STADT)
lieferung: LIEFERUNG(L#, T#, P#, MENGE)
```

e) Suche die Nummern von Projekten, für die von mindestens einem Lieferanten Teile geliefert wurden, der sich nicht in derselben Stadt befindet wie das Projekt.

```
lieferant: LIEFERANT(L#, LNAME, STATUS, STADT)
teil: TEIL(T#, TNAME, FARBE, GEWICHT, STADT)
projekt: PROJEKT(P#, PNAME, STADT)
lieferung: LIEFERUNG(L#, T#, P#, MENGE)
```

f) Gib die Namen aller Projekte aus, für die Lieferant L1 Teile geliefert hat.

```
lieferant: LIEFERANT(L#, LNAME, STATUS, STADT)
teil: TEIL(T#, TNAME, FARBE, GEWICHT, STADT)
projekt: PROJEKT(P#, PNAME, STADT)
lieferung: LIEFERUNG(L#, T#, P#, MENGE)
```

g) Suche alle Nummern von Teilen, die nicht für ein Projekt in London geliefert wurden.

```
lieferant: LIEFERANT(L#, LNAME, STATUS, STADT)
teil: TEIL(T#, TNAME, FARBE, GEWICHT, STADT)
projekt: PROJEKT(P#, PNAME, STADT)
lieferung: LIEFERUNG(L#, T#, P#, MENGE)
```

h) Suche die Nummern von Lieferanten außer L2, die gleiche Teile liefern wie der Lieferant L2.

»Auf Wiedersehen
und ein
schönes Wochenende!«

Angewandte Informatik I, Tutorium am Freitag
WS 2005 / 06

Henrik Simon

Übungsblatt Nr.7

Aufgabe 1:

r1		
a	**b**	**c**
1	1	1
1	2	2
2	0	2

r2	
a	**d**
1	1
0	1
2	0

r4		
a	**b**	**c**
1	2	2
1	2	3

Geg.: b1: (a>=1), b2: (c=2)

a.)

1) $\sigma_{[b1]}(\sigma_{[b2]}(r_1)) \; = \; \sigma_{[b2]}(\sigma_{[b1]}(r_1))$

$\sigma_{[a>=1]}(\sigma_{[c=2]}\, r1) =$

a	**b**	**c**
1	2	2
2	0	2

$= \sigma_{[c=2]}\,(\sigma_{[a>=1]} r1)$

Übungsblatt Nr.7

Aufgabe 1:

r1		
a	**b**	**c**
1	1	1
1	2	2
2	0	2

2) $\sigma_{[b1 \wedge b2]}(r_1) \quad = \quad \sigma_{[b1]}(\sigma_{[b2]}(r_1))$

$\sigma_{[a>=1 \wedge c=2]}$ (r1) =

a	**b**	**c**
1	2	2
2	0	2

= $\sigma_{[a>=1]}$ ($\sigma_{[c=2]}$ r1)

3) $\sigma_{[b1 \vee b2]}(r) \quad = \quad \sigma_{[b1]}(r) \cup \sigma_{[b2]}(r)$

$\sigma_{[a>=1 \vee c=2]}$ (r1) =

a	**b**	**c**
1	1	1
1	2	2
2	0	2

= $\sigma_{[a>=1]}$ (r1) $\cup$ $\sigma_{[c=2]}$ (r1)

Übungsblatt Nr.7

Aufgabe 1:

4) $\sigma_{[b]}(r_1 \cup r_2) = \sigma_{[b]}(r_1) \cup \sigma_{[b]}(r_2)$

r1		
a	**b**	**c**
1	1	1
1	2	2
2	0	2

r4		
a	**b**	**c**
1	2	2
1	2	3

$\sigma_{[c=2]}(r1 \cup r4) =$

a	**b**	**c**
1	2	2
2	0	2

$= \sigma_{[c=2]}(r1) \cup \sigma_{[c=2]}(r4)$

Übungsblatt Nr.7

Aufgabe 1:

5) $\sigma_{[b]}(r_1 \times r_2) \quad = \begin{cases} \sigma_{[b]}(r_1) \times r_2 & \text{falls b nur Attribute aus A} \\ r_1 \times \sigma_{[b]}(r_2) & \text{falls b nur Attribute aus B} \end{cases}$

r1		
a	b	c
1	1	1
1	2	2
2	0	2

r2	
a	d
1	1
0	1
2	0

a	a‘	b	c	d
1	1	2	2	1
1	0	2	2	1
1	2	2	2	0
2	1	0	2	1
2	0	0	2	1
2	2	0	2	0
~~1~~	~~1~~	~~1~~	~~1~~	~~1~~
~~1~~	~~0~~	~~1~~	~~1~~	~~1~~
~~1~~	~~2~~	~~1~~	~~1~~	~~0~~

$\sigma_{[c=2]}(r1 \times r2) = \sigma_{[c=2]}(r1) \times r2 =$

a	a‘	b	c	d
1	1	2	2	1
1	0	2	2	1
1	2	2	2	0
2	1	0	2	1
2	0	0	2	1
2	2	0	2	0

Das Attribut c
kommt nur in r1 vor!

Übungsblatt Nr.7

Aufgabe 1:

r1		
a	b	c
1	1	1
1	2	2
2	0	2

r2	
a	d
1	1
0	1
2	0

1) $r_1 * r_2 = r_2 * r_1$

r1 * r2 =

a	**b**	**c**	**d**
1	1	1	1
1	2	2	1
2	0	2	0

= r2 * r1

Übungsblatt Nr.7

Aufgabe 1:

r1		
a	b	c
1	1	1
1	2	2
2	0	2

r2	
a	d
1	1
0	1
2	0

r4		
a	b	c
1	2	2
1	2	3

2) $(r_1 * r_2) * r_3 = r_1 * (r_2 * r_3) = r_1 * r_2 * r_3$

(r1 * r2) * r4 =

a	b	c	d
1	2	2	1

= r1 * (r2 * r4) = r1 * r2 * r4

(r1 * r2):

a	b	c	d
1	1	1	1
1	2	2	1
2	0	2	0

(r2 * r4):

a	b	c	d
1	2	2	1
1	2	3	1

Übungsblatt Nr.7

Aufgabe 1:

3) $A \cap B = \varnothing \quad \Rightarrow r_1 * r_2 = r_1 \times r_2$

r2	
a	**d**
1	1
0	1
2	0

r4		
a	**b**	**c**
1	2	2
1	2	3

$(\pi_{[d]}\ r2) * r4 =$

a	**b**	**c**	**d**
1	2	2	1
1	2	3	1
1	2	2	0
1	2	3	0

= r2 x r4

d
1
0

$(A = \{d\} \cap B = \{a,b,c\}) = \{\ \}$

Übungsblatt Nr.7

Aufgabe 1:

4) $r_1 = \emptyset \quad \Rightarrow r_1 * r_2 = \emptyset$

r2	
a	d
1	1
0	1
2	0

$r2 * r5 = \emptyset$

$r5 = \emptyset$

Übungsblatt Nr.7

Aufgabe 1:

5) **A = B** $\Rightarrow r_1 * r_2 = r_1 \cap r_2$

r1		
a	**b**	**c**
1	1	1
1	2	2
2	0	2

r4		
a	**b**	**c**
1	2	2
1	2	3

r1 * r4 =

a	**b**	**c**
1	2	2

= r1 ∩ r4

(A = {a,b,c}) = (B = {a,b,c})

Übungsblatt Nr.7

Aufgabe 1:

6) $r_1 * r_1 = r_1$

r1		
a	**b**	**c**
1	1	1
1	2	2
2	0	2

r1 * r1 = r1 $\cap$ r1 = r1

A = {a,b,c} = A

⬇

Sonderfall von 5.)

Übungsblatt Nr.7

Aufgabe 1:

7) $\sigma_{[b]}(r_1 * r_2) = \begin{cases} \sigma_{[b]}(r_1) * \sigma_{[b]}(r_2) & \text{falls b nur Attribute aus } A \cap B \text{ besitzt} \\ \sigma_{[b]}(r_1) * r_2 & \text{falls b nur Attribute aus A besitzt} \\ r_1 * \sigma_{[b]}(r_2) & \text{falls b nur Attribute aus B besitzt} \end{cases}$

r1		
a	b	c
1	1	1
1	2	2
2	0	2

r2	
a	d
1	1
0	1
2	0

$\sigma_{[c=2]}(r1 * r2) = \sigma_{[c=2]}(r1) * r2$

a	b	c	d
~~1~~	~~1~~	~~1~~	~~1~~
1	2	2	1
2	0	2	0

Das Attribut c
kommt nur in r1 vor!

a	b	c	d
1	2	2	1
2	0	2	0

a	b	c	d
1	2	2	1
2	0	2	0

Übungsblatt Nr.7

r2	
a	**d**
1	1
0	1
2	0

r4		
a	**b**	**c**
1	2	2
1	2	3

Aufgabe 1:

b.) Zeigen Sie anhand eines Beispiels, dass die Reihenfolge von Projektion und Join wichtig ist.

Bsp.:

$(\pi_{[d]}\ r2) * r4 \neq \pi_{[d]}\ (r2 * r4) =$

d
1

a	**b**	**c**	**d**
1	2	2	1
1	2	3	1
1	2	2	0
1	2	3	0

aus Folie 7

Übungsblatt Nr.7

Aufgabe 1:

c.) Wofür könnte man die algebraischen Eigenschaften bei der Implementierung von DBMS verwenden?

Ziel: Performance

Bsp.: Aufgabe 1. Nr. 7

7) $$\sigma_{[b]}(r_1 * r_2) = \begin{cases} \sigma_{[b]}(r_1) * \sigma_{[b]}(r_2) & \text{falls b nur Attribute aus } A \cap B \text{ besitzt} \\ \sigma_{[b]}(r_1) * r_2 & \text{falls b nur Attribute aus A besitzt} \\ r_1 * \sigma_{[b]}(r_2) & \text{falls b nur Attribute aus B besitzt} \end{cases}$$

r2	
a	d
1	1
0	1
2	0

r4		
a	b	c
1	2	2
1	2	3

Übungsbla

Theta-Join: (θ -Verbund)

- Verbindung von Relationen bezüglich beliebiger Attribute,
- bei beliebigen Vergleichsmöglichkeiten mit Prädikat θ

Schreibweise: $r_1 * [Join{-}Bedingung\ \theta]\ r_2$

oder auch: $r_1 \bowtie_{[Join-Bedingung\ \theta]} r_2$

Definition: $r_1 *_{[\theta]} r_2 = \sigma_{[\theta]}(r_1 \times r_2)$

- In die Ergebnisrelation werden alle Tupel aus r_1 x r_2 übernommen, für die das Prädikat θ durch Einsetzen der Werte der Tupel in den entsprechenden Variablen den Wert „wahr" zurückliefert.

Aufgabe 2:

Theta-Join:

Bsp.:

$r2 *_{[d = 0]} r4 = \sigma_{[d = 0]}(r2 \times r4) = \sigma_{[d = 0]}(r2) \times r4 =$

a	**a‘**	**b**	**c**	**d**
2	1	2	2	0
2	1	2	3	0

Eigenschaften von σ

a	**d**
2	0

r2	
a	d
1	1
0	1
2	0

r4		
a	b	c
1	2	2
1	2	3

Übungsblatt Nr.7

Aufgabe 2:

Equi-Join:

Falls θ ausschließlich auf Gleichheit unter den Attributen aus A,B prüft, wird die Verbindung auch **Equi-Join** genannt *(Spezialfall des Theta-Joins)*

Bsp.:

$r2 *_{[d = 0]} r4 = \sigma_{[d = 0]}(r2 \times r4) = \sigma_{[d = 0]}(r2) \times r4 =$

a	**a'**	**b**	**c**	**d**
2	1	2	2	0
2	1	2	3	0

Eigenschaften von σ

a	**d**
2	0

r2	
a	d
1	1
0	1
2	0

r4		
a	b	c
1	2	2
1	2	3

Übungsblatt Nr.7

Aufgabe 2:

(Left) Semi-Join

$$r_1 \ltimes r_2 = \pi_{[A]}(r_1 * r_2)$$

- wird benutzt, wenn die Information benötigt wird, welche Tupel der einen Relation (r_1) erfolgreich mit Tupeln der anderen Relation (r_2) verbunden werden können
- Anwendung in verteilten Datenbanken;
  Ziel: weniger Daten (d.h. nicht komplette Relationen) übertragen

Left Semi-Join:

Bsp.:

r2 $\ltimes$ r4 = $\pi_{[a,d]}$ (r2 * r4) =

a	**d**
1	1
0	1

r2	
a	**d**
1	1
0	1
2	0

r4		
a	**b**	**c**
1	2	2
1	2	3

Left-Outer-Theta-Join

$$\mathbf{r_1} \mathbin{⟕}_{[\theta]} \mathbf{r_2} = r_1 *_{[\theta]} r_2 \cup \underbrace{(r_1 \setminus \pi_{[A]}(r_1 *_{[\theta]} r_2)) \times \{NULL\}^{|B|}}$$

- nicht verbindbare Tupel der linksstehenden Relation (r_1) werden trotzdem ins Join-Ergebnis übernommen, und
- solche Tupel von r_1 werden mit $\underbrace{\text{(NULL,…, NULL)}}_{|B|\text{ mal}}$ verkettet

Aufgabe 2:

Left-Outer-Theta-Join:

<u>Bsp.:</u>

$r2 \mathbin{⟕}_{[r2.a\ =\ r4.a]} r4 =$

a	**a‘**	**b**	**c**	**d**
1	1	2	2	1
1	1	2	3	1
0	Null	Null	Null	1
2	Null	Null	Null	0

r2	
a	d
1	1
0	1
2	0

r4		
a	b	c
1	2	2
1	2	3

Jbu

Right-Outer-Theta-Join

$$r_1 \; ⟖_{[\theta]} \; r_2 = r_1 *_{[\theta]} r_2 \;\cup\; \underbrace{\{NULL\}^{|A|} \times (r_2 \setminus \pi_{[B]}(r_1 *_{[\theta]} r_2))}$$

- nicht verbindbare Tupel der rechtsstehenden Relation (r_2) werden trotzdem ins Join-Ergebnis übernommen, und
- solche Tupel von r_2 werden mit $\underbrace{\text{(NULL,…, NULL)}}_{|A|\text{ mal}}$ verkettet

Aufgabe 2:

Right-Outer-Theta-Join:

Bsp.:

$r2 \; ⟖_{[r2.a = r4.a]} \; r4 =$

a	**a‘**	**b**	**c**	**d**
1	1	2	1	1
1	1	2	3	1

r2	
a	**d**
1	1
0	1
2	0

r4		
a	**b**	**c**
1	2	2
1	2	3

Right-Outer-Theta-Join

$$\mathbf{r_1} ⟖_{[\theta]} \mathbf{r_2} = r_1 *_{[\theta]} r_2 \cup \underbrace{\{NULL\}^{|A|} \times (r_2 \setminus \pi_{[B]}(r_1 *_{[\theta]} r_2))}$$

- nicht verbindbare Tupel der rechtsstehenden Relation (r_2) werden trotzdem ins Join-Ergebnis übernommen, und
- solche Tupel von r_2 werden mit $\underbrace{\text{(NULL,…, NULL)}}_{|A| \text{ mal}}$ verkettet

Aufgabe 2:

Right-Outer-Theta-Join:

<u>Bsp.:</u>

r2 ⟖ $_{[c = 2]}$ r4 =

a	**a‘**	**b**	**c**	**d**
1	1	2	2	1
0	1	2	2	1
2	1	2	2	0
Null	1	2	3	Null

r2	
a	d
1	1
0	1
2	0

r4		
a	b	c
1	2	2
1	2	3

Jbungsblatt Nr 7

Aufgabe 2:

Full-Outer-Theta-Join:

(Full) Outer-Theta-Join

$$r_1 \bowtie_{[\theta]} r_2 = r_1 *_{[\theta]} r_2 \cup (r_1 \setminus \pi_{[A]}(r_1 *_{[\theta]} r_2)) \times \{NULL\}^{|B|} \cup \{NULL\}^{|A|} \times (r_2 \setminus \pi_{[B]}(r_1 *_{[\theta]} r_2))$$

- nicht verbindbare Tupel sowohl der linksstehenden Relation (r_1) als auch der rechtsstehenden Relation (r_2) werden trotzdem ins Join-Ergebnis übernommen

Bsp.:

$r2 \bowtie_{[r2.a = r4.a \wedge c = 2]} r4 =$

a	**a‘**	**b**	**c**	**d**
1	1	2	2	1
0	Null	Null	Null	1
2	Null	Null	Null	0
Null	1	2	3	Null

Übungsblatt Nr.7

```
lieferant: LIEFERANT(L#, LNAME, STATUS, STADT)
teil: TEIL(T#, TNAME, FARBE, GEWICHT, STADT)
projekt: PROJEKT(P#, PNAME, STADT)
lieferung: LIEFERUNG(L#, T#, P#, MENGE)
```

Aufgabe 3:

a.) Liste sämtliche Lieferungen auf, deren Mengen zwischen 300 und 750 liegen.

$$\sigma_{[\mathbf{Menge} > \mathbf{300} \ \wedge \ \mathbf{Menge} < \mathbf{750}]}\mathbf{(lieferung)}$$

Übungsblatt Nr.7

Aufgabe 3:

```
lieferant: LIEFERANT(L#, LNAME, STATUS, STADT)
teil: TEIL(T#, TNAME, FARBE, GEWICHT, STADT)
projekt: PROJEKT(P#, PNAME, STADT)
lieferung: LIEFERUNG(L#, T#, P#, MENGE)
```

b.) Gib alle Kombinationen von Lieferanten-, Teil- und Projektnummern aus, für die gilt: Lieferant, Teil und Projekt befinden sich in derselben Stadt.

$$\pi_{[L\#,\ T\#,\ P\#]}\ \textbf{(lieferant * (teil * projekt))}$$

Übungsblatt Nr.7

```
lieferant: LIEFERANT(L#, LNAME, STATUS, STADT)
teil: TEIL(T#, TNAME, FARBE, GEWICHT, STADT)
projekt: PROJEKT(P#, PNAME, STADT)
lieferung: LIEFERUNG(L#, T#, P#, MENGE)
```

Aufgabe 3:

c.) Gib die Nummern aller Teile aus, die von einem Lieferanten aus London geliefert wurden.

$$\pi_{[T\#]}\ \textbf{(lieferung * (}\sigma_{[\text{Stadt} = \text{"London“}]}\ \textbf{(lieferant))}$$

Übungsblatt Nr.7

Aufgabe 3:

```
lieferant: LIEFERANT(L#, LNAME, STATUS, STADT)
teil: TEIL(T#, TNAME, FARBE, GEWICHT, STADT)
projekt: PROJEKT(P#, PNAME, STADT)
lieferung: LIEFERUNG(L#, T#, P#, MENGE)
```

d.) Gib die Nummern aller Teile aus, die für ein Projekt in London von einem Lieferanten aus London geliefert wurden.

$$\pi_{[T\#]}\ \textbf{(lieferung * (}\sigma_{[Stadt\ =\ "London"]}\ \textbf{(projekt) * }\sigma_{[Stadt\ =\ "London"]}\ \textbf{(lieferant)))}$$

Übungsblatt Nr.7

```
lieferant: LIEFERANT(L#, LNAME, STATUS, STADT)
teil: TEIL(T#, TNAME, FARBE, GEWICHT, STADT)
projekt: PROJEKT(P#, PNAME, STADT)
lieferung: LIEFERUNG(L#, T#, P#, MENGE)
```

Aufgabe 3:

e.) Suche die Nummern von Projekten, für die von mindestens einem Lieferanten Teile geliefert wurden, der sich nicht in derselben Stadt befindet wie das Projekt.

$$\pi_{[P\#]}\ ((\text{projekt} *_{[\text{projekt.STADT} \neq \text{lieferant.STADT}]} \text{lieferant}) * \text{lieferung})$$

Übungsblatt Nr.7

```
lieferant: LIEFERANT(L#, LNAME, STATUS, STADT)
teil: TEIL(T#, TNAME, FARBE, GEWICHT, STADT)
projekt: PROJEKT(P#, PNAME, STADT)
lieferung: LIEFERUNG(L#, T#, P#, MENGE)
```

Aufgabe 3:

f.) Gib die Namen aller Projekte aus, für die Lieferant L1 Teile geliefert hat.

$$\pi_{[PNAME]}\ \textbf{(projekt * (}\sigma_{[L\# = "L1"]}\ \textbf{(lieferung)))}$$

Übungsblatt Nr.7

Aufgabe 3:

```
lieferant: LIEFERANT(L#, LNAME, STATUS, STADT)
teil: TEIL(T#, TNAME, FARBE, GEWICHT, STADT)
projekt: PROJEKT(P#, PNAME, STADT)
lieferung: LIEFERUNG(L#, T#, P#, MENGE)
```

g.) Suche alle Nummern von Teilen, die nicht für ein Projekt in London geliefert wurden.

$$\pi_{[T\#]}\ \textbf{(teil)} \setminus \pi_{[T\#]}\ (\sigma_{[STADT=\text{"LONDON"}]}\ \textbf{(projekt * lieferung))}$$

Die Mengen müssen kompatibel sein, deshalb jeweils Projektionen auf T#

Übungsblatt Nr.7

```
lieferant: LIEFERANT(L#, LNAME, STATUS, STADT)
teil: TEIL(T#, TNAME, FARBE, GEWICHT, STADT)
projekt: PROJEKT(P#, PNAME, STADT)
lieferung: LIEFERUNG(L#, T#, P#, MENGE)
```

Aufgabe 3:

h.) Suche die Nummern von Lieferanten außer L2, die gleiche Teile liefern wie der Lieferant L2.

$$\underbrace{\pi_{[L\#]}\ (\text{lieferung} * \pi_{[T\#]}\ (\sigma_{[L\# = "L2"]}\ (\text{lieferung})))}_{\text{Alle L\#, welche die Teile von L2 liefern können. L2 ist hier auch dabei (*)!}}\ \setminus\ \pi_{[L\#]}\ (\sigma_{[L\# = "L2"]}\ (\text{lieferung}))$$

Alle L#, welche die Teile von L2 liefern können. L2 ist hier auch dabei (*)!

Wird wegen (*) aus der Ergebnismenge genommen