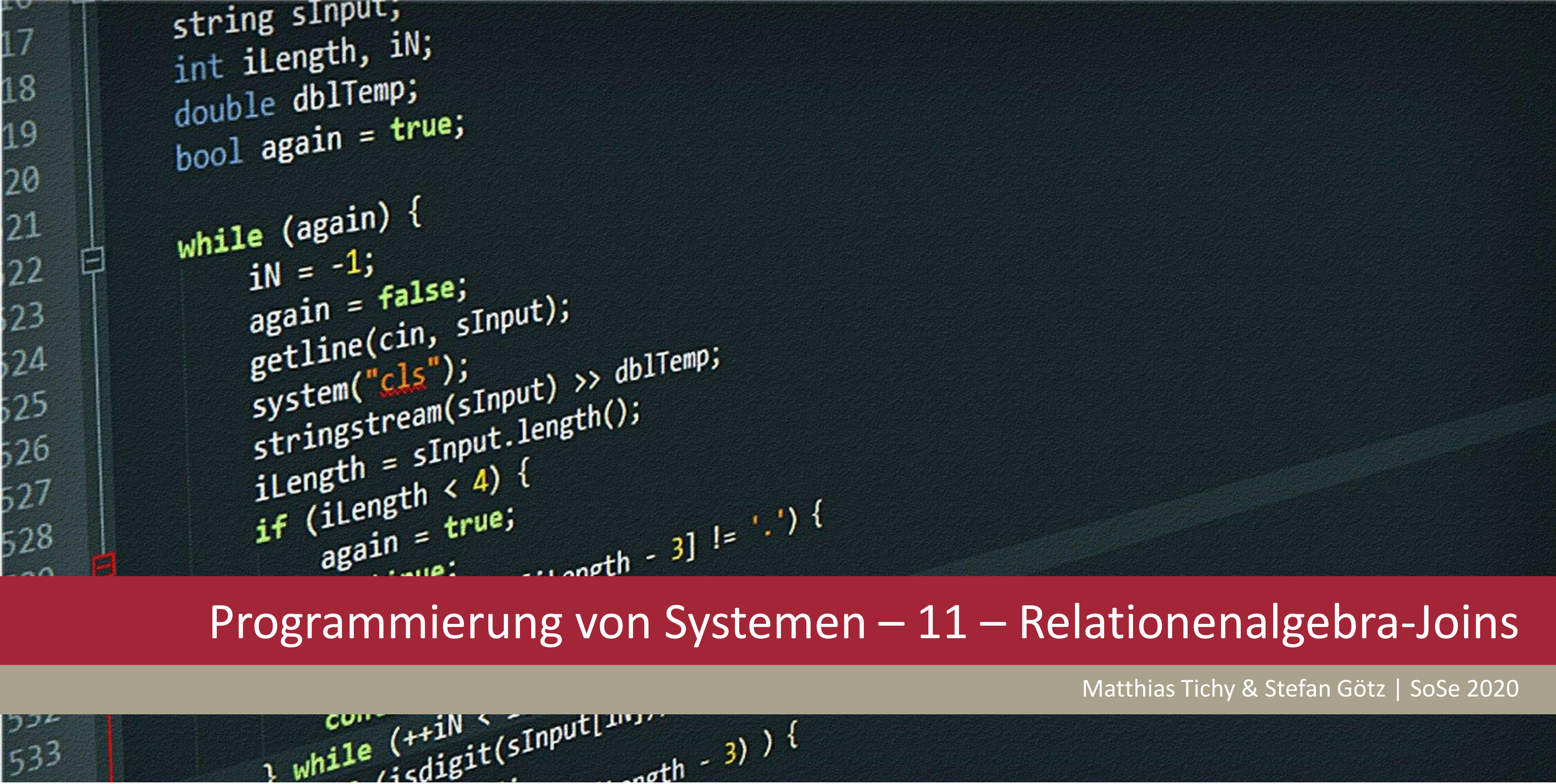


Programmierung von Systemen – 11 – Relationenalgebra-Joins

Matthias Tichy & Stefan Götz | SoSe 2020

Software Engineering
Programming Languages

ulm university universität uulm

Ziele

- Operatoren der Relationenalgebra anwenden können, um Anfragen zu formulieren

- Joins verstehen

Abgeleiteter Operator $\bowtie$

$$R \bowtie_F S$$

- steht für den Verbund (join) der Relationen R und S
  unter Verwendung der Verbund-Bedingung F
- Die Join-Bedingung kann auch mittels <, >, <=, >=, <>
  sowie unter Verwendung von $\wedge$, $\vee$ und $\neg$ formuliert werden.
- $R \bowtie_F S$ semantisch äquivalent zu $\sigma_F (R \times S)$

$\bowtie$-Ausführungslogik („nested loop“-Algorithmus)

```
foreach Tupel x in R do
    foreach Tupel y in S do
        if R.Joinattribut = S.Joinattribut then erzeuge Resultattupel x || y
    end
end
```

Beispiel zu ⋈

„Gib alle Aufträge aus, und zwar mit Auftragsnummer, Kundennummer, Kundenname, Auftragsposition, Teilenummer, Farbe und Anzahl“

Gewünschtes Resultat (strukturell):

AuftrNr	KdNr	KdName	Pos	TeileNr	Farbe	Anzahl
15	177	Zwack	1	AAG	1	5
15	177	Zwack	2	AAG	2	2
15	177	Zwack	3	CAH	1	1
16	123	Babel	1	CKK	1	2

Benötigte Relationen:

Auftraege		
AuftrNr	KdNr	AuftrDatum

Kunden		
KdNr	KdName	KdStadt

AuftragsPos				
AuftrNr	Pos	TeileNr	Farbe	Anzahl

Algebra-Ausdruck (mit Join)?

Beispiel zu ⋈

„Gib alle Aufträge aus, und zwar mit Auftragsnummer, Kundennummer, Kundenname, Auftragsposition, Teilenummer, Farbe und Anzahl“

Gewünschtes Resultat (strukturell):

AuftrNr	KdNr	KdName	Pos	TeileNr	Farbe	Anzahl
15	177	Zwack	1	AAG	1	5
15	177	Zwack	2	AAG	2	2
15	177	Zwack	3	CAH	1	1
16	123	Babel	1	CKK	1	2

$\bowtie_F$

Benötigte Relationen:

Auftraege		
AuftrNr	KdNr	AuftrDatum

Kunden		
KdNr	KdName	KdStadt

AuftragsPos				
AuftrNr	Pos	TeileNr	Farbe	Anzahl

Algebra-Ausdruck (mit Join)?

$$\pi_{\{AuftrNr,\, KdNr,\, KdName,\, Pos,\, TeileNr,\, Farbe,\, Anzahl\}} \Big(\big(Auftraege \bowtie_{Auftraege.KdNr = Kunden.KdNr} Kunden\big) \bowtie_{Auftraege.AuftrNr = AuftragsPos.AuftrNr} AuftragsPos \Big)$$

Beispiel zu $\bowtie$

„Gib alle Mitarbeiter (PersNr, Name, Vorname, Wohnort) aus der Abteilung 'Verkauf' aus, die mehr als 2.500 verdienen“

Welche Relationen werden benötigt?

Welche Algebra-Operationen werden benötigt?

Beispiel zu $\bowtie$

„Gib alle Mitarbeiter (PersNr, Name, Vorname, Wohnort) aus der Abteilung 'Verkauf' aus, die mehr als 2.500 verdienen“

Welche Relationen werden benötigt?

Mitarbeiter, Abteilung

Welche Algebra-Operationen werden benötigt?

$\sigma, \pi, (\bowtie = \sigma(\times))$

Beispiel zu $\bowtie$

Welche der folgenden Ausdrücke liefern das gewünschte Resultat?

$$\pi_{\{PersNr,\ Name,\ Vorname,\ Wohnort\}}\ (\sigma_{Gehalt > 2500\ \wedge\ AbtName='Verkauf'}\ (Mitarbeiter \bowtie_{Mitarbeiter.AbtNr\ =\ Abteilungen.AbtNr}\ Abteilungen))$$

Antwort:

$$\sigma_{Gehalt > 2500\ \wedge\ AbtName='Verkauf'}\ (\pi_{\{PersNr,\ Name,\ Vorname,\ Wohnort\}}\ (Mitarbeiter \bowtie_{Mitarbeiter.AbtNr\ =\ Abteilungen.AbtNr}\ Abteilungen))$$

Antwort:

$$\pi_{\{PersNr,\ Name,\ Vorname,\ Wohnort\}}\ ((\sigma_{Gehalt > 2500}\ Mitarbeiter) \bowtie_{Mitarbeiter.AbtNr\ =\ Abteilungen.AbtNr}\ (\sigma_{AbtName='Verkauf'}\ Abteilungen))$$

Antwort:

Beispiel zu $\bowtie$

Welche der folgenden Ausdrücke liefern das gewünschte Resultat?

$$\pi_{\{PersNr,\ Name,\ Vorname,\ Wohnort\}}(\sigma_{Gehalt > 2500 \wedge AbtName='Verkauf'}(Mitarbeiter \bowtie_{Mitarbeiter.AbtNr = Abteilungen.AbtNr} Abteilungen))$$

Antwort: ja

$$\sigma_{Gehalt > 2500 \wedge AbtName='Verkauf'}(\pi_{\{PersNr,\ Name,\ Vorname,\ Wohnort\}}(Mitarbeiter \bowtie_{Mitarbeiter.AbtNr = Abteilungen.AbtNr} Abteilungen))$$

Antwort: nein, da nach Projektion 'Gehalt' u. 'AbtName' weg

$$\pi_{\{PersNr,\ Name,\ Vorname,\ Wohnort\}}((\sigma_{Gehalt > 2500} Mitarbeiter) \bowtie_{Mitarbeiter.AbtNr = Abteilungen.AbtNr} (\sigma_{AbtName='Verkauf'} Abteilungen))$$

Antwort: ja

Beispiel zu ⋈

Welche der folgenden Ausdrücke liefern das gewünschte Resultat?

$$\pi_{\{PersNr,\ Name,\ Vorname,\ Wohnort\}}$$
$$(\text{Mitarbeiter} \bowtie_{Mitarbeiter.AbtNr\ =\ Abteilungen.AbtNr} (\sigma_{Gehalt\ >\ 2500\ \wedge\ AbtName='Verkauf'}\ \text{Abteilungen}))$$

Antwort:

$$(\pi_{\{PersNr,\ Name,\ Vorname,\ Wohnort\}} (\sigma_{Gehalt\ >\ 2500}\ \text{Mitarbeiter}))$$
$$\bowtie_{Mitarbeiter.AbtNr\ =\ Abteilungen.AbtNr} (\sigma_{AbtName='Verkauf'}\ \text{Abteilungen})$$

Antwort:

Beispiel zu $\bowtie$

Welche der folgenden Ausdrücke liefern das gewünschte Resultat?

$$\pi_{\{\text{PersNr, Name, Vorname, Wohnort}\}}$$
$$(\text{Mitarbeiter} \bowtie_{\text{Mitarbeiter.AbtNr = Abteilungen.AbtNr}} (\sigma_{\text{Gehalt} > 2500 \land \text{AbtName='Verkauf'}} \text{Abteilungen}))$$

Antwort:

nein, 'Gehalt' nicht in 'Abteilungen'

$$(\pi_{\{\text{PersNr, Name, Vorname, Wohnort}\}} (\sigma_{\text{Gehalt} > 2500} \text{Mitarbeiter}))$$
$$\bowtie_{\text{Mitarbeiter.AbtNr = Abteilungen.AbtNr}} (\sigma_{\text{AbtName='Verkauf'}} \text{Abteilungen})$$

Antwort:

nein, 'AbtNr' nach π nicht mehr da

Spezieller Join-Operator

$R \bowtie S$

- steht für den natürlichen Verbund (natural join)
- Unterscheidet sich vom „normalen“ Join nur durch das fehlende Join-Prädikat
- Wirkungsweise
  - Automatische Ableitung der Join-Bedingung aus den beteiligten Relationen
  - Alle gleichnamigen Attribute in R und S werden auf Gleichheit verglichen
  - Auf diese Weise entstehende „redundante“ (da inhaltsgleiche) Attribute werden entfernt
  - d.h. $sch(R \bowtie S) = sch(R) \cup sch(S)$

„normaler“ Join: $R \bowtie_F S$

Beispiele zu Natural Join

sch(R)	sch(S)	sch(R ⋈ S)	R ⋈ S ist äquivalent zu
ABC	ABF	**ABCF**	$\pi_{\{A:R.A,\ B:R.B,\ C,\ F\}} R \bowtie_{R.A=S.A \wedge R.B=S.B} S$
AB	AB		
AB	CD		

Beispiele zu Natural Join

sch(R)	sch(S)	sch(R ⋈ S)	R ⋈ S ist äquivalent zu
ABC	ABF	**ABCF**	$\pi_{\{A:R.A,\ B:R.B,\ C,\ F\}} R \bowtie_{R.A=S.A \wedge R.B=S.B} S$
AB	AB	AB	HA
AB	CD	ABCD	R x S

Abgeleiteter Operator $⟕_F$

$R ⟕_F S$

- Left Outer Join, ein linksseitiger „Außen-Verbund“
- Das Resultat-Schema entspricht einem normalen Verbund

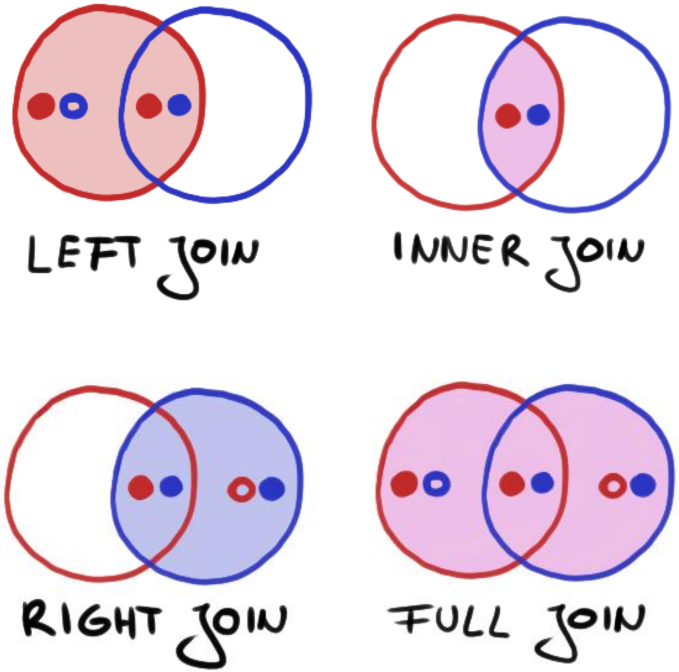
- Alle Tupel der <u>linken</u> Relation (d.h. alle R-Tupel) sind im Ergebnis enthalten

- Gibt es für ein Tupel $r \in R$ kein S-Tupel, welches das Join-Prädikat F erfüllt, dann werden die S-Attribute im Ergebnis-Tupel mit Nullwerten aufgefüllt

Joins ohne Ende

$R ⟖_F S$	Right (Outer) Join
$R ⟗_F S$	Full (Outer) Join
$R ⟕ S$	Natural Left (Outer) Join
$R ⟖ S$	Natural Right (Outer) Join
$R ⟗ S$	Natural Full (Outer) Join

Veranschaulichung



Beispiele

Gegeben:

R	A	B	C
	1	a	d
	3	c	c
	4	d	f
	5	d	b
	6	e	f

S	B	D
	a	100
	b	300
	c	400
	d	200
	e	150

T	B	D
	a	100
	d	200
	f	400
	g	120

Gesucht:

$\sigma_{D<300}$ S		

$\pi_{\{A,C\}}$ R		

$\pi_{\{C\}}$ R	

Beispiele

Gegeben:

R	A	B	C
	1	a	d
	3	c	c
	4	d	f
	5	d	b
	6	e	f

S	B	D
	a	100
	b	300
	c	400
	d	200
	e	150

T	B	D
	a	100
	d	200
	f	400
	g	120

Gesucht:

$\sigma_{D<300}$ S	B	D
	a	100
	d	200
	e	150

$\pi_{\{A,C\}}$ R	A	C
	1	d
	3	c
	4	f
	5	b
	6	f

$\pi_{\{C\}}$ R	C
	d
	c
	f
	b

Beispiele

Gegeben:

R	A	B	C
	1	a	d
	3	c	c
	4	d	f
	5	d	b
	6	e	f

S	B	D
	a	100
	b	300
	c	400
	d	200
	e	150

T	B	D
	a	100
	d	200
	f	400
	g	120

Gesucht:

S ∪ T		

S − T		

T − S		

Beispiele

Gegeben:

R	A	B	C
	1	a	d
	3	c	c
	4	d	f
	5	d	b
	6	e	f

S	B	D
	a	100
	b	300
	c	400
	d	200
	e	150

T	B	D
	a	100
	d	200
	f	400
	g	120

Gesucht:

S ∪ T	B	D
	a	100
	b	300
	c	400
	d	200
	e	150
	f	400
	g	120

S – T	B	D
	b	300
	c	400
	e	150

T – S	B	D
	f	400
	g	120

Beispiele

Gegeben:

R	A	B	C
	1	a	d
	3	c	c
	4	d	f
	5	d	b
	6	e	f

S	B	D
	a	100
	b	300
	c	400
	d	200
	e	150

T	B	D
	a	100
	d	200
	f	400
	g	120

Gesucht:

$R \bowtie_{R.B=S.B} S$					

Beispiele

Gegeben:

R	A	B	C
	1	a	d
	3	c	c
	4	d	f
	5	d	b
	6	e	f

S	B	D
	a	100
	b	300
	c	400
	d	200
	e	150

T	B	D
	a	100
	d	200
	f	400
	g	120

Gesucht:

$R \bowtie_{R.B=S.B} S$	A	R.B	C	S.B	D
	1	a	d	a	100
	3	c	c	c	400
	4	d	f	d	200
	5	d	b	d	200
	6	e	f	e	150

Beispiele

Gegeben:

R	A	B	C
	1	a	d
	3	c	c
	4	d	f

T	B	D
	a	100
	d	200

Gesucht:

R × T					

Beispiele

Gegeben:

R	A	B	C
	1	a	d
	3	c	c
	4	d	f

T	B	D
	a	100
	d	200

Gesucht:

R × T	A	R.B	C	T.B	D
	1	a	d	a	100
	1	a	d	d	200
	3	c	c	a	100
	3	c	c	d	200
	4	d	f	a	100
	4	d	f	d	200

Beispiele

Gegeben:

R	A	B	C
	1	a	d
	3	c	c
	4	d	f
	5	d	b
	6	e	f

S	B	D
	a	100
	b	300
	c	400
	d	200
	e	150

T	B	D
	a	100
	d	200
	f	400
	g	120

Gesucht:

$S \bowtie T$		

$R \bowtie T$				

$R \bowtie_{A*100=D} S$					

Beispiele

Gegeben:

R	A	B	C
	1	a	d
	3	c	c
	4	d	f
	5	d	b
	6	e	f

S	B	D
	a	100
	b	300
	c	400
	d	200
	e	150

T	B	D
	a	100
	d	200
	f	400
	g	120

Gesucht:

$S \bowtie T$	B	D
	a	100
	d	200

$R \bowtie T$	A	B	C	D
	1	a	d	100
	4	d	f	200
	5	d	b	200

$R \bowtie_{A*100=D} S$	A	R.B	C	S.B	D
	1	a	d	a	100
	3	c	c	b	300
	4	d	f	c	400

Beispiele

Gegeben:

R	A	B
	a1	b1
	a2	b2
	a3	b4
	a4	b2
	a4	b5
	a5	b1

S	B	C
	b1	c1
	b1	c2
	b2	c3
	b3	c6

Gesucht:

$R \bowtie_{R.B=S.B} S$				

Beispiele

Gegeben:

R	A	B
	a1	b1
	a2	b2
	a3	b4
	a4	b2
	a4	b5
	a5	b1

S	B	C
	b1	c1
	b1	c2
	b2	c3
	b3	c6

Gesucht:

$R \mathbin{]\!\bowtie}_{R.B=S.B} S$	A	R.B	S.B	C
	a1	b1	b1	c1
	a1	b1	b1	c2
	a2	b2	b2	c3
	a4	b2	b2	c3
	a5	b1	b1	c1
	a5	b1	b1	c2
	a3	b4	–	–
	a4	b5	–	–

Abgeleiteteter Operator $\cap$

$$R \cap S$$

- steht für die Schnittmenge
- Wie üblich: $R \cap S \equiv R - (R - S)$

Abgeleiteteter Operator ÷

$R \div S$

- steht für Division
- Ergebnisrelation enthält diejenigen Attribute aus R, die in jeder Kombination mit den Attributen aus S in R vorkommen
- Sei $D = R \div S$, dann muss gelten:
  - $sch(S) \subset sch(R)$
  - $sch(D) = sch(R) - sch(S)$
  - $t \in D \Leftrightarrow \forall s \in val(S): \langle t,s \rangle \in val(R)$
- Berechnung von D äquivalent zu:

(1) $Temp1 \leftarrow \pi_{sch(R) - sch(S)} R$

(2) $Temp2 \leftarrow \pi_{sch(R) - sch(S)} ((S \times Temp1) - R))$

(3) $D \leftarrow Temp1 - Temp2$

Beispiel zu ÷

R	A	B
	a1	b1
	a2	b1
	a3	b1
	a4	b1
	a1	b2
	a3	b2
	a2	b3
	a3	b3
	a4	b3
	a1	b4
	a2	b4
	a3	b4

÷

S	A
	a1
	a2
	a3

=

R ÷ S	B
	b1
	b4

Typische Anwendung bei Vorkommen von "alle":

- „Welche Lieferanten liefern alle Teile?“
- „Welche Mitarbeiter arbeiten an allen Projekten mit?“
- „Welche Kursleiter können alle Kurse halten?“

Zusammenfassung

- Basisoperatoren
  - Selektion $\sigma_F\ R$
  - Projektion $\pi_L\ R$
  - Umbenennung $\rho_{[neu \leftarrow alt]}\ R$ $\pi_{\{\text{neu:alt}\}}\ R$
  - Vereinigung $R \cup S$
  - Differenz $R - S$ $R \setminus S$
  - kartesische Produkt $R \times S$

- abgeleitete Operatoren
  - Schnittmenge $R \cap S$
  - Division $R \div S$
  - Joins $R \bowtie_F S$ $R ⟕_F S$ $R ⟖_F S$ $R ⟗_F S$
    $R \bowtie S$ $R ⟕ S$ $R ⟖ S$ $R ⟗ S$

Äquivalenzumformungen

- Kommutativität von <u>unären</u> Operationen: $\mathbf{U_1}\ \mathbf{U_2}\ R \equiv \mathbf{U_2}\ \mathbf{U_1}\ R$

  Beispiel: $\sigma_{F1}\sigma_{F2}\ R \equiv \sigma_{F2}\sigma_{F1}\ R$

- Kommutativität von <u>binären</u> Operationen: $R\ \mathbf{B}\ S \equiv S\ \mathbf{B}\ R$

  Beispiele: $R \cup S \equiv S \cup R$, $R \times S \equiv S \times R$, $R \bowtie S \equiv S \bowtie R$

- Assoziativität von binären Operationen: $R\ \mathbf{B}\ (S\ \mathbf{B}\ T) \equiv (R\ \mathbf{B}\ S)\ \mathbf{B}\ T$

  Beispiele: $R \cup (S \cup T) \equiv (R \cup S) \cup T$, $R \cap (S \cap T) \equiv (R \cap S) \cap T$,
  $R \times (S \times T) \equiv (R \times S) \times T$, $R \bowtie (S \bowtie T)\ (R \bowtie S) \bowtie T)$

- Distributivität von unären in Bezug auf binäre Op.:
  $\mathbf{U}\ (R\ \mathbf{B}\ S) \equiv (\mathbf{U}\ R)\ \mathbf{B}\ (\mathbf{U}\ S)$

  Beispiele: $\sigma_F\ (R \cup S) \equiv (\sigma_F\ R) \cup (\sigma_F\ S)$,
  $\sigma_F\ (R \bowtie S) \equiv (\sigma_F\ R) \bowtie (\sigma_F\ S)$

Abschließende Bemerkungen

- Relationen-Algebra nur bei den ersten relationalen DBMS-Prototypen als Anfragesprache verwendet
- formale Grundlage für DBMS-interne Anfrage-Optimierung
- Außerdem: theoretisches Maß für Ausdrucks-mächtigkeit relationaler DB-Sprachen (➔ relationale Vollständigkeit)

Ziele

- Operatoren der Relationenalgebra anwenden können um Anfragen zu formulieren

- Joins verstehen