DAB1 – Datenbanken 1

Dr. Daniel Aebi (aebd@zhaw.ch)

Lektion 4: Relationale Algebra, Relationale Bags

Wo stehen wir?

Logische Grundlagen

Einführung

Relationale Algebra
Relationale Bags

"You are here"

Entity-Relationship
Design

SQL

Rückblick

- Operationen:
  - Vereinigung $\cup$
  - Durchschnitt $\cap$
  - Differenz \ (oder -)
  - Kreuzprodukt (kartesisches Produkt) x
  - Verbundvarianten
    - Natural join $\bowtie$
    - Theta join $\bowtie_\theta$
    - Outer joins (left, right, full) ⟕, ⟖, ⟗
    - …
- Umbenennung: ρ_S (R) (Umbenennung von R in S)
  $\rho_{S(D,E)}$ (R(A,B)) (Umbenennung R.A $\rightarrow$ S.D, R.B $\rightarrow$ S.E)

Lernziele Lektion 4

Zürcher Hochschule für Angewandte Wissenschaften
School of Engineering
InIT Institut für angewandte Informationstechnologie

- Folgende (binäre) Operation der relationalen Algebra verstehen:
  - Division ÷
- Aggregatfunktionen verstehen
- Grundlagen der 'bag'-Theorie verstehen

Bemerkungen zur Notation

- Bisher eingeführt:
  - t: Tupelvariable <t(A),t(B),t(C)> oder einfach t(R)
  - r, ext(R) $\subseteq$ dom(A) x dom(B) x dom(C) (Relation)
  - Seien R(A,B,C,D) und S(B,D,E) dann werden die "join-Formate" wie folgt bezeichnet: **R·S**(A,B,C,D,E) und **S·R**(B,D,E,A,C)

- Bemerkungen:
  - Die Mengendifferenz wird oft auch mit dem Symbol - (minus) bezeichnet (statt mit backslash), also z.B. S-R statt S\R
  - Die Reihenfolge der Attribute spielt eine Rolle, ist aber nicht 'wichtig', es gilt: R(A,B,C) $\neq$ S(B,C,A) (aber S kann durch Projektion auf R erzeugt werden)
  - Rangfolge: σ, π, $\rho \Rightarrow$ x, $\bowtie \Rightarrow \cap \Rightarrow \cup$, \

Symbole der relationalen Algebra

Übersicht der Operationen:

- σ Selektion
- π Projektion
- ∪ Mengenvereinigung
- ∩ Mengendurchschnitt
- \ Mengendifferenz
- x Kreuzprodukt
- ⋈ Join (Verbund)
- ÷ Division
- ρ Umbenennung
- Linker äusserer Verbund (Symbol, oft weggelassen: ⟕)
- Rechter äusserer Verbund (Symbol, oft weggelassen: ⟖)
- Voller äusserer Verbund (Symbol, oft weggelassen: ⟗)

Division ÷

Zürcher Hochschule für Angewandte Wissenschaften
School of Engineering
InIT Institut für angewandte Informationstechnologie

- Darstellung: R÷S Voraussetzung: S $\subseteq$ R (bezogen auf Attribute)
- Resultat:
  - Ergebnis hat die Differenz der Attributmengen als Attributmenge und umfasst jene Tupel aus R die eingeschränkt auf die Differenz der Attribute (R \ S) für alle Tupel von S denselben Wert haben

- Die Division ist NICHT kommutativ: R÷S $\neq$ S÷R

- Der Divisionsoperator erlaubt die kompakte Formulierung von "für-alle"-Anfragen. Kann auch (muss) durch Projektion, Differenz und Kreuzprodukt gebildet werden:

$$R \div S = \pi_{R\text{-}S}(R) \setminus \pi_{R\text{-}S}((\pi_{R\text{-}S}(R) \times S) \setminus R)$$

Attribute von R ohne Attribute von S

Division ÷

Gegeben: 2 Relationenformate R(A, B, C) und S(A, B)

t = r ÷ s (t hat Relationenformat T(C))

Dabei gilt:

- $t \bowtie s \subseteq r$
- $(r \bowtie s) \div s = r$
- t = r ÷ s ist die maximale Relation, für welche gilt: $t \bowtie s \subseteq r$

Vergleich ganzzahlige Division:

- j = i : k ist maximales j, für welches gilt: j * k <= i

r_1	
A	B
a_1	b_1
a_1	b_2
a_1	b_3
a_2	b_2
a_2	b_3

÷

r_2
B
b_1
b_2

=

$r_1 \div r_2$
A
a_1

Division ÷

- Division in SQL ~~leider~~ nicht implementiert

- Beispiel:

  Geg.: Relationenformat Trinken(Besucher, Biersorte, Restaurant) mit zugehöriger Relation r

  $(r \div \pi_{Biersorte}(r)) \div \pi_{Restaurant}(r)$

  = Besucher, welche jede (alle) Biersorte in jedem Restaurant trinken

Division ÷

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

Beispiel Division: Was ist r÷s ?

÷

s	A
	a1
	a2
	a3

=

t	B
	b1
	b4

«Bei welchen B-Werten kommen die in s gegebenen A-Werte alle vor?»

Zwischenstand

Zürcher Hochschule für Angewandte Wissenschaften
School of Engineering
InIT Institut für angewandte Informationstechnologie

- Was können wir bisher:
  - Komplexe Anfragen formulieren durch Kombination (Schachtelung) von Ausdrücken. Das Resultat eines (Teil)ausdruckes ist immer wieder eine Relation.
  - Darstellung: Als geschachtelter Ausdruck mittels Klammerung.
  - Einfaches Beispiel:

Products

PNo	Descr	Weight	Price	Warehouse	Stock
1	'Paper'	2.000	20	'Zürich'	10000
2	'Disk Drive'	1.000	2500	'Bern'	400

Finde Beschreibung, Lagerort und Menge an Lager für alle Produkte, von denen weniger als 1000 an Lager sind

$\pi_{\text{Descr, Warehouse, Stock}}$ Products

Descr	Warehouse	Stock
'Paper'	'Zürich'	10000
'Disk Drive'	'Bern'	400

$\sigma_{\text{Stock} < 1000}$ Products

PNo	Descr	Weight	Price	Warehouse	Stock
2	'Disk Drive'	1.000	2500	'Bern'	400

$\sigma_{\text{Stock} < 1000} (\pi_{\text{Descr, Warehouse, Stock}} \text{Products}) = \pi_{\text{Descr, Warehouse, Stock}} (\sigma_{\text{Stock} < 1000} \text{Products})$

Descr	Warehouse	Stock
'Disk Drive'	'Bern'	400

Zwischenstand

- Was fehlt?
  - Zusammenfassen von Attributwerten (aggregieren)
  - Gruppieren
  - Sortierung
  - …

Aggregat-Funktionen

- Alle bisherigen Operationen behandeln nur einzelne Tupel. Diese sind ungeordnet in der Menge (Relation) enthalten und «wissen» nichts voneinander. Viele Aufgaben in einer Datenbank müssen zueinander gehörende Daten zusammentragen (dem Lateinischen entlehnt: aggregieren). Wir brauchen also sogenannte Aggregatfunktionen.

- Beispiele:
  - Wie gross ist die Summe aller Verkäufe? Aggregation über alle Tupel
  - Wieviel hat welcher Verkäufer verkauft? Aggregation pro Verkäufer (Gruppierung)

- Typische Aggregatfunktionen: Summe, Durchschnitt, Minimum, Maximum, Anzahl Tupel

Gruppierung, Aggregat-Funktionen

- Sei R ein Relationenformat und S ein Teilformat von R
(d.h. es gilt $R \cdot S = R$)

- Sei X ein Attribut (keine Attributkombination!) von R aber nicht von S
(d.h. $R \cdot X = R$, aber $S \cdot X \neq S$). Zudem soll dom(X) Summenbildung zulassen.

- Beispiel: R(A,B,X) und S(A), dom(X) = INTEGER

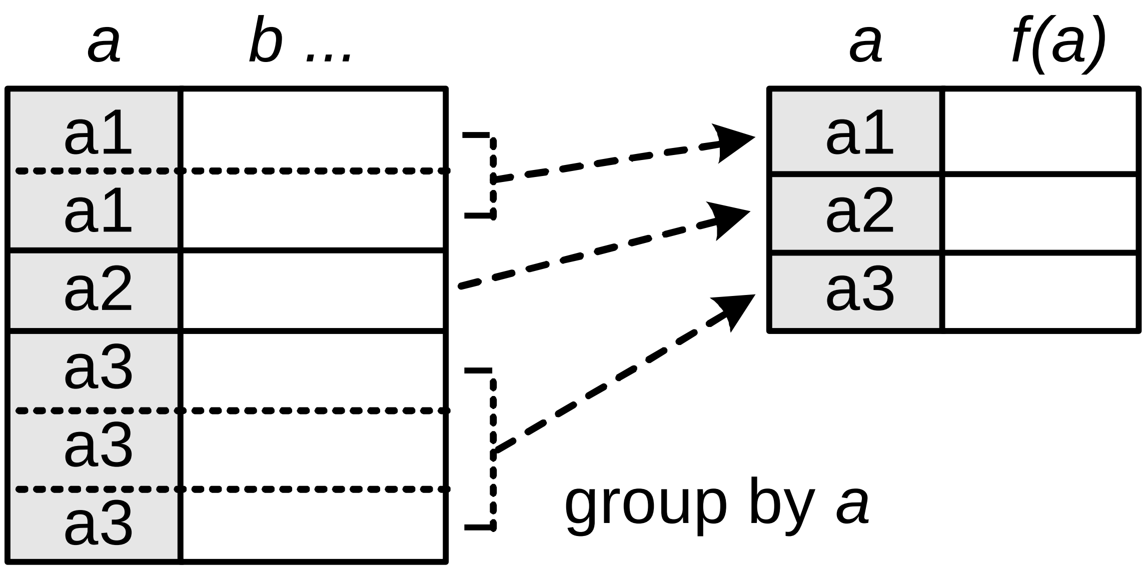
Aggregat-Funktionen

- Pro Gruppe mit gleichen A-Werten wird dann die Summe über Attribut X gebildet wie folgt:
- r = {<a1,b1,2>,<a1,b2,3>,<a2,b1,4>}, Teilformat ist S(A)
- $\Sigma_{S,X}(r) = \{<a1,5>, <a2,4>\}$
- Wir haben also einen neuen Operator: die Summe (Σ)

Gruppierung

- Jede Gruppe ist eine Multimenge von Tupeln mit denselben Werten für das Gruppierungskriterium (Attributkombination)

- Zusammenfassung nach gleichen Werten für Attribut a:



Aggregat-Funktionen

- Analog dem zur Summe gesagten, führen wir einen Aggregats-Operator $\mathcal{F}$ (ausgesprochen: «Skript-F») ein

- Es gibt 5 verschiedene Aggregat-Funktionen:
  - COUNT = Anzahl Tupel
  - MAX = Grösster Wert des betrachteten Attributs *
  - MIN = Kleinster Wert des betrachteten Attributs *
  - SUM = Summe des betrachteten Attributs *
  - AVG = Durchschnitt des betrachteten Attributs *
    (= Summe / Anzahl)

- * Das Attribut muss einen zählbaren Domain haben

Aggregat-Funktionen

- COUNT (nicht verwechseln mit SUM!) weicht von den anderen Operatoren ab, indem er nicht Attribute bzw. Attributswerte, sondern Tupel zählt

$$\text{Anzahl} = \mathcal{F}_{\text{COUNT}(r)}$$

- Bemerkung: Das Resultat einer Aggregatfunktion ist wie bei den anderen relationalen Operatoren wiederum eine Relation (ggf. eine Relation mit einem einzigen Tupel und einem einzigen Attribut)

Aggregat-Funktionen mit Gruppierung

- Allgemein: geg. Relation R(A, B, C, D), C sei ein «zählbares» Attribut
- Resultat = $\mathcal{F}$ $\text{Operator}_{S,C}(r)$, mit Gruppierung beispielsweise auf S(A, B)
- Alternative Schreibweise: Resultat = <A, B> $\mathcal{F}$ Operator_{C} (r)
- r wird nach Attributskombination A, B gruppiert
- Für jede eindeutige Kombination von A, B enthält das Resultat ein Tupel
- Genaueres hierzu folgt im Kapitel SQL!

Bags: Theorie und Praxis

Projektion π

- Eine Konsequenz der Projektion ist die Notwendigkeit einer Duplikatelimination, falls nach der Projektion Tupel „zusammenfallen"

- Aufwand der Duplikatelimination: im wesentlichen gleich zur Sortierung

- Bemerkung: Aufwand Sortierung
  - simple Methoden (Bubblesort, Insertion Sort): $O(n^2)$
    - 1000 Elemente: Grössenordnung 1'000'000 Vergleiche
    - 1'000'000 Elemente: Grössenordnung 1'000'000'000'000 Vergleiche
  - „beste" Methoden (Quicksort, Heapsort u.a.): $O(n \log_n)$
    - 1000 Elemente: Grössenordnung 3000 Vergleiche
    - 1'000'000 Elemente: Grössenordnung 13,000,000 Vergleiche

Relationale Bags

Die Versuchung war gross, sich die Duplikatelimination zu sparen…

- Schon die ersten Datenbanksysteme, welche die relationale Idee aufgriffen, wichen vom relationalen Modell in dieser Hinsicht ab (70er-Jahre)

- Es gelten dann aber nicht mehr genau dieselben Gesetze wie bei relationalen Operationen

- Neuer Zweig der Theorie: „relationale Bags" (U. Dayal, N. Goodman, R.H. Katz: „An Extended Relational Algebra with Control over Duplicate Elimination", 1982)

- SQL ist «nicht ganz relational»

Relationale Bags

- Ein Bag ist ein „Sack“, auch Multimenge genannt
- Man kann nicht unterscheiden zwischen identischen Elementen
- „Entnimmt“ man ein Exemplar eines mehrfach vorhandenes Elements aus dem „Sack“, so wählt man ein willkürliches Exemplar
- Es ist aber unterscheidbar, ob ein Bag ein Element keinmal, einmal, zweimal, dreimal, … enthält

Beispiel

- Sei R(A,B) gegeben mit dom(A) und dom(B) = STRING

Anzahl Vorkommen

- Bag r = {<a,b,1>, <a,bb,3>, <aa,b,2>, <aa,bb,1>}
  (wir listen nur jene t ∈ dom(R) mit Anzahl > 0)

- Darstellungen in Tabellenform, Beispiel:

- Reihenfolge nicht festgelegt!

A	B
a	b
a	bb
a	bb
a	bb
aa	b
aa	b
aa	bb

A	B
a	bb
aa	bb
a	bb
aa	b
aa	b
a	bb
a	b

Mathematische Definition

- Sei $R(A_1, A_2, \dots A_n)$ ein Relationenformat mit den Attributen $A_1, A_2, \dots A_n$ und ihren Domänen $dom(A_1), dom(A_2), \dots dom(A_n)$ und $dom(R) = dom(A_1) \times dom(A_2) \times \dots dom(A_n)$
- Ein Bag zum Format R sei eine Abbildung:

$$r : dom(R) \rightarrow \mathbb{N}, \quad t \mapsto r(t), \text{ wobei } \mathbb{N} = \text{natürliche Zahlen}$$

- Es wird jedem Tupel t eine natürliche Zahl $r(t) \geq 0$ zugeordnet, die Multiplizität (= «Anzahl gleiche Tupel») von t in r

Verhältnis Bag/Relation

- Ein Bag r kann genau dann mit einer Relation r identifiziert werden, wenn für die Tupel t von dom(R) nur die Multiplizitäten 0 und 1 vorkommen, d.h. wenn die Menge $\{t \in dom(R) \mid r(t) > 1\}$ leer ist.

- Wir sagen dann „der Bag ist eine Relation“, da er der Relation $\{t \in dom(R) \mid r(t) > 0\}$ entspricht

  Dies ist die Trägerrelation des Bag r.

Operationen auf Bags

Neue Operation: Duplicate Elimination $\delta(r)$

$$\delta(r) = \{<t,k> \in dom(R) \times \mathbb{N} \mid k = \min\{1,r(t)\} \}$$

r	
A	B
a	b
a	bb
a	bb
a	bb
aa	b
aa	b
aa	bb

$\delta(r)$	
A	B
a	b
a	bb
aa	b
aa	bb

Selektion σ

Zürcher Hochschule
für Angewandte Wissenschaften
zh aw School of Engineering
InIT Institut für angewandte Informationstechnologie

- Die Selektion bei Bags ist einfach zu verstehen

- Es werden alle Tupel ausgewählt, die die Bedingung erfüllen, zusammen mit ihren Multiplizitäten, Beispiel:

$$\sigma_{A=a}(r) = \{<t,k> \in \mathrm{dom}(R) \times \mathbb{N} \mid (t(A) = a \wedge k=r(t)) \vee (t(A) \neq a \wedge k=0)\}$$

r	
A	B
a	b
a	bb
a	bb
a	bb
aa	b
aa	b
aa	bb

$\sigma_{A=a}(r)$	
A	B
a	b
a	bb
a	bb
a	bb

Beispiel: Selektion σ

- Gegeben: Relationenformat R(A,B,C), dom(A)=dom(B)=dom(C)=$\mathbb{N}$
- Bag r = {<1,2,3,17>,<4,5,6,1>,<1,8,9,3>} zum Format R
- Berechnen Sie: $\sigma_{A=1}(\sigma_{B=2}(r))$
- Lösung: {<1,2,3,17>}

Projektion π

- Die Projektion ist etwas kniffliger. Es können Tupel «zusammenfallen», die vorher verschieden waren. Projektion von u auf das Teilformat R von U:

$$\pi_R(u) = \{\langle t,k\rangle \in \mathrm{dom}(R) \times \mathbb{N} \mid k = \Sigma_{x \in \mathrm{dom}(U) \wedge x(R)=t}\, u(x) \}$$

- Beachten Sie: wenn u eine Relation ist, d.h. $u(x) \in \{0,1\}$ für alle $x \in \mathrm{dom}(U)$, wird die Summe zur Anzahl der x, für welche $x(R)=t$.

r

A	B
a	b
a	bb
a	bb
a	bb
aa	b
aa	b
aa	bb

$\pi_A(r)$

A
a
a
a
a
aa
aa
aa

Beispiel: Projektion π

- Sei U(A,B,C) und das Teilformat R(B,C) gegeben
- Bag u={<0,1,1,2>,<1,1,1,3>,<0,0,0,1>,<1,0,0,2>} zum Format U
- Berechnen Sie: $\pi_R(u)$
- Lösung: {<1,1,5>,<0,0,3>}

Join Operator ⋈

Zürcher Hochschule für Angewandte Wissenschaften
School of Engineering
InIT Institut für angewandte Informationstechnologie

- Der (natural) join u = r ⋈ s der Bags r und s ist definiert als (es gilt U=R·S):

$$r \bowtie s = \{<t,k> \in dom(U) \times \mathbb{N} \mid <t(R),r(t(R))> \in r \wedge <t(S),s(t(S))> \in s \wedge k = r(t(R)) * s(t(S))\}$$

- Hier werden die Multiplizitäten von entsprechenden Tupeln der Bags r und s miteinander multipliziert, was im Grenzfall mit r, s als Relationen die bereits bekannte Definition liefert.

r

A	B
a	b
a	b
aa	bb
aa	bb

s

B	C
b	c
bb	c
bb	c
bb	cc

r⋈s

A	B	C
a	b	c
a	b	c
aa	bb	c
aa	bb	c
aa	bb	c
aa	bb	c
aa	bb	cc
aa	bb	cc

Beispiel: Join Operator ⋈

- Bags r1 mit Format R1(A,B) und r2 mit Format R2(B,C)

- r1 = {<0,0,2>, <1,0,2>, <1,1,1>}
- r2 = {<0,0,2>,<0,1,2>, <1,0,1>}

- Berechnen Sie: r1 ⋈ r2

- Lösung: {<0,0,0,4>, <0,0,1,4>, <1,0,0,4>, <1,0,1,4>, <1,1,0,1>}

Vereinigung, Durchschnitt, Differenz

- Bei den bisherigen Verallgemeinerungen von Operationen von Bags handelte es sich um solche, die sich aufdrängen
- Bei den Mengenoperationen gibt es aber Konflikte

- Zum Beispiel galt bisher:
  - Gesetz 1: $\sigma_{A=a \vee B=b}(u) = \sigma_{A=a}(u) \cup \sigma_{B=b}(u)$
  - Gesetz 2: $\pi_R(u_1 \cup u_2) = \pi_R(u_1) \cup \pi_R(u_2)$

- Was müssen wir tun, um diese Eigenschaften zu erhalten?

Vereinigung

Zürcher Hochschule
für Angewandte Wissenschaften
School of Engineering
InIT Institut für angewandte Informationstechnologie

- Es sei allgemein: $r \cup s = \{<t,k> \in dom(R) \times \mathbb{N} \mid k=\varphi(r(t),s(t))\}$

- Wir wählen nun φ ("phi") geeignet, um die Multiplizitäten zu bestimmen

- Behauptung: mit φ = max gilt Gesetz 1, aber nicht Gesetz 2

- Behauptung: mit φ = + gilt Gesetz 2, nicht aber Gesetz 1

Beispiel zu Gesetz 1

- Gilt: $\sigma_{A=a \vee B=b}(u) = \sigma_{A=a}(u) \cup \sigma_{B=b}(u)$ (Gesetz 1)?
- Beispiel: R(A,B,C) mit r = {<0,0,1,7>,<0,1,0,5>,<0,1,1,3>,<1,0,0,4>}

- $\sigma_{A=0 \vee B=1}(r) = \{<0,0,1,7>,<0,1,0,5>,<0,1,1,3>\}$
- $\sigma_{A=0}(r) = \{<0,0,1,7>,<0,1,0,5>,<0,1,1,3>\}$
- $\sigma_{B=1}(r) = \{<0,1,0,5>,<0,1,1,3>\}$

- Bei Definition mit φ = max:
  - $\sigma_{A=0}(r) \cup \sigma_{B=1}(r) = \{<0,0,1,7>,<0,1,0,5>,<0,1,1,3>\}$
- Bei Definition mit φ = +:
  - $\sigma_{A=0}(r) \cup \sigma_{B=1}(r) = \{<0,0,1,7>,<0,1,0,10>,<0,1,1,6>\}$

- Gesetz 1 gilt (nur) bei Definition mit φ = max!

Beispiel zu Gesetz 2

- Gilt: $\pi_R(u_1 \cup u_2) = \pi_R(u_1) \cup \pi_R(u_2)$ (Gesetz 2)?
- Beispiel: R(A,B,C) und S(A,B,C) mit

  r = {<0,0,1,7>,<0,1,0,5>,<0,1,1,3>,<1,0,0,4>}
  s = {<0,0,1,3>,<0,1,1,4>,<1,0,1,2>,<1,0,0,3>}

- Bei Definition mit φ = max:
  - $r \cup s$ = {<0,0,1,7>,<0,1,0,5>,<0,1,1,4>,<1,0,0,4>,<1,0,1,2>}
  - $\pi_{A,B}(r \cup s)$ = {<0,0,7>,<0,1,9>,<1,0,6>}
- Bei Definition mit φ = +:
  - $r \cup s$ = {<0,0,1,10>,<0,1,0,5>,<0,1,1,7>,<1,0,0,7>,<1,0,1,2>}
  - $\pi_{A,B}(r \cup s)$ = {<0,0,10>,<0,1,12>,<1,0,9>}

Beispiel zu Gesetz 2

- R(A,B,C) und S(A,B,C)
- $r = \{\langle 0,0,1,7\rangle,\langle 0,1,0,5\rangle,\langle 0,1,1,3\rangle,\langle 1,0,0,4\rangle\}$
- $s = \{\langle 0,0,1,3\rangle,\langle 0,1,1,4\rangle,\langle 1,0,1,2\rangle,\langle 1,0,0,3\rangle\}$

- $\pi_{A,B}(r) = \{\langle 0,0,7\rangle,\langle 0,1,8\rangle,\langle 1,0,4\rangle\}$
- $\pi_{A,B}(s) = \{\langle 0,0,3\rangle,\langle 0,1,4\rangle,\langle 1,0,5\rangle\}$

- Bei Definition mit φ = max:
  - $\pi_{A,B}(r) \cup \pi_{A,B}(s) = \{\langle 0,0,7\rangle,\langle 0,1,8\rangle,\langle 1,0,5\rangle\}$

- Bei Definition mit φ = +:
  - $\pi_{A,B}(r) \cup \pi_{A,B}(s) =\{\langle 0,0,10\rangle,\langle 0,1,12\rangle,\langle 1,0,9\rangle\}$

- Gesetz 2 gilt (nur) bei Definition mit $\varphi = +$!

Vereinigung

- Wir wählen:

- $r \cup s = \{\langle t,k \rangle \in dom(R) \times \mathbb{N} \mid k = max(r(t),s(t))\}$ "bag union"

- $r \sqcup s = \{\langle t,k \rangle \in dom(R) \times \mathbb{N} \mid k = r(t) + s(t)\}$ "bag concatenation"

- SQL bietet folgendes:

  $\delta(r \cup s)$ als UNION DISTINCT (bzw. als UNION, d.h. default ist DISTINCT)
  $r \sqcup s$ als UNION ALL

- Es gilt: $r \cup s = (r \setminus s) \sqcup s$

Durchschnitt, Differenz

- Zwei ergänzende Definitionen:

- $r \cap s = \{<t,k> \in dom(R) \times \mathbb{N} \mid k = \min\{r(t),s(t)\}\}$
- $r \setminus s \; = \{<t,k> \in dom(R) \times \mathbb{N} \mid k = \max\{0,r(t) - s(t)\}\}$

- Die Definitionen stimmen im Grenzfall der Relationen mit den entsprechenden Mengenoperationen überein

Zeit für ein Bierchen!

- Gegeben sind die Relationenformate:
  - Gast(Besucher,Restaurant)
  - Sortiment(Restaurant,Biersorte)
  - Vorzug(Besucher,Biersorte)
  - sowie je ein Bag
    - g zum Format Gast
    - s zum Format Sortiment
    - v zum Format Vorzug
- Man weiss nicht, ob die Bags Relationen sind oder nicht.
- Gesucht: Alle Besucher des Restaurant Ochsen, die keine Biere bevorzugen

«The attack of the Clones»

Was ist nun korrekt?

$$\pi_{Besucher}(\sigma_{Restaurant='Ochsen'}(g)) \setminus \pi_{Besucher}(v)$$

ODER

$$\delta(\pi_{Besucher}(\sigma_{Restaurant='Ochsen'}(g))) \setminus \pi_{Besucher}(v)$$

Wie oft gibt es jede Person?

Wie oft verzeichnen wir ihren Besuch in einem Restaurant?

Und weiter..

Zürcher Hochschule für Angewandte Wissenschaften
School of Engineering
InIT Institut für angewandte Informationstechnologie

Das nächste Mal: Schlüssel, Datenbankentwurf