



Grundlagen des relationalen Datenmodells

Medieninformatik Bachelor
Modul 9:
Datenbanksysteme



- **Aufgabe 1 Anfragen & Modellierung“**

Denken Sie mal darüber nach, welche Anfragen Sie an die AOL Daten stellen möchten. Bitte Sie bitte ein logisches und physisches Schema zur Beantwortung dieser Anfragen.

- **Aufgabe 2 „SQL und Abfrageausführung“**

Bitte formulieren Sie für Ihre Analyseideen aus 1.) die SQL Anfragen. Sie verstehen auch Möglichkeiten der Abfrageausführung bzw. Optimierung.

- **Aufgabe 3 „Datenintegration“**

Zur Ausführung der Ausführung fehlen Ihnen noch externe Daten, z.B. aus dem Internet Archive, DMOZ oder Freebase.org. Bitte ergänzen Sie Ihr Schema und die Datenbasis.

- **Aufgabe 4 „Analyse, Erkenntnisgewinn und Wert“**

Stellen Sie in 5 Minuten die wichtigsten Erkenntnisse aus den Daten vor. Bewerten Sie den Erkenntnisgewinn, z.B. gegenüber Ihren Kommilitonen oder der Literatur! Welche Erkenntnisse hätten einen kommerziellen Wert?



- Was sind Datenbanken?
 - Motivation, Historie, Datenunabhängigkeit, Einsatzgebiete
- Datenbankentwurf im ER-Modell & Relationaler Datenbankentwurf
 - Entities, Relationships, Kardinalitäten, Diagramme
 - Relationales Modell, ER -> Relational, Normalformen, Transformationseigenschaften
- Relationale Algebra & SQL
 - Kriterien für Anfragesprachen, Operatoren, Transformationen
 - SQL DDL, SQL DML, SELECT ... FROM ... WHERE ...
- Datenintegration & Transaktionsverwaltung
 - JDBC, Cursor, ETL
 - Mehrbenutzerbetrieb, Serialisierbarkeit, Sperrprotokolle, Fehlerbehandlung, Isolationsebenen in SQL
- Ausblick
 - Map/Reduce, HDFS, Hive ...
 - Wert von Daten



- Operationen der Relationenalgebra
 - Einführung
 - Basisoperatoren
 - Komplexe Ausdrücke
 - Abgeleitete Operatoren
 - Erweiterte Operatoren



- Operationen der Relationenalgebra
 - **Einführung**
 - Basisoperatoren
 - Komplexe Ausdrücke
 - Abgeleitete Operatoren
 - Erweiterte Operatoren



- Bisher
 - Relationenschemata mit Basisrelationen, die in der Datenbank gespeichert sind
- Jetzt
 - „Abgeleitete“ Relationenschemata mit virtuellen Relationen, die aus den Basisrelationen berechnet werden
 - Definiert durch Anfragen
 - Basisrelationen bleiben unverändert





- **Ad-Hoc-Formulierung**
 - Benutzer soll eine Anfrage formulieren können, ohne ein vollständiges Programm schreiben zu müssen
- **Deskriptivität**
 - Benutzer soll formulieren „Was will ich haben?“ und nicht „Wie komme ich an das, was ich haben will?“
 - **Deklarativ**
- **Mengenorientiertheit**
 - Operationen auf Mengen von Daten
 - Nicht navigierend nur auf einzelnen Elementen („tuple-at-a-time“)
- **Abgeschlossenheit**
 - Ergebnis ist wieder eine Relation und kann wieder als Eingabe für die nächste Anfrage verwendet werden.





- **Adäquatheit**
 - Alle Konstrukte des zugrundeliegenden Datenmodells werden unterstützt
- **Orthogonalität**
 - Sprachkonstrukte sind in ähnlichen Situationen auch ähnlich anwendbar
- **Optimierbarkeit**
 - Sprache besteht aus wenigen Operationen, für die es Optimierungsregeln gibt
- **Effizienz**
 - Jede Operation ist effizient ausführbar
 - Im relationalen Modell hat jede Operation eine Komplexität $\leq O(n^2)$, n Anzahl der Tupel einer Relation.





- **Sicherheit**
 - Keine Anfrage, die syntaktisch korrekt ist, darf in eine Endlosschleife geraten oder ein unendliches Ergebnis liefern.
- **Eingeschränktheit**
 - Anfragesprache darf keine komplette Programmiersprache sein
 - Folgt aus Sicherheit, Optimierbarkeit, Effizienz
- **Vollständigkeit**
 - Sprache muss mindestens die Anfragen einer Standardsprache (z.B. relationale Algebra) ausdrücken können.





■ **Mathematik**

- Algebra: Definiert durch Wertebereich und auf diesem definierte Operatoren
- Operand: Variablen oder Werte aus denen neue Werte konstruiert werden können
- Operator: Symbole, die Prozeduren repräsentieren, die aus gegebenen Werten neue Werte produzieren

■ **Für Datenbank Anfragen**

- Inhalte der Datenbank (Relationen) sind Operanden
- Operatoren definieren Funktionen zum Berechnen von Anfrageergebnissen
 - Grundlegenden Dinge, die wir mit Relationen tun wollen.
- Relationale Algebra (Relationenalgebra, RA)
 - Anfragesprache für das relationale Modell



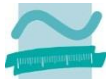


- Relation: Menge von Tupeln
- Datenbanktabelle: Multimenge von Tupeln
- Operatoren der relationalen Algebra: Operatoren auf Mengen
- Operatoren auf DBMS: SQL Anfragen
 - Rel. DBMS speichern Multimengen
- Motivation: Effizienzsteigerung
 - Beispiel:
 - Vereinigung als Multimenge
 - Vereinigung als Menge



- Operationen der Relationenalgebra
 - Einführung
 - **Basisoperatoren**
 - Komplexe Ausdrücke
 - Abgeleitete Operatoren
 - Erweiterte Operatoren





- Die mittels SQL formulierten Anfragen werden vom DBMS auf diese relationalen Operatoren abgebildet.
- Kann jeder Term der Relationenalgebra in der konkreten Anfragesprache umgesetzt werden, so heißt sie **relational vollständig**. SQL ist relat. vollständig.

Enter SQL Statement:

```
SELECT FIRMA, NAME  
FROM kdst, vert  
WHERE VERTRETER = VERTNR  
AND VERTNR IN  
(SELECT VERTRETER  
FROM kdst  
WHERE ORT = 'München');
```

Results | Script Output | Explain | Autotrace | DBMS Output | OWA Output

OPERATION	OBJECT_NAME	OPTIONS	COST
SELECT STATEMENT			8
HASH JOIN		SEMI	8
Access Predicates		VERTNR=VERTRETER	
NESTED LOOPS			4
TABLE ACCESS	KDST	FULL	3
TABLE ACCESS	VERT	BY INDEX ROWID	1
INDEX	SYS_C0038403	UNIQUE SCAN	0
Access Predicates		VERTRETER=VERTNR	
TABLE ACCESS	KDST	FULL	3
Filter Predicates		ORT='München'	



Basisoperationen (5+1 $\sigma, \pi, \cup, \times, -, \rho$)

- Projektion (Attributauswahl π)
- Selektion (Tupelauswahl σ)
- Kartesisches Produkt (Kreuzprodukt \times)
- Vereinigung (Union \cup)
- Differenz (Except / Minus $-$)
- Umbenennung (ρ)

Aus Basisoperatoren können weitere Mengenoperationen abgeleitet werden

- Vereinigung (\cup) bzw. Schnitt (\cap)
- Natürlicher Join (\bowtie)
- Theta-Join (\bowtie)
- Division ($/$)
- Symmetrische Differenz

Es gibt auch Erweiterungen, wie Duplikateliminierung, Aggregation, Gruppierung, Sortierung, erweiterte Projektion, Outer Join, Outer Union, Semijoin





Mengenoperationen - Allgemeines:

- Um die Mengenoperationen “Vereinigung”, “Differenz”, “Durchschnitt” und “Symmetrische Differenz” auf zwei Relationen R und S anwenden zu können, müssen beide Relationen zueinander kompatibel sein.
- Die dafür erforderliche **Vereinigungsverträglichkeit** ist gegeben, wenn:
 - R und S den gleichen Grad (d.h. eine identische Attributmenge) aufweisen
 - Die Wertebereiche der einzelnen korrespondierenden Attribute von R und S identisch sind
- Ist die Vereinigungsverträglichkeit nicht gegeben, so können die Mengenoperationen nicht durchgeführt werden. Mit Durchführung einer Projektion auf R und/oder S können jedoch ggf. Mengenoperationen auf Teilattributmengen beider Relationen durchgeführt werden.





Projektion

- Selektiert einzelne Attribute (auch Projektionsliste β genannt) aus der zur Verfügung stehenden Attributmenge $\{A_1, \dots, A_n\}$ einer Relation R
- Abstrakter Operator: π (kleiner gr. Buchstabe “Pi”)
- Formale **Definition** der Operation:

Sei R eine Relation über $\{A_1, \dots, A_n\}$ und $\beta \subseteq \{A_1, \dots, A_n\}$.

$$\pi_{\beta}(R) := \{t_{\beta} \mid t \in R\}$$

- Lineare Schreibweise: $R[\beta]$

t_{β} heißt, die Tupel der Ergebnisrelation enthalten nur die Attribute der Projektionsliste β

Beispiel:

R:

A	B	C
1	2	3
4	5	6
7	8	9

R[A,B]:

A	B
1	2
4	5
7	8

- **Relationenalgebra:**

$$\pi_{A,B}(R)$$

- **SQL:**

```
select A,B
from R;
```




Selektion

- Selektiert einzelne Tupel einer Relation R anhand eines Vergleichsausdrucks
- Abstrakter Operator: σ (kleiner gr. Buchstabe “Sigma”)
- Formale **Definition** der Operation:

Sei R eine Relation. Alle Tupel t der Ergebnisrelation definieren sich wie folgt:

$$\sigma_{\langle \text{Ausdruck} \rangle}(R) := \{t \mid t \in R \wedge t \text{ erfüllt } \langle \text{Ausdruck} \rangle\}$$

- Lineare Schreibweise: $R[\langle \text{Ausdruck} \rangle]$

$\langle \text{Ausdruck} \rangle$ ist ein Platzhalter für eine Formel:

- Konstantenselektion, z.B.: $C > 3$
- Attributselektion, z.B.: $A = C$
- oder sonst. logische Prädikate, die nach Auswertung zu „TRUE“ evaluieren (z.B. mit Aggregatfunktionen).

Beispiel:

R:

A	B	C
1	2	3
4	5	6
7	8	9
1	9	3

$R[C > 3]$:

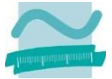
A	B	C
4	5	6
7	8	9

- **Relationenalgebra:**

$$\sigma_{C > 3}(R)$$

- **SQL:**

```
select *  
from R  
where C > 3;
```



Kartesisches Produkt (Kreuzprodukt):

- Ergebnisrelation enthält die Menge aller Kombinationen der Tupel aus R und S
- Die **Anzahl der Tupel** (Zeilen) der Ergebnisrelation ergibt sich aus der Multiplikation der Tupelanzahl von R mit der Tupelanzahl von S
- **Keine Mengenoperation**, deshalb keine Vereinigungsverträglichkeit notwendig
- Formale **Definition** der Operation:

Sei R eine Relation über $\{A_1, \dots, A_n\}$ und S eine Relation über $\{B_1, \dots, B_m\}$.

$$R \times S := \{(A_1, \dots, A_n, B_1, \dots, B_m) \mid (A_1, \dots, A_n) \in R \wedge (B_1, \dots, B_m) \in S\}$$

Beispiel:

R:

A	B	C
1	2	3
4	5	6
7	8	9

S:

D	E
1	2
3	4

$R \times S$:

A	B	C	D	E
1	2	3	1	2
4	5	6	1	2
7	8	9	1	2
1	2	3	3	4
4	5	6	3	4
7	8	9	3	4

- **Relationenalgebra:**

$R \times S$

- **SQL:**

select R., S.*
from R, S;*



Mengenoperationen - Differenz (Except / Minus):

- Entfernt alle Tupel der ersten Relation R, die auch in der zweiten Relation S vorkommen (“R mit Ausnahme von S” oder “R ohne S”)
- Je nach SQL-Dialekt des DBMS kann der Befehl **except** oder **minus** lauten
- Formale **Definition** der Operation:

Seien *R* und *S* Relationen deren Vereinigungsverträglichkeit gegeben ist.

$$R \setminus S := \{t \mid t \in R \wedge t \notin S\}$$

Lies: Für alle Tupel *t* der Ergebnisrelation $R \setminus S$ gilt:
t ist Element aus *R* und *t* ist *nicht* Element aus *S*

Beispiel:

R:

A	B	C
1	2	3
4	5	6

S:

A	B	C
7	8	9
4	5	6

$R \setminus S$:

A	B	C
1	2	3

■ Relationenalgebra:

$R \setminus S$

■ SQL:

```
select * from R
except
select * from S;
```



Mengenoperationen - Vereinigung (Union):

- Vereint alle Tupel der Relation R mit allen Tupeln der Relation S
- **Duplikat-Filter:** Tupel, die sowohl in R als auch S vorkommen, treten in der Ergebnisrelation nur einmal auf (Erfüllung der Tupel-Eindeutigkeit)
- Formale **Definition** der Operation:

Seien R und S Relationen deren Vereinigungsverträglichkeit gegeben ist.

$$R \cup S := \{t \mid t \in R \vee t \in S\}$$

Lies: Für alle Tupel t der Ergebnisrelation $R \cup S$ gilt:
 t ist Element aus R oder t ist Element aus S

Beispiel:

R:

A	B	C
1	2	3
4	5	6

S:

A	B	C
7	8	9
4	5	6

$R \cup S$:

A	B	C
1	2	3
4	5	6
7	8	9

■ Relationenalgebra:

$$R \cup S$$

■ SQL:

```
select * from R
union      // Achtung: union all
select * from S;
```



Umbenennung:

- Dient der (temporären) Umbenennung von Attributen und Relationen.
- Zweck: z.B. Mengenoperationen bei unterschiedl. Attributnamen zu ermöglichen
- Formale **Definition** der Operation:

Sei R eine Relation. Alle Tupel t' der Ergebnisrelation definieren sich wie folgt:

$$\rho_{\langle \text{neu} \rangle \leftarrow \langle \text{alt} \rangle}(R) := \{t' \mid t'(R - \langle \text{neu} \rangle) = t(R - \langle \text{alt} \rangle) \wedge t'(\langle \text{neu} \rangle) = t(\langle \text{alt} \rangle)\}$$

- Lineare Schreibweise: $R[\langle \text{neu} \rangle \leftarrow \langle \text{alt} \rangle]$

Sinngemäß: Für alle Tupel t' der Ergebnisrelation mit einem umbenannten Element gilt: Die Relation R abzüglich des alten und des neuen (umbenannten) Attributs ist in jedem Tupel hinsichtlich des Attributnamens und Attributwertes identisch. Außerdem sind die Werte des neuen und des alten Attributs in jedem Tupel identisch.

Beispiel:

R:

A	B	C
1	2	3
4	5	6
7	8	9

$R[U \leftarrow C]$:

A	B	U
1	2	3
4	5	6
7	8	9

- **Relationenalgebra:**

$$\rho_{U \leftarrow C}(\pi_{A,B,C}(R))$$

- **SQL:**

```
select A, B, C as U
from R;
```



- Bitte erstellen Sie eine Multiple Choice Aufgabe zum Thema Relationale Algebra
 - Formulieren Sie eine Frage und 3 Antworten (A, B, C)
 - Davon sollte mindestens eine Antwort richtig und mindestens eine Antwort falsch sein
- Geben Sie die Aufgabe an Ihren rechten Nachbarn. Diskutieren Sie gemeinsam und markieren Sie die richtigen Lösungen
- Geben Sie am Ende der Vorlesung Ihre Aufgabe bei mir ab

5 min



The bar exam after the bar exam.



EXKURS: KOMPLEXE AUSDRÜCKE





- Idee: Kombination (Schachtelung) von Ausdrücken zur Formulierung komplexer Anfragen.
 - Abgeschlossenheit der relationalen Algebra
 - Output eines Ausdrucks ist immer eine Relation.
 - Darstellung
 - Als geschachtelter Ausdruck mittels Klammerung
 - Als Baum





- Gesucht: Titel und Jahr von Filmen, die von Fox produziert wurden und mindestens 100 Minuten lang sind.
 - Suche alle Filme von Fox
 - Suche alle Filme mit mindestens 100 Minuten
 - Bilde die Schnittmenge der beiden Zwischenergebnisse
 - Projiziere die Relation auf die Attribute Titel und Jahr.
- $\pi_{\text{Titel}, \text{Jahr}}(\sigma_{\text{Länge} \geq 100}(\text{Filme}) \cap \sigma_{\text{StudioName} = \text{'Fox'}}(\text{Filme}))$

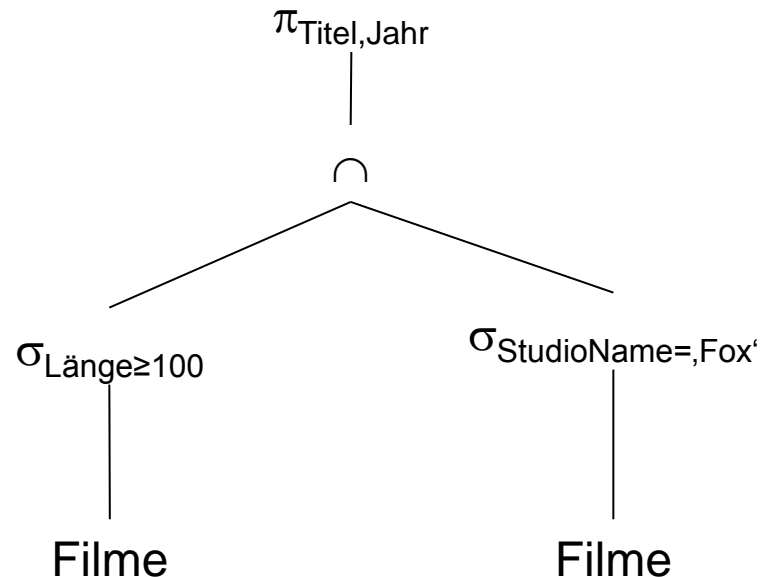
Filme

Titel	Jahr	Länge	Typ	StudioName
Total Recall	1990	113	Farbe	Fox
Basic Instinct	1992	127	Farbe	Disney
Dead Man	1995	90	s/w	Paramount





- $\pi_{\text{Titel}, \text{Jahr}}(\sigma_{\text{Länge} \geq 100}(\text{Filme}) \cap \sigma_{\text{StudioName} = \text{'Fox'}}(\text{Filme}))$



- Alternative:
 $\pi_{\text{Titel}, \text{Jahr}}(\sigma_{\text{Länge} \geq 100 \text{ AND StudioName} = \text{'Fox'}}(\text{Filme}))$



Basisoperationen (5+1 $\sigma, \pi, \cup, \times, -, \rho$)

- Projektion (Attributauswahl π)
- Selektion (TupelAuswahl σ)
- Kartesisches Produkt (Kreuzprodukt \times)
- Vereinigung (Union \cup)
- Differenz (Except / Minus $-$)
- Umbenennung (ρ)

Aus Basisoperatoren können weitere Mengenoperationen abgeleitet werden

- Vereinigung (\cup) bzw. Schnitt (\cap)
- Natürlicher Join (\bowtie)
- Theta-Join (\bowtie)
- Division ($/$)
- Symmetrische Differenz

Es gibt auch Erweiterungen, wie Duplikateliminierung, Aggregation, Gruppierung, Sortierung, erweiterte Projektion, Outer Join, Outer Union, Semijoin



Mengenoperationen - Schnittmenge (Intersect):

- Die Ergebnisrelation enthält alle Tupel, die sowohl in R als auch in S vorkommen
- Duplikat-Filter: Tupel, die sowohl in R als auch S vorkommen, treten in der Ergebnisrelation nur einmal auf (Erfüllung der Tupel-Eindeutigkeit)
- Formale **Definition** der Operation:

Seien R und S Relationen deren Vereinigungsverträglichkeit gegeben ist.

$$R \cap S := \{t \mid t \in R \wedge t \in S\}$$

Lies: Für alle Tupel t der Ergebnisrelation $R \cap S$ gilt: t ist Element aus R und t ist (gleichzeitig) Element aus S

Beispiel:

R:

A	B	C
1	2	3
4	5	6

S:

A	B	C
7	8	9
4	5	6

$R \cap S$:

A	B	C
4	5	6

Relationenalgebra:

$$R \cap S \text{ ist äquivalent zu } R \setminus (R \setminus S)$$

SQL:

*select * from R*

intersect

*select * from S;*



- Nicht als primitiver Operator unterstützt.
- Finde alle Segler, die alle Segelboote reserviert haben.
- Relation $R(x,y)$, Relation $S(y)$
 - $R/S = \{ t \mid \exists x, y \in R \forall y \in S \}$
 - R/S enthält alle x -Tupel (Segler), so dass es für jedes y -Tupel (Boot) in S ein xy -Tupel in R gibt.
 - Andersherum: Falls die Menge der y -Werte (Boote), die mit einem x -Wert (Segler) assoziiert sind, alle y -Werte in S enthält, so ist der x -Wert in R/S .

Folie und Beispiel aus: Ramakrishnan, Gehrke „Database Management Systems“





sno	pno
s1	p1
s1	p2
s1	p3
s1	p4
s2	p1
s2	p2
s3	p2
s4	p2
s4	p4

A

pno
p2

B1

sno
s1
s2
s3
s4

A/B1

pno
p2
p4

B2

sno
s1
s4

A/B2

pno
p1
p2
p4

B3

sno
s1

A/B3



- Division ist kein essentieller Operator, nur nützliche Abkürzung.
 - Ebenso wie Joins, aber Joins sind so üblich, dass Systeme sie speziell unterstützen.
- Idee: Um R/S zu berechnen, berechne alle x -Werte, die nicht durch einen y -Wert in S „disqualifiziert“ werden.
 - x -Wert ist disqualifiziert, falls man durch Anfügen eines y -Wertes ein xy -Tupel erhält, das nicht in R ist.
- Disqualifizierte x -Werte: $\pi_x ((\pi_x(R) \times S) - R)$
- R/S : $\pi_x (R) -$ alle disqualifizierten Tupel
- Also formal: $R/S = \pi_x(R) - \pi_x((\pi_x(R) \times S) - R)$



Verbund (Join):

- Ergebnisrelation enthält die Menge der Kombinationen der Tupel aus R und S, die einem bestimmten Vergleichsausdruck (Verbundprädikat) zwischen einem Attribut aus R und einem Attribut aus S genügen
- Damit stellt ein Verbund die **sequentielle Ausführung** der Operationen “**Kartesisches Produkt**” und “**Selektion**” dar.
- Formale **Definition** der Operation:

Sei R eine Relation über $r = \{A_1, \dots, A_n\}$ und S eine Relation über $s = \{B_1, \dots, B_m\}$.

$$R \bowtie_{\langle \text{Ausdruck} \rangle} S := \{r \cup s \mid r \in R \wedge s \in S \wedge \langle \text{Ausdruck} \rangle\} := \sigma_{\langle \text{Ausdruck} \rangle}(R \times S)$$

Beispiel:

R:

A	B	C
1	2	3
4	5	6
7	8	9

S:

D	E
1	2
3	4

$(R \times S)[R.A \neq S.D]$:

A	B	C	D	E
4	5	6	1	2
7	8	9	1	2
1	2	3	3	4
4	5	6	3	4
7	8	9	3	4

■ Relationenalgebra:

$$R \bowtie_{R.A \neq S.D} S \quad \text{bzw.} \quad \sigma_{R.A \neq S.D}(R \times S)$$

■ SQL:

```
select R.*, S.*
from R, S
where R.A <> S.D;
```




Mengenoperationen – Symmetrische Differenz

- Die Ergebnisrelation enthält alle Tupel, die entweder in R oder in S, jedoch nicht in beiden gleichzeitig vorkommen.
- Oder ausgedrückt mit den anderen Mengenoperationen:
 - Die “Vereinigung von R und S” minus dem “Durchschnitt von R und S” oder
 - Die “Differenz aus R und S” vereint mit der “Differenz aus S und R”
- Formale **Definition** der Operation:

Seien R und S Relationen deren Vereinigungsverträglichkeit gegeben ist.

$$R \Delta S := \{t \mid (t \in R \vee t \in S) \wedge t \notin (R \cap S)\}$$

Beispiel:

R:

A	B	C
1	2	3
4	5	6

S:

A	B	C
7	8	9
4	5	6

$R \Delta S$:

A	B	C
1	2	3
7	8	9

Relationenalgebra:

$R \Delta S$ ist äquivalent zu $(R \cup S) \setminus (S \cap R)$

SQL:

`(select * from R union select * from S)`

except

`(select * from S intersect select * from R);`



- Gesucht: Namen der Stars, die in Filmen spielten, die mindestens 100 Minuten lang sind.
 - Verjoine beide Relationen (natürlicher Join)
 - Selektiere Filme, die mindestens 100 Minuten lang sind.
 - $\pi_{\text{SchauspName}}(\sigma_{\text{Länge} \geq 100}(\text{Filme1} \bowtie \text{Filme2}))$

Filme1

Titel	Jahr	Länge	Typ	StudioName
Total Recall	1990	113	Farbe	Fox
Basic Instinct	1992	127	Farbe	Disney
Dead Man	1995	121	s/w	Paramount

Filme2

Titel	Jahr	SchauspName
Total Recall	1990	Sharon Stone
Basic Instinct	1992	Sharon Stone
Total Recall	1990	Arnold
Dead Man	1995	Johnny Depp



ZUSAMMENFASSUNG





- Minimale Relationenalgebra:
 - $\pi, \sigma, \times, \rho, \cup$ und $-$
- Unabhängig:
 - Kein Operator kann weggelassen werden ohne Vollständigkeit zu verlieren.
- Natural Join, Join, Division und Schnittmenge sind redundant
 - $R \cap S = R - (R - S)$
 - $R \bowtie_C S = \sigma_C(R \times S)$
 - $R \bowtie S = \pi_L(\sigma_{R.A_1=S.A_1 \text{ AND } \dots \text{ AND } R.A_n=S.A_n}(R \times S))$
 - $R/S = \pi_X(R) - \pi_X((\pi_X(R) \times S) - R)$





- Beispiele für algebraische Regeln zur Transformation
 - $R \bowtie S = S \bowtie R$
 - $(R \bowtie S) \bowtie T = R \bowtie (S \bowtie T)$
 - $\pi Y(\pi X(R)) = \pi Y(R)$
 - $\sigma_{A=a}(R \cup S) = \sigma_{A=a}(R) \cup \sigma_{A=a}(S)$
- Jeweils Frage: Welche Seite ist besser? -> In zwei VL





Basisoperationen (5+1 $\sigma, \pi, \cup, \times, -, \rho$)

- Projektion (Attributauswahl π)
- Selektion (TupelAuswahl σ)
- Kartesisches Produkt (Kreuzprodukt \times)
- Vereinigung (Union \cup)
- Differenz (Except / Minus $-$)
- Umbenennung (ρ)

Aus Basisoperatoren können weitere Mengenoperationen abgeleitet werden

- Vereinigung (\cup) bzw. Schnitt (\cap)
- Natürlicher Join (\bowtie)
- Theta-Join (\bowtie)
- Division ($/$)
- Symmetrische Differenz

Es gibt auch Erweiterungen, wie Duplikateliminierung, Aggregation, Gruppierung, Sortierung, erweiterte Projektion, Outer Join, Outer Union, Semijoin



- **Aufgabe 1 Anfragen & Modellierung“**

Denken Sie mal darüber nach, welche Anfragen Sie an die AOL Daten stellen möchten. Bitte Sie bitte ein logisches und physisches Schema zur Beantwortung dieser Anfragen.

- **Aufgabe 2 „SQL und Abfrageausführung“**

Bitte formulieren Sie für Ihre Analyseideen aus 1.) die SQL Anfragen. Sie verstehen auch Möglichkeiten der Abfrageausführung bzw. Optimierung.

- **Aufgabe 3 „Datenintegration“**

Zur Ausführung der Ausführung fehlen Ihnen noch externe Daten, z.B. aus dem Internet Archive, DMOZ oder Freebase.org. Bitte ergänzen Sie Ihr Schema und die Datenbasis.

- **Aufgabe 4 „Analyse, Erkenntnisgewinn und Wert“**

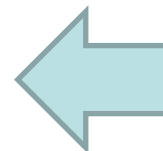
Stellen Sie in 5 Minuten die wichtigsten Erkenntnisse aus den Daten vor. Bewerten Sie den Erkenntnisgewinn, z.B. gegenüber Ihren Kommilitonen oder der Literatur! Welche Erkenntnisse hätten einen kommerziellen Wert?



- Was sind Datenbanken?
 - Motivation, Historie, Datenunabhängigkeit, Einsatzgebiete
- Datenbankentwurf im ER-Modell & Relationaler Datenbankentwurf
 - Entities, Relationships, Kardinalitäten, Diagramme
 - Relationales Modell, ER -> Relational, Normalformen, Transformationseigenschaften
- Relationale Algebra & SQL
 - Kriterien für Anfragesprachen, Operatoren, Transformationen
 - SQL DDL, SQL DML, SELECT ... FROM ... WHERE ...
- Datenintegration & Transaktionsverwaltung
 - JDBC, Cursor, ETL
 - Mehrbenutzerbetrieb, Serialisierbarkeit, Sperrprotokolle, Fehlerbehandlung, Isolationsebenen in SQL
- Ausblick
 - Map/Reduce, HDFS, Hive ...
 - Wert von Daten



- Einführung
- Basisoperatoren
 - Selektion (σ)
 - Projektion (π)
 - Vereinigung (\cup)
 - Differenz (\setminus oder $-$)
 - Cartesisches Produkt (\times)
 - Umbenennung (ρ)
- Abgeleitete Operatoren
 - Join (\bowtie)
 - Schnitt (\cap)
 - Division ($/$)
- **Erweiterte Operatoren**
 - **Duplikateliminierung**
 - **Generalisierte Projektion (Gruppierung und Aggregation)**
 - **Outer-Joins und Semi-Joins**
 - **Sortierung**



In der nächsten Veranstaltung





ÜBUNG



SQL Island Adventure - Mozilla Firefox

www.lgis.informatik.uni-kl.de/extra/game/


Nach einem Flugzeugabsturz stellst du fest, dass du der einzige Überlebende bist. Du landest auf einer Insel, erkundest sie und findest einige Dörfer:

> SELECT * FROM dorf

dorfnr	name	haeuptling
1	Affenstadt	9
2	Gurkendorf	6
3	Zwiebelhausen	13

[Weiter](#)

Du triffst während deiner Erkundung auf viele Bewohner der Insel. Welche?



[DORF](#) (dorfnr, name, haeuptling) [BEWOHNER](#) (bewohnernr, name, dorfnr, geschlecht, beruf, gold, status) [GEGENSTAND](#) (gegenstand, javascript:query('continue'));

<http://www.lgis.informatik.uni-kl.de/extra/game/>