
Non-Standard-Datenbanken

Probabilistische Datenbanken

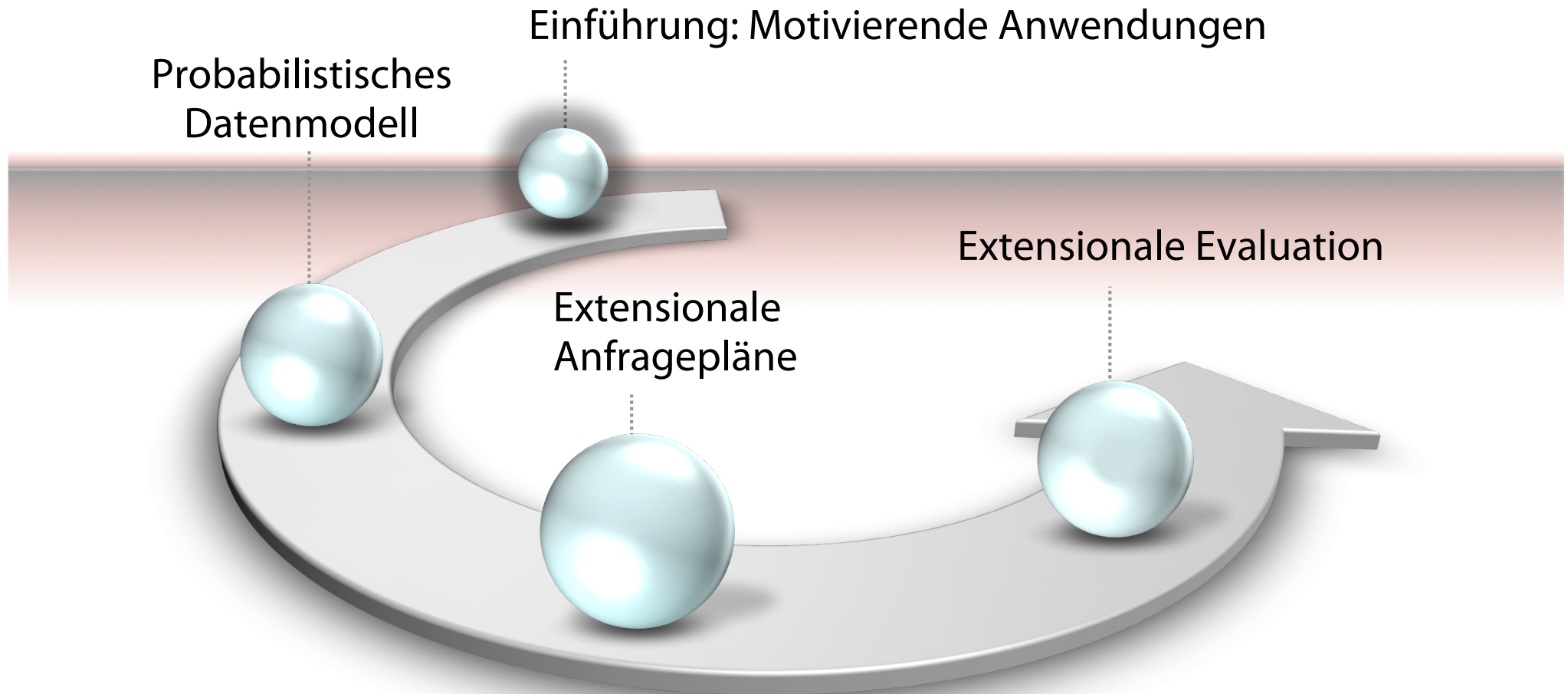
Prof. Dr. Ralf Möller

Universität zu Lübeck

Institut für Informationssysteme

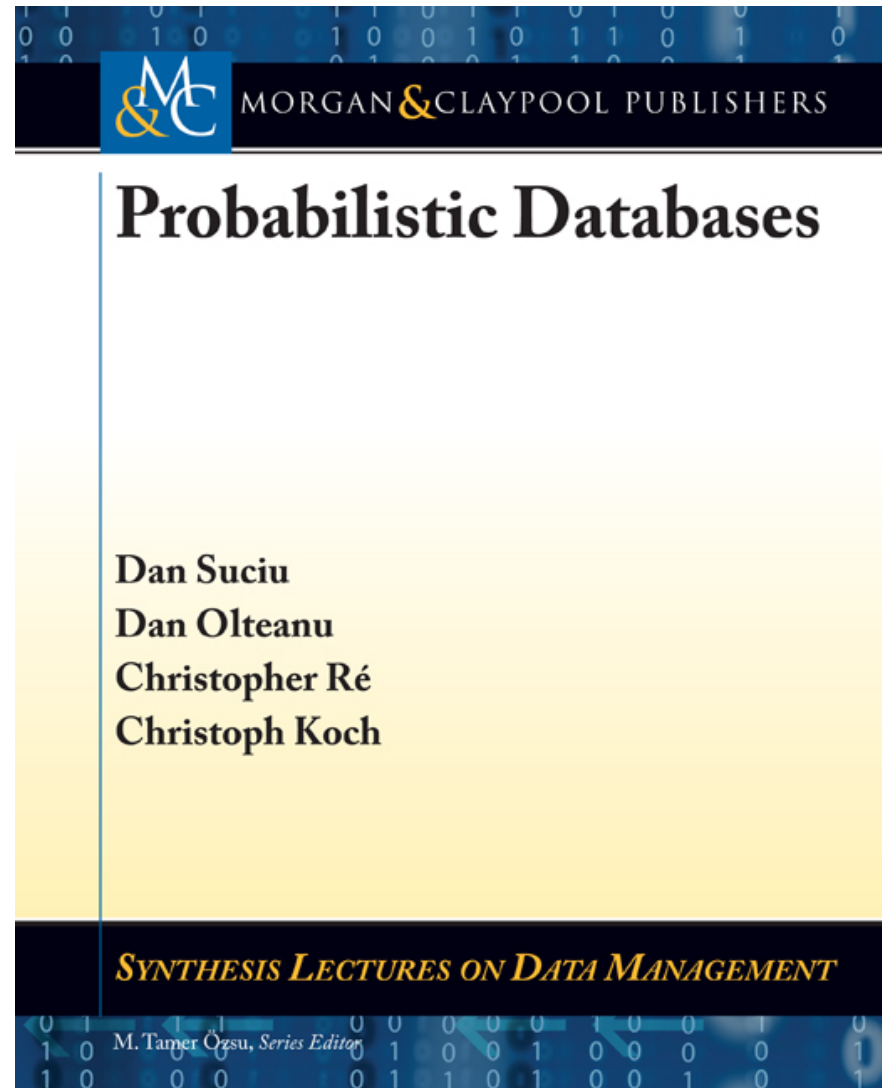
Non-Standard-Datenbanken

Probabilistische Datenbanken



Danksagung

Die Präsentationen sind
nach einem Tutorial von Dan
Suciu gestaltet und basieren
auf dem Lehrbuch
Probabilistic Databases



Probabilistische Datenbanken

- **Daten**: Relationale Daten plus **Wahrscheinlichkeiten**, um Grad der Unsicherheit auszudrücken
- **Anfragen**: SQL-Anfragen, deren Antworten annotiert sind mit **Ausgabewahrscheinlichkeiten**
- Ermöglicht Ihnen einen neuen Blick auf beides, Datenbanken und Wahrscheinlichkeiten

Beispiel 1: Informationsextraktion

52-A Goregaon West Mumbai 400 076

Extraktion

Standard DB: Speichere nur
wahrscheinlichste Extraktion

Id	House_no	Area	City	Pincode	Prob
1	52	Goregaon West	Mumbai	400 062	0.1
1	52-A	Goregaon	West Mumbai	400 062	0.2
1	52-A	Goregaon West	Mumbai	400 062	0.5
1	52	Goregaon	West Mumbai	400 062	0.2

Probabilistische DB: Speichert die meisten/alle
Extraktionen um **Recall zu erhöhen**

Kernidee: Wahrscheinlichkeiten gegeben durch Extraktion
korrelieren mit der Präzision der Extraktion

Beispiel 2: Modellierung fehlender Daten

Kernidee:
Inferiere Verteilung
für fehlende Daten.

Stoyanovich,
Davidson, Milo,
Tannen: Deriving
probabilistic
databases with
inference
ensembles. ICDE
2011

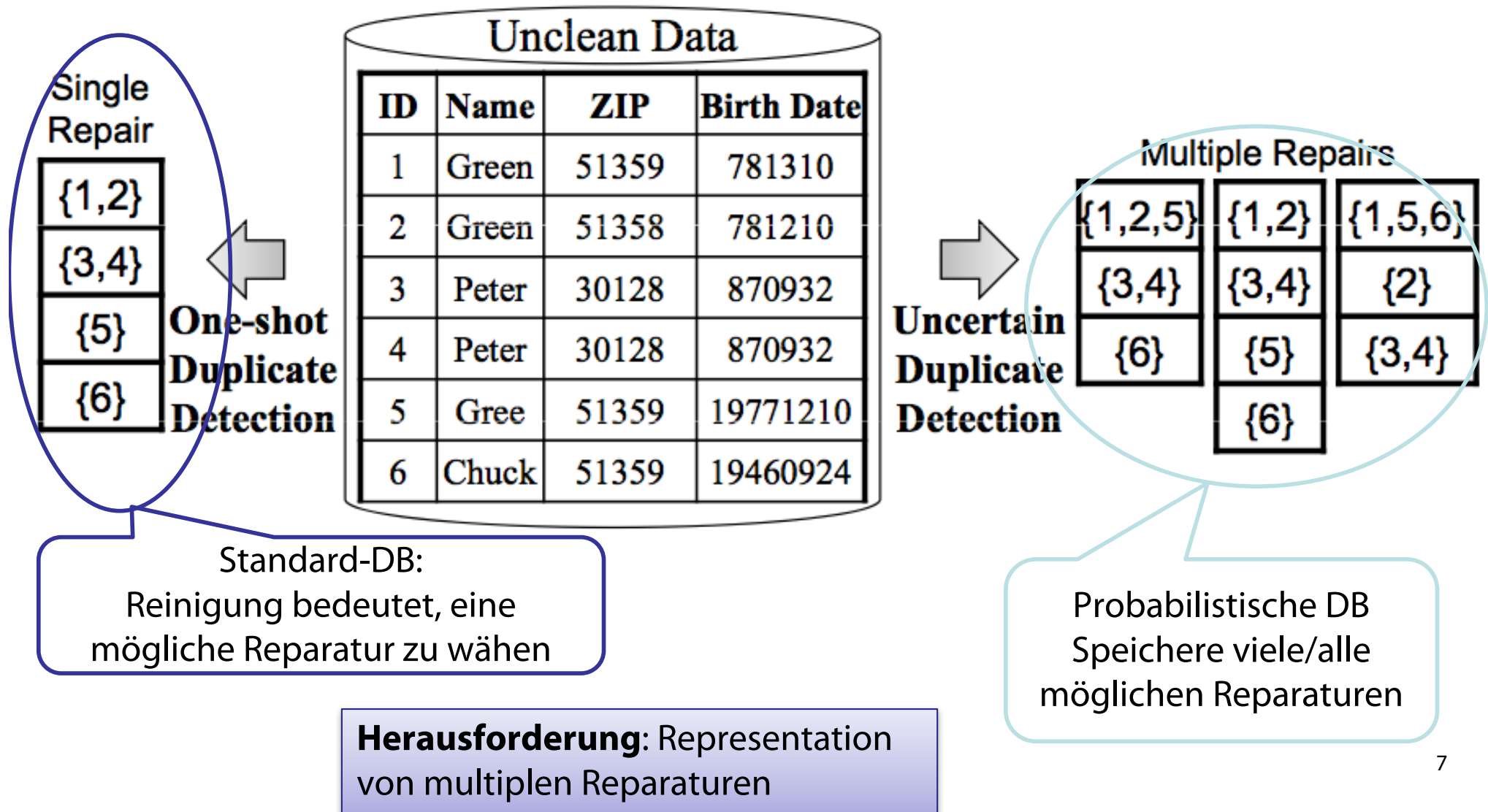
id	age	edu	inc	nw
t1	20	HS	?	?
t2	20	BS	50K	100K
t3	20	?	50K	?
t4	20	HS	100K	500K
t5	20	?	?	?
t6	20	HS	50K	100K
t7	20	HS	50K	500K
t8	?	HS	?	?
t9	30	BS	100K	100K
t10	30	?	100K	?
t11	30	HS	?	?
t12	30	MS	?	?
t13	40	BS	100K	100K
t14	40	HS	?	?
t15	40	BS	50K	500K
t16	40	HS	?	500K
t17	40	HS	100K	500K

Standard-DB: NULL

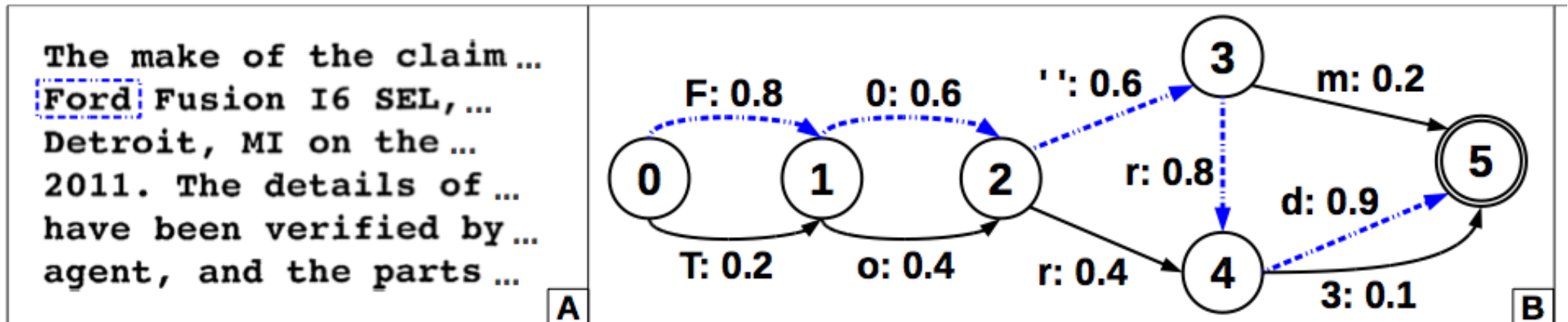
Probabilistische DB:
Verteilung auf mögl. Werten

id	age	edu	inc	nw	prob
t12.1	30	MS	50K	100K	0.30
t12.2	30	MS	50K	500K	0.45
t12.3	30	MS	100K	100K	0.10
t12.4	30	MS	100K	500K	0.15

Beispiel 3: Datenreinigung



Beispiel 4: OCR



Verwendung von OCRopus von Google Books: Ausgabe ist stochastischer Automat

Üblicherweise wird nur Maximum A priori Estimate (MAP) gespeichert

Mit probabilistischer Databasis: Speicherung verschiedener Möglichkeiten: Erhöhe Recall.

```
SELECT DocId, Loss
FROM Claims
WHERE Year = 2010
      AND DocData LIKE '%Ford%';
```

Zusammenfassung der Anwendungen

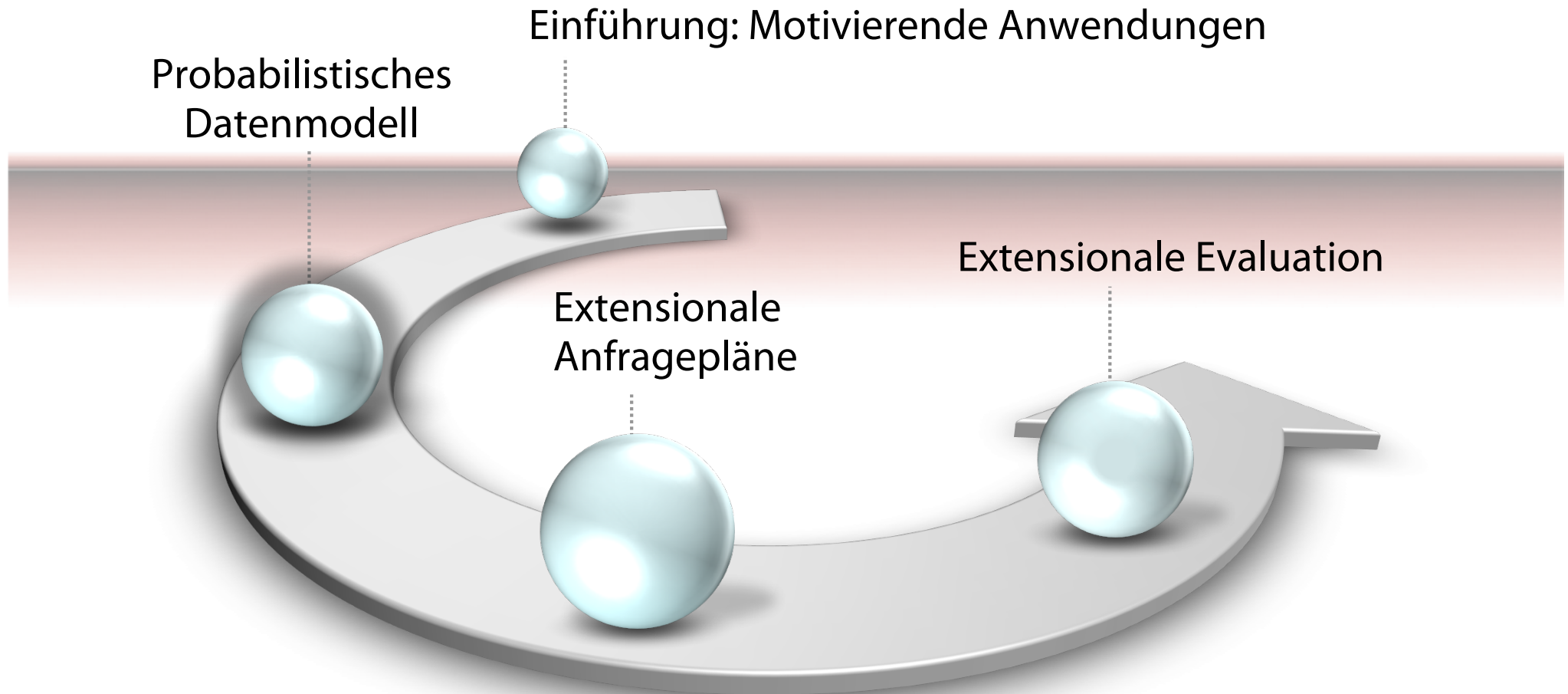
- Strukturierte, aber unsichere Daten
- Modelliert als **probabilistische Daten**
- Antworten für **SQL queries** annotiert mit **Wahrscheinlichkeiten**

Probabilistische Datenbank:

- Kombination aus Standard-Datenmanagement mit probabilistischer Inferenz

Non-Standard-Datenbanken

Probabilistische Datenbanken



Wiederholung: Relationales Datenmodell

Daten:
gespeichert in Relationen (= Tabellen)

Owner

Name	Object
Joe	Book302
Joe	Laptop77
Jim	Laptop77
Fred	GgleGlass

Location

<u>Object</u>	<u>Time</u>	Loc
Laptop77	5:07	Hall
Laptop77	9:05	Office
Book302	8:18	Office

Wiederholung: Relationales Datenmodell

Daten:
gespeichert in Relationen (= Tabellen)

Owner

Name	Object
Joe	Book302
Joe	Laptop77
Jim	Laptop77
Fred	GgleGlass

Location

Object	Time	Loc
Laptop77	5:07	Hall
Laptop77	9:05	Office
Book302	8:18	Office

Anfragen: SQL,

Find all owners of objects in the Office

-- SQL: z.B. Postgres

```
SELECT DISTINCT Owner.name
FROM Owner, Location
WHERE Owner.object = Location.object
and Location.loc = 'Office'
```


Wiederholung: Relationales Datenmodell

Daten:
gespeichert in Relationen (= Tabellen)

Owner

Name	Object
Joe	Book302
Joe	Laptop77
Jim	Laptop77
Fred	GgleGlass

Location

Object	Time	Loc
Laptop77	5:07	Hall
Laptop77	9:05	Office
Book302	8:18	Office

Anfragen: SQL,

Find all owners of objects in the Office

-- SQL: z.B. Postgres

```
SELECT DISTINCT Owner.name
FROM Owner, Location
WHERE Owner.object = Location.object
and Location.loc = 'Office'
```

Vereinigung konjunktiver Anfragen
Unions of Conjunctive Queries (UCQs)

$$Q(z) = \text{Owner}(z,x), \text{Location}(x,t,y), y='Office'$$

NB x,t sind existenzquantifiziert:

$$Q(z) = \exists x \exists t (\text{Owner}(z,x), \text{Location}(x,t,'Office'))$$

Wiederholung: Relationales Datenmodell

Daten:
gespeichert in Relationen (= Tabellen)

Owner

Name	Object
Joe	Book302
Joe	Laptop77
Jim	Laptop77
Fred	GgleGlass

Location

Object	Time	Loc
Laptop77	5:07	Hall
Laptop77	9:05	Office
Book302	8:18	Office

Anfragen: SQL,

Find all owners of objects in the Office

-- SQL: z.B. Postgres

```
SELECT DISTINCT Owner.name
FROM Owner, Location
WHERE Owner.object = Location.object
and Location.loc = 'Office'
```

Antwort: Q=

Name
Joe
Jim

Vereinigung konjunktiver Anfragen
Unions of Conjunctive Queries (UCQs)

$$Q(z) = \text{Owner}(z,x), \text{Location}(x,t,y), y='Office'$$

NB x,t sind existenzquantifiziert:

$$Q(z) = \exists x \exists t (\text{Owner}(z,x), \text{Location}(x,t,'Office'))$$

Wiederholung: Relationales Datenmodell

Daten:
gespeichert in Relationen (= Tabellen)

Owner		Location		
Name	Object	Object	Time	Loc
Joe	Book302	Laptop77	5:07	Hall
Joe	Laptop77	Laptop77	9:05	Office
Jim	Laptop77	Book302	8:18	Office
Fred	GgleGlass			

Anfragen: SQL,

Find all owners of objects in the Office

-- SQL: z.B. Postgres

```
SELECT DISTINCT Owner.name
FROM Owner, Location
WHERE Owner.object = Location.object
and Location.loc = 'Office'
```

Antwort: Q=

Name
Joe
Jim

Vereinigung konjunktiver Anfragen
Unions of Conjunctive Queries (UCQs)

$$Q(z) = \text{Owner}(z,x), \text{Location}(x,t,y), y='Office'$$

NB x,t sind existenzquantifiziert:

$$Q(z) = \exists x \exists t (\text{Owner}(z,x), \text{Location}(x,t,'Office'))$$

Wiederholung: Komplexität der Anfragebeantwortung

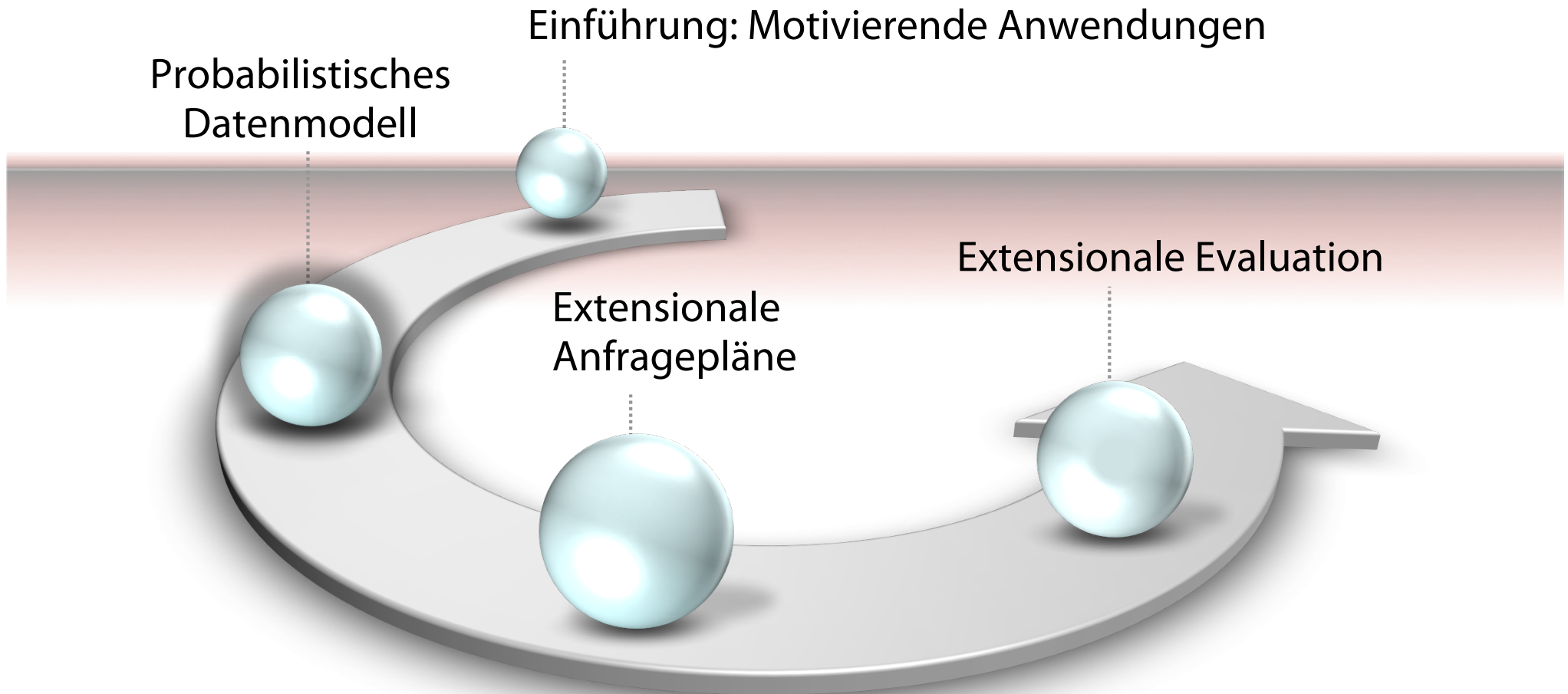
Anfrage Q , Datenbank D

- Datenkomplexität:
fix Q , Komplexität = $f(D)$
- Anfragekomplexität:
fix D , Komplexität = $f(Q)$
- Kombinierte Komplexität: Komplexität = $f(D, Q)$

Datenkomplexität wird im Bereich
der Datenbankforschung betrachtet

Non-Standard-Datenbanken

Probabilistische Datenbanken



Unvollständige Datenbank

Definition Eine **unvollständige Datenbank** ist eine endliche Menge von Datenbankinstanzen

$$\mathbf{W} = (W_1, W_2, \dots, W_n)$$

Jedes W_i heißt mögliche Welt

Unvollständige Datenbank

Definition Eine **unvollständige Datenbank** ist eine endliche Menge von Datenbankinstanzen

$$\mathbf{W} = (W_1, W_2, \dots, W_n)$$

Jedes W_i heißt mögliche Welt

W_1	W_2	W_3	W_4																						
Owner <table><tr><th>Name</th><th>Object</th></tr><tr><td>Joe</td><td>Book302</td></tr><tr><td>Joe</td><td>Laptop77</td></tr><tr><td>Jim</td><td>Laptop77</td></tr><tr><td>Fred</td><td>GgleGlass</td></tr></table> Location <table><tr><th>Object</th><th>Time</th><th>Loc</th></tr><tr><td>Laptop77</td><td>5:07</td><td>Hall</td></tr><tr><td>Laptop77</td><td>9:05</td><td>Office</td></tr><tr><td>Book302</td><td>8:18</td><td>Office</td></tr></table>	Name	Object	Joe	Book302	Joe	Laptop77	Jim	Laptop77	Fred	GgleGlass	Object	Time	Loc	Laptop77	5:07	Hall	Laptop77	9:05	Office	Book302	8:18	Office			
Name	Object																								
Joe	Book302																								
Joe	Laptop77																								
Jim	Laptop77																								
Fred	GgleGlass																								
Object	Time	Loc																							
Laptop77	5:07	Hall																							
Laptop77	9:05	Office																							
Book302	8:18	Office																							

Unvollständige Datenbank

Definition Eine **unvollständige Datenbank** ist eine endliche Menge von Datenbankinstanzen

$$\mathbf{W} = (W_1, W_2, \dots, W_n)$$

Jedes W_i heißt mögliche Welt

W_1	W_2	W_3	W_4																																				
Owner <table><tr><th>Name</th><th>Object</th></tr><tr><td>Joe</td><td>Book302</td></tr><tr><td>Joe</td><td>Laptop77</td></tr><tr><td>Jim</td><td>Laptop77</td></tr><tr><td>Fred</td><td>GgleGlass</td></tr></table> Location <table><tr><th>Object</th><th>Time</th><th>Loc</th></tr><tr><td>Laptop77</td><td>5:07</td><td>Hall</td></tr><tr><td>Laptop77</td><td>9:05</td><td>Office</td></tr><tr><td>Book302</td><td>8:18</td><td>Office</td></tr></table>	Name	Object	Joe	Book302	Joe	Laptop77	Jim	Laptop77	Fred	GgleGlass	Object	Time	Loc	Laptop77	5:07	Hall	Laptop77	9:05	Office	Book302	8:18	Office	Owner <table><tr><th>Name</th><th>Object</th></tr><tr><td>Joe</td><td>Book302</td></tr><tr><td>Jim</td><td>Laptop77</td></tr><tr><td>Fred</td><td>GgleGlass</td></tr></table> Location <table><tr><th>Object</th><th>Time</th><th>Loc</th></tr><tr><td>Book302</td><td>8:18</td><td>Office</td></tr></table>	Name	Object	Joe	Book302	Jim	Laptop77	Fred	GgleGlass	Object	Time	Loc	Book302	8:18	Office		
Name	Object																																						
Joe	Book302																																						
Joe	Laptop77																																						
Jim	Laptop77																																						
Fred	GgleGlass																																						
Object	Time	Loc																																					
Laptop77	5:07	Hall																																					
Laptop77	9:05	Office																																					
Book302	8:18	Office																																					
Name	Object																																						
Joe	Book302																																						
Jim	Laptop77																																						
Fred	GgleGlass																																						
Object	Time	Loc																																					
Book302	8:18	Office																																					

Unvollständige Datenbank

Definition Eine **unvollständige Datenbank** ist eine endliche Menge von Datenbankinstanzen

$$\mathbf{W} = (W_1, W_2, \dots, W_n)$$

Jedes W_i heißt mögliche Welt

W_1	W_2	W_3	W_4																																							
Owner <table><tr><th>Name</th><th>Object</th></tr><tr><td>Joe</td><td>Book302</td></tr><tr><td>Joe</td><td>Laptop77</td></tr><tr><td>Jim</td><td>Laptop77</td></tr><tr><td>Fred</td><td>GgleGlass</td></tr></table>	Name	Object	Joe	Book302	Joe	Laptop77	Jim	Laptop77	Fred	GgleGlass	Owner <table><tr><th>Name</th><th>Object</th></tr><tr><td>Joe</td><td>Book302</td></tr><tr><td>Jim</td><td>Laptop77</td></tr><tr><td>Fred</td><td>GgleGlass</td></tr></table>	Name	Object	Joe	Book302	Jim	Laptop77	Fred	GgleGlass	Owner <table><tr><th>Name</th><th>Object</th></tr><tr><td>Jim</td><td>Laptop77</td></tr></table>	Name	Object	Jim	Laptop77	Owner <table><tr><th>Name</th><th>Object</th></tr><tr><td>Joe</td><td>Book302</td></tr><tr><td>Jim</td><td>Laptop77</td></tr><tr><td>Fred</td><td>GgleGlass</td></tr></table>	Name	Object	Joe	Book302	Jim	Laptop77	Fred	GgleGlass									
Name	Object																																									
Joe	Book302																																									
Joe	Laptop77																																									
Jim	Laptop77																																									
Fred	GgleGlass																																									
Name	Object																																									
Joe	Book302																																									
Jim	Laptop77																																									
Fred	GgleGlass																																									
Name	Object																																									
Jim	Laptop77																																									
Name	Object																																									
Joe	Book302																																									
Jim	Laptop77																																									
Fred	GgleGlass																																									
Location <table><tr><th>Object</th><th>Time</th><th>Loc</th></tr><tr><td>Laptop77</td><td>5:07</td><td>Hall</td></tr><tr><td>Laptop77</td><td>9:05</td><td>Office</td></tr><tr><td>Book302</td><td>8:18</td><td>Office</td></tr></table>	Object	Time	Loc	Laptop77	5:07	Hall	Laptop77	9:05	Office	Book302	8:18	Office	Location <table><tr><th>Object</th><th>Time</th><th>Loc</th></tr><tr><td>Book302</td><td>8:18</td><td>Office</td></tr></table>	Object	Time	Loc	Book302	8:18	Office	Location <table><tr><th>Object</th><th>Time</th><th>Loc</th></tr><tr><td>Laptop77</td><td>5:07</td><td>Hall</td></tr><tr><td>Laptop77</td><td>9:05</td><td>Office</td></tr></table>	Object	Time	Loc	Laptop77	5:07	Hall	Laptop77	9:05	Office	Location <table><tr><th>Object</th><th>Time</th><th>Loc</th></tr><tr><td>Laptop77</td><td>5:07</td><td>Hall</td></tr><tr><td>Laptop77</td><td>9:05</td><td>Office</td></tr><tr><td>Book302</td><td>8:18</td><td>Office</td></tr></table>	Object	Time	Loc	Laptop77	5:07	Hall	Laptop77	9:05	Office	Book302	8:18	Office
Object	Time	Loc																																								
Laptop77	5:07	Hall																																								
Laptop77	9:05	Office																																								
Book302	8:18	Office																																								
Object	Time	Loc																																								
Book302	8:18	Office																																								
Object	Time	Loc																																								
Laptop77	5:07	Hall																																								
Laptop77	9:05	Office																																								
Object	Time	Loc																																								
Laptop77	5:07	Hall																																								
Laptop77	9:05	Office																																								
Book302	8:18	Office																																								

Unvollständige Datenbank: Anfragenesemantik

Definition Gegeben eine Anfrage Q , eine unvollständige DB W :

- Eine Antwort t ist **sicher (certain)**, falls $\forall W_i, t \in Q(W_i)$
- Eine Antwort t ist **möglich (possible)** falls $\exists W_i, t \in Q(W_i)$

Unvollständige Datenbank: Anfragenesemantik

Definition Gegeben eine Anfrage Q , eine unvollständige DB W :

- Eine Antwort t ist **sicher (certain)**, falls $\forall W_i, t \in Q(W_i)$
- Eine Antwort t ist **möglich (possible)** falls $\exists W_i, t \in Q(W_i)$

$$Q(z) = \text{Owner}(z,x), \\ \text{Location}(x,t,\text{'Office'})$$

W_1	W_2	W_3	W_4																																																																					
Owner <table><tr><th>Name</th><th>Object</th></tr><tr><td>Joe</td><td>Book302</td></tr><tr><td>Joe</td><td>Laptop77</td></tr><tr><td>Jim</td><td>Laptop77</td></tr><tr><td>Fred</td><td>GgleGlass</td></tr></table> Location <table><tr><th>Object</th><th>Time</th><th>Loc</th></tr><tr><td>Laptop77</td><td>5:07</td><td>Hall</td></tr><tr><td>Laptop77</td><td>9:05</td><td>Office</td></tr><tr><td>Book302</td><td>8:18</td><td>Office</td></tr></table>	Name	Object	Joe	Book302	Joe	Laptop77	Jim	Laptop77	Fred	GgleGlass	Object	Time	Loc	Laptop77	5:07	Hall	Laptop77	9:05	Office	Book302	8:18	Office	Owner <table><tr><th>Name</th><th>Object</th></tr><tr><td>Joe</td><td>Book302</td></tr><tr><td>Jim</td><td>Laptop77</td></tr><tr><td>Fred</td><td>GgleGlass</td></tr></table> Location <table><tr><th>Object</th><th>Time</th><th>Loc</th></tr><tr><td>Book302</td><td>8:18</td><td>Office</td></tr></table>	Name	Object	Joe	Book302	Jim	Laptop77	Fred	GgleGlass	Object	Time	Loc	Book302	8:18	Office	Owner <table><tr><th>Name</th><th>Object</th></tr><tr><td>Joe</td><td>Laptop77</td></tr></table> Location <table><tr><th>Object</th><th>Time</th><th>Loc</th></tr><tr><td>Laptop77</td><td>5:07</td><td>Hall</td></tr><tr><td>Laptop77</td><td>9:05</td><td>Office</td></tr></table>	Name	Object	Joe	Laptop77	Object	Time	Loc	Laptop77	5:07	Hall	Laptop77	9:05	Office	Owner <table><tr><th>Name</th><th>Object</th></tr><tr><td>Joe</td><td>Book302</td></tr><tr><td>Jim</td><td>Laptop77</td></tr><tr><td>Fred</td><td>GgleGlass</td></tr></table> Location <table><tr><th>Object</th><th>Time</th><th>Loc</th></tr><tr><td>Laptop77</td><td>5:07</td><td>Hall</td></tr><tr><td>Laptop77</td><td>9:05</td><td>Office</td></tr><tr><td>Book302</td><td>8:18</td><td>Office</td></tr></table>	Name	Object	Joe	Book302	Jim	Laptop77	Fred	GgleGlass	Object	Time	Loc	Laptop77	5:07	Hall	Laptop77	9:05	Office	Book302	8:18	Office
Name	Object																																																																							
Joe	Book302																																																																							
Joe	Laptop77																																																																							
Jim	Laptop77																																																																							
Fred	GgleGlass																																																																							
Object	Time	Loc																																																																						
Laptop77	5:07	Hall																																																																						
Laptop77	9:05	Office																																																																						
Book302	8:18	Office																																																																						
Name	Object																																																																							
Joe	Book302																																																																							
Jim	Laptop77																																																																							
Fred	GgleGlass																																																																							
Object	Time	Loc																																																																						
Book302	8:18	Office																																																																						
Name	Object																																																																							
Joe	Laptop77																																																																							
Object	Time	Loc																																																																						
Laptop77	5:07	Hall																																																																						
Laptop77	9:05	Office																																																																						
Name	Object																																																																							
Joe	Book302																																																																							
Jim	Laptop77																																																																							
Fred	GgleGlass																																																																							
Object	Time	Loc																																																																						
Laptop77	5:07	Hall																																																																						
Laptop77	9:05	Office																																																																						
Book302	8:18	Office																																																																						

Unvollständige Datenbank: Anfragenesemantik

Definition Gegeben eine Anfrage Q , eine unvollständige DB W :

- Eine Antwort t ist **sicher (certain)**, falls $\forall W_i, t \in Q(W_i)$
- Eine Antwort t ist **möglich (possible)** falls $\exists W_i, t \in Q(W_i)$

$$Q(z) = \text{Owner}(z,x), \\ \text{Location}(x,t,\text{'Office'})$$

W_1	W_2	W_3	W_4																																																																					
Owner <table><tr><th>Name</th><th>Object</th></tr><tr><td>Joe</td><td>Book302</td></tr><tr><td>Joe</td><td>Laptop77</td></tr><tr><td>Jim</td><td>Laptop77</td></tr><tr><td>Fred</td><td>GgleGlass</td></tr></table> Location <table><tr><th>Object</th><th>Time</th><th>Loc</th></tr><tr><td>Laptop77</td><td>5:07</td><td>Hall</td></tr><tr><td>Laptop77</td><td>9:05</td><td>Office</td></tr><tr><td>Book302</td><td>8:18</td><td>Office</td></tr></table>	Name	Object	Joe	Book302	Joe	Laptop77	Jim	Laptop77	Fred	GgleGlass	Object	Time	Loc	Laptop77	5:07	Hall	Laptop77	9:05	Office	Book302	8:18	Office	Owner <table><tr><th>Name</th><th>Object</th></tr><tr><td>Joe</td><td>Book302</td></tr><tr><td>Jim</td><td>Laptop77</td></tr><tr><td>Fred</td><td>GgleGlass</td></tr></table> Location <table><tr><th>Object</th><th>Time</th><th>Loc</th></tr><tr><td>Book302</td><td>8:18</td><td>Office</td></tr></table>	Name	Object	Joe	Book302	Jim	Laptop77	Fred	GgleGlass	Object	Time	Loc	Book302	8:18	Office	Owner <table><tr><th>Name</th><th>Object</th></tr><tr><td>Joe</td><td>Laptop77</td></tr></table> Location <table><tr><th>Object</th><th>Time</th><th>Loc</th></tr><tr><td>Laptop77</td><td>5:07</td><td>Hall</td></tr><tr><td>Laptop77</td><td>9:05</td><td>Office</td></tr></table>	Name	Object	Joe	Laptop77	Object	Time	Loc	Laptop77	5:07	Hall	Laptop77	9:05	Office	Owner <table><tr><th>Name</th><th>Object</th></tr><tr><td>Joe</td><td>Book302</td></tr><tr><td>Jim</td><td>Laptop77</td></tr><tr><td>Fred</td><td>GgleGlass</td></tr></table> Location <table><tr><th>Object</th><th>Time</th><th>Loc</th></tr><tr><td>Laptop77</td><td>5:07</td><td>Hall</td></tr><tr><td>Laptop77</td><td>9:05</td><td>Office</td></tr><tr><td>Book302</td><td>8:18</td><td>Office</td></tr></table>	Name	Object	Joe	Book302	Jim	Laptop77	Fred	GgleGlass	Object	Time	Loc	Laptop77	5:07	Hall	Laptop77	9:05	Office	Book302	8:18	Office
Name	Object																																																																							
Joe	Book302																																																																							
Joe	Laptop77																																																																							
Jim	Laptop77																																																																							
Fred	GgleGlass																																																																							
Object	Time	Loc																																																																						
Laptop77	5:07	Hall																																																																						
Laptop77	9:05	Office																																																																						
Book302	8:18	Office																																																																						
Name	Object																																																																							
Joe	Book302																																																																							
Jim	Laptop77																																																																							
Fred	GgleGlass																																																																							
Object	Time	Loc																																																																						
Book302	8:18	Office																																																																						
Name	Object																																																																							
Joe	Laptop77																																																																							
Object	Time	Loc																																																																						
Laptop77	5:07	Hall																																																																						
Laptop77	9:05	Office																																																																						
Name	Object																																																																							
Joe	Book302																																																																							
Jim	Laptop77																																																																							
Fred	GgleGlass																																																																							
Object	Time	Loc																																																																						
Laptop77	5:07	Hall																																																																						
Laptop77	9:05	Office																																																																						
Book302	8:18	Office																																																																						
 <table><tr><td>Joe</td></tr><tr><td>Jim</td></tr></table>	Joe	Jim	$Q=$ <table><tr><td>Joe</td></tr></table>	Joe	$Q=$ <table><tr><td>Joe</td></tr></table>	Joe	$Q=$ <table><tr><td>Joe</td></tr><tr><td>Jim</td></tr></table> 24	Joe	Jim																																																															
Joe																																																																								
Jim																																																																								
Joe																																																																								
Joe																																																																								
Joe																																																																								
Jim																																																																								

Unvollständige Datenbank: Anfragenesemantik

Definition Gegeben eine Anfrage Q , und eine unvollständige Datenbasis W :

- Eine Antwort t heißt **sicher (certain)**, falls $\forall W_i: t \in Q(W_i)$
- Eine Antwort t heißt **possible**, falls $\exists W_i: t \in Q(W_i)$

$Q(z) = \text{Owner}(z,x),$
 $\text{Location}(x,t,\text{'Office'})$

Sichere Antwort für Q : Joe

Mögliche Antwort f. Q : Joe, Jim

W_1	W_2	W_3	W_4																																							
Owner <table><tr><th>Name</th><th>Object</th></tr><tr><td>Joe</td><td>Book302</td></tr><tr><td>Joe</td><td>Laptop77</td></tr><tr><td>Jim</td><td>Laptop77</td></tr><tr><td>Fred</td><td>GgleGlass</td></tr></table>	Name	Object	Joe	Book302	Joe	Laptop77	Jim	Laptop77	Fred	GgleGlass	Owner <table><tr><th>Name</th><th>Object</th></tr><tr><td>Joe</td><td>Book302</td></tr><tr><td>Jim</td><td>Laptop77</td></tr><tr><td>Fred</td><td>GgleGlass</td></tr></table>	Name	Object	Joe	Book302	Jim	Laptop77	Fred	GgleGlass	Owner <table><tr><th>Name</th><th>Object</th></tr><tr><td>Joe</td><td>Laptop77</td></tr></table>	Name	Object	Joe	Laptop77	Owner <table><tr><th>Name</th><th>Object</th></tr><tr><td>Joe</td><td>Book302</td></tr><tr><td>Jim</td><td>Laptop77</td></tr><tr><td>Fred</td><td>GgleGlass</td></tr></table>	Name	Object	Joe	Book302	Jim	Laptop77	Fred	GgleGlass									
Name	Object																																									
Joe	Book302																																									
Joe	Laptop77																																									
Jim	Laptop77																																									
Fred	GgleGlass																																									
Name	Object																																									
Joe	Book302																																									
Jim	Laptop77																																									
Fred	GgleGlass																																									
Name	Object																																									
Joe	Laptop77																																									
Name	Object																																									
Joe	Book302																																									
Jim	Laptop77																																									
Fred	GgleGlass																																									
Location <table><tr><th>Object</th><th>Time</th><th>Loc</th></tr><tr><td>Laptop77</td><td>5:07</td><td>Hall</td></tr><tr><td>Laptop77</td><td>9:05</td><td>Office</td></tr><tr><td>Book302</td><td>8:18</td><td>Office</td></tr></table>	Object	Time	Loc	Laptop77	5:07	Hall	Laptop77	9:05	Office	Book302	8:18	Office	Location <table><tr><th>Object</th><th>Time</th><th>Loc</th></tr><tr><td>Book302</td><td>8:18</td><td>Office</td></tr></table>	Object	Time	Loc	Book302	8:18	Office	Location <table><tr><th>Object</th><th>Time</th><th>Loc</th></tr><tr><td>Laptop77</td><td>5:07</td><td>Hall</td></tr><tr><td>Laptop77</td><td>9:05</td><td>Office</td></tr></table>	Object	Time	Loc	Laptop77	5:07	Hall	Laptop77	9:05	Office	Location <table><tr><th>Object</th><th>Time</th><th>Loc</th></tr><tr><td>Laptop77</td><td>5:07</td><td>Hall</td></tr><tr><td>Laptop77</td><td>9:05</td><td>Office</td></tr><tr><td>Book302</td><td>8:18</td><td>Office</td></tr></table>	Object	Time	Loc	Laptop77	5:07	Hall	Laptop77	9:05	Office	Book302	8:18	Office
Object	Time	Loc																																								
Laptop77	5:07	Hall																																								
Laptop77	9:05	Office																																								
Book302	8:18	Office																																								
Object	Time	Loc																																								
Book302	8:18	Office																																								
Object	Time	Loc																																								
Laptop77	5:07	Hall																																								
Laptop77	9:05	Office																																								
Object	Time	Loc																																								
Laptop77	5:07	Hall																																								
Laptop77	9:05	Office																																								
Book302	8:18	Office																																								
 <table><tr><td>Joe</td></tr><tr><td>Jim</td></tr></table>	Joe	Jim	$Q=$ <table><tr><td>Joe</td></tr></table>	Joe	$Q=$ <table><tr><td>Joe</td></tr></table>	Joe	$Q=$ <table><tr><td>Joe</td></tr><tr><td>Jim</td></tr></table>	Joe	Jim																																	
Joe																																										
Jim																																										
Joe																																										
Joe																																										
Joe																																										
Jim																																										

Probabilistische Datenbank

Definition Eine **probabilistische DB** ist ein Tupel $(\mathbf{W}, \mathbf{P})$, wobei $\mathbf{W}$ eine unvollständige DB und $\mathbf{P}: \mathbf{W} \rightarrow [0,1]$ eine Wahrscheinlichkeitsverteilung ist: $\sum_{i=1,n} \mathbf{P}(W_i) = 1$

Probabilistische Datenbank

Definition Eine **probabilistische DB** ist ein Tupel $(\mathbf{W}, \mathbf{P})$, wobei $\mathbf{W}$ eine unvollständige DB und $\mathbf{P}: \mathbf{W} \rightarrow [0,1]$ eine Wahrscheinlichkeitsverteilung ist: $\sum_{i=1,n} P(W_i) = 1$

W_1			W_2			W_3			W_4																																									
Owner 0.3			Owner 0.4			Owner 0.2			Owner 0.1																																									
<table><tr><th>Name</th><th>Object</th></tr><tr><td>Joe</td><td>Book302</td></tr><tr><td>Joe</td><td>Laptop77</td></tr><tr><td>Jim</td><td>Laptop77</td></tr><tr><td>Fred</td><td>GgleGlass</td></tr></table>			Name	Object	Joe	Book302	Joe	Laptop77	Jim	Laptop77	Fred	GgleGlass	<table><tr><th>Name</th><th>Object</th></tr><tr><td>Joe</td><td>Book302</td></tr><tr><td>Jim</td><td>Laptop77</td></tr><tr><td>Fred</td><td>GgleGlass</td></tr></table>			Name	Object	Joe	Book302	Jim	Laptop77	Fred	GgleGlass	<table><tr><th>Name</th><th>Object</th></tr><tr><td>Jim</td><td>Laptop77</td></tr></table>			Name	Object	Jim	Laptop77	<table><tr><th>Name</th><th>Object</th></tr><tr><td>Joe</td><td>Book302</td></tr><tr><td>Jim</td><td>Laptop77</td></tr><tr><td>Fred</td><td>GgleGlass</td></tr></table>			Name	Object	Joe	Book302	Jim	Laptop77	Fred	GgleGlass									
Name	Object																																																	
Joe	Book302																																																	
Joe	Laptop77																																																	
Jim	Laptop77																																																	
Fred	GgleGlass																																																	
Name	Object																																																	
Joe	Book302																																																	
Jim	Laptop77																																																	
Fred	GgleGlass																																																	
Name	Object																																																	
Jim	Laptop77																																																	
Name	Object																																																	
Joe	Book302																																																	
Jim	Laptop77																																																	
Fred	GgleGlass																																																	
Location			Location			Location			Location																																									
<table><tr><th>Object</th><th>Time</th><th>Loc</th></tr><tr><td>Laptop77</td><td>5:07</td><td>Hall</td></tr><tr><td>Laptop77</td><td>9:05</td><td>Office</td></tr><tr><td>Book302</td><td>8:18</td><td>Office</td></tr></table>			Object	Time	Loc	Laptop77	5:07	Hall	Laptop77	9:05	Office	Book302	8:18	Office	<table><tr><th>Object</th><th>Time</th><th>Loc</th></tr><tr><td>Book302</td><td>8:18</td><td>Office</td></tr></table>			Object	Time	Loc	Book302	8:18	Office	<table><tr><th>Object</th><th>Time</th><th>Loc</th></tr><tr><td>Laptop77</td><td>5:07</td><td>Hall</td></tr><tr><td>Laptop77</td><td>9:05</td><td>Office</td></tr></table>			Object	Time	Loc	Laptop77	5:07	Hall	Laptop77	9:05	Office	<table><tr><th>Object</th><th>Time</th><th>Loc</th></tr><tr><td>Laptop77</td><td>5:07</td><td>Hall</td></tr><tr><td>Laptop77</td><td>9:05</td><td>Office</td></tr><tr><td>Book302</td><td>8:18</td><td>Office</td></tr></table>			Object	Time	Loc	Laptop77	5:07	Hall	Laptop77	9:05	Office	Book302	8:18	Office
Object	Time	Loc																																																
Laptop77	5:07	Hall																																																
Laptop77	9:05	Office																																																
Book302	8:18	Office																																																
Object	Time	Loc																																																
Book302	8:18	Office																																																
Object	Time	Loc																																																
Laptop77	5:07	Hall																																																
Laptop77	9:05	Office																																																
Object	Time	Loc																																																
Laptop77	5:07	Hall																																																
Laptop77	9:05	Office																																																
Book302	8:18	Office																																																

Probabilistische Datenbank: Anfragesemantik

Definition Gegeben eine Anfrage Q , eine probabilistische DB $(\mathbf{W}, \mathbf{P})$:

- Die Randwahrscheinlichkeit einer Antwort t ist:

$$\mathbf{P}(t) = \sum \{ \mathbf{P}(W_i) \mid W_i \in \mathbf{W}, t \in Q(W_i) \}$$

Probabilistische Datenbank: Anfragenesemantik

Definition Gegeben eine Anfrage Q , eine probabilistische DB $(\mathbf{W}, \mathbf{P})$:

- Die Randwahrscheinlichkeit einer Antwort t ist:

$$P(t) = \sum \{ P(W_i) \mid W_i \in \mathbf{W}, t \in Q(W_i) \}$$

$$Q(z) = \text{Owner}(z, x), \\ \text{Location}(x, t, \text{'Office'})$$

W_1	W_2	W_3	W_4																																							
Owner 0.3	Owner 0.4	Owner 0.2	Owner 0.1																																							
<table><tr><th>Name</th><th>Object</th></tr><tr><td>Joe</td><td>Book302</td></tr><tr><td>Joe</td><td>Laptop77</td></tr><tr><td>Jim</td><td>Laptop77</td></tr><tr><td>Fred</td><td>GgleGlass</td></tr></table>	Name	Object	Joe	Book302	Joe	Laptop77	Jim	Laptop77	Fred	GgleGlass	<table><tr><th>Name</th><th>Object</th></tr><tr><td>Joe</td><td>Book302</td></tr><tr><td>Jim</td><td>Laptop77</td></tr><tr><td>Fred</td><td>GgleGlass</td></tr></table>	Name	Object	Joe	Book302	Jim	Laptop77	Fred	GgleGlass	<table><tr><th>Name</th><th>Object</th></tr><tr><td>Jim</td><td>Laptop77</td></tr></table>	Name	Object	Jim	Laptop77	<table><tr><th>Name</th><th>Object</th></tr><tr><td>Joe</td><td>Book302</td></tr><tr><td>Jim</td><td>Laptop77</td></tr><tr><td>Fred</td><td>GgleGlass</td></tr></table>	Name	Object	Joe	Book302	Jim	Laptop77	Fred	GgleGlass									
Name	Object																																									
Joe	Book302																																									
Joe	Laptop77																																									
Jim	Laptop77																																									
Fred	GgleGlass																																									
Name	Object																																									
Joe	Book302																																									
Jim	Laptop77																																									
Fred	GgleGlass																																									
Name	Object																																									
Jim	Laptop77																																									
Name	Object																																									
Joe	Book302																																									
Jim	Laptop77																																									
Fred	GgleGlass																																									
Location	Location	Location	Location																																							
<table><tr><th>Object</th><th>Time</th><th>Loc</th></tr><tr><td>Laptop77</td><td>5:07</td><td>Hall</td></tr><tr><td>Laptop77</td><td>9:05</td><td>Office</td></tr><tr><td>Book302</td><td>8:18</td><td>Office</td></tr></table>	Object	Time	Loc	Laptop77	5:07	Hall	Laptop77	9:05	Office	Book302	8:18	Office	<table><tr><th>Object</th><th>Time</th><th>Loc</th></tr><tr><td>Book302</td><td>8:18</td><td>Office</td></tr></table>	Object	Time	Loc	Book302	8:18	Office	<table><tr><th>Object</th><th>Time</th><th>Loc</th></tr><tr><td>Laptop77</td><td>5:07</td><td>Hall</td></tr><tr><td>Laptop77</td><td>9:05</td><td>Office</td></tr></table>	Object	Time	Loc	Laptop77	5:07	Hall	Laptop77	9:05	Office	<table><tr><th>Object</th><th>Time</th><th>Loc</th></tr><tr><td>Laptop77</td><td>5:07</td><td>Hall</td></tr><tr><td>Laptop77</td><td>9:05</td><td>Office</td></tr><tr><td>Book302</td><td>8:18</td><td>Office</td></tr></table>	Object	Time	Loc	Laptop77	5:07	Hall	Laptop77	9:05	Office	Book302	8:18	Office
Object	Time	Loc																																								
Laptop77	5:07	Hall																																								
Laptop77	9:05	Office																																								
Book302	8:18	Office																																								
Object	Time	Loc																																								
Book302	8:18	Office																																								
Object	Time	Loc																																								
Laptop77	5:07	Hall																																								
Laptop77	9:05	Office																																								
Object	Time	Loc																																								
Laptop77	5:07	Hall																																								
Laptop77	9:05	Office																																								
Book302	8:18	Office																																								

Probabilistische Datenbank: Anfragenesemantik

Definition Gegeben eine Anfrage Q , eine probabilistische DB $(\mathbf{W}, \mathbf{P})$:

- Die Randwahrscheinlichkeit einer Antwort t ist:

$$P(t) = \sum \{ P(W_i) \mid W_i \in \mathbf{W}, t \in Q(W_i) \}$$

$$Q(z) = \text{Owner}(z, x), \\ \text{Location}(x, t, \text{'Office'})$$

W_1	W_2	W_3	W_4																																							
Owner 0.3	Owner 0.4	Owner 0.2	Owner 0.1																																							
<table><tr><th>Name</th><th>Object</th></tr><tr><td>Joe</td><td>Book302</td></tr><tr><td>Joe</td><td>Laptop77</td></tr><tr><td>Jim</td><td>Laptop77</td></tr><tr><td>Fred</td><td>GgleGlass</td></tr></table>	Name	Object	Joe	Book302	Joe	Laptop77	Jim	Laptop77	Fred	GgleGlass	<table><tr><th>Name</th><th>Object</th></tr><tr><td>Joe</td><td>Book302</td></tr><tr><td>Jim</td><td>Laptop77</td></tr><tr><td>Fred</td><td>GgleGlass</td></tr></table>	Name	Object	Joe	Book302	Jim	Laptop77	Fred	GgleGlass	<table><tr><th>Name</th><th>Object</th></tr><tr><td>Joe</td><td>Laptop77</td></tr></table>	Name	Object	Joe	Laptop77	<table><tr><th>Name</th><th>Object</th></tr><tr><td>Joe</td><td>Book302</td></tr><tr><td>Jim</td><td>Laptop77</td></tr><tr><td>Fred</td><td>GgleGlass</td></tr></table>	Name	Object	Joe	Book302	Jim	Laptop77	Fred	GgleGlass									
Name	Object																																									
Joe	Book302																																									
Joe	Laptop77																																									
Jim	Laptop77																																									
Fred	GgleGlass																																									
Name	Object																																									
Joe	Book302																																									
Jim	Laptop77																																									
Fred	GgleGlass																																									
Name	Object																																									
Joe	Laptop77																																									
Name	Object																																									
Joe	Book302																																									
Jim	Laptop77																																									
Fred	GgleGlass																																									
Location	Location	Location	Location																																							
<table><tr><th>Object</th><th>Time</th><th>Loc</th></tr><tr><td>Laptop77</td><td>5:07</td><td>Hall</td></tr><tr><td>Laptop77</td><td>9:05</td><td>Office</td></tr><tr><td>Book302</td><td>8:18</td><td>Office</td></tr></table>	Object	Time	Loc	Laptop77	5:07	Hall	Laptop77	9:05	Office	Book302	8:18	Office	<table><tr><th>Object</th><th>Time</th><th>Loc</th></tr><tr><td>Book302</td><td>8:18</td><td>Office</td></tr></table>	Object	Time	Loc	Book302	8:18	Office	<table><tr><th>Object</th><th>Time</th><th>Loc</th></tr><tr><td>Laptop77</td><td>5:07</td><td>Hall</td></tr><tr><td>Laptop77</td><td>9:05</td><td>Office</td></tr></table>	Object	Time	Loc	Laptop77	5:07	Hall	Laptop77	9:05	Office	<table><tr><th>Object</th><th>Time</th><th>Loc</th></tr><tr><td>Laptop77</td><td>5:07</td><td>Hall</td></tr><tr><td>Laptop77</td><td>9:05</td><td>Office</td></tr><tr><td>Book302</td><td>8:18</td><td>Office</td></tr></table>	Object	Time	Loc	Laptop77	5:07	Hall	Laptop77	9:05	Office	Book302	8:18	Office
Object	Time	Loc																																								
Laptop77	5:07	Hall																																								
Laptop77	9:05	Office																																								
Book302	8:18	Office																																								
Object	Time	Loc																																								
Book302	8:18	Office																																								
Object	Time	Loc																																								
Laptop77	5:07	Hall																																								
Laptop77	9:05	Office																																								
Object	Time	Loc																																								
Laptop77	5:07	Hall																																								
Laptop77	9:05	Office																																								
Book302	8:18	Office																																								
 <table><tr><td>Joe</td></tr><tr><td>Jim</td></tr></table>	Joe	Jim	$Q=$ <table><tr><td>Joe</td></tr></table>	Joe	$Q=$ <table><tr><td>Joe</td></tr></table>	Joe	$Q=$ <table><tr><td>Joe</td></tr><tr><td>Jim</td></tr></table>	Joe	Jim																																	
Joe																																										
Jim																																										
Joe																																										
Joe																																										
Joe																																										
Jim																																										

Probabilistische Datenbank: Anfragenesemantik

Definition Gegeben eine Anfrage Q , eine probabilistische DB $(\mathbf{W}, \mathbf{P})$:

- Die Randwahrscheinlichkeit einer Antwort t ist:

$$P(t) = \sum \{ P(W_i) \mid W_i \in \mathbf{W}, t \in Q(W_i) \}$$

$$Q(z) = \text{Owner}(z, x), \\ \text{Location}(x, t, \text{'Office'})$$

$$P(\text{Joe}) = 1.0$$

$$P(\text{Jim}) = 0.4$$

W_1			W_2			W_3			W_4																																									
Owner 0.3			Owner 0.4			Owner 0.2			Owner 0.1																																									
<table><tr><th>Name</th><th>Object</th></tr><tr><td>Joe</td><td>Book302</td></tr><tr><td>Joe</td><td>Laptop77</td></tr><tr><td>Jim</td><td>Laptop77</td></tr><tr><td>Fred</td><td>GgleGlass</td></tr></table>			Name	Object	Joe	Book302	Joe	Laptop77	Jim	Laptop77	Fred	GgleGlass	<table><tr><th>Name</th><th>Object</th></tr><tr><td>Joe</td><td>Book302</td></tr><tr><td>Jim</td><td>Laptop77</td></tr><tr><td>Fred</td><td>GgleGlass</td></tr></table>			Name	Object	Joe	Book302	Jim	Laptop77	Fred	GgleGlass	<table><tr><th>Name</th><th>Object</th></tr><tr><td>Joe</td><td>Laptop77</td></tr></table>			Name	Object	Joe	Laptop77	<table><tr><th>Name</th><th>Object</th></tr><tr><td>Joe</td><td>Book302</td></tr><tr><td>Jim</td><td>Laptop77</td></tr><tr><td>Fred</td><td>GgleGlass</td></tr></table>			Name	Object	Joe	Book302	Jim	Laptop77	Fred	GgleGlass									
Name	Object																																																	
Joe	Book302																																																	
Joe	Laptop77																																																	
Jim	Laptop77																																																	
Fred	GgleGlass																																																	
Name	Object																																																	
Joe	Book302																																																	
Jim	Laptop77																																																	
Fred	GgleGlass																																																	
Name	Object																																																	
Joe	Laptop77																																																	
Name	Object																																																	
Joe	Book302																																																	
Jim	Laptop77																																																	
Fred	GgleGlass																																																	
Location			Location			Location			Location																																									
<table><tr><th>Object</th><th>Time</th><th>Loc</th></tr><tr><td>Laptop77</td><td>5:07</td><td>Hall</td></tr><tr><td>Laptop77</td><td>9:05</td><td>Office</td></tr><tr><td>Book302</td><td>8:18</td><td>Office</td></tr></table>			Object	Time	Loc	Laptop77	5:07	Hall	Laptop77	9:05	Office	Book302	8:18	Office	<table><tr><th>Object</th><th>Time</th><th>Loc</th></tr><tr><td>Book302</td><td>8:18</td><td>Office</td></tr></table>			Object	Time	Loc	Book302	8:18	Office	<table><tr><th>Object</th><th>Time</th><th>Loc</th></tr><tr><td>Laptop77</td><td>5:07</td><td>Hall</td></tr><tr><td>Laptop77</td><td>9:05</td><td>Office</td></tr></table>			Object	Time	Loc	Laptop77	5:07	Hall	Laptop77	9:05	Office	<table><tr><th>Object</th><th>Time</th><th>Loc</th></tr><tr><td>Laptop77</td><td>5:07</td><td>Hall</td></tr><tr><td>Laptop77</td><td>9:05</td><td>Office</td></tr><tr><td>Book302</td><td>8:18</td><td>Office</td></tr></table>			Object	Time	Loc	Laptop77	5:07	Hall	Laptop77	9:05	Office	Book302	8:18	Office
Object	Time	Loc																																																
Laptop77	5:07	Hall																																																
Laptop77	9:05	Office																																																
Book302	8:18	Office																																																
Object	Time	Loc																																																
Book302	8:18	Office																																																
Object	Time	Loc																																																
Laptop77	5:07	Hall																																																
Laptop77	9:05	Office																																																
Object	Time	Loc																																																
Laptop77	5:07	Hall																																																
Laptop77	9:05	Office																																																
Book302	8:18	Office																																																
 <table><tr><td>Joe</td></tr><tr><td>Jim</td></tr></table>			Joe	Jim	Q= <table><tr><td>Joe</td></tr></table>			Joe	Q= <table><tr><td>Joe</td></tr></table>			Joe	Q= <table><tr><td>Joe</td></tr><tr><td>Jim</td></tr></table>			Joe	Jim																																	
Joe																																																		
Jim																																																		
Joe																																																		
Joe																																																		
Joe																																																		
Jim																																																		

Diskussion

- Intuition: Eine probabilistische Datenbank sagt aus, dass eine Datenbank in einem von verschiedenen möglichen Zuständen ist. Jeder Zustand hat eine Wahrscheinlichkeit
- **Mögliche Anfrageantworten:** Eine Menge von Antworten, annotiert mit Wahrscheinlichkeiten:

$(t_1, p_1), (t_2, p_2), (t_3, p_3), \dots$

Üblicherweise: $p_1 \geq p_2 \geq p_3 \geq \dots$

- **Problem:** Die Anzahl der möglichen Welten in einer probabilistischen Datenbank ist sehr groß.
- Ziel: Anfragebeantwortung ohne explizite Generierung aller möglichen Welten (eventuell Einschränkungen in der Ausdrucksstärke hinnehmen)

Unabhängige und disjunkte Tupel

Definition Gegeben eine probabilistische DB (W, P) .
Zwei Tupel t_1, t_2 werden heißen:

- **unabhängig**, falls: $P(t_1 t_2) = P(t_1) P(t_2)$
- **disjunkt** (or exklusiv), falls: $P(t_1 t_2) = 0$

Unabhängige und disjunkte Tupel

Definition Gegeben eine probabilistische DB (W, P) .
Zwei Tupel t_1, t_2 werden heißen:

- **unabhängig**, falls: $P(t_1 t_2) = P(t_1) P(t_2)$
- **disjunkt** (or exklusiv), falls: $P(t_1 t_2) = 0$

Definition Eine probabilistische DB heißt *block-unabhängig-disjunkt* (BUD), falls die Tupel in Blöcke gruppiert werden können, so dass:

- Tupel vom gleichen Block **disjunkt** sind
- Tupel von verschiedenen Blöcken **unabhängig** sind.

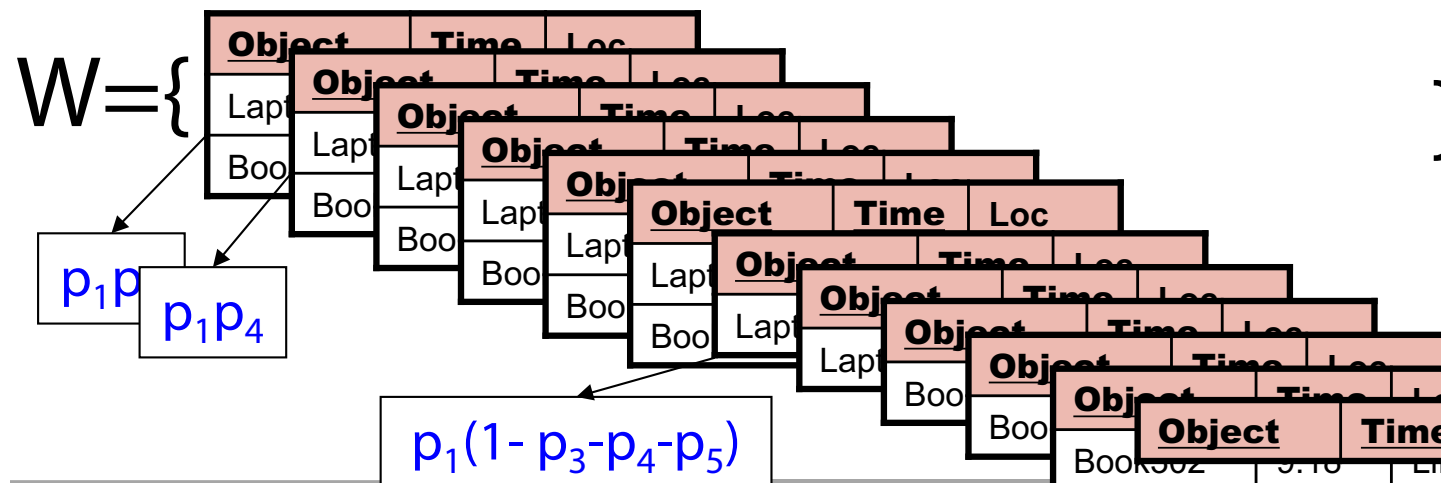
Beispiel: BUD-Tabelle

Object	Time	Loc	P
Laptop77	9:07	Rm444	p_1
Laptop77	9:07	Hall	p_2
Book302	9:18	Office	p_3
Book302	9:18	Rm444	p_4
Book302	9:18	Lift	p_5

BUD Tabelle

disjunkt

disjunkt

 Unab-
hängig

 Mögliche
Welten

Das Anfrage-Evaluationsproblem

Gegeben: BUD-Datenbank D , Anfrage Q , Ausgabetupel t

Berechne: $P(t)$

NB: D habe, sagen wir, 1.000.000 Tupel,
dann ist die Anzahl der möglichen Welten: $2^{1.000.000}$

Herausforderung: Berechne $P(t)$ effizient, in der Größe von D

Datenkomplexität: die Komplexität von P
hängt von D , der Anzahl der Datenelemente, ab.

Ein Beispiel

```
SELECT DISTINCT 'true'
FROM R, S
WHERE R.x = S.x
```

Boolesche Anfrage:
Join-Tupel vorhanden?

$Q() = R(x), S(x,y)$

$$P(Q) = 1 - \{1 - p_1 * [1 - (1 - q_1) * (1 - q_2)]\} * \\ \{1 - p_2 * [1 - (1 - q_3) * (1 - q_4) * (1 - q_5)]\}$$

Man kann $P(Q)$ in PTIME
bzgl. der Größe der DB D bestimmen

R

x	P
a1	p_1
a2	p_2
a3	p_3

S

x	y	P
a1	b1	q_1
a1	b2	q_2
a2	b3	q_3
a2	b4	q_4
a2	b5	q_5

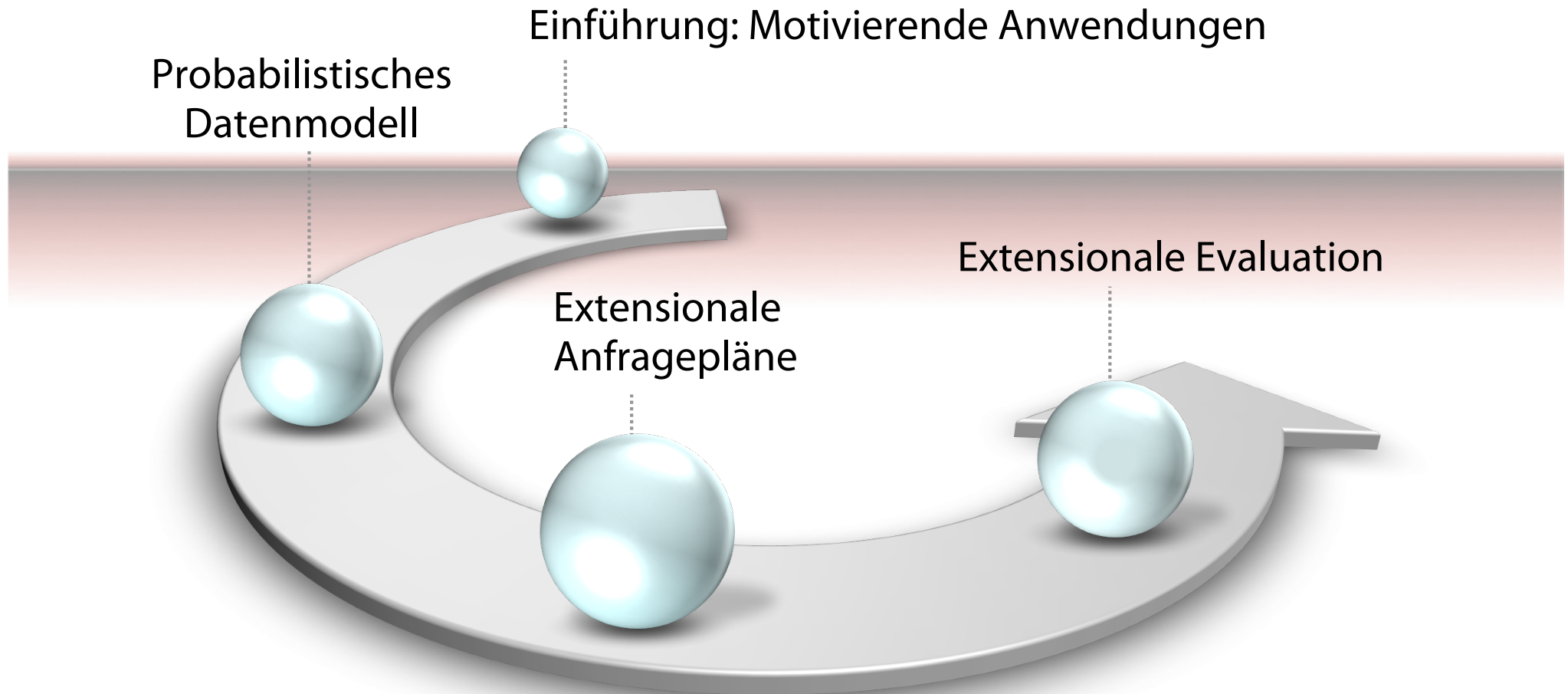
Zusammenfassung: Das probabilistische Datenmodell

- Mögliche-Welten-Semantik:
Mächtig, aber schwierig zu repräsentieren
- **Block-unabhängig-disjunkte** Datenbasen haben
effiziente Repräsentationen:
D wird in traditioneller DB gespeichert
- **Unabhängige Datenbasen**: noch einfacher

Herausforderung: evaluiere Q effizient
bzgl. der Größe von **D**

Non-Standard-Datenbanken

Probabilistische Datenbanken



Relationale Algebra

1. Verbund (join) $\bowtie$

2. Projektion
(mit Duplikat-
Elimination) Π

3. Vereinigung $\cup$

1. Auswahl (selection) σ

2. Differenz: $-$
hier nicht verwendet

Wiederholung: Anfragebearbeitungspläne

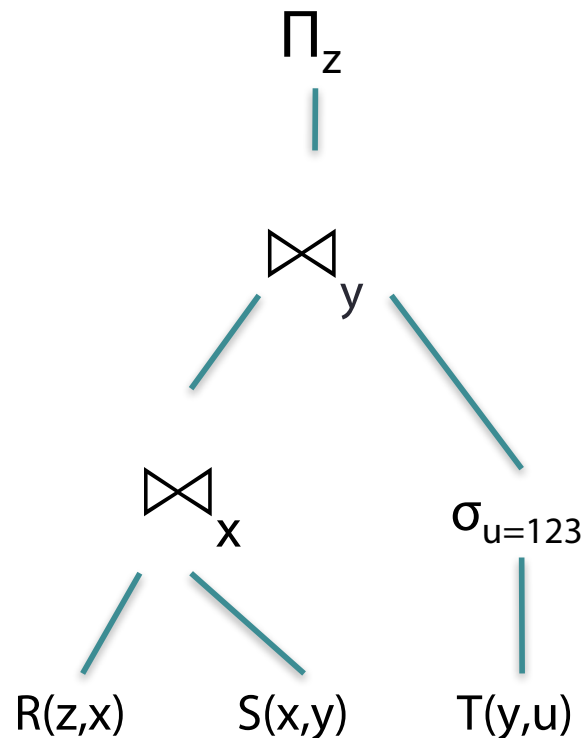
```
SELECT DISTINCT R.z  
FROM R, S, T  
WHERE R.x = S.x  
      and S.y=T.y  
      and T.u = 123
```

$$Q(z) = R(z,x), S(x,y), T(y,u)$$

Wiederholung: Anfragebearbeitungspläne

```
SELECT DISTINCT R.z  
FROM R, S, T  
WHERE R.x = S.x  
and S.y=T.y  
and T.u = 123
```

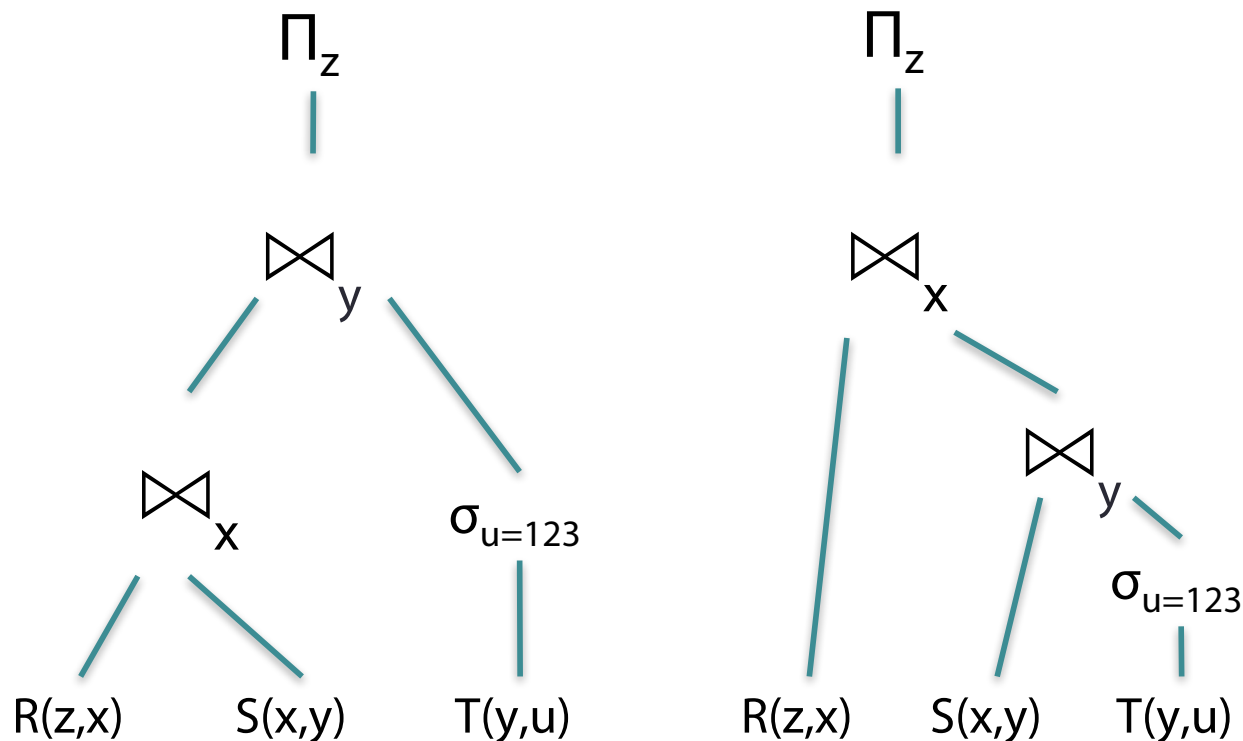
$Q(z) = R(z,x), S(x,y), T(y,u)$



Wiederholung: Anfragebearbeitungspläne

SELECT DISTINCT R.z
FROM R, S, T
WHERE R.x = S.x
and S.y=T.y
and T.u = 123

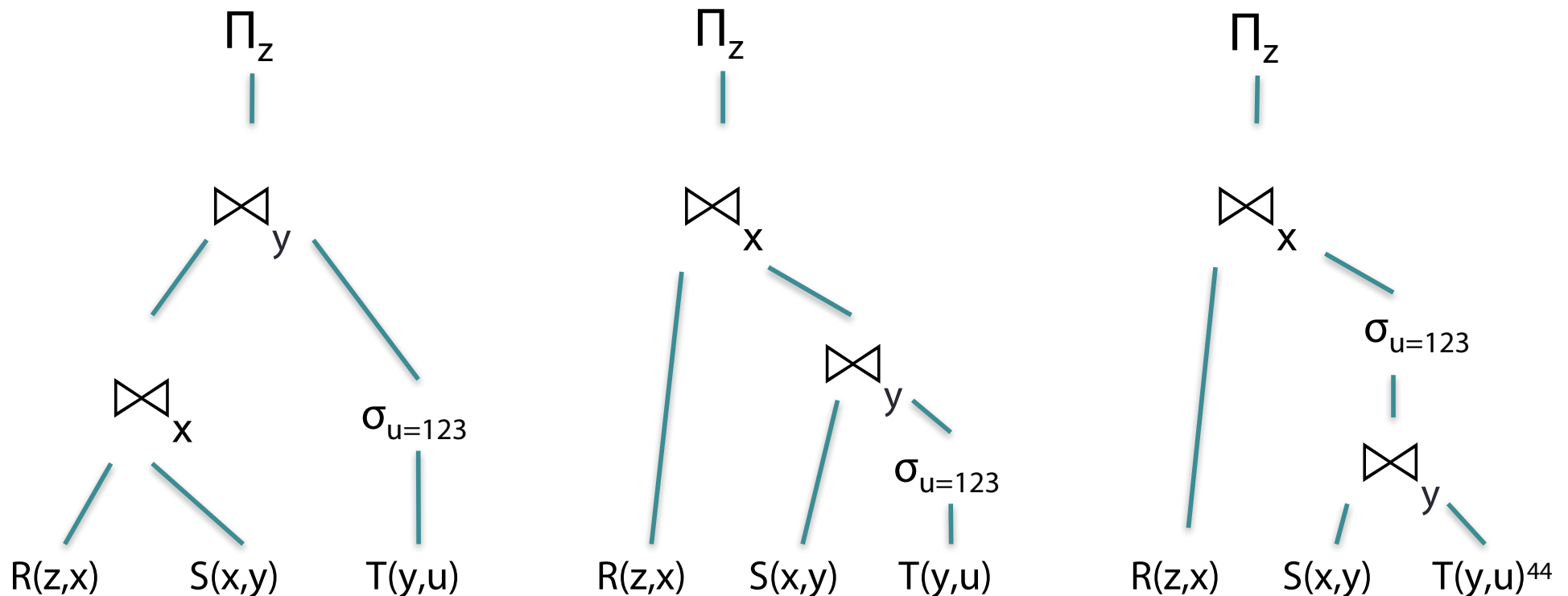
$Q(z) = R(z,x), S(x,y), T(y,u)$



Wiederholung: Anfragebearbeitungspläne

SELECT DISTINCT R.z
FROM R, S, T
WHERE R.x = S.x
and S.y=T.y
and T.u = 123

$Q(z) = R(z,x), S(x,y), T(y,u)$



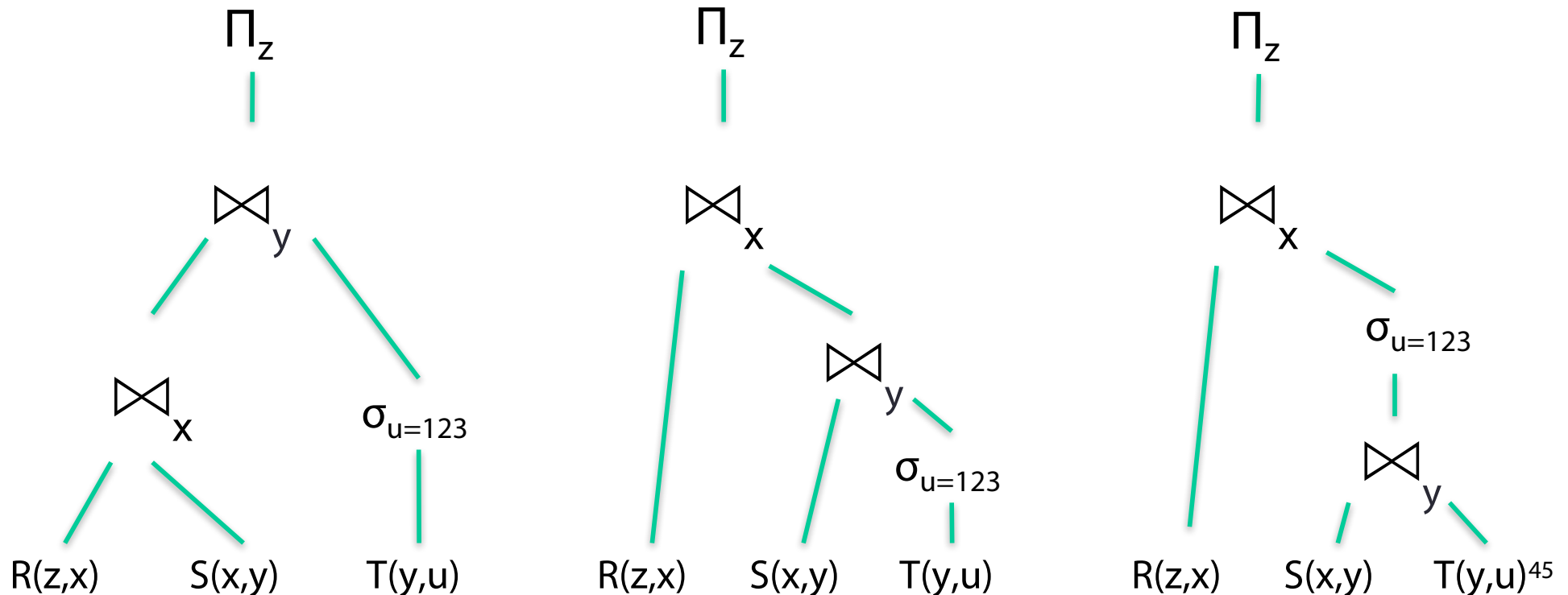
Wiederholung: Anfragebearbeitungspläne

SELECT DISTINCT R.z
FROM R, S, T
WHERE R.x = S.x
and S.y=T.y
and T.u = 123

$Q(z) = R(z,x), S(x,y), T(y,u)$

Diese Pläne sind äquivalent (liefern gleiche Ergebnisse)

Der Anfrageoptimierer wählt den Plan mit den geschätzt geringsten Kosten



Extensionale Pläne

- Kernidee:
 - Modifiziere jeden Operator, sodass Wahrscheinlichkeiten für die Ausgabe berechnet werden
- Annahmen notwendig:
 - Ereignisse sind
 - unabhängig oder
 - disjunkt (exklusive)

Extensionale Operatoren

Independent
join

A	B	P
a1	b1	$p1 \cdot q1$
a1	b2	$p1 \cdot q2$
a2	b3	$p2 \cdot q3$
a2	b4	$p2 \cdot q4$
a2	b5	$p2 \cdot q5$

i



i für
independent

R(A)

A	P
a1	$p1$
a2	$p2$
a3	$p3$

S(A,B)

A	B	P
a1	b1	$q1$
a1	b2	$q2$
a2	b3	$q3$
a2	b4	$q4$
a2	b5	$q5$

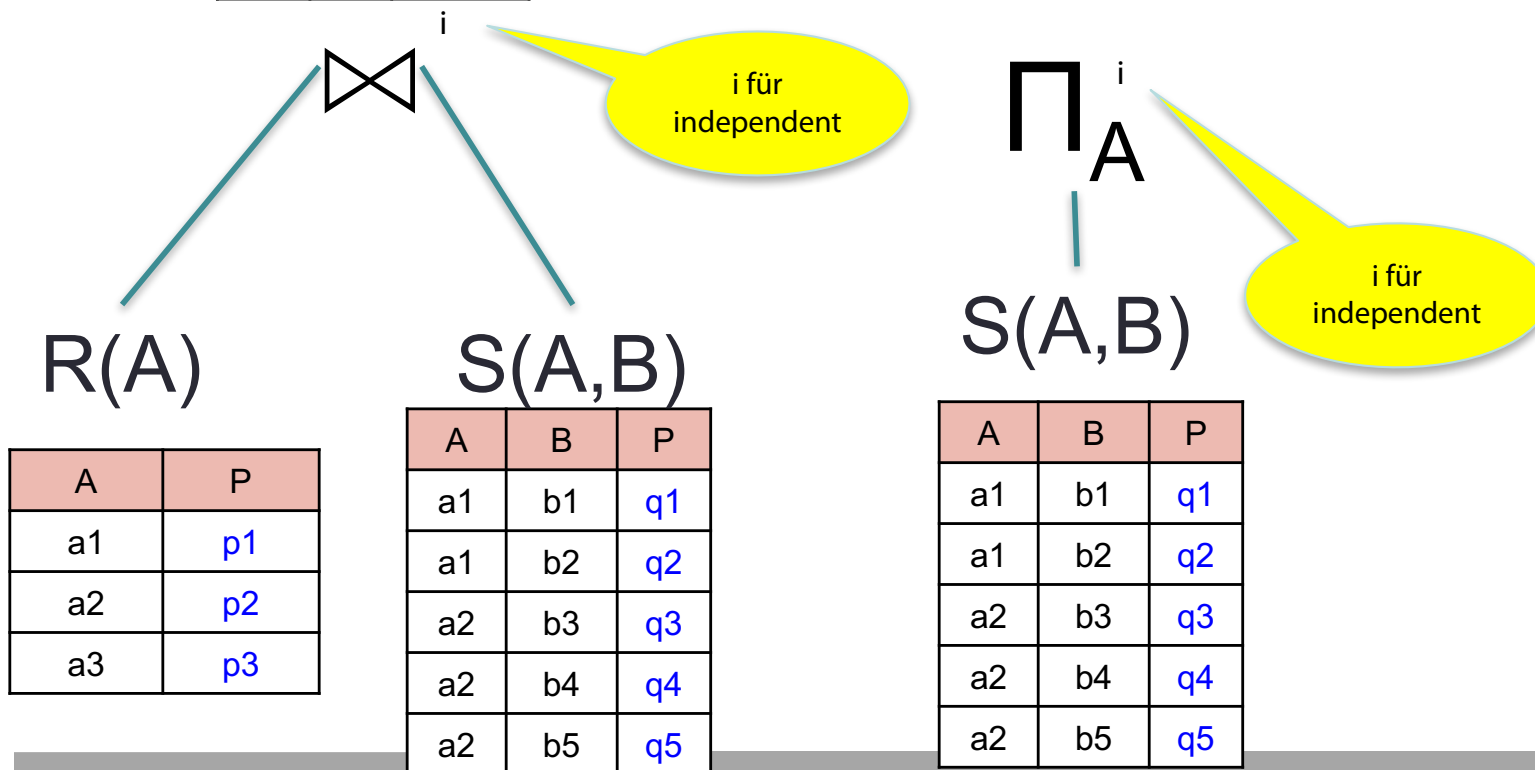
Extensionale Operatoren

Independent
join

A	B	P
a1	b1	$p1 \cdot q1$
a1	b2	$p1 \cdot q2$
a2	b3	$p2 \cdot q3$
a2	b4	$p2 \cdot q4$
a2	b5	$p2 \cdot q5$

Independent
project

A	P
a1	$1 - (1-q1) \cdot (1-q2)$
a2	$1 - (1-q3) \cdot (1-q4) \cdot (1-q5)$



Extensionale Operatoren

Independent
join

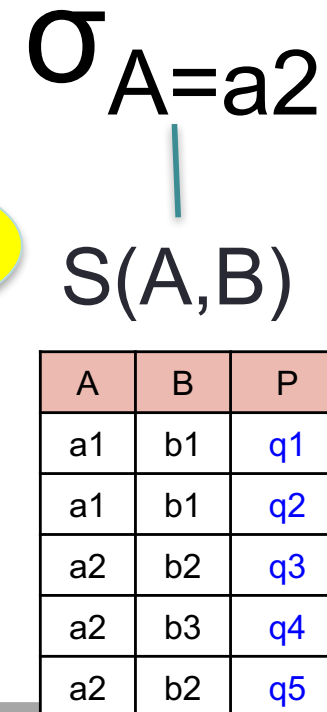
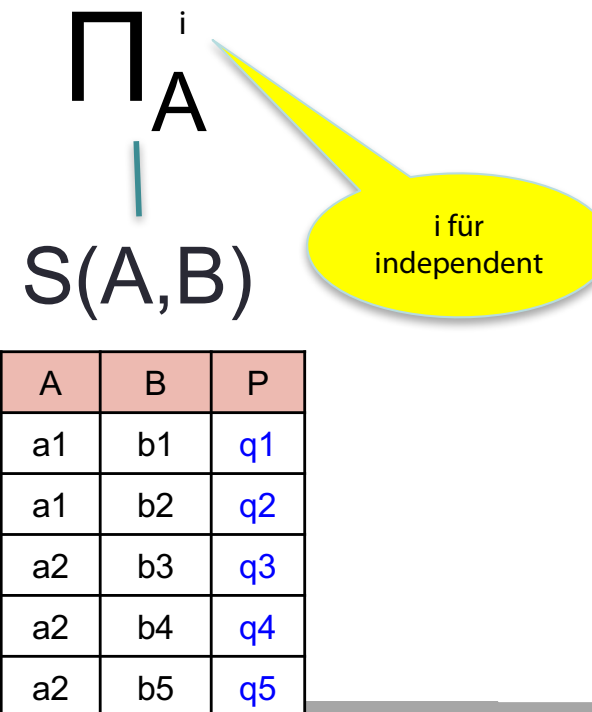
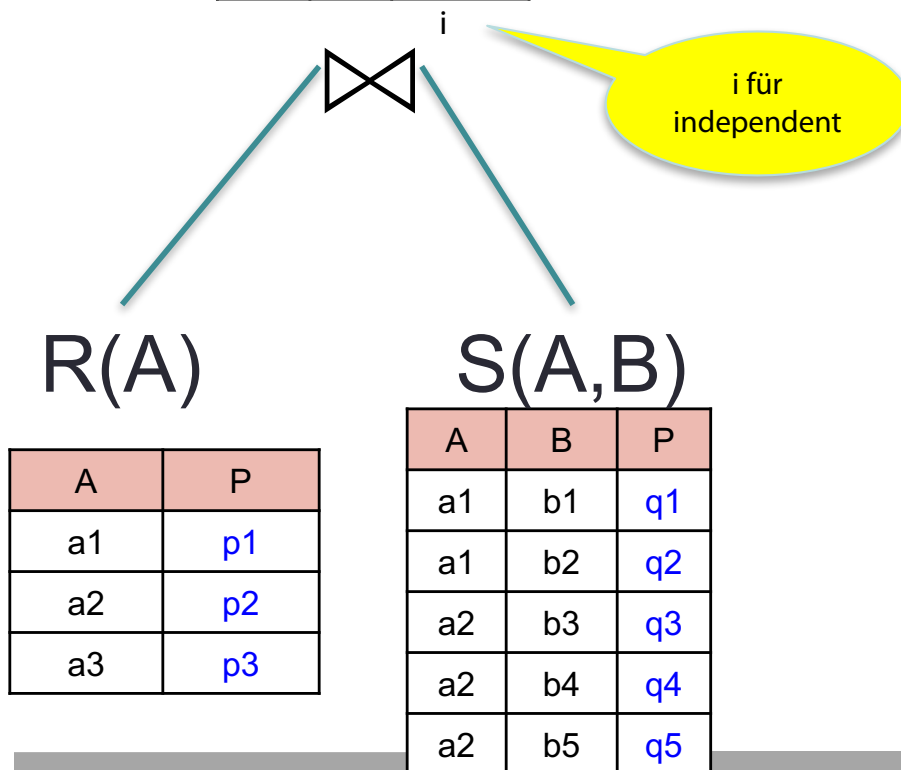
A	B	P
a1	b1	$p1 \cdot q1$
a1	b2	$p1 \cdot q2$
a2	b3	$p2 \cdot q3$
a2	b4	$p2 \cdot q4$
a2	b5	$p2 \cdot q5$

Independent
project

A	P
a1	$1 - (1-q1) \cdot (1-q2)$
a2	$1 - (1-q3) \cdot (1-q4) \cdot (1-q5)$

Selection

A	B	P
a2	b2	$q3$
a2	b3	$q4$
a2	b2	$q5$

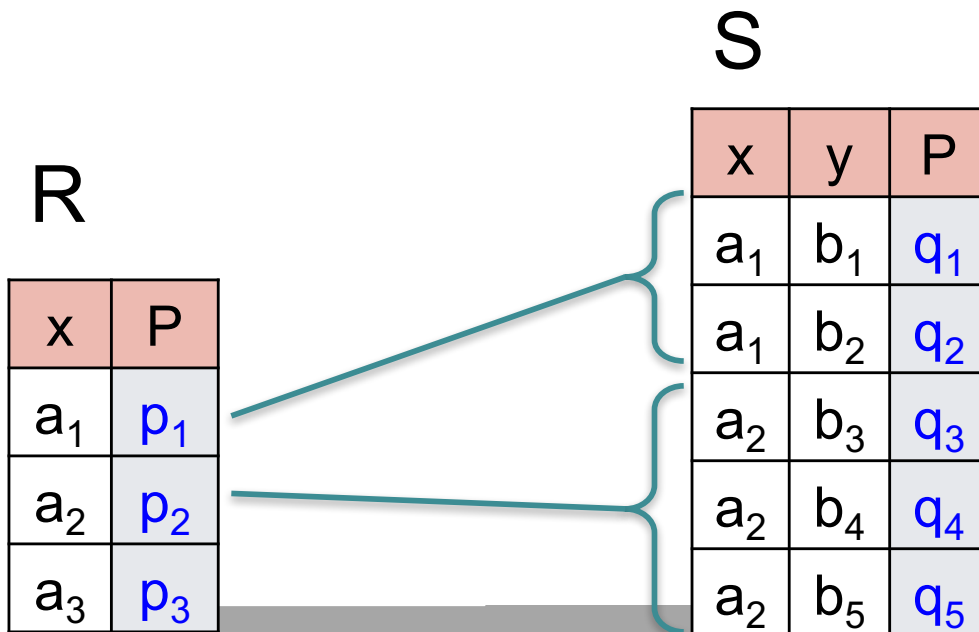


```
SELECT DISTINCT 'true'
FROM R, S
WHERE R.x = S.x
```

 $Q() = R(x), S(x,y)$

$$P(Q) = 1 - [1 - p_1 * (1 - (1 - q_1) * (1 - q_2))] \\ * [1 - p_2 * (1 - (1 - q_3) * (1 - q_4) * (1 - q_5))]$$

Beispiel



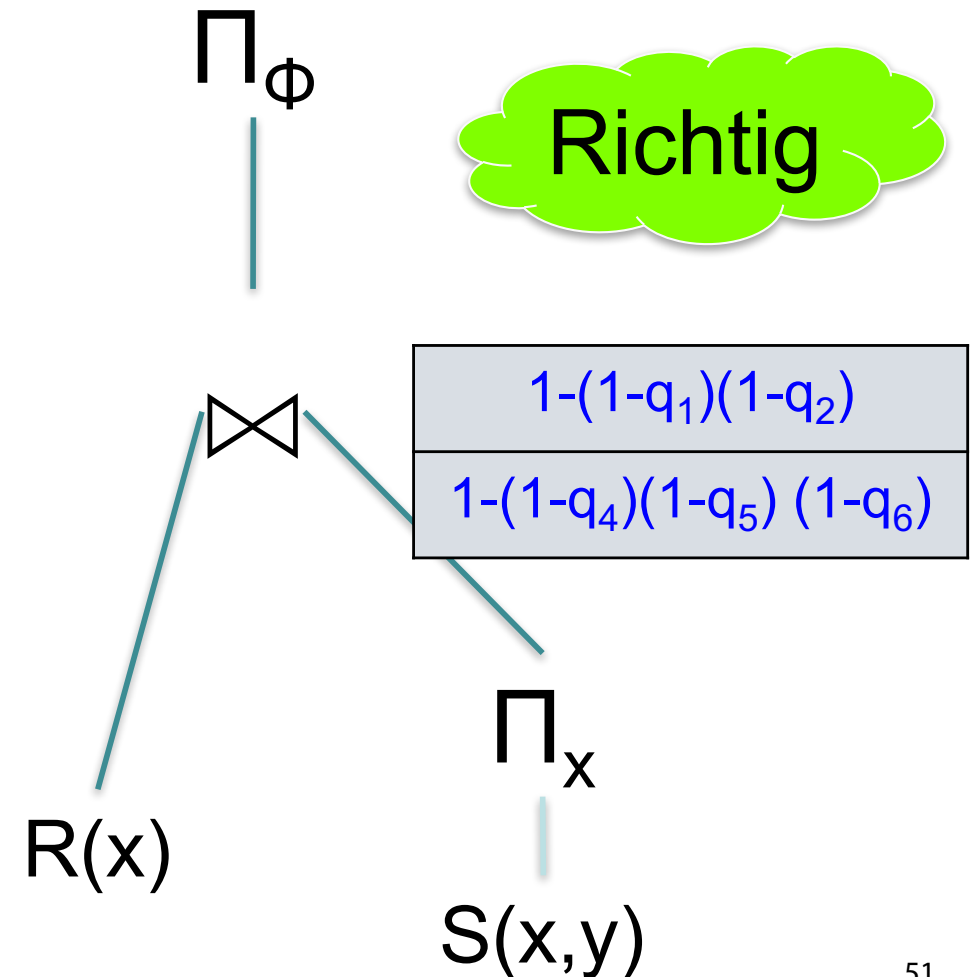
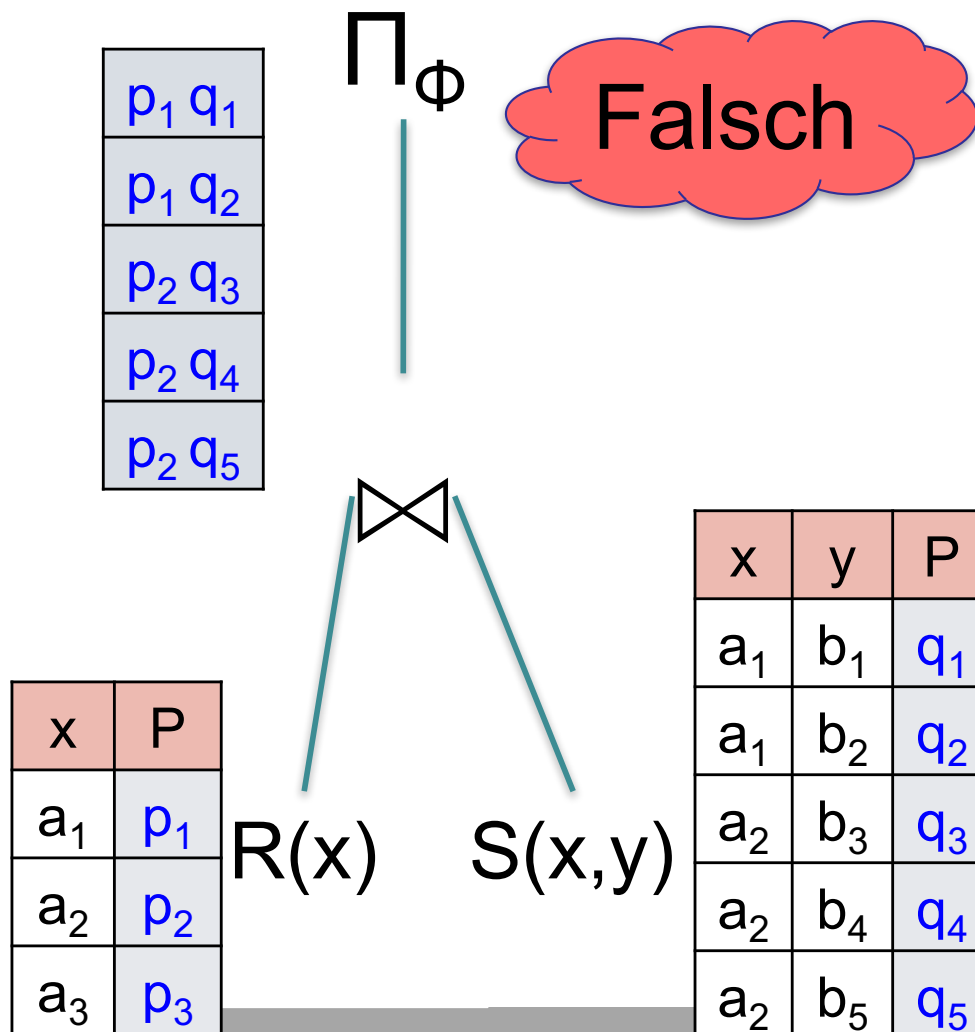
SELECT DISTINCT 'true'
FROM R, S
WHERE R.x = S.x

$Q() = R(x), S(x,y)$

$$P(Q) = 1 - [1 - p_1 * (1 - (1 - q_1) * (1 - q_2))] * [1 - p_2 * (1 - (1 - q_3) * (1 - q_4) * (1 - q_5))]$$

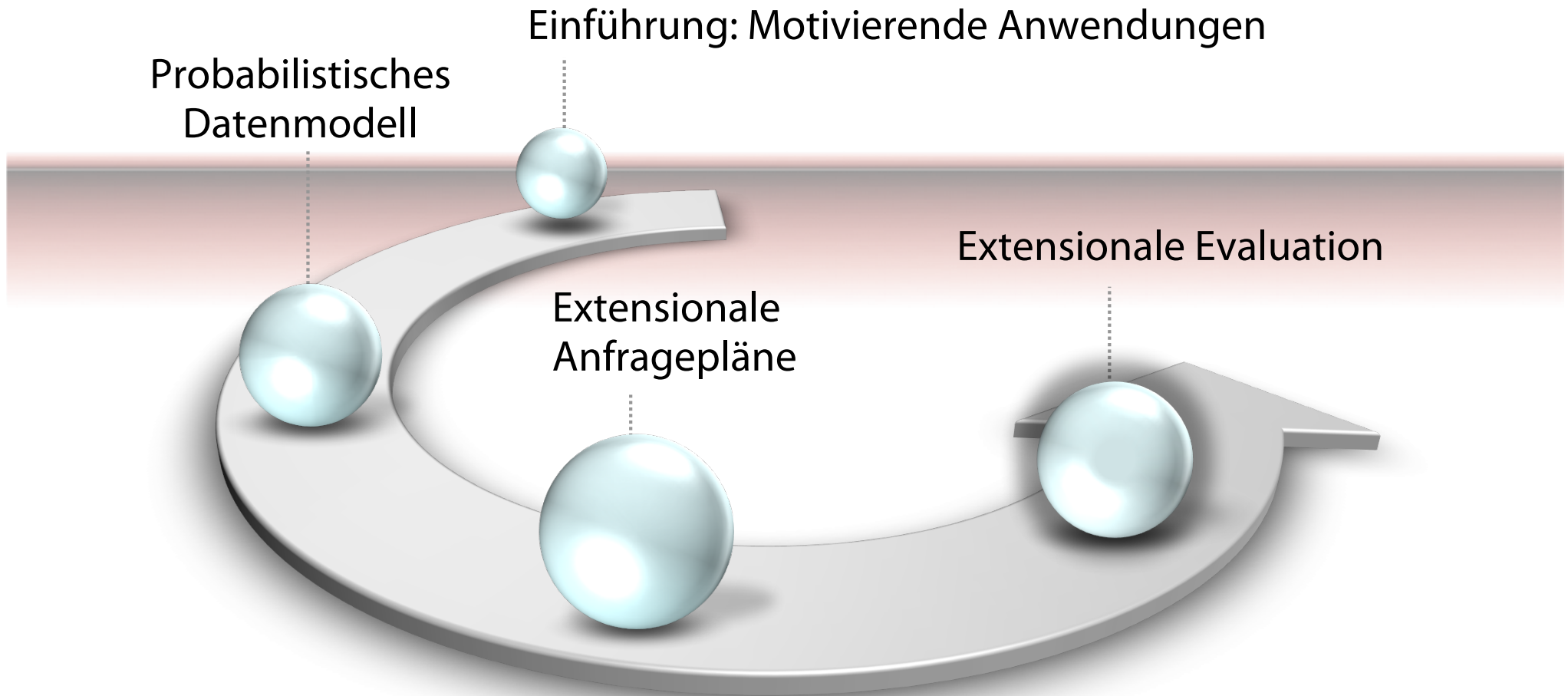
$$1 - (1 - p_1 q_1)(1 - p_1 q_2)(1 - p_2 q_3)(1 - p_2 q_4)(1 - p_2 q_5)$$

$$1 - \{1 - p_1 [1 - (1 - q_1)(1 - q_2)]\} * \{1 - p_2 [1 - (1 - q_3)(1 - q_4)(1 - q_5)]\}$$



Non-Standard-Datenbanken

Probabilistische Datenbanken



Extensionale Operatoren

Independent
join

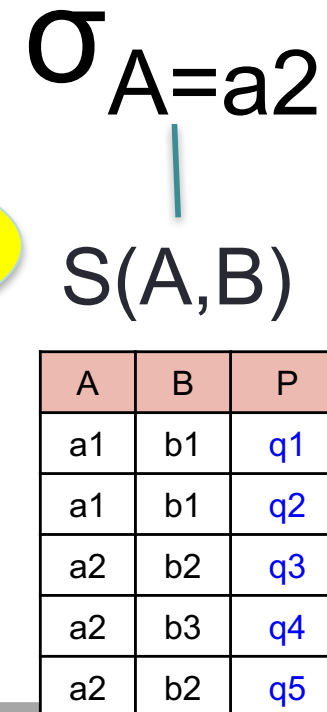
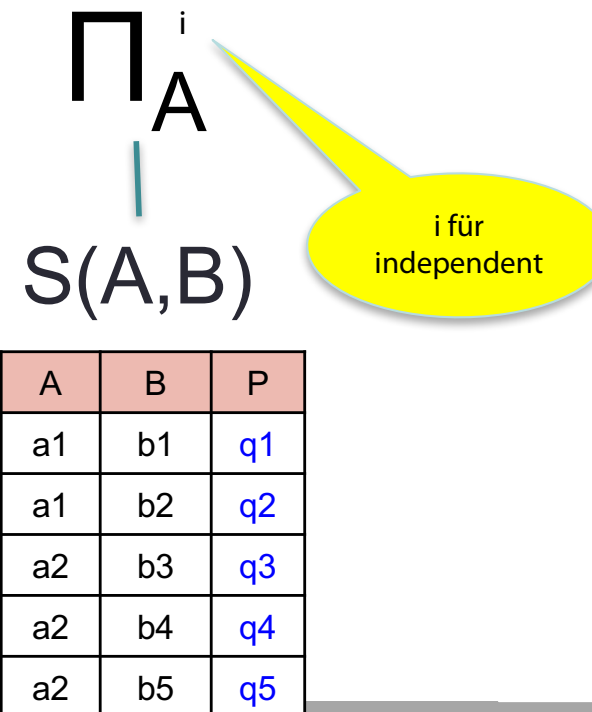
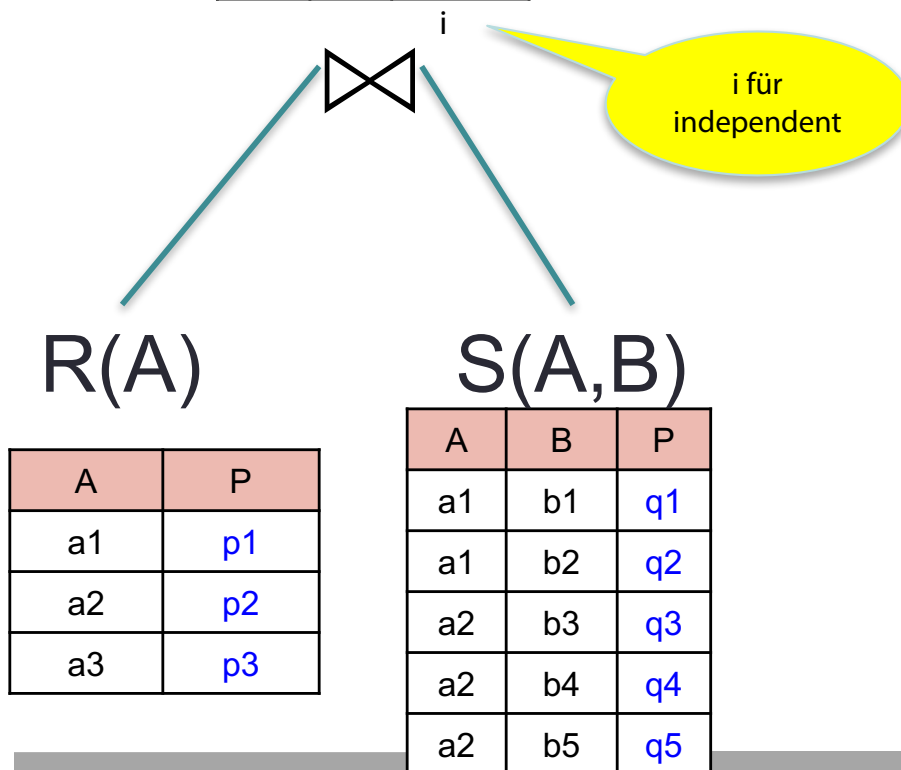
A	B	P
a1	b1	$p1 \cdot q1$
a1	b2	$p1 \cdot q2$
a2	b3	$p2 \cdot q3$
a2	b4	$p2 \cdot q4$
a2	b5	$p2 \cdot q5$

Independent
project

A	P
a1	$1 - (1-q1) \cdot (1-q2)$
a2	$1 - (1-q3) \cdot (1-q4) \cdot (1-q5)$

Selection

A	B	P
a2	b2	$q3$
a2	b3	$q4$
a2	b2	$q5$



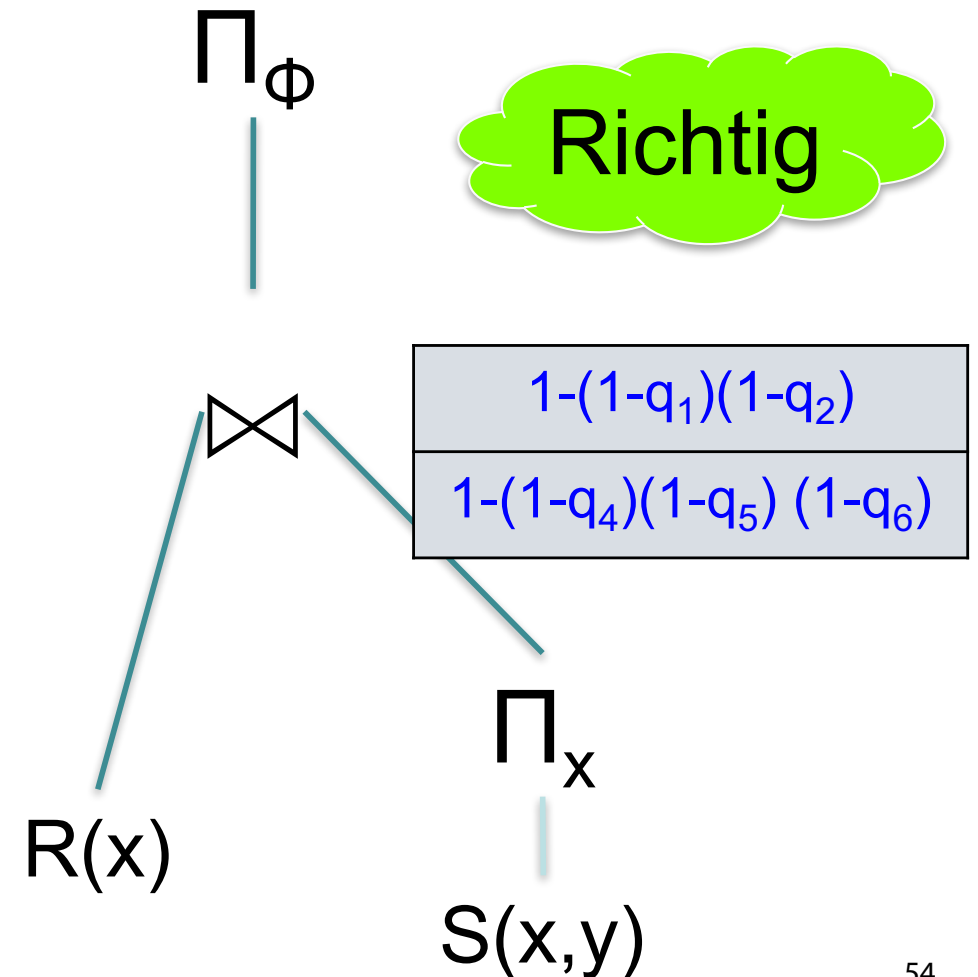
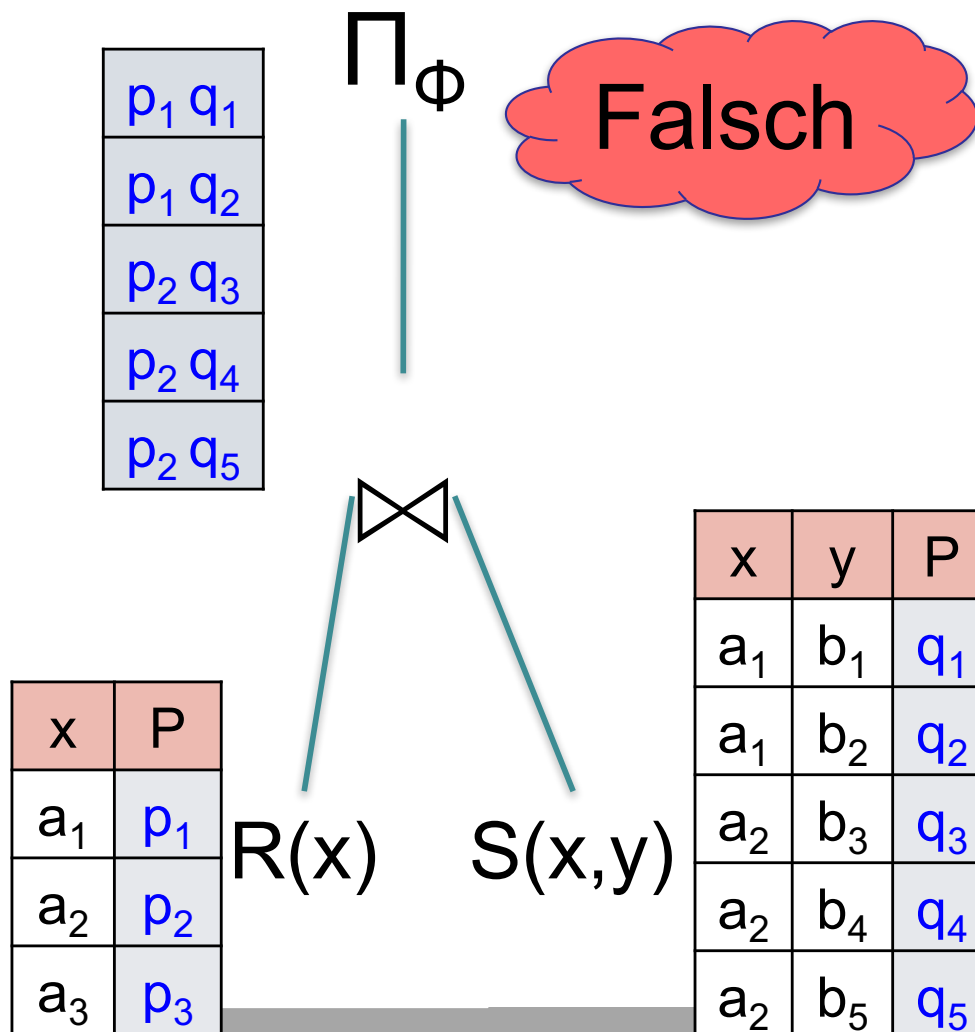
SELECT DISTINCT 'true'
FROM R, S
WHERE R.x = S.x

$Q() = R(x), S(x,y)$

$$P(Q) = 1 - [1 - p_1 * (1 - (1 - q_1) * (1 - q_2))] * [1 - p_2 * (1 - (1 - q_3) * (1 - q_4) * (1 - q_5))]$$

$$1 - (1 - p_1 q_1)(1 - p_1 q_2)(1 - p_2 q_3)(1 - p_2 q_4)(1 - p_2 q_5)$$

$$1 - \{1 - p_1 [1 - (1 - q_1)(1 - q_2)]\} * \{1 - p_2 [1 - (1 - q_3)(1 - q_4)(1 - q_5)]\}$$



Sichere Pläne

- Sei ein Schema für eine probabilist. DB gegeben
 - Relationen tupel-unabhängig oder BUD bei gegebenem Schlüssel

Definition: Ein Plan heißt *sicher*, wenn er die Wahrscheinlichkeiten für die Ausgabe richtig berechnet

- Anfrageoptimierung: Finde kostengünstigen aber sicheren Plan
- Gibt es für jede Anfrage einen sicheren Plan?

Unsichere Anfragen

R

X	P
x1	p1
x2	p2

S

X	Y
x1	y1
x1	y2
x2	y2

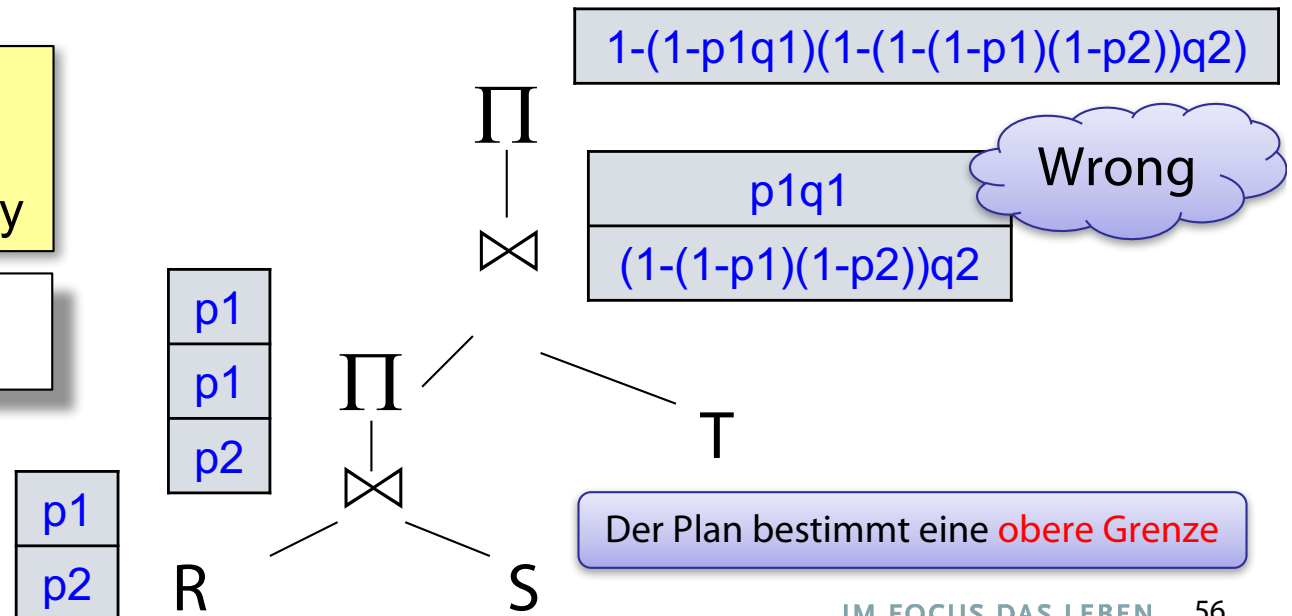
T

Y	P
y1	q1
y2	q2

SELECT DISTINCT 'yes'
FROM R, S, T
WHERE R.x = S.x and S.y = T.y

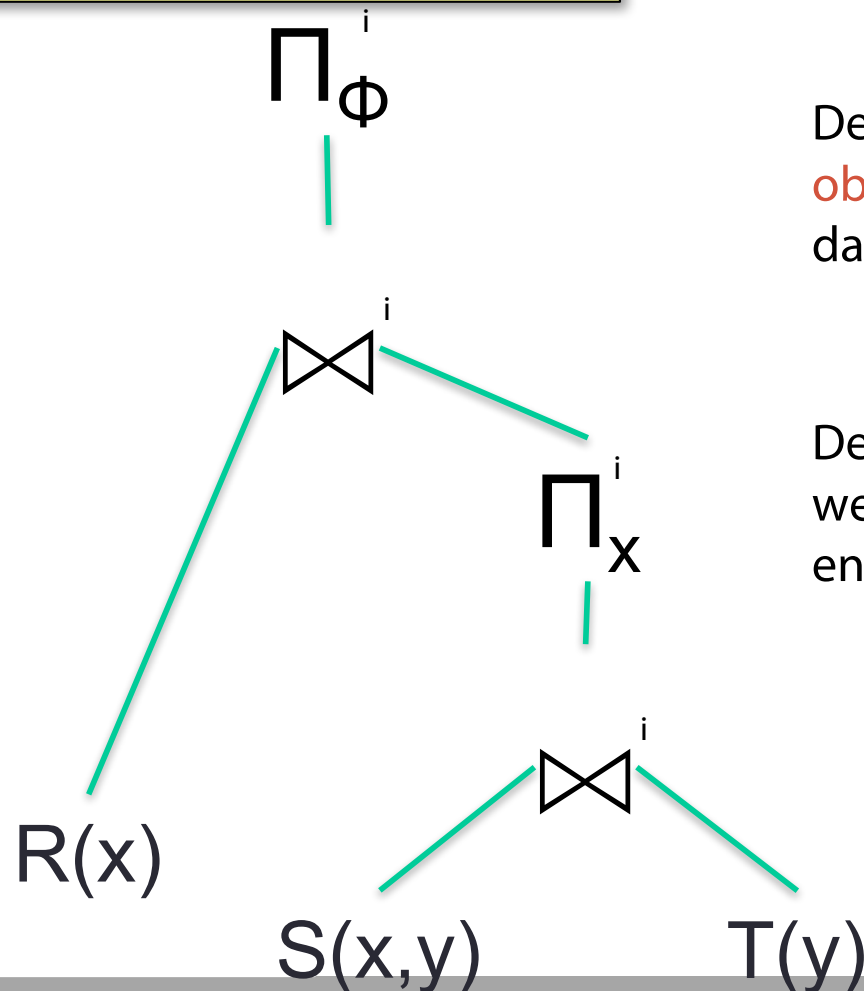
$H_0 :- R(x), S(x,y), T(y)$

W Gatterbauer, D Suciu, Oblivious bounds on
the probability of Boolean functions, ACM
Transactions on Database Systems (TODS) 39
(1), 5, 2013



Extensionale Pläne in PostgreSQL

```
SELECT DISTINCT 'yes'
FROM R, S, T
WHERE R.x = S.x and S.y = T.y
```



Diese Anfrage ist **unsicher**.

Der Plan ist **unsicher**, aber er liefert eine **obere Grenze** für den Wahrscheinlichkeitswert, dass die Anfrage mit 'yes' beantwortet wird.

Der Plan generiert **untere Grenze**, wenn die Wahrscheinlichkeiten in T entsprechend angepasst werden

W Gatterbauer, D Suciu, Oblivious bounds on the probability of Boolean functions, ACM Transactions on Database Systems (TODS) 39 (1), 5, **2013**

Untere Grenzen

S

X	Y
x1	y1
x1	y2
x2	y2

$\bowtie_i$

T

Y	P
y1	q1
y2	q2

T'

```
SELECT T.y, 1-exp((ln(1-T.p))/count(*)) AS p
FROM S,T
WHERE S.y=T.y
GROUP by T.y, T.p
```

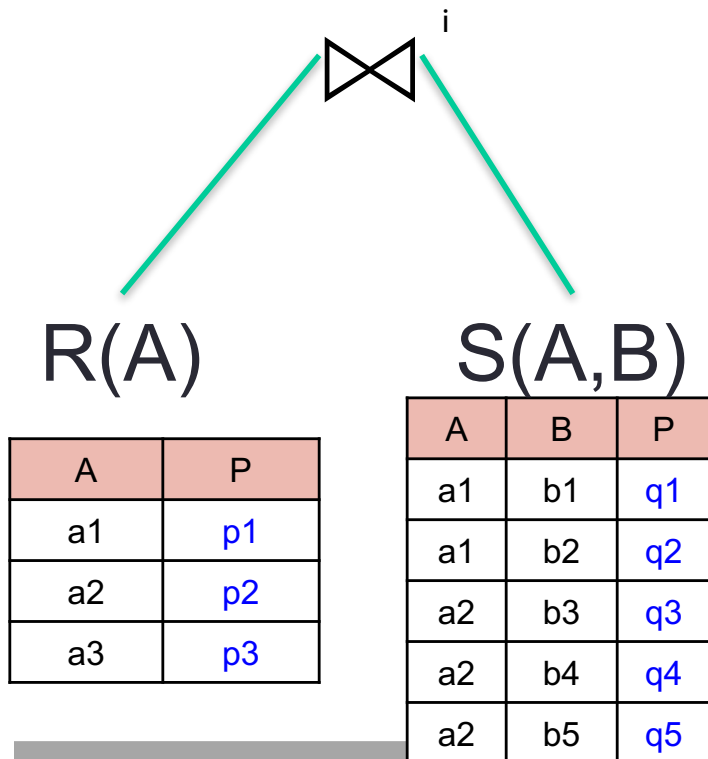
Diskussion

- Sichere Anfragen haben einen sicheren Plan und können effektiv berechnet werden
- Für unsichere Anfragen kann kein sicherer Plan bestimmt werden, und es kann gezeigt werden, dass sie nicht effizient berechnet werden können
- Jeder extensionale Plan (sicher oder unsicher) kann direkt in SQL ausgedrückt werden – gezeigt am Beispiel von PostgreSQL

Extensionale Pläne in PostgreSQL

A	B	P
a1	b1	$p1 \cdot q1$
a1	b2	$p1 \cdot q2$
a2	b3	$p2 \cdot q3$
a2	b4	$p2 \cdot q4$
a2	b5	$p2 \cdot q5$

```
SELECT R.A, S.B, R.P*S.P
FROM   R, S
WHERE  R.A=S.A
```



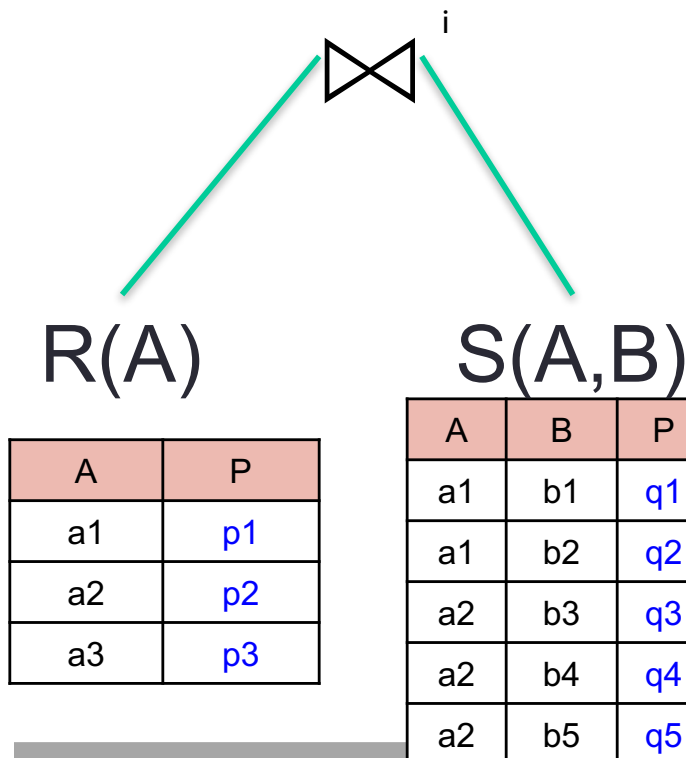
Extensionale Pläne in PostgreSQL

A	B	P
a1	b1	$p1 \cdot q1$
a1	b2	$p1 \cdot q2$
a2	b3	$p2 \cdot q3$
a2	b4	$p2 \cdot q4$
a2	b5	$p2 \cdot q5$

```
SELECT R.A, S.B, R.P*S.P
FROM   R, S
WHERE  R.A=S.A
```

```
SELECT S.A, 1.0-prod(1.0 - S.p)
FROM     S
GROUP BY S.A
```

A	P
a1	$1 - (1-q1) \cdot (1-q2)$
a2	$1 - (1-q3) \cdot (1-q4) \cdot (1-q5)$



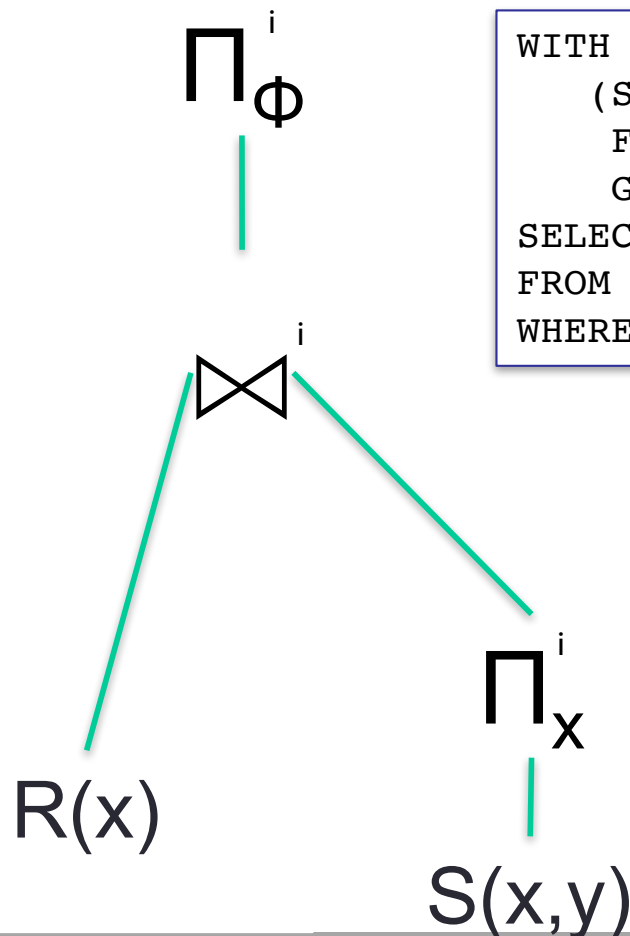
Π_{i A}

A	B	P
a1	b1	$q1$
a1	b2	$q2$
a2	b3	$q3$
a2	b4	$q4$
a2	b5	$q5$

```
create or replace
function combine_prod(float, float)
returns float as
'select $1 * $2' language SQL;
create or replace
function final_prod(float)
returns float as
'select $1' language SQL;
drop aggregate if exists prod (float);
create aggregate prod(float)
( sfunc = combine_prod,
  stype = float,
  finalfunc = final_prod,
  initcond = '1.0'
);
```

Extensionale Pläne in PostgreSQL

```
SELECT DISTINCT 'true'
FROM R, S
WHERE R.x = S.x
```



```
WITH Temp AS
  (SELECT S.x, 1.0-prod(1.0 - S.p) as p
   FROM S
   GROUP BY S.x)
SELECT 'true' as z, 1.0-prod(1.0 - R.P * Temp.P) as p
FROM R, Temp
WHERE R.x = Temp.x
```

Eingaben für PostgreSQL:

```

create or replace function combine_prod (float, float) returns float as 'select $1 * $2' language SQL;
create or replace function final_prod (float) returns float as 'select $1' language SQL;
drop aggregate if exists prod (float);
create aggregate prod (float)
(
  sfunc = combine_prod,
  stype = float,
  finalfunc = final_prod,
  initcond = '1.0'
);

create table R(z char(8), x char(8), p float);
create table S(x char(8), y char(8), p float);

insert into R values('c', 'a1', 0.5);
insert into R values('c', 'a2', 0.5);
insert into R values('c', 'a3', 0.5);

insert into S values('a1', 'b1', 0.5);
insert into S values('a1', 'b2', 0.5);
insert into S values('a2', 'b2', 0.5);
insert into S values('a2', 'b3', 0.5);
insert into S values('a2', 'b4', 0.5);

-- computing the query  $Q(z) = R(z,x), S(x,y)$ 
-- a safe plan:
with Temp as
  (select S.x, 1.0-prod(1.0-p) as p
   from S
   group by S.x)
select R.z, 1.0-prod(1-R.p*Temp.p)
from R, Temp
where R.x=Temp.x
group by R.z;

-- an unsafe plan; guaranteed to return an upper bound on the probability
select R.z, 1.0-prod(1-R.p*S.p)
from R, S
where R.x=S.x
group by R.z;

```

Eingaben für PostgreSQL:

```

-----
-- The following approximation plans for unsafe queries are from
-- Gatterbauer, Suciu: Oblivious Bounds on the Probability of Boolean Functions

-- create a third table
create table T(y char(8), p float);

insert into T values('b1', 0.5);
insert into T values('b2', 0.5);
insert into T values('b3', 0.5);
insert into T values('b4', 0.5);

-- computing the query  $Q(z) = R(z,x), S(x,y), T(y)$ 
-- This query has no safe plans

-- Next two unsafe plans compute upper bounds on the probability:
-- Unsafe plan #1
with Temp as
  (select S.x, 1.0-prod(1.0-S.p*T.p) as p
   from S,T
   where S.y=T.y
   group by S.x)
select R.z, 1.0-prod(1-R.p*Temp.p)
from R, Temp
where R.x=Temp.x
group by R.z;

-- Unsafe plan #2
with Temp as
  (select R.z,S.y,1.0-prod(1.0-R.p*S.p) as p
   from R,S
   where R.x=S.x
   group by R.z,S.y)
select Temp.z, 1.0-prod(1-Temp.p*T.p)
from Temp, T
where Temp.y=T.y
group by Temp.z;

```

```

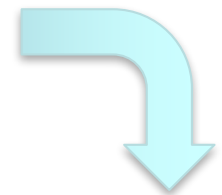
-- Next two unsafe plans compute lower bounds on the probability:
with newT as
  (select T.y, 1-exp((ln(1-T.p))/count(*)) as p
   from S,T
   where S.y=T.y
   group by T.y, T.p),
Temp as
  (select S.x, 1.0-prod(1.0-S.p*newT.p) as p
   from S,newT
   where S.y=newT.y
   group by S.x)
select R.z, 1.0-prod(1-R.p*Temp.p)
from R, Temp
where R.x=Temp.x
group by R.z;

with newR as
  (select R.z, R.x, 1-exp((ln(1-R.p))/count(*)) as p
   from R,S
   where R.x=S.x
   group by R.z,R.x,R.p),
Temp as
  (select newR.z, S.y, 1.0-prod(1.0-newR.p*S.p) as p
   from newR, S
   where newR.x=S.x
   group by newR.z, S.y)
select Temp.z, 1.0-prod(1-Temp.p*T.p)
from Temp, T
where Temp.y=T.y
group by Temp.z;

```

Einsichten 2

- Man benötigt kein neues probabilistisches DB-System für eine probabilistische Datenbasis, wenn man mit Approximationen leben kann
- Wie gut sind die?
- Was man benötigt, sind SQL-Kenntnisse und Kenntnisse in Wahrscheinlichkeitstheorie



Wiederholung: Unions of Conjunctive Queries

Owners of items in either "Office444" or "Hall7":

Atom

$$Q(z) = \exists x_1 \exists t_1 (\text{Owner}(z, x_1) \wedge \text{Location}(x_1, t_1, \text{"Office444"})) \vee \\ \exists x_2 \exists t_2 (\text{Owner}(z, x_2) \wedge \text{Location}(x_2, t_2, \text{"Hall7"}))$$

Ohne Quantoren:

$$Q(z) = \text{Owner}(z, x_1), \text{Location}(x_1, t_1, \text{"Office444"}) \vee \text{Owner}(z, x_2), \text{Location}(x_2, t_2, \text{"Hall7"})$$

Wiederholung: Unions of Conjunctive Queries

Owners of items in either "Office444" or "Hall7":

$$Q(z) = \exists x_1 \exists t_1 (\text{Owner}(z, x_1) \wedge \text{Location}(x_1, t_1, \text{"Office444"})) \vee \\ \exists x_2 \exists t_2 (\text{Owner}(z, x_2) \wedge \text{Location}(x_2, t_2, \text{"Hall7"}))$$

Ohne Quantoren:

Union of conjunctive queries

$$Q(z) = \text{Owner}(z, x_1), \text{Location}(x_1, t_1, \text{"Office444"}) \vee \text{Owner}(z, x_2), \text{Location}(x_2, t_2, \text{"Hall7"})$$

Wiederholung: Unions of Conjunctive Queries

Owners of items in either "Office444" or "Hall7":

$$Q(z) = \exists x_1 \exists t_1 (\text{Owner}(z, x_1) \wedge \text{Location}(x_1, t_1, \text{"Office444"})) \vee \\ \exists x_2 \exists t_2 (\text{Owner}(z, x_2) \wedge \text{Location}(x_2, t_2, \text{"Hall7"}))$$

Ohne Quantoren:

Union of conjunctive queries

$$Q(z) = \text{Owner}(z, x_1), \text{Location}(x_1, t_1, \text{"Office444"}) \vee \text{Owner}(z, x_2), \text{Location}(x_2, t_2, \text{"Hall7"})$$

Nach Umformung:

$$Q(z) = \text{Owner}(z, x) \wedge \exists t [\text{Location}(x, t, \text{"Office444"}) \vee \text{Location}(x, t, \text{"Hall7"})]$$

Wiederholung: Unions of Conjunctive Queries

Owners of items in either "Office444" or "Hall7":

$$Q(z) = \exists x_1 \exists t_1 (\text{Owner}(z, x_1) \wedge \text{Location}(x_1, t_1, \text{"Office444"})) \vee \exists x_2 \exists t_2 (\text{Owner}(z, x_2) \wedge \text{Location}(x_2, t_2, \text{"Hall7"}))$$

Ohne Quantoren:

Union of conjunctive queries

$$Q(z) = \text{Owner}(z, x_1), \text{Location}(x_1, t_1, \text{"Office444"}) \vee \text{Owner}(z, x_2), \text{Location}(x_2, t_2, \text{"Hall7"})$$

Nach Umformung:

$$Q(z) = \text{Owner}(z, x) \wedge \exists t [\text{Location}(x, t, \text{"Office444"}) \vee \text{Location}(x, t, \text{"Hall7"})]$$

Unter Verwendung von:

1. Distributivgesetz für $\vee, \wedge$
2. Gesetz für $\exists, \vee$: $(\exists x P(x)) \vee (\exists y T(y)) = \exists z (P(z) \vee T(z))$

Vier Regeln, um sichere Anfragen zu erzeugen

- Independent join
- Independent project
- Independent union
- Inclusion/exclusion

Wir beschränken uns auf **Boolesche Anfragen**.

Regel 1: Independent Join

$$P(Q1 \wedge Q2) = P(Q1)P(Q2)$$

Wenn Q1 und Q2 unabhängig sind
(also keine gemeinsamen Atome haben)

Regel 1: Independent Join

$$P(Q1 \wedge Q2) = P(Q1)P(Q2)$$

Wenn Q1 und Q2 unabhängig sind
(also keine gemeinsamen Atome haben)

Regel 2: Independent Project

$$P(\exists z Q) = 1 - \prod_{a \in \text{Domain}} (1 - P(Q[a/z]))$$

Wenn z eine "Separatorvariable" in Q
ist, also für Konstanten a,b, Q[a/z]
und Q[b/z] unabhängig sind

Regel 1: Independent Join

$$P(Q1 \wedge Q2) = P(Q1)P(Q2)$$

Wenn Q1 und Q2 unabhängig sind
(also keine gemeinsamen Atome haben)

Regel 2: Independent Project

$$P(\exists z Q) = 1 - \prod_{a \in \text{Domain}} (1 - P(Q[a/z]))$$

Wenn z eine "Separatorvariable" in Q
ist, also für Konstanten a,b, Q[a/z]
und Q[b/z] unabhängig sind

Regel 3: Independent Union

$$P(Q1 \vee Q2) = 1 - (1 - P(Q1))(1 - P(Q2))$$

Wenn Q1 und Q2 unabhängig sind
(also keine gemeinsamen Atome haben)

Beispiel

$$Q_U = R(x_1), S(x_1, y_1) \vee T(x_2), S(x_2, y_2)$$

$$= \exists x_1 \exists y_1 R(x_1) \wedge S(x_1, y_1) \vee \exists x_2 \exists y_2 T(x_2) \wedge S(x_2, y_2)$$

Beispiel

$$Q_U = R(x_1), S(x_1, y_1) \vee T(x_2), S(x_2, y_2)$$

$$= \exists x_1 \exists y_1 R(x_1) \wedge S(x_1, y_1) \vee \exists x_2 \exists y_2 T(x_2) \wedge S(x_2, y_2)$$

$$Q_U = \exists z [R(z) \wedge S(z, y_1) \vee T(z) \wedge S(z, y_2)]$$

Kommutiere $\exists$ mit $\vee$

Beispiel

$$Q_U = R(x_1), S(x_1, y_1) \vee T(x_2), S(x_2, y_2)$$

$$= \exists x_1 \exists y_1 R(x_1) \wedge S(x_1, y_1) \vee \exists x_2 \exists y_2 T(x_2) \wedge S(x_2, y_2)$$

$$Q_U = \exists z [R(z) \wedge S(z, y_1) \vee T(z) \wedge S(z, y_2)]$$

Gesetz für $\exists$ mit $\vee$

$$P(Q_U) = 1 - \prod_{a \in \text{Domain}} (1 - P[R(a) \wedge \underline{S(a, y_1)} \vee T(a) \wedge \underline{S(a, y_2)}])$$

Independent project: Für $a \neq b$, sind $Q_U[a/z]$ und $Q_U[b/z]$ unabhängig weil die Atome $R(a), S(a, y_1), T(a), S(a, y_2)$ disjunkt sind von $R(b), S(b, y_1), T(b), S(b, y_2)$

Beispiel

$$Q_U = R(x_1), S(x_1, y_1) \vee T(x_2), S(x_2, y_2)$$

$$= \exists x_1 \exists y_1 R(x_1) \wedge S(x_1, y_1) \vee \exists x_2 \exists y_2 T(x_2) \wedge S(x_2, y_2)$$

$$Q_U = \exists z [R(z) \wedge S(z, y_1) \vee T(z) \wedge S(z, y_2)]$$

Gesetz für $\exists$ mit $\vee$

$$P(Q_U) = 1 - \prod_{a \in \text{Domain}} (1 - P[R(a) \wedge \underline{S(a, y_1)} \vee T(a) \wedge \underline{S(a, y_2)}])$$

Independent project: Für $a \neq b$, sind $Q_U[a/z]$ und $Q_U[b/z]$ unabhängig weil die Atome $R(a), S(a, y_1), T(a), S(a, y_2)$ disjunkt sind von $R(b), S(b, y_1), T(b), S(b, y_2)$

$$P(Q_U) = 1 - \prod_{a \in \text{Domain}} (1 - P[\underline{(R(a) \vee T(a))} \wedge \underline{\exists y. S(a, y)}])$$

Distribution $\wedge$ über $\vee$

Beispiel

$$Q_U = R(x_1), S(x_1, y_1) \vee T(x_2), S(x_2, y_2)$$

$$= \exists x_1 \exists y_1 R(x_1) \wedge S(x_1, y_1) \vee \exists x_2 \exists y_2 T(x_2) \wedge S(x_2, y_2)$$

$$Q_U = \exists z [R(z) \wedge S(z, y_1) \vee T(z) \wedge S(z, y_2)]$$

Kommutiere $\exists$ mit $\vee$

$$P(Q_U) = 1 - \prod_{a \in \text{Domain}} (1 - P[R(a) \wedge \underline{S(a, y_1)} \vee T(a) \wedge \underline{S(a, y_2)}])$$

Independent project: Für $a \neq b$, sind $Q_U[a/z]$ und $Q_U[b/z]$ unabhängig weil die Atome $R(a), S(a, y_1), T(a), S(a, y_2)$ disjunkt sind von $R(b), S(b, y_1), T(b), S(b, y_2)$

$$P(Q_U) = 1 - \prod_{a \in \text{Domain}} (1 - P[\underline{(R(a) \vee T(a))} \wedge \underline{\exists y. S(a, y)}])$$

Distribution $\wedge$ über $\vee$

$$P(Q_U) = 1 - \prod_{a \in \text{Domain}} (1 - P[R(a) \vee T(a)] P[\exists y. S(a, y)])$$

Independent join

Beispiel

$$Q_U = R(x_1), S(x_1, y_1) \vee T(x_2), S(x_2, y_2)$$

$$= \exists x_1 \exists y_1 R(x_1) \wedge S(x_1, y_1) \vee \exists x_2 \exists y_2 T(x_2) \wedge S(x_2, y_2)$$

$$Q_U = \exists z [R(z) \wedge S(z, y_1) \vee T(z) \wedge S(z, y_2)]$$

Kommutiere $\exists$ mit $\vee$

$$P(Q_U) = 1 - \prod_{a \in \text{Domain}} (1 - P[R(a) \wedge \underline{S(a, y_1)} \vee T(a) \wedge \underline{S(a, y_2)}])$$

Independent project: Für $a \neq b$, sind $Q_U[a/z]$ und $Q_U[b/z]$ unabhängig weil die Atome $R(a), S(a, y_1), T(a), S(a, y_2)$ disjunkt sind von $R(b), S(b, y_1), T(b), S(b, y_2)$

$$P(Q_U) = 1 - \prod_{a \in \text{Domain}} (1 - P[\underline{(R(a) \vee T(a))} \wedge \underline{\exists y. S(a, y)}])$$

Distribution $\wedge$ über $\vee$

$$P(Q_U) = 1 - \prod_{a \in \text{Domain}} (1 - P[R(a) \vee T(a)] P[\exists y. S(a, y)])$$

Independent join

$$P(Q_U) = 1 - \prod_{a \in \text{Domain}} (1 - (1 - (1 - P[R(a)])(1 - P[T(a)])) (1 - \prod_{b \in \text{Domain}} (1 - P[S(a, b)])))$$

Hierarchische Anfragen

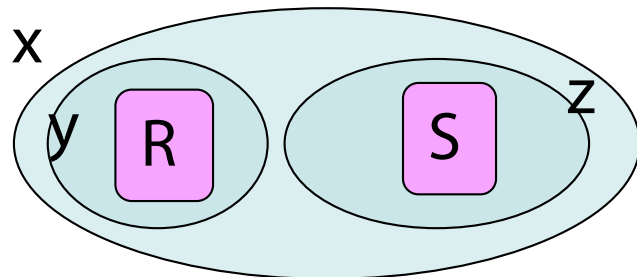
$at(x)$ = Menge von Atomen mit Variable x

Definition Eine Anfrage Q ist **hierarchisch** falls für alle existentiellen Variablen x, y gilt:

$$at(x) \supseteq at(y) \quad \text{oder} \quad at(y) \supseteq at(x) \quad \text{oder} \quad at(x) \cap at(y) = \emptyset$$

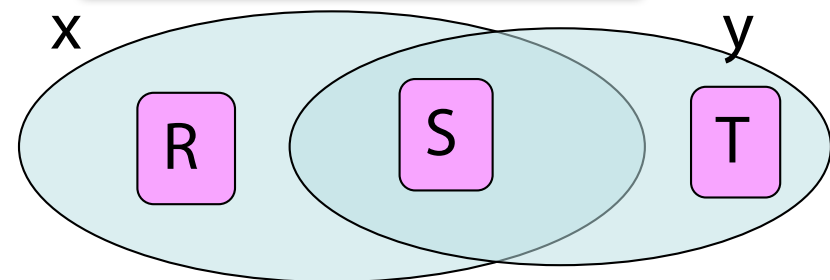
Hierarchisch

$$Q = R(x, y), S(x, z)$$



Nicht-hierarchisch

$$H_0 = R(x), S(x, y), T(y)$$



Regeln funktionieren für hierarchische Anfragen (Unabhängigkeit)

Warum sind nicht-hierarchische Regeln schwierig?

Lifted Inference Sometimes Fails

$$H_0 = \forall x \forall y (\text{Smoker}(x) \vee \text{Friend}(x,y) \vee \text{Jogger}(y))$$

No rule applies here!

The $\forall$ -rule does not apply, because $H_0[\text{Alice}/x]$ and $H_0[\text{Bob}/x]$ are dependent:

$$H_0[\text{Alice}/x] = \forall y (\text{Smoker}(\text{Alice}) \vee \text{Friend}(\text{Alice}, y) \vee \text{Jogger}(y))$$

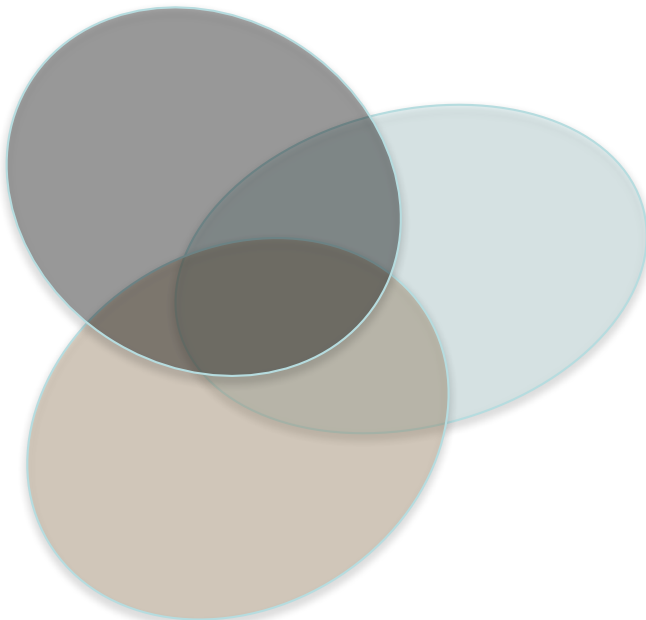
$$H_0[\text{Bob}/x] = \forall y (\text{Smoker}(\text{Bob}) \vee \text{Friend}(\text{Bob}, y) \vee \text{Jogger}(y))$$

Dependent

Theorem. [Dalvi'04] Computing $P(H_0 \mid D)$ is #P-hard in $|D|$

Regel 4: Inclusion-Exclusion

$$\begin{aligned}
 P(Q1 \wedge Q2 \wedge Q3) = & P(Q1) + P(Q2) + P(Q3) \\
 & - P(Q1 \vee Q2) - P(Q1 \vee Q3) - P(Q2 \vee Q3) \\
 & + P(Q1 \vee Q2 \vee Q3)
 \end{aligned}$$



NB: Dieses ist dual zur häufiger verwendeten Formel:

$$\begin{aligned}
 P(Q1 \vee Q2 \vee Q3) = & P(Q1) + P(Q2) + P(Q3) \\
 & - P(Q1 \wedge Q2) - P(Q1 \wedge Q3) - P(Q2 \wedge Q3) \\
 & + P(Q1 \wedge Q2 \wedge Q3)
 \end{aligned}$$

Beispiel

$$Q_j = R(x_1), S(x_1, y_1), T(x_2), S(x_2, y_2)$$

$$= [\exists x_1 \exists y_1 R(x_1) \wedge S(x_1, y_1)] \wedge [\exists x_2 \exists y_2 T(x_2) \wedge S(x_2, y_2)]$$

Beispiel

$$Q_J = R(x_1), S(x_1, y_1), T(x_2), S(x_2, y_2)$$

$$= [\exists x_1 \exists y_1 R(x_1) \wedge S(x_1, y_1)] \wedge [\exists x_2 \exists y_2 T(x_2) \wedge S(x_2, y_2)]$$

$$Q_J = Q_1 \wedge Q_2$$

wobei

$$Q_1 = R(x_1), S(x_1, y_1)$$

$$Q_2 = T(x_2), S(x_2, y_2)$$

Beispiel

$$Q_J = R(x_1), S(x_1, y_1), T(x_2), S(x_2, y_2)$$

$$= [\exists x_1 \exists y_1 R(x_1) \wedge S(x_1, y_1)] \wedge [\exists x_2 \exists y_2 T(x_2) \wedge S(x_2, y_2)]$$

$$Q_J = Q_1 \wedge Q_2$$

wobei

$$Q_1 = R(x_1), S(x_1, y_1)$$

$$Q_2 = T(x_2), S(x_2, y_2)$$

$$P(Q_J) = P(Q_1) + P(Q_2) - P(Q_1 \vee Q_2)$$

Q_1 = eine hierarchische CQ ohne Self-Joins

Q_2 = dito

$Q_1 \vee Q_2 = Q_U$, siehe vorige Folien

Einsicht 3

Vereinigung (union) für Self-Joins!

- Conjunctive Queries = Keine “natürliche” Klassen von Anfragen von Probabilistische DBs
- Unions of Conjunctive Queries = die “natürliche” Klasse von Anfragen

Zusammenfassung

- Falls Anfrageformel mit Regeln transformiert werden kann, dann ist das Ergebnis sicher (→ **Lifted Inference**)
- Regeln sind aber nicht vollständig
 - Suciu et al. stellen noch eine weitere Regel vor
Damit lassen sich alle sicheren Pläne erzeugen
- Aber: Nicht alle Anfragen lassen sich mit den Regeln behandeln (→ nicht immer gibt es einen sicheren Plan)
 - Für bestimmte Anfragen müssen potentiell die Welten betrachtet werden (→ **Grounded Inference**)
 - Frühes probabilistisches DB-System:
 - MayBMS - A Probabilistic Database Management System (<http://maybms.sourceforge.net>)

Neuere Forschung

- Anwendung von Techniken des Weighted Model Counting zur Lösung von #SAT-Problemen
- Umwandlung von Wahrscheinlichkeiten in Gewichten
- Folgende Präsentation sind angelehnt an ein Tutorial:

Lifted Probabilistic Inference in Relational Models

Guy Van den Broeck Dan Suciu
KU Leuven U. of Washington

SAT und #SAT

Gegenben eine KNF-Formel

- SAT: Finde erfüllende Belegung
 - NP-vollständig: SAT, graph coloring, Hamiltonian cycle, ...
- #SAT: Zähle erfüllende Belegungen
 - #P-vollständig: #SAT, ...
- Beispiel: $F = (x \vee y) \wedge (y \vee \neg z)$
 - 5 erfüllende Belegungen
 - $(0,1,0), (0,1,1), (1,1,0), (1,1,1), (1, 0, 0)$

Wiederholung: DPLL(KNF-Formel)

DPLL(F)

if F is empty **then** return 1

if F contains an empty clause **then** return 0

else choose a variable x to branch

return $\text{DPLL}(F|_{x=1}) \vee \text{DPLL}(F|_{x=0})$

#DPLL(F) // computes satisfying probability of F

if F is empty **then** **return** 1

if F contains an empty clause **then** return 0

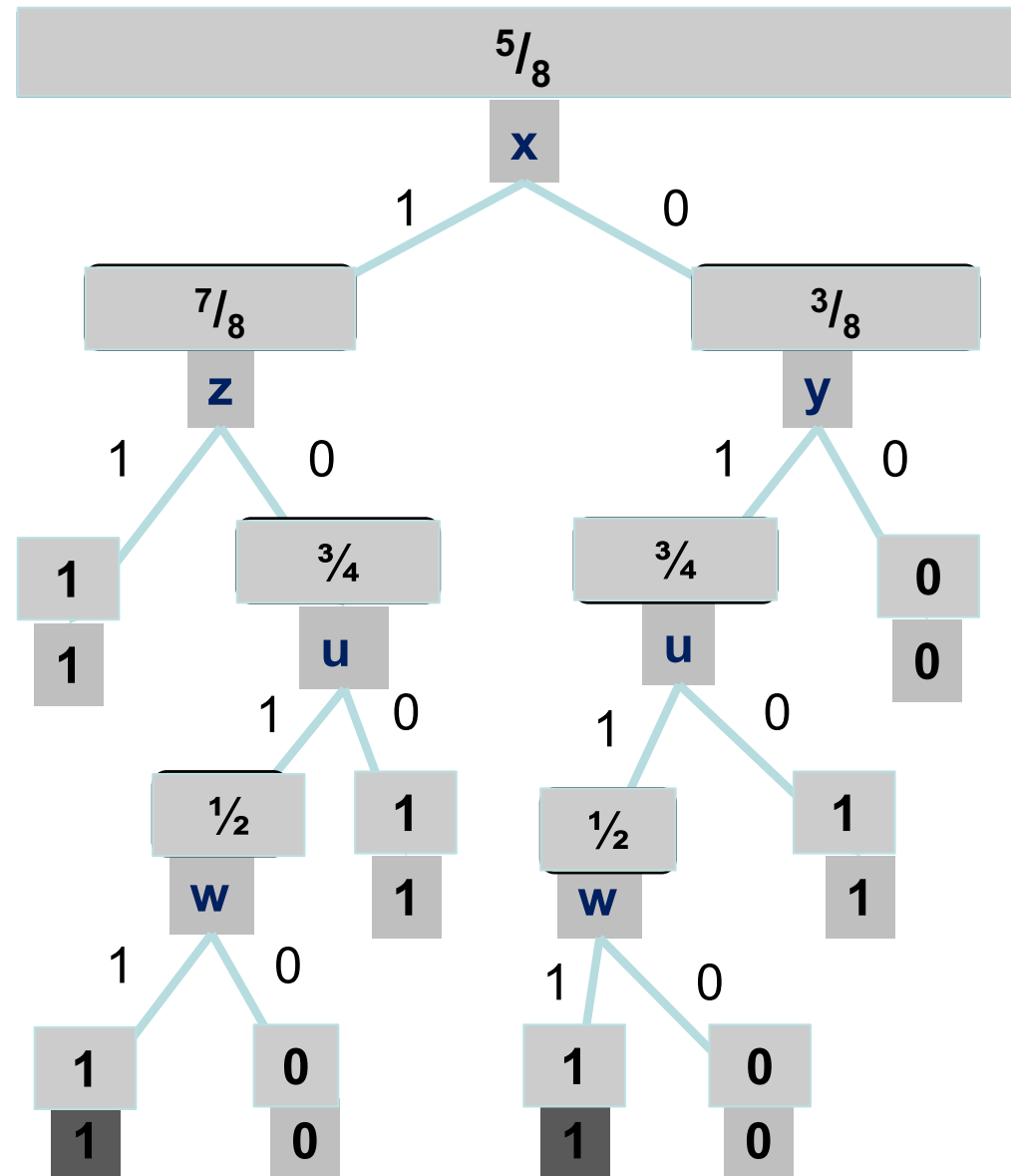
else choose a variable x to branch

return $0.5 * \text{\#DPLL}(F|_{x=1}) + 0.5 * \text{\#DPLL}(F|_{x=0})$

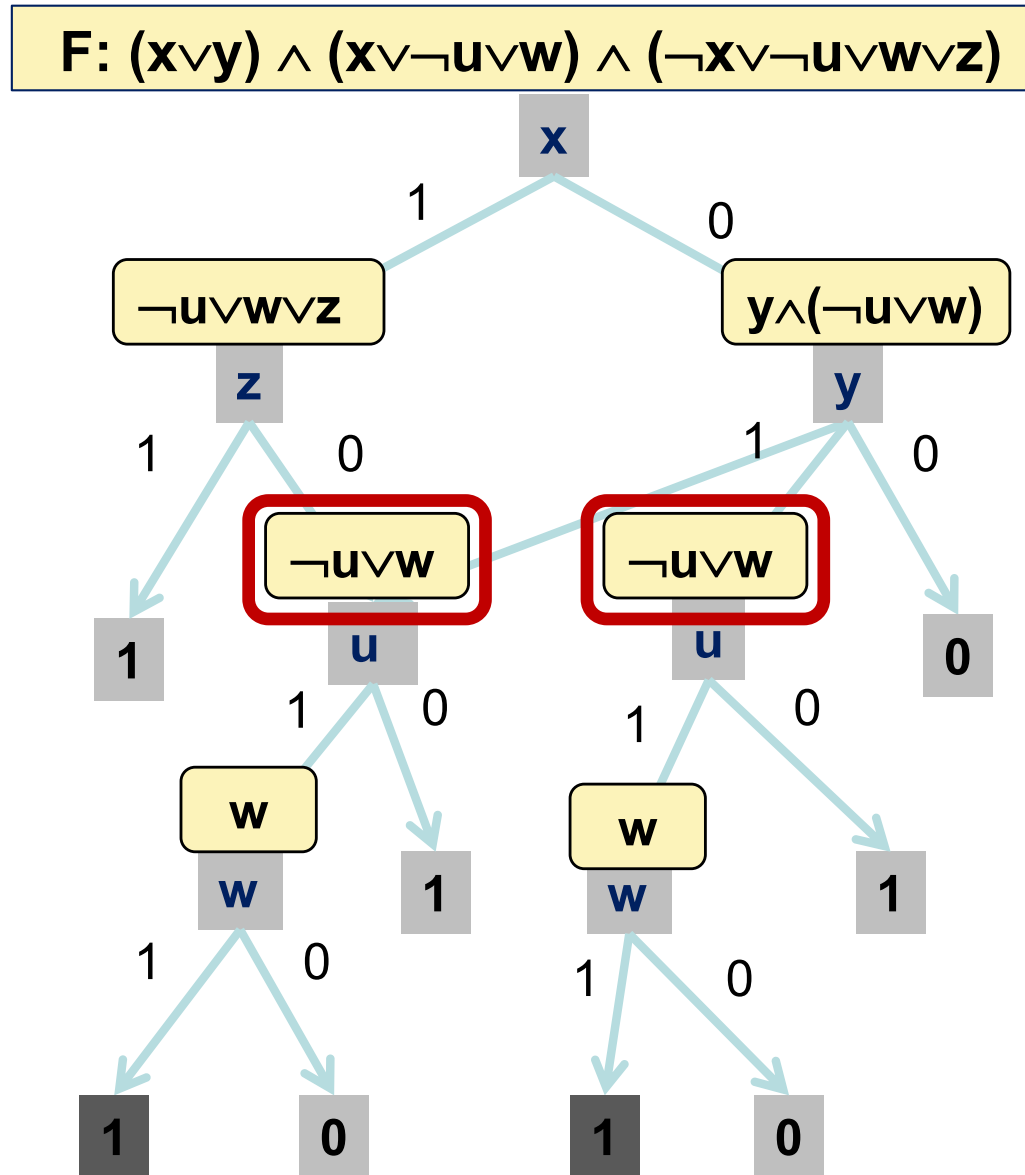
Davis, Martin; Putnam, Hilary. "A Computing Procedure for Quantification Theory". Journal of the ACM. 7 (3): 201–215. 1960.

Davis, Martin; Logemann, George; Loveland, Donald. "A Machine Program for Theorem Proving". Communications of the ACM. 5 (7): 394–397. 1962.

DPLL: Beispiel



Caching



Weighted Model Counting (WMC)

- Jedes Literal hat Gewicht $\in [0,1]$
 - Gewicht eines Modells = Produkt der Literalgewichte
 - Gewicht einer Formel = Summe der Gewichte der Modelle

Basic Weighted Model Counting (BWMC)

BWMC(F)

if F is empty **then return** 1

if F contains an empty clause **then return** 0

else choose a variable x to branch

return $\text{BWMC}(F|_{x=1}) * \text{weight}(x) +$
 $\text{BWMC}(F|_{x=0}) * \text{weight}(\neg x)$

Weighted Model Counting

- Modell = Erfüllende Belegung einer aussagenlogischen Formel Δ
- Zählen der Modelle: #SAT
- Weighted Model Counting (WMC)
 - Gewichte für Variablenzuweisung
 - Modellgewicht = Produkt der Variablengewichte $w(\cdot)$

$$\Delta = (\text{Rain} \Rightarrow \text{Cloudy})$$

Rain		Cloudy	
$w(R)$	$w(\neg R)$	$w(C)$	$w(\neg C)$
1	2	3	5

Rain	Cloudy	Model?	Weight
T	T	Yes	$1 * 3 = 3$
T	F	No	0
F	T	Yes	$2 * 3 = 6$
F	F	Yes	$2 * 5 = 10$
		+ —————	+ —————
		#SAT = 3	WMC = 19

Weighted First-Order Model Counting

Modell = Erfüllende Belegung einer aussagenlogischen Formel Δ

$$\Delta = \forall d (Rain(d) \Rightarrow Cloudy(d))$$

Days = {Monday
Tuesday}

Rain

d	$w(R(d))$	$w(\neg R(d))$
M	1	2
T	4	1

Cloudy

d	$w(C(d))$	$w(\neg C(d))$
M	3	5
T	6	2

Rain(M)	Cloudy(M)	Rain(T)	Cloudy(T)	Model?	Weight
T	T	T	T	Yes	$1 * 3 * 4 * 6 = 72$
T	F	T	T	No	0
F	T	T	T	Yes	$2 * 3 * 4 * 6 = 144$
F	F	T	T	Yes	$2 * 5 * 4 * 6 = 240$
T	T	T	F	No	0
T	F	T	F	No	0
F	T	T	F	No	0
F	F	T	F	No	0
T	T	F	T	Yes	$1 * 3 * 1 * 6 = 18$
T	F	F	T	No	0
F	T	F	T	Yes	$2 * 3 * 1 * 6 = 36$
F	F	F	T	Yes	$2 * 5 * 1 * 6 = 60$
T	T	F	F	Yes	$1 * 3 * 1 * 2 = 6$
T	F	F	F	No	0
F	T	F	F	Yes	$2 * 3 * 1 * 2 = 12$
F	F	F	F	Yes	$2 * 5 * 1 * 2 = 20$

+ +
#SAT = 9 **WFOMC = 608**

Gogate, V., & Domingos, P. Probabilistic theorem proving. Proc. UAI, **2012**.

Van den Broeck, G., Taghipour, N., Meert, W., Davis, J., & De Raedt, L., Lifted probabilistic inference by first-order knowledge compilation. In Proc. IJCAI-11, pp. 2178-2185, **2011**.

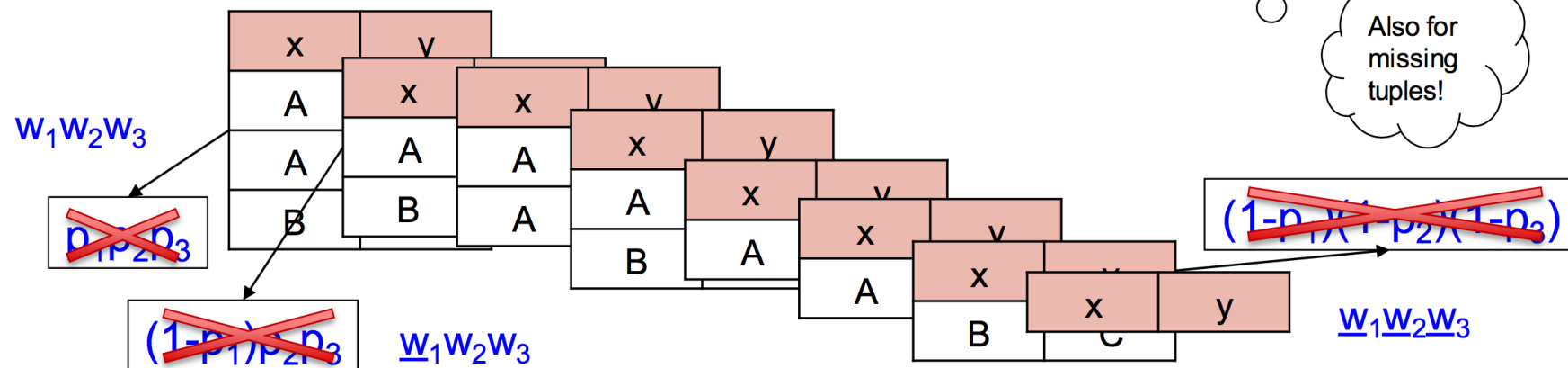
Von Wahrscheinlichkeiten zu Gewichten

Friend

x	y	P
A	B	p_1
A	C	p_2
B	C	p_3



x	y	$w(\text{Friend}(x,y))$	$w(\neg\text{Friend}(x,y))$
A	B	$w_1 = p_1$	$\underline{w}_1 = 1-p_1$
A	C	$w_2 = p_2$	$\underline{w}_2 = 1-p_2$
B	C	$w_3 = p_3$	$\underline{w}_3 = 1-p_3$
A	A	$w_4 = 0$	$\underline{w}_4 = 1$
A	C	$w_5 = 0$	$\underline{w}_5 = 1$
	...	...	

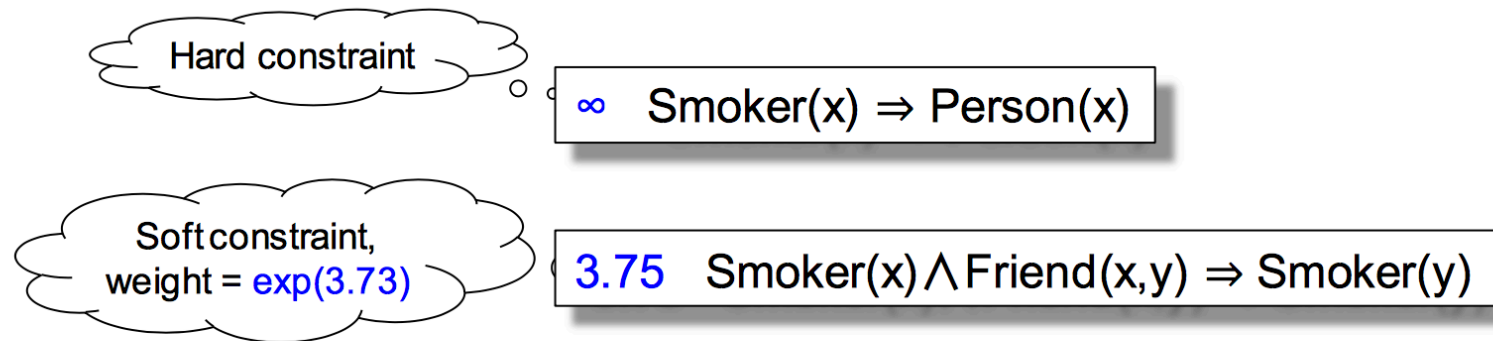


Diskussion

- Einfache Idee: Ersetze $p, 1-p$ durch $w, \underline{w}$
- Anfragebeantwortung durch WFOMC
- Für Wahrscheinlichkeitsraum:
 - Dividiere Weltgewicht durch Z = Summe aller Weltgewichte
- Warum Gewichte statt Wahrscheinlichkeiten?
- Verschiedene Formalismen zur Darstellung von Wahrscheinlichkeitsverteilungen

Markov Logic

Gewichtete Formeln zur Modellierung von Einschränkungen



An **MLN** is a set of constraints ($w, \Gamma(\mathbf{x})$), where w =weight, $\Gamma(\mathbf{x})$ =FO formula

Weight of a world = product of $\exp(w)$, for all **MLN** rules ($w, \Gamma(\mathbf{x})$) and grounding $\Gamma(\mathbf{a})$ that hold in that world

Probability of a world = **Weight** / **Z**
Z = sum of weights of all worlds (no longer a simple expression!)

Einschub: Warum exp?

- Gegeben:
 - Zustände, Formeln, ...: $s_1, s_2, \dots, s_n$
 - Dichte $p(s) = p_s$
- Maximum-Entropy-Prinzip:
 - Ohne weitere Information, wähle Dichte p_s , so dass Entropie maximiert wird

$$-\sum_s p_s \log p_s$$

- in Bezug auf Einschränkungen

$$\sum_s p_s f_i(s) = D_i, \quad \forall i$$

Einschub: Warum exp?

- Betrachte Lagrange-Funktional zur Bestimmung von p_s

$$L = -p_s \log p_s - \sum_i \lambda_i (\sum_s p_s f_i(s) - D_i) - \mu (\sum_s p_s - 1)$$

- Partielle Ableitungen von L in Bezug auf $p_s \rightarrow$

Nullstellenbestimmung ergibt (Boltzmann-Gibbs-Dichte):

$$p_s = \frac{\exp\left(-\sum_i \lambda_i f_i(s)\right)}{Z}$$

wobei Z ein Normalisierungsfaktor ist

Anfragebeantwortungsproblem

Gegeben

MLN:

- 0.7 $\text{Actor}(a) \Rightarrow \neg \text{Director}(a)$
- 1.2 $\text{Director}(a) \Rightarrow \neg \text{WorkedFor}(a,b)$
- 1.4 $\text{InMovie}(m,a) \wedge \text{WorkedFor}(a,b) \Rightarrow \text{InMovie}(m,b)$

Datenbanktabellen (wenn fehlend w=1)

Actor:

Name	w
Brando	2.9
Cruise	3.8
Coppola	1.1

WorkedFor:

Actor	Director	w
Brando	Coppola	2.5
Coppola	Brando	0.2
Cruise	Coppola	1.7

Berechne

$P(\text{InMovie}(\text{GodFather}, \text{Brando}) = ??$

Z-Berechnung

1. Formula Δ

If all MLN constraints are hard: $\Delta = \bigwedge_{(\infty, \Gamma(\mathbf{x})) \in \text{MLN}} (\forall \mathbf{x} \ \Gamma(\mathbf{x}))$

If $(w_i, \Gamma_i(\mathbf{x}))$ is a soft MLN constraint, then:

- Remove $(w_i, \Gamma_i(\mathbf{x}))$ from the MLN
- Add new probabilistic relation $F_i(\mathbf{x})$
- Add hard constraint $(\infty, \forall \mathbf{x} (F_i(\mathbf{x}) \Leftrightarrow \Gamma_i(\mathbf{x})))$

2. Weight function $w(.)$

For all constants $\mathbf{A}$, relations F_i ,

set $w(F_i(\mathbf{A})) = \exp(w_i)$, $w(\neg F_i(\mathbf{A})) = 1$

Theorem: $Z = \text{WFOMC}(\Delta)$

Better rewritings in
[Jha'12],[V.d.Broeck'14]

Van den Broeck, G., Meert, W., & Darwiche, A. (2013). Skolemization for weighted first-order model counting. Proceedings of KR.

Jha, A., & Suciu, D. (2012). Probabilistic databases with MarkoViews. Proceedings of the VLDB Endowment, 5(11), 1160-1171

Beispiel

1. Formula Δ

∞ $\text{Smoker}(x) \Rightarrow \text{Person}(x)$

3.75 $\text{Smoker}(x) \wedge \text{Friend}(x,y) \Rightarrow \text{Smoker}(y)$

$\Delta = \forall x (\text{Smoker}(x) \Rightarrow \text{Person}(x))$
 $\wedge \forall x \forall y (\text{F}(x,y) \Leftrightarrow [\text{Smoker}(x) \wedge \text{Friend}(x,y) \Rightarrow \text{Smoker}(y)])$

2. Weight function $w(.)$

F

x	y	$w(\text{F}(x,y))$	$w(\neg \text{F}(x,y))$
A	A	$\exp(3.75)$	1
A	B	$\exp(3.75)$	1
A	C	$\exp(3.75)$	1
B	A	$\exp(3.75)$	1
	...	...	

Note: if no tables given
for Smoker, Person, etc,
(i.e. no evidence)
then set their $w = \underline{w} = 1$

$$Z = \text{WFOMC}(\Delta)$$

Non-Standard-Datenbanken

Probabilistische Datenbanken

