

Datenbanken

Prof. Dr. Ralf Möller

Universität zu Lübeck

Institut für Informationssysteme

Karsten Martiny (Übungen)

und studentische Tutoren



Danksagung

Einige Präsentationen zu wurden mit Änderungen übernommen
aus einer Präsentation von Stephanie Scherzinger

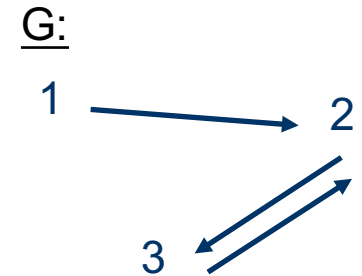


Datalog: Recursive Queries

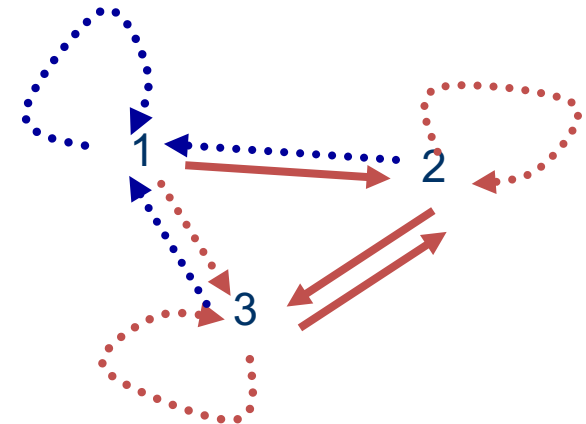
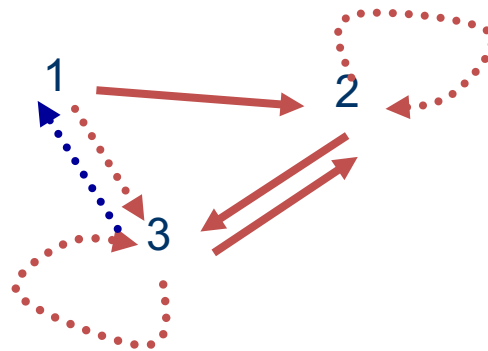
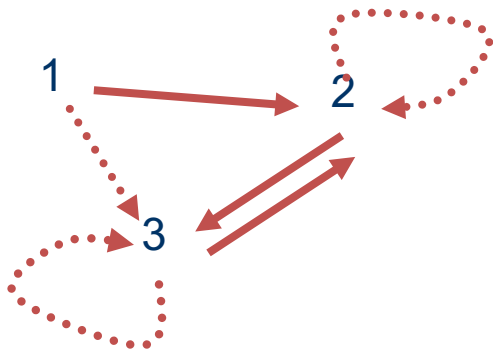
$$\forall x,y (T(x,y) \leftarrow G(x,y))$$

$$\forall x,y,z (T(x,y) \leftarrow (G(x,z) \wedge T(z,y)))$$

$$G(1,2), G(2,3), G(3,2)$$



Possible solutions:



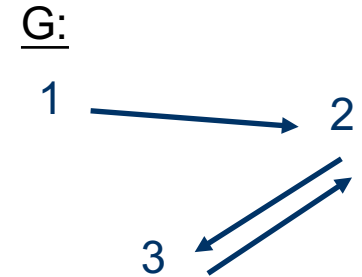
Herve Gallaire und Jack Minker *Logic and Data Bases*.
Symposium on Logic and Data Bases, Centre d'études et de
recherches de Toulouse in „Advances in Data Base Theory“.
Plenum Press, New York **1978**

Model Theoretic Approach

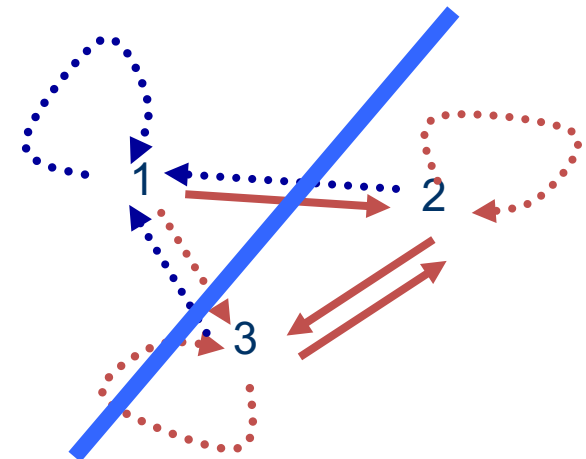
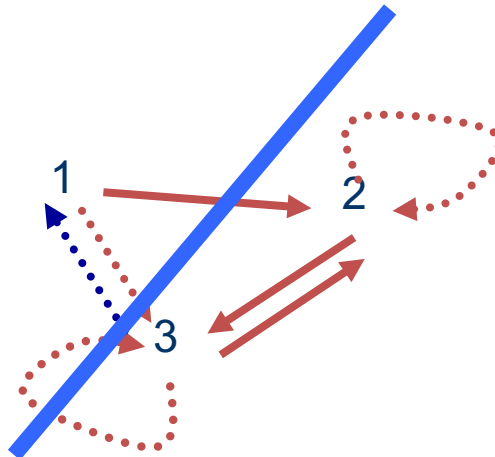
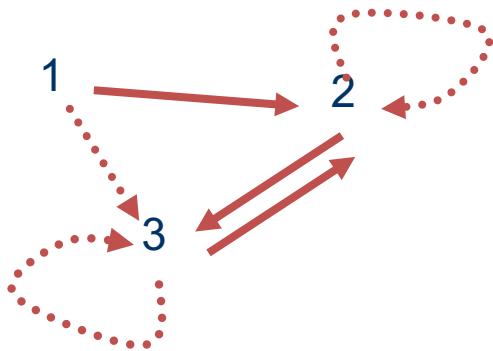
$$\forall x,y (T(x,y) \leftarrow G(x,y))$$

$$\forall x,y,z (T(x,y) \leftarrow (G(x,z) \wedge T(z,y)))$$

$$G(1,2), G(2,3), G(3,2)$$



Possible solutions:



→ Chose the *minimum model* ←

T consists of the smallest set of facts that make the sentences true.

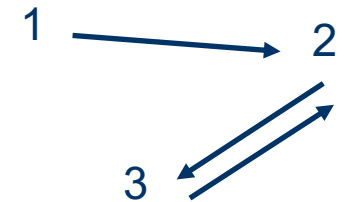
Recursive query in Datalog_(rec, no-neg) notation

Reachability or Transitive closure of a graph:

$$T(x, y) \quad :- \quad G(x, y)$$
$$T(x, y) \quad :- \quad G(x, z) , \quad T(z, y)$$

Intuition

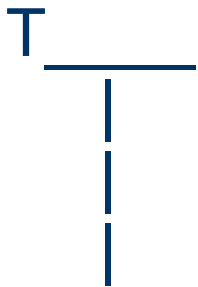
Transitive closure of a graph:



$$T(x, y) \quad :- \quad G(x, y)$$

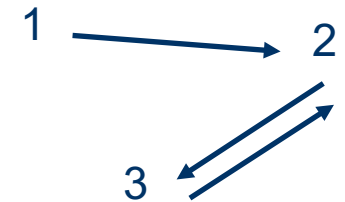
$$T(x, y) \quad :- \quad G(x, z) , \quad T(z, y)$$

G	
1	2
2	3
3	2



Intuition

Transitive closure of a graph:



$T(x, y) \text{ :- } G(1, 2)$

$T(x, y) \text{ :- } G(x, z), T(z, y)$

G

1	2
2	3
3	2

(1) Map from instances
over the relations in the rule body

Intuition

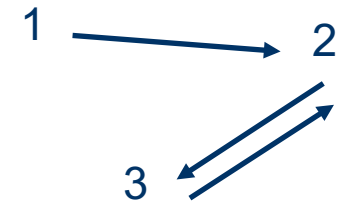
Transitive closure of a graph:

T

—
|
—

$T(1,2) :- G(1,2)$

$T(x,y) :- G(x,z), T(z,y)$

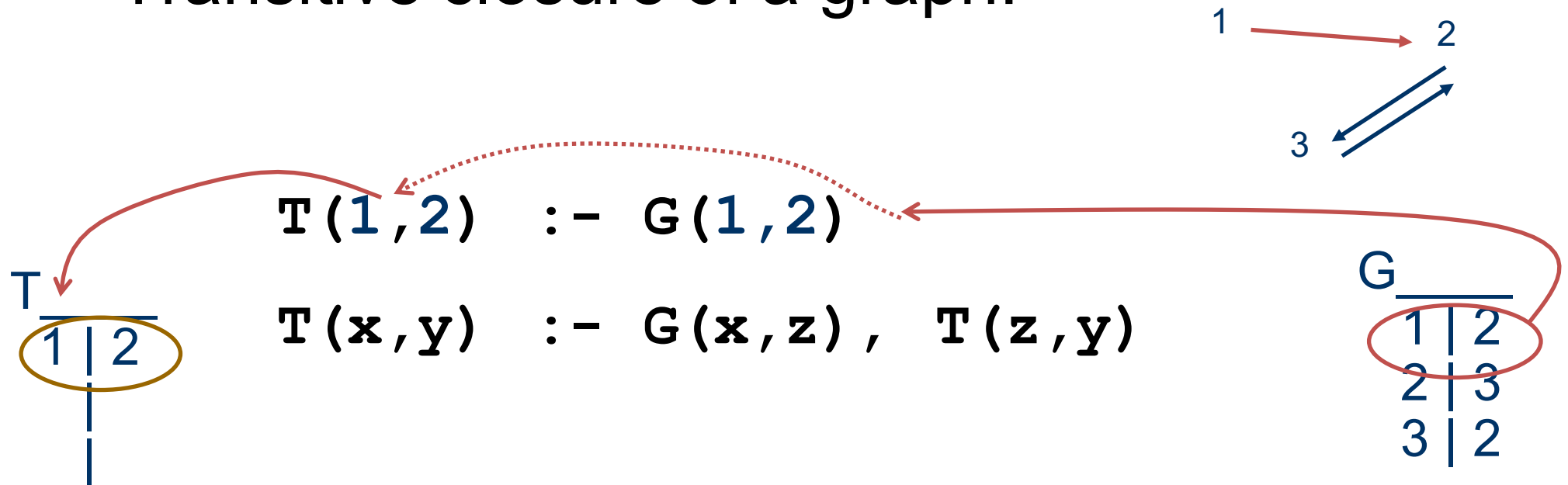


G	
1	2
2	3
3	2

(2) ... map to instances
over the relations in the rule head

Intuition

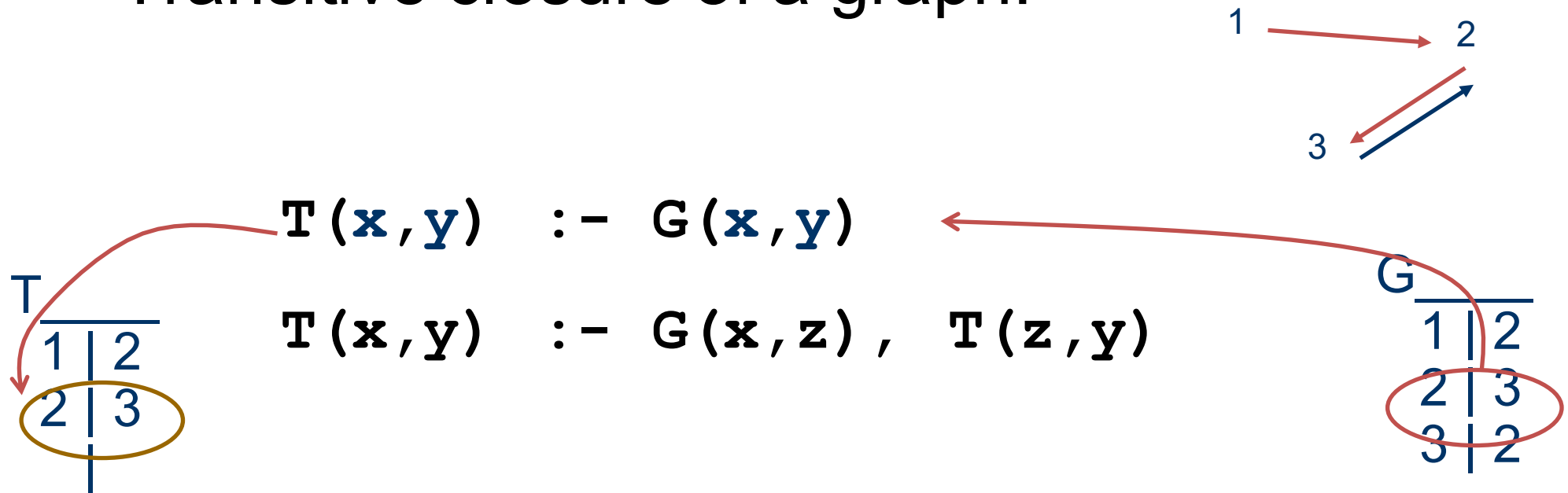
Transitive closure of a graph:



(2) ... map to instances
over the relations in the rule head

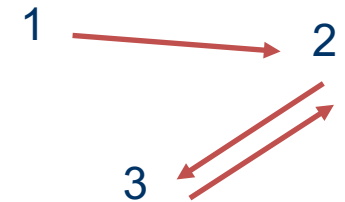
Intuition

Transitive closure of a graph:



Intuition

Transitive closure of a graph:



$T(x, y) :- G(x, y)$

$T(x, y) :- G(x, z), T(z, y)$

T

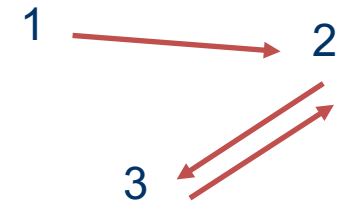
1	2
2	3
3	2

G

1	2
2	3
3	2

Intuition

Transitive closure of a graph:



$T(x, y) :- G(x, y)$

$T(x, y) :- G(1, 2), T(2, 3)$

T

1	2
2	3
3	2

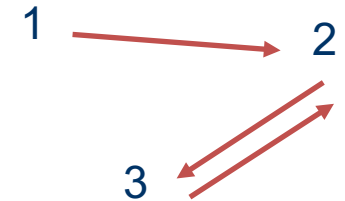
G

1	2
2	3
3	2

(1) Map from instances
over the relations in the rule body

Intuition

Transitive closure of a graph:



T	
1	2
2	3
3	2

$T(x, y) :- G(x, y)$

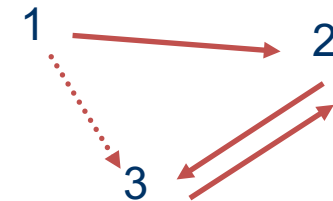
$T(1, 3) :- G(1, 2), T(2, 3)$

G	
1	2
2	3
3	2

(2) ... map to instances
over the relations in the rule head

Intuition

Transitive closure of a graph:



$T(x, y) \text{ :- } G(x, y)$

$T(1, 3) \text{ :- } G(1, 2), T(2, 3)$

T	
1	2
2	3
3	2
1	3

G	
1	2
2	3
3	2

(2) ... map to instances
over the relations in the rule head

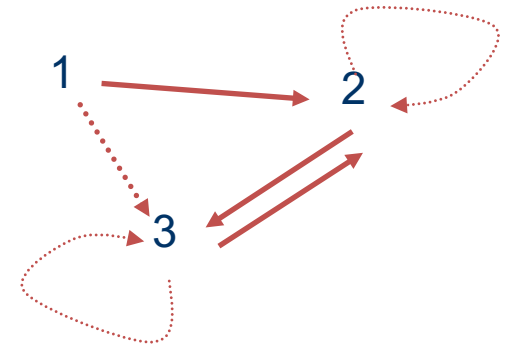
Intuition

Transitive closure of a graph:

T	
1	2
2	3
3	2
1	3
2	2
3	3

$T(x, y) \text{ :- } G(x, y)$

$T(x, y) \text{ :- } G(x, z), T(z, y)$



G	
1	2
2	3
3	2

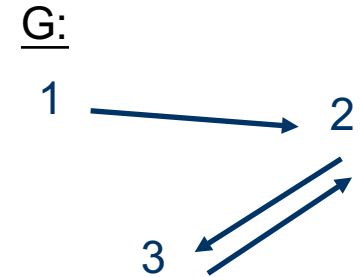
... repeat until fixpoint is reached
(datalog without negation is monotone)

Correctness

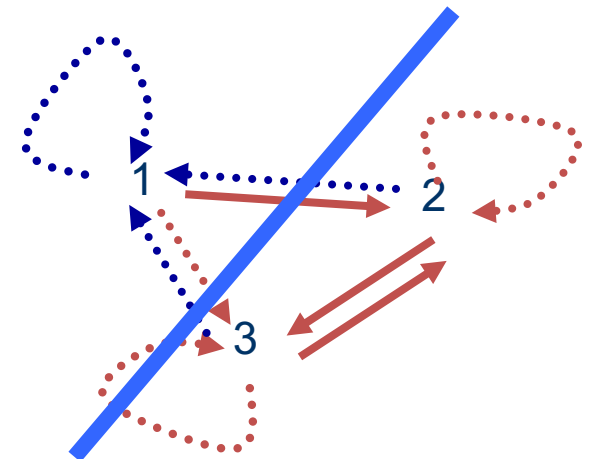
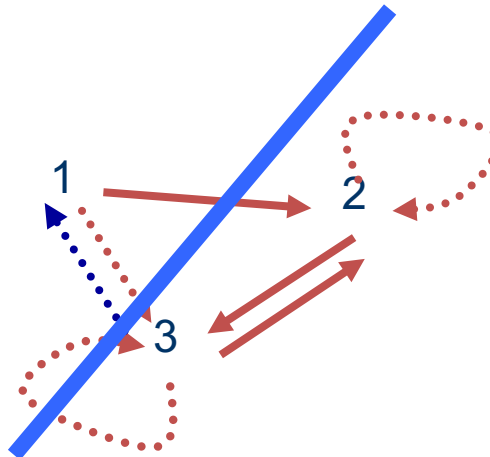
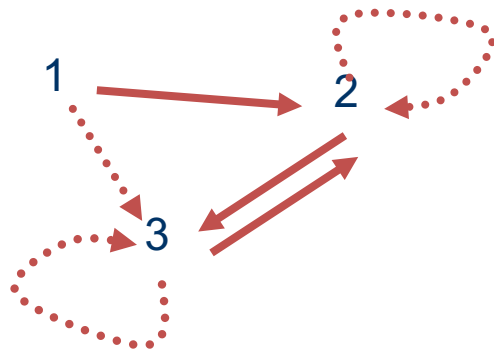
$$\forall x,y \quad (T(x,y) \leftarrow G(x,y))$$

$$\forall x,y,z \quad (T(x,y) \leftarrow (G(x,z) \wedge T(z,y)))$$

$$G(1,2), \quad G(2,3), \quad G(3,2)$$



Possible solutions:



→ Chose the *minimum model* ←

T consists of the smallest set of facts that make the sentences true.

Works only for Safe Datalog

- Here: Datalog without negation.
- Notion of *range-restricted* queries.
- This query is not safe:
`Colored_edges(x,y,col) :- G(x,y)`

→ Every variable in the head of a rule must also appear in the body (or must be range-restricted in case of built-in ops)

Evaluation in SQL with Joins

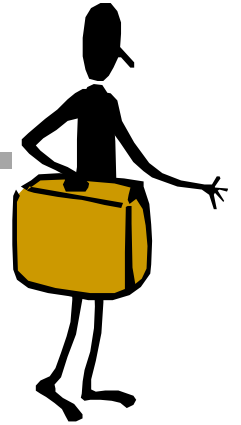
$$\forall x,y \quad (T(x,y) \leftarrow G(x,y))$$

$$\forall x,y,z \quad (T(x,y) \leftarrow (G(x,z) \wedge T(z,y)))$$

$$G(1,2), \quad G(2,3), \quad G(3, 2)$$

- $G(x,z) \wedge T(z,y)$ is a join
- Need joins until fixpoint over all data (naive evaluation)
- Need to consider only newly derived data (semi-naive)

Zusammenfassung, Kernpunkte



Keine Angst vor rekursiven Anfragen:

- Effiziente Auswertung möglich
- Keine Endlosschleifen zu befürchten
- Aber: Sicherheitsbedingungen
(gilt auch für rekursive Anfragen in SQL)