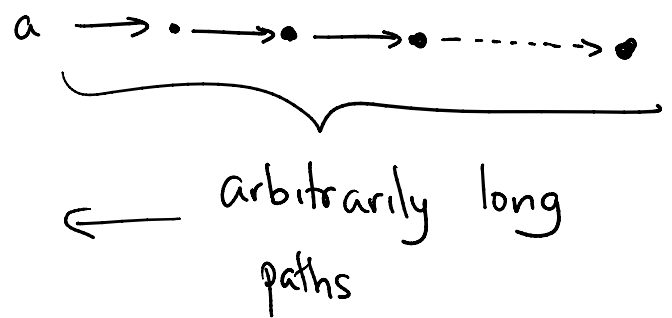


Graph Query Languages

Key idea: Recursion

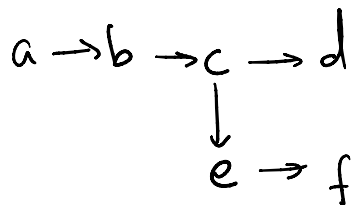
"All nodes which can be reached from a"



Recursion

Transitive closure

- If you confine yourself to relational algebra, you can only determine "bounded" reachability



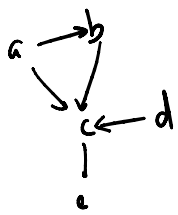
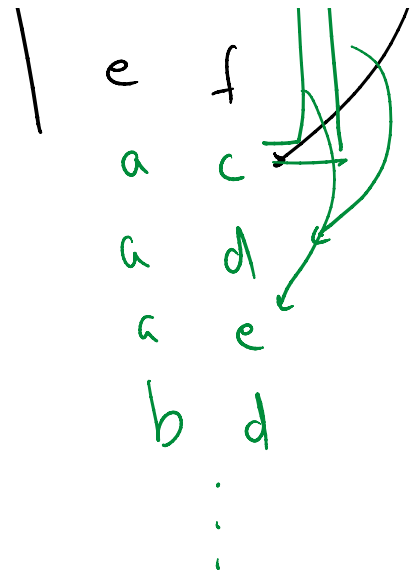
Edge

a	b
b	c
c	d
c	e
e	f

Edge⁺

a	b
b	c
c	d
c	e
e	f

e f



Edge	
a	b
a	c
b	c
d	c
c	e

Reachability	
a	b
a	c
a	e
b	c
b	e
d	c
d	e
c	e

① If there is an edge from a to b then b is "reachable" from a.

② If there is an edge from a to b & a path from b to c, then

there is a path from a to c.

$(\forall a, b)$
Edge (a, b)

①

Reachable (a, b)

$(\forall a, b, c)$

Edge (a, b)

Reachable (b, c)

Fixpoint

[Repeat until nothing more can be derived.]

②

Reachable (a, c)

②

$\text{fib}(x) =$ if $x=0$ or 1 then 1
else $\text{fib}(x-1) + \text{fib}(x-2)$

while (—)
do {
} ← Repeat these instructions
↑ until this becomes false

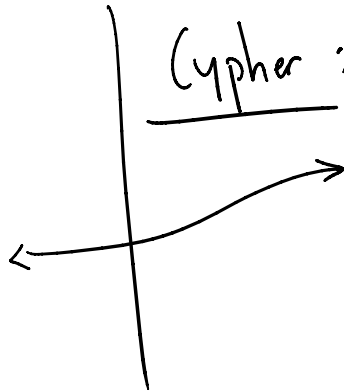
Repeat
|
|
| until nothing changes

bool changed := true
while (changed) {
 changed := false
 {
 if (—) changed := true
 }

Neo 4j
System

Cypher
Query language

SQL: select _____
from _____
where _____



Cypher: match _____
where _____
return _____

path query



"Find all actors who co-starred with CE"

match

(p: Person) — [ACTED IN] — (m: Move)

where p.name = "CE"

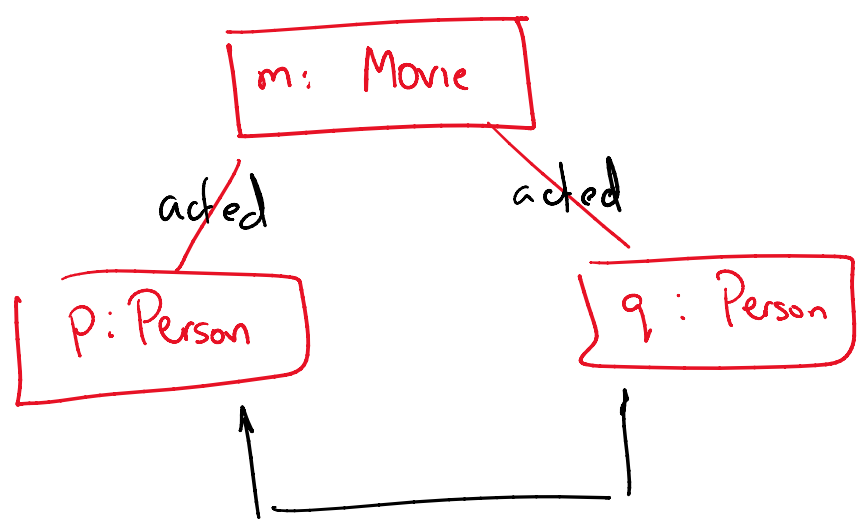
return q

[ACTED IN]

(q: Person)

1 m.

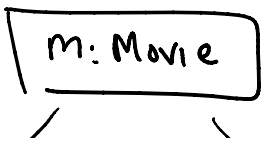
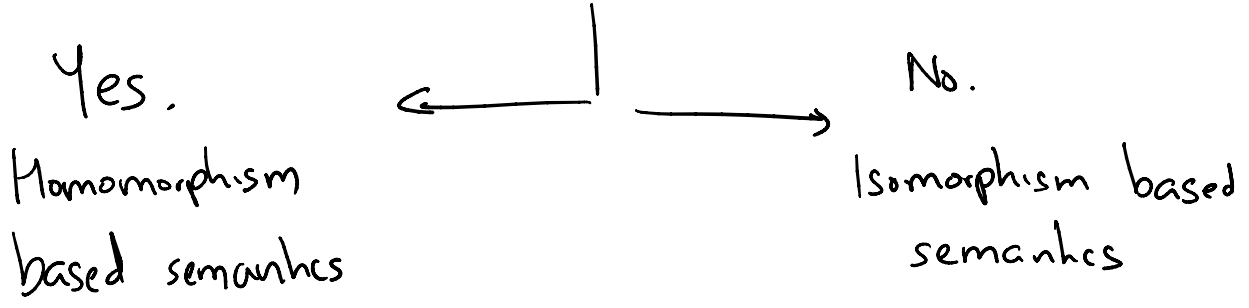
(q: Person)

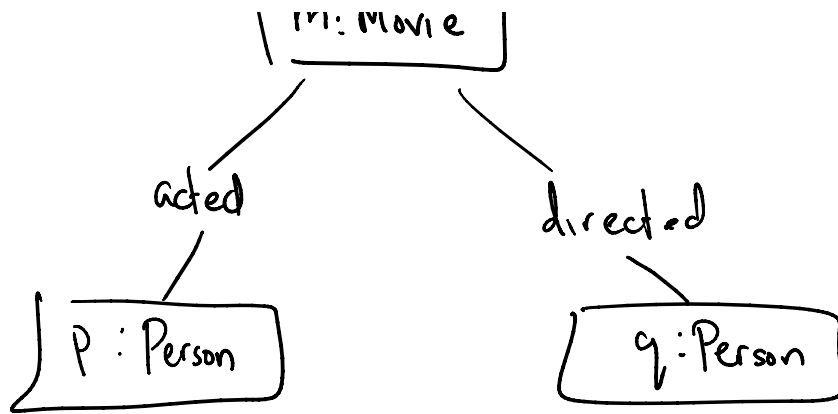


p & q are different nodes in the template.

Should they therefore be different things in the result?

"Did CE co-star in a movie with himself?"





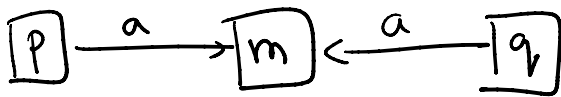
Homomorphic

Emit all p, q, m s
such that p acted in m
& q directed m

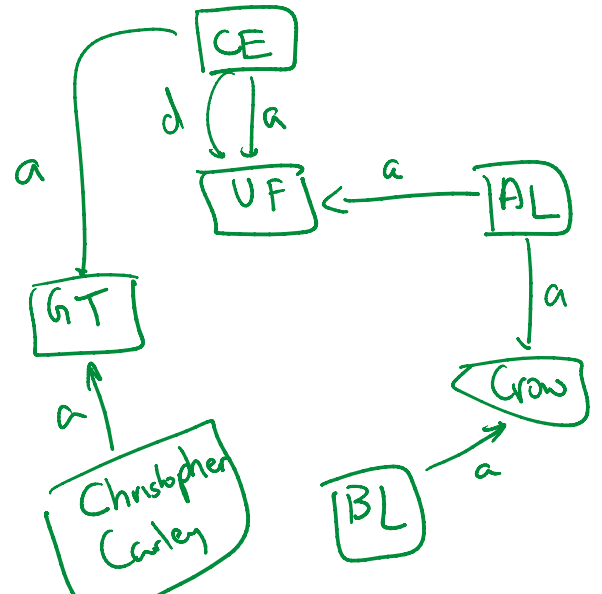
"CE"
Isomorphic

Emit all p, q, m s
such that p acted in m
& q directed m
& $p \neq q, q \neq m, m \neq p$

How to evaluate graph queries?



- Guess value for p .
- Guess value for m .
- Guess value for q .



guess value for q .

Carley PL

If match, then emit!

Repeat

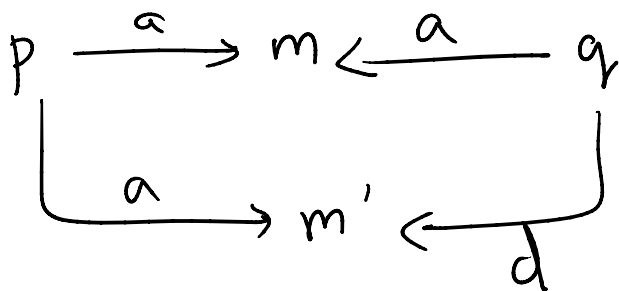
for all p

for all q

for all m

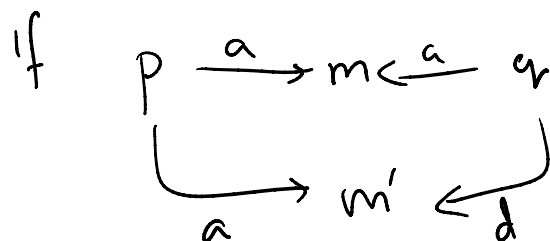
if $p \xrightarrow{a} m \xleftarrow{a} q$

then emit



for all p , for all q :

for all m , for all m' :



then emit

for all p :

for all m s.t. $p \xrightarrow{a} m$

for all q s.t. $m \xleftarrow{a} q$

if $\exists m' \quad p \xrightarrow{a} m' \xleftarrow{d} q$

then emit _____

otherwise, "guess again"!

Complexity of query evaluation

Data complexity: P

Query complexity: NP-complete

Subgraph isomorphism \uparrow