

Query a Graph

with

graph query engine

A Graph

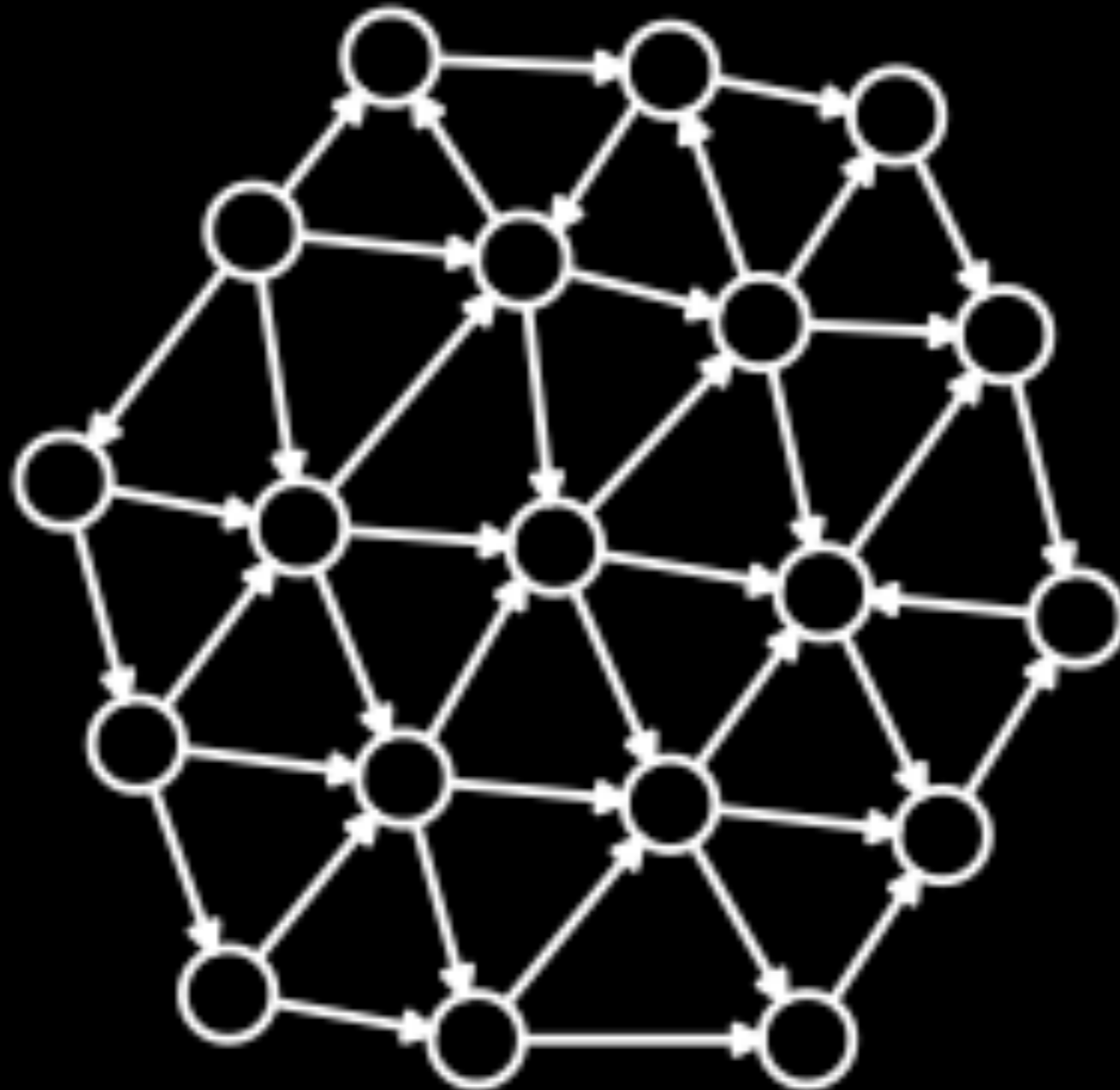
Can't see the Patterns for the Trees



<http://maxdemarzi.com/2012/02/13/visualizing-a-network-with-cypher/>

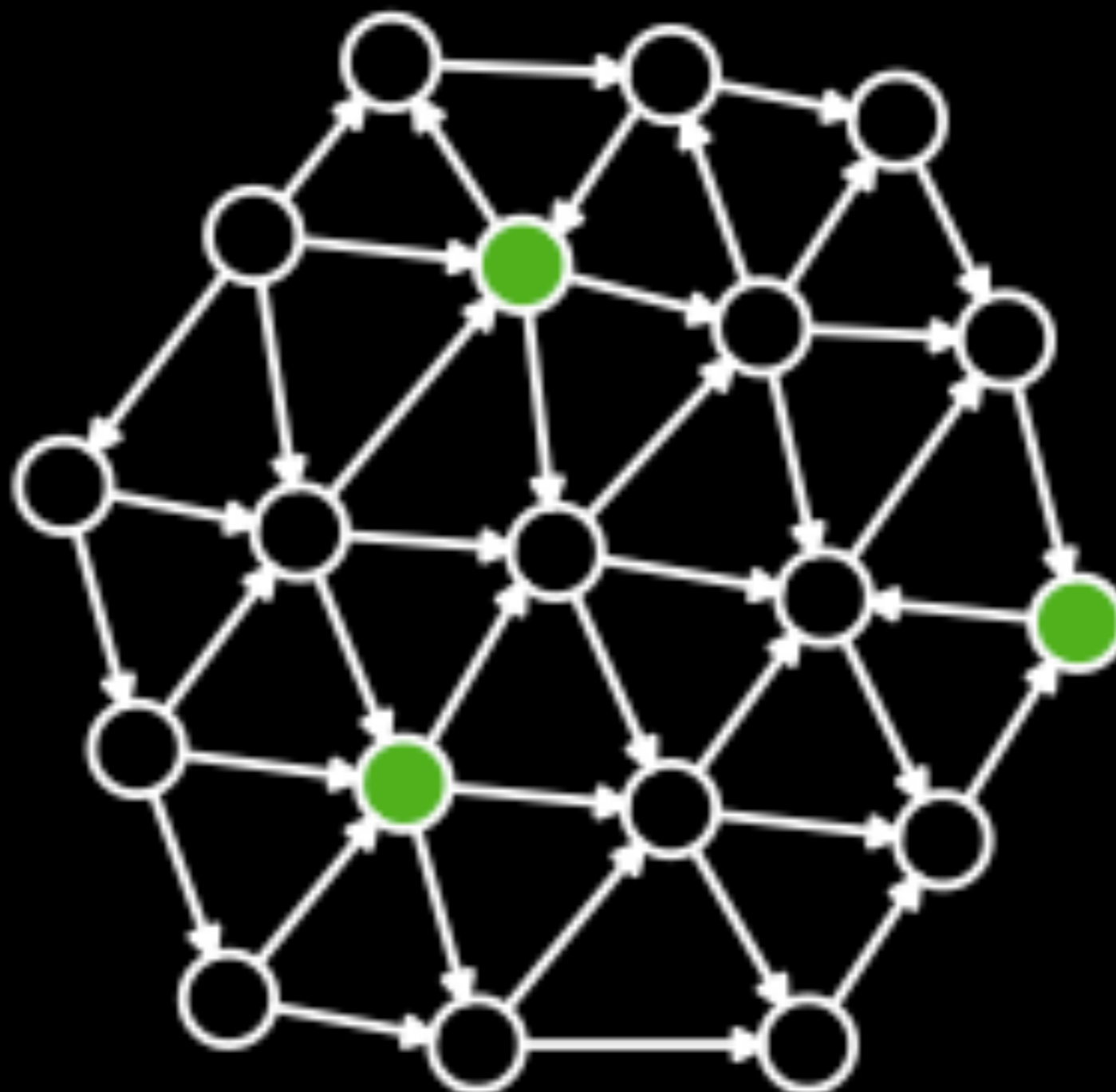
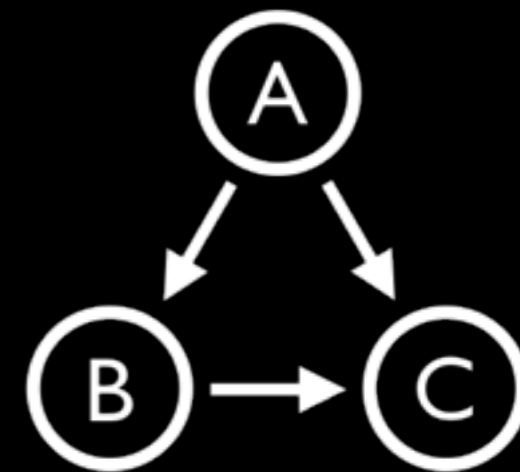
Patterns?

Where?



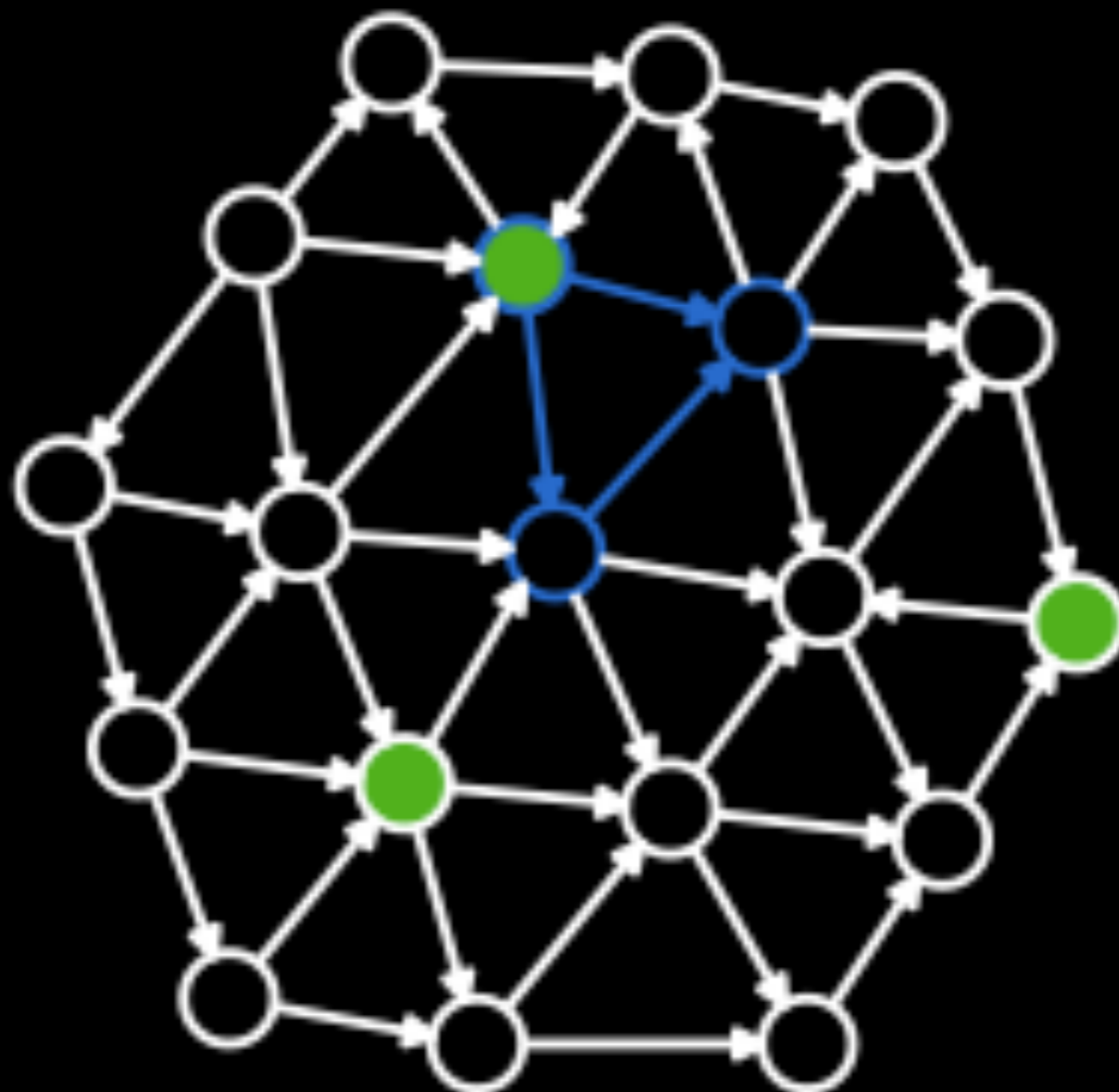
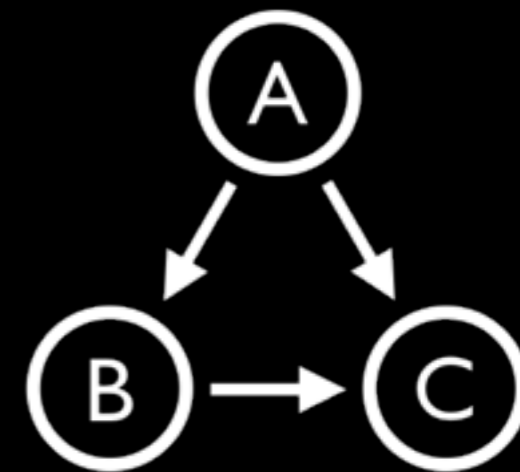
Patterns?

Describe them!



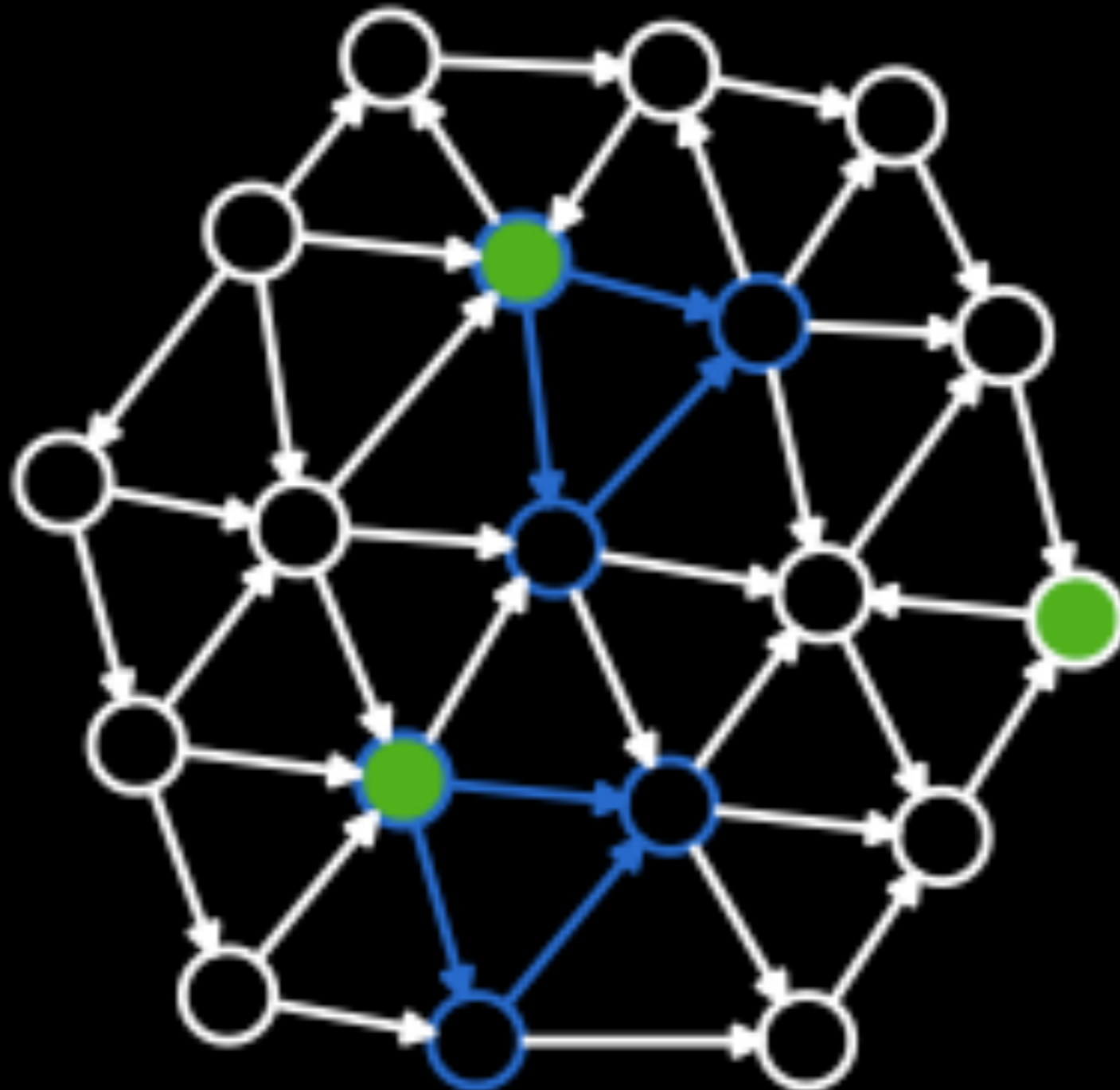
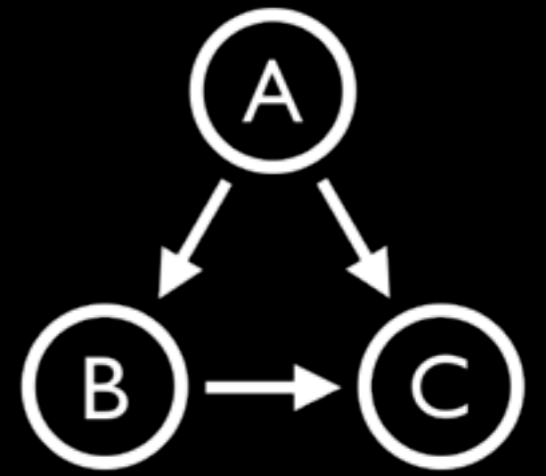
Patterns?

Find them on bound nodes



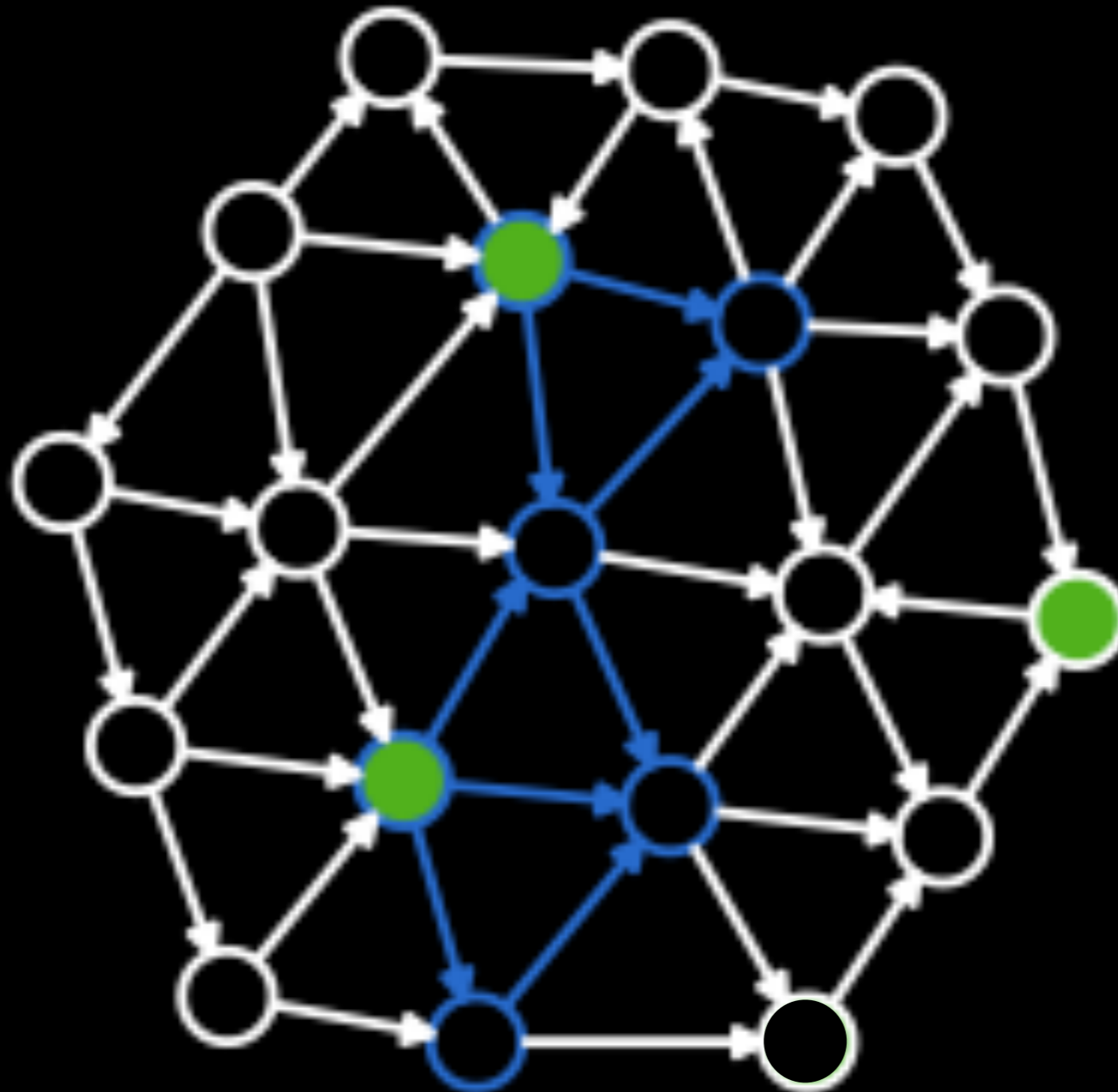
Patterns?

There are more !



Patterns?

Make them visible!



How ?

A Graph Query Language
with
Pattern Matching

SparQL, SQL ?

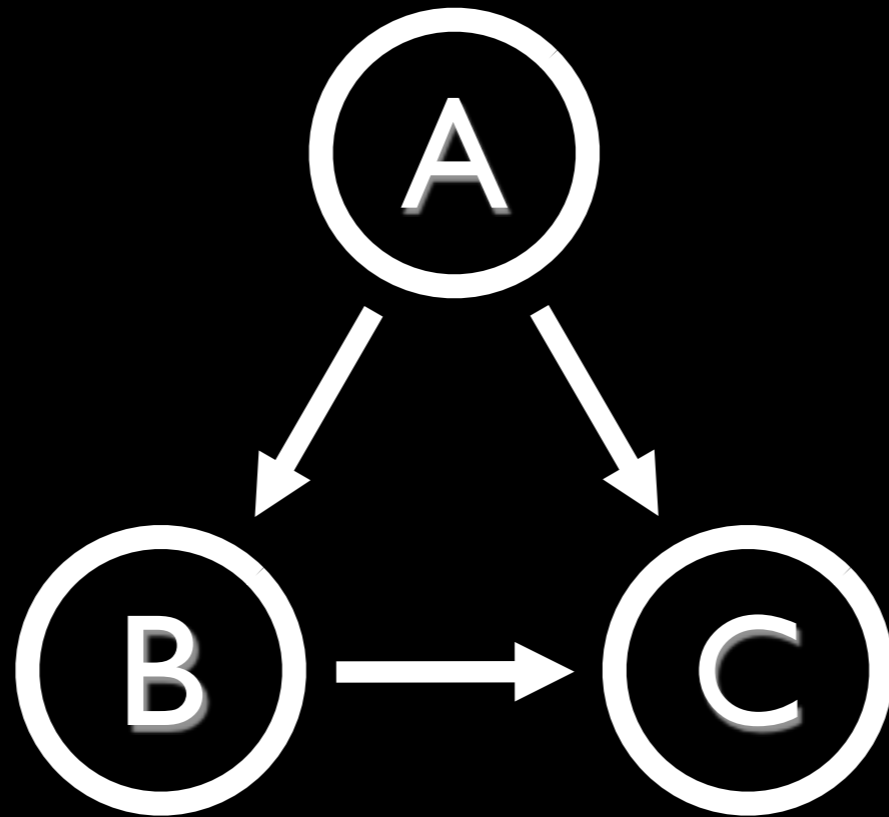
No,

Cypher

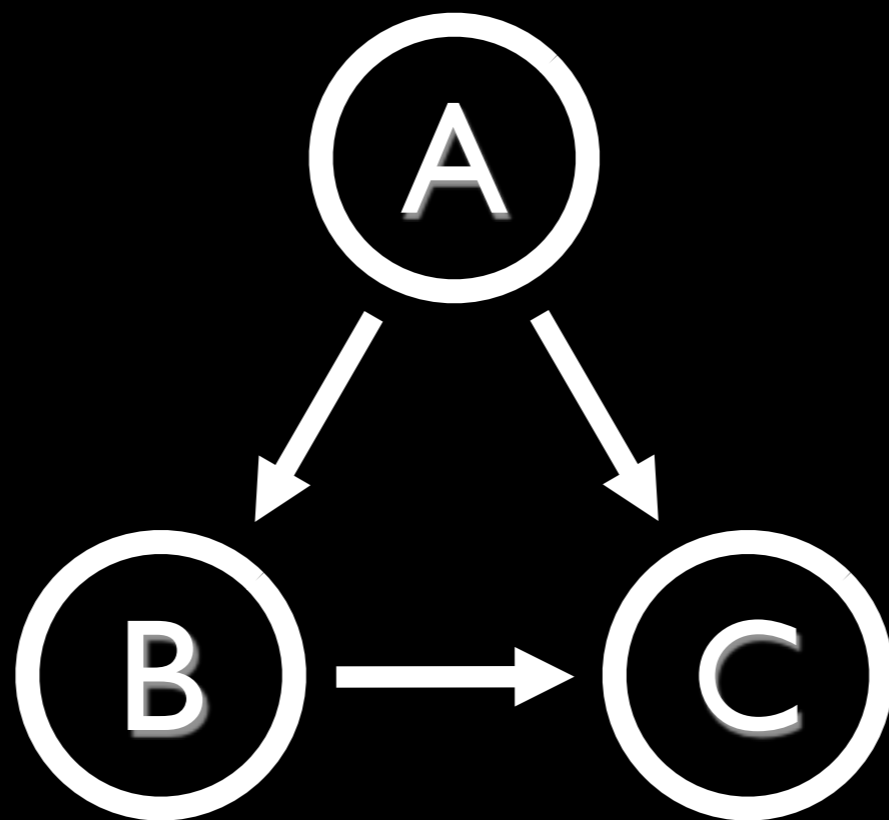
developed by and for

Neo4j

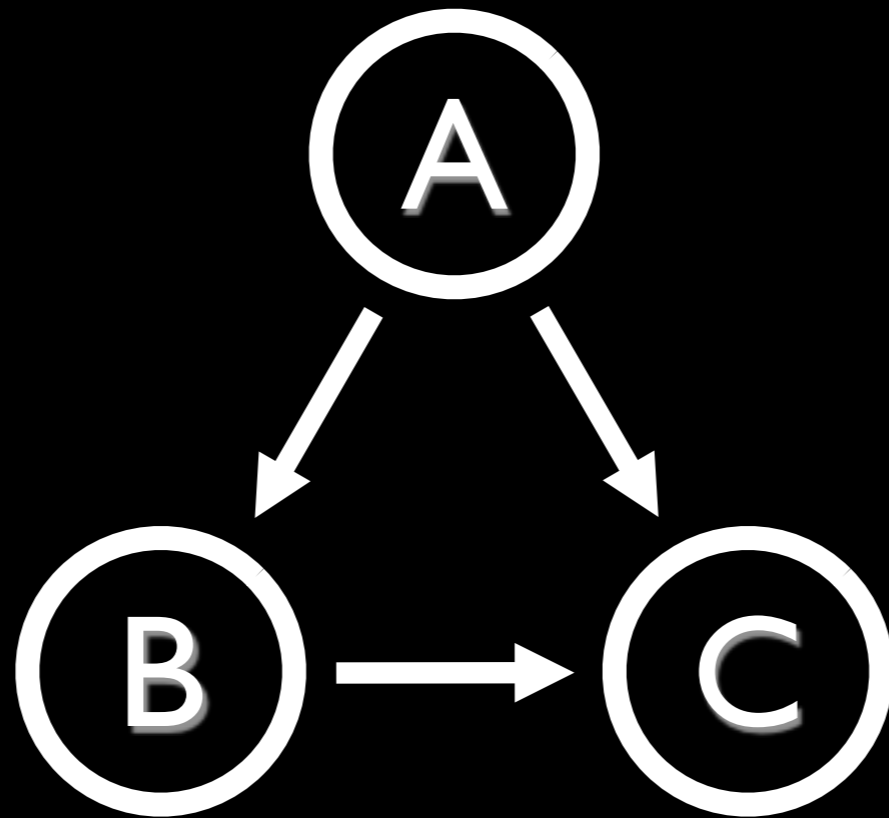
Can I draw Patterns?



Sure, with ASCII ART!

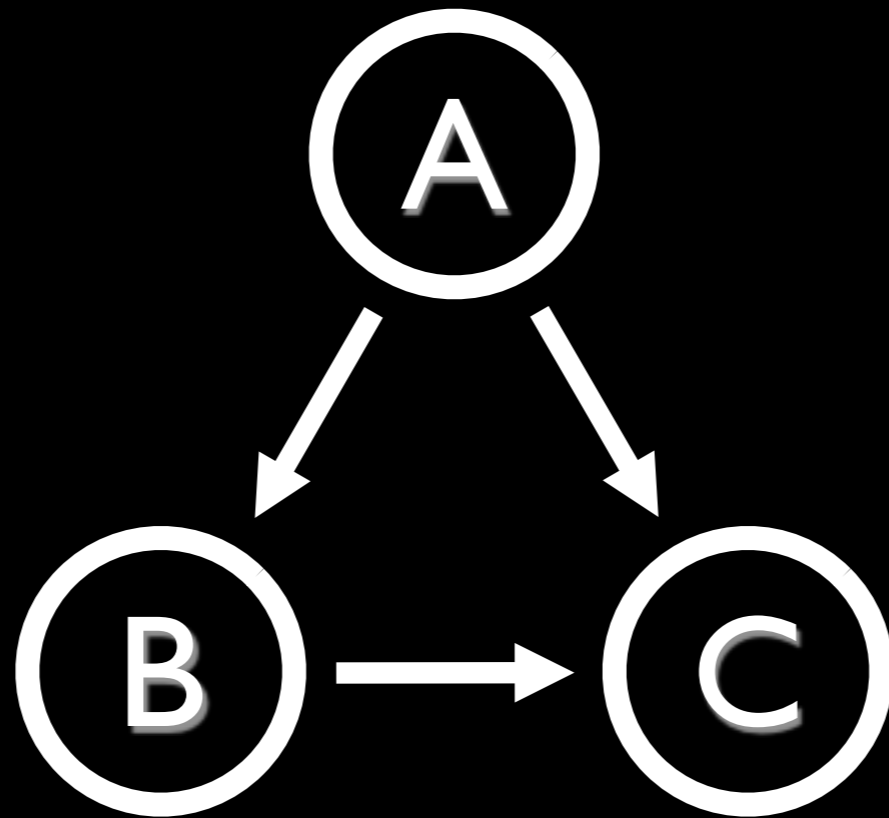


Sure, with ASCII ART!



A --> B --> C, A --> C

Sure, with ASCII ART!



A --> B --> C, A --> C

A --> B --> C <-- A

Example

Labeled Directed Relationship



Example

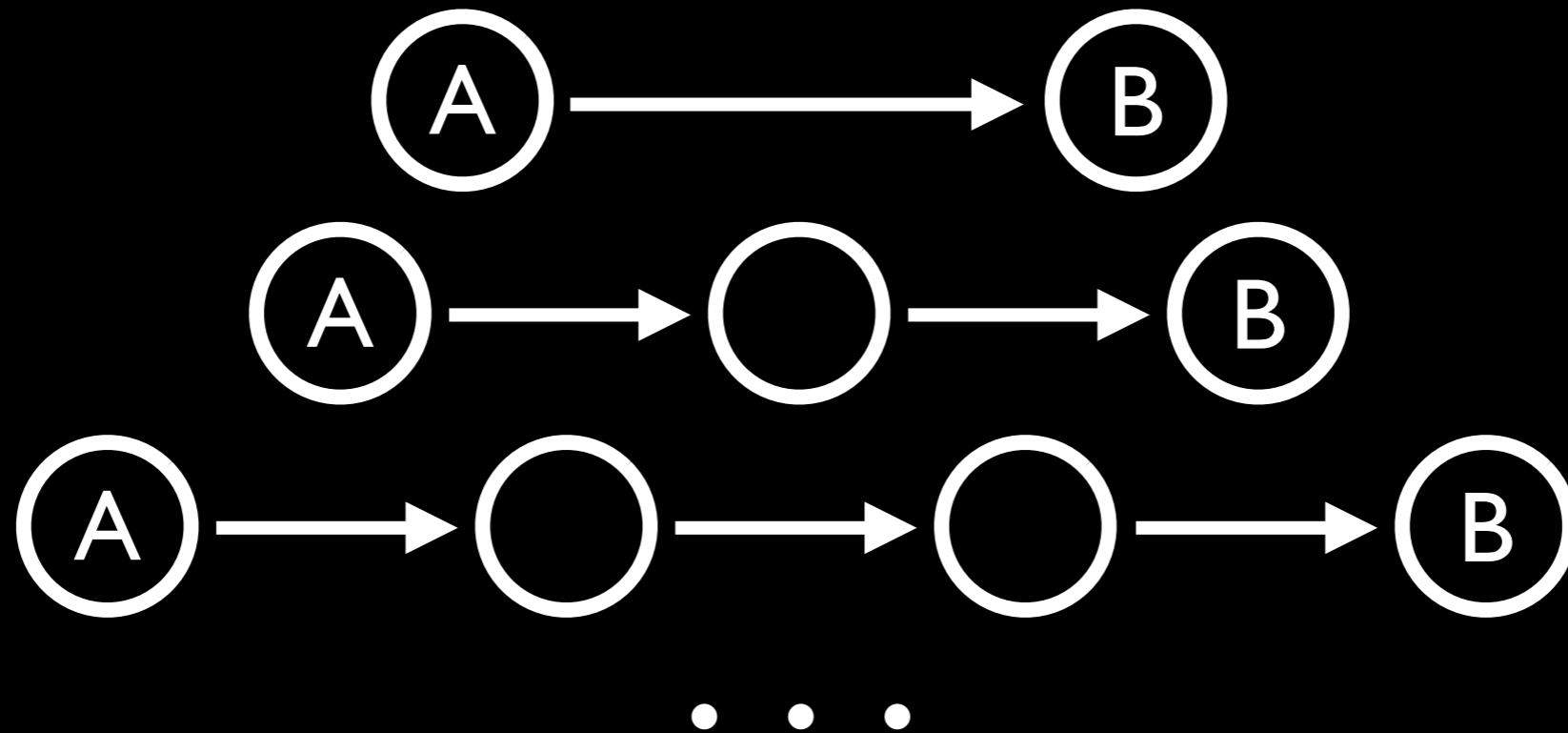
Labeled Directed Relationship



$(A) - [: \text{LOVES}] - \> (B)$

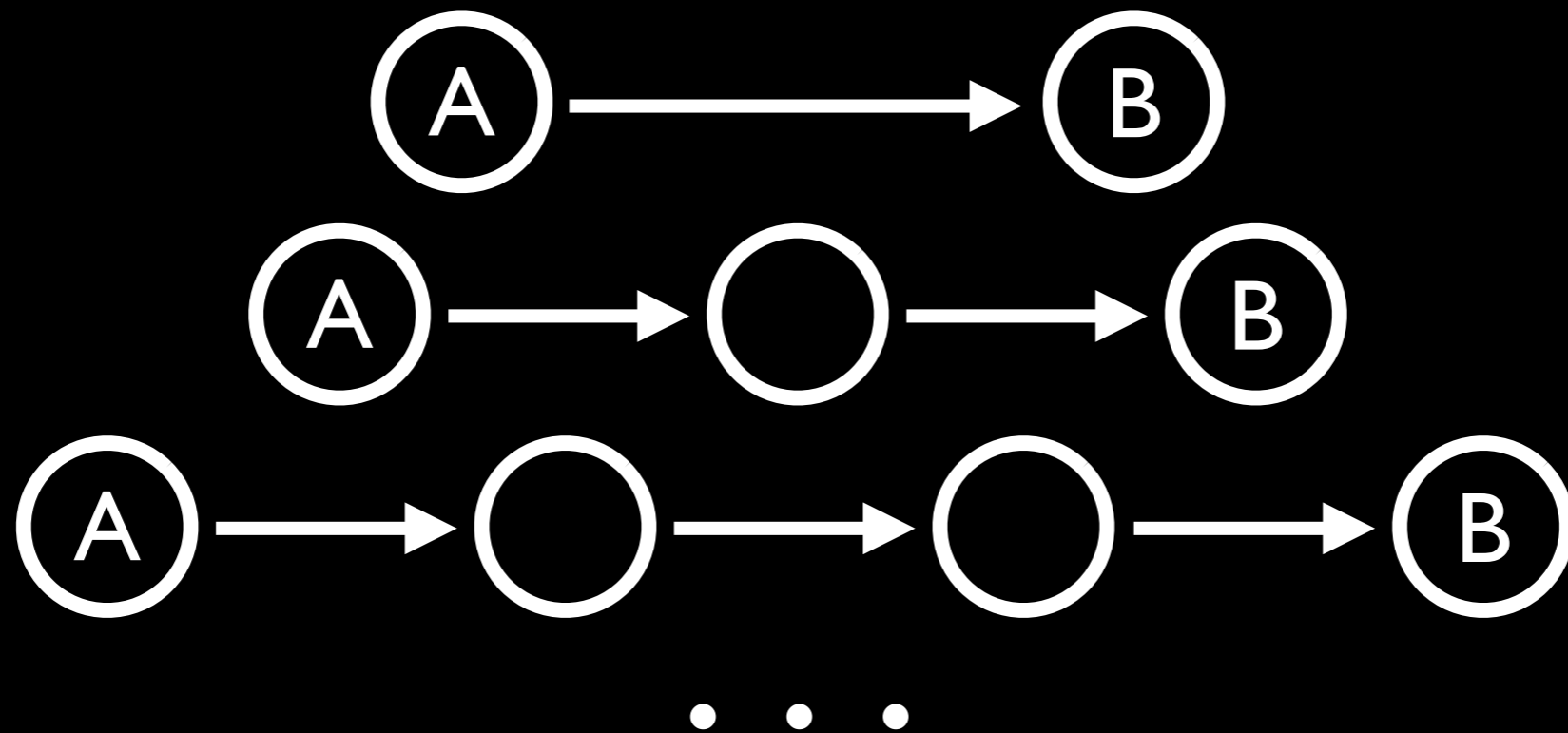
Example

Variable Length Path



Example

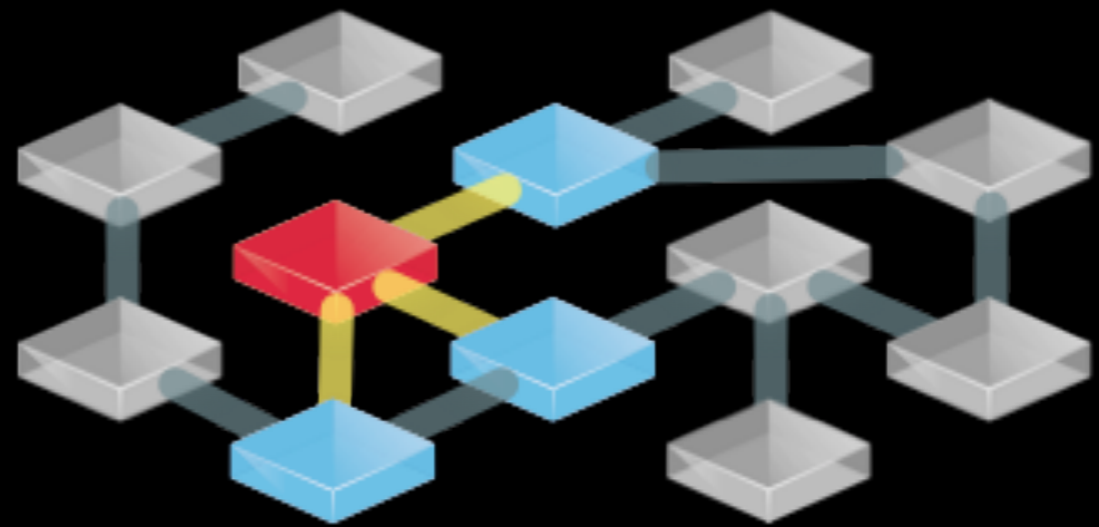
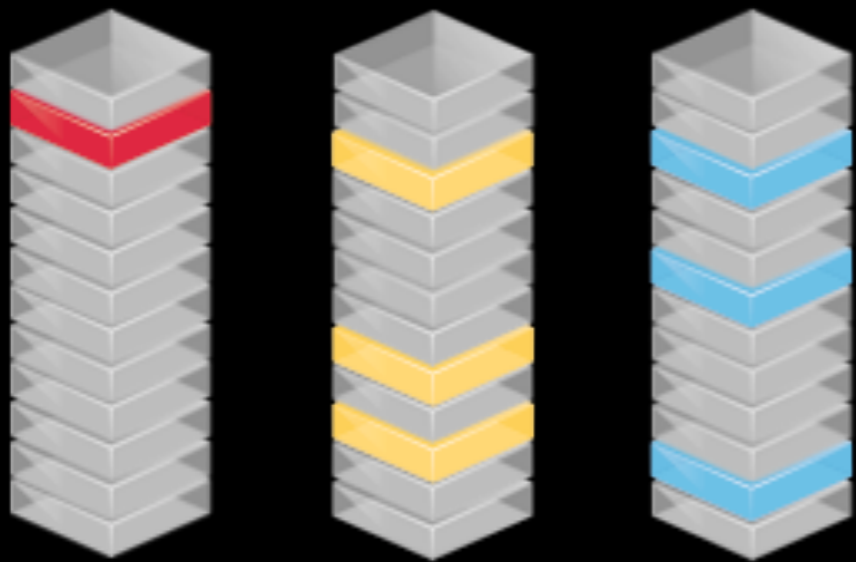
Variable Length Path



A - [*] -> B

Give me: JOINS

```
SELECT skills.*, user_skill.*  
FROM users  
JOIN user_skill ON users.id = user_skill.user_id  
JOIN skills ON user_skill.skill_id = skill.id  
WHERE users.id = 1
```



user



user_skill



skill

```
START user = node(1)  
MATCH user -[r:USER_SKILL]-> skill  
RETURN skill, r
```

Give me:

Old, Influential Friends

```
START me = node(...)
MATCH (me) - [f:FRIEND] - (old_friend)
      - [:FRIEND] - (fof)
WHERE ({today}-f.begin) > 365*10

WITH old_friend, collect(fof.name) as names

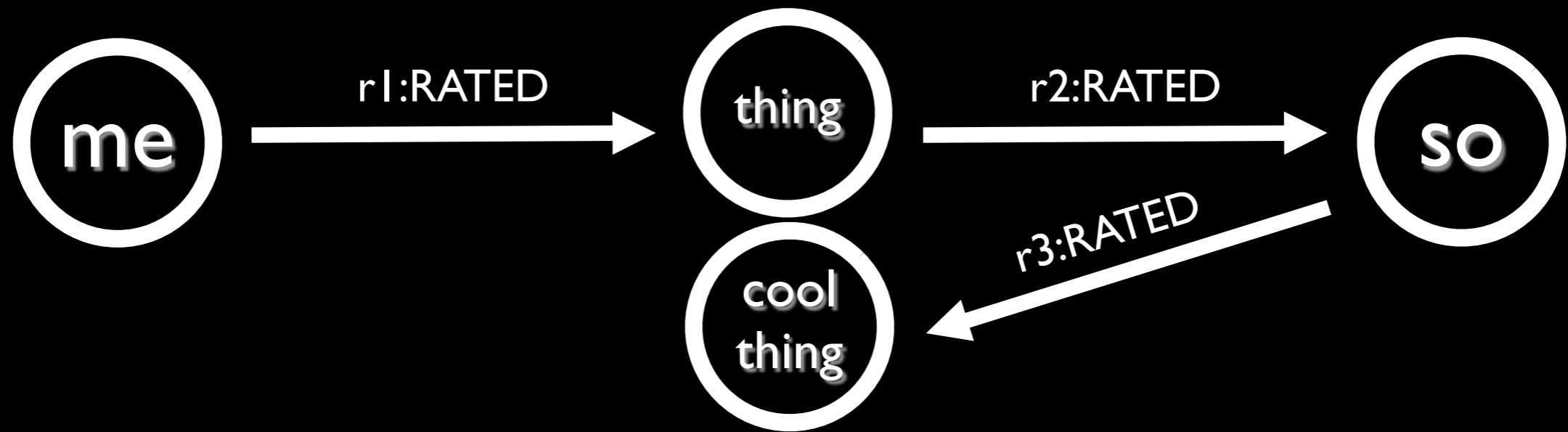
WHERE length(names) > 100
RETURN old_friend, names
ORDER BY old_friend.name ASC
```



Give me:

Simple Recommendation

```
START me = node(...)
MATCH (me) -[r1:RATED ]->(thing)
      <-[r2:RATED ]- (someone)
      -[r3:RATED ]->(cool_thing)
WHERE ABS(r1.stars-r2.stars) <= 2
      AND r3.stars > 3
RETURN cool_thing, count(*) AS cnt
ORDER BY cnt DESC LIMIT 10
```





Results ?

- Tables:
for Human Brainz & Tools
- Graphs:
to highlight, visualize,
export & refining queries

The RabbitHole

current    

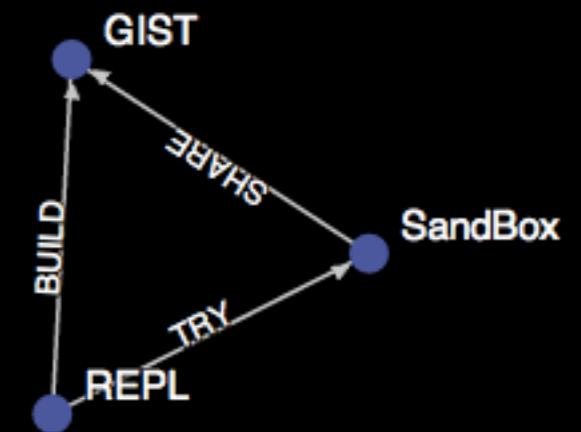
You can modify and query this graph by entering statements in the input field at the bottom.
For some syntax help hit the  button. If you want to share your graph, just do it with 

```
start n=node(*)
match n-[r]-m
return n,r
```

n	r
Node[0]{name->"REPL"}	:TRY[0] {}
Node[0]{name->"REPL"}	:BUILD[2] {}
Node[1]{name->"GIST"}	:SHARE[1] {}
Node[1]{name->"GIST"}	:BUILD[2] {}
Node[2]{name->"SandBox"}	:TRY[0] {}
Node[2]{name->"SandBox"}	:SHARE[1] {}

6 rows
0 ms

```
start n=node(*) match n-[r]-m return n,r
```



<http://console.neo4j.org>

This Graph: <http://tinyurl.com/7cnvmlq>

Cypher: Some Goals

Declarative

ASCII-art patterns

Pattern Matching

External DSL

Closures

SQL Familiarity

