NoSQL Day 2014

Neo4J: Graph Database

@michellesanver

About: Michelle Sanver

@michellesanver

About: Michelle Sanver

President

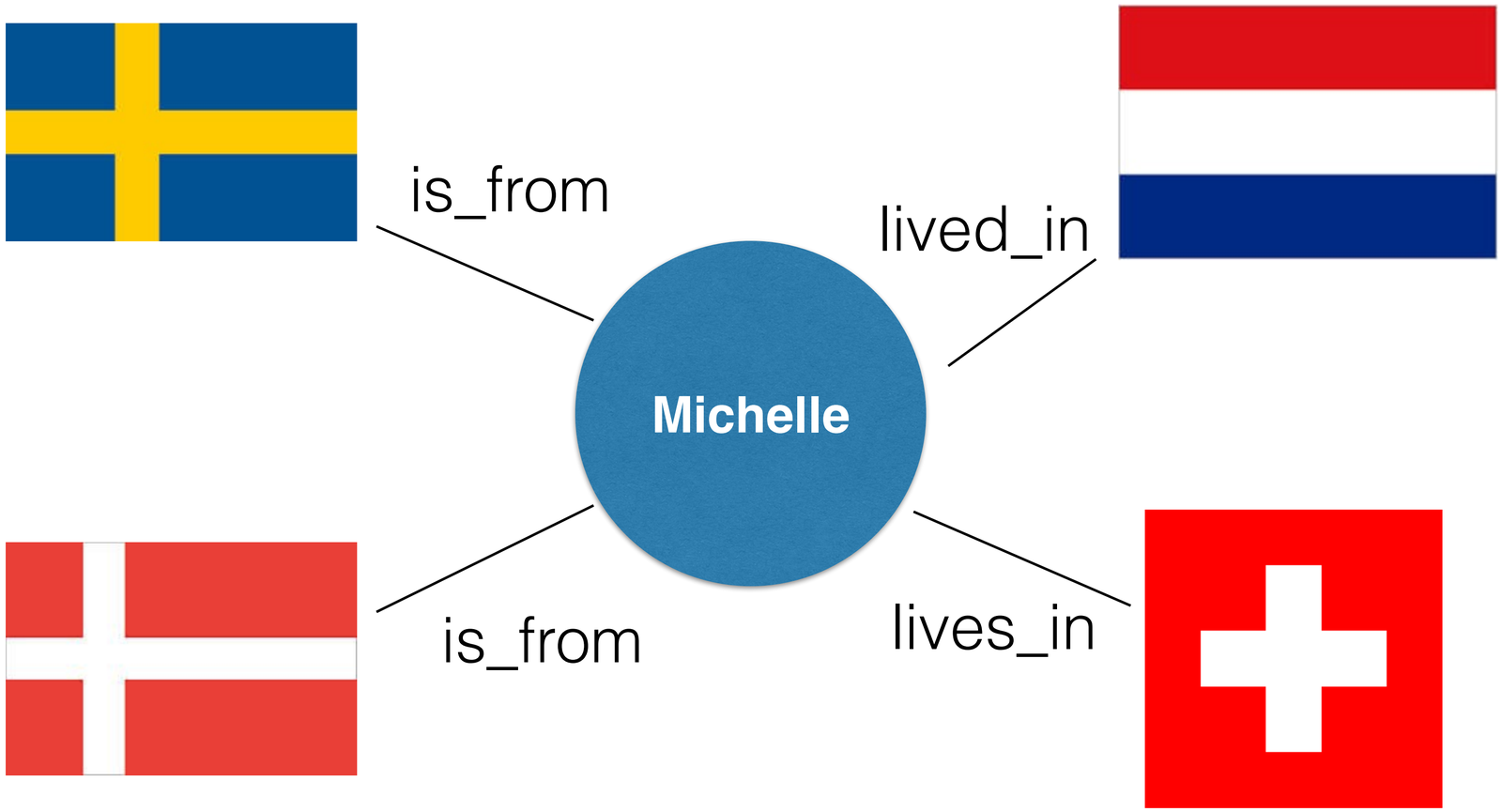
About: Michelle Sanver

Code Addict

About: Michelle Sanver

Zürich, Switzerland

About: Michelle Sanver



(Michelle)-[:LOVES]->(Neo4j)

@michellesanver

Setting expectations.

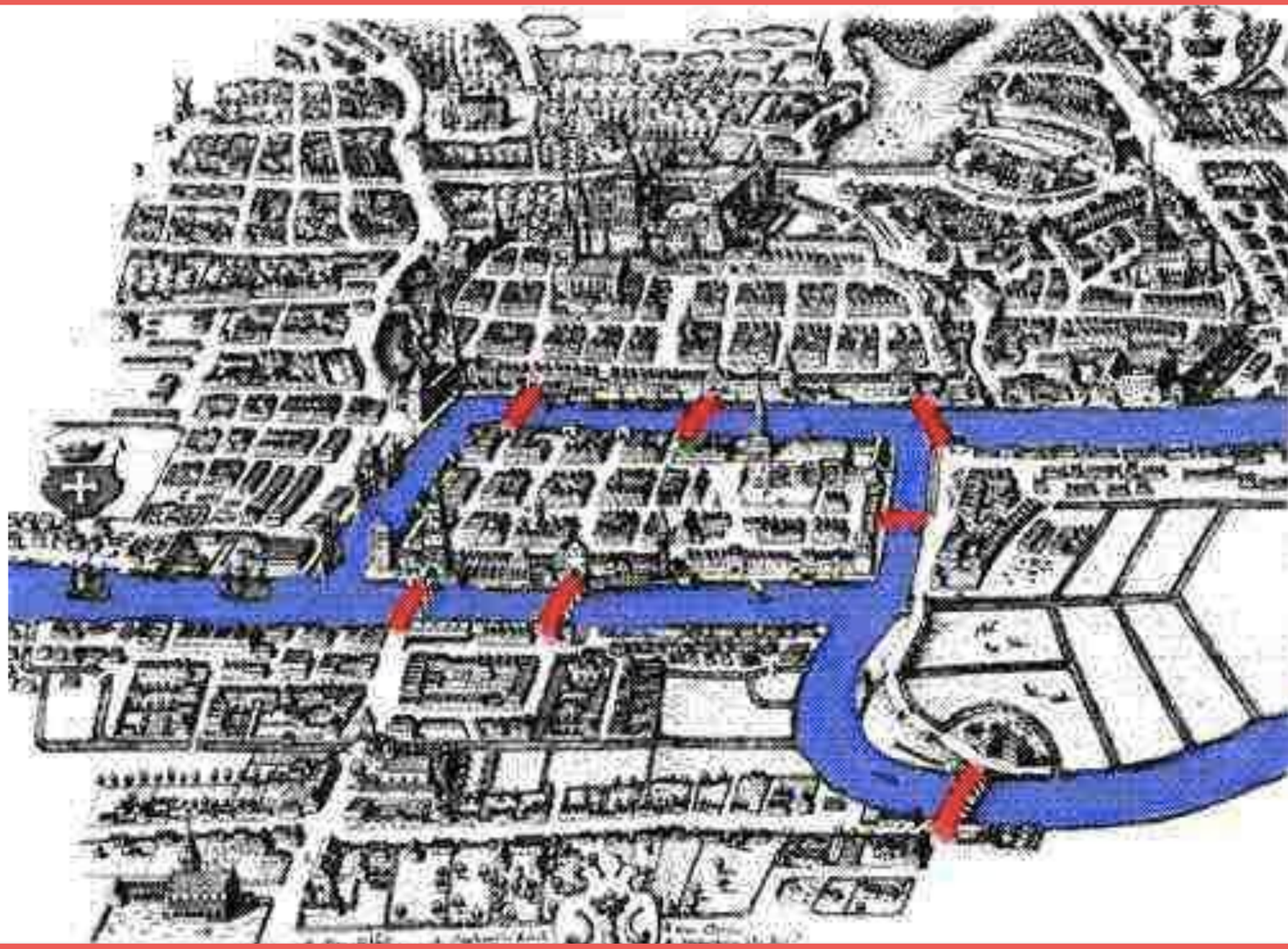
Graphs \o/

(Michelle)-[:LOVES]->(Neo4j)

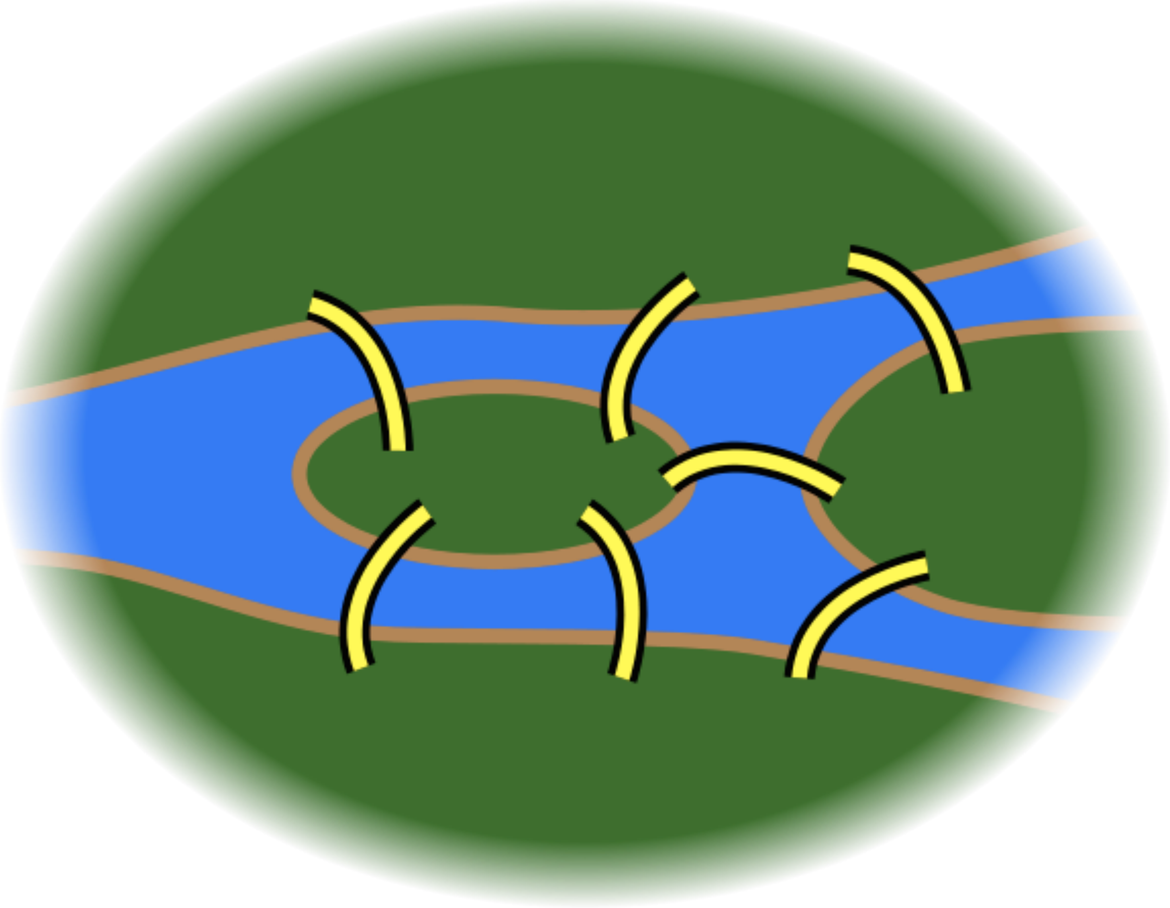
@michellesanver

Graphs \o/

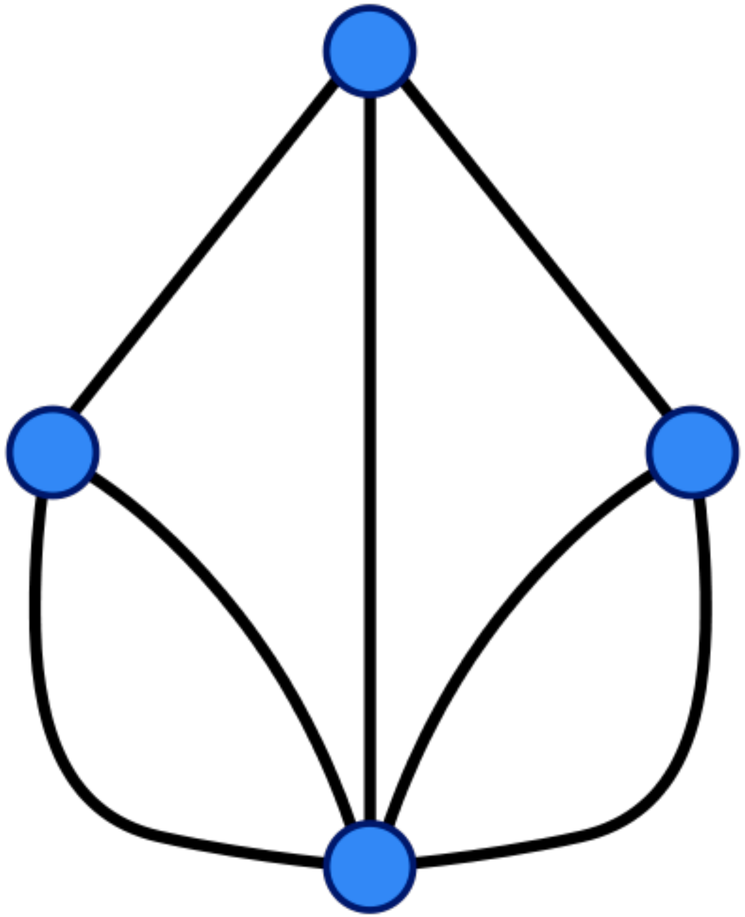
Graph theory has been studied since Leonard Euler's Bridges 1736



Graphs \o/



Graphs \o/



(Michelle)-[:LOVES]->(Neo4j)

@michellesanver

Graphs \o/

Graphs are really just connected data… They are everywhere

Graphs \o/

(Michelle)-[:LOVES]->(Neo4j)

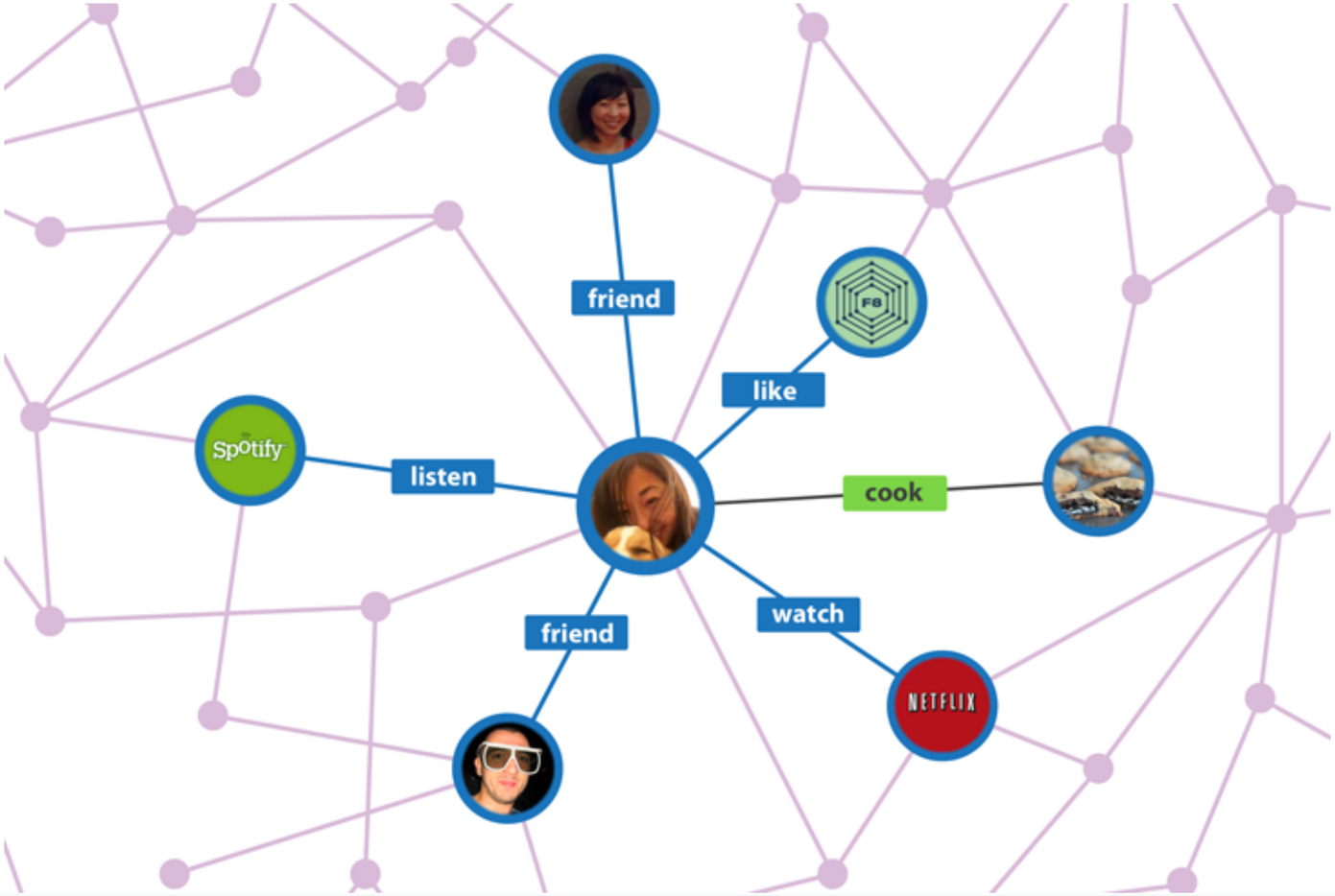
@michellesanver

Graphs \o/

It’s modern.

Graphs \o/

Facebook open graph

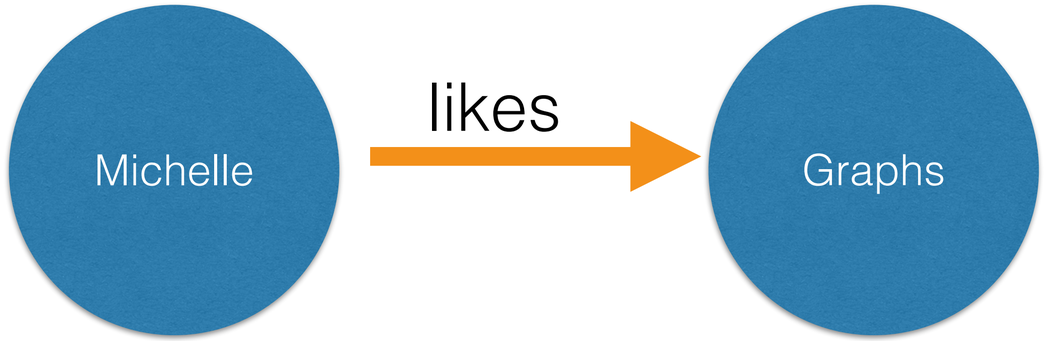


(Michelle)-[:LOVES]->(Neo4j)

@michellesanver

Graphs \o/

A graph is an easy way to visualise connected data.



Graphs vs. Relational databases

Relational question: Average age of everyone in this list?

Graph question: Who knows me from third parties?

Graphs vs. Relational databases

Relational databases have tables

Recipes

Graphs vs. Relational databases

Tables have relationships

Recipes

Groups

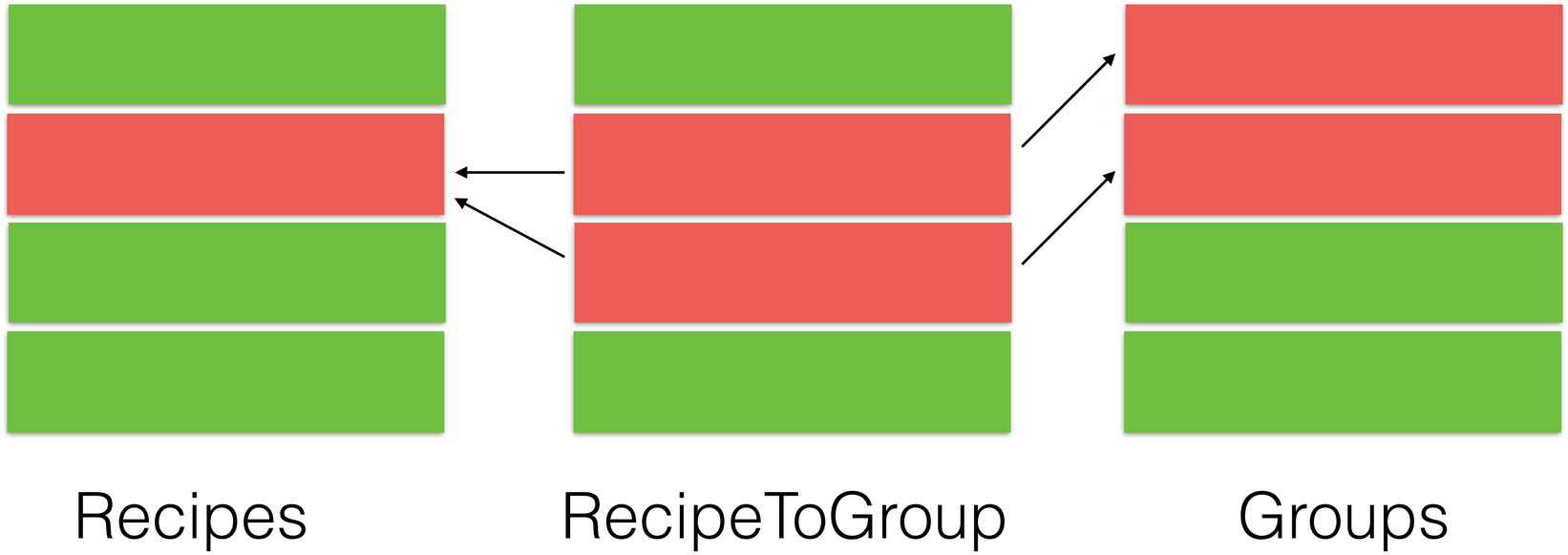
Graphs vs. Relational databases

Which causes a join table…

Recipes RecipeToGroup Groups

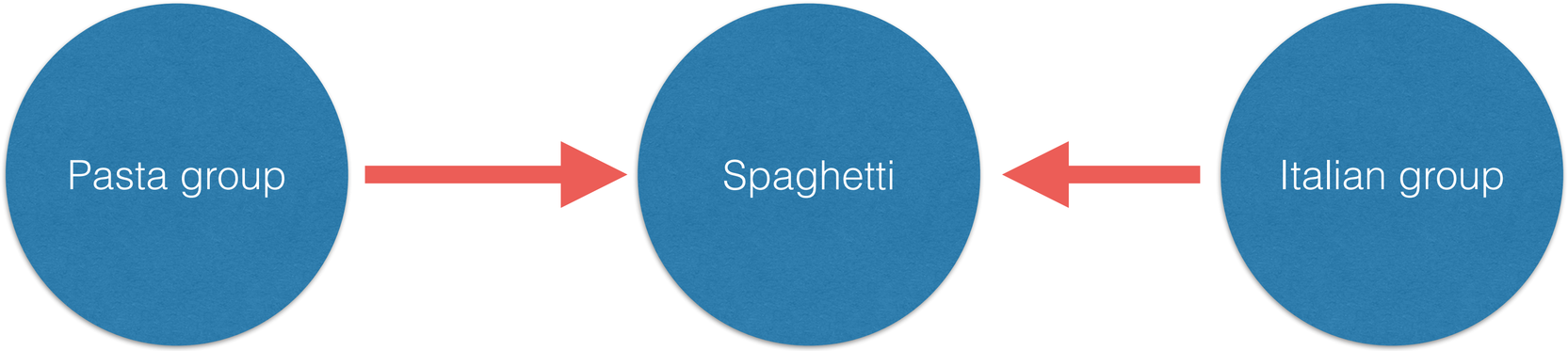
Graphs vs. Relational databases

You query via the table

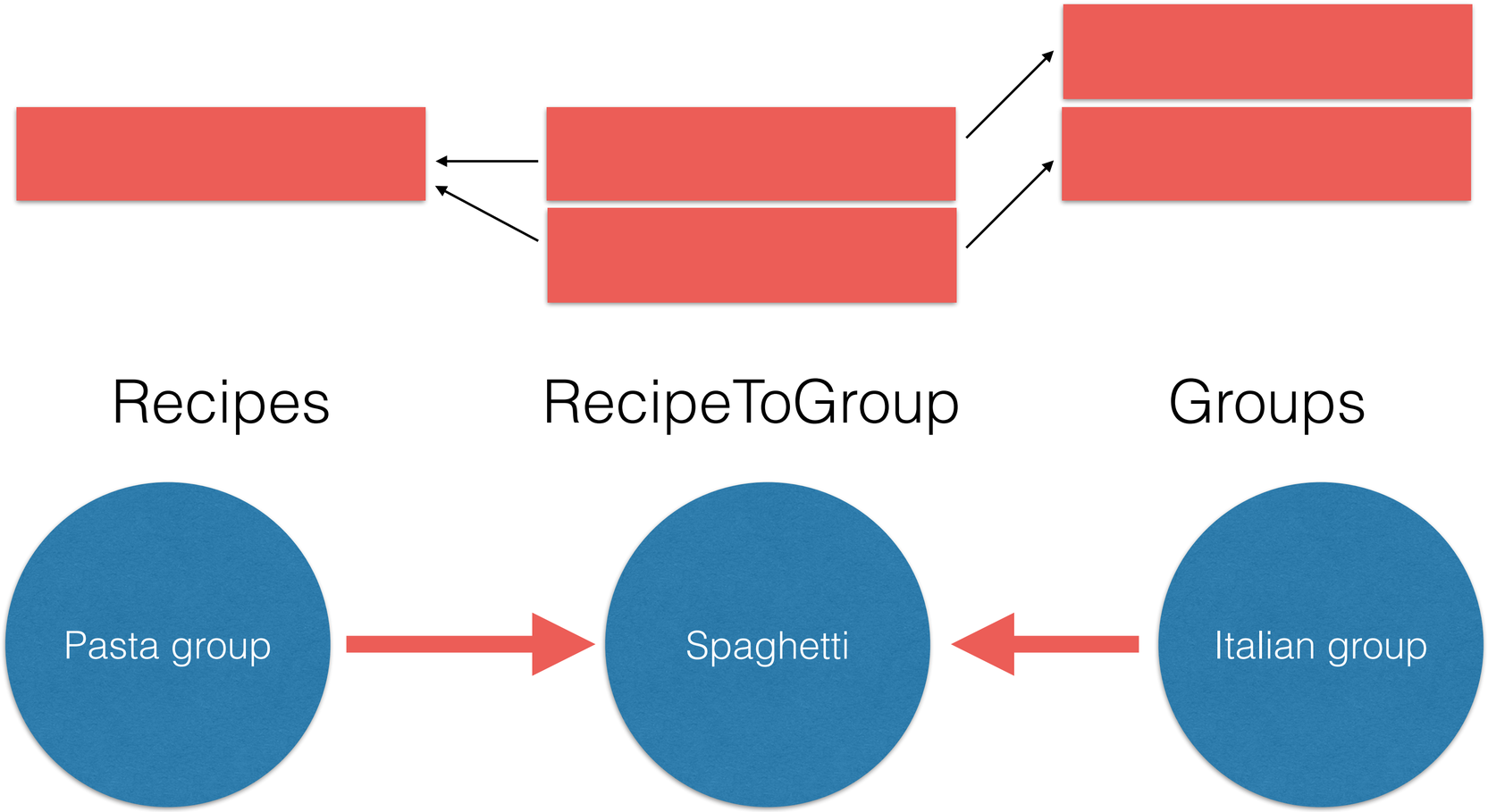


Graphs vs. Relational databases

Imagine having *actual* relations



Graphs vs. Relational databases



Graphs vs. Relational databases

A big query.

The EAV model.

Graphs vs. Relational databases

```
SET @entityid = '3';

SELECT ea.attribute_id,
ea.attribute_code, eav.value AS 'value',
'varchar' AS 'type'
FROM catalog_category_entity e
JOIN catalog_category_entity_varchar
eav
ON e.entity_id = eav.entity_id
JOIN eav_attribute ea
ON eav.attribute_id = ea.attribute_id
WHERE e.entity_id = @entityid
UNION
```

```
SELECT ea.attribute_id,
ea.attribute_code, eav.value AS
'value', 'int' AS 'type'
FROM catalog_category_entity
e
JOIN
catalog_category_entity_int eav
ON e.entity_id = eav.entity_id
JOIN eav_attribute ea
ON eav.attribute_id =
ea.attribute_id
WHERE e.entity_id = @entityid
UNION
```

Graphs vs. Relational databases

```
SELECT ea.attribute_id,
ea.attribute_code,
eav.value AS 'value',
'decimal' AS 'type'
FROM
catalog_category_entity e
JOIN
catalog_category_entity_d
ecimal eav
ON e.entity_id =
eav.entity_id
JOIN eav_attribute ea
ON eav.attribute_id =
ea.attribute_id
WHERE e.entity_id =
@entityid
UNION
```

```
SELECT ea.attribute_id, ea.attribute_code,
eav.value AS 'value', 'datetime' AS 'type'
FROM catalog_category_entity e
JOIN catalog_category_entity_datetime eav
ON e.entity_id = eav.entity_id
JOIN eav_attribute ea
ON eav.attribute_id = ea.attribute_id
WHERE e.entity_id = @entityid
UNION
SELECT ea.attribute_id, ea.attribute_code,
eav.value AS 'value', 'text' AS 'type'
FROM catalog_category_entity e
JOIN catalog_category_entity_text eav
ON e.entity_id = eav.entity_id
JOIN eav_attribute ea
ON eav.attribute_id = ea.attribute_id
WHERE e.entity_id = @entityid
```

Graphs vs. Relational databases

Having a flexible schema database costs a lot.

Neo4j

(Michelle)-[:LOVES]->(Neo4j)

@michellesanver

neo4j.org

(Michelle)-[:LOVES]->(Neo4j)

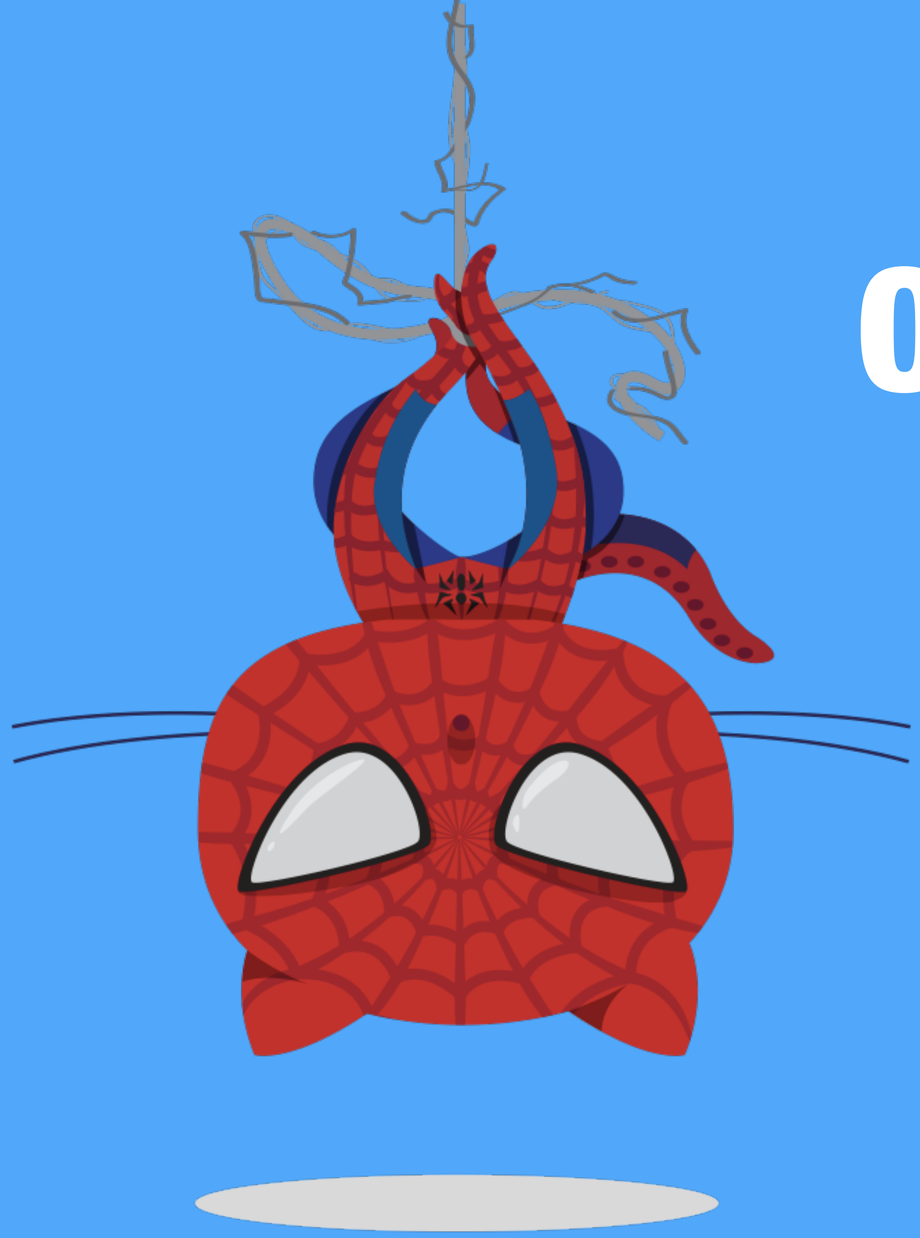
@michellesanver

neotechnology.com

(Michelle)-[:LOVES]->(Neo4j)

@michellesanver

Java Based



Open Source!

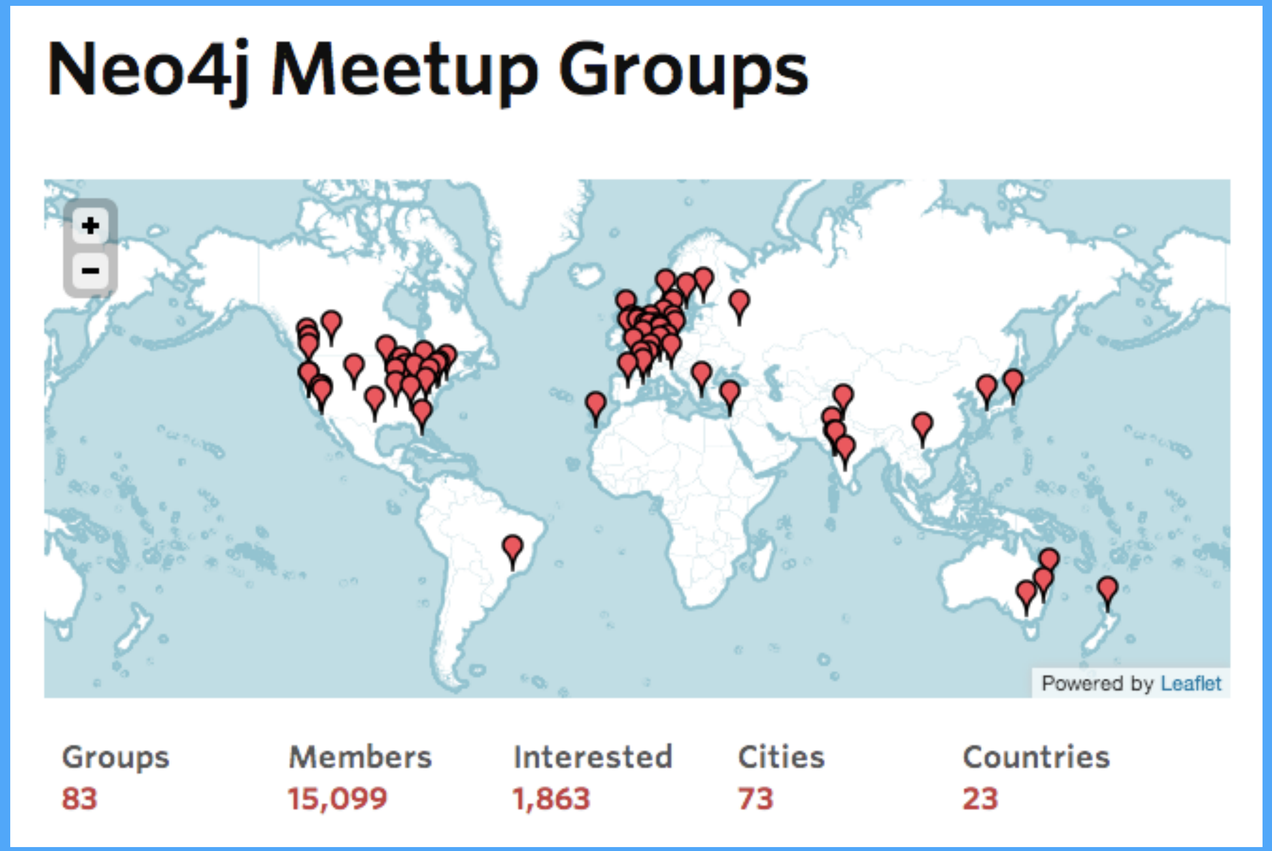
github.com/neo4j

(Michelle)-[:LOVES]->(Neo4j)

@michellesanver

Meetups everywhere

neo4j.meetup.com



(Michelle)-[:LOVES]->(Neo4j)

@michellesanver

Graphs and Neo4j

Graphs have…

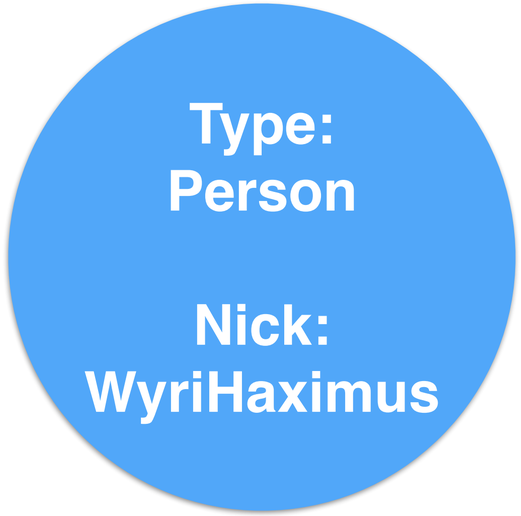
Graphs and Neo4j

(Nodes)

Graphs and Neo4j

(Node) { Properties }

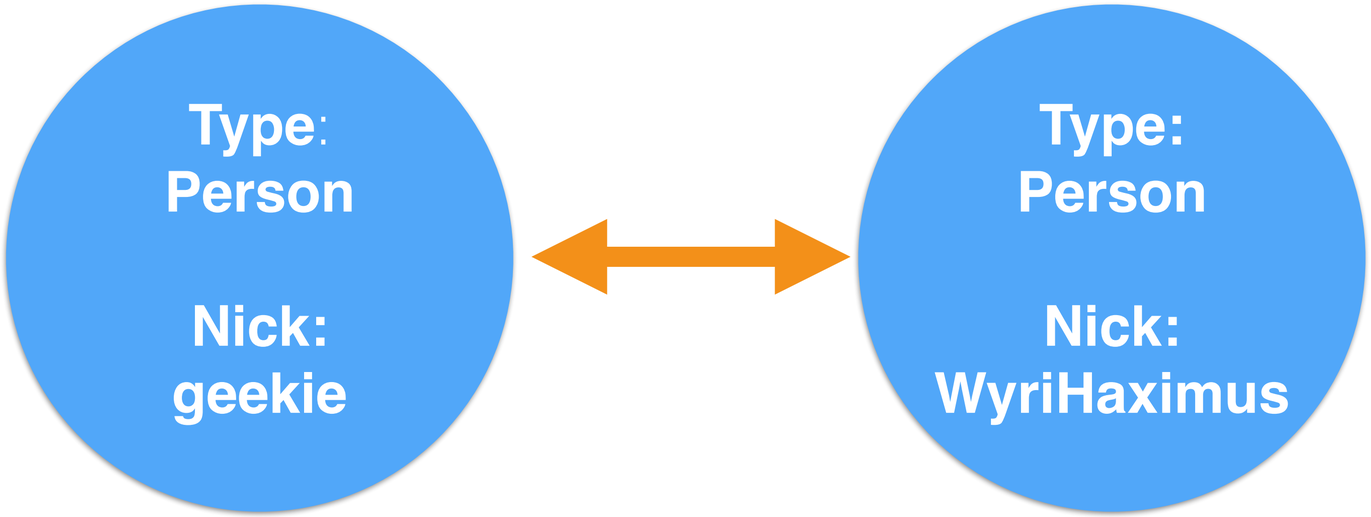
As many as you want



Graphs and Neo4j

(Node) [Relationships]

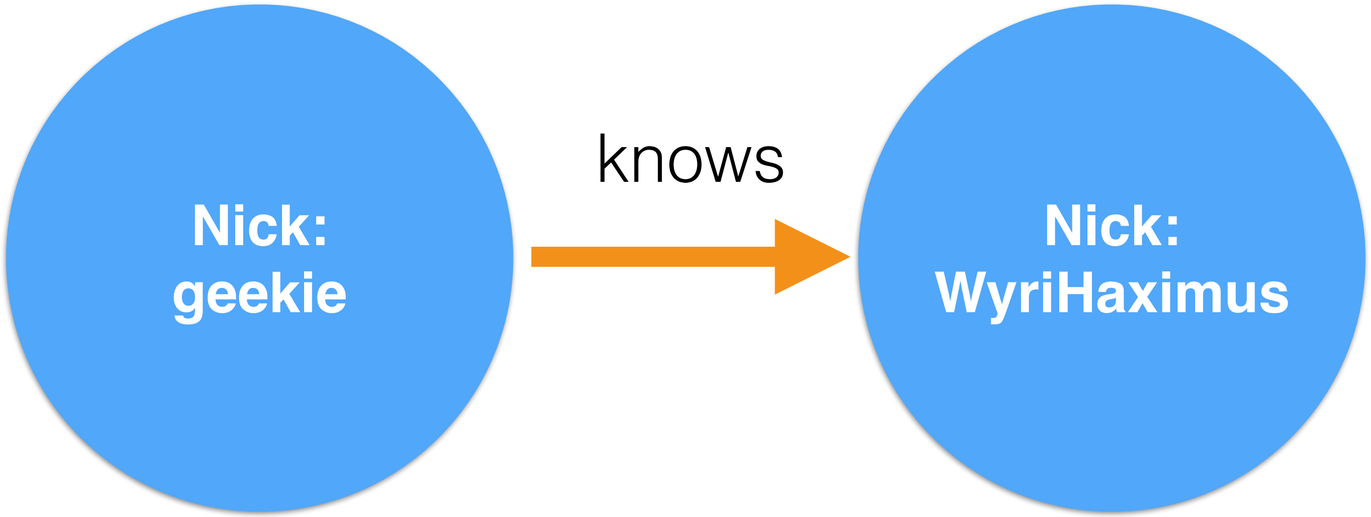
As many as you want



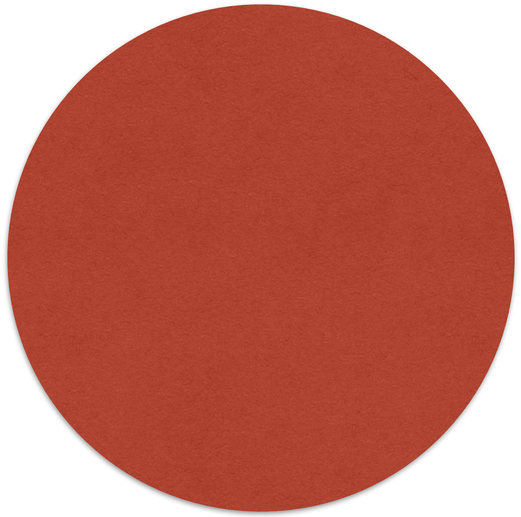
Graphs and Neo4j

[Relationship { properties }]

As many as you want



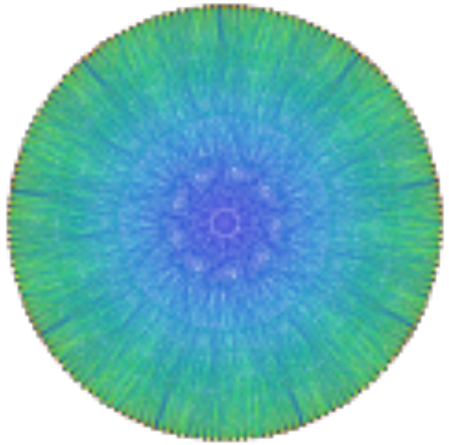
Labels



Indexes for easy lookup

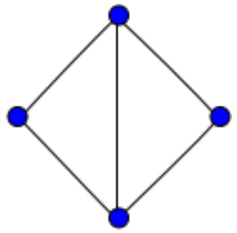
Common named graphs

Common named graphs

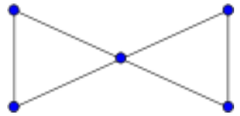


Local McLaughlin graph

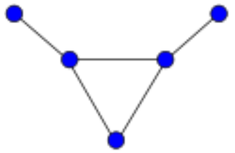
Common named graphs



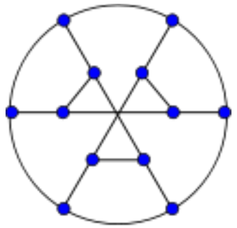
Diamond



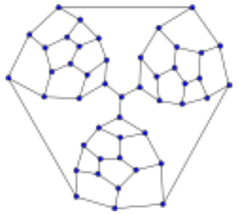
Butterfly



Bull



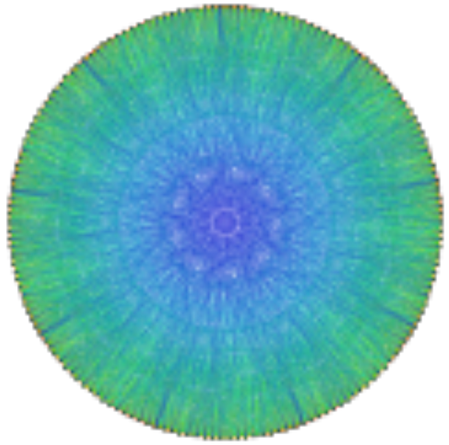
Franklin

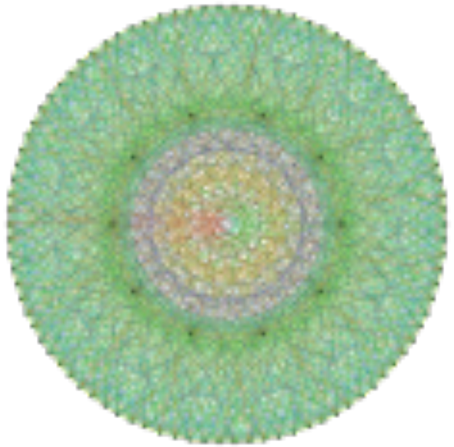


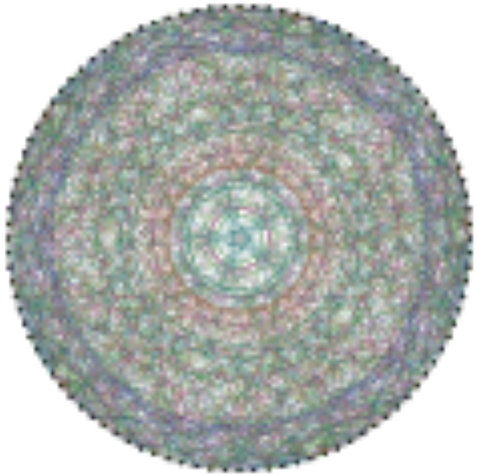
Tutte

You can make art out of your DB.

(Don't)







OmNomHub

https://github.com/Omnomhub

(Michelle)-[:LOVES]->(Neo4j)

@michellesanver

OmNomHub

Like GitHub but for recipes!

Fork a recipe

See all forks

Join groups

Search similar recipes

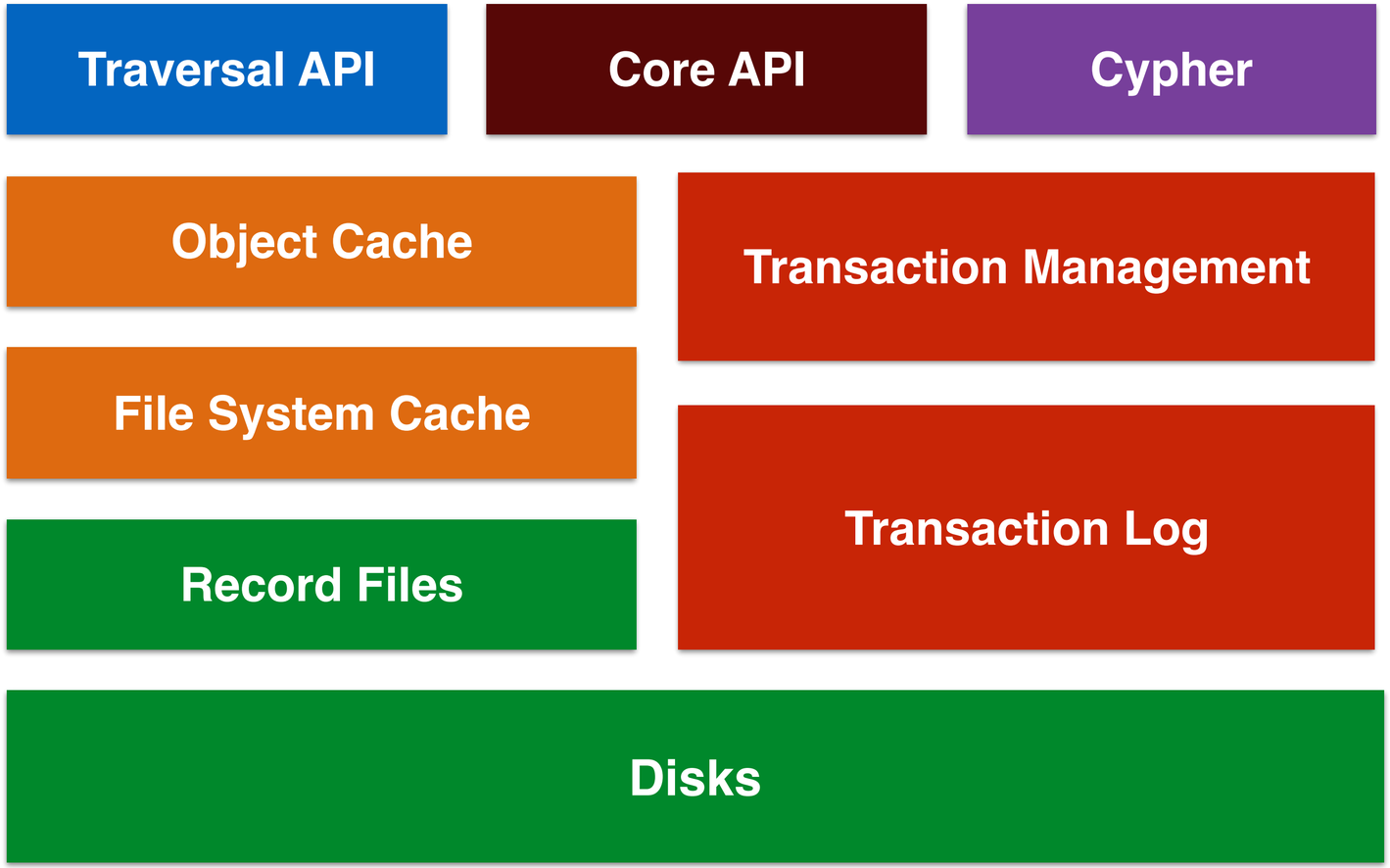
Have fun! ;-)

Neo4j architecture

(Michelle)-[:LOVES]->(Neo4j)

@michellesanver

Neo4j architecture



Hang in there

(Michelle)-[:LOVES]->(Kittens)

@michellesanver

Neo4j architecture: Traversals

Traversals

Pathexpanders: Define what to traverse

Traversals

Order: Depth-first or breadth-first.

Traversals

Uniqueness: Visit once.

Traversals

Evaluator: Should I stay or should I go?

Traversals

Starting nodes: And so it begins.

Neo4j architecture: Cypher and Browser

(Michelle)-[:LOVES]->(Neo4j)

@michellesanver

Cypher

Cypher: SQL for graph databases.

Cypher

(Parentheses means nodes)

(Or hugs)

Cypher

{ Curly braces means properties }

Cypher

[Square brackets means relationships]

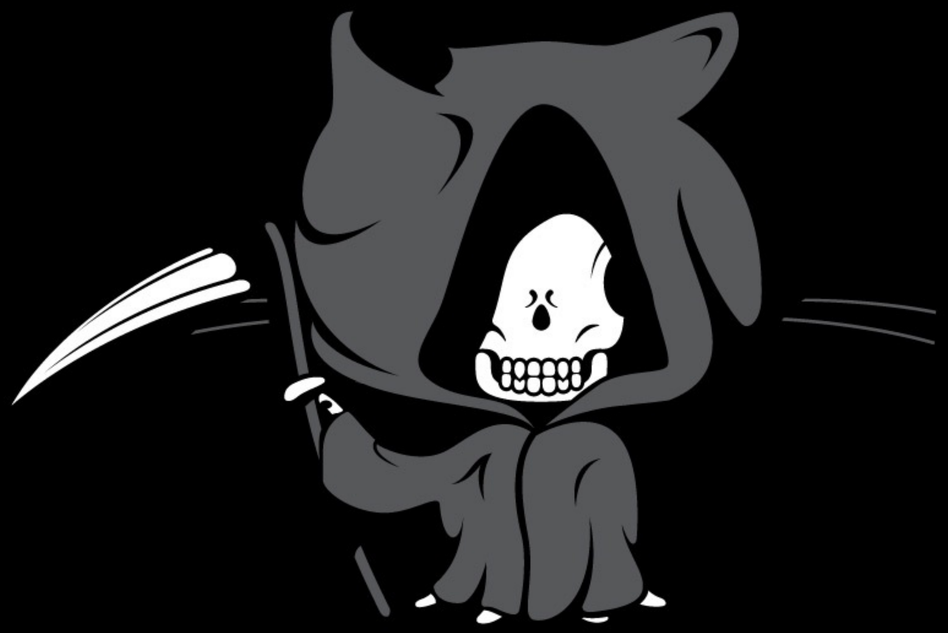
Browser

Makes it easy to visualise and query the data.

Browser

Let’s learn cypher in the browser!

DEMO: Cypher and Browser



(Michelle)-[:LOVES]->(Neo4j)

@michellesanver

Neo4j architecture
Neostore File Storage

Neostore

Several different store files

Neostore

Each store has specific data

Neostore

Nodes

neostore.nodestore.db

9 bytes

Neostore

Nodes

in-use key: 1 byte

Neostore

Fixed size == FAST

Neostore

Relationships

neostore.relationshipstore.db

33 bytes

Neostore

Properties

neostore.propertystore.db

neostore.propertystore.db.index

neostore.propertystore.db.strings

neostore.propertystore.db.arrays

Hardware matters!

Cache

Two-tiered Cache

Cache

Filesystem cache

Cache

Filesystem cache

Can be fine-tuned

Object Cache

Properties and references to their relationships

OmNomHub: Beyond Pizza

(Michelle)-[:LOVES]->(Neo4j)

@michellesanver

Connecting users

"You both like"

Connecting recipes

"These have similar ingredients and user base"

Being smart

"You might like"

Being smart

Smart recipe collections!

Being creepy

"Don't like meat huh?"

Wrapup

Wrapup

Graphs are everywhere

Wrapup

They make it easy to connect data

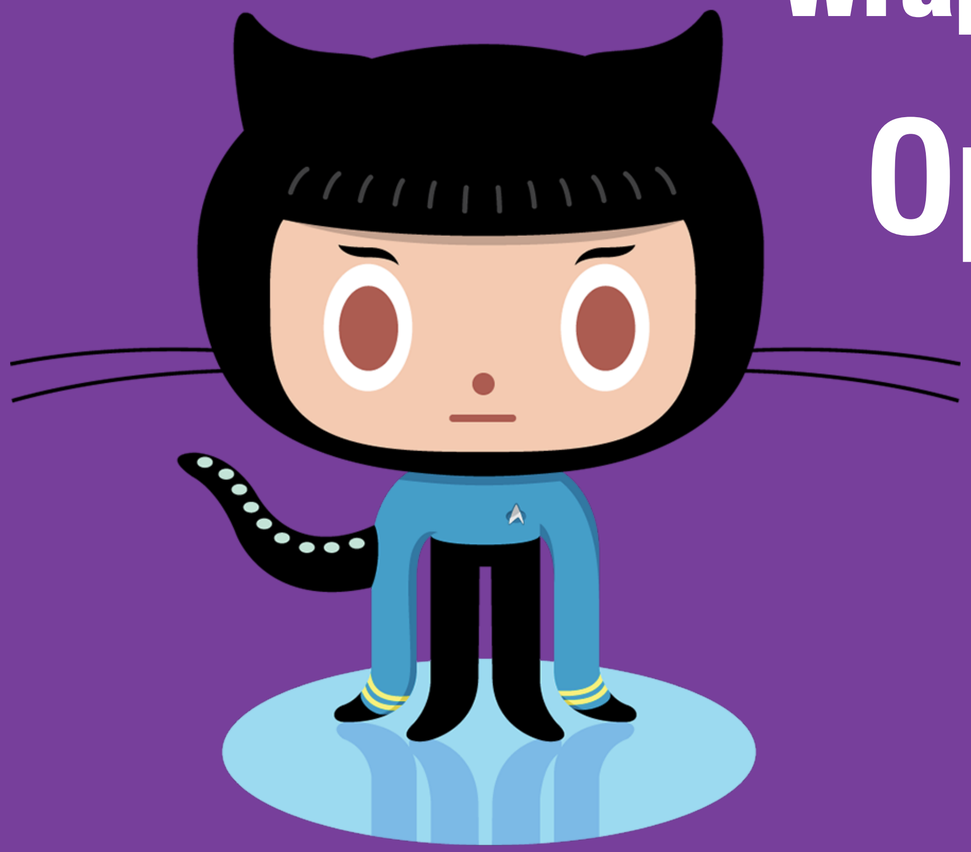
Wrapup

Easier to be creepy <3

Wrapup

OmNomHub will be **EPIC :D**

Wrapup



Open Source!

https://github.com/Omnomhub

Resources

docs.neo4j.org

Graph Databases - Ian Robinson,
neo technology gives you the e-book version for free.

neo4j.org/learn

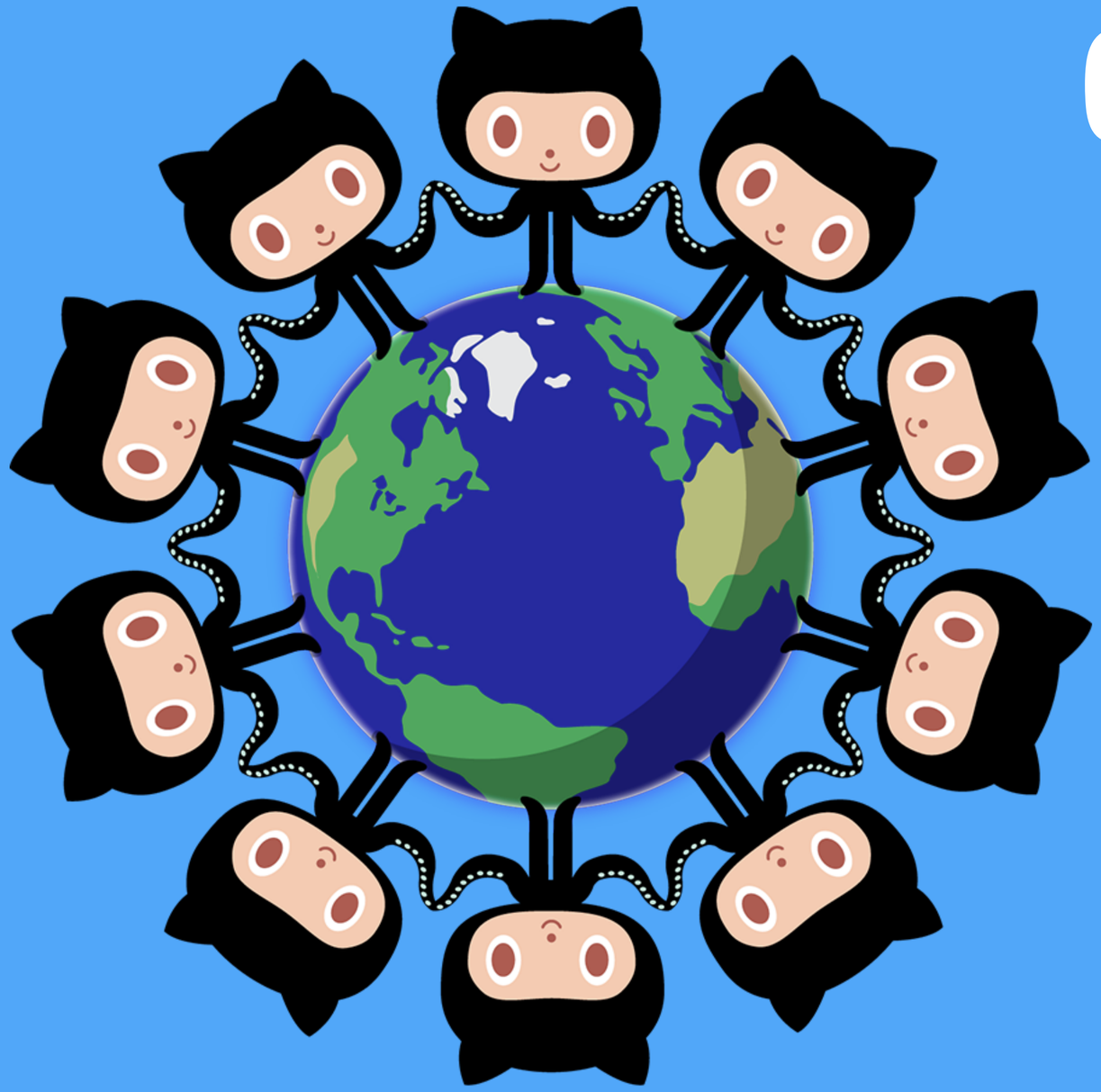
Pleaaaaase....

Please

(Michelle)-[:LOVES]->(Neo4j)

@michellesanver

Please



Give me feedback

https://joind.in/12745

(Michelle)-[:LOVES]->(Neo4j)

@michellesanver

Thank you

NoSQL Day 2014

(We)-[:LOVE]->(Neo4j)

@michellesanver

Questions?

(Michelle)-[:LOVES]->(Neo4j)

@michellesanver

How would you use it?

(Michelle)-[:LOVES]->(Neo4j)

@michellesanver