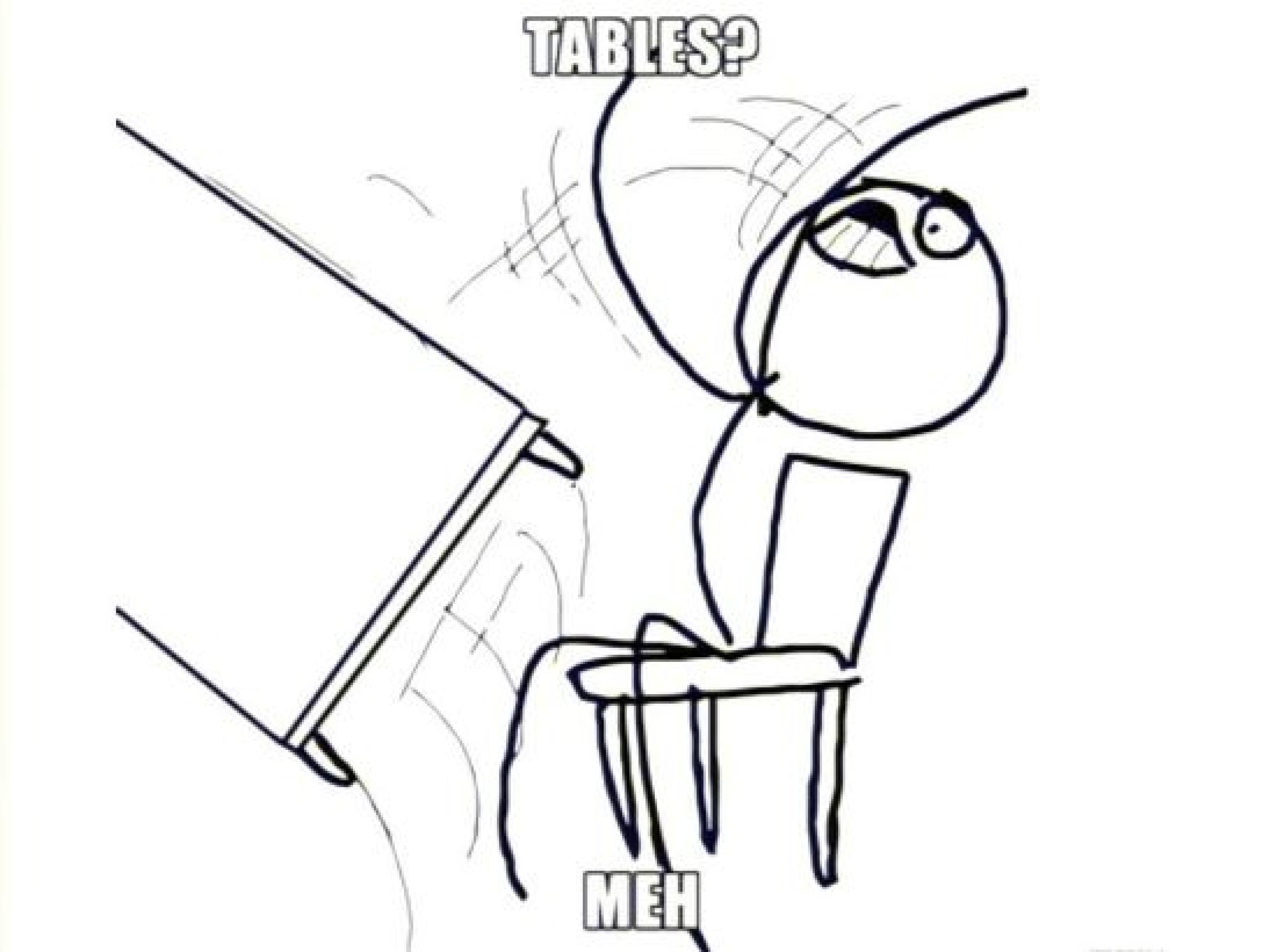
A Little Graph Theory for the Busy Developer

Stefan Armbruster
Field Engineer, Neo Technology
@darthvader42
(slides from Jim Webber, @jimwebber)

Roadmap

- Imprisoned data
- Graph models
- Graph theory
  - Local properties, global behaviours
  - Predictive techniques
- Graph matching
  - Predictive, real-time analytics for fun and profit
- Fin


TABLES?
MEH

ACCUTIRE
THE KEY TO
MADE IN CANADA
Jeep

http://gallery.nen.gov.uk/image82582-.html

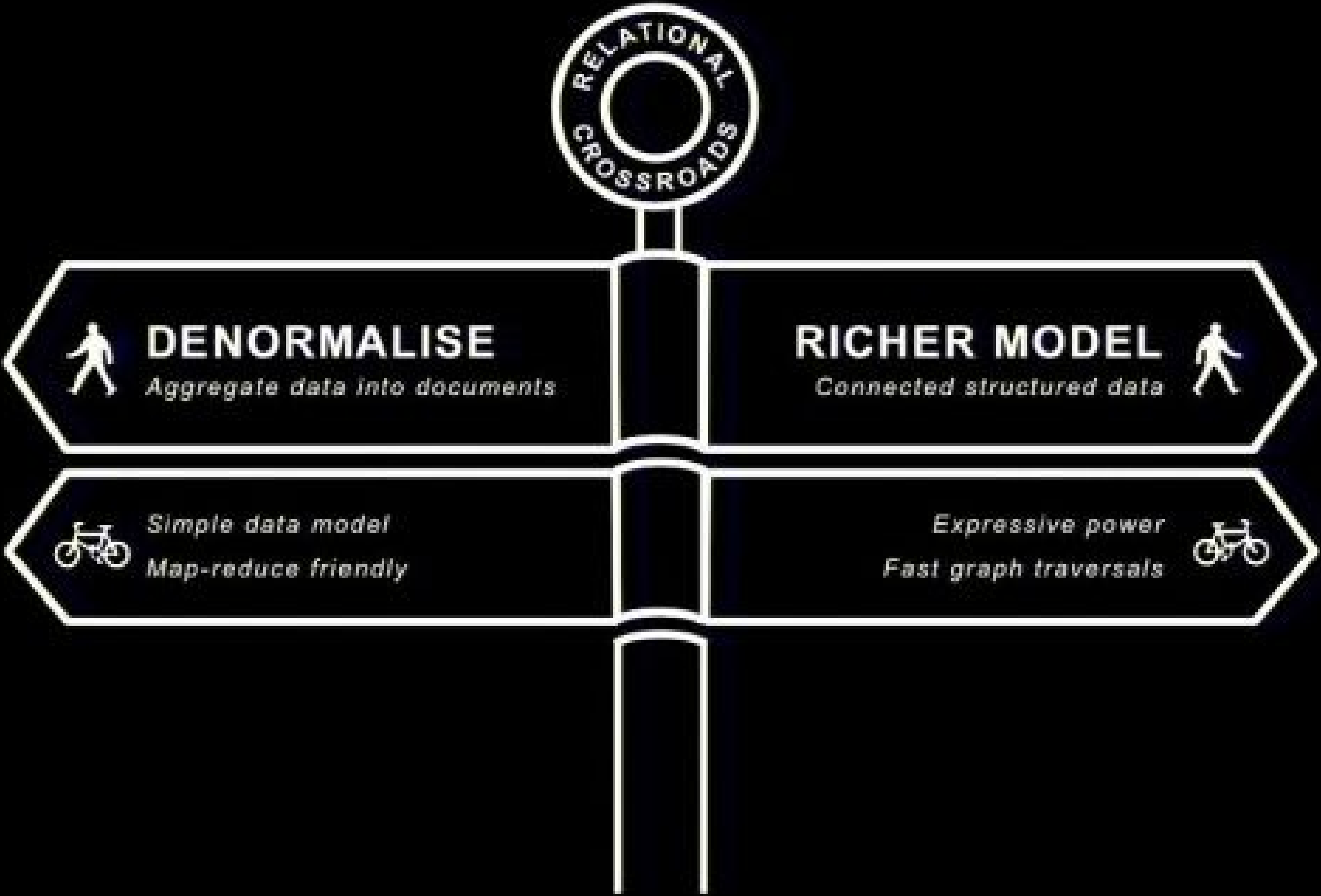
http://www.xtranormal.com/watch/6995033/mongo-db-is-web-scale

Aggregate-Oriented Data

http://martinfowler.com/bliki/AggregateOrientedDatabase.html

"There is a significant downside - the whole approach works really well when data access is aligned with the aggregates, but what if you want to look at the data in a different way? Order entry naturally stores orders as aggregates, but analyzing product sales cuts across the aggregate structure. The advantage of not using an aggregate structure in the database is that it allows you to slice and dice your data different ways for different audiences.

This is why aggregate-oriented stores talk so much about map-reduce."


RELATIONAL CROSSROADS
DENORMALISE
Aggregate data into documents
Simple data model
Map-reduce friendly
RICHER MODEL
Connected structured data
Expressive power
Fast graph traversals

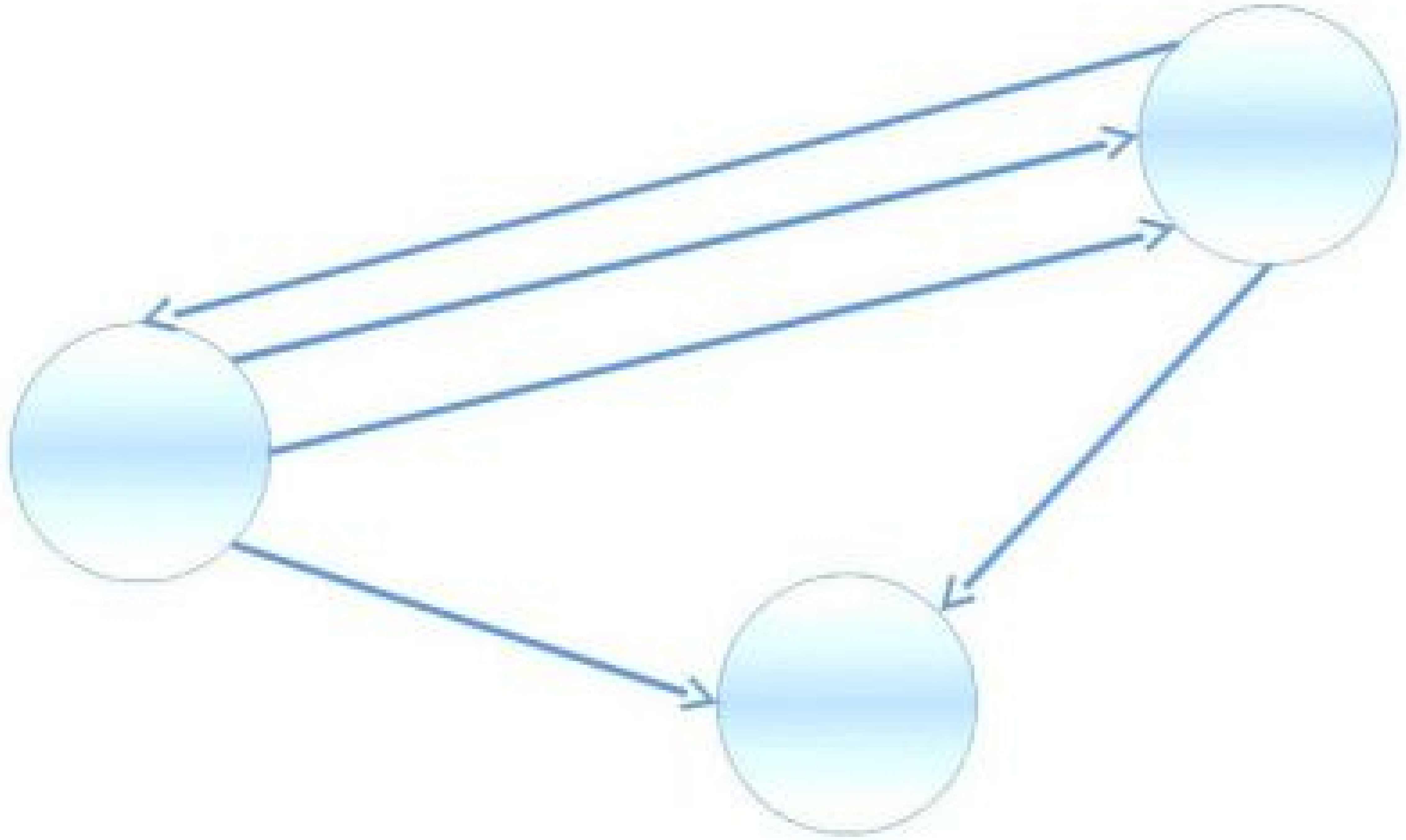
$$complexity = f(size, connectedness, uniformity)$$

http://www.bbc.co.uk/london/travel/downloads/tube_map.html

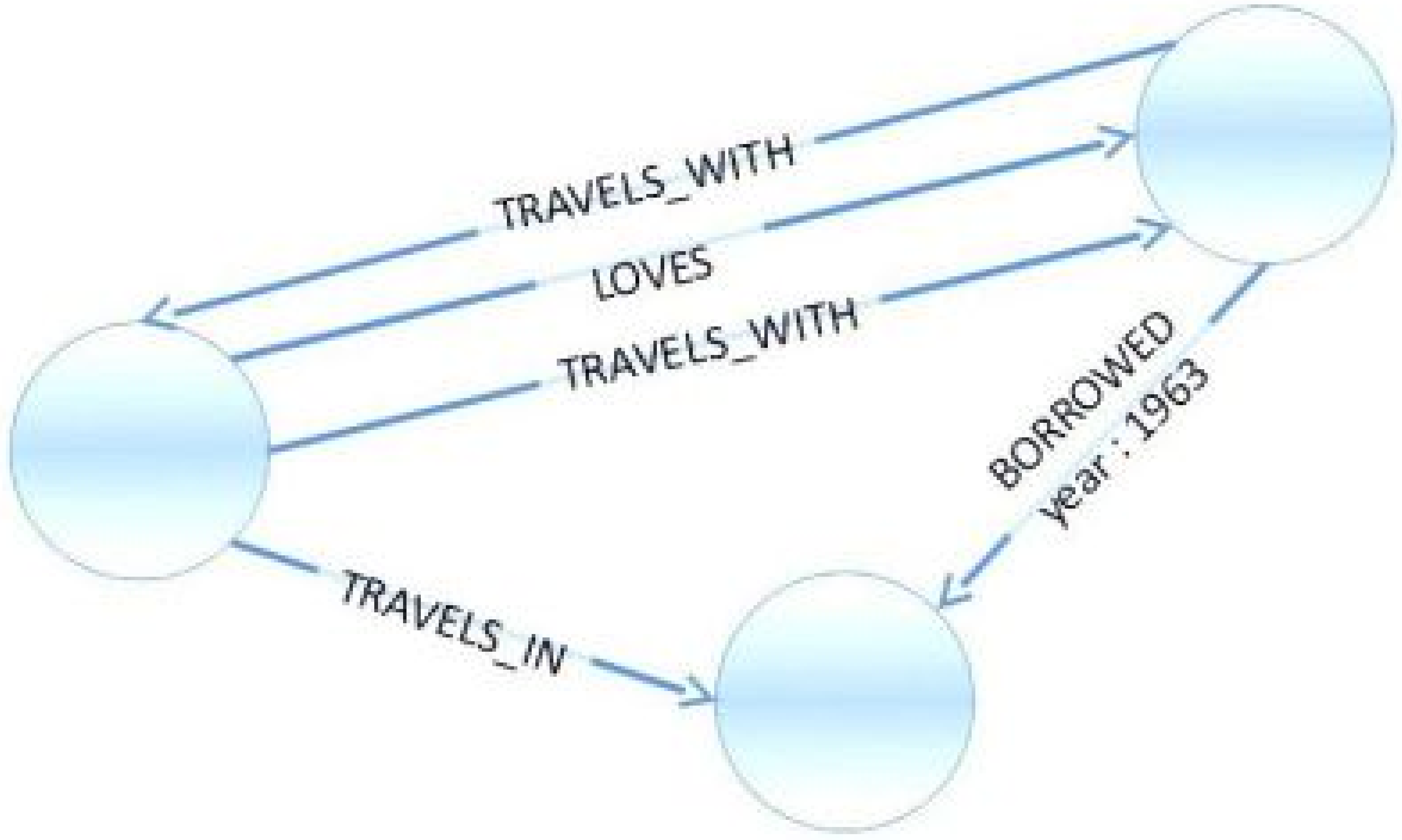
Property graphs

- Property graph model:
  - Nodes with properties
  - Named, directed relationships with properties
  - Relationships have exactly one start and end node
    - Which may be the same node

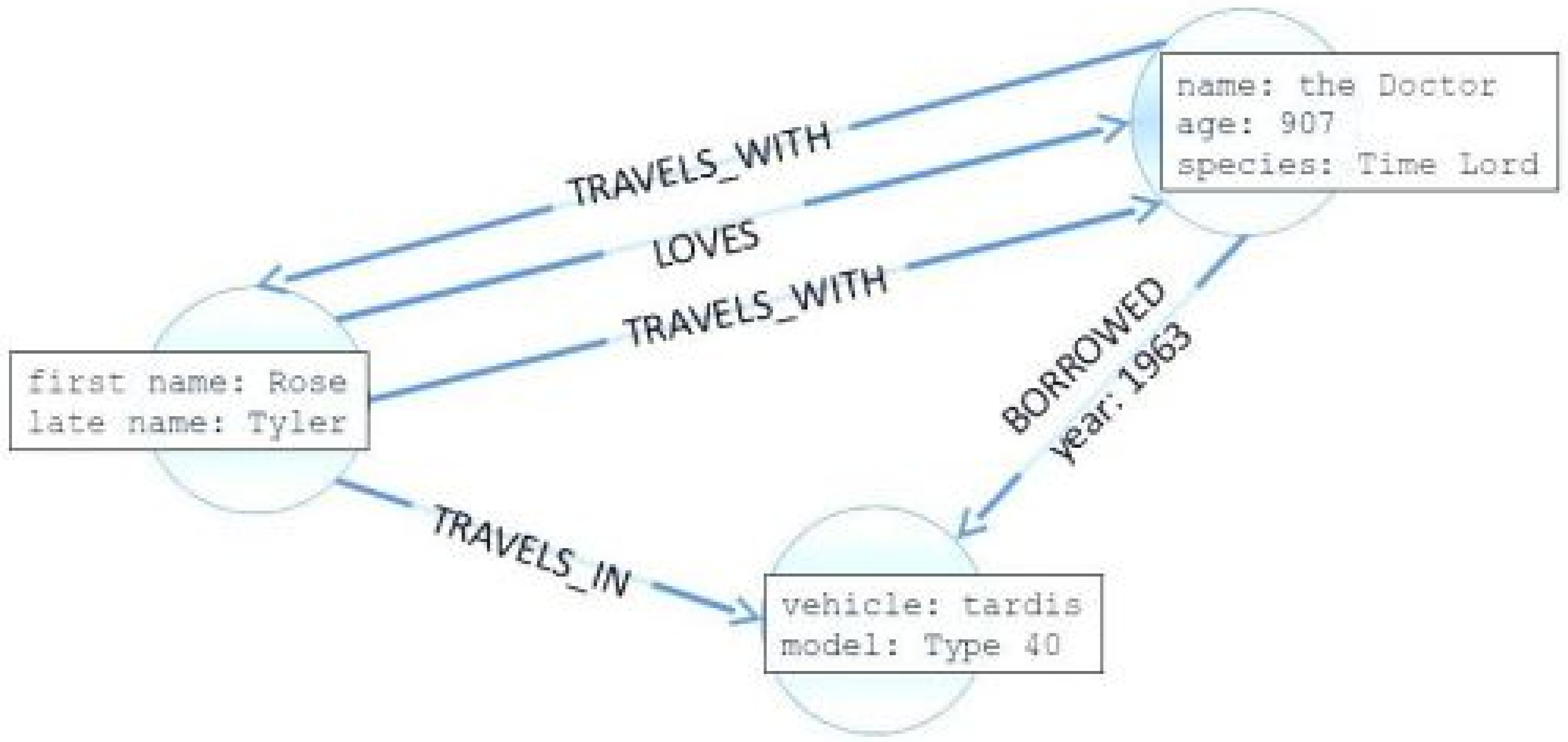
Property Graph Model



Property Graph Model

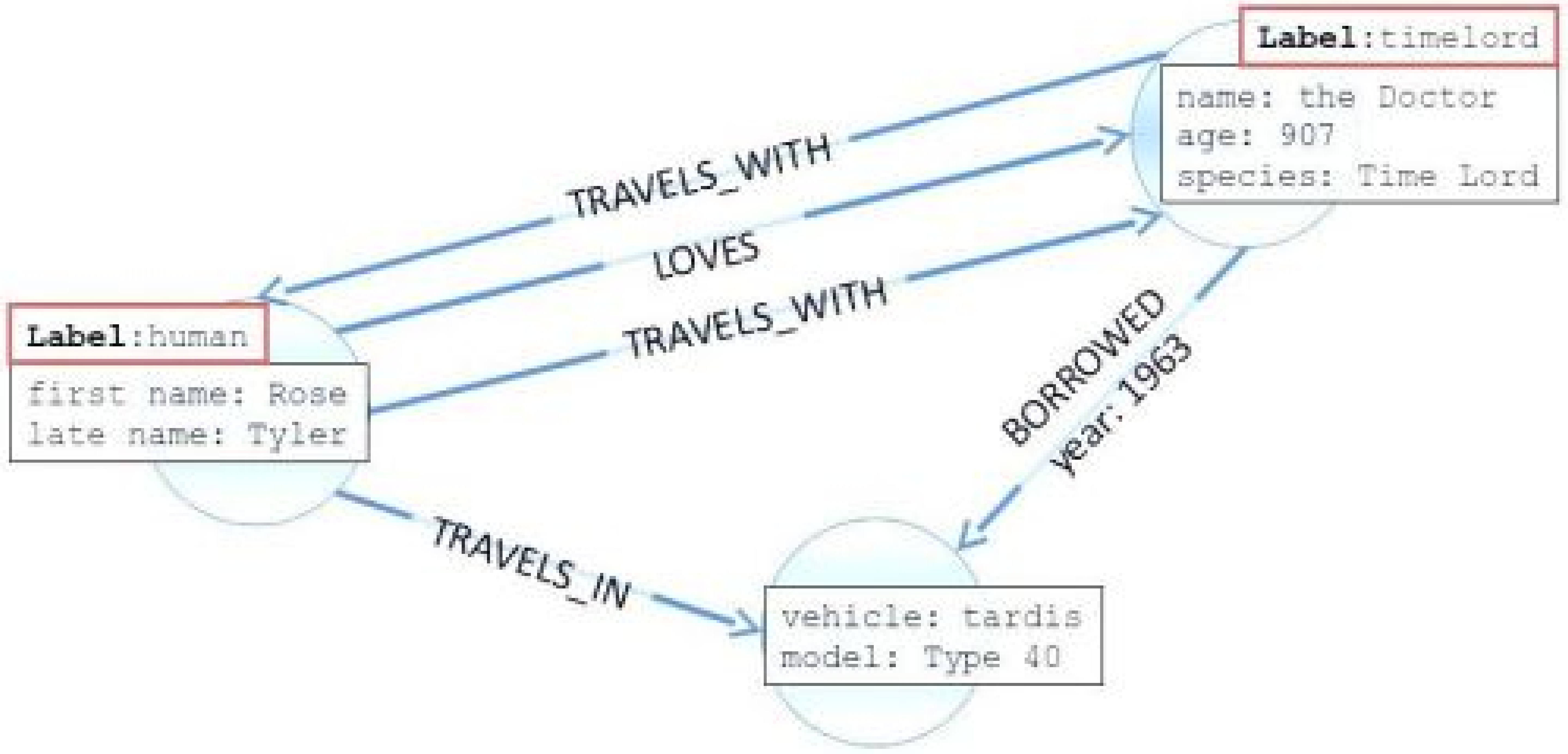


Property Graph Model

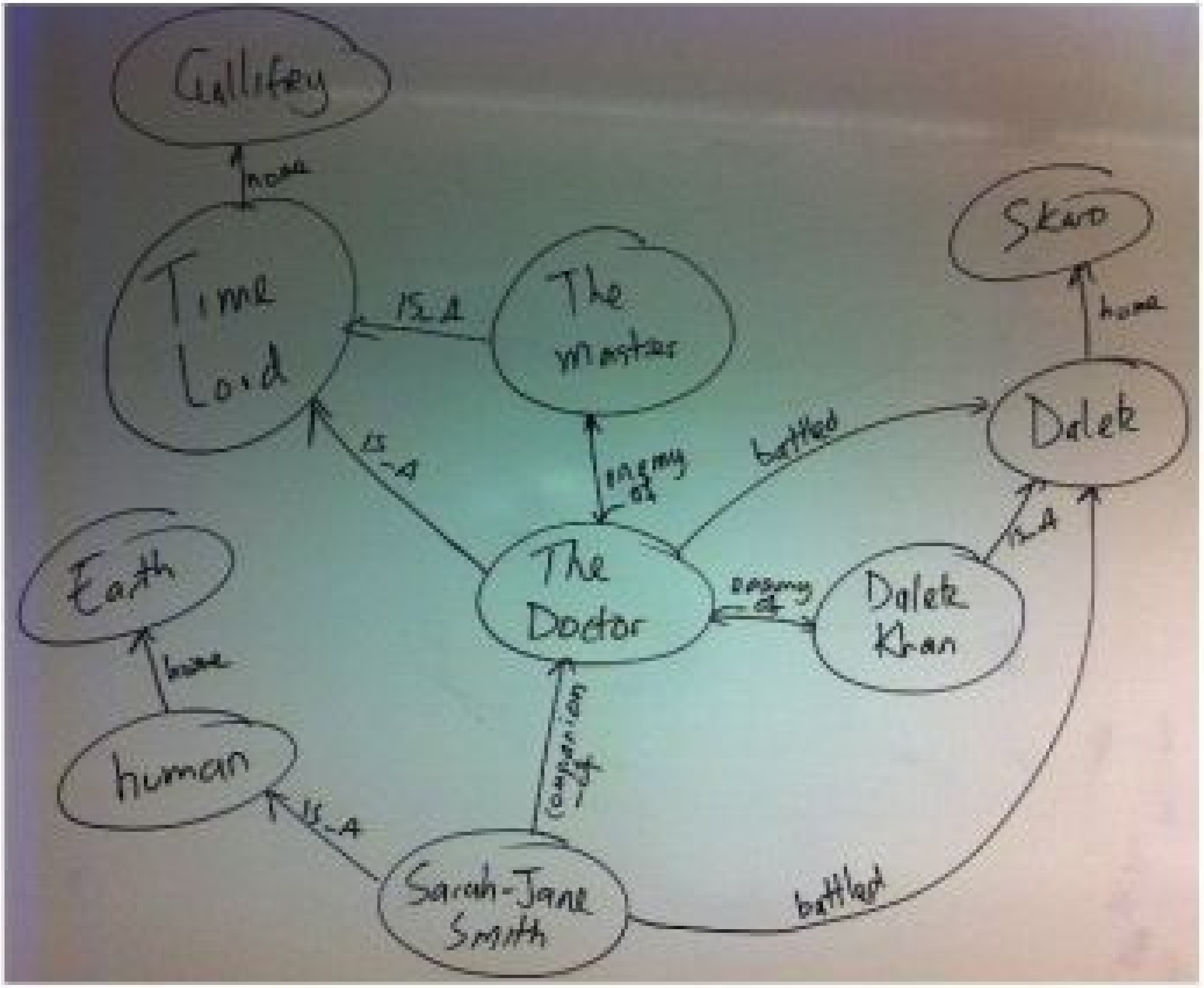


Labeled Property Graph Model

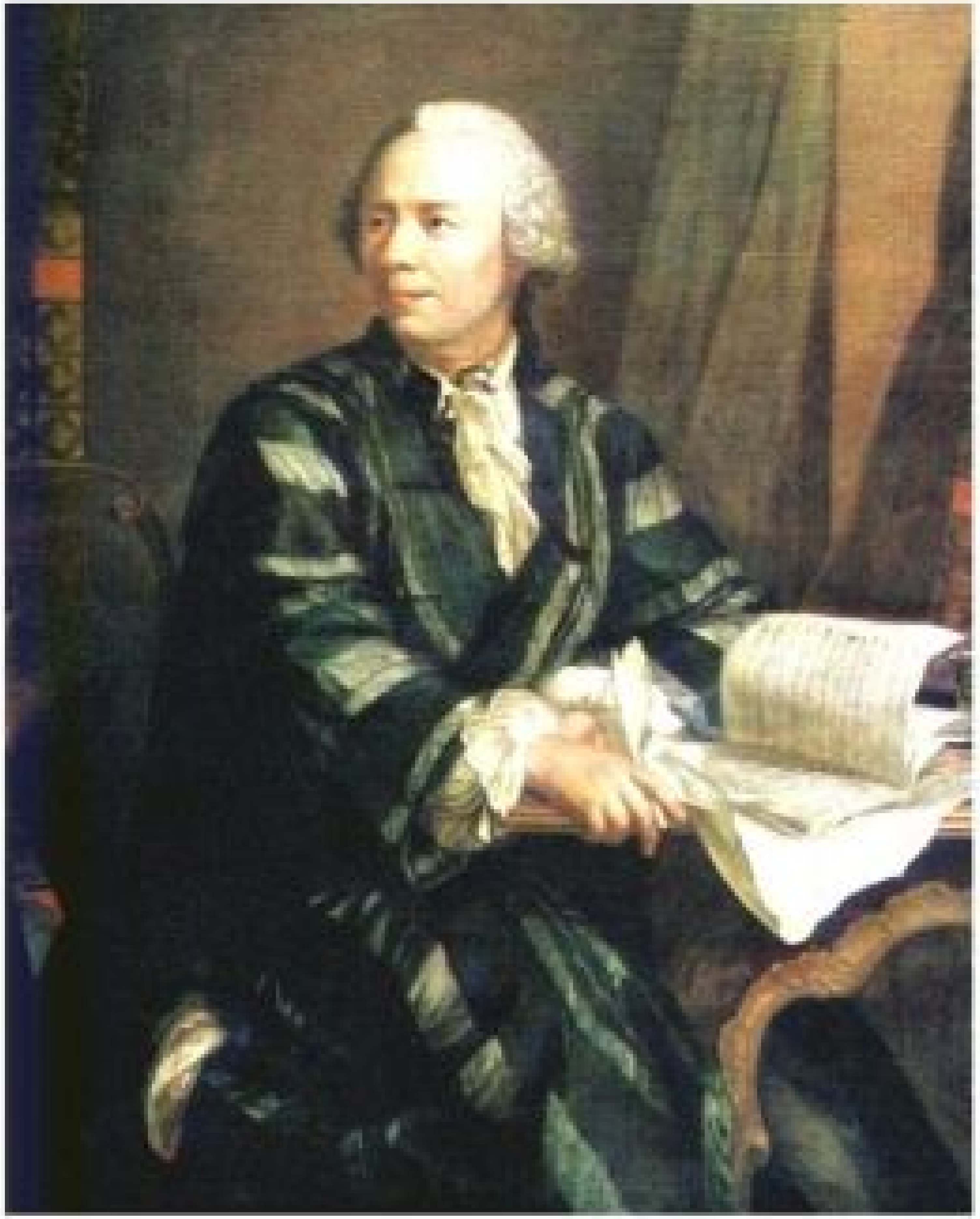
(Neo4j 2.0)



Property graphs are *very* whiteboard-friendly

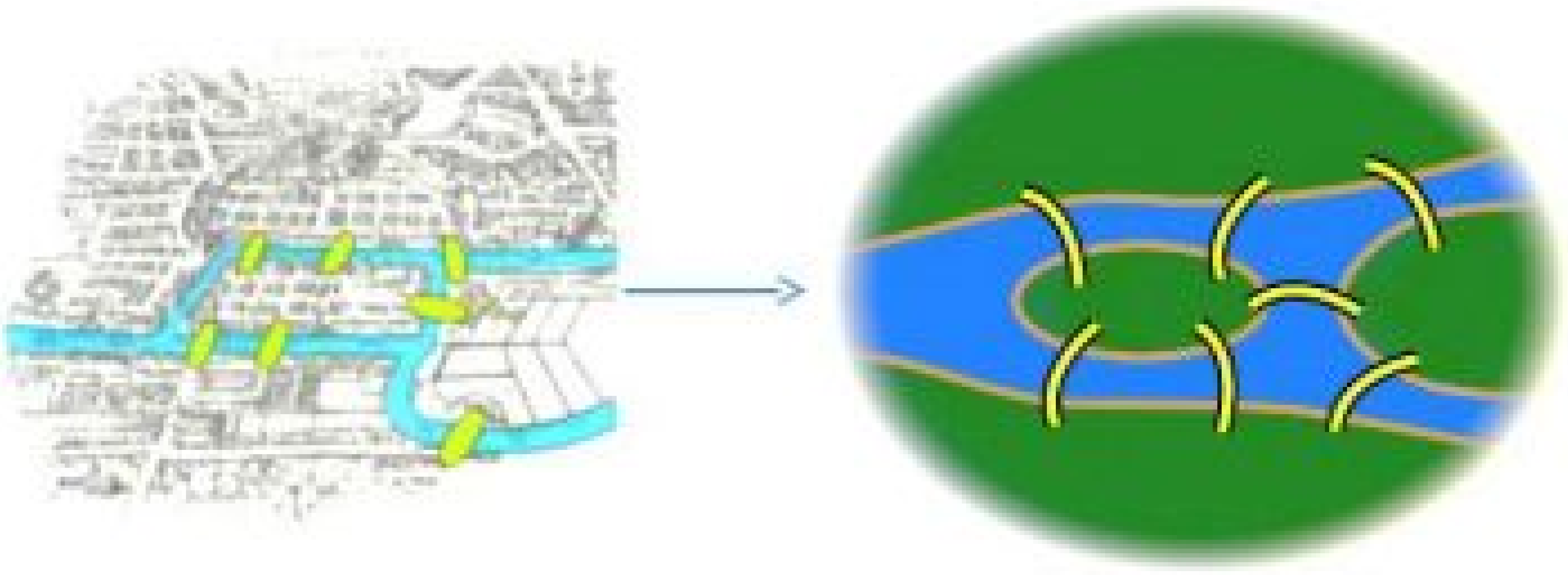


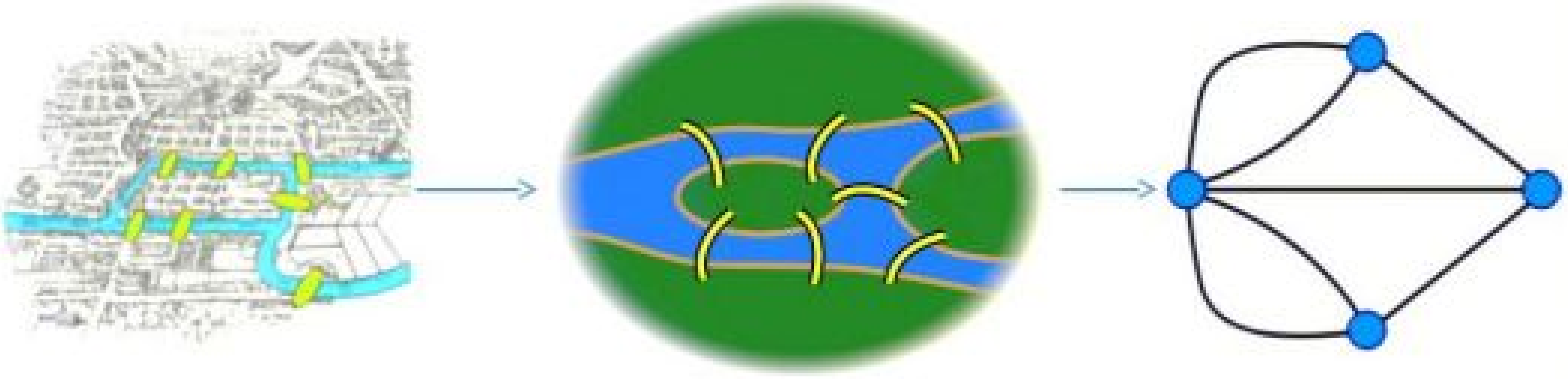
http://blogs.adobe.com/digitalmarketing/analytics/predictive-analytics/predictive-analytics-and-the-digital-marketer/



Meet Leonhard Euler

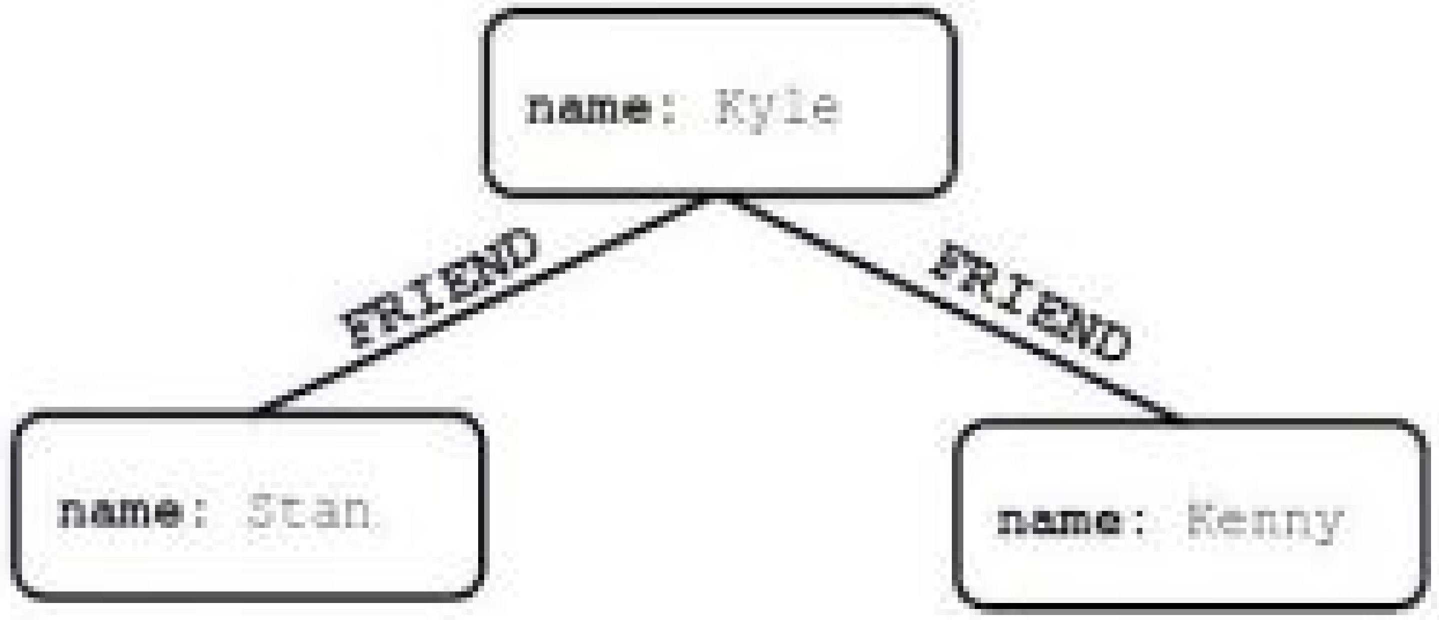
- Swiss mathematician
- Inventor of Graph Theory (1736)



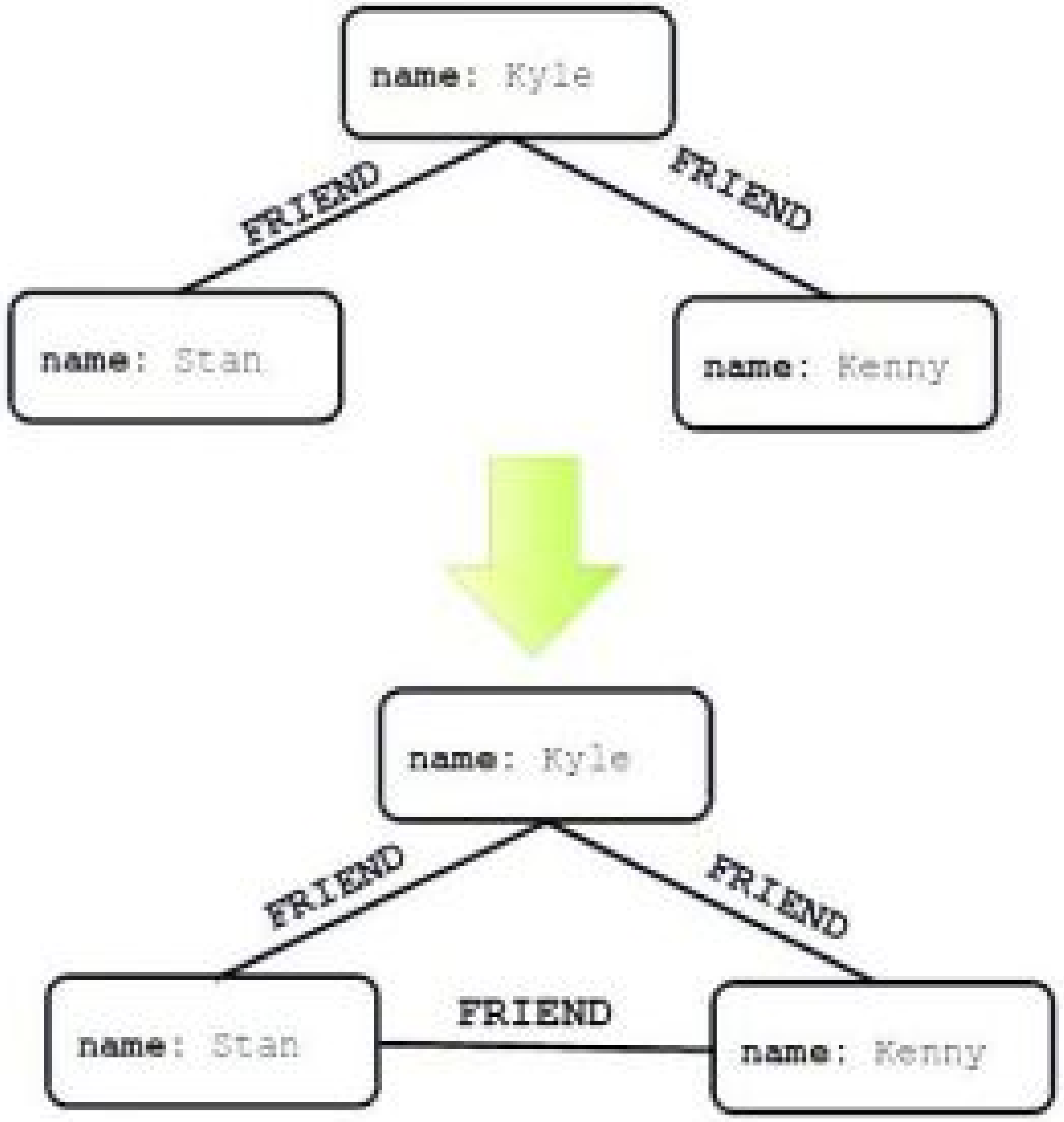


DULWICH SUPERMARKET & OFF LICENCE
ENGLISH , TURKISH , GREEK , MEDITERRANEAN FOOD
18
DELICATESSEN & ORGANICS
TEL : 020 8299 2214
FRESHLY CUT SANDWICHES - BILTONG
BILTONG

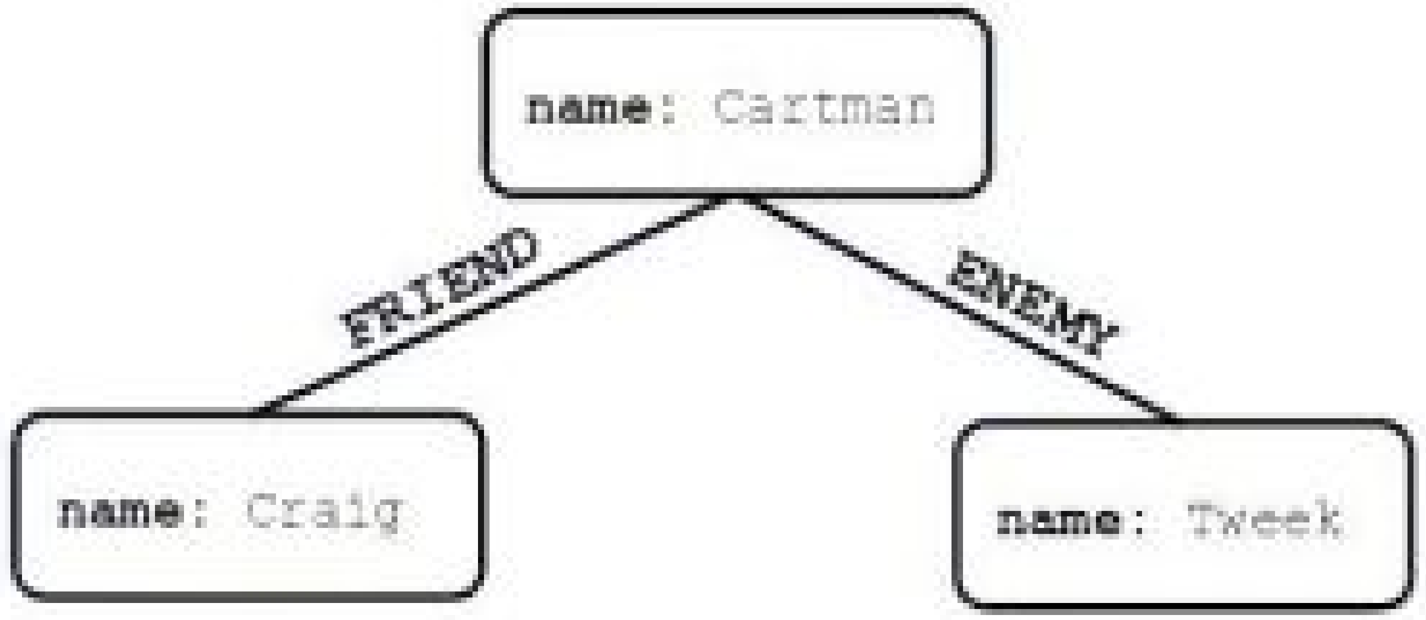
Triadic Closure



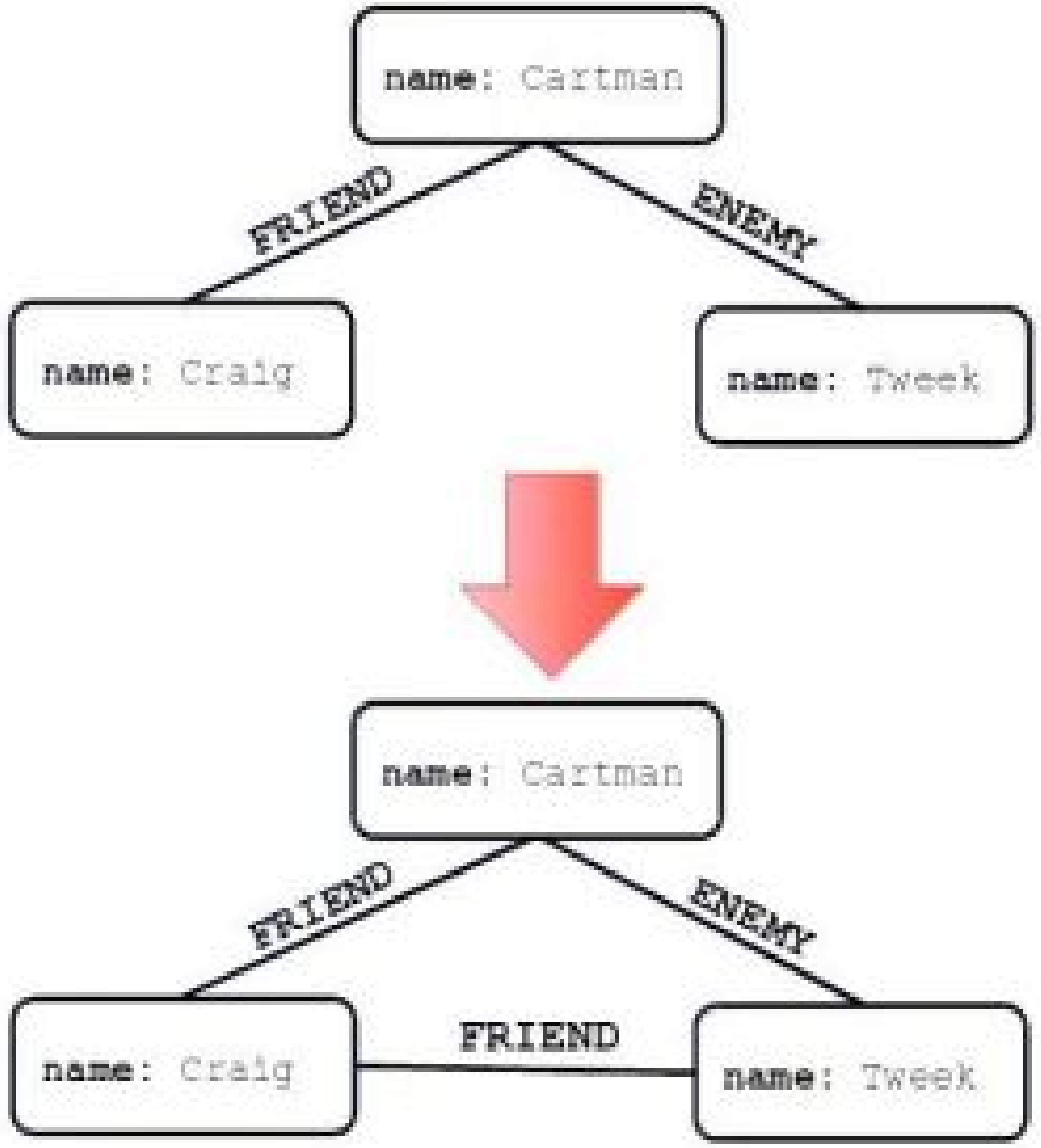
Triadic Closure



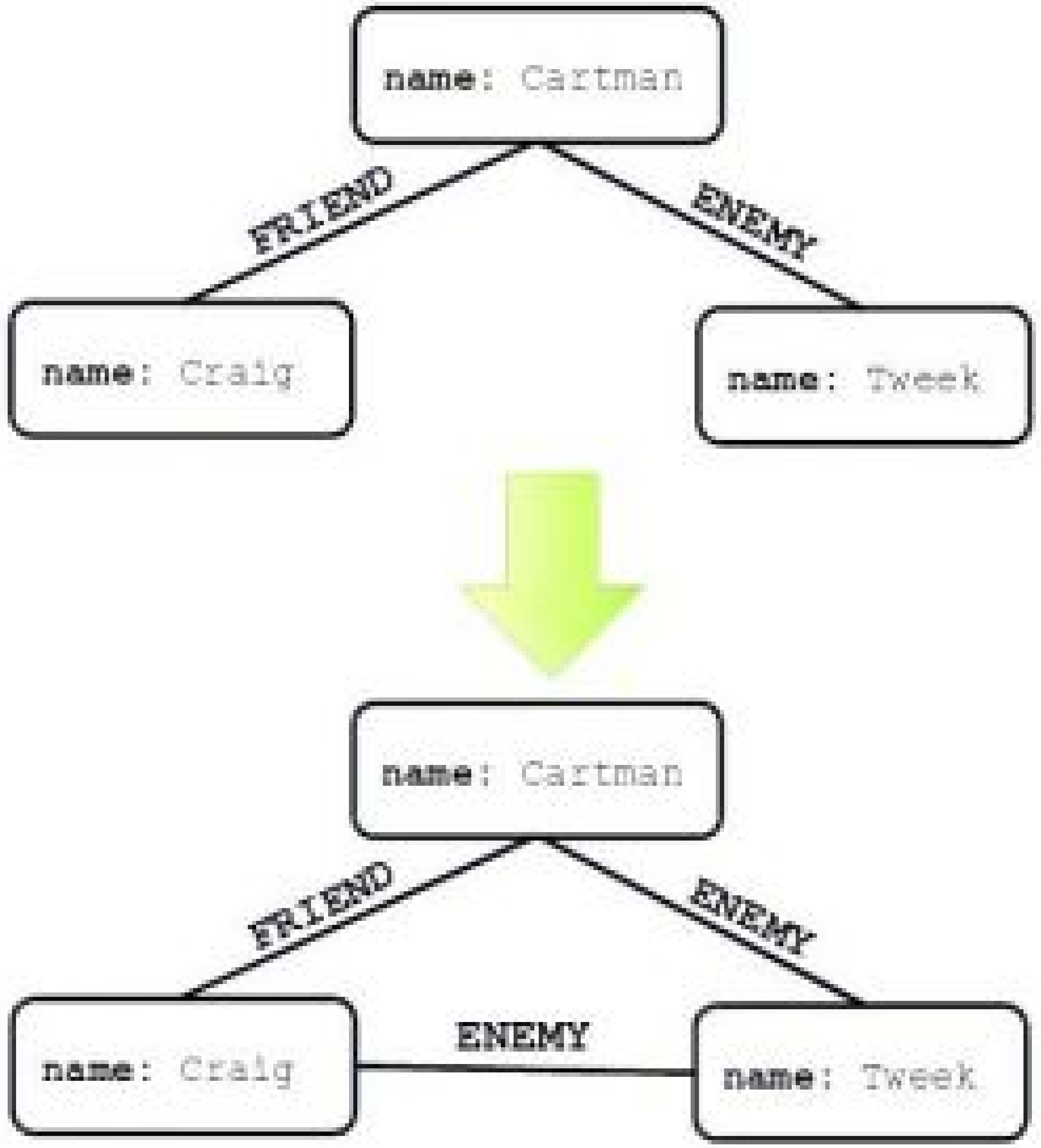
Structural Balance



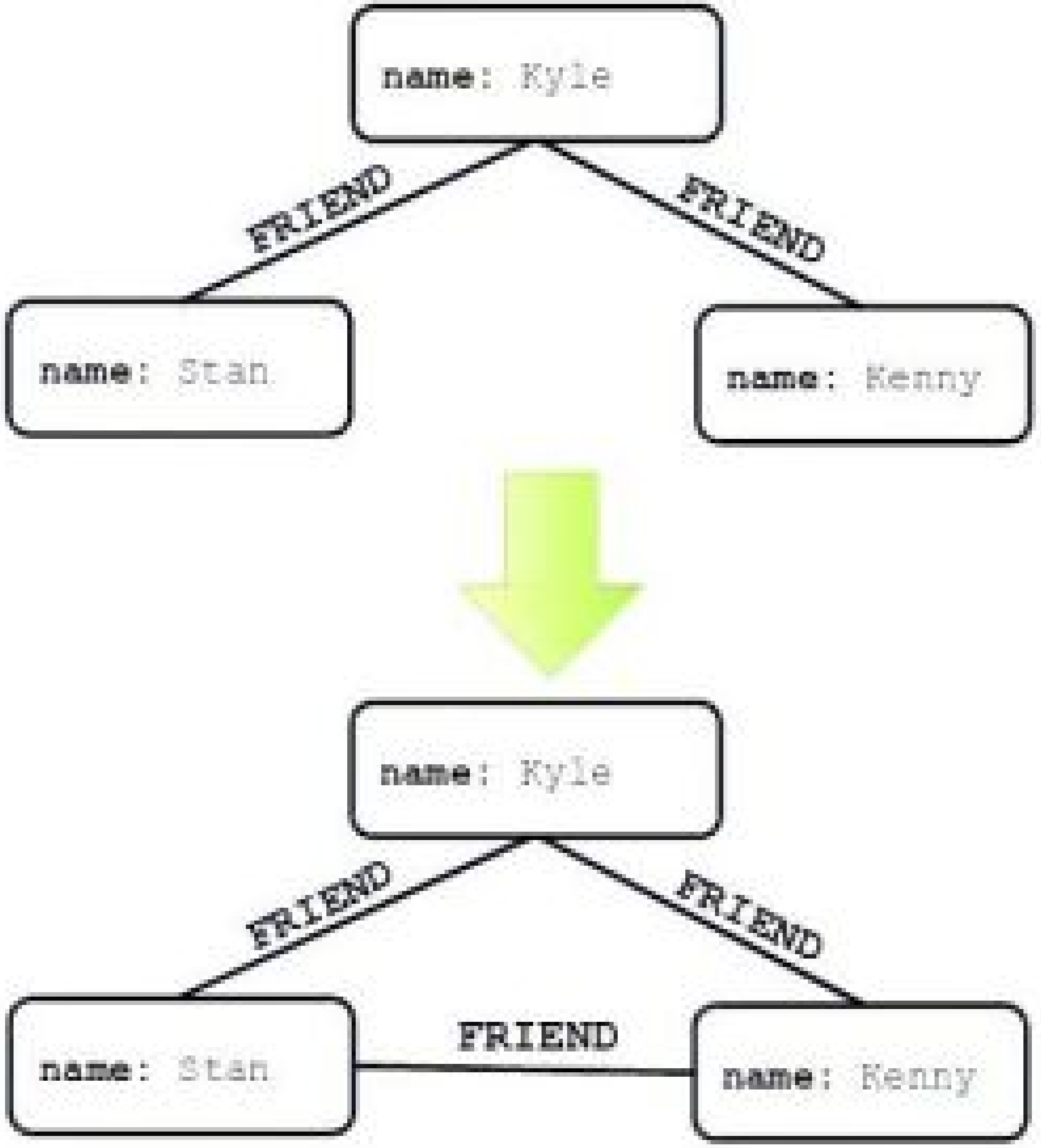
Structural Balance



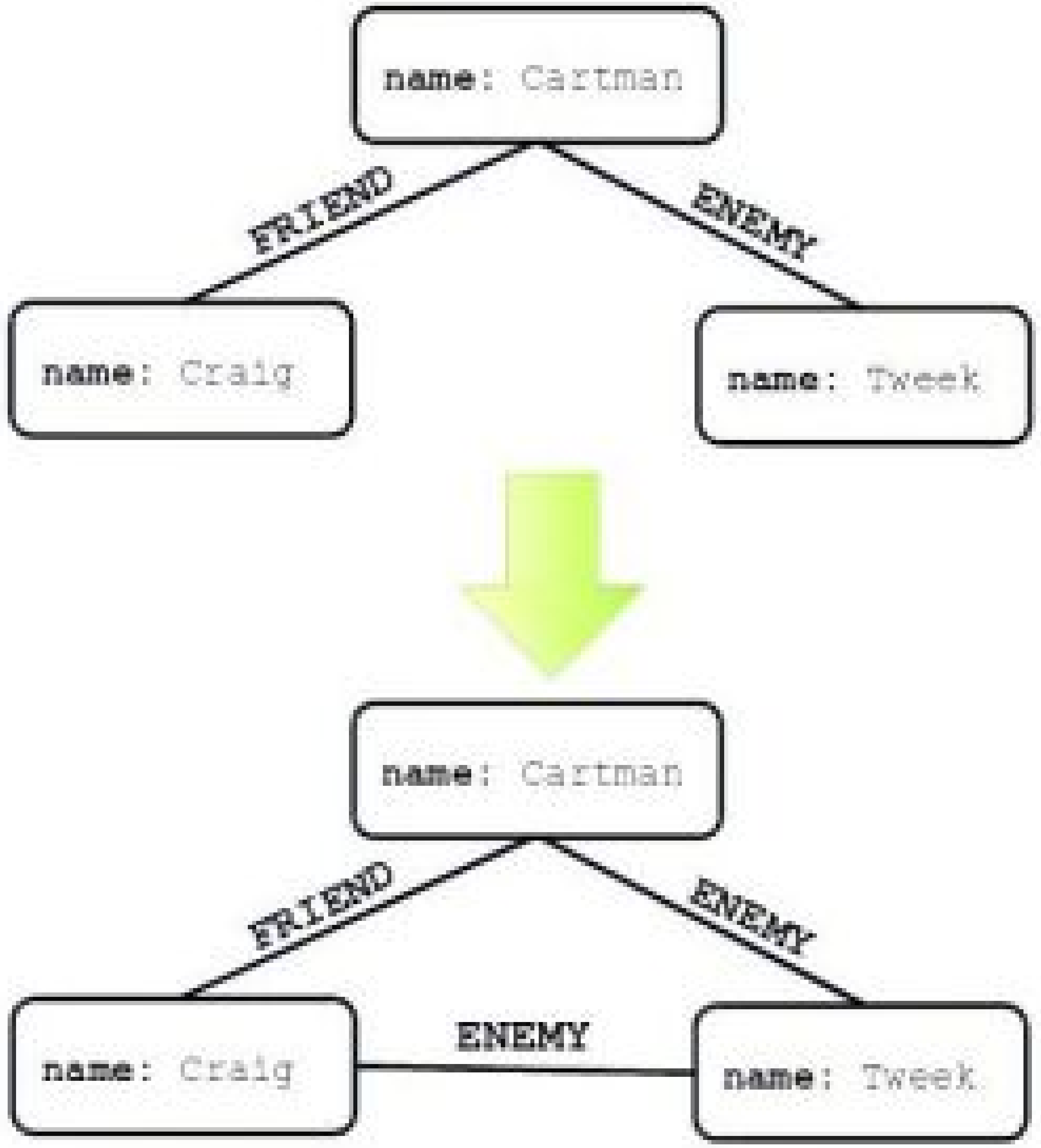
Structural Balance



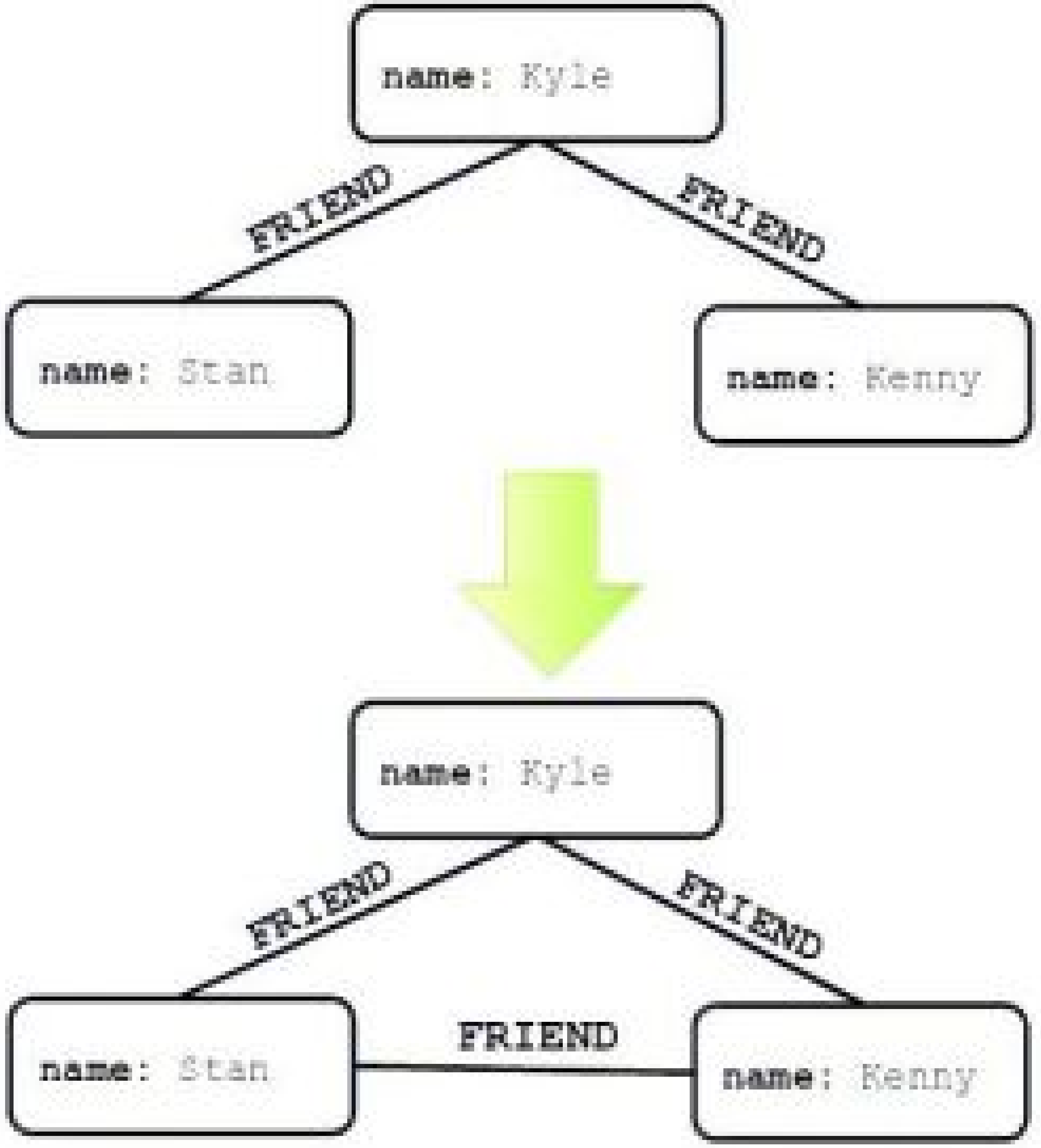
Structural Balance



Structural Balance



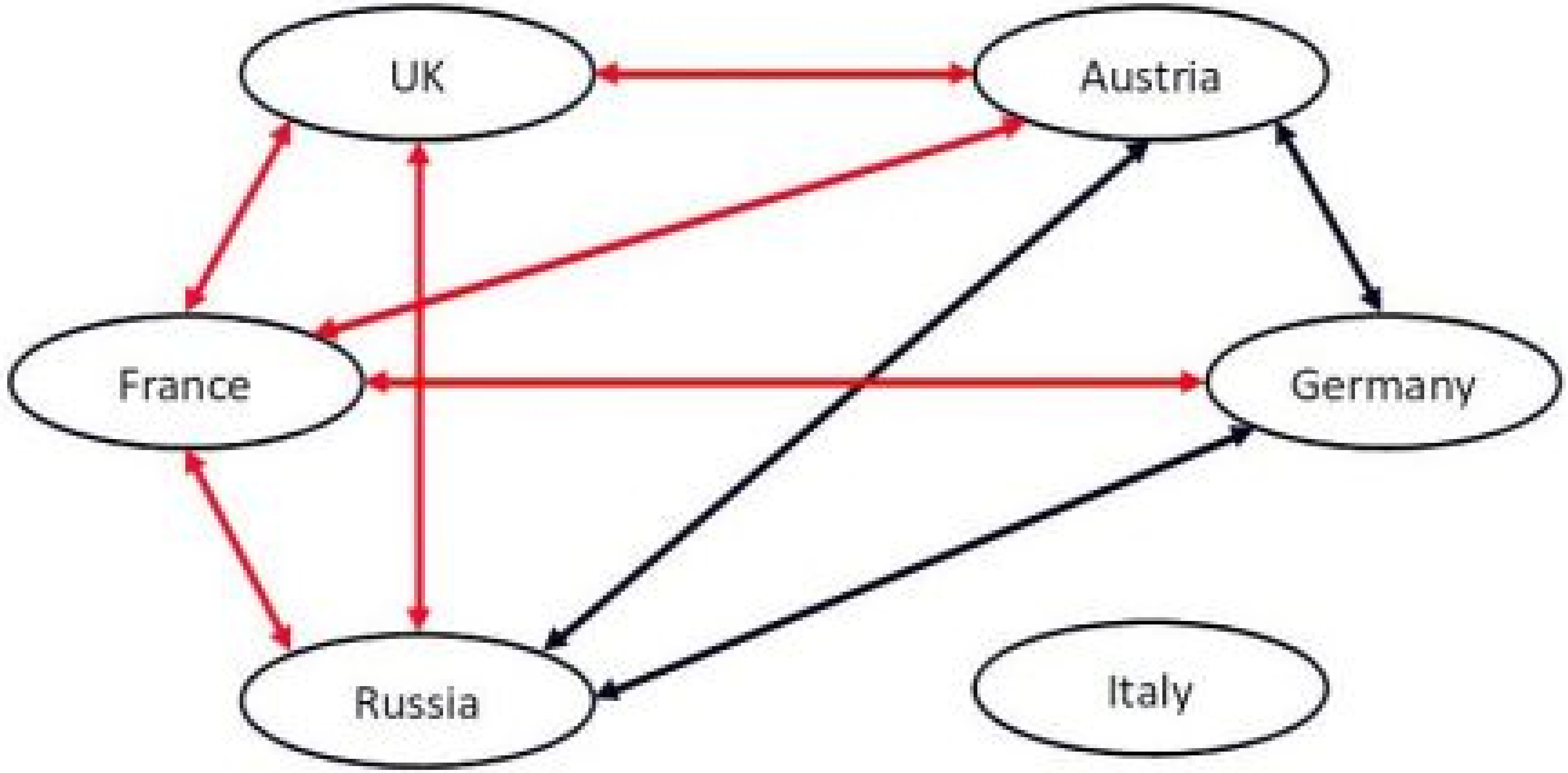
Structural Balance



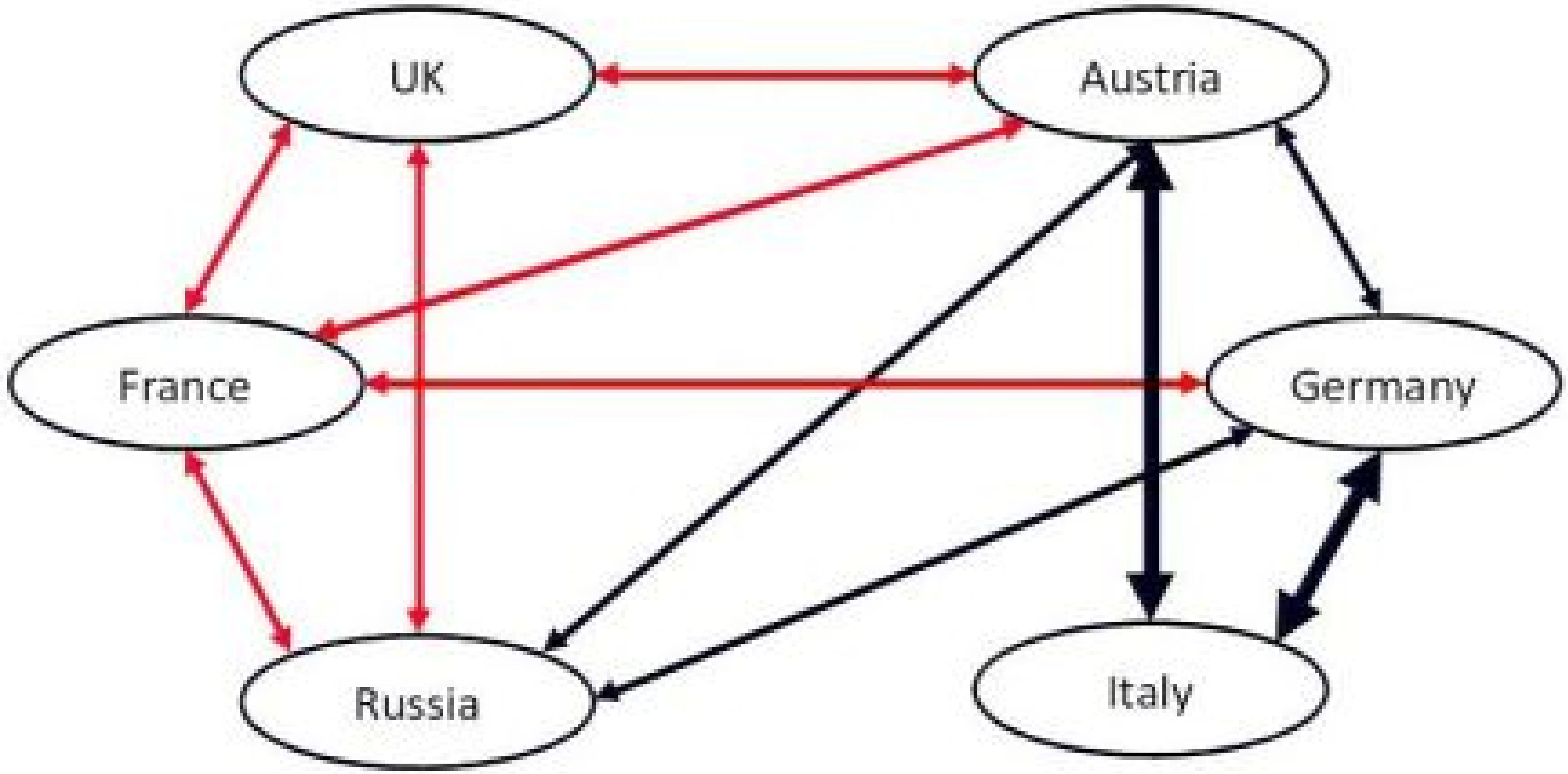
Structural Balance is a *key* predictive technique

And it’s domain-agnostic

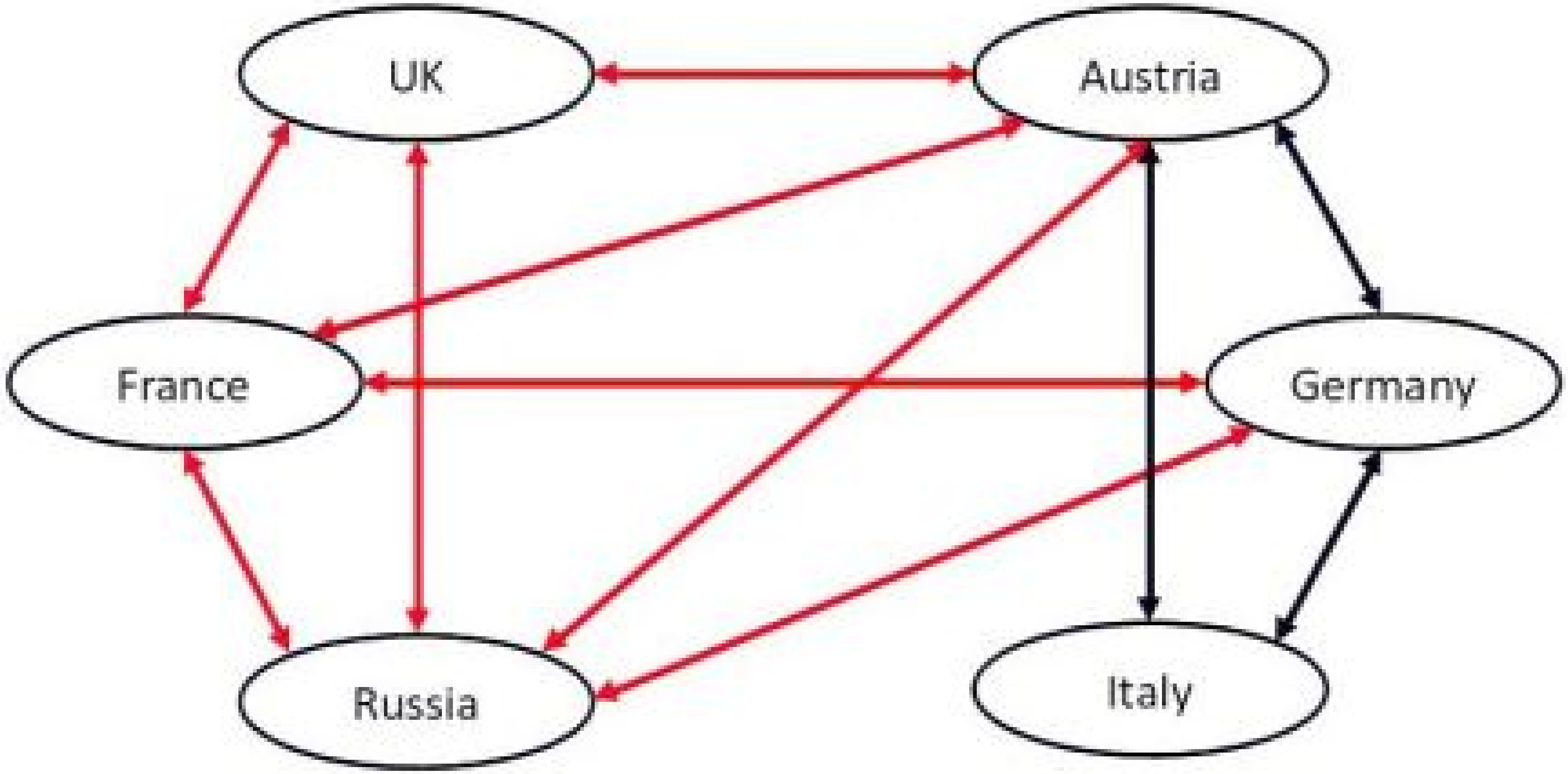
Allies and Enemies



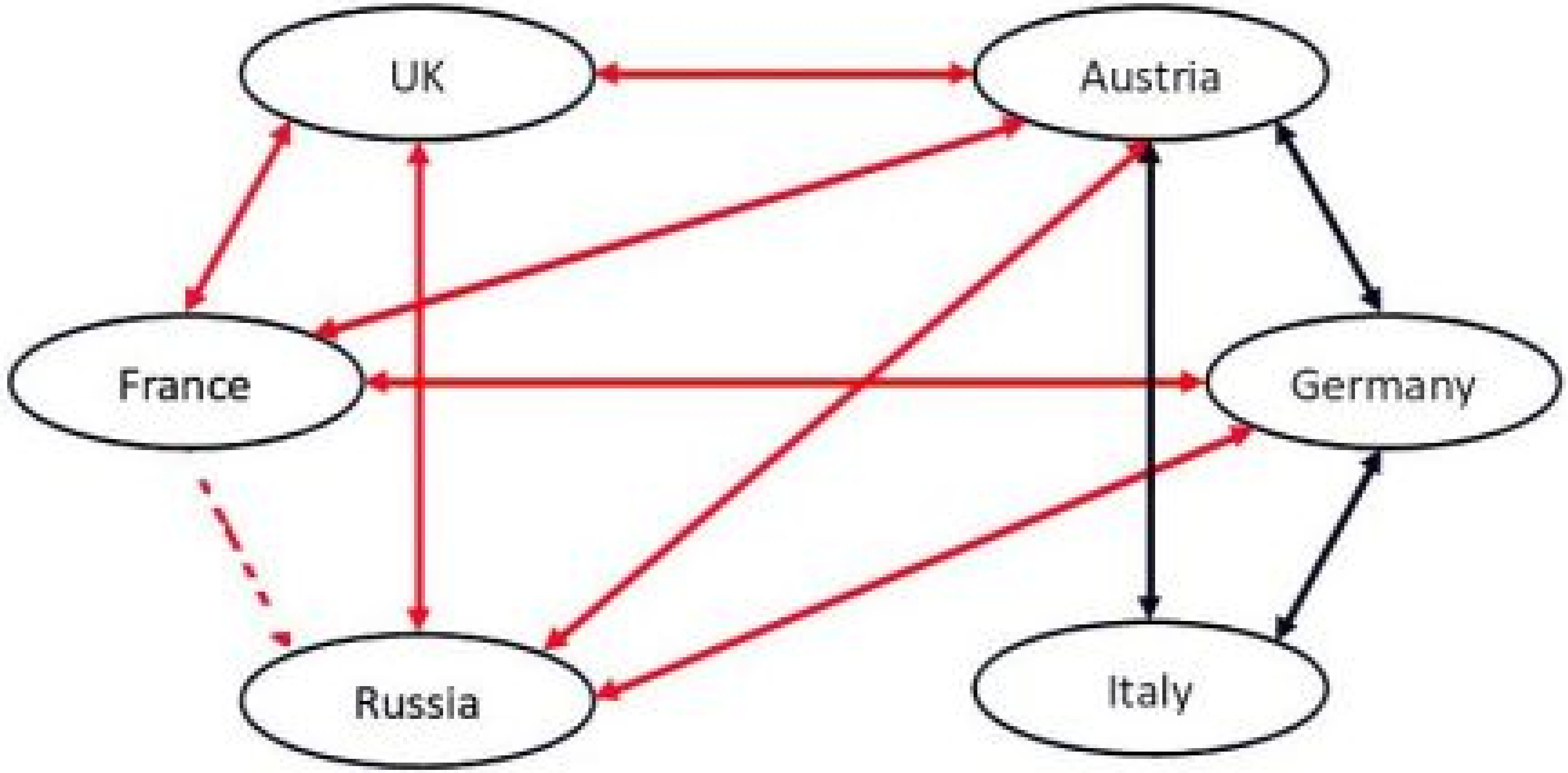
Allies and Enemies



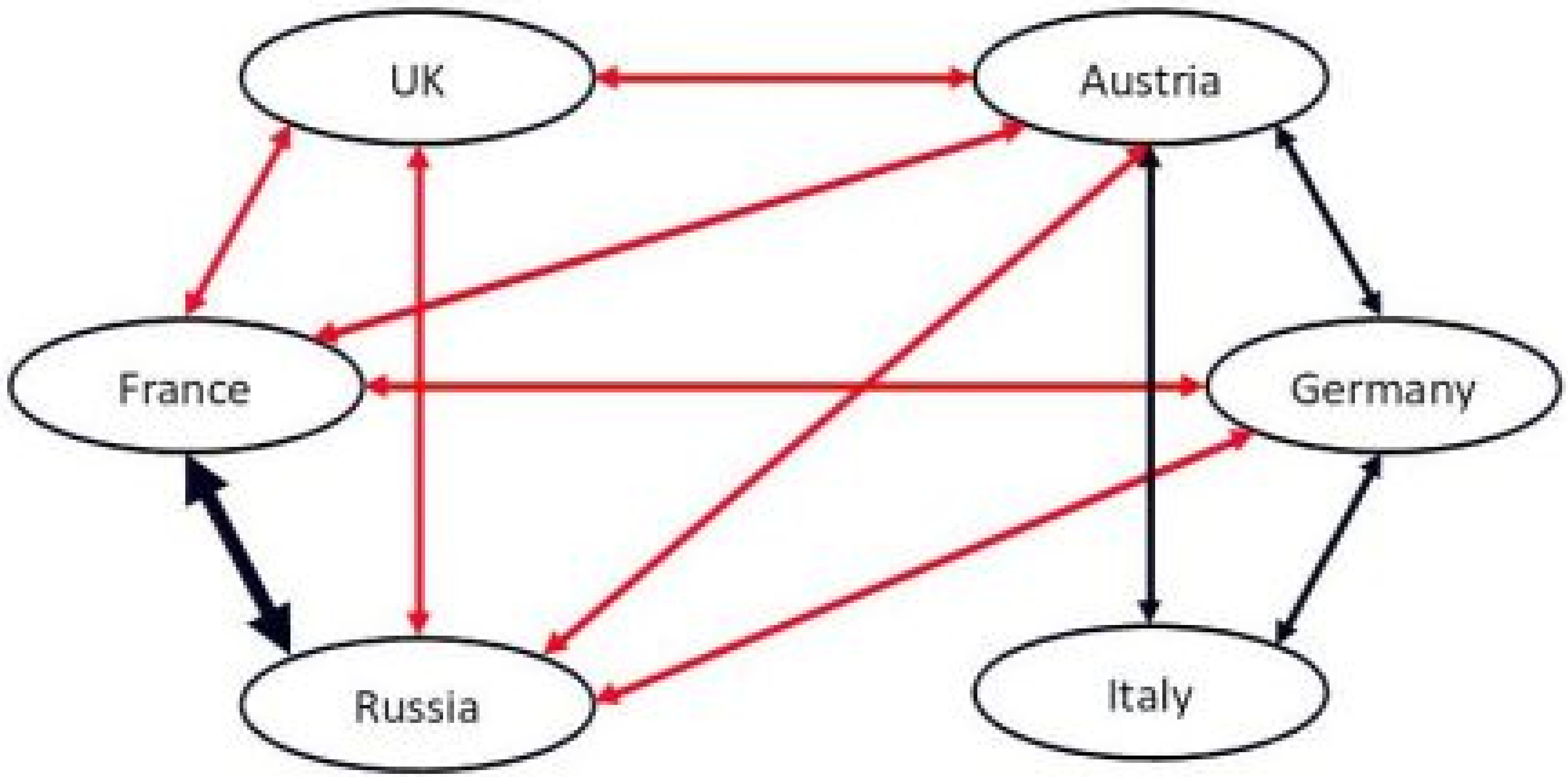
Allies and Enemies



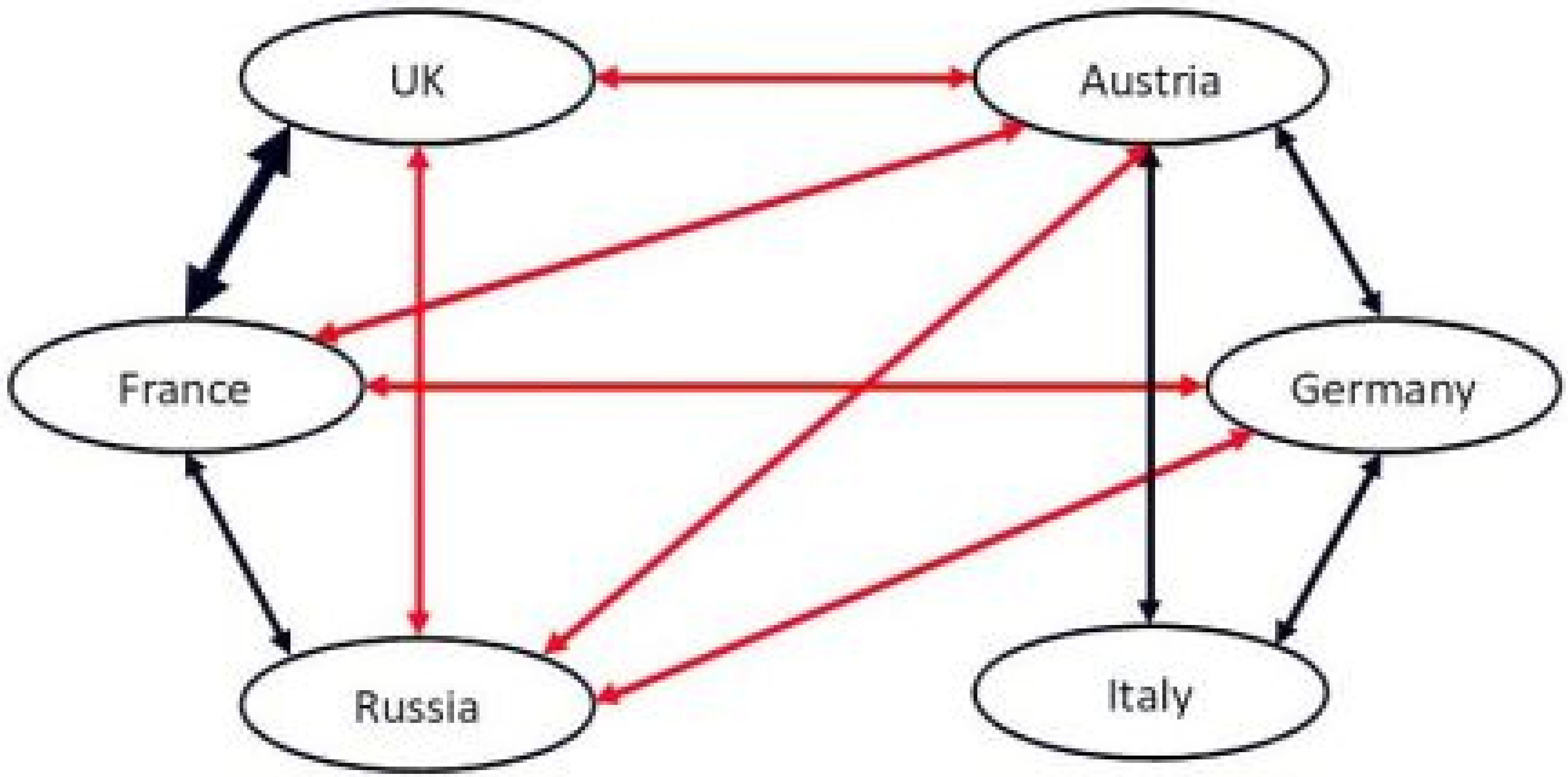
Allies and Enemies



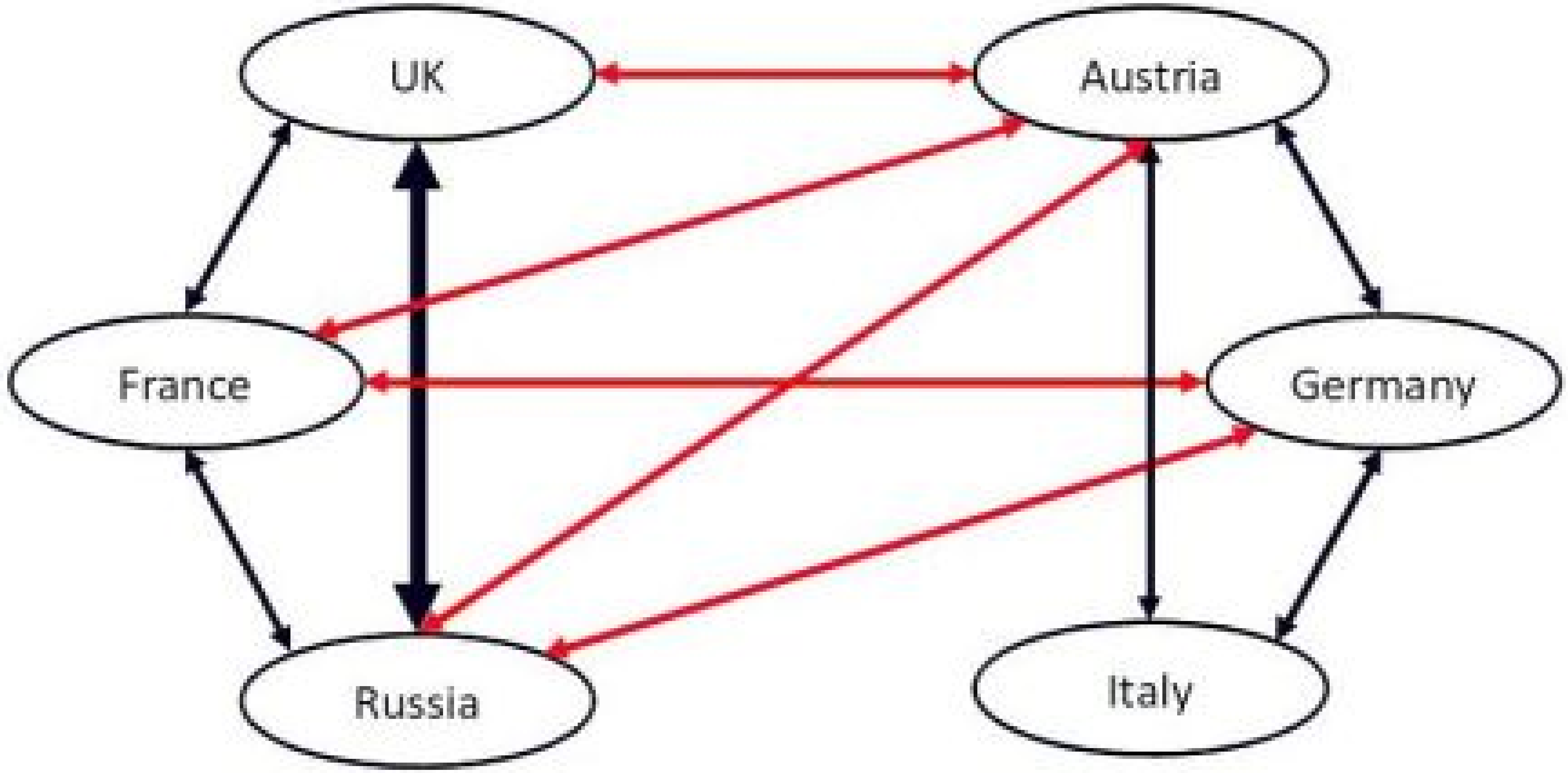
Allies and Enemies



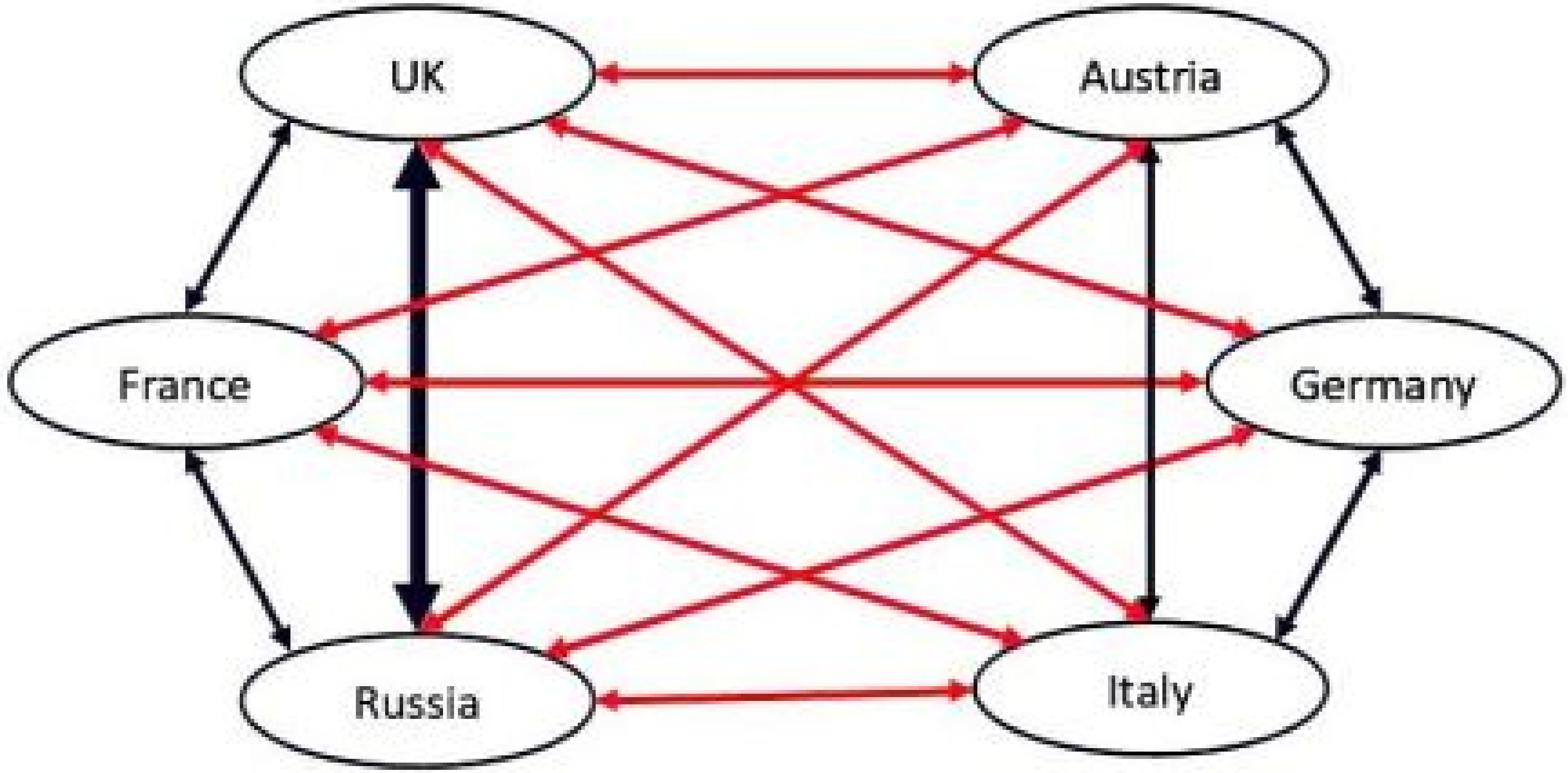
Allies and Enemies



Allies and Enemies

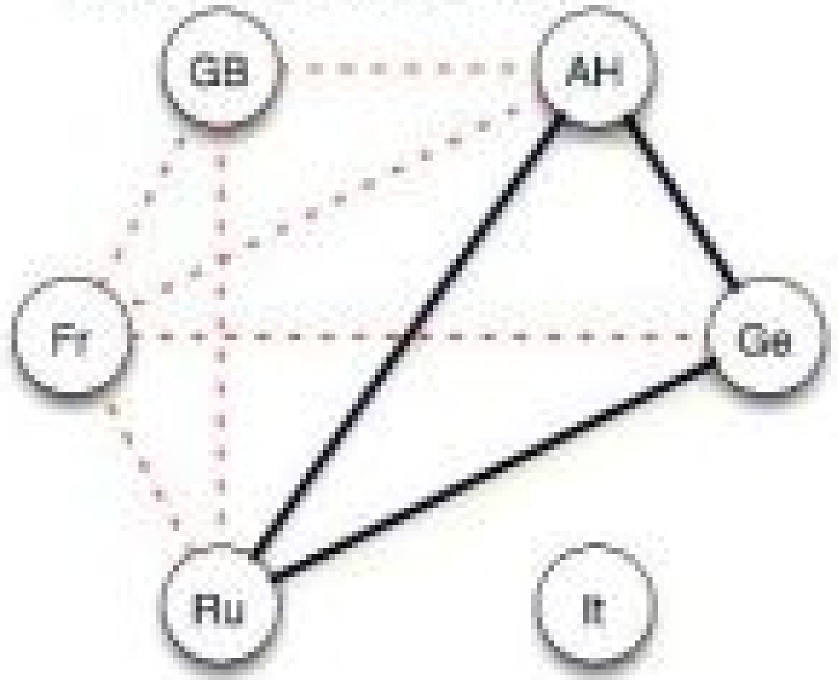


Allies and Enemies

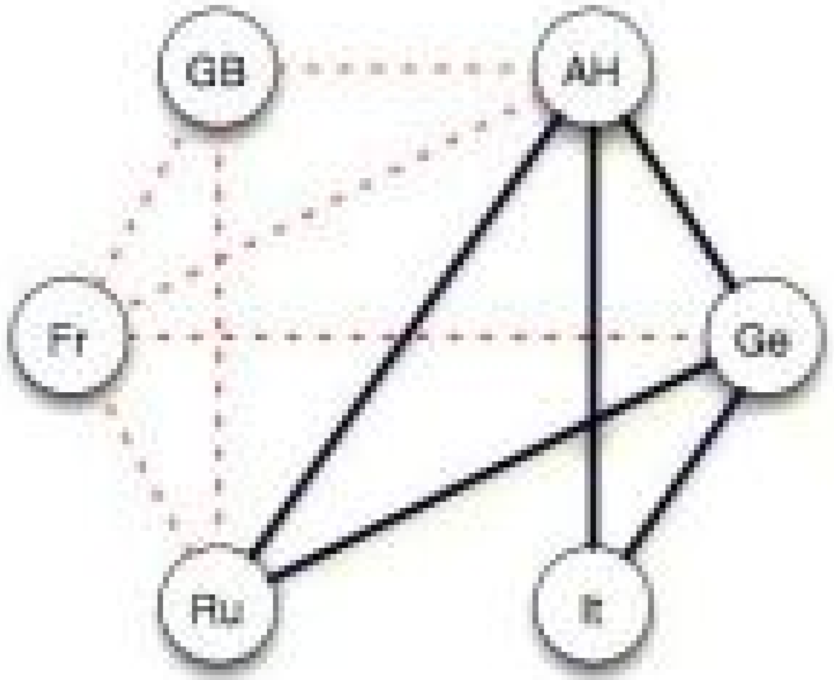


Predicting WWI

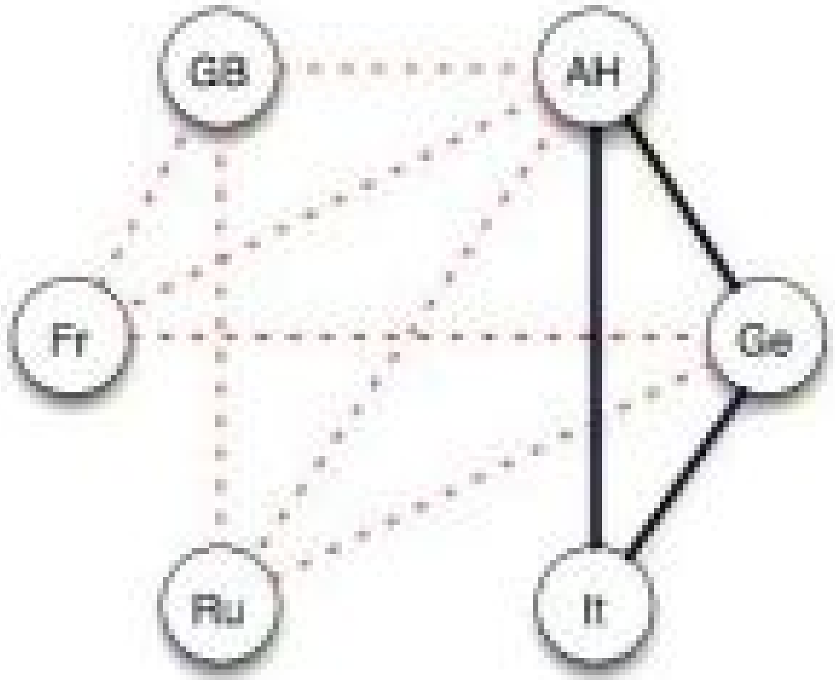
[Easley and Kleinberg]



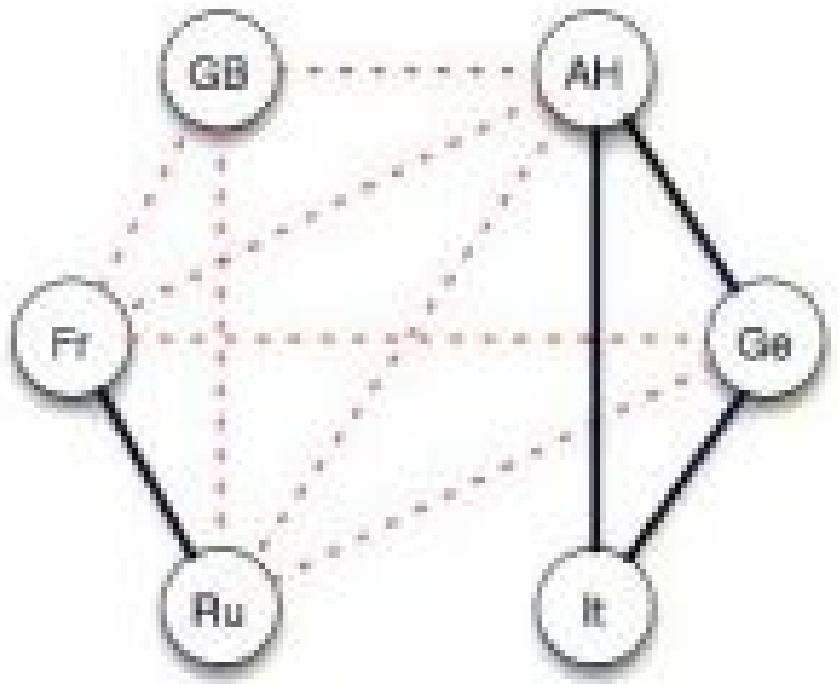
(a) *Three Emperors' League 1872–81*



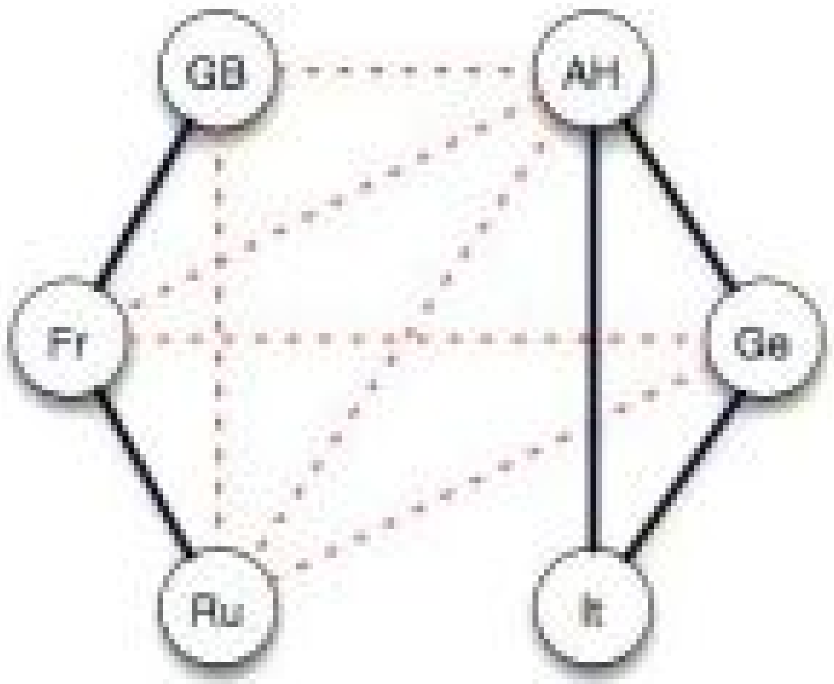
(b) *Triple Alliance 1882*



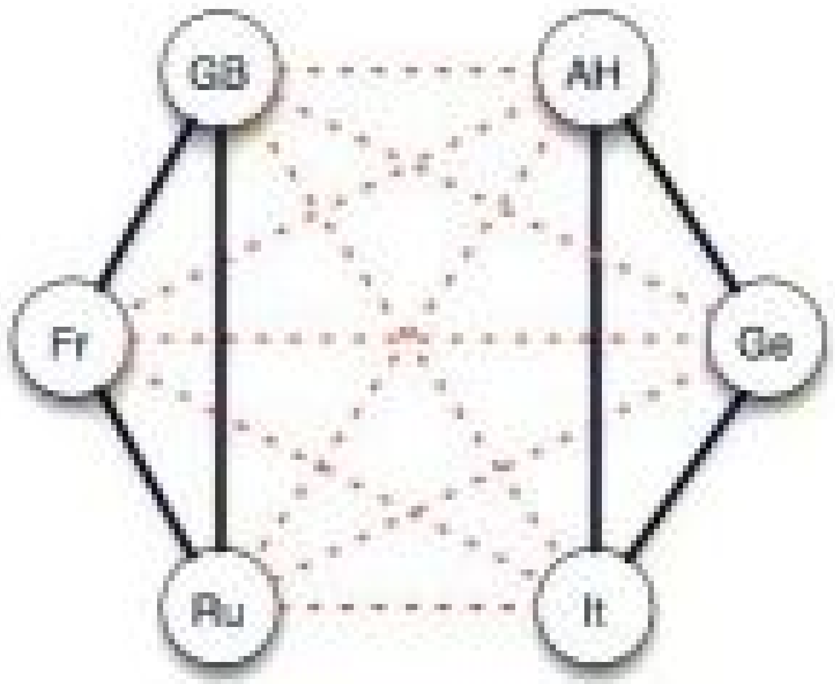
(c) *German-Russian Lapse 1890*



(d) *French-Russian Alliance 1891–94*



(e) *Entente Cordiale 1904*



(f) *British Russian Alliance 1907*

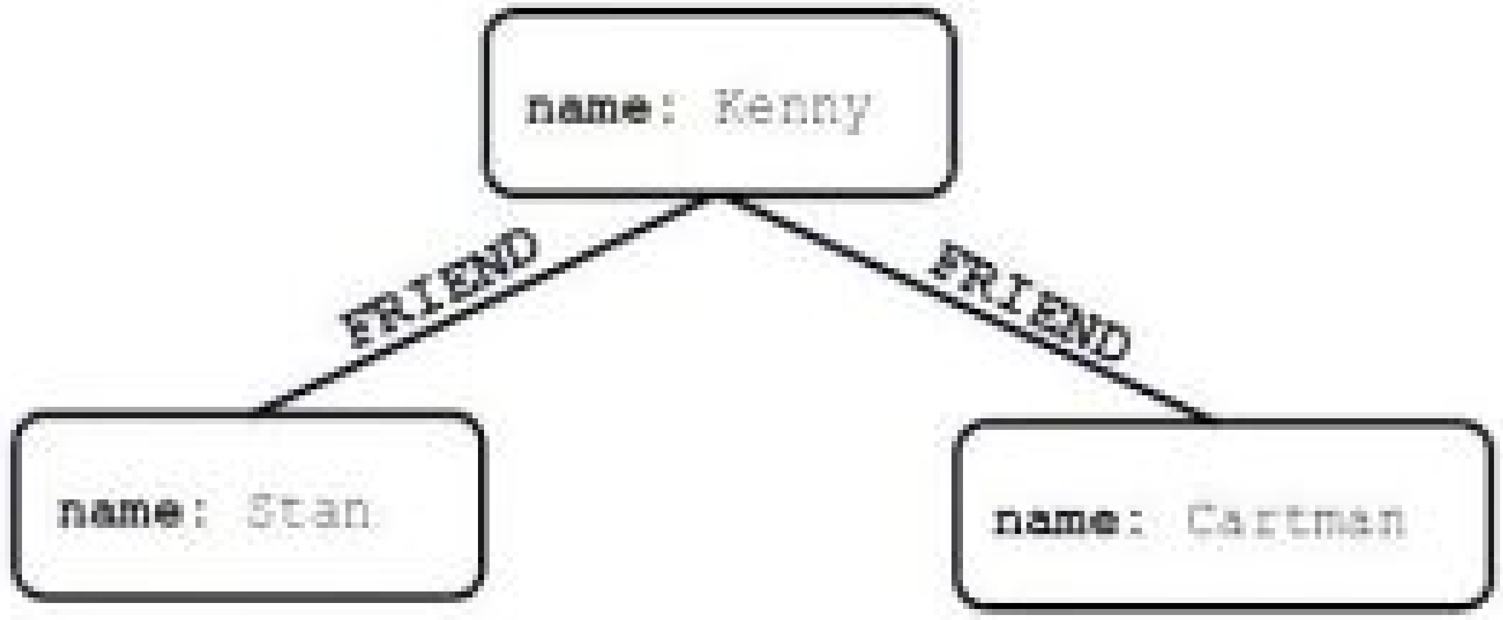
Strong Triadic Closure

It if a node has strong relationships to two neighbours, then these neighbours must have at least a weak relationship between them.

[Wikipedia]

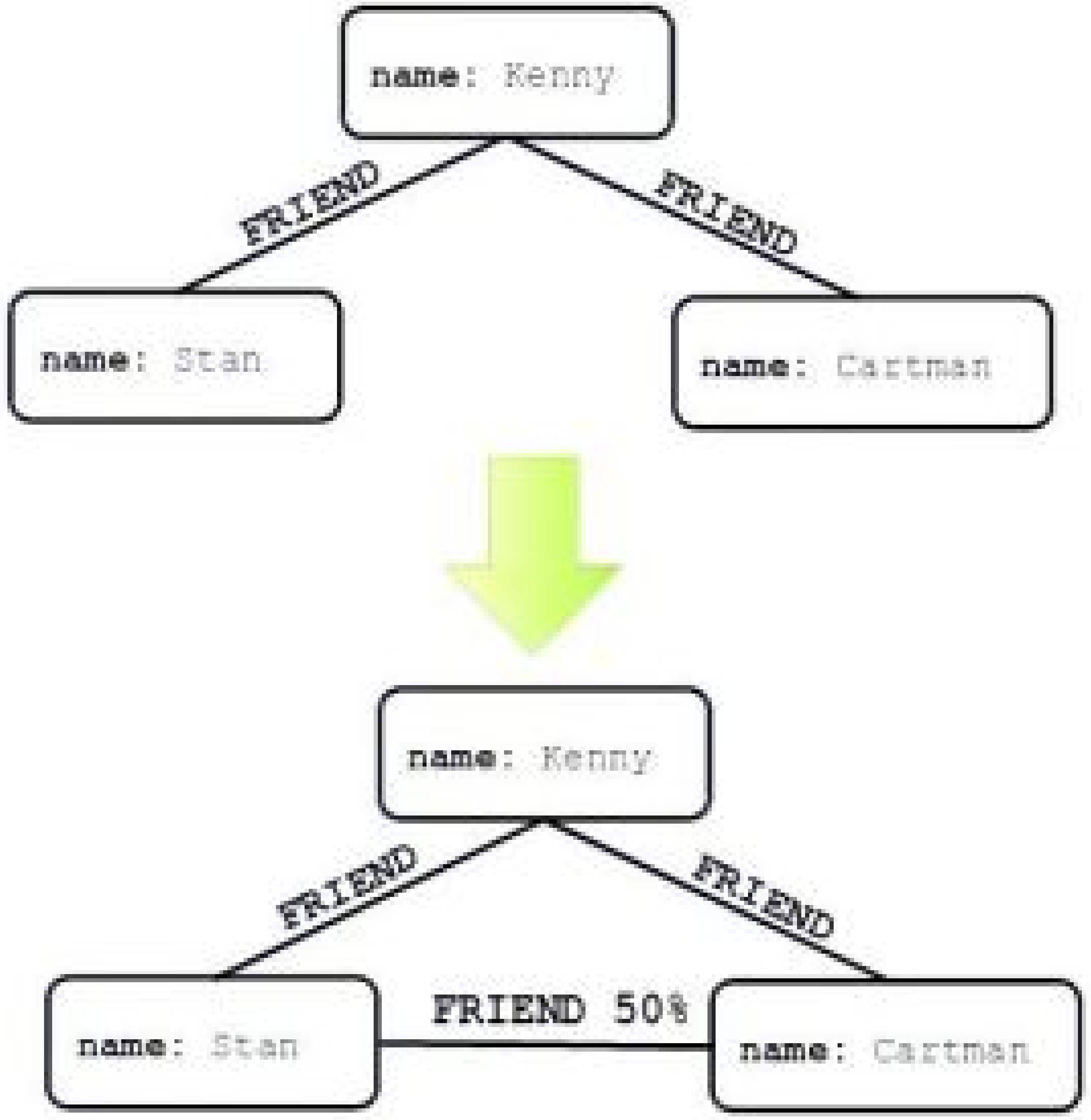
Triadic Closure

(weak relationship)



Triadic Closure

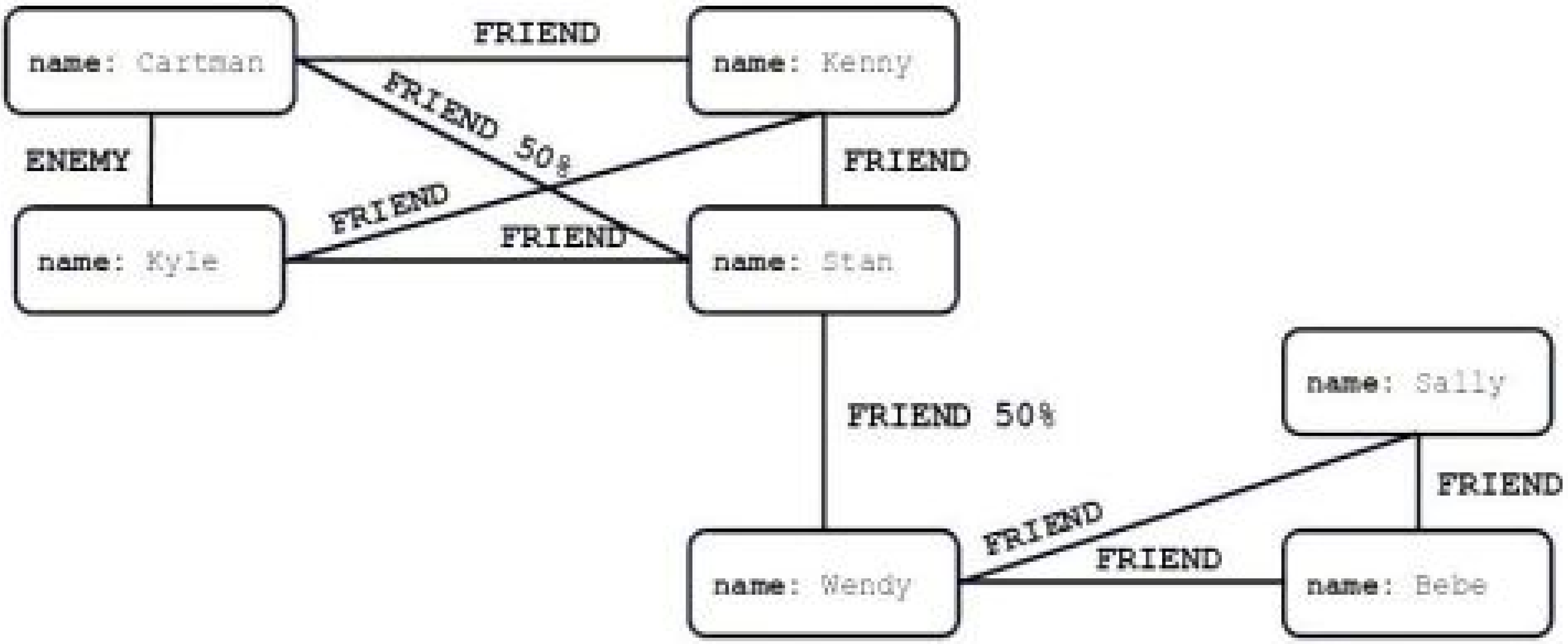
(weak relationship)



Weak relationships

- Relationships can have “strength” as well as intent
  - Think: weighting on a relationship in a property graph
- Weak links play another super-important structural role in graph theory
  - They bridge neighbourhoods

Local Bridges

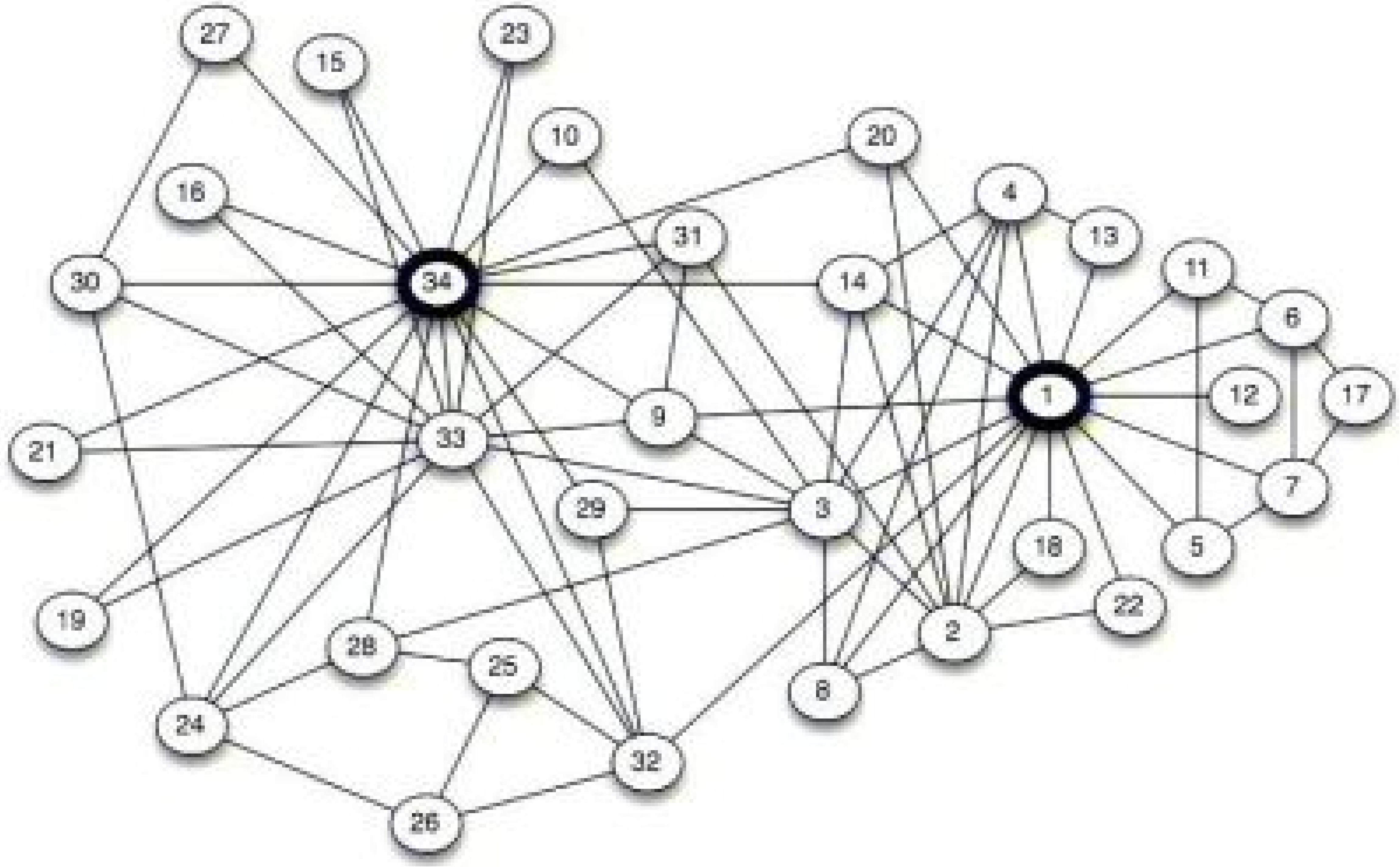


Local Bridge Property

*"If a node **A** in a network satisfies the Strong Triadic Closure Property and is involved in at least two strong relationships, then any local bridge it is involved in must be a weak relationship."*

[Easley and Kleinberg]

University Karate Club

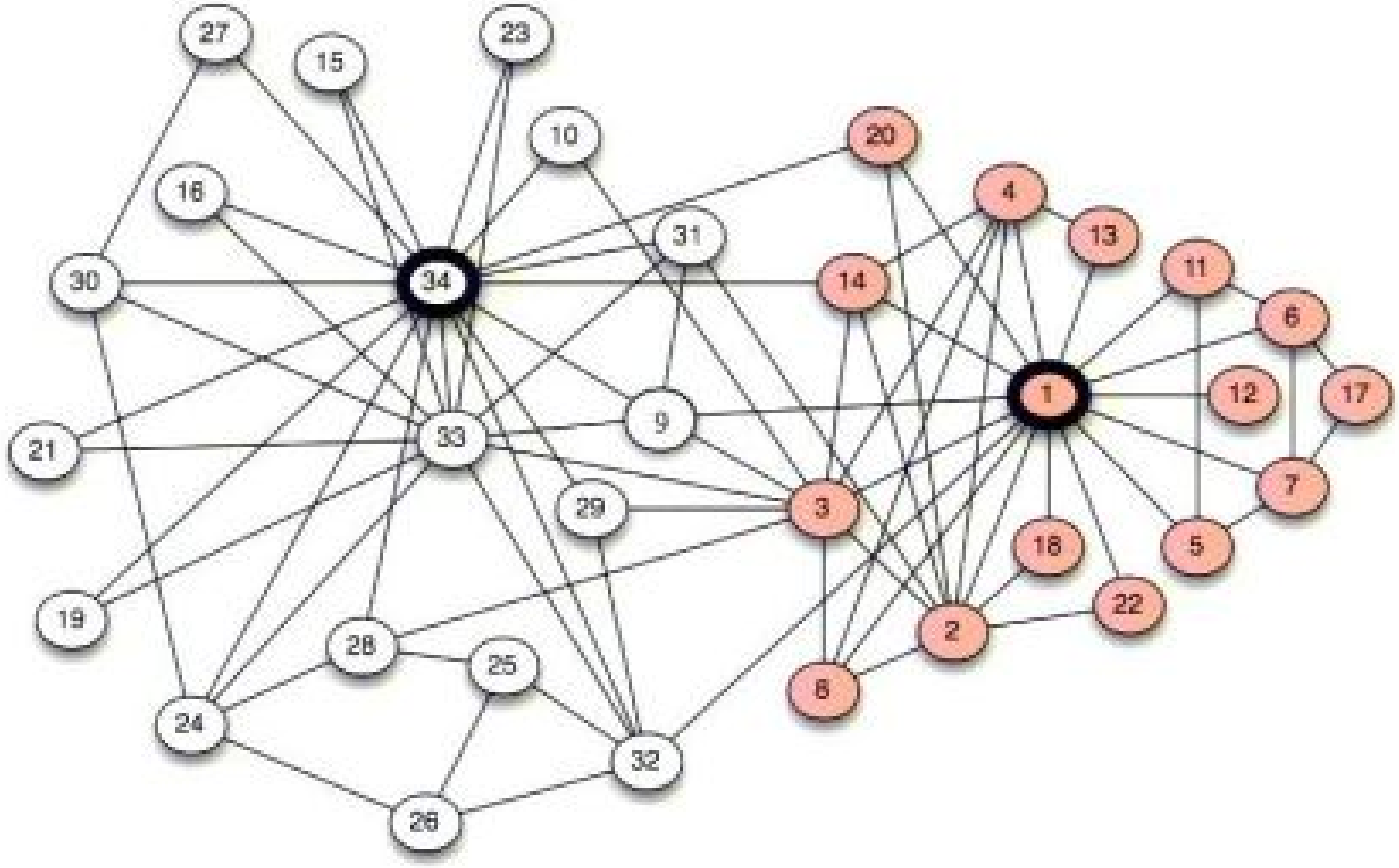


Graph Partitioning

- (NP) Hard problem
  - Recursively remove the spanning links between dense regions
  - Or recursively merge nodes into ever larger “subgraph” nodes
  - Choose your algorithm carefully – some are better than others for a given domain
- Can use to (almost exactly) predict the break up of the karate club!

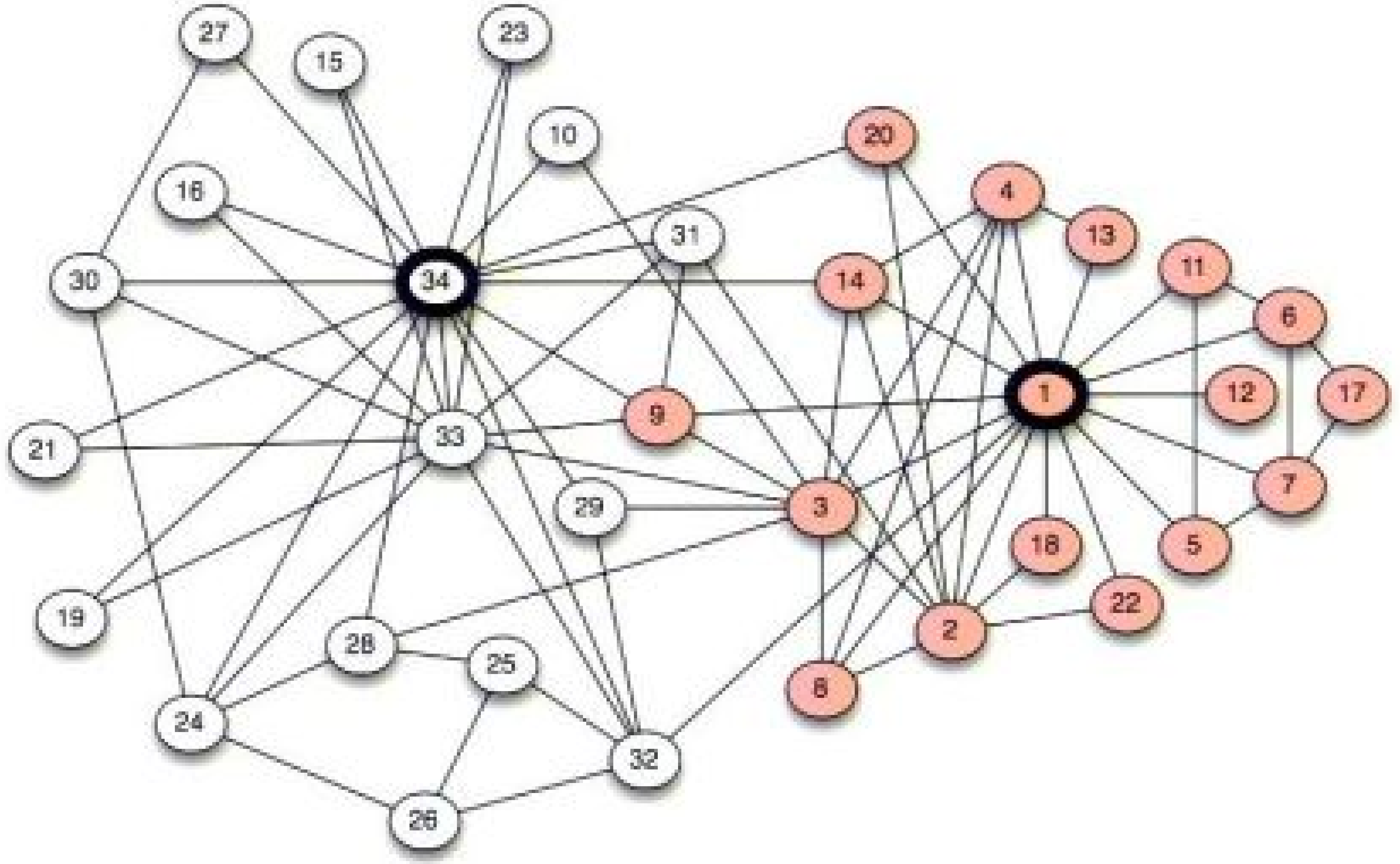
University Karate Clubs

(predicted by Graph Theory)



University Karate Clubs

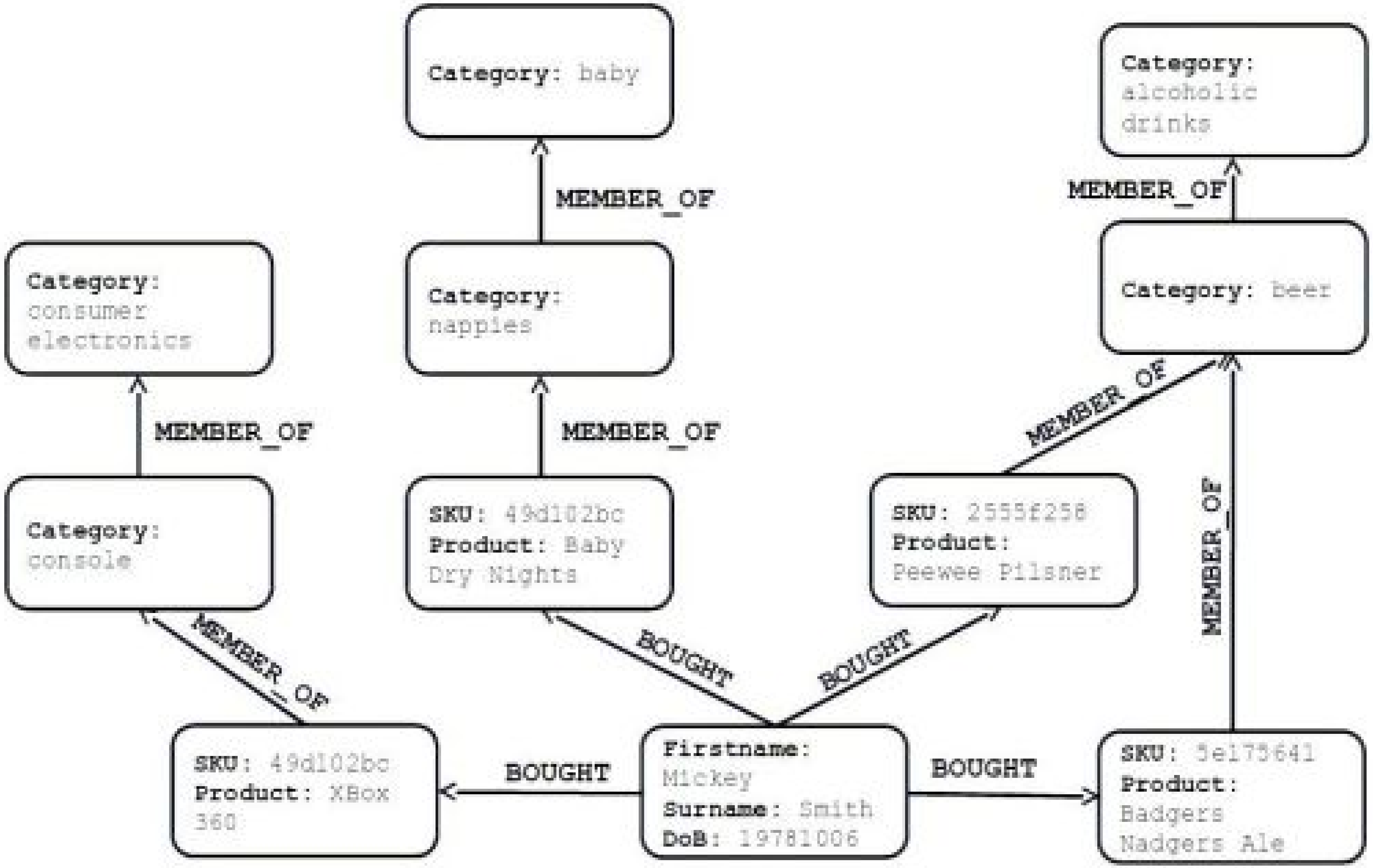
(what actually happened!)



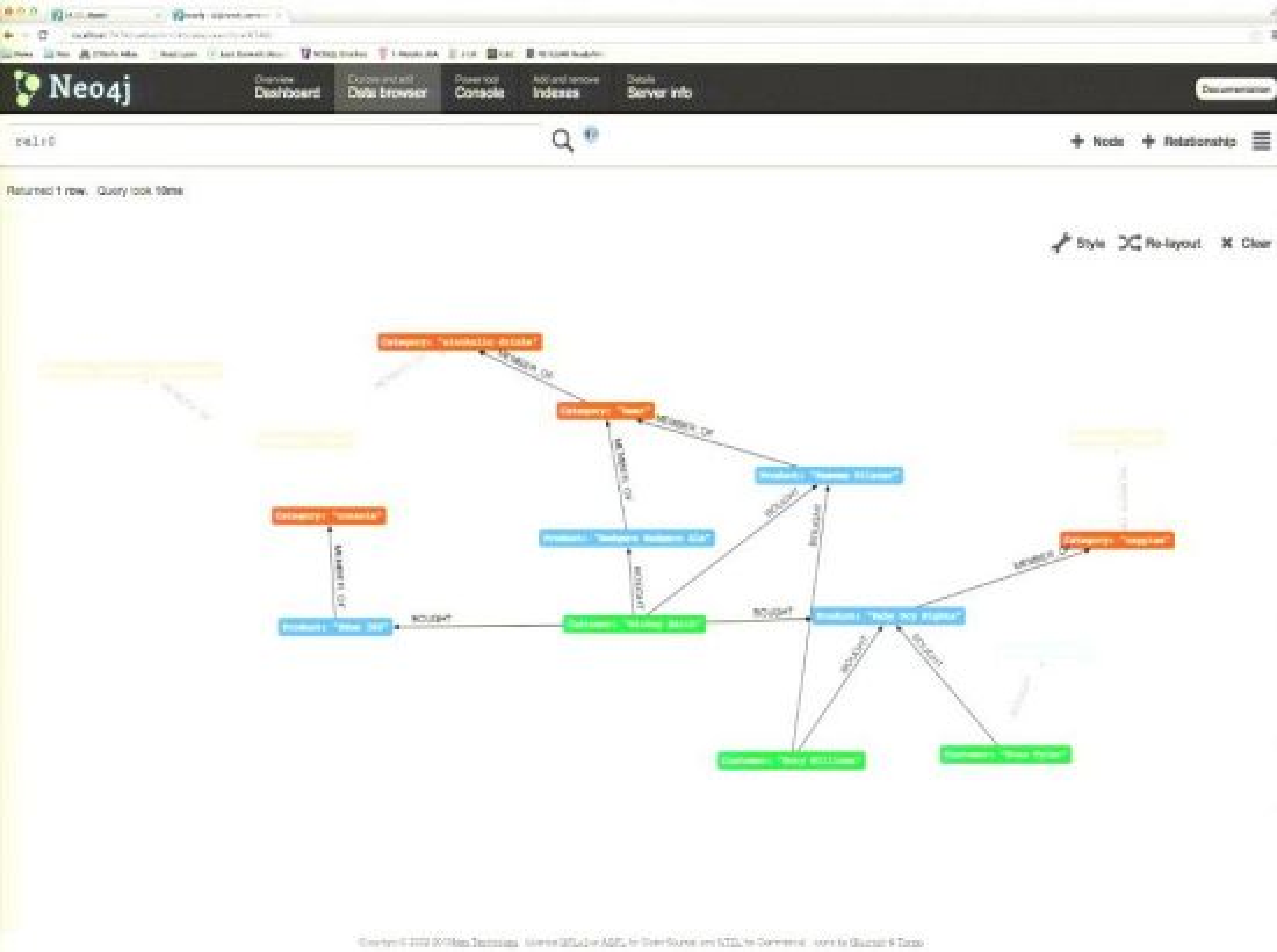
John
Crane
Ltd

Cypher

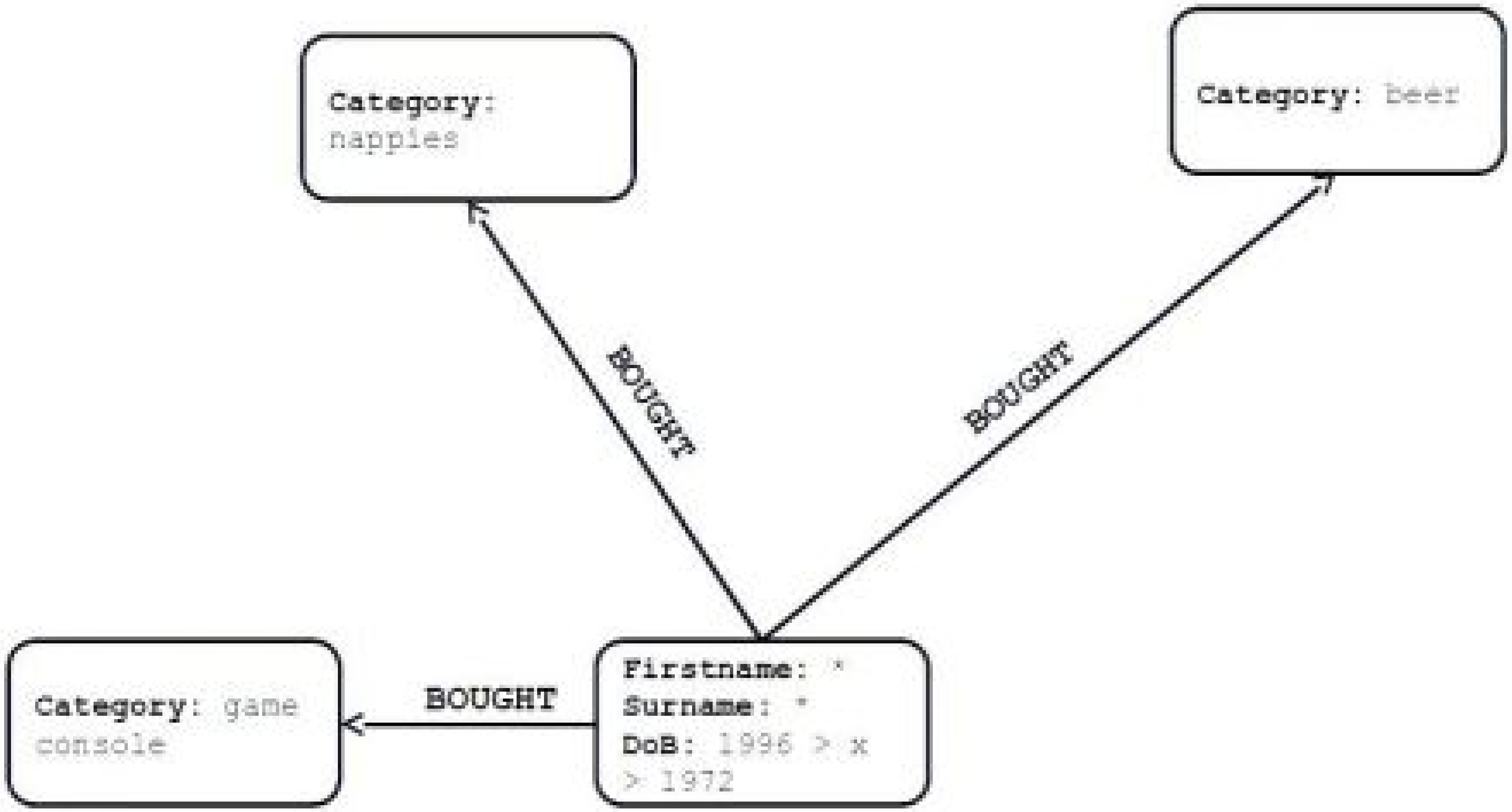
- Declarative graph pattern matching language
  - “SQL for graphs”
  - Columnar results
- Supports graph matching commands and queries
  - Find me stuff like this…
  - Aggregation, ordering and limit, etc.

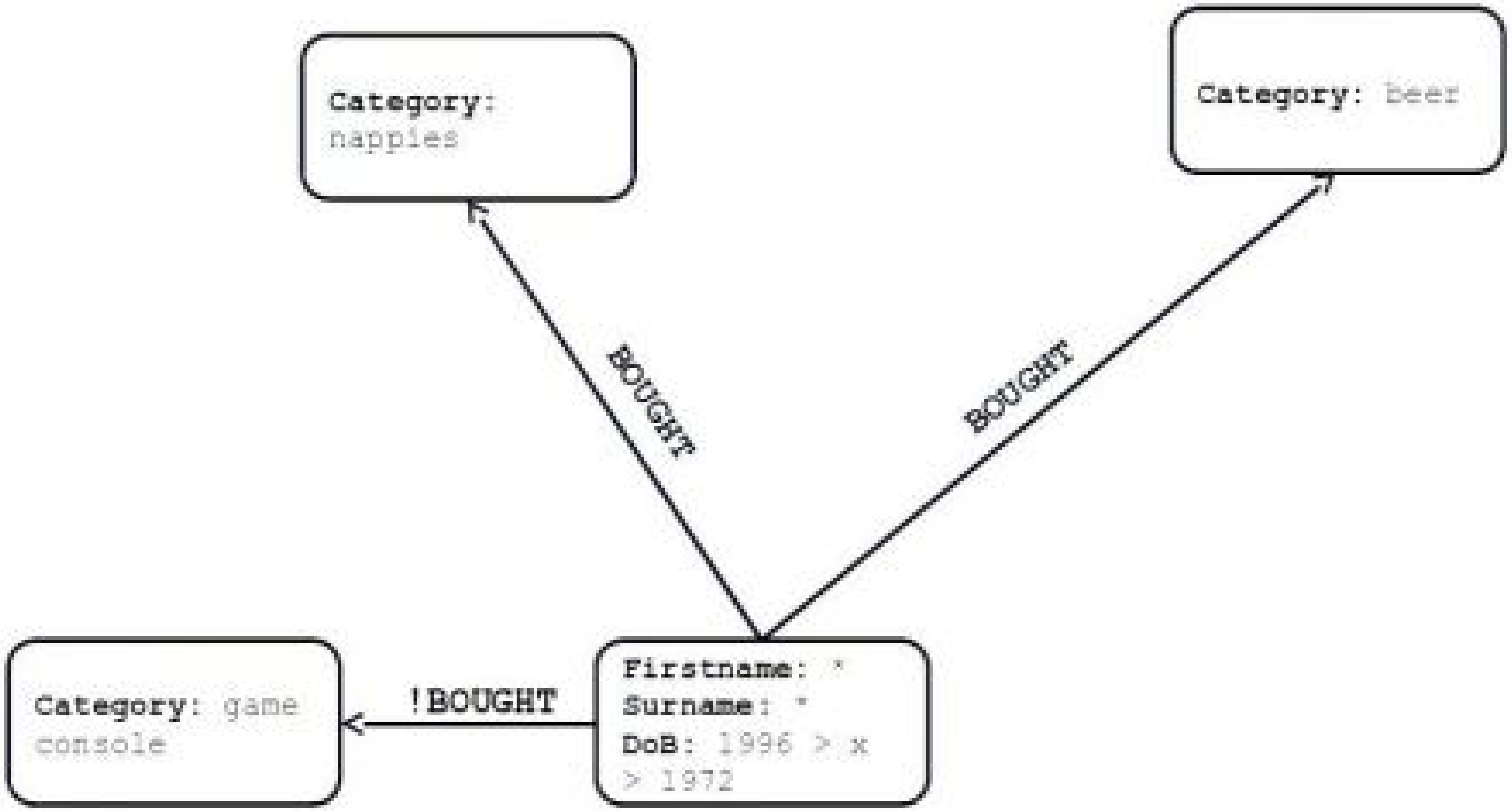

Category: baby
Category: alcoholic drinks
MEMBER_OF
MEMBER_OF
Category: consumer electronics
Category: nappies
Category: beer
MEMBER_OF
MEMBER_OF
MEMBER_OF
Category: console
SKU: 49d102bc
Product: Baby Dry Nights
SKU: 2555f258
Product: Peewee Pilsner
MEMBER_OF
MEMBER_OF
BOUGHT
BOUGHT
SKU: 49d102bc
Product: XBox 360
BOUGHT
Firstname: Mickey
Surname: Smith
DoB: 19781006
BOUGHT
SKU: 5e175641
Product: Badgers Nadgers Ale

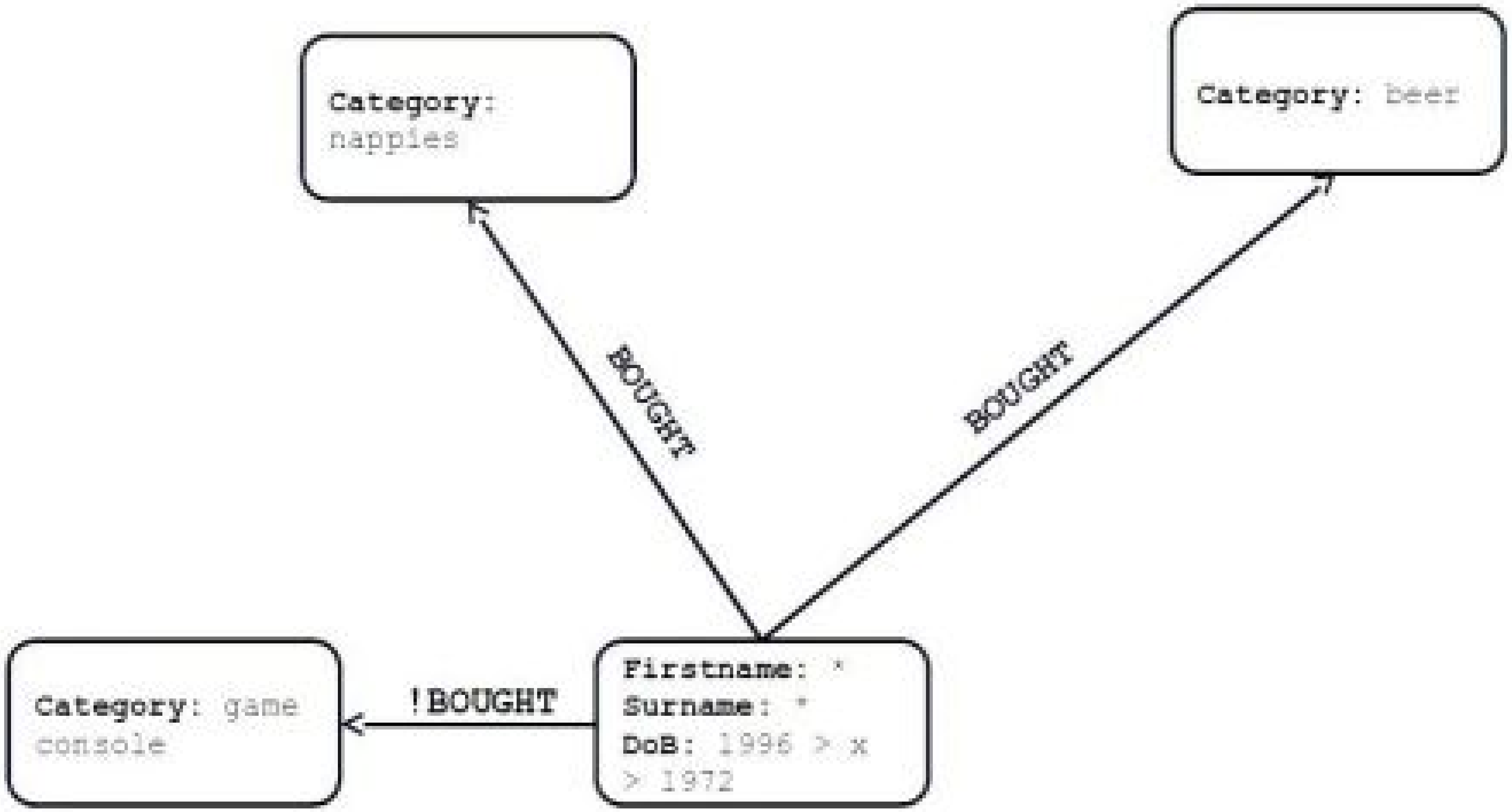
Σ

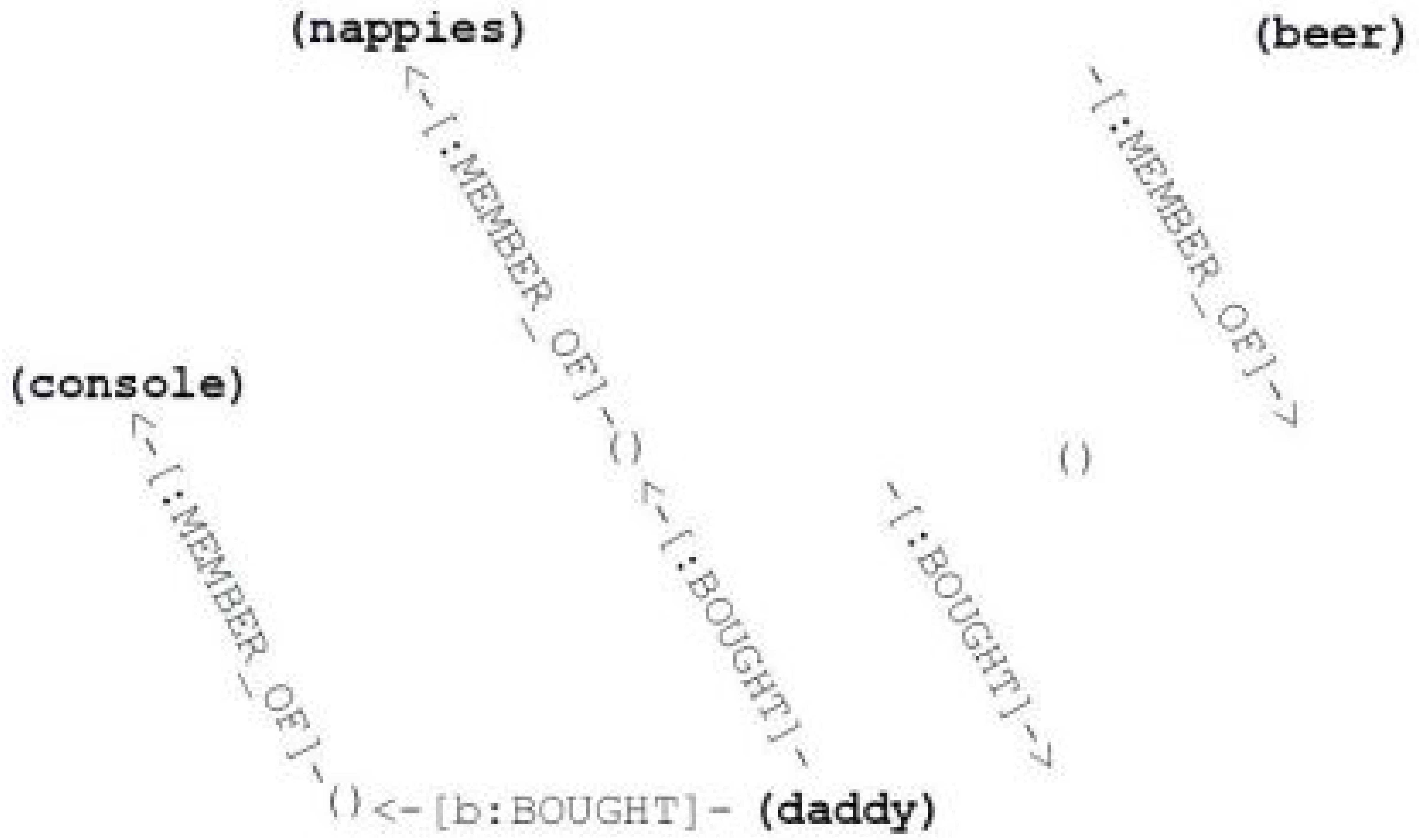


Returned 1 row. Query took 10ms


Category: nappies
Category: beer
BOUGHT
BOUGHT
Category: game console
BOUGHT
Firstname: *
Surname: *
DoB: 1996 > x > 1972


Category: nappies
Category: beer
BOUGHT
BOUGHT
Category: game console
!BOUGHT
Firstname: *
Surname: *
DoB: 1996 > x > 1972


Category: nappies
Category: beer
BOUGHT
BOUGHT
Category: game console
!BOUGHT
Firstname: *
Surname: *
DoB: 1996 > x > 1972


(nappies)
(beer)
(console)
<-[:MEMBER_OF]-()
-[:MEMBER_OF]->
()
<-[:MEMBER_OF]-()
<-[:BOUGHT]-
-[:BOUGHT]->
()<-[b:BOUGHT]- (daddy)

Flatten the graph

```
(daddy)-[:BOUGHT]->()-[:MEMBER_OF]->(nappies)
(daddy)-[:BOUGHT]->()-[:MEMBER_OF]->(beer)
(daddy)-[b:BOUGHT]->()-[:MEMBER_OF]->(console)
```

Wrap in a Cypher MATCH clause

```
MATCH (daddy)-[:BOUGHT]->()-[:MEMBER_OF]->(nappies),
(daddy)-[:BOUGHT]->()-[:MEMBER_OF]->(beer),
(daddy)-[b:BOUGHT]->()-[:MEMBER_OF]->(console)
```

Cypher WHERE clause

```
MATCH (daddy)-[:BOUGHT]->()-[:MEMBER_OF]->(nappies),
(daddy)-[:BOUGHT]->()-[:MEMBER_OF]->(beer),
(daddy)-[b:BOUGHT]->()-[:MEMBER_OF]->(console)
WHERE b is null
```

Full Cypher query

```
START beer=node:categories(category='beer'),
  nappies=node:categories(category='nappies'),
  xbox=node:products(product='xbox 360')

MATCH (daddy)-[:BOUGHT]->()-[:MEMBER_OF]->(beer),
  (daddy)-[:BOUGHT]->()-[:MEMBER_OF]->(nappies),
  (daddy)-[b?:BOUGHT]->(xbox)

WHERE b is null

RETURN distinct daddy
```

Results

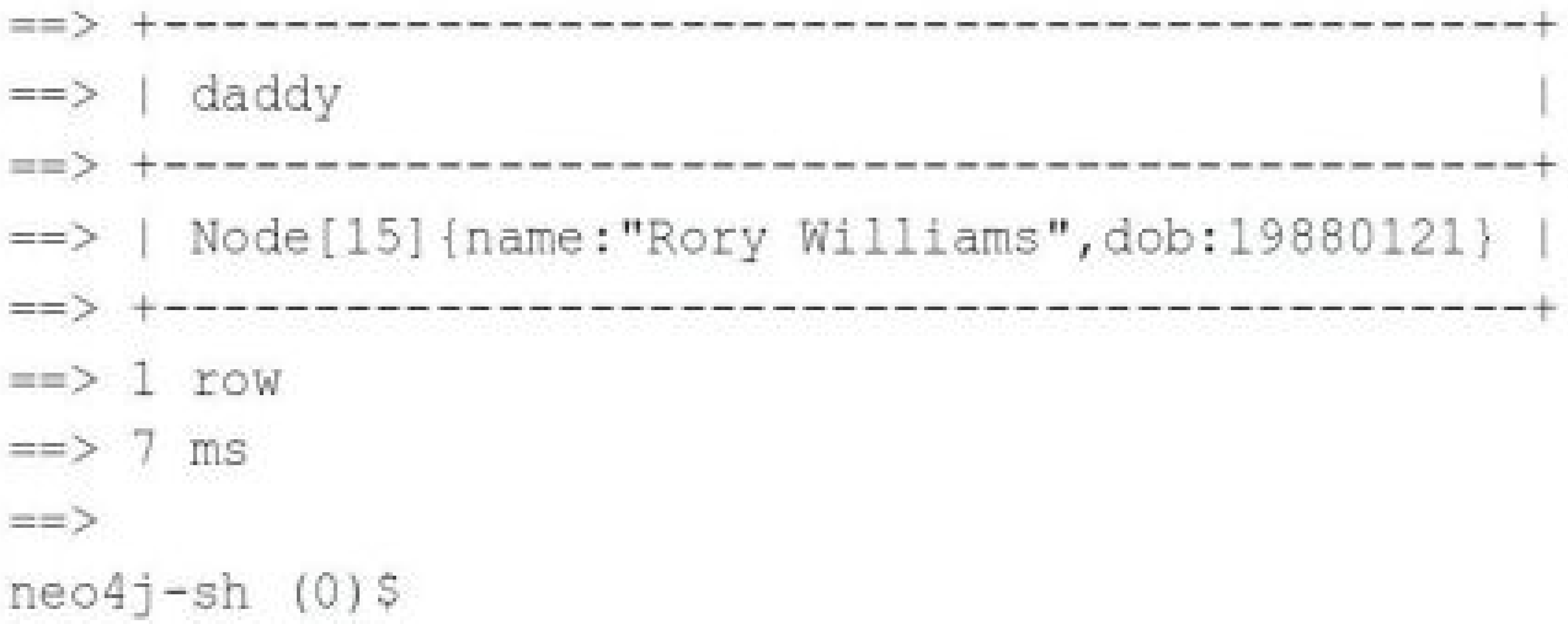
```
==> +---------------------------------------------+
==> | daddy                                       |
==> +---------------------------------------------+
==> | Node[15]{name:"Rory Williams",dob:19880121} |
==> +---------------------------------------------+
==> 1 row
==> 7 ms
==>
neo4j-sh (0)$
```

http://console.neo4j.org/?id=dptxq8

Full Cypher query

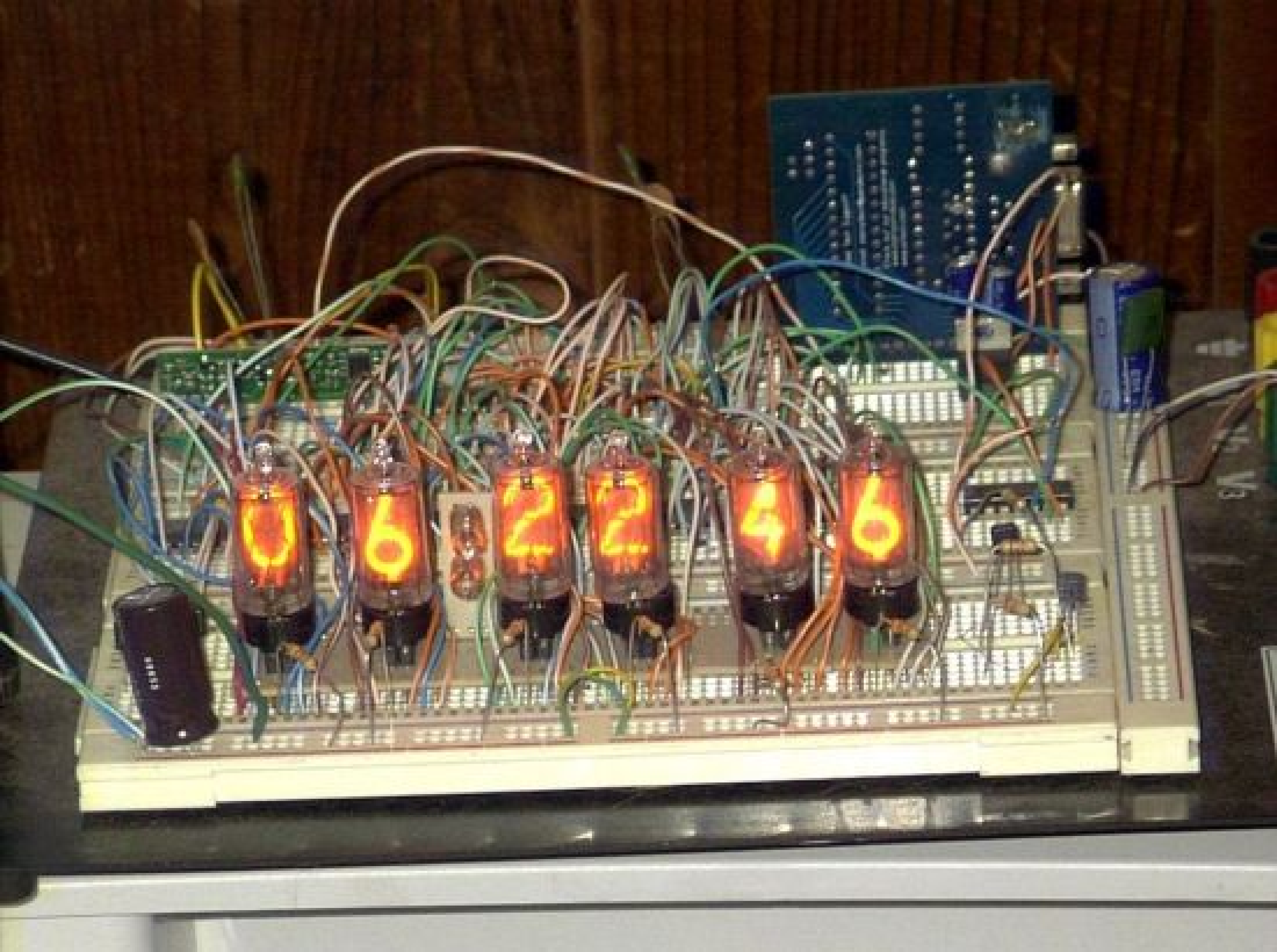
```
START beer=node:categories(category='beer'),
  nappies=node:categories(category='nappies'),
  xbox=node:products(product='xbox 360')

MATCH (daddy)-[:BOUGHT]->()-[:MEMBER_OF]->(beer),
  (daddy)-[:BOUGHT]->()-[:MEMBER_OF]->(nappies),
  (daddy)-[b?:BOUGHT]->(xbox)

WHERE b is null

RETURN distinct daddy
```

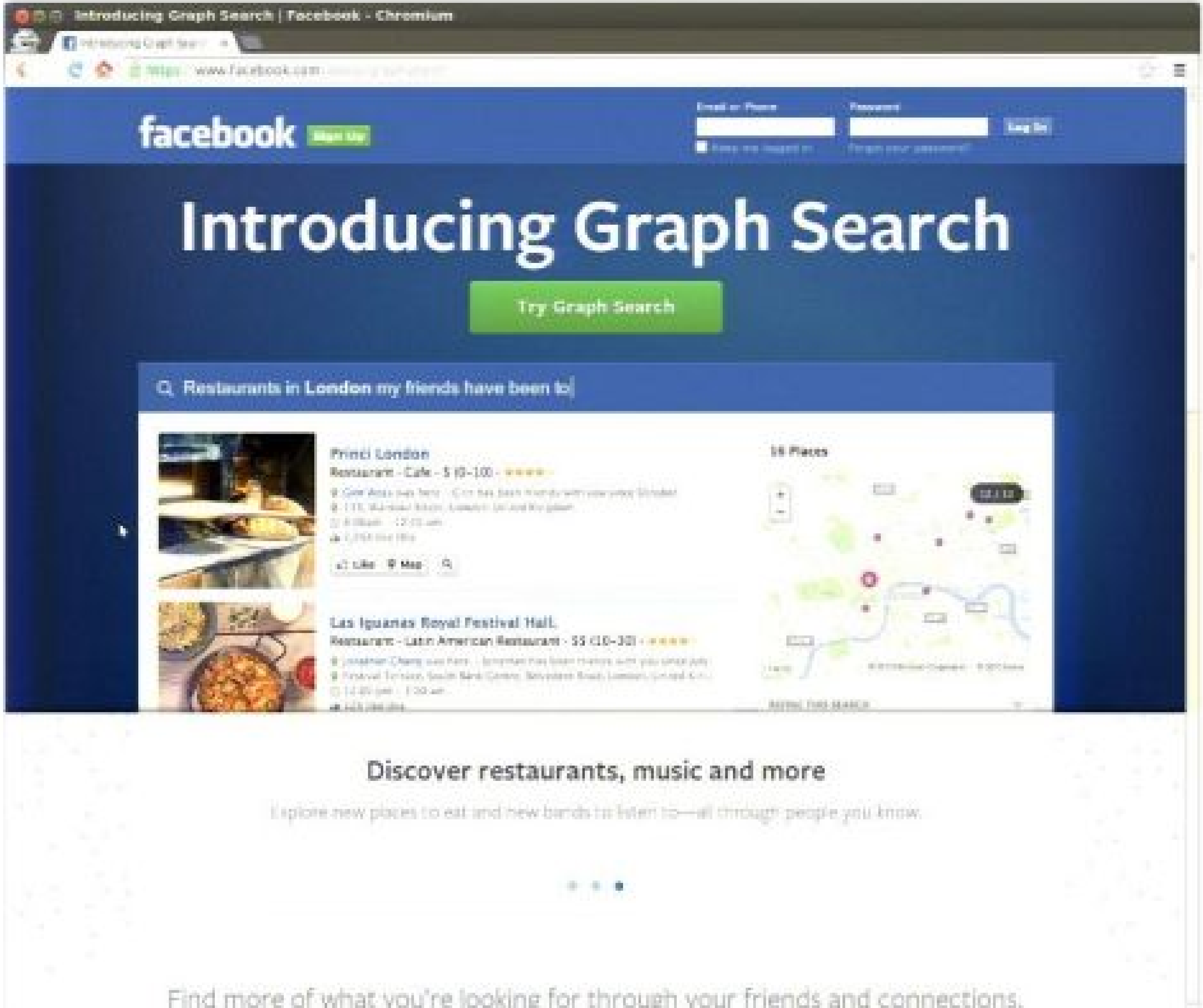
Results



```
==> +---------------------------------------------+
==> | daddy                                       |
==> +---------------------------------------------+
==> | Node[15]{name:"Rory Williams",dob:19880121} |
==> +---------------------------------------------+
==> 1 row
==> 7 ms
==>
neo4j-sh (0)$
```

http://console.neo4j.org/?id=dptxq8


062246



Introducing Graph Search

Restaurants in London my friends have been to

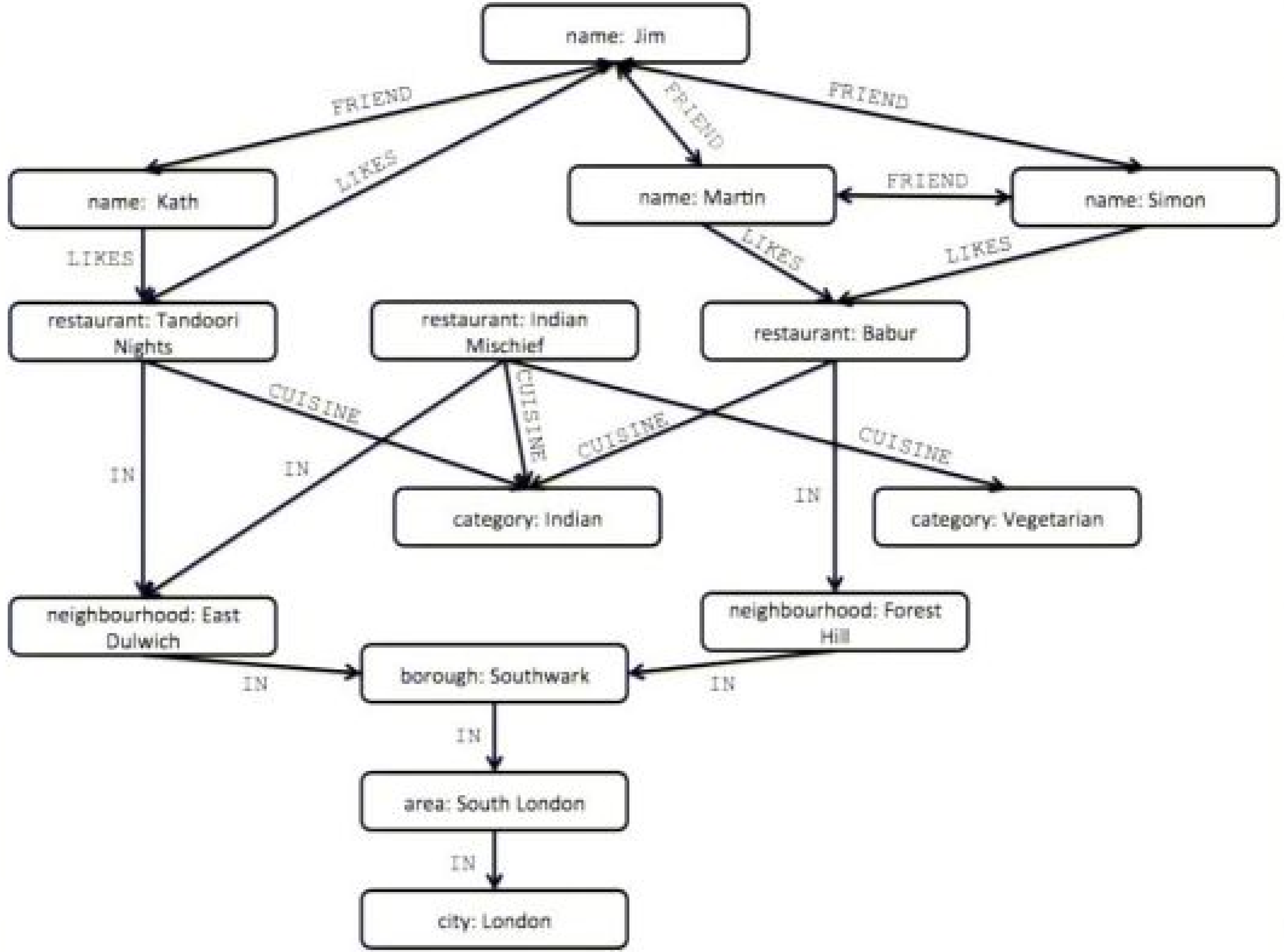
Discover restaurants, music and more

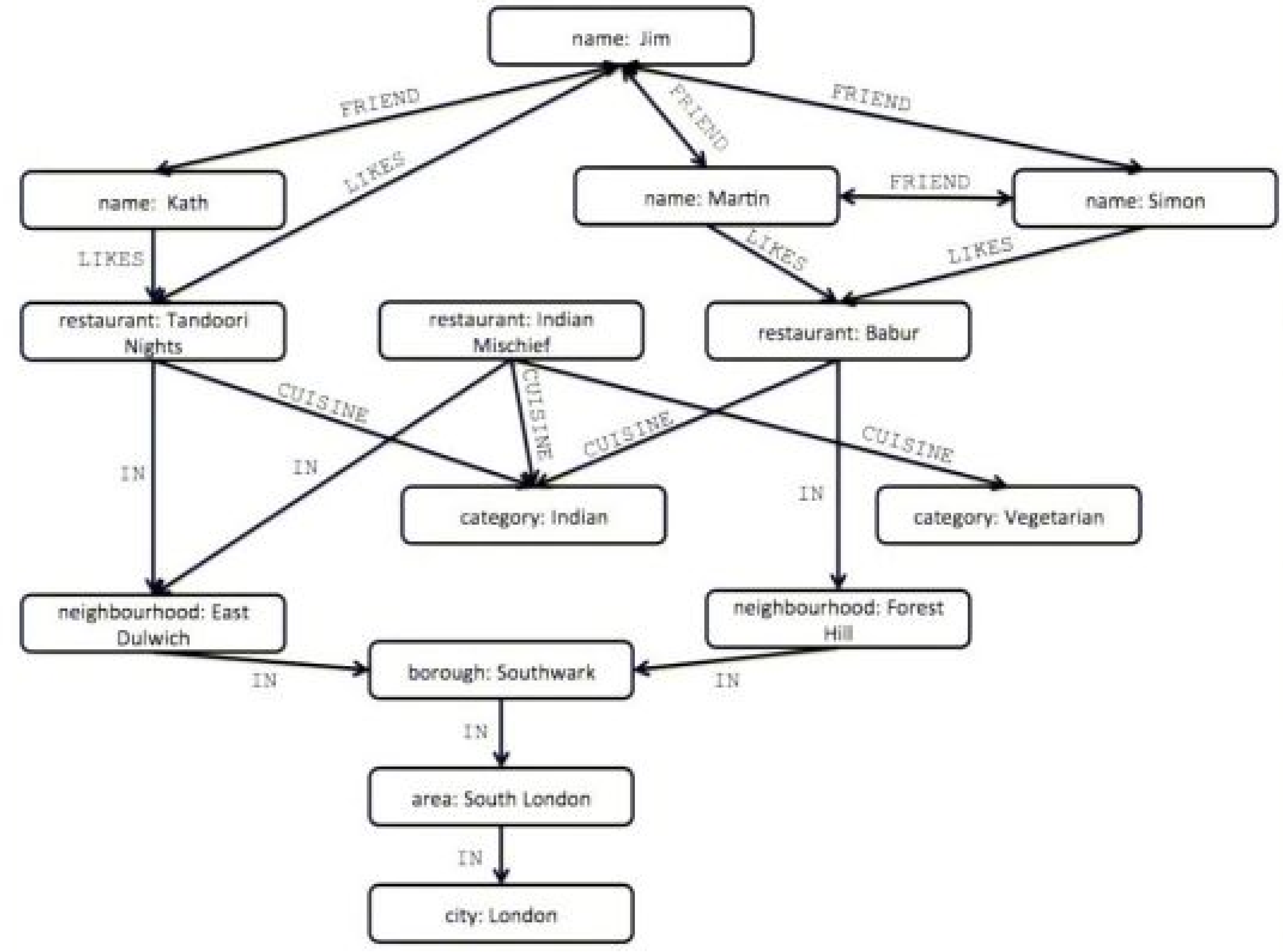
Explore new places to eat and new bands to listen to—all through people you know.

Find more of what you're looking for through your friends and connections.

Facebook Graph Search

„Find indian restaurants in Southwark which my friends like.“


name: Jim
FRIEND
FRIEND
FRIEND
LIKES
name: Kath
name: Martin
FRIEND
name: Simon
LIKES
LIKES
LIKES
restaurant: Tandoori Nights
restaurant: Indian Mischief
restaurant: Babur
CUISINE
CUISINE
CUISINE
CUISINE
IN
IN
IN
category: Indian
category: Vegetarian
neighbourhood: East Dulwich
neighbourhood: Forest Hill
IN
borough: Southwark
IN
IN
area: South London
IN
city: London

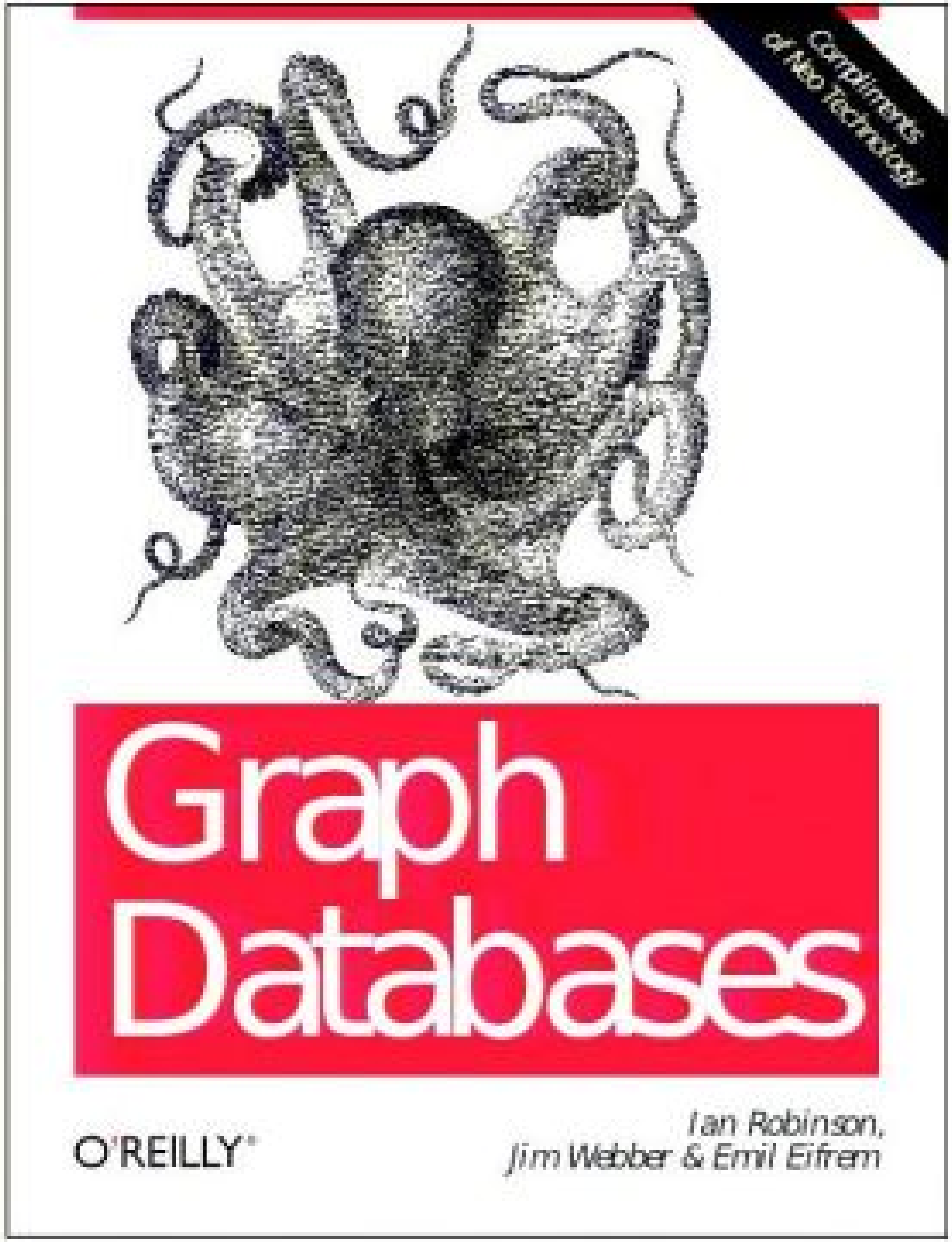

name: Jim
FRIEND
LIKES
FRIEND
FRIEND
name: Kath
name: Martin
FRIEND
name: Simon
LIKES
LIKES
LIKES
restaurant: Tandoori Nights
restaurant: Indian Mischief
restaurant: Babur
CUISINE
CUISINE
CUISINE
CUISINE
IN
IN
IN
category: Indian
category: Vegetarian
neighbourhood: East Dulwich
neighbourhood: Forest Hill
IN
borough: Southwark
IN
IN
area: South London
IN
city: London

FB Graph Search with Cypher

```
START jim = node:node_auto_index(name='Jim'),
      southwark = node:node_auto_index(borough='Southwark'),
      indian = node:node_auto_index(cuisine='Indian')

MATCH jim-[:FRIEND]->friend,
         friend-[:LIKES]->restaurant,
         restaurant-[:IN]->()-[:IN]->southwark,
         restaurant-[:CUISINE]->indian

RETURN restaurant
```

What are graphs good for?

- Recommendations
- Business intelligence
- Social computing
- Geospatial
- MDM
- Data centre management
- Web of things
- Genealogy
- Time series data
- Product catalogue
- Web analytics
- Scientific computing (especially bioinformatics)
- Indexing your *slow* RDBMS
- And much more!



Free O’Reilly eBook!

Visit:
http://GraphDatabases.com

Thank you!

Neo4j: http://neo4j.org

Me: @darthvader42

What are graphs good for?

- Recommendations
- Business intelligence
- Social computing
- Geospatial
- MDM
- Data centre management
- Web of things
- Genealogy
- Time series data
- Product catalogue
- Web analytics
- Scientific computing (especially bioinformatics)
- Indexing your *slow* RDBMS
- And much more!