The Challenge of Connected Data

Jim Webber

Chief Scientist, Neo Technology

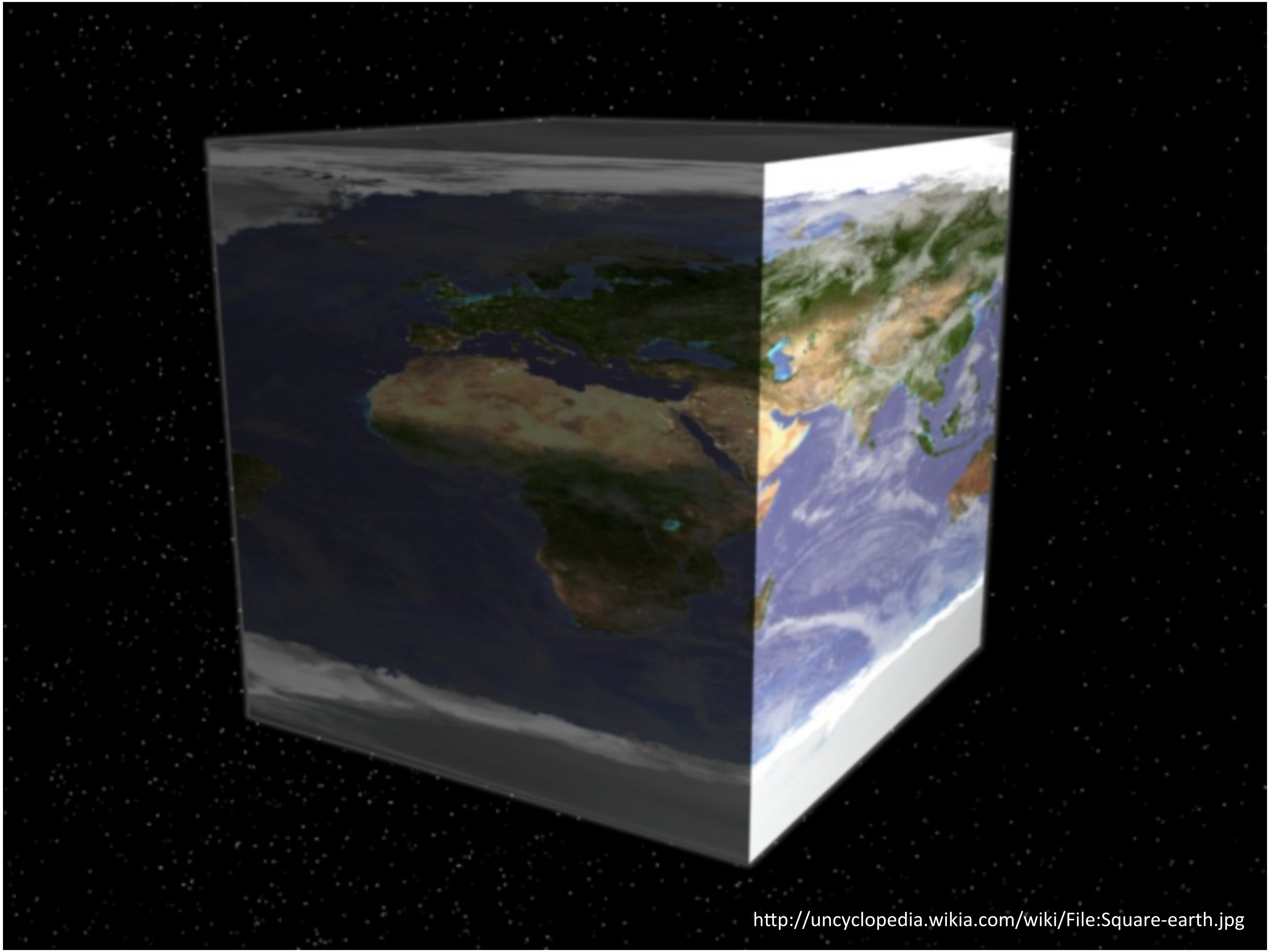
@jimwebber

http://slantmagazine.com/film/review/saturday-night-fever/4210

http://freenewfoundlandlabrador.blogspot.com/2011/07/nl-fisheries-500-years-of-turmoil.html

IBM®

http://www.oopsla.org/oopsla2008/support



http://uncyclopedia.wikia.com/wiki/File:Square-earth.jpg

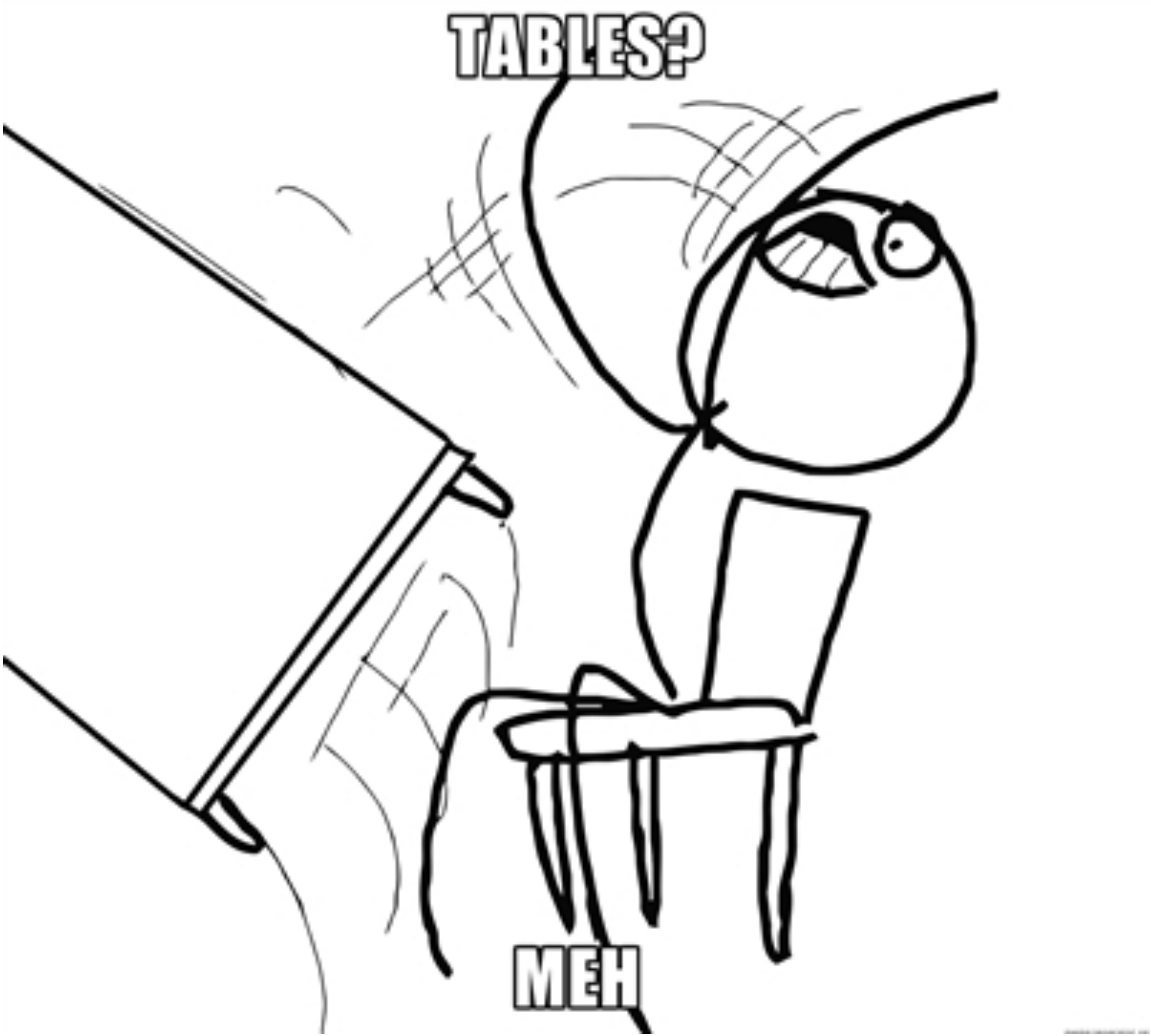
http://www.telegraph.co.uk/finance/newsbysector/industry/8470355/The-fall-and-rise-of-the-British-car-industry-timeline.html

http://www.answers.com/topic/margaret-thatcher-large-image-3

http://www.themoviedb.org/movie/26305

http://iremedy.net/blog/2010/05/your-office-without-boundaries/

Share everything!
Privacy schmivacy!
Social graph!
I’m CEO, bitch!


TABLES?
MEH

http://www.flickr.com/photos/crazyneighborlady/355232758/

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

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

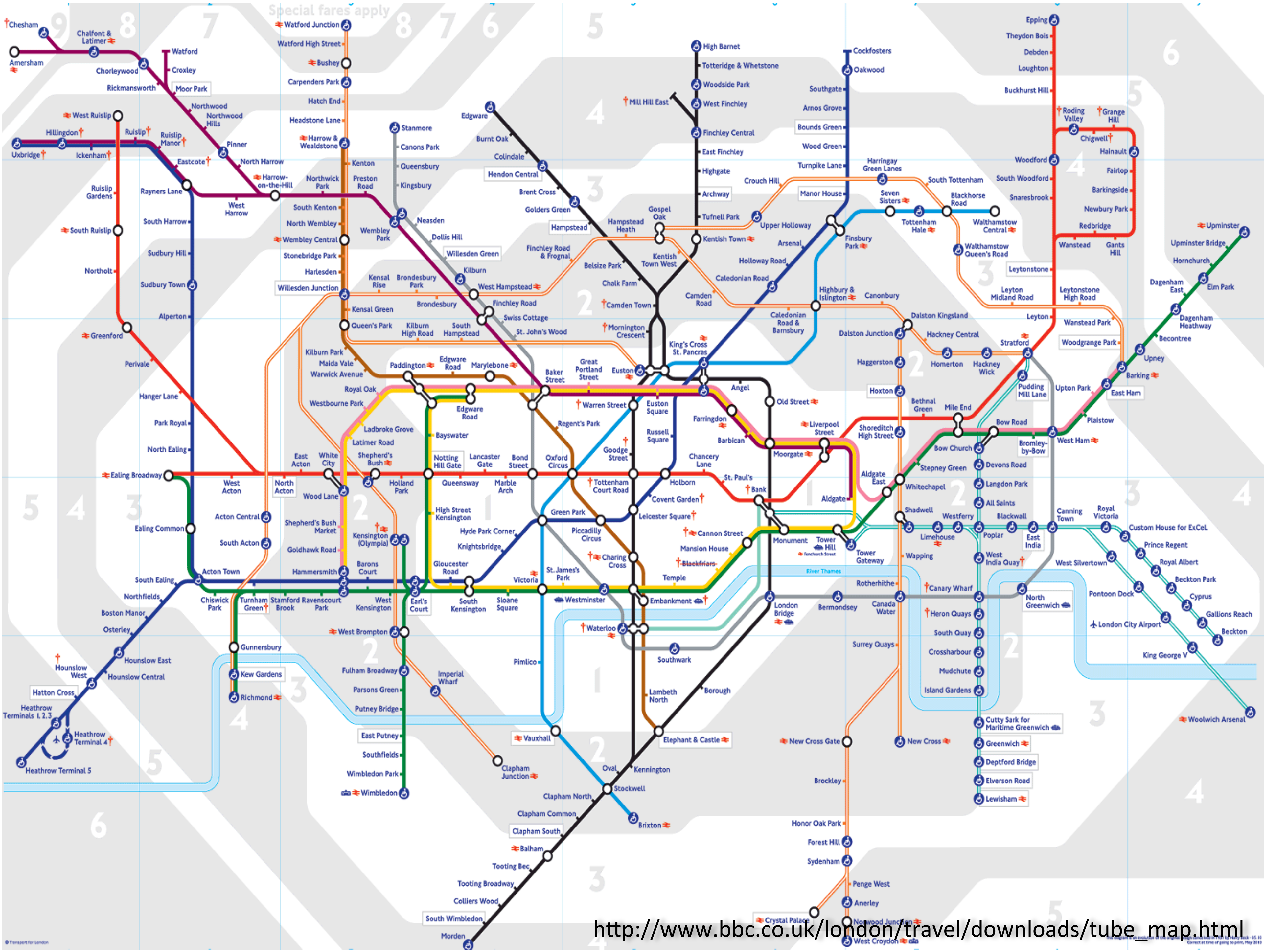
Aggregate-Oriented Databases

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

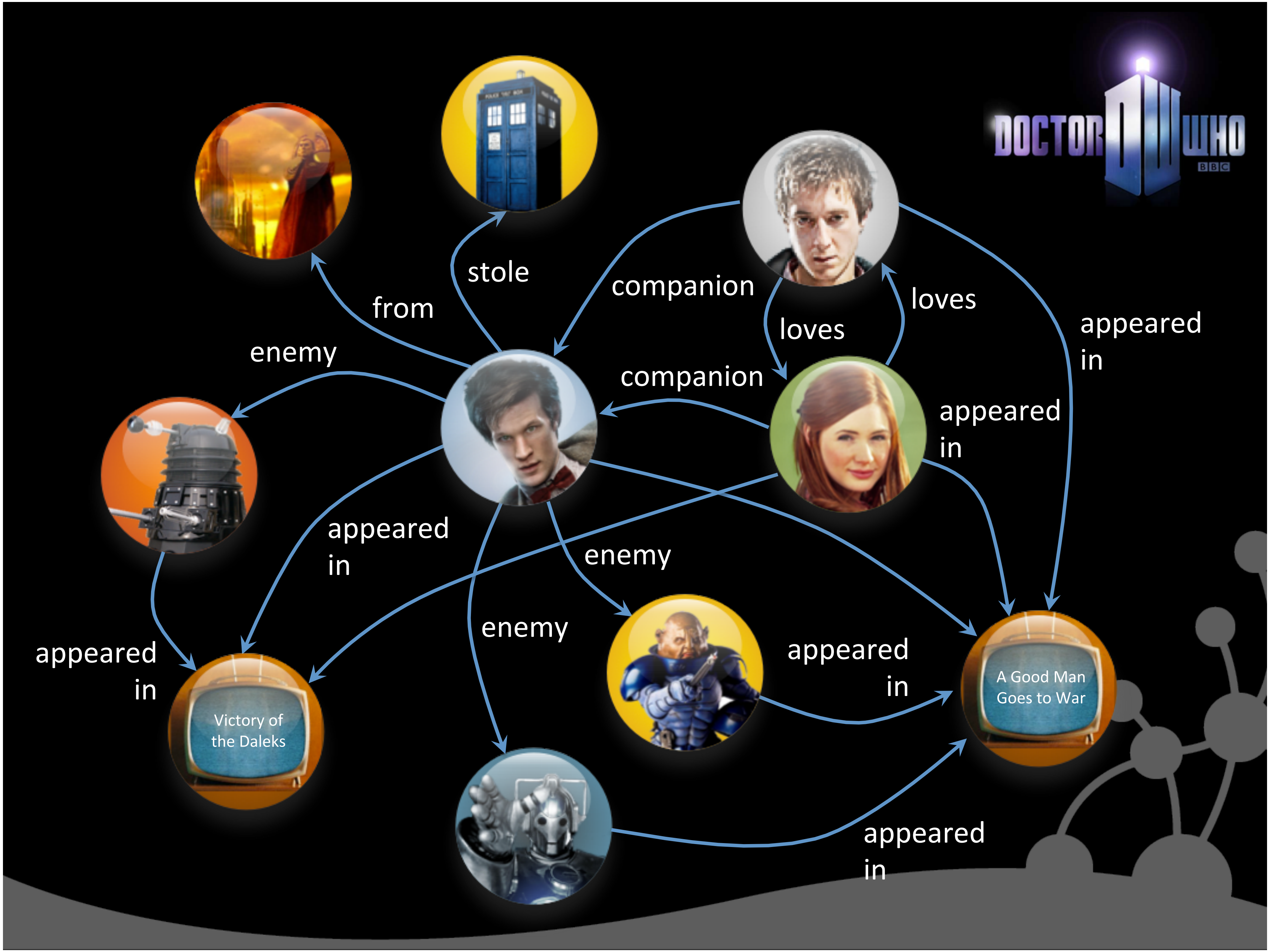
“There is a significant downside - the whole approach works really well when data access is aligned with the aggregates, but what if you want to look at the data in a different way? Order entry naturally stores orders as aggregates, but analyzing product sales cuts across the aggregate structure. The 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.”

complexity = f(size, connectedness, uniformity)



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



DOCTOR WHO
BBC
stole
from
companion
loves
loves
appeared in
enemy
companion
appeared in
appeared in
enemy
enemy
appeared in
appeared in
A Good Man Goes to War
Victory of the Daleks
appeared in

Neo4j: Creating Nodes

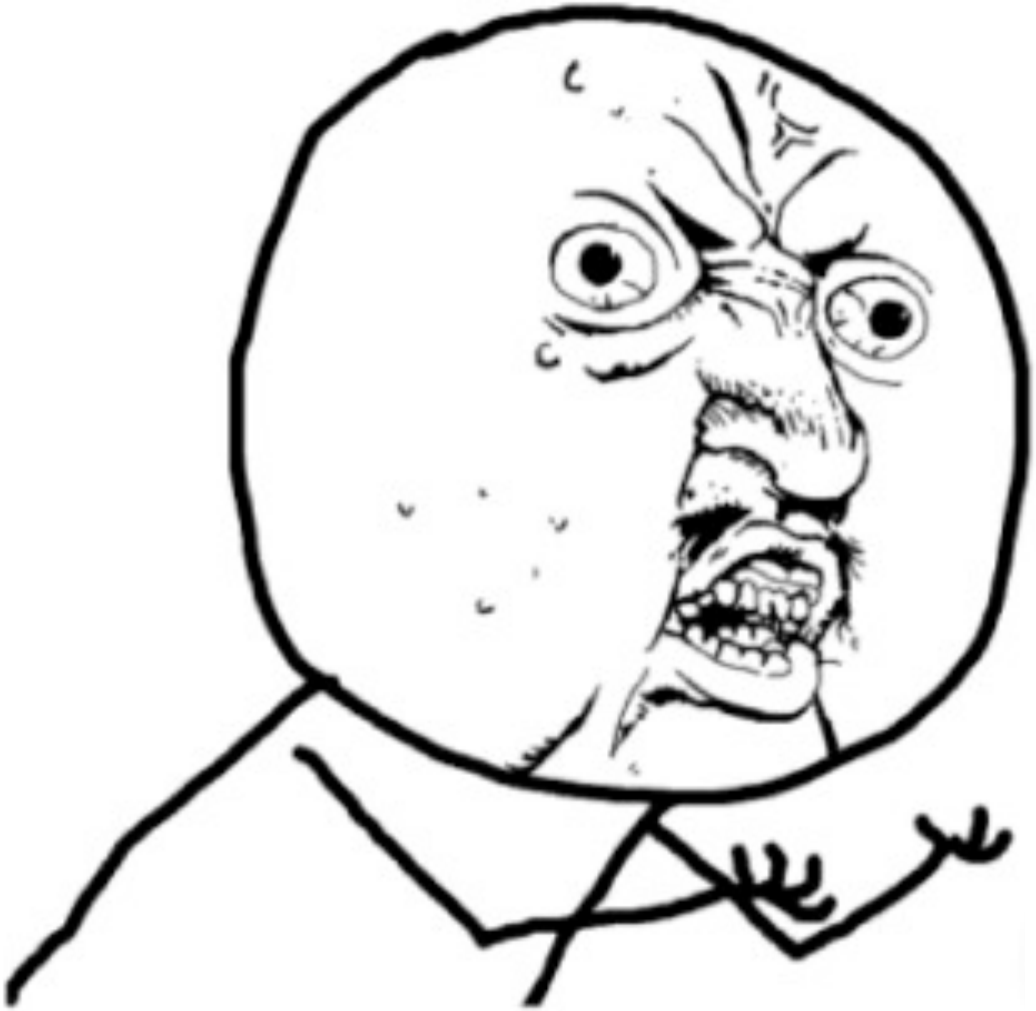
```
GraphDatabaseService db = new
    EmbeddedGraphDatabase("/tmp/neo");
Transaction tx = db.beginTx();
try {
  Node theDoctor = db.createNode();
  theDoctor.setProperty("name", "the Doctor");
  tx.success();
} finally {
  tx.finish();
}
```

ACID TRANSACTIONS
FTW!
memegenerator.net

Neo4j: Creating Relationships

```
Transaction tx = db.beginTx();
try {
  Node theDoctor = db.createNode();
  theDoctor.setProperty("name", "The Doctor");

  Node susan = db.createNode();
  susan.setProperty("firstname", "Susan");
  susan.setProperty("lastname", "Campbell");

  susan.createRelationshipTo(theDoctor, COMPANION_OF));

  tx.success();
} finally {
  tx.finish();
}
```

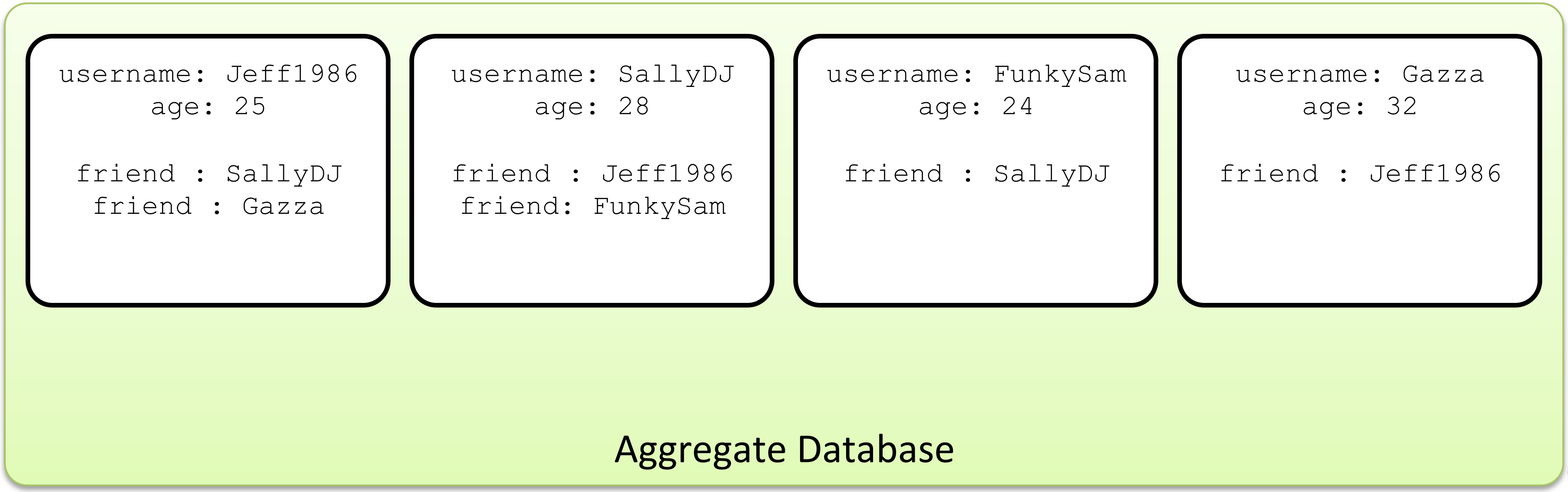
HIPSTER DEVS, Y U NO LIKE JAVA?



Cypher – A Humane “Query” Language

```
// Create a simple graph
CREATE doctor = { character : 'Doctor' },
       master = { character : 'Master' },
       (doctor)<-[e:ENEMY_OF]-(master)

// Add some data into it
SET doctor.awesomeness = 11
SET master.niceness = “very”
SET e.weight = 100


// Ooops!
DELETE master.niceness
SET master.nastiness = “100%”
```

http://www.tolkienlibrary.com/press/922-Isildur_Poker_Champion.php



username: Jeff1986
age: 25

friend : SallyDJ
friend : Gazza

username: SallyDJ
age: 28

friend : Jeff1986
friend: FunkySam

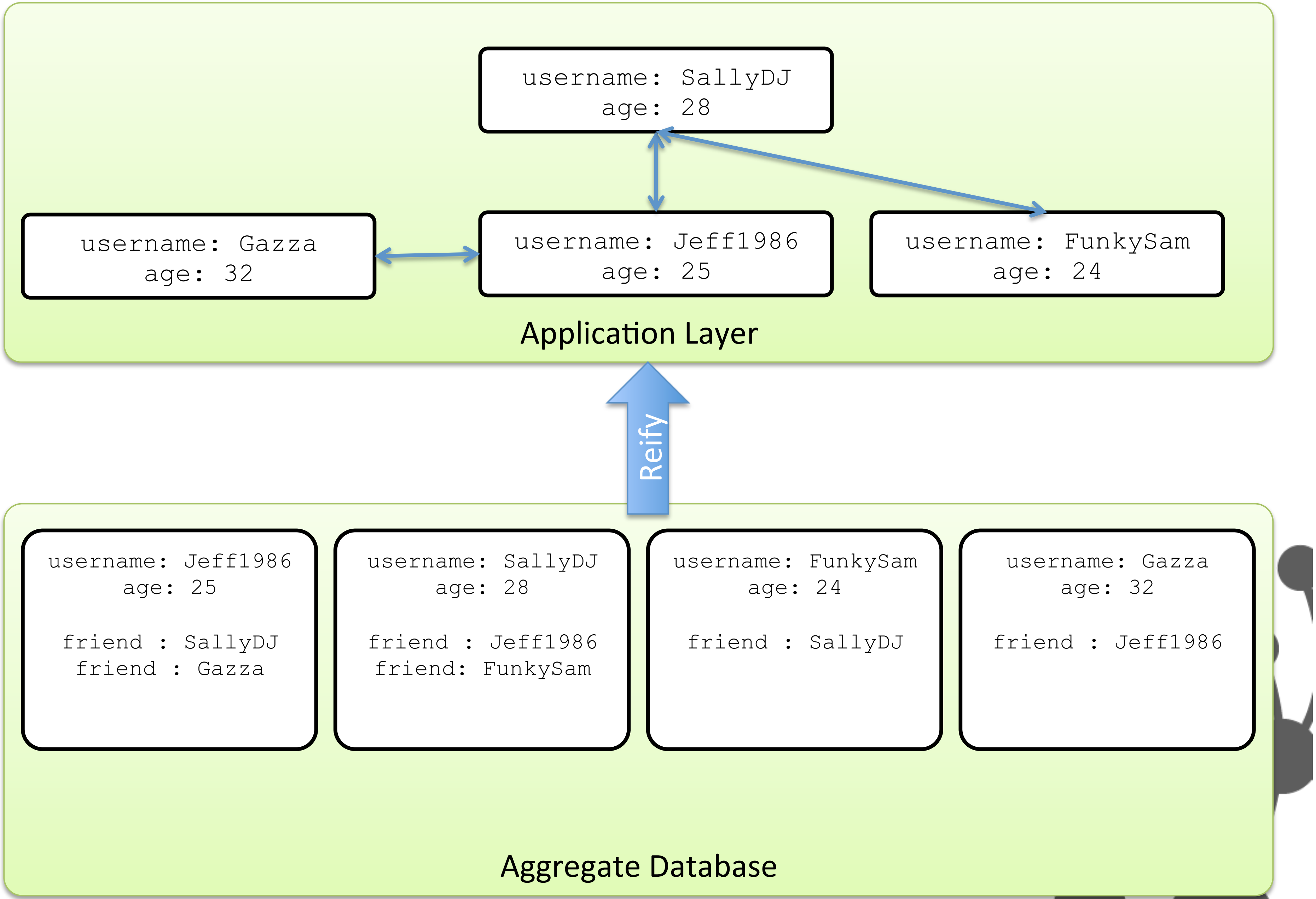
username: FunkySam
age: 24

friend : SallyDJ

username: Gazza
age: 32

friend : Jeff1986

Aggregate Database



username: SallyDJ
age: 28

username: Gazza
age: 32

username: Jeff1986
age: 25

username: FunkySam
age: 24

Application Layer

username: Jeff1986
age: 25

friend : SallyDJ
friend : Gazza

username: SallyDJ
age: 28

friend : Jeff1986
friend: FunkySam

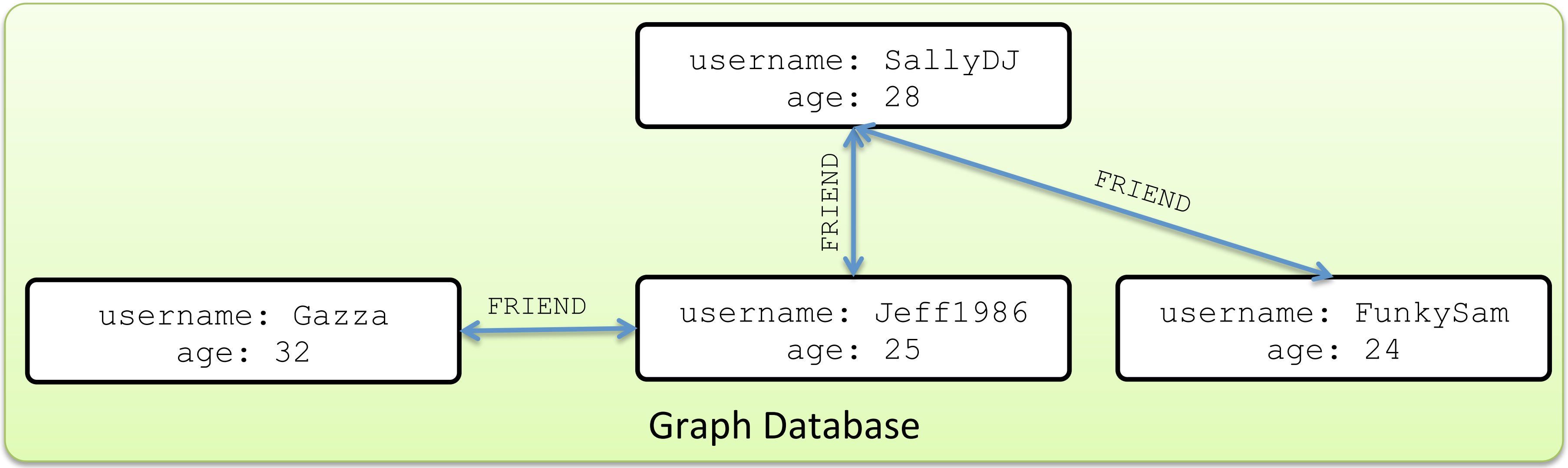
username: FunkySam
age: 24

friend : SallyDJ

username: Gazza
age: 32

friend : Jeff1986

Aggregate Database



username: SallyDJ
age: 28

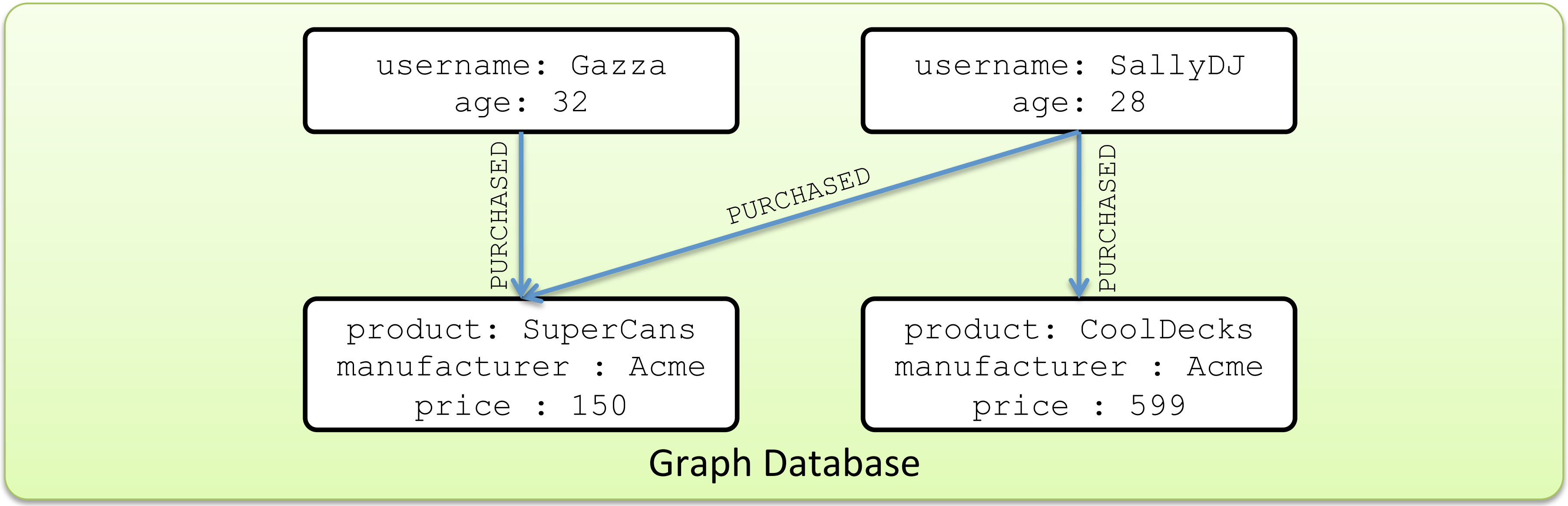
username: Gazza
age: 32

username: Jeff1986
age: 25

username: FunkySam
age: 24

Graph Database

http://www.freewebs.com/fictionfrek101/han.jpg



username: Gazza
age: 32

username: SallyDJ
age: 28

PURCHASED

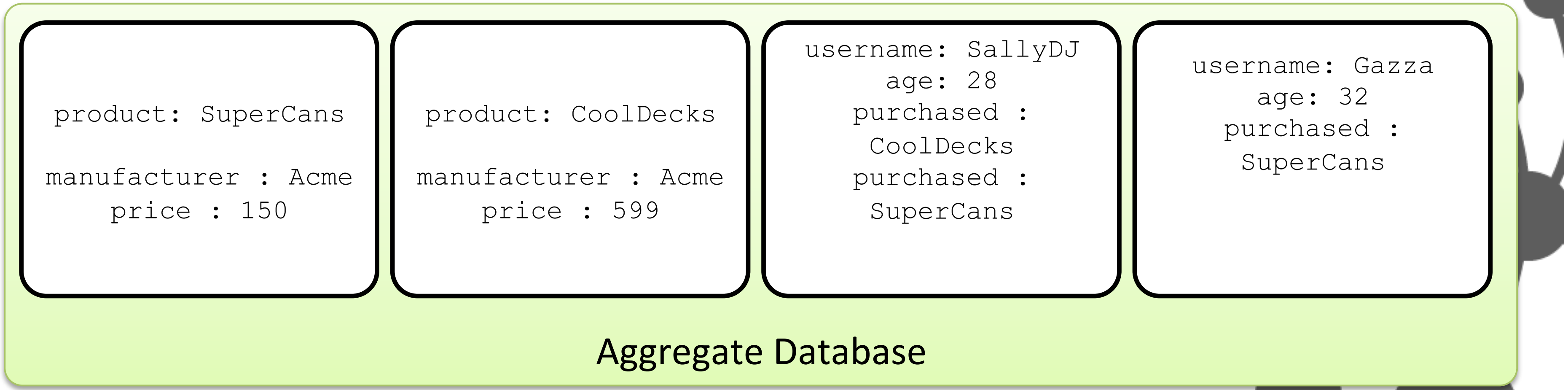
PURCHASED

PURCHASED

product: SuperCans
manufacturer : Acme
price : 150

product: CoolDecks
manufacturer : Acme
price : 599

Graph Database



product: SuperCans

manufacturer : Acme
price : 150

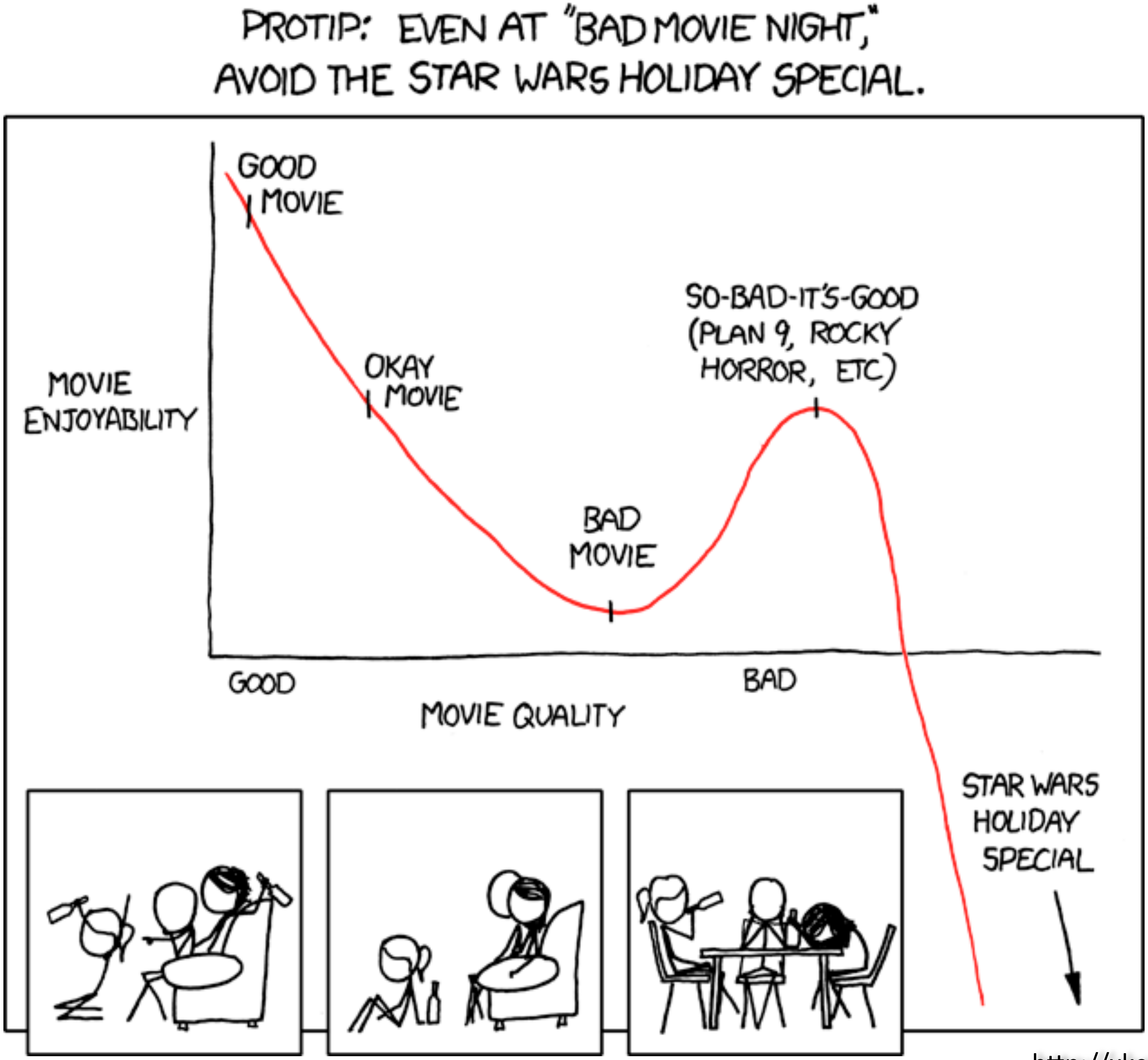
product: CoolDecks

manufacturer : Acme
price : 599

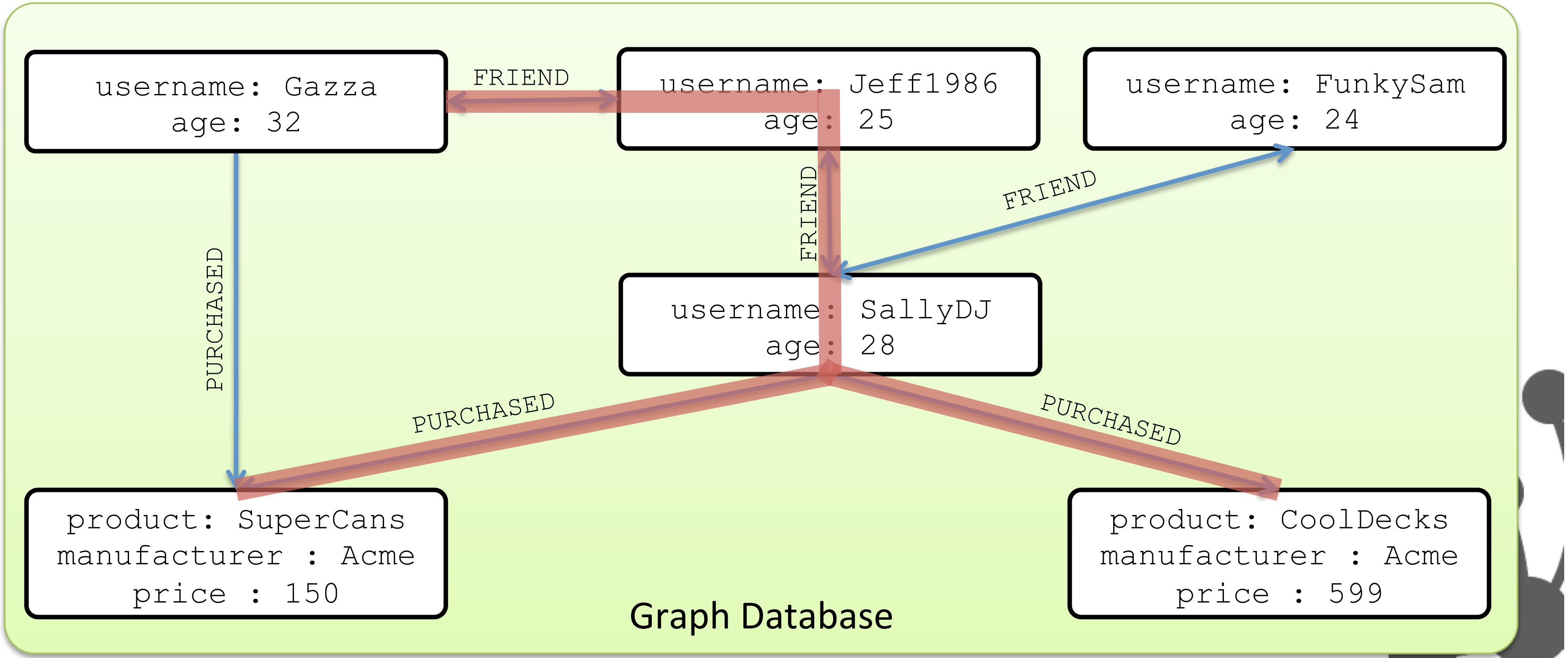
username: SallyDJ
age: 28
purchased :
CoolDecks
purchased :
SuperCans

username: Gazza
age: 32
purchased :
SuperCans

Aggregate Database


PROTIP: EVEN AT "BAD MOVIE NIGHT," AVOID THE STAR WARS HOLIDAY SPECIAL.
MOVIE ENJOYABILITY
MOVIE QUALITY
GOOD
BAD
GOOD MOVIE
OKAY MOVIE
BAD MOVIE
SO-BAD-IT'S-GOOD (PLAN 9, ROCKY HORROR, ETC)
STAR WARS HOLIDAY SPECIAL

http://xkcd.com/653/


username: Gazza
age: 32
FRIEND
username: Jeff1986
age: 25
username: FunkySam
age: 24
FRIEND
FRIEND
PURCHASED
username: SallyDJ
age: 28
PURCHASED
PURCHASED
product: SuperCans
manufacturer : Acme
price : 150
product: CoolDecks
manufacturer : Acme
price : 599
Graph Database

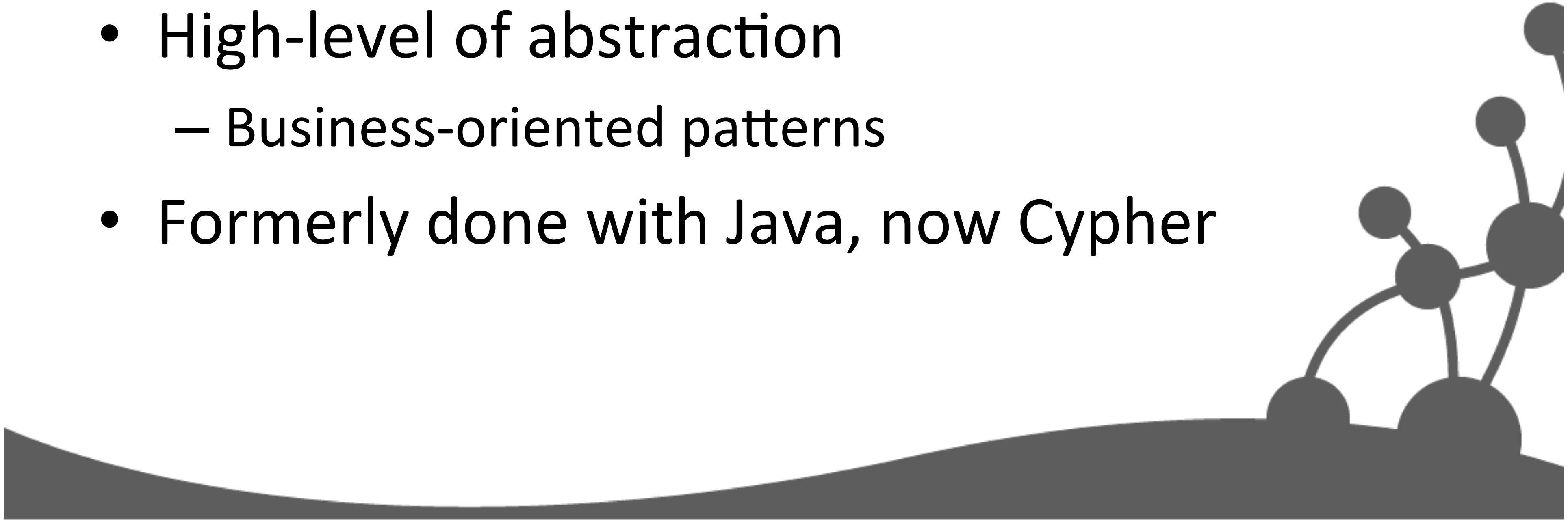
Internet reply

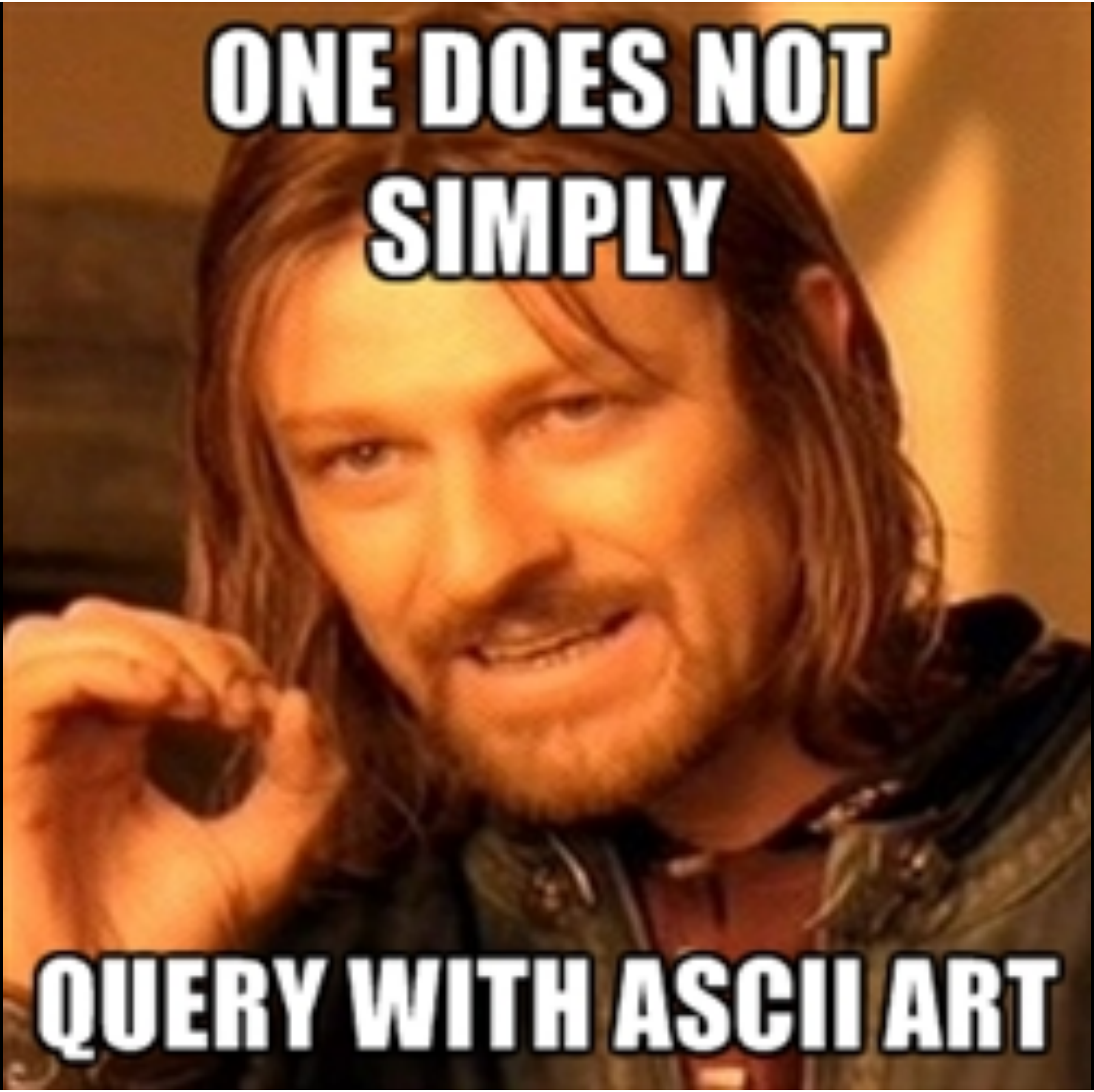
```
>hello-world
segmentation fault (core dumped)

>readnews

alt.binaries.pictures.erotica
rec.arts.startrek.fandom
rec.org.mensa
rec.games.corewar
alt.comics.radioactiveman

>Need know Star
```

http://void.iddqd.cz/comic-book-guy-pc.jpg

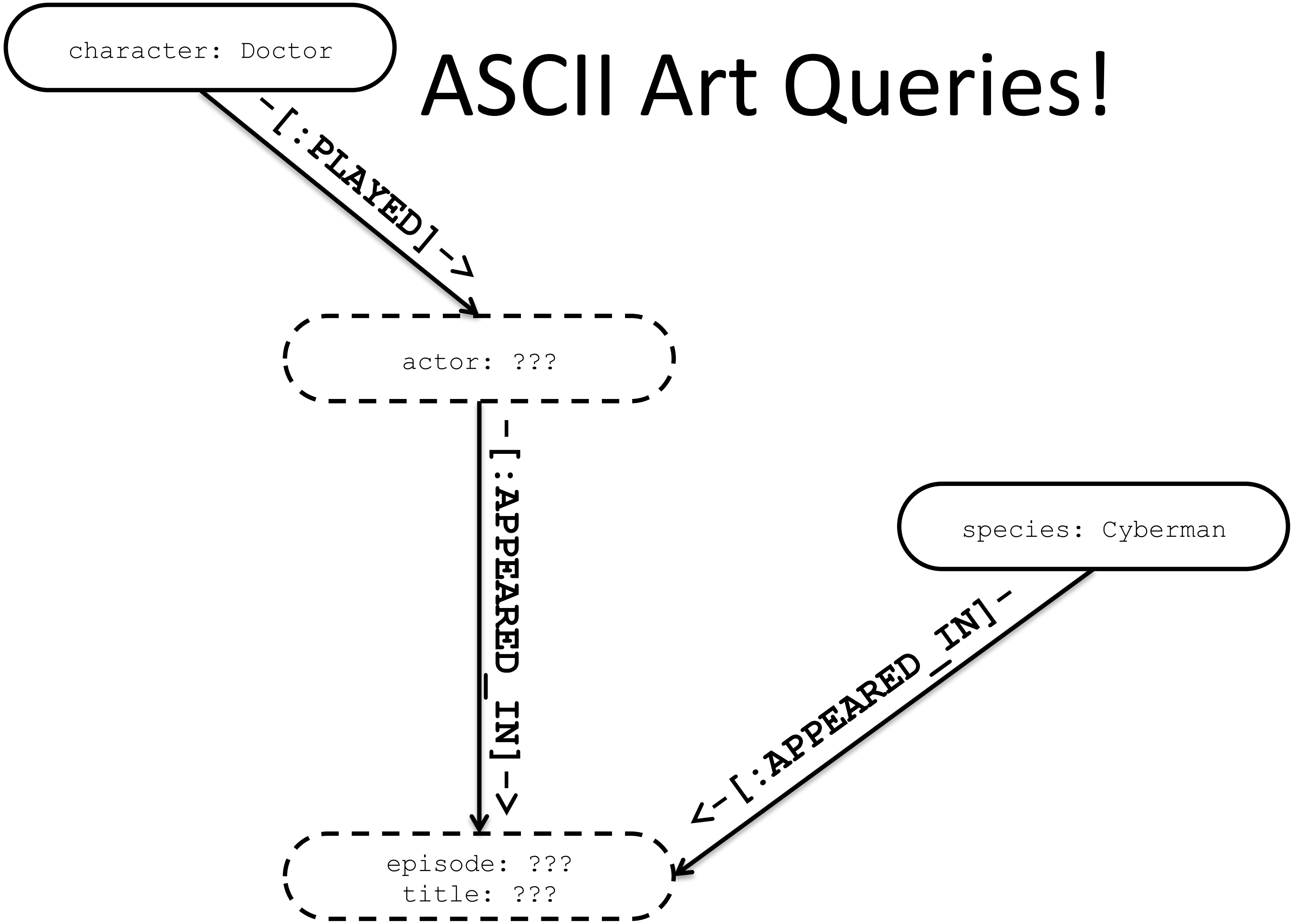
Graph matching

- It's super-powerful for looking for patterns in a data set
  - Retail analytics
  - Real-time upselling
- High-level of abstraction
  - Business-oriented patterns
- Formerly done with Java, now Cypher




ONE DOES NOT SIMPLY
QUERY WITH ASCII ART

ASCII Art Queries!



character: Doctor

-[:PLAYED]->

actor: ???

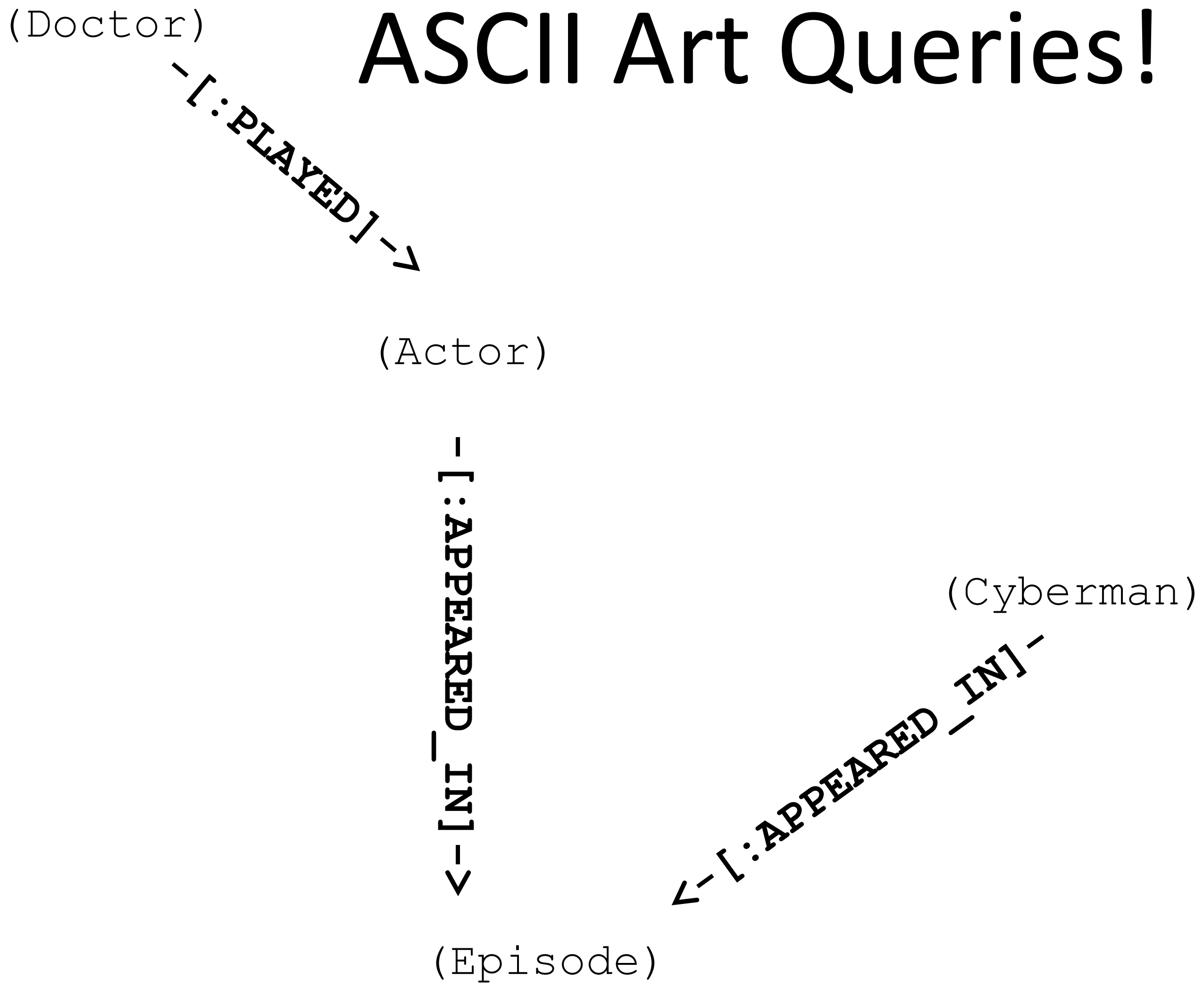
-[:APPEARED_IN]->

species: Cyberman

<-[:APPEARED_IN]-

episode: ???
title: ???

ASCII Art Queries!



(Doctor)

-[:PLAYED]->

(Actor)

-[:APPEARED_IN]->

(Cyberman)

<-[:APPEARED_IN]-

(Episode)

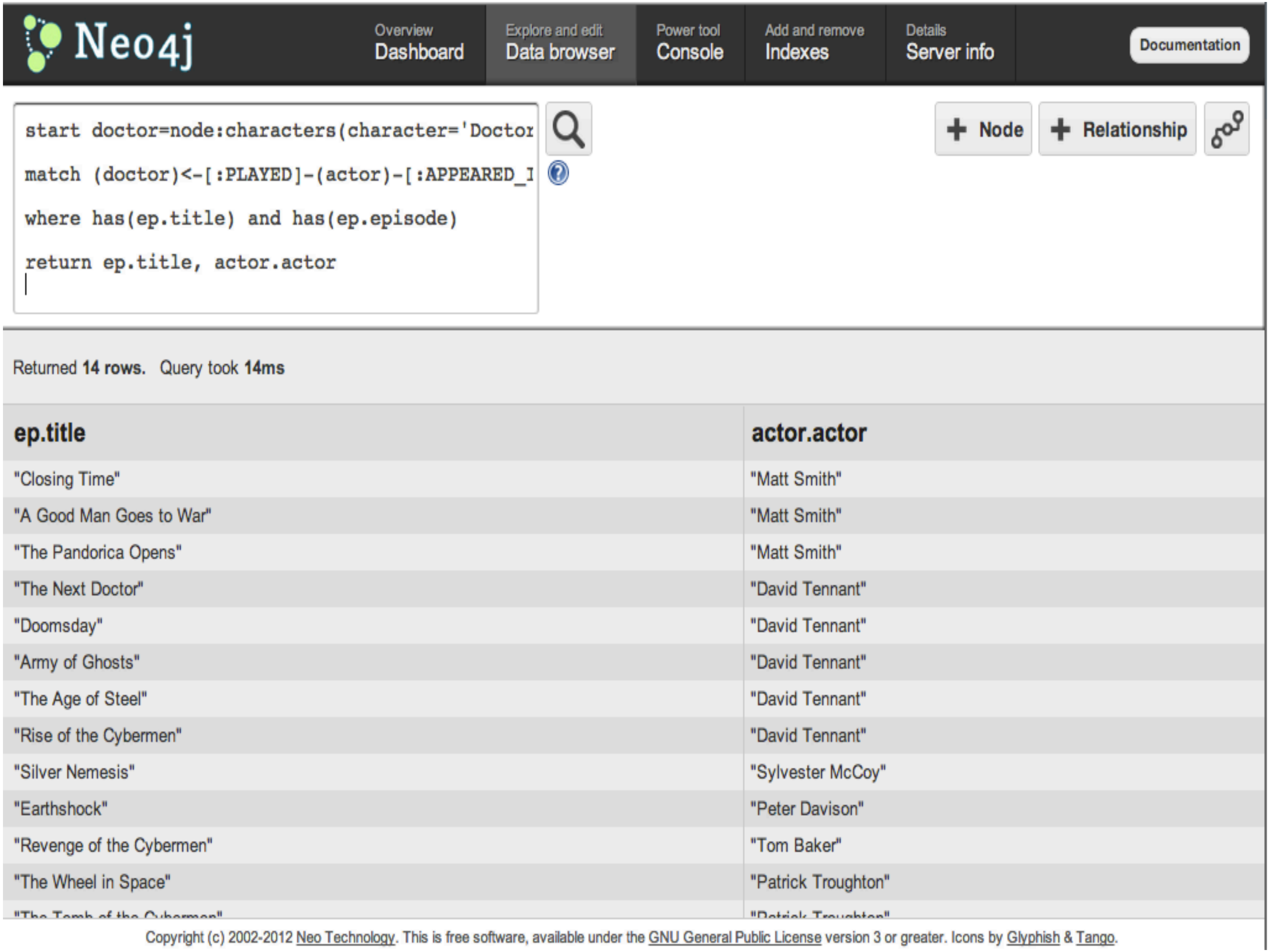
ASCII Art Queries!

```
match (doctor)<-[:PLAYED]-(actor)
       -[:APPEARED_IN]->(ep)
       <-[:APPEARED_IN]-(cybermen)
```

Cypher Query

In which episodes did the cybermen appear, and who played the Doctor?

```
start doctor=node:characters(character='Doctor'),
 cybermen=node:species(species='Cyberman')

match (doctor)<-[:PLAYED]-(actor)
       -[:APPEARED_IN]->(ep)
       <-[:APPEARED_IN]-(cybermen)

where has(ep.title) and has(ep.episode)

return ep.title, actor.actor
```



```
start doctor=node:characters(character='Doctor
match (doctor)<-[:PLAYED]-(actor)-[:APPEARED_I
where has(ep.title) and has(ep.episode)
return ep.title, actor.actor
```

Returned **14 rows.** Query took **14ms**

ep.title	actor.actor
"Closing Time"	"Matt Smith"
"A Good Man Goes to War"	"Matt Smith"
"The Pandorica Opens"	"Matt Smith"
"The Next Doctor"	"David Tennant"
"Doomsday"	"David Tennant"
"Army of Ghosts"	"David Tennant"
"The Age of Steel"	"David Tennant"
"Rise of the Cybermen"	"David Tennant"
"Silver Nemesis"	"Sylvester McCoy"
"Earthshock"	"Peter Davison"
"Revenge of the Cybermen"	"Tom Baker"
"The Wheel in Space"	"Patrick Troughton"
"The Tomb of the Cybermen"	"Patrick Troughton"

Copyright (c) 2002-2012 Neo Technology. This is free software, available under the GNU General Public License version 3 or greater. Icons by Glyphish & Tango.

- Adobe Creative Cloud
  - Gazillions of social
  - Single large global cluster
- Thingdom
  - Node.js, Neo4j
  - Now acquired by FiftyThree
- Cisco network management
  - Nuff said
- And now… Bingo Friendzy
  - Facebook gaming app from Gamesys

NEO4J
ALL THE THINGS!
memegenerator.net

Thanks for listening

Neo4j: **http://neo4j.org**

Neo Technology: **http://neotechnology.com**

Me: **@jimwebber**