Understanding Trolls with Efficient Analytics of Large Graphs in Neo4j

David Allen, Amy E.Hodler, Michael Hunger, Martin Knobloch,
William Lyon, Mark Needham, Hannes Voigt

BTW Rostock
Feb 2019

Michael Hunger

Director Neo4j Labs at Neo4j
@mesirii | michael@neo4j.com

Agenda

1. Graph Databases vs. Graph Processing
2. Neo4j Graph Platform
3. Neo4j Graph Algorithms
4. Application in SNA on Twitter Troll Dataset

Why graphs?

The world is a graph – everything is connected

- people, places, events
- companies, markets
- countries, history, politics
- sciences, art, teaching
- technology, networks, machines, applications, users
- software, code, dependencies, architecture, deployments
- criminals, fraudsters and their behavior

What are people using Neo4j for?

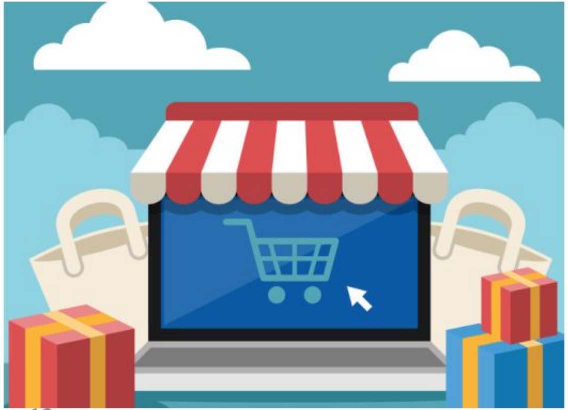
Neo4j - Transforming 100s of Large Enterprises

For Over 14 Years

Real-time promotion recommendations

FORTUNE 50 RETAIL

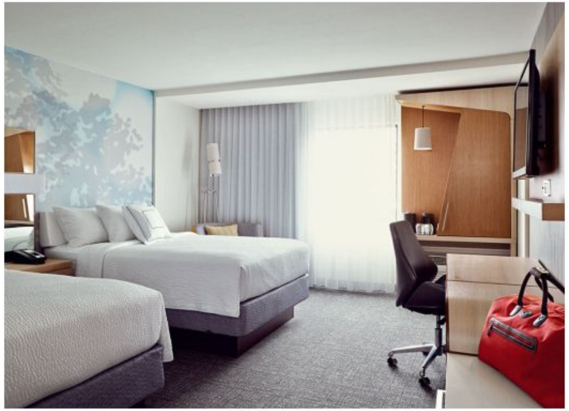
- Record “Cyber Monday” sales
- About 35M daily transactions
- Each transaction is 3-22 hops
- Queries executed in 4ms or less
- Replaced IBM Websphere commerce



Marriott’s Real-time Pricing Engine

MARRIOTT

- 300M pricing operations per day
- 10x transaction throughput on half the hardware compared to Oracle
- Replaced Oracle database



Handling Package Routing in Real-Time

- Large postal service with over 500k employees
- Neo4j routes 7M+ packages daily at peak, with peaks of 5,000+ routing operations per second.

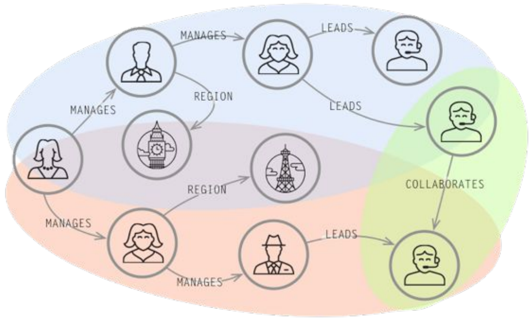
Use Cases

Internal Applications

Master Data Management

Network and
IT Operations

Fraud Detection

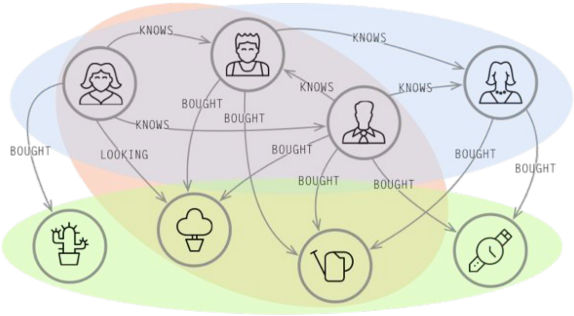


Customer-Facing Applications

Real-Time Recommendations

Graph-Based Search

Identity and
Access Management

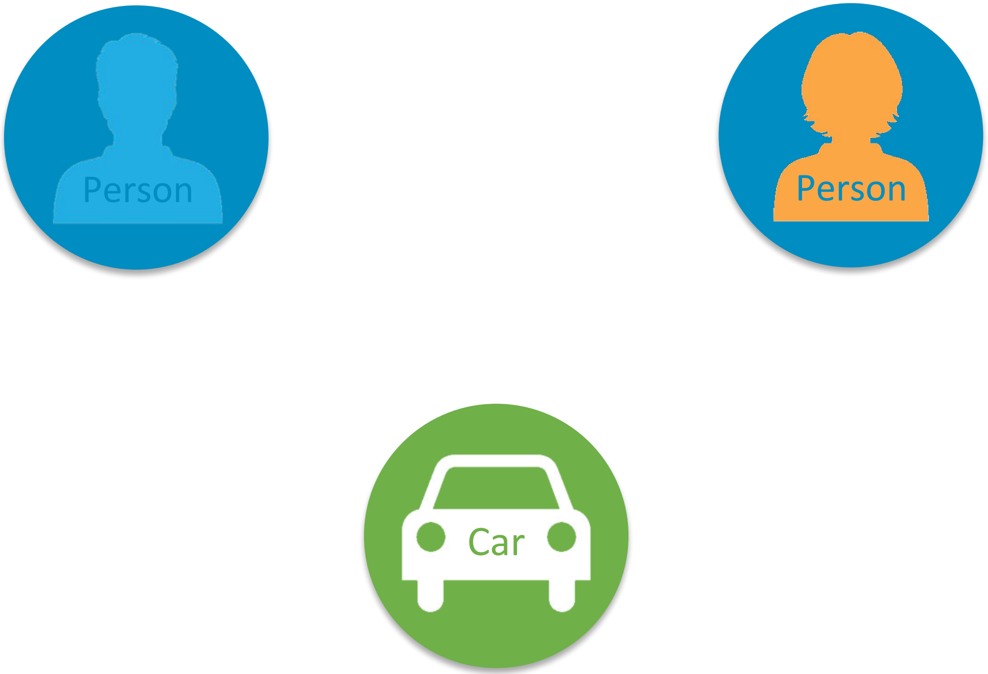


The labeled property graph model

Property Graph Model Components

Nodes

- Represent the objects in the graph
- Can be *labeled*



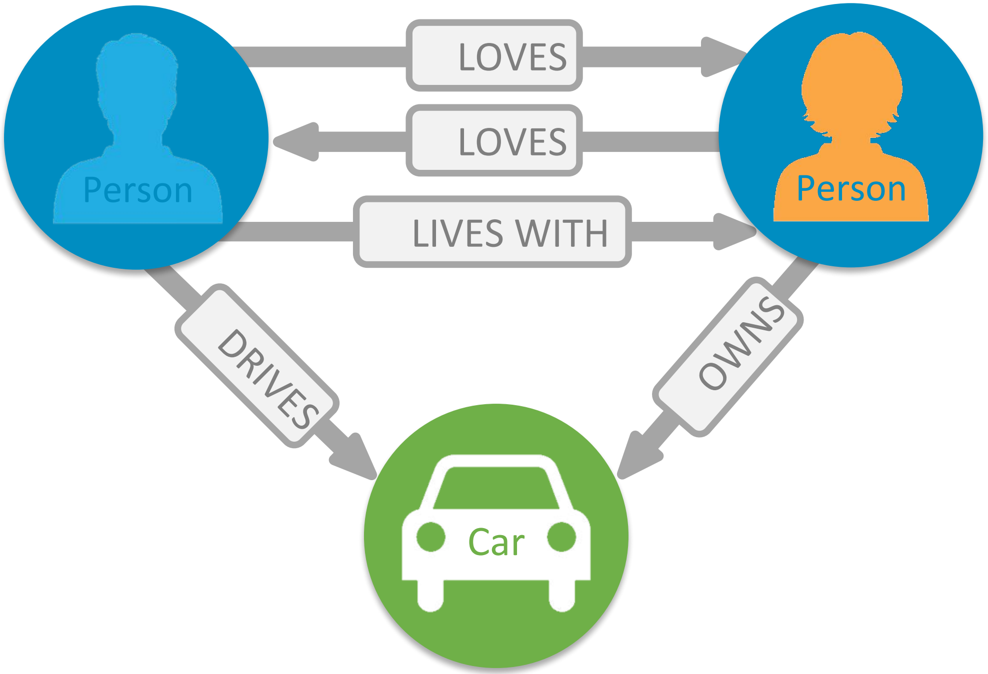
Property Graph Model Components

Nodes

- Represent the objects in the graph
- Can be *labeled*

Relationships

- Relate nodes by *type* and *direction*



Property Graph Model Components

Nodes

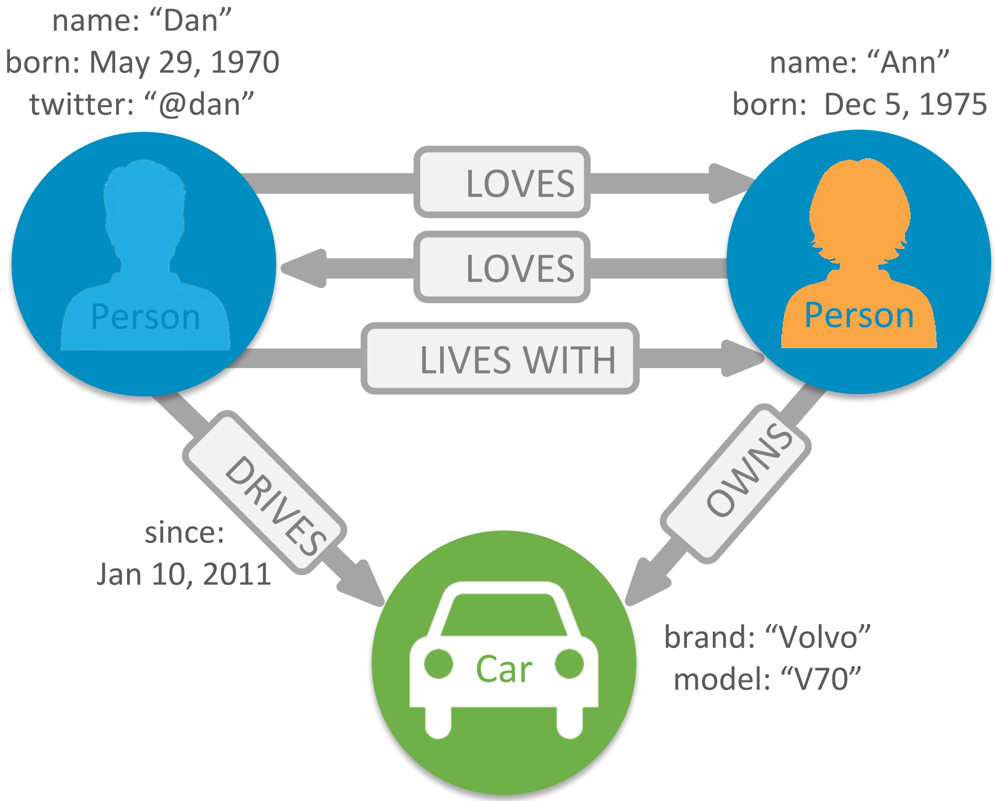
- Represent the objects in the graph
- Can be *labeled*

Relationships

- Relate nodes by *type* and *direction*

Properties

- Name-value pairs that can go on nodes and relationships.

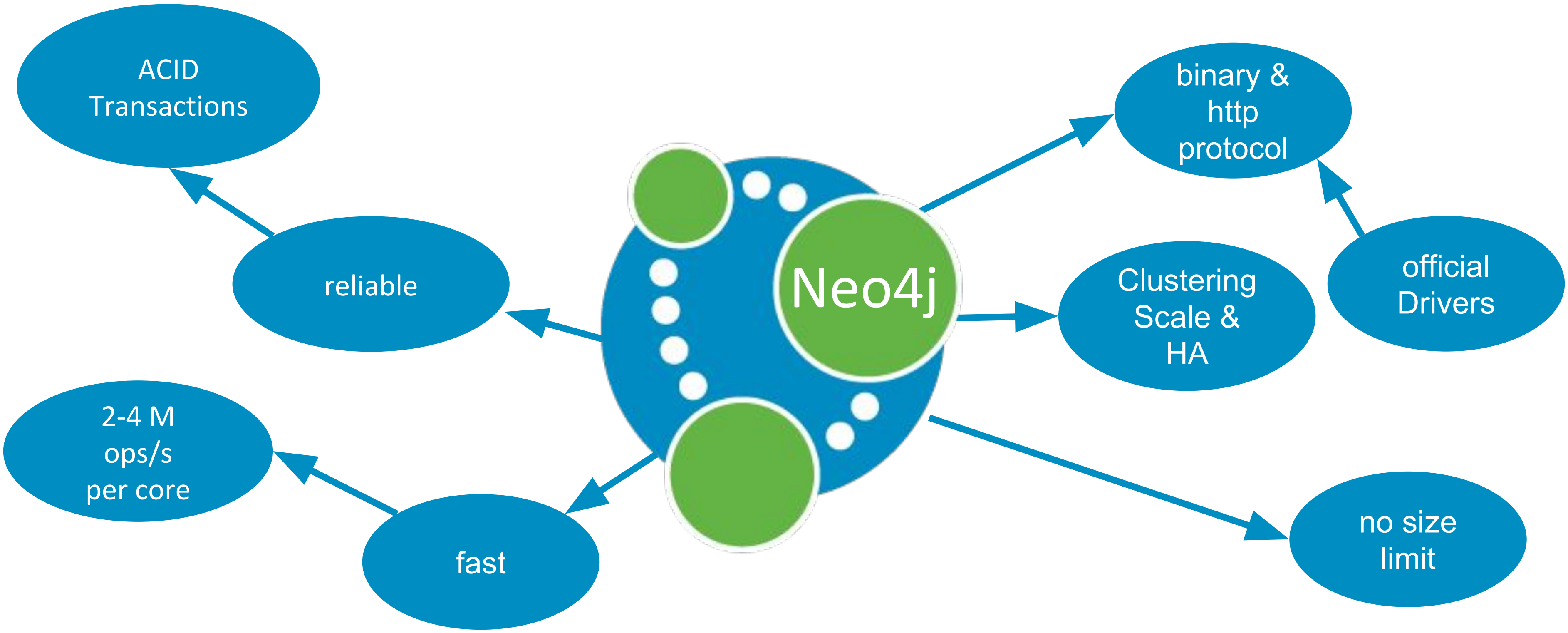


Summary of the graph building blocks

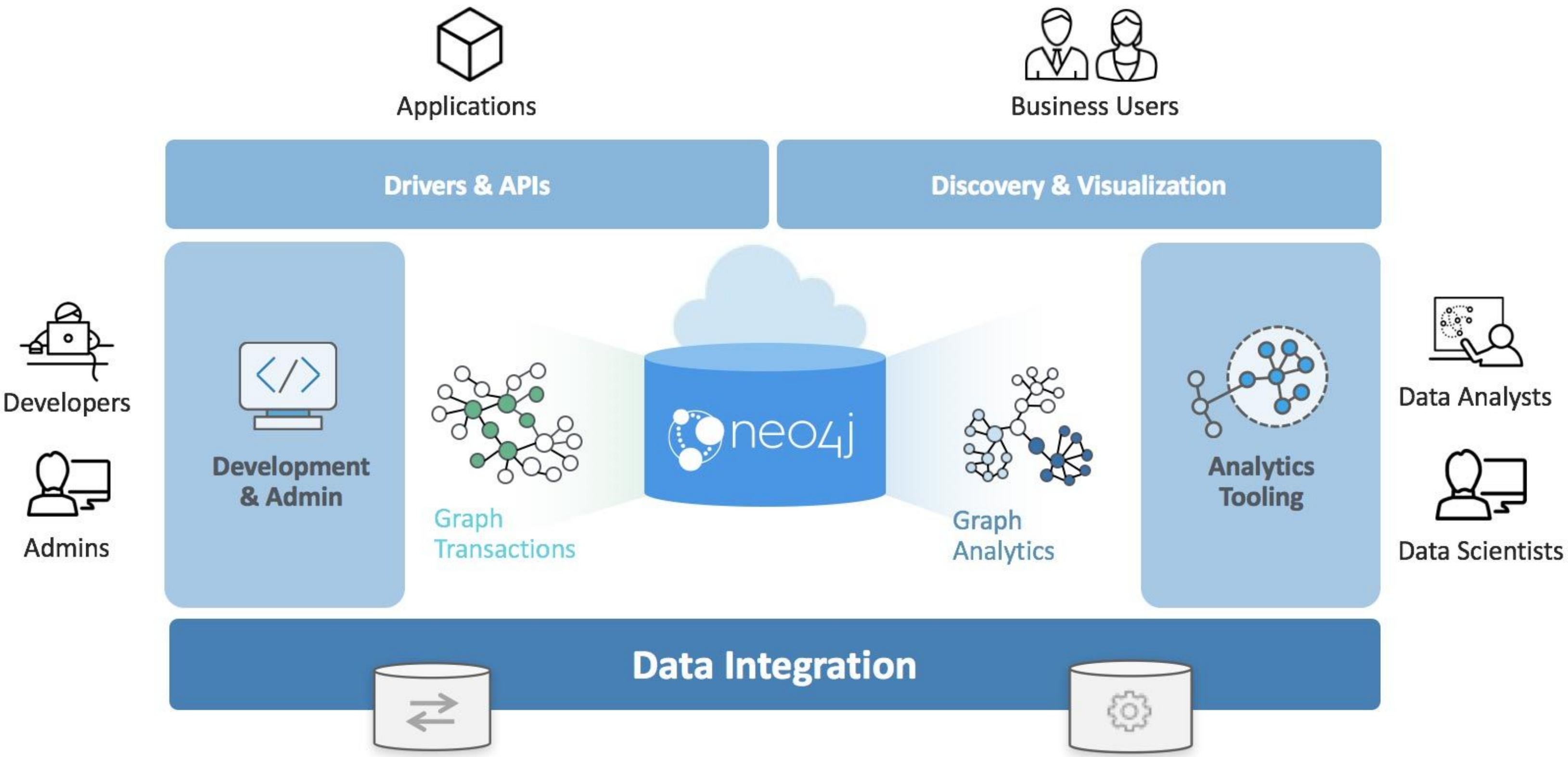
- **Nodes** - Entities and complex value types
- **Relationships** - Connect entities and structure domain
- **Properties** - Entity attributes, relationship qualities, metadata
- **Labels** - Group nodes by role

Neo4j is a Graph Platform

Neo4j is a database



Neo4j is a graph platform



Graph Querying

Cypher

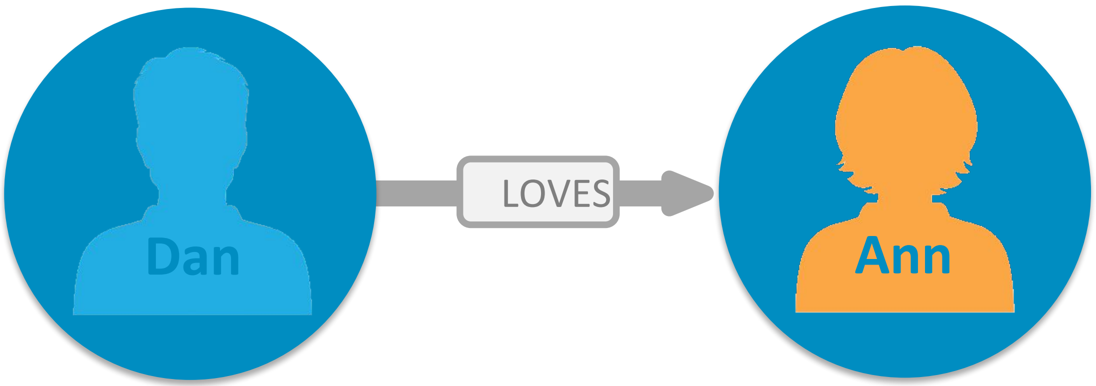
A pattern matching query language made for graphs

- Declarative
- Expressive
- Pattern Matching

Formal specification, SIGMOD paper:

https://homepages.inf.ed.ac.uk/libkin/papers/sigmod18.pdf

Cypher: Express Graph Patterns

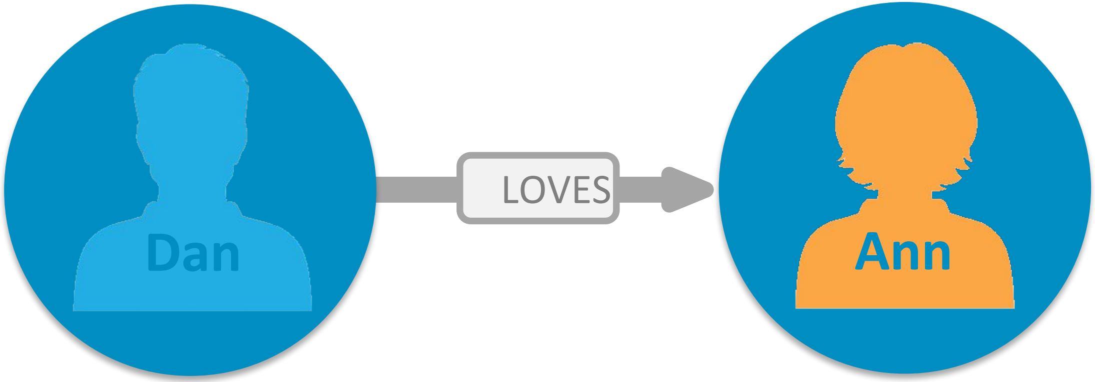


NODE Relationship NODE

(:Person { name:"Dan"}) -[:LOVES]-> (:Person { name:"Ann"})

LABEL PROPERTY LABEL PROPERTY

Cypher: CREATE Graph Patterns

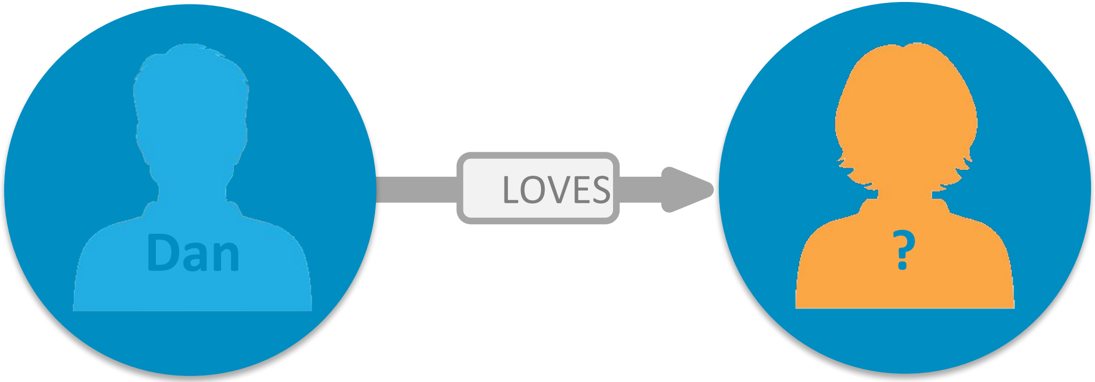


NODE Relationship NODE

CREATE (:Person { name:"Dan"}) -[:LOVES]-> (:Person { name:"Ann"})

LABEL PROPERTY LABEL PROPERTY

Cypher: MATCH Graph Patterns

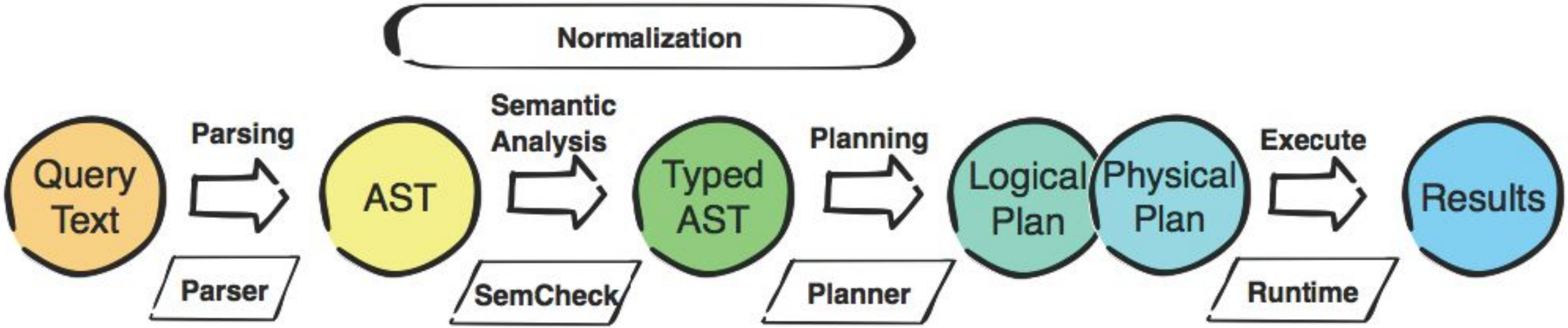


NODE Relationship NODE

MATCH (:Person { name:"Dan"}) -[:LOVES]-> (whom) **RETURN** whom

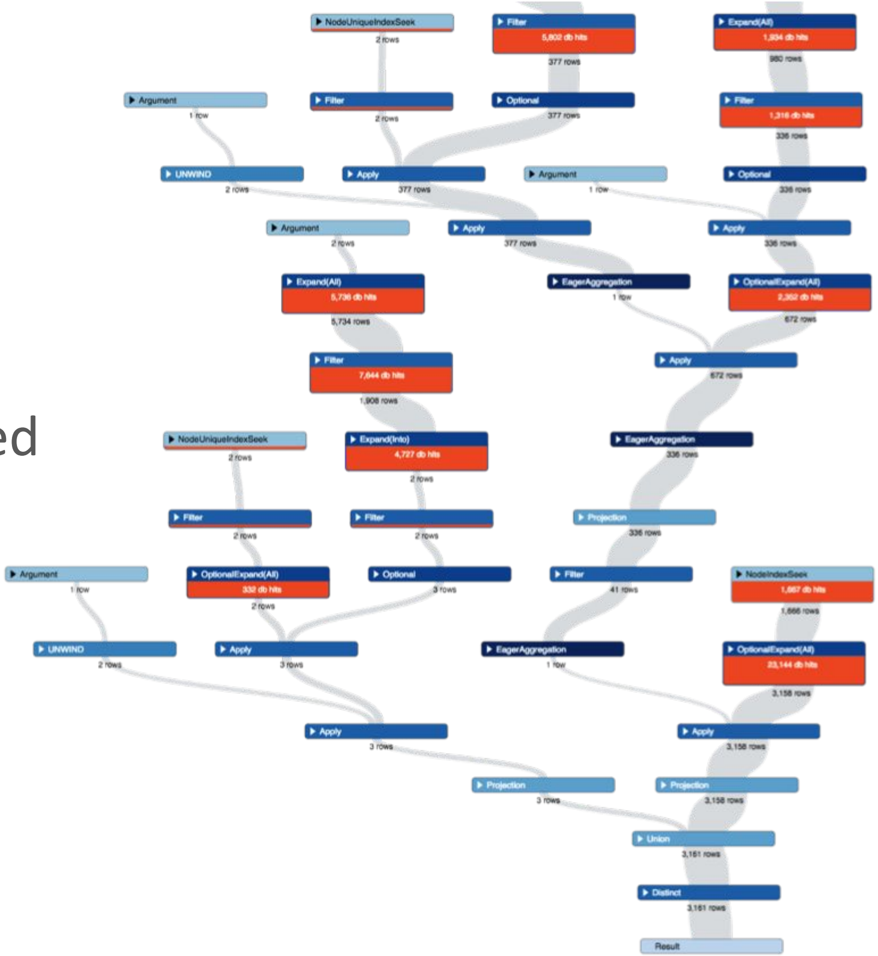
LABEL PROPERTY VARIABLE

Cypher: Query Planner



Cypher: Query Plan

- different planners
- e.g. IDP planner
- different runtimes
- e.g. bytecode compiled



openCypher / GQL

- open source graph query language specification and reference implementation
- Multi-Vendor effort to standardize a Graph Query Language, see: gqlstandards.org

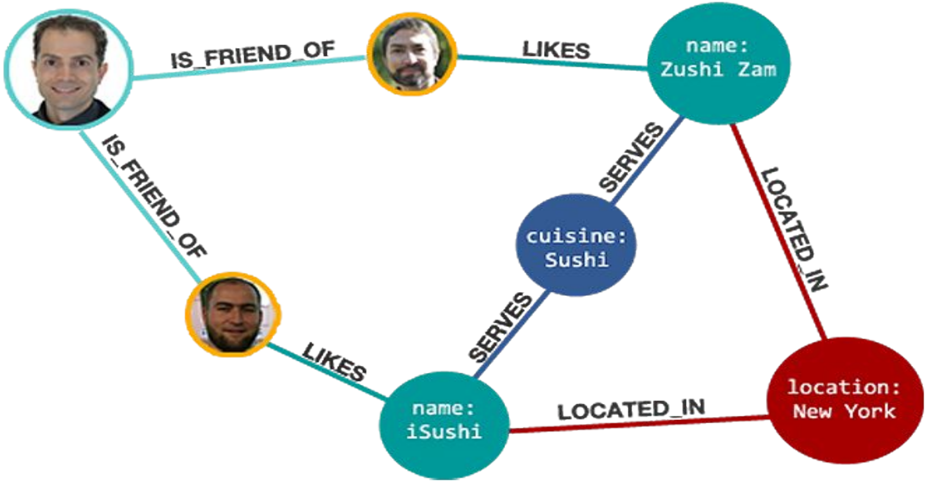
GQL is a proposed new international standard language for property graph querying. The idea of a standalone graph query language to complement SQL was raised by ISO SC32/ WG3 members in early 2017, and is echoed in the GQL manifesto of May 2018.

GQL supporters aim to develop a next-generation declarative graph query language that builds on the foundations of SQL and integrates proven ideas from the existing openCypher, PGQL, and G-CORE languages.

GQL will incorporate this prior work, as part of an expanded set of features including regular path queries, graph compositional queries (enabling views) and schema support.

A graph query example

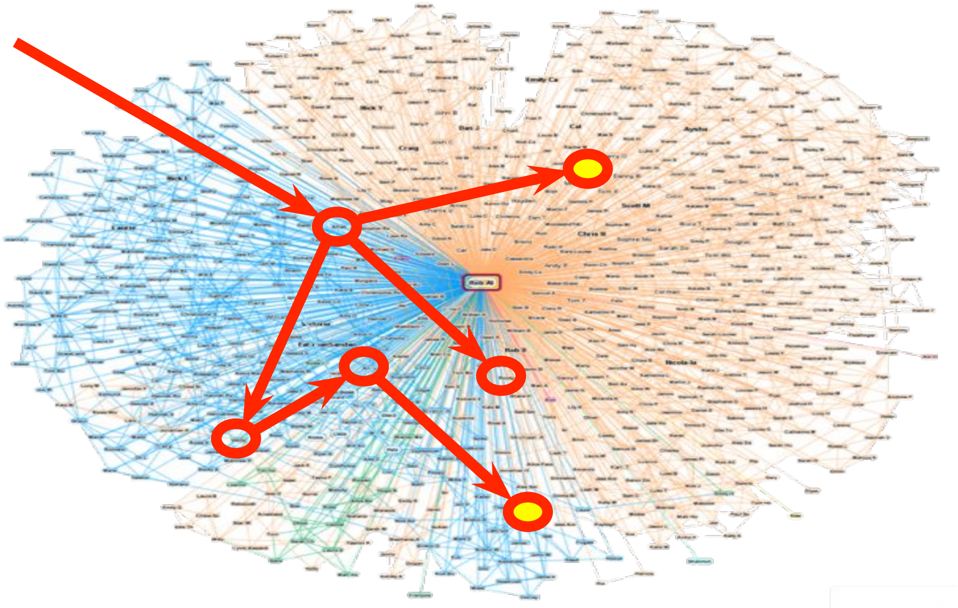
A social recommendation



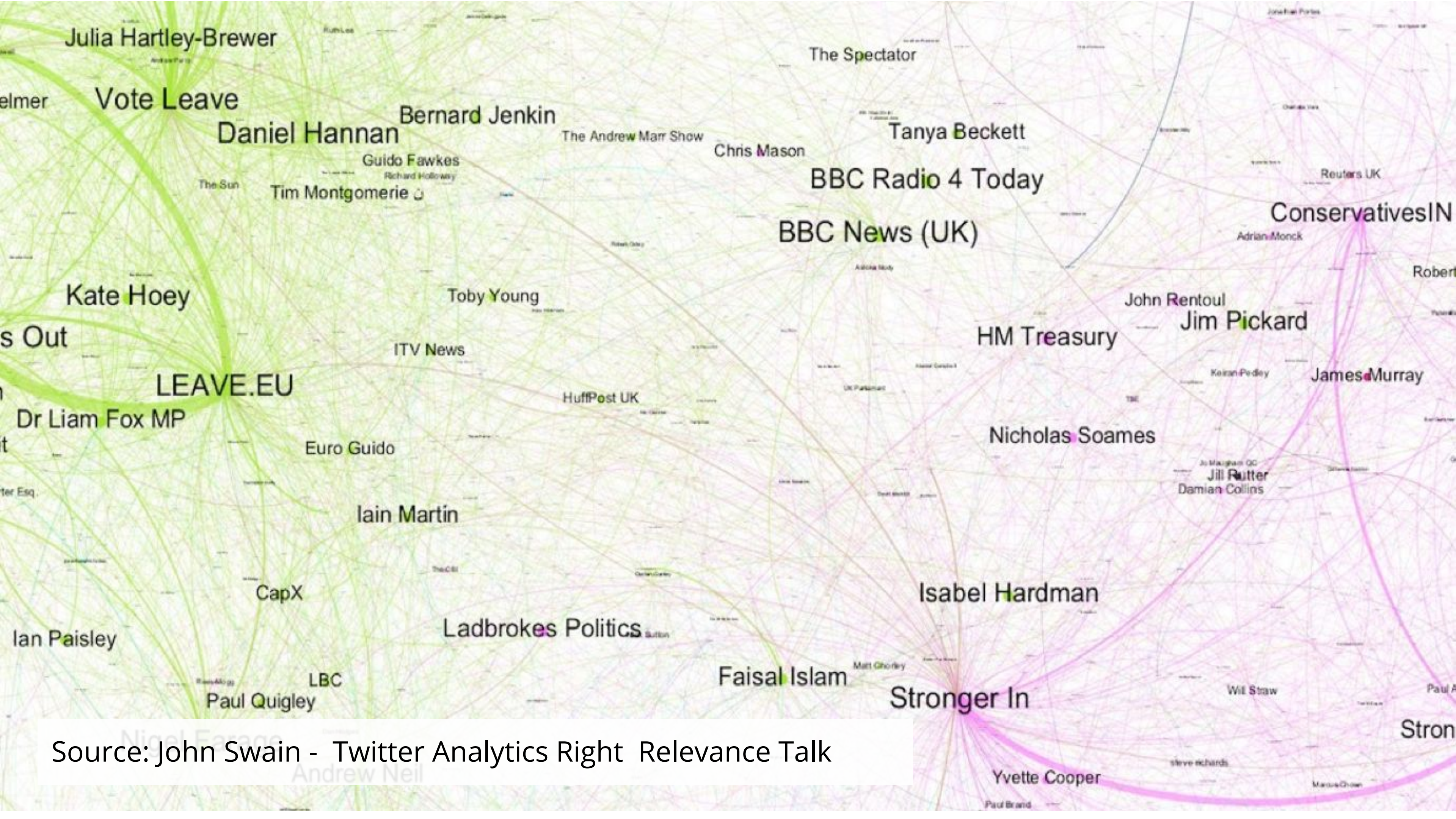
A social recommendation


```
MATCH (person:Person)-[:IS_FRIEND_OF]->(friend),
      (friend)-[:LIKES]->(restaurant),
      (restaurant)-[:LOCATED_IN]->(loc:Location),
      (restaurant)-[:SERVES]->(type:Cuisine)
WHERE person.name = 'Philip'
AND loc.location='New York'
AND type.cuisine='Sushi'
RETURN restaurant.name
```

A social recommendation

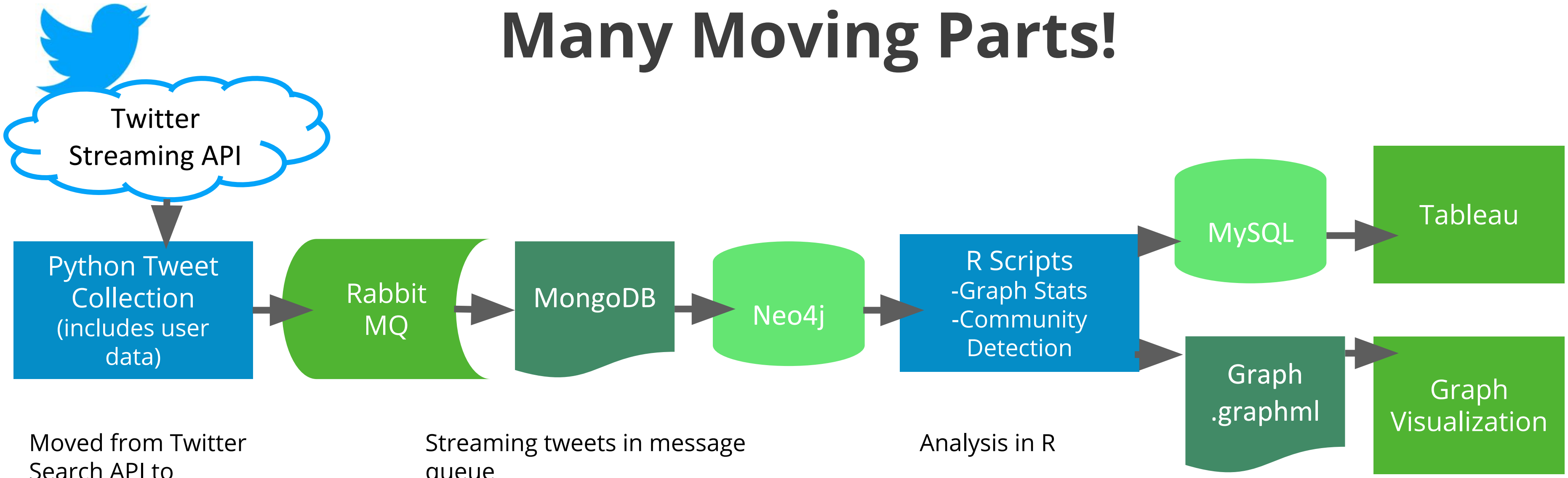


Graph Algorithms



Source: John Swain - Twitter Analytics Right Relevance Talk

Many Moving Parts!



Moved from Twitter Search API to Streaming API

Replaced Python Twitter libraries (Tweepy) with raw API calls

Streaming tweets in message queue

Full tweets and user data stored in MongoDB

Built graph for analysis in Neo4j from tweets persisted in MongoDB

Analysis in R

iGraph libraries for algorithms

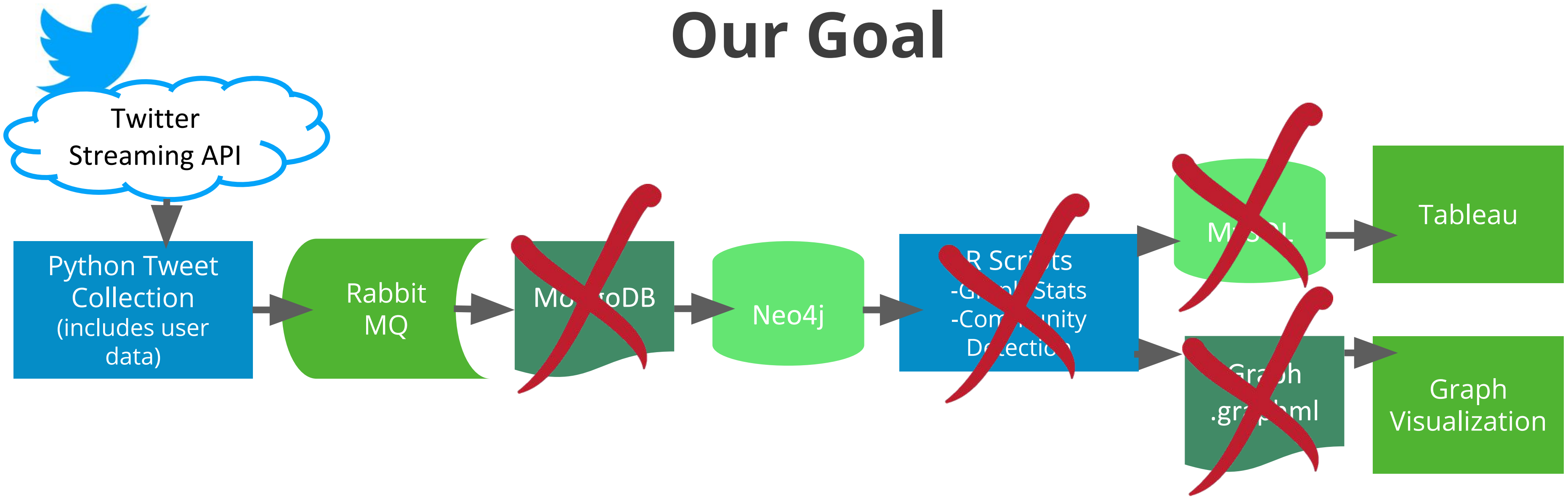
Some text analysis e.g. LDA topics

Results published in MySQL for Tableau

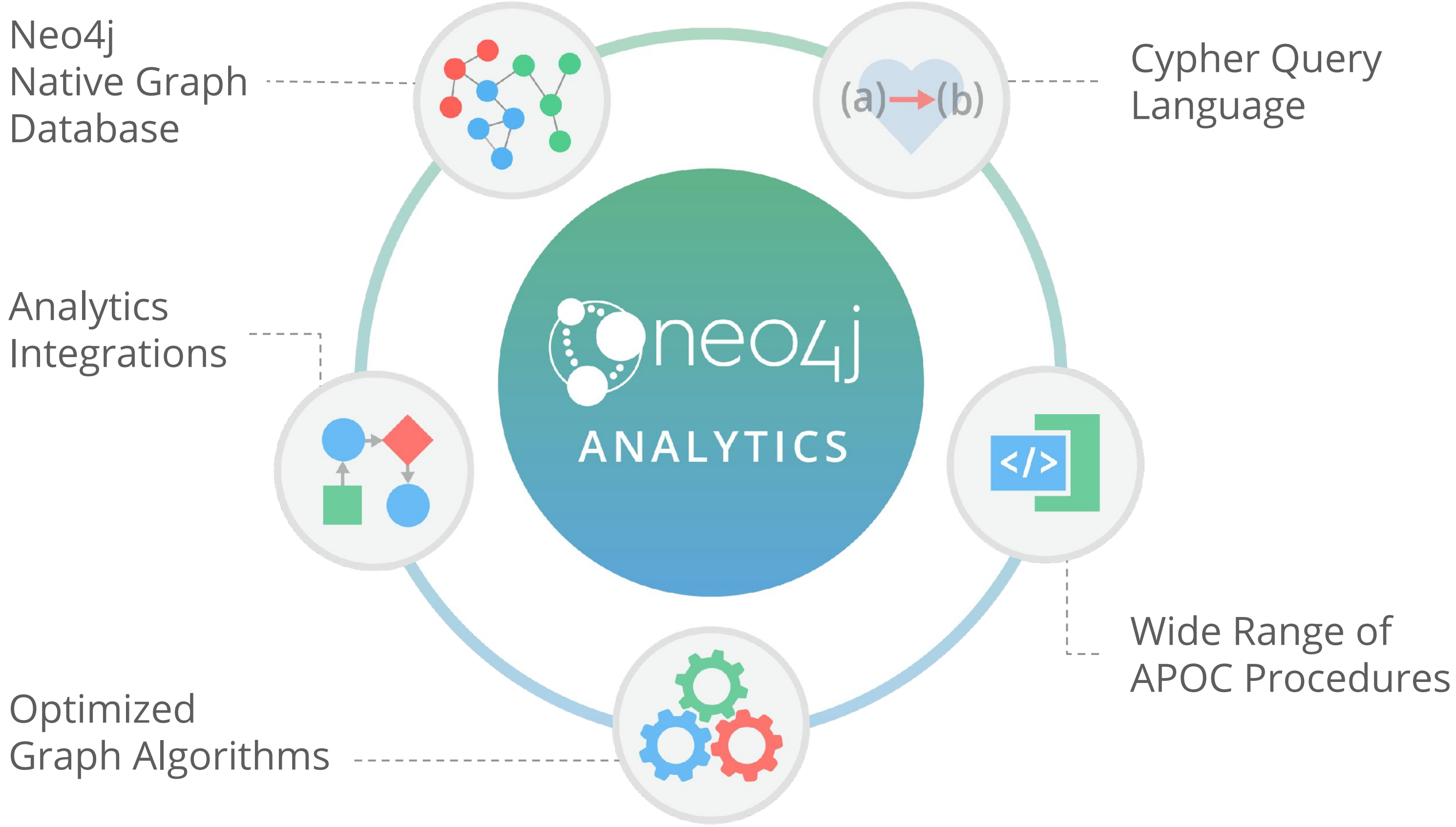
Graphml for import to Gephi with stats precalculated

Example Workflow Pipeline

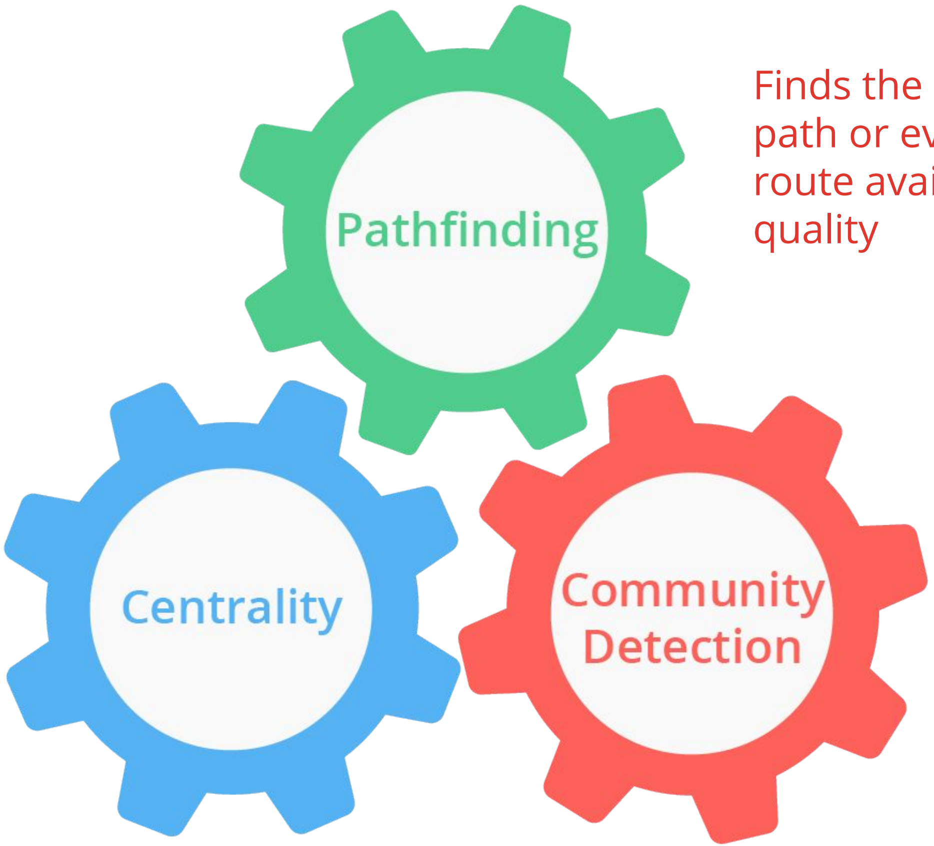
Our Goal



Example Workflow Pipeline



Neo4j
Native Graph
Database
(a)→(b)
Cypher Query
Language
neo4j
ANALYTICS
Analytics
Integrations
</>
Wide Range of
APOC Procedures
Optimized
Graph Algorithms



Finds the optimal path or evaluates route availability and quality

Determines the importance of distinct nodes in the network

Evaluates how a group is clustered or partitioned

Usage

1. Call as Cypher procedure
2. Pass in specification (Label, Prop, Query) and configuration
3. ~.stream variant returns (**a lot**) of results

```
CALL algo.<name>.stream('Label','TYPE',{conf})
YIELD nodeId, score
```

4. non-stream variant writes results to graph returns statistics

```
CALL algo.<name>('Label','TYPE',{conf})
```

Cypher Projection

Pass in Cypher statement for node- and relationship-lists.

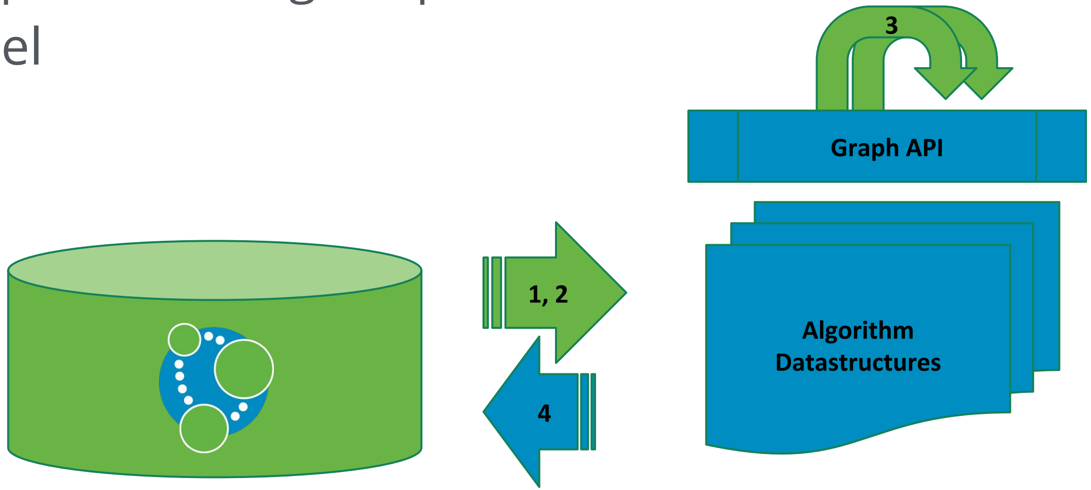
```
CALL algo.<name>(
  'MATCH ... RETURN id(n)',
  'MATCH (n)-->(m)
   RETURN id(n) as source,
          id(m) as target',   {graph:'cypher'})
```

Design Considerations

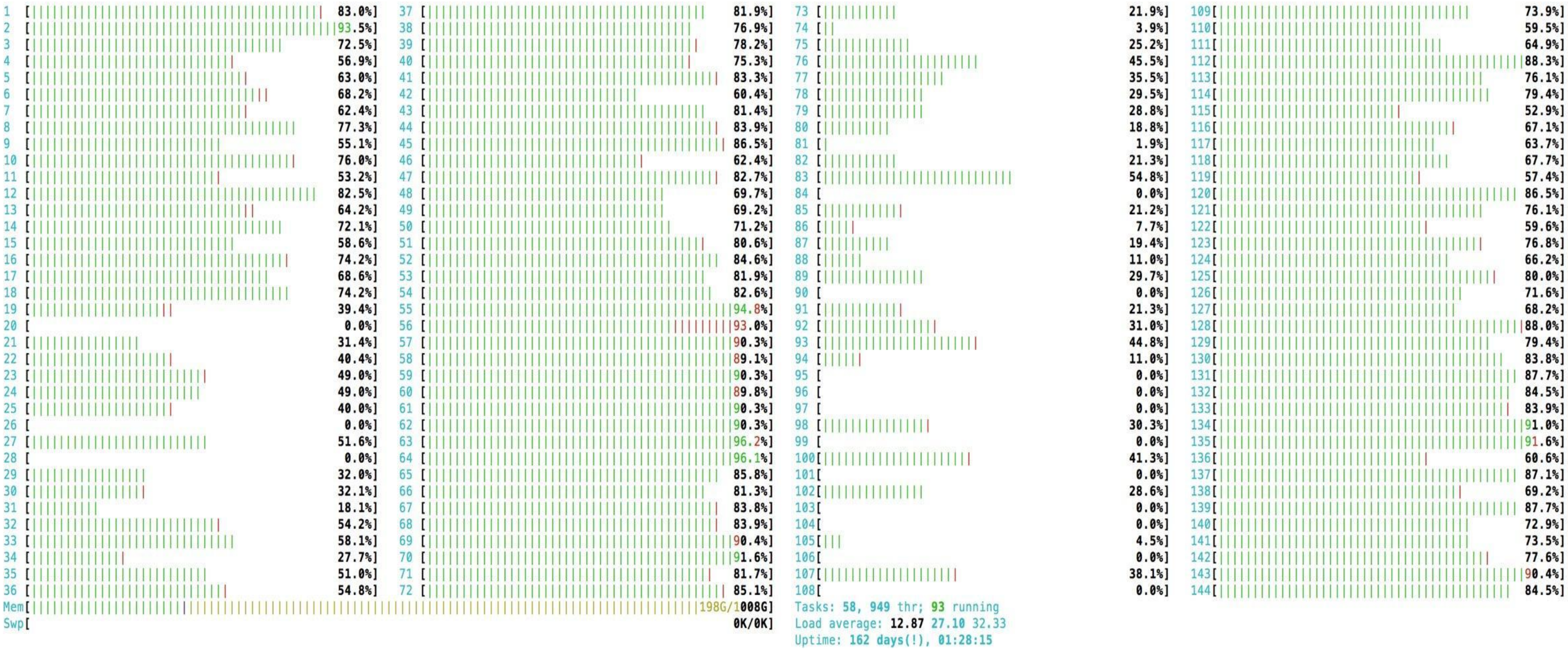
- Ease of Use – Call as Procedures
- Parallelize everything: load, compute, write
- Efficiency: Use direct access, efficient datastructures, provide high-level API
- Scale to billions of nodes and relationships
  Use up to hundreds of CPUs and Terabytes of RAM

Architecture

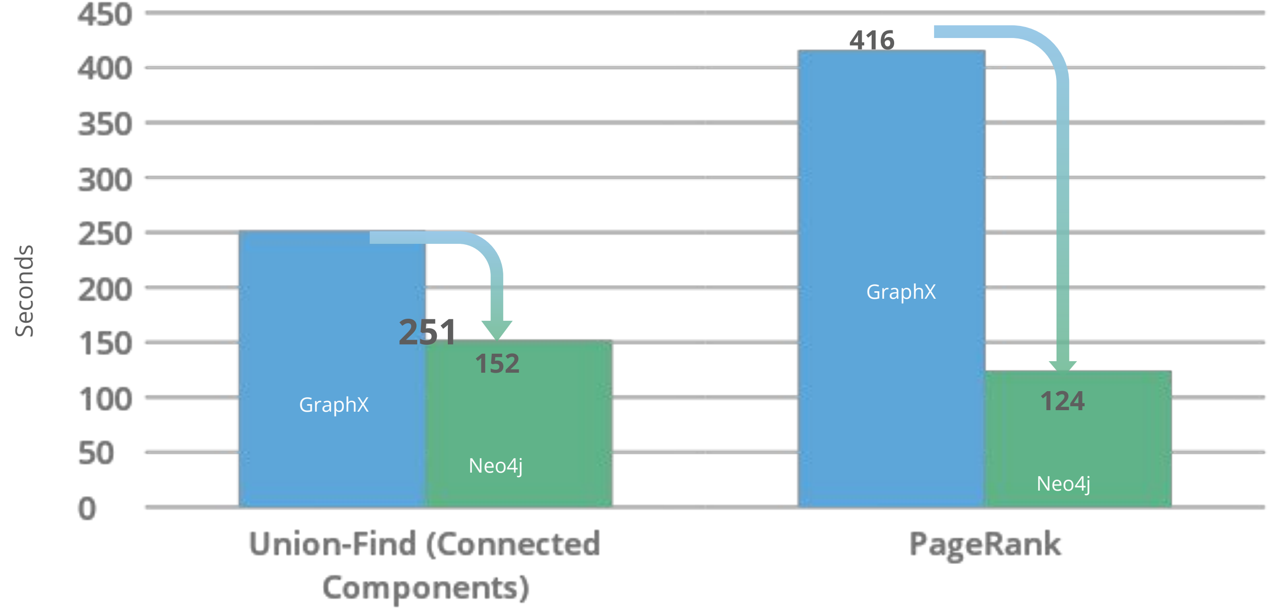
1. Load Data in parallel from Neo4j
2. Store in efficient data structures
3. Run Graph Algorithm in parallel using Graph API
4. Write data back in parallel



Scale: 144 CPU



Neo4j Graph Platform with Neo4j Algorithms vs. Apache Spark's GraphX



Neo4j provides same order of magnitude performace

Twitter 2010 Dataset
- 1.47 Billion Relationships
- 41.65 Million Nodes

Spark GraphX results publicly available
- Amazon EC2 cluster running 64-bit Linux
- 128 CPUs with 68 GB of memory, 2 hard disks

Neo4j Configuration
- Physical machine running 64-bit Linux
- 128 CPUs with 55 GB RAM, SSDs

Compute At Scale – Payment Graph

3,000,000,000 nodes and 18,000,000,000 relationships (600G)
PageRank (20 iterations) on 1 machine, 20 threads, 700G RAM

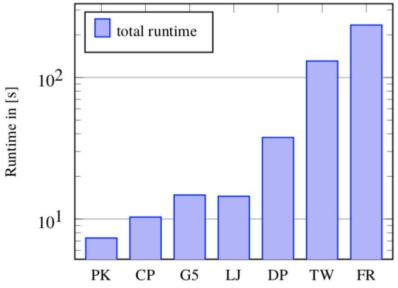
```
call algo.pageRank('Account','SENT',{graph:'big', iterations:20,write:false});
+--------------------------------------------------------+
| nodes      | iterations | loadMillis | computeMillis |
+--------------------------------------------------------+
| 3000000096 | 20         | 0          | 9845756       |
+--------------------------------------------------------+
1 row
9845794 ms -> 2h 44m
```

Evaluation

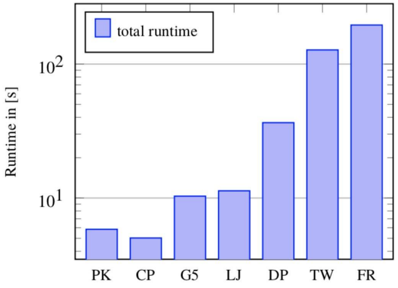
Graph		#Nodes [M]	#Relationships [M]	Avg. out degree	Disk size [GB]
Pokec	(PK)	1.63	30.62	18.75	0.99
cit-patents	(CP)	3.77	16.52	4.38	0.58
Graphs500-23	(G5)	4.61	129.33	28.05	4.17
soc-LifeJournal1	(LJ)	4.85	68.99	14.23	2.27
DBPedia	(DP)	11.47	116.60	10.16	3.87
Twitter-2010	(TW)	41.65	1468.37	35.25	47.60
Friendster	(FR)	65.61	1806.07	27.53	58.94

Tab. 2: Graph datasets used in measurements.

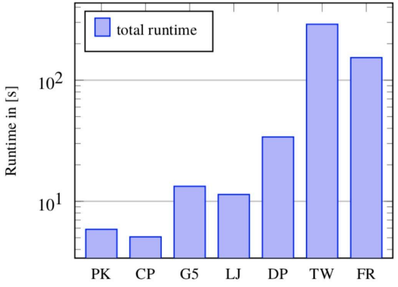
Evaluation



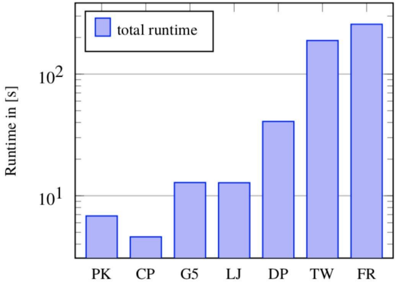
(a) Pagerank.



(b) Union Find.



(c) Label Propagation.



(d) Strongly-Connected Components.

Fig. 4: Total runtimes.

Twitter Troll Analysis

neo4j

TECH DEC 20 2017, 11:11 AM ET

Russian trolls went on attack during key election moments

by BEN POPKEN

https://www.nbcnews.com/tech/social-media/russian-trolls-went-attack-during-key-election-moments-n827176

NBC NEWS SECTIONS NIGHTLY NEWS MSNBC MEET THE PRESS DATELINE TODAY

TECH > SOCIAL MEDIA

GADGETS INTERNET SECURITY INNOVATION MOBILE

TECH DEC 20 2017, 11:11 AM ET

Russian trolls went on attack during key election moments

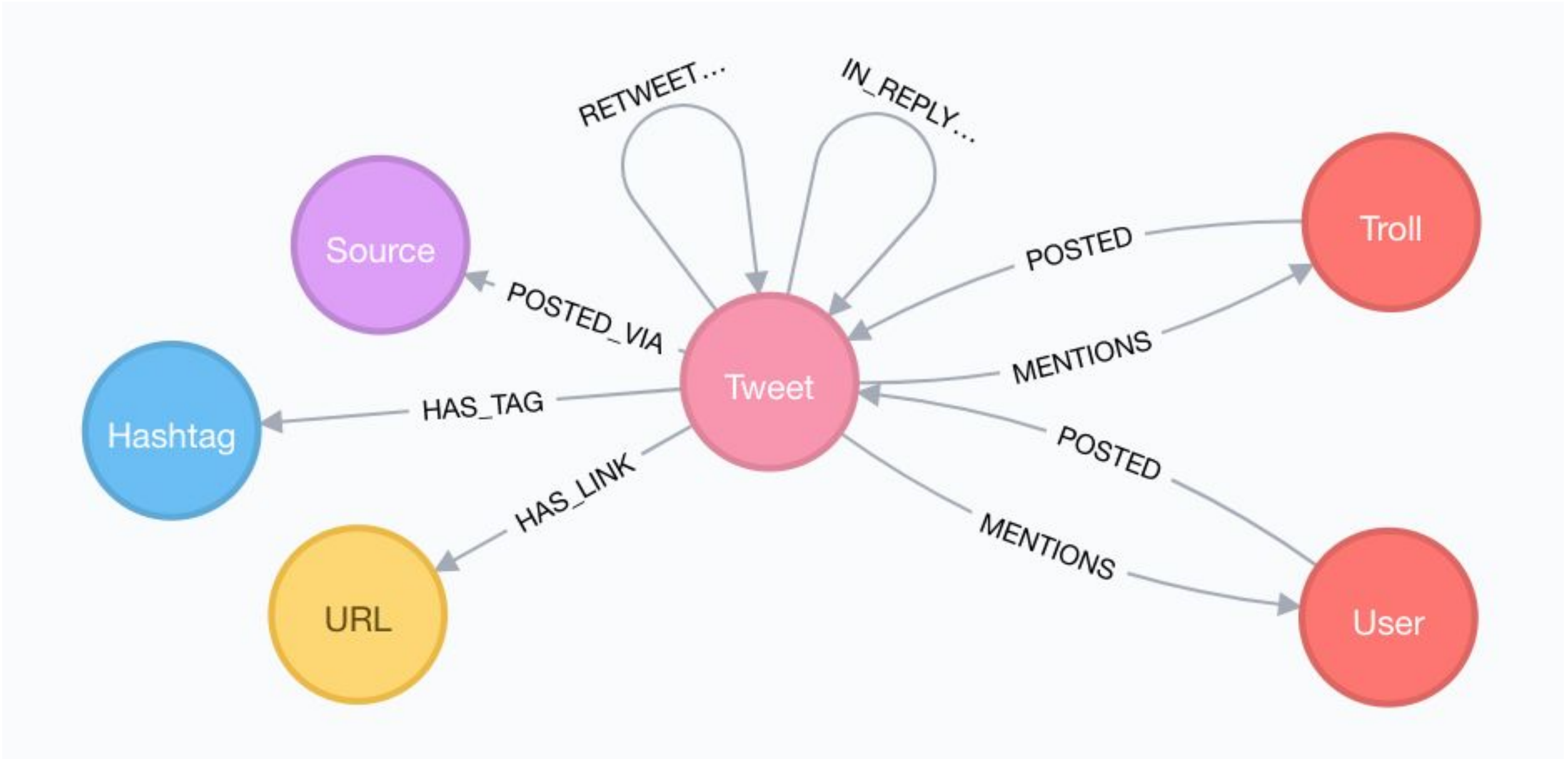
by BEN POPKEN

Ben Popken
@bpopken

Following

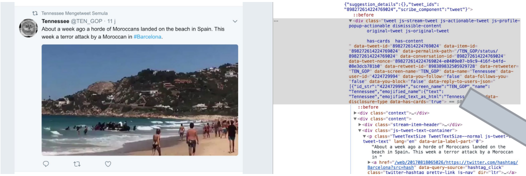
Huge props and thank you to @neo4j and their @mdavidallen and @lyonwj for helping compile and analyze the deleted twitter data, surfacing trends and uncovering new angles.

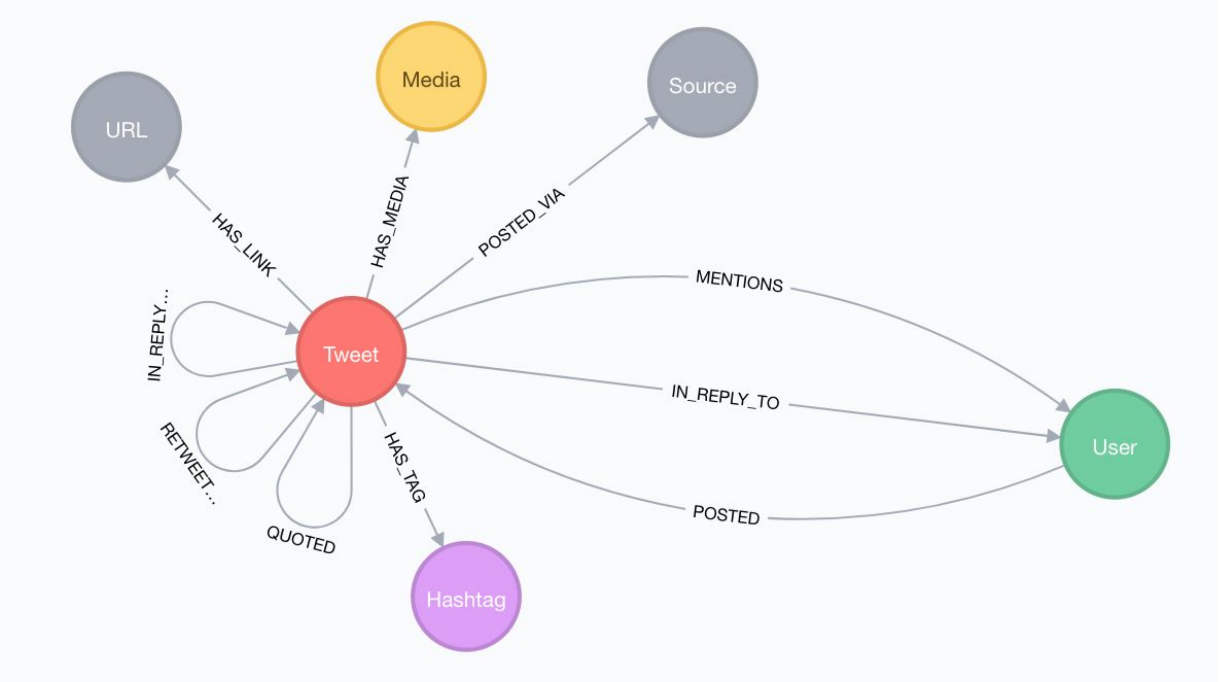
https://www.nbcnews.com/pages/author/ben-popken



http://www.lyonwj.com/2017/11/12/scraping-russian-twitter-trolls-python-neo4j/

345k **Tweets**, 41k **Users** (454 Russian **Trolls**)





Your typical American Citizen?

@LeroyLovesUSA

@ClevelandOnline

Your typical Local News Publication?

Your typical Local Political Party?

@TEN_GOP

Your typical **Russian Troll**

@LeroyLovesUSA

Your typical **Russian Troll**

@ClevelandOnline

Your typical **Russian Troll**

@TEN_GOP

IRA - Internet Research Agency

npr

SIGN IN SHOP DONATE

NEWS ARTS & LIFE MUSIC SHOWS & PODCASTS SEARCH

NATIONAL SECURITY

The Russia Investigations: Mueller Indicts The 'Internet Research Agency'

February 17, 2018 · 7:00 AM ET

PHILIP EWING

This week in the Russia investigations: A major new indictment from the special counsel's office that charges thirteen individuals and three companies and shakes up the political rhetoric as new facts are revealed in the sprawling imbroglio.

Justice Department special counsel Robert Mueller prefers to let his work do the talking for him. On Friday, he delivered a stemwinder.

Thirteen Russians and three Russian entities were indicted by a federal grand jury in connection with the attack on the 2016 election. The indictment lays out a number of detailed allegations against the Internet Research Agency located in St. Petersburg and against individuals who owned, controlled, funded or worked for the organization.

NATIONAL SECURITY

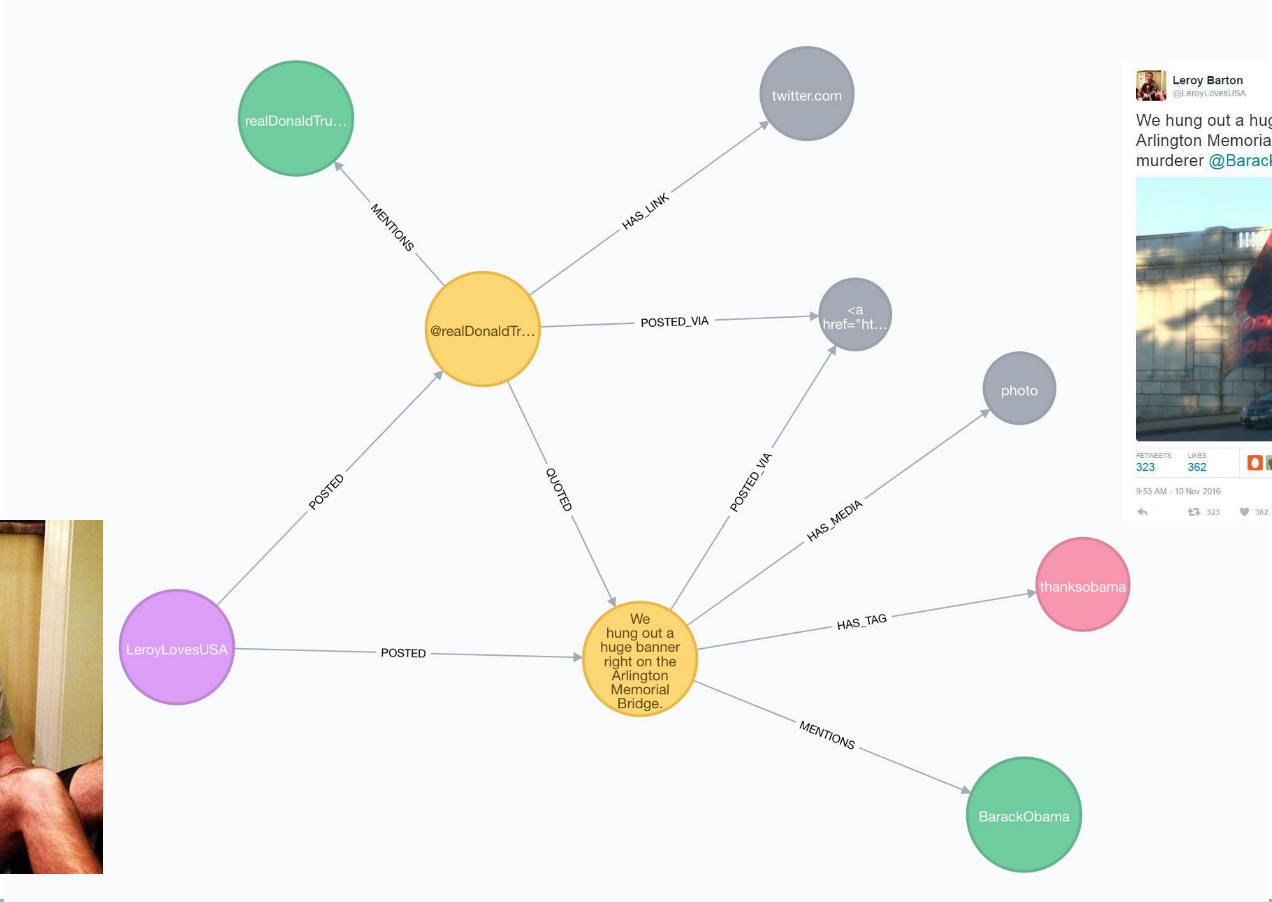
Grand Jury Indicts Russians Linked To Interference In 2016 Election

National Security

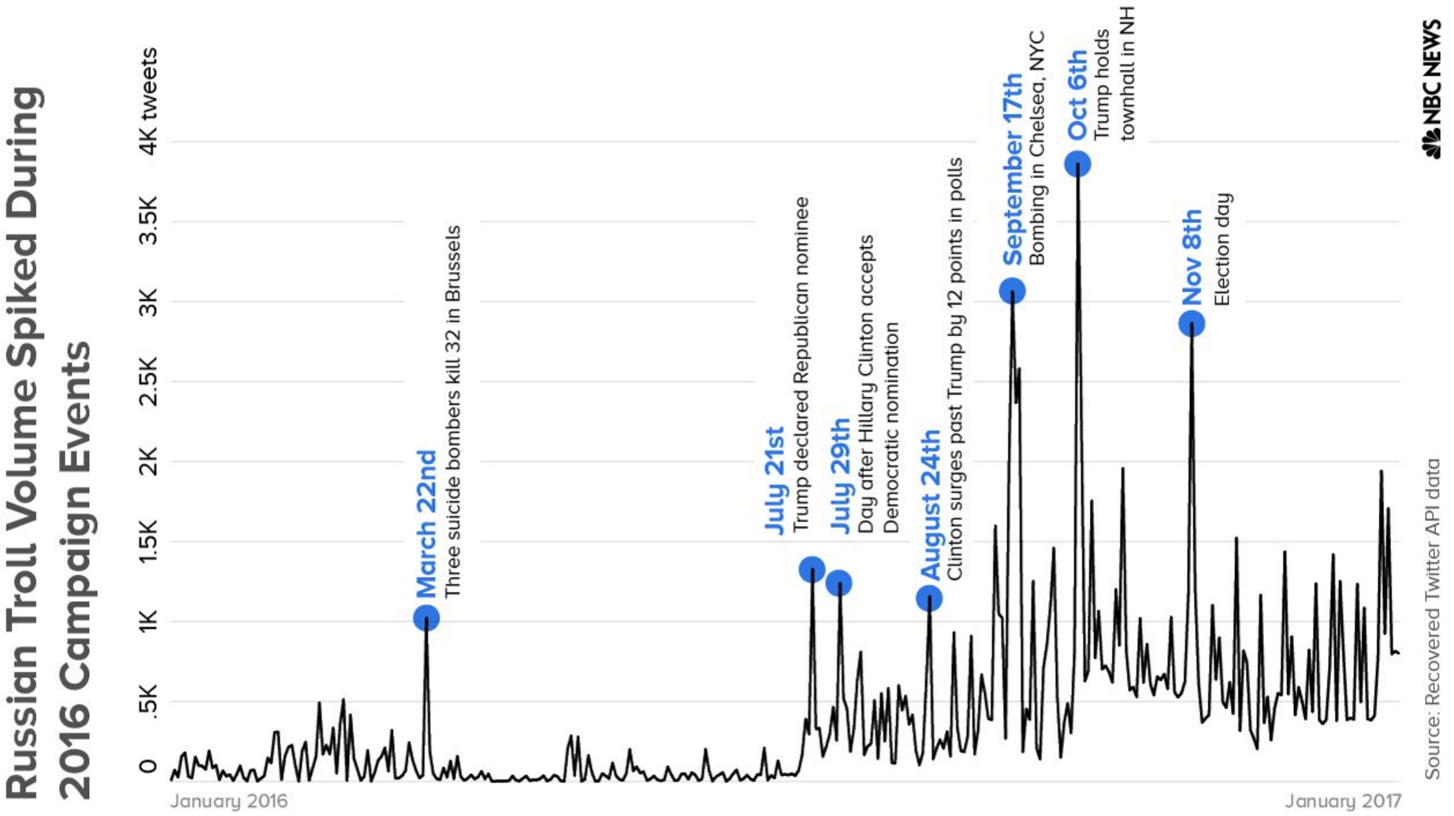
U.S. Cyber Command operation disrupted Internet access of Russian troll factory on day of 2018 midterms

The building that housed the Internet Research Agency in St. Petersburg, shown in 2018. (Dmitri Lovetsky/AP)

```
MATCH
(u:User {screen_name: "LeroyLovesUSA"})-[:POSTED]->(t:Tweet)-[:HAS_TAG]->(ht:Hashtag {key: "thanksobama"})
RETURN *
```



```
MATCH (t:Tweet)<-[:POSTED]-(u:Troll)
RETURN t.dayOfYear AS day, COUNT(*) AS num ORDER BY day
```



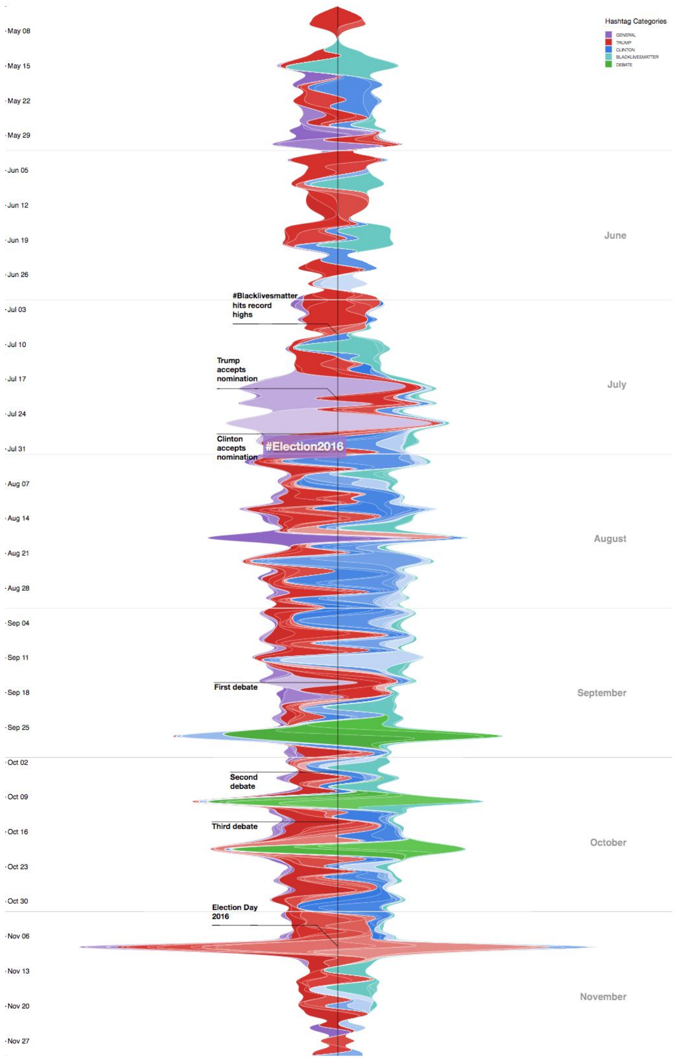
https://www.nbcnews.com/tech/social-media/russian-trolls-went-attack-during-key-election-moments-n827176

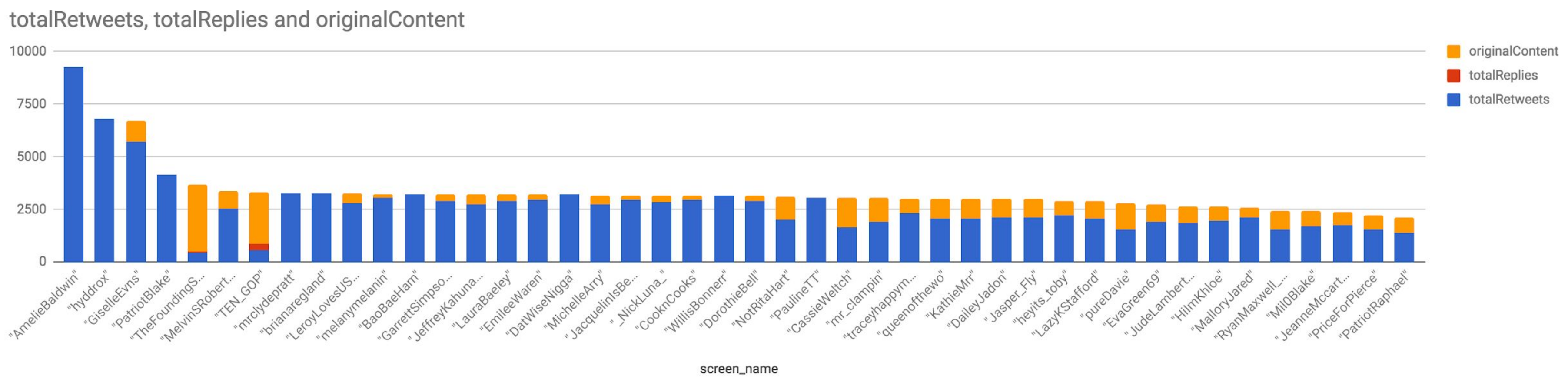
Hashtags

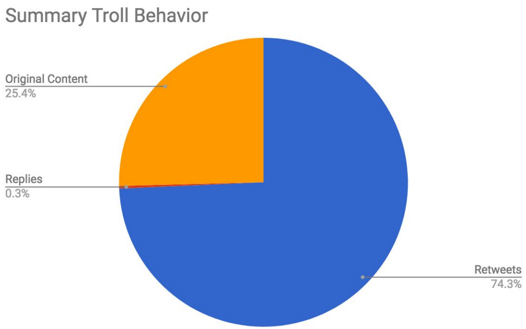
- Use of hashtags to gain visibility and insert into conversation

- **@WorldOfHashtags**
  - #RejectedDebateTopics

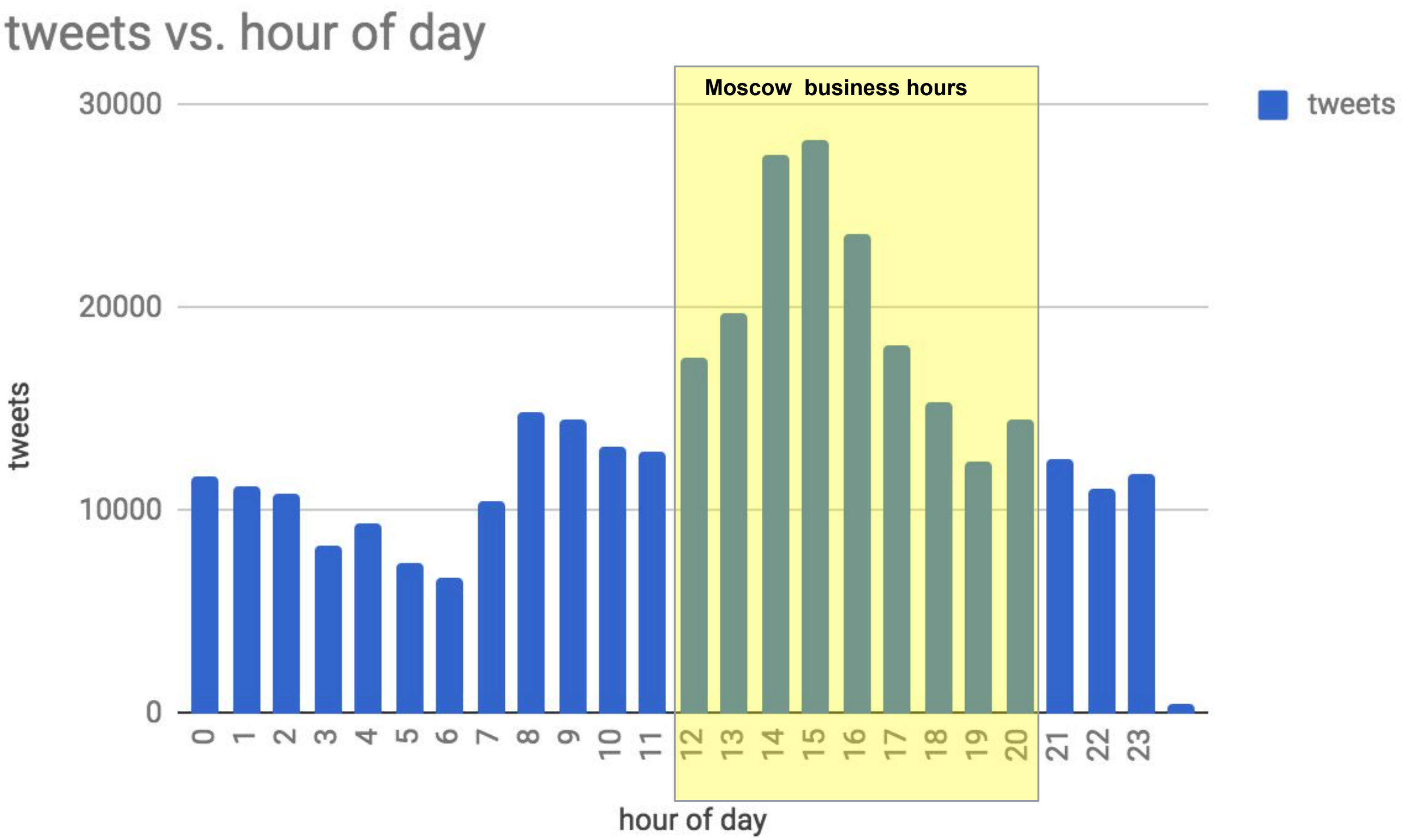
https://www.nbcnews.com/tech/social-media/russian-trolls-went-attack-during-key-election-moments-n827176







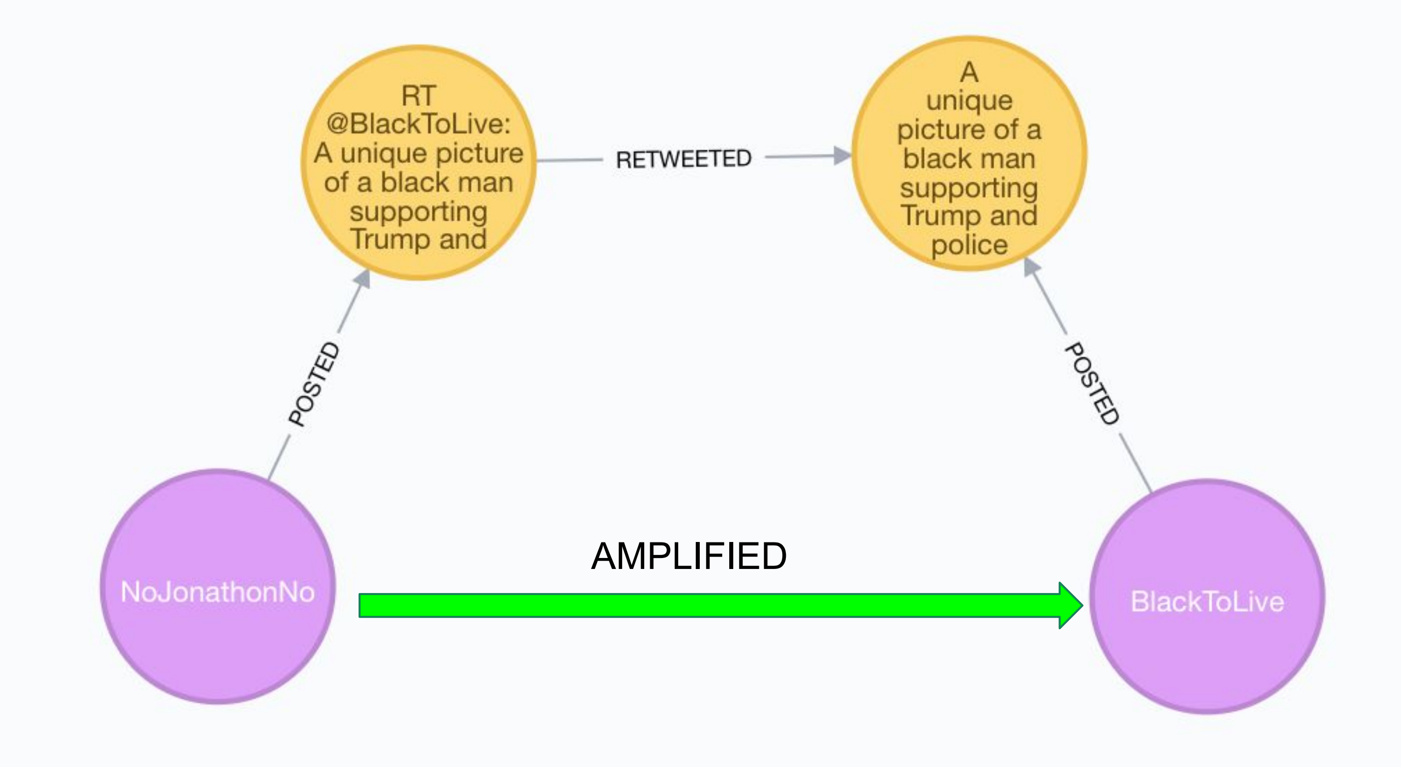

```
MATCH (tr:Troll)-[:POSTED]->(tw:Tweet) WITH tr, tw
OPTIONAL MATCH (tw)-[:RETWEETED]-(rt:Tweet)
OPTIONAL MATCH (tw)-[:IN_REPLY_TO]-(irp:Tweet)
RETURN distinct tr.screen_name as screen_name, count(tw) as totalTweets,
  count(rt) as totalRetweets, count(irp) as totalReplies,
  (count(tw) - (count(rt) + count(irp))) as originalContent
ORDER BY totalTweets DESC;
```



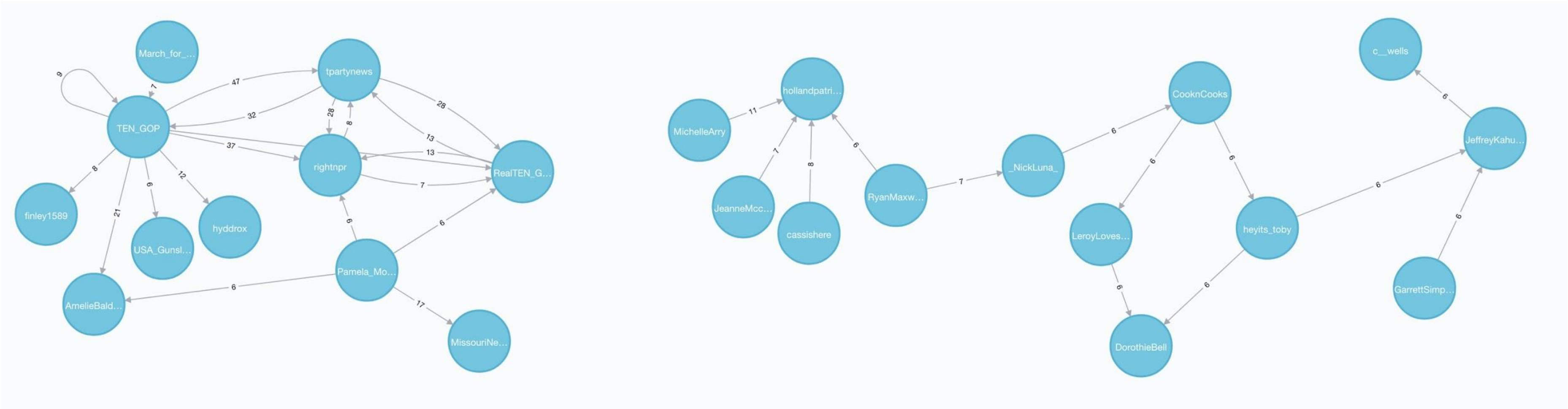
tweets vs. hour of day
Moscow business hours
tweets
30000
20000
10000
0
tweets
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
hour of day

Inferred Relationships

```
1 MATCH (r1:Troll)-[:POSTED]->(t1:Tweet)<-[:RETWEETED]-(t2:Tweet)<-[:POSTED]-(r2:Troll)
```

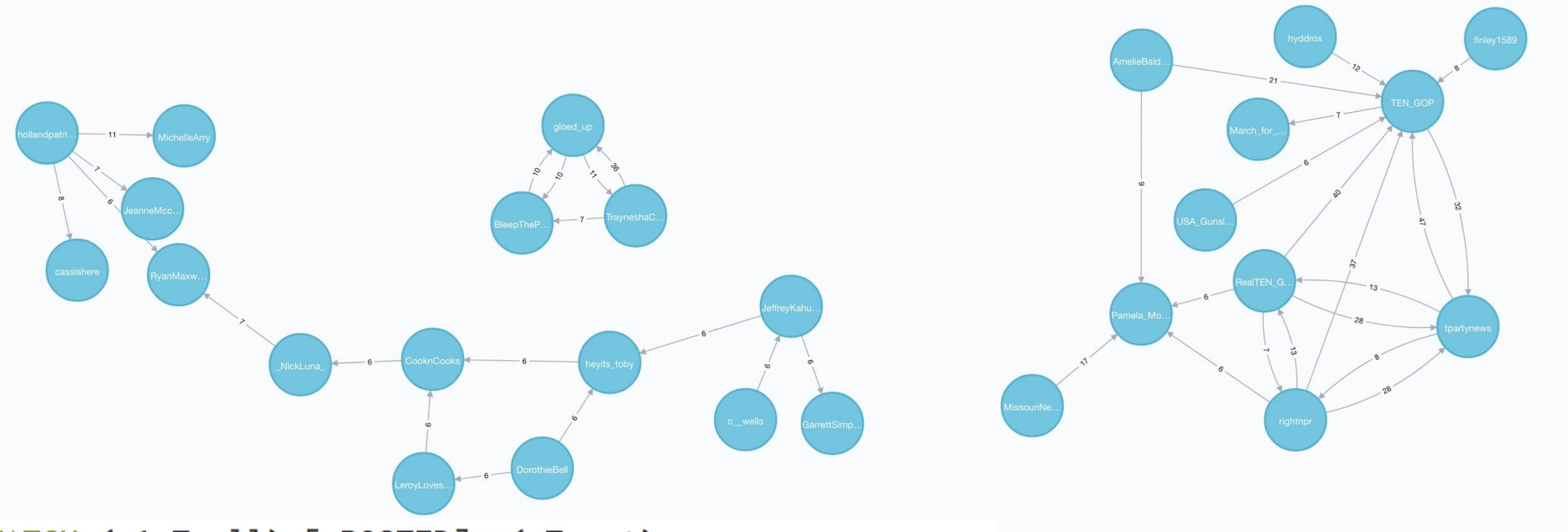


Inferred Relationships




```
MATCH (r1:Troll)-[:POSTED]->(:Tweet)
       <-[:RETWEETED]-(:Tweet)<-[:POSTED]-(r2:Troll)
WITH r1,r2, count(*) as freq where freq > 5
RETURN r1,r2, apoc.create.vRelationship(r1,'AMPLIFIED',{freq:freq},r2) as rel
```

Inferred Relationships




```
MATCH (r1:Troll)-[:POSTED]->(:Tweet)
        <-[:RETWEETED]-(:Tweet)<-[:POSTED]-(r2:Troll)
WHERE r1 <> r2
WITH r1, r2, count(*) as freq
CREATE (r2)-[:AMPLIFIED {weight:freq}]->(r1)
```

Weighted In-Degree Centrality

```
match (t:Troll)<-[r:AMPLIFIED]-() with t, sum(r.weight) as total return t.screen_name, total order by total desc limit 5
```

"t.screen_name"	"total"
"TEN_GOP"	239
"NotRitaHart"	107
"GiselleEvns"	104
"tpartynews"	98
"DaileyJadon"	84

PageRank on Inferred AMPLIFIED Graph

```
CALL algo.pageRank(
"MATCH (r:Troll) WHERE exists( (r)-[:POSTED]->() )
 RETURN id(r) as id",
"MATCH (r1:Troll)-[:POSTED]->(:Tweet)
       <-[:RETWEETED]-(:Tweet)<-[:POSTED]-(r2:Troll)
 RETURN id(r2) as source, id(r1) as target",
{graph:'cypher'})
```

PageRank on Inferred AMPLIFIED Graph

```
match (t:Troll) where exists (t.pagerank) return t.screen_name, t.pagerank order by t.pagerank desc limit 5
```

"t.screen_name"	"t.pagerank"
"TEN_GOP"	10.3859635
"TheFoundingSon"	8.281644000000002
"GiselleEvns"	6.4624315
"tpartynews"	6.3289815
"ChrixMorgan"	4.231436500000001

Graph Visualization

Based on metrics computed by graph algorithms

neo4j

Graph Visualization

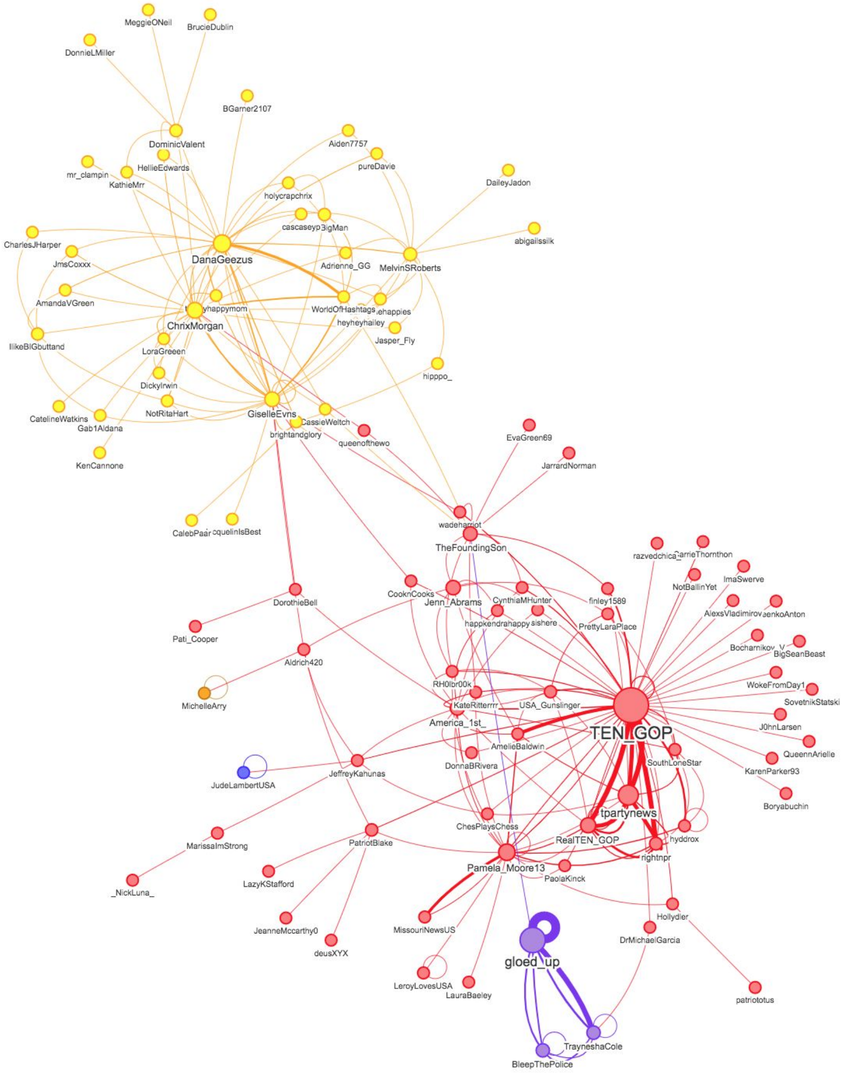
Centrality & community detection

AMPLIFIED relationships

Node size ⟶ PageRank

Color ⟶ community detection

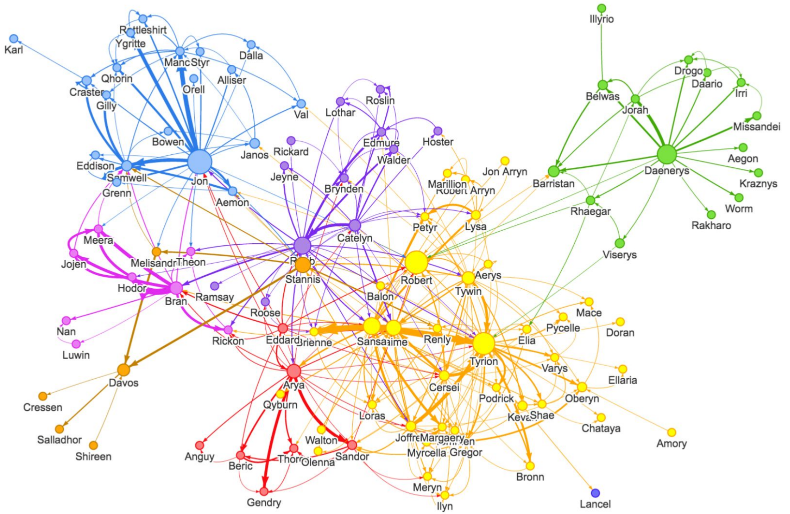
Rel Thickness ⟶ weight



Graph Visualization

neovis.js

Graph visualizations powered by vis.js with data from Neo4j.



```
var config = {
                container_id: "viz",
                server_url: "bolt://localhost:7687",
                server_user: "neo4j",
                server_password: "sorts-swims-burglaries",
                labels: {
                    //"Character": "name",
                    "Character": {
                        "caption": "name",
                        "size": "pagerank",
                        "community": "community"
                        //"sizeCypher": "MATCH (n) WHERE id(n) = {id} MATCH (n)-[r]-() RETURN sum(r.weight)
                    }
                },
                relationships: {
                    "INTERACTS": {
                        "thickness": "weight",
                        "caption": false
                    }
                },
                initial_cypher: "MATCH (n)-[r:INTERACTS]->(m) RETURN n,r,m"
            };

            viz = new NeoVis.default(config);
            viz.render();
```

https://github.com/neo4j-contrib/neovis.js

2 days later - IRA taken to court and indicted

Feb 14:

NEWS

Twitter Deleted 200,000 Russian Troll Tweets. Read Them Here.

GET THE DATA:

- *Regular reader?* Download streamlined spreadsheet (29 mb) with just usernames, tweet and timestamps. We recommend you right click on links and select "save link as" or similar, otherwise it may take a long time to load in your browser.
- View full data for ten influential accounts in Google Sheets
- *Researcher?* Download tweets.csv (50 mb) and users.csv with full underlying data
- Explore a graph database in Neo4j

Feb 16:

IN THE UNITED STATES DISTRICT COURT
FOR THE DISTRICT OF COLUMBIA

UNITED STATES OF AMERICA	*	
	*	CRIMINAL NO.
v.	*	
	*	(18 U.S.C. §§ 2, 371, 1349, 1028A)
INTERNET RESEARCH AGENCY LLC	*	
A/K/A MEDIASINTEZ LLC A/K/A	*	
GLAVSET LLC A/K/A MIXINFO	*	
LLC A/K/A AZIMUT LLC A/K/A	*	
NOVINFO LLC,	*	
CONCORD MANAGEMENT AND	*	
CONSULTING LLC,	*	
CONCORD CATERING,	*	
YEVGENIY VIKTOROVICH	*	
PRIGOZHIN,	*	
MIKHAIL IVANOVICH BYSTROV,	*	
MIKHAIL LEONIDOVICH BURCHIK	*	
A/K/A MIKHAIL ABRAMOV,	*	
ALEKSANDRA YURYEVNA	*	
KRYLOVA,	*	
ANNA VLADISLAVOVNA	*	
BOGACHEVA,	*	
SERGEY PAVLOVICH POLOZOV,	*	
MARIA ANATOLYEVNA BOVDA	*	
A/K/A MARIA ANATOLYEVNA	*	
BELYAEVA,	*	
ROBERT SERGEYEVICH BOVDA,	*	
DZHEYKHUN NASIMI OGLY	*	
ASLANOV A/K/A JAYHOON	*	
ASLANOV A/K/A JAY ASLANOV,	*	
VADIM VLADIMIROVICH	*	
PODKOPAEV,	*	
GLEB IGOREVICH VASILCHENKO,	*	
IRINA VIKTOROVNA KAVERZINA,	*	
and	*	
VLADIMIR VENKOV.	*******	

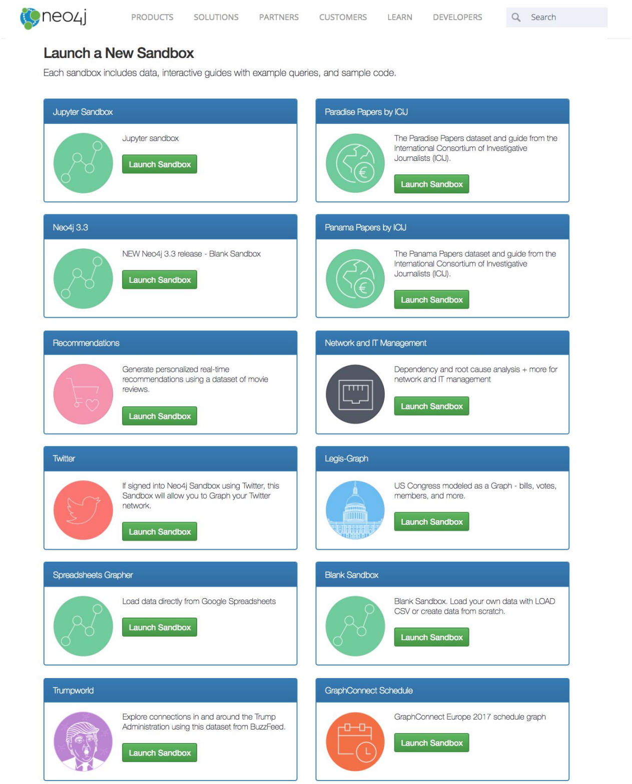
https://www.nbcnews.com/tech/social-media/now-available-more-200-000-deleted-russian-troll-tweets-n844731

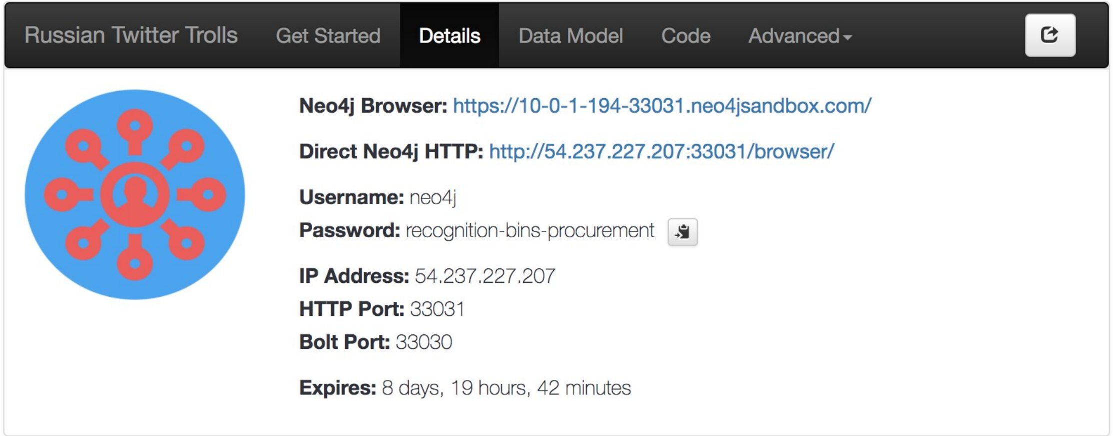
Surprising Takeaways

- Amplifying w/ retweets
- Used social media automation tools
  - Not necessarily live responses
- Meddling in elections is just another 9-5 job
- **Data availability**

- See lyonwj.com for code, etc.
- https://www.nbcnews.com/tech/social-media/russian-trolls-went-attack-during-key-election-moments-n827176

neo4jsandbox.com





https://hackernoon.com/six-ways-to-explore-the-russian-twitter-trolls-database-in-neo4j-6e52394c38f1

Thank You

Questions?