



# Graph Analysis of Russian Twitter Trolls

William Lyon  
@lyonwj  
lyonwj.com

Stanford University  
Feb 2018

# Agenda

**Overview of Neo4j and graph databases in  
the context of Russian Twitter Troll data**

# William Lyon

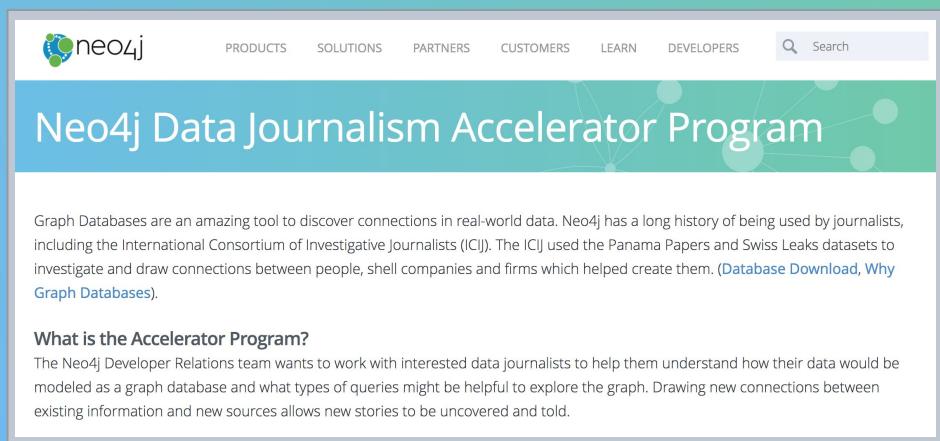
Software Developer, Neo4j

*(Developer Relations, integrations, GraphQL)*

[will@neo4j.com](mailto:will@neo4j.com)

@lyonwj

lyonwj.com



The screenshot shows a white header with the Neo4j logo and a navigation bar with links for PRODUCTS, SOLUTIONS, PARTNERS, CUSTOMERS, LEARN, DEVELOPERS, and a search bar. Below the header is a blue banner with the text "Neo4j Data Journalism Accelerator Program". The main content area has a light gray background and contains the following text:

Graph Databases are an amazing tool to discover connections in real-world data. Neo4j has a long history of being used by journalists, including the International Consortium of Investigative Journalists (ICIJ). The ICIJ used the Panama Papers and Swiss Leaks datasets to investigate and draw connections between people, shell companies and firms which helped create them. ([Database Download](#), [Why Graph Databases](#)).

**What is the Accelerator Program?**

The Neo4j Developer Relations team wants to work with interested data journalists to help them understand how their data would be modeled as a graph database and what types of queries might be helpful to explore the graph. Drawing new connections between existing information and new sources allows new stories to be uncovered and told.





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

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

# What's a Graph Database???



# What's a Graph Database???



**William Lyon**

@lyonwj

#Neo4jIn140Characters

**William Lyon** @lyonwj

Neo4j: Open source software that stores and queries data as nodes and relationships using the Cypher query language w/ index free adjacency.

3:45 PM - 23 Feb 2017



**William Lyon**  
@lyonwj

Neo4j: Open source software that stores and queries data as nodes and relationships using the Cypher query language w/ index free adjacency.

3:44 PM - 23 Feb 2017 from [Dunlap, Seattle](#)



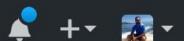
Reply to @lyonwj



This repository

Search

Pull requests Issues Gist



[neo4j / neo4j](#)

Watch ▾ 326

Unstar 3,502

Fork 1,079

Code

Issues 379

Pull requests 33

Projects 0

Wiki

Pulse

Graphs

Graphs for Everyone <http://neo4j.com>

45,336 commits

18 branches

173 releases

141 contributors



**William Lyon**  
@lyonwj

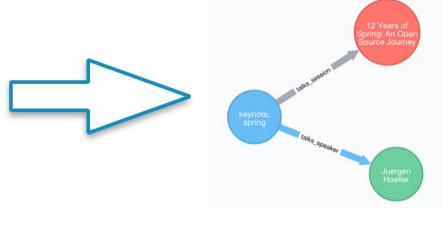
Neo4j: Open source software that stores and queries data as nodes and relationships using the Cypher query language w/ index free adjacency.

3:44 PM - 23 Feb 2017 from Dunlap, Seattle



Reply to @lyonwj

```
1  {
2     "title": "12 Years of Spring: An Open Source Journey",
3     "abstract": "Spring emerged as a core open source project in early 2003 and evolved to a broad por",
4     "topics": ["keynote", "spring"],
5     "room": "Auditorium",
6     "timeslot": "Wed 29th, 09:30-10:30",
7     "speaker": {
8         "name": "Juergen Hoeller",
9         "bio": "Juergen Hoeller is co-founder of the Spring Framework open source project and has been s",
10         "twitter": "https://twitter.com/springjuergen",
11         "picture": "http://www.spring.io.net/wp-content/uploads/2014/11/juergen_hoeller-220x220.jpeg"
12     }
13 }
```



- Data durability
- ACID transactions



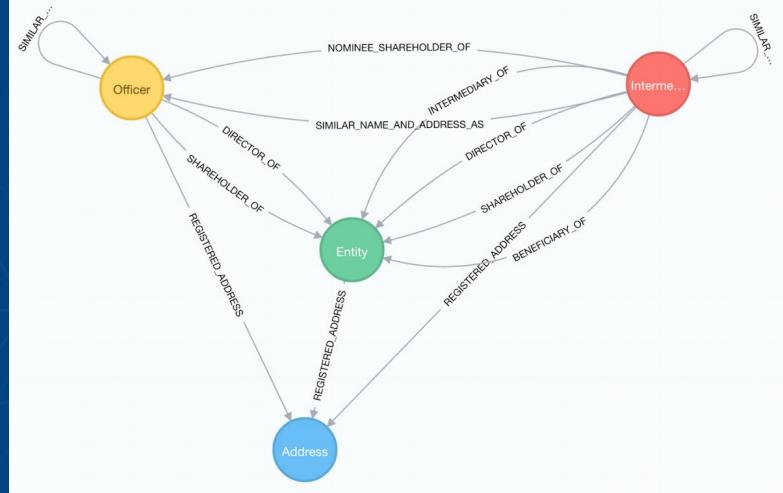
**William Lyon**  
@lyonwj

Neo4j: Open source software that stores and queries data **as nodes and relationships** using the Cypher query language w/ index free adjacency.

3:44 PM - 23 Feb 2017 from Dunlap, Seattle



Reply to @lyonwj





**William Lyon**

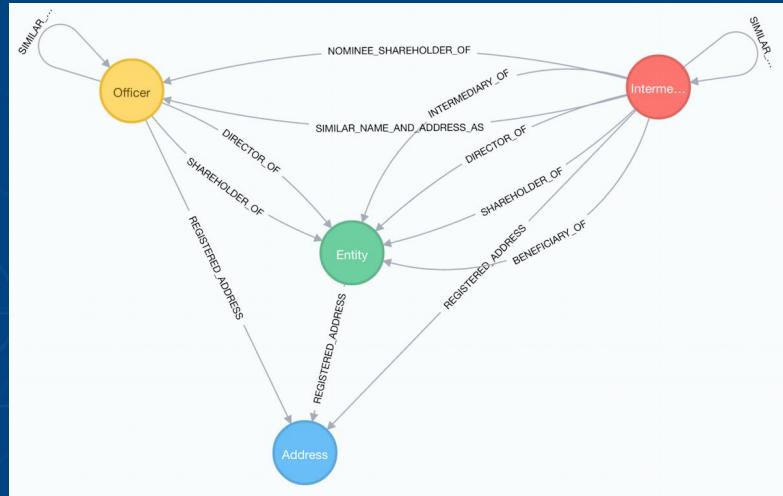
@lyonwj

Neo4j: Open source software that stores and queries data **as nodes and relationships** using the Cypher query language w/ index free adjacency.

3:44 PM - 23 Feb 2017 from Dunlap, Seattle



Reply to @lyonwj



<https://offshoreleaks.icij.org/pages/database>



```
1 MATCH (a:Address)<-[r:REGISTERED_ADDRESS]-(o:Officer)-->(e:Entity)
2 WHERE a.address CONTAINS "Palo Alto"
3 RETURN *
```



**William Lyon**  
@lyonwj

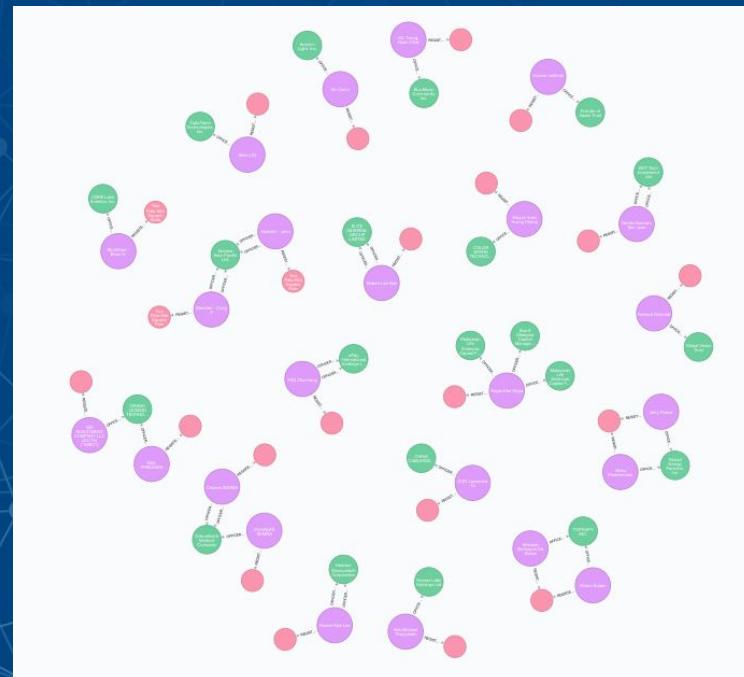
Neo4j: Open source software that stores and queries data as nodes and relationships using the Cypher query language w/ index free adjacency.

3:44 PM - 23 Feb 2017 from Dunlap, Seattle



Reply to @lyonwj

```
1 MATCH (a:Address)-[r:REGISTERED_ADDRESS]-(o:Officer)-->(e:Entity)  
2 WHERE a.address CONTAINS "Palo Alto"  
3 RETURN *
```



```
1 MATCH (a:Address)-[r:REGISTERED_ADDRESS]-(o:Officer)-->(e:Entity)
2 WHERE a.address CONTAINS "Palo Alto"
3 RETURN e.jurisdiction_description AS description, COUNT(*) AS num
4 ORDER BY num DESC
```



**William Lyon**  
@lyonwj

Neo4j: Open source software that stores and queries data as nodes and relationships using the Cypher query language w/ index free adjacency.

3:44 PM - 23 Feb 2017 from Dunlap, Seattle



Reply to @lyonwj

```
1 MATCH (a:Address)-[r:REGISTERED_ADDRESS]-(o:Officer)-->(e:Entity)
2 WHERE a.address CONTAINS "Palo Alto"
3 RETURN e.jurisdiction_description AS description, COUNT(*) AS num
4 ORDER BY num DESC
```

"description"	"num"
"British Virgin Islands"	18
"Bermuda"	4
"Undetermined"	3
"Labuan"	3
"Cook Islands"	2
"Cayman"	1
"Cayman Islands"	1
"Samoa"	1



**William Lyon**  
@lyonwj

Neo4j: Open source software that stores and queries data as nodes and relationships using the Cypher query language w/ index free adjacency.

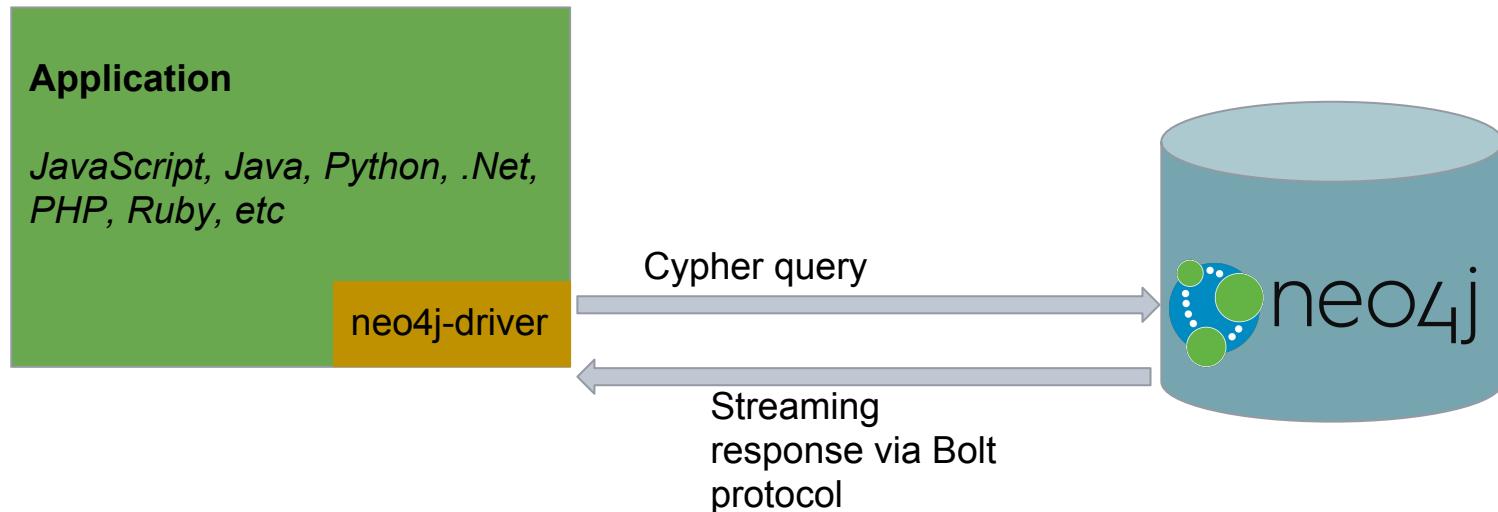
3:44 PM - 23 Feb 2017 from Dunlap, Seattle



Reply to @lyonwj

# Using Cypher With Client Drivers

Client driver libraries use Cypher over Bolt protocol



# Bolt Protocol

## Bolt vs HTTP

### HTTP

**Request [317 bytes]**  
POST http://localhost:7474/db/data/transaction/commit  
Content-Type: application/json  
Host: localhost:7474  
User-Agent: py2neo/3.0.0 HTTPStream/1.5.0 Python/3.5.0-final-0 (linux)  
X-Stream: true  
Authorization: Basic bmVNGo6cGFzc3dvcmQ=  
  
{"statements": [{"statement": "RETURN 1", "parameters": {},  
 "resultDataContents": ["REST"]}]}

### Response [213 bytes]

200 OK  
Date: Mon, 18 Apr 2016 09:23:12 GMT  
Content-Type: application/json  
Access-Control-Allow-Origin: \*  
Content-Length: 65  
Server: Jetty(9.2.9.v20150224)  
{"results": [{"columns": ["1"], "data": [{"rest": "1"}]}, {"errors": []}]}

JSON payload

### Bolt

Stateful: client and auth info only sent once per session

### Request [22 bytes]

00:0C:B2:10:8B:52:45:54:55:52:4E:20:31:A0:00:00  
00:02:B0:3F:00:00

### Response [39 bytes]

00:0D:B1:70:A1:B6:66:69:65:6C:64:73:91:81:31:00  
00:0B:04:B1:71:91:01:00:00:00:0A:B1:70:A1:B4:74  
79:70:65:B1:72:00:00

Type-safe binary payload

Significantly smaller requests and responses

- Unified type system
- Stateful (auth)
- Less data over wire
- Efficient serialization

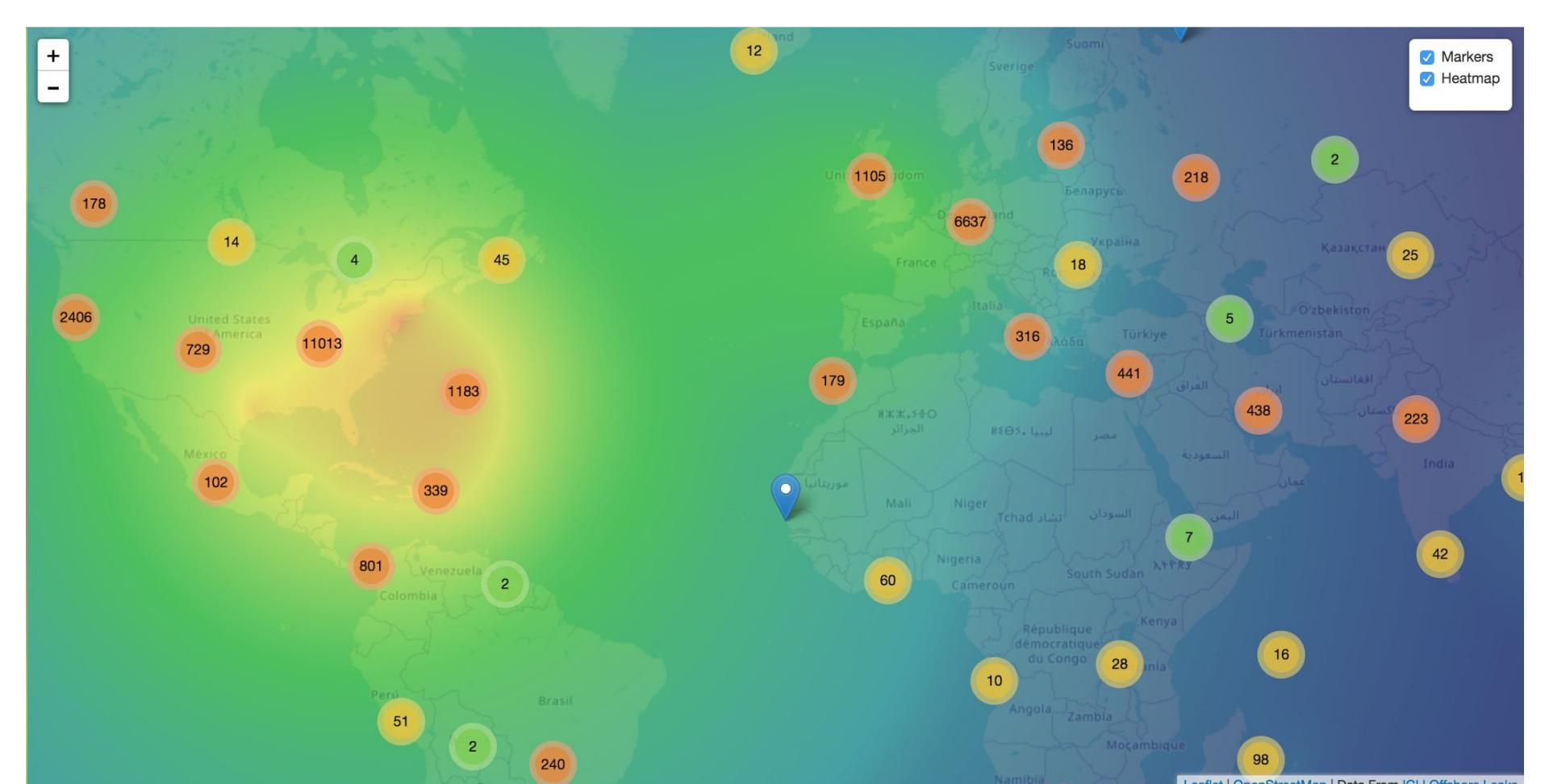
```

1 var driver = neo4j.v1.driver("bolt://localhost:7687", neo4j.v1.auth.basic("neo4j", "MYPASSWORD"));
2 var session = driver.session();
3 session
4   .run(`MATCH (a:Address)-[:REGISTERED_ADDRESS]-(o:Officer)-[:CONNECTED_TO|OFFICER_OF]-(e:Entity)
5       WHERE exists(a.latitude) and exists(a.longitude)
6       RETURN a.name AS address, a.latitude AS latitude, a.longitude AS longitude,
7             COLLECT(DISTINCT o.name) AS officers, COLLECT(DISTINCT e.name) AS entities,
8             COLLECT(DISTINCT e.jurisdiction_description) AS jurisdictions, 1.0*COUNT(*) AS strength`)
9   .subscribe({
10     onNext: function (record) {
11       //console.log(record);
12       var marker = new L.Marker(new L.LatLng(record.get('latitude'), record.get('longitude')));
13       marker.bindPopup('<b>Address:</b> ' + record.get('address') + '<br>' + '<b>Officers:</b> ' + record.get('officers').toString());
14       markerLayers.addLayer(marker);
15       heatmap.pushData(record.get('latitude'), record.get('longitude'), record.get('strength')*0.01);
16     },
17     onCompleted: function () {
18       var overlayMaps = {'Markers': markerLayers, 'Heatmap': heatmap};
19       var controls = L.control.layers(null, overlayMaps, {collapsed: false, autoZIndex: true});
20       map = new L.Map('map', {center: new L.LatLng(51.505, -0.09), zoom: 3, layers: [baseLayer]});
21       map.addLayer(markerLayers);
22       map.addLayer(heatmap);
23       controls.addTo(map);
24       session.close();
25     },
26     onError: function (error) {
27       console.log(error);
28     }
29   });

```

+  
-

- Markers
- Heatmap



Leaflet | OpenStreetMap | Data From ICIJ Offshore Leaks

<http://www.lyonwj.com/2017/11/28/geocoding-paradise-papers-neo4j-spatial-visualization/>

<http://www.lyonwj.com/pp-viz/heatmap/>

# openCypher

openCypher



About

Blog

Events

Usage of Cypher

New Features

Implementers Group

References

Resources

## What is openCypher?

The openCypher project aims to deliver a full and open specification of the industry's most widely adopted graph database query language: **Cypher**.

### ◎ Focus on Your Domain

Graphs naturally describe your domain, and Cypher lets you focus on that domain instead of getting lost in the mechanics of data access. Both *expressive* and *efficient*, Cypher is *intuitive* and immediately familiar.

### ◎ Human Readable

Cypher is a *human-readable* query language that makes common operations easy. A combination of English prose and *intuitive iconography*, Cypher is *accessible* to developers and operations professionals alike.

### ◎ Complete Open Source Access

The openCypher project means you can use Cypher as your query language for graph processing capabilities within any product or application. Databases like **SAP HANA Graph**, **Redis**, **AgensGraph** and **Neo4j** all use the Cypher query language – now you can too.

<http://www.opencypher.org/>

neo4j

# Cypher For Apache Spark

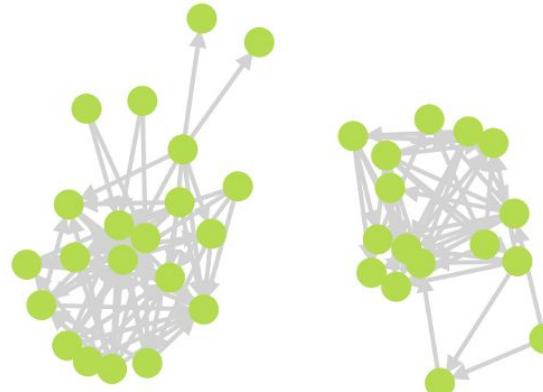
```
val CITYFRIENDS_NA = SN_NA.cypher(  
    """  
    MATCH (a:Person)-[:IS_LOCATED_IN]->(city:City)<-[:IS_LOCATED_IN]-(b:Person),  
    (a)-[:KNOWS*1..2]->(b)  
    WHERE city.name = "New_York" OR city.name = "San_Francisco"  
    RETURN GRAPH result OF (a)-[r:SIMILAR_CIRCLE]->(b)  
    """".stripMargin).graphs("result").cache
```

```
CITYFRIENDS_NA.asZeppelinGraph
```



Nodes 37 : Person

Relationships 99 : SIMILAR\_CIRCLE



<https://github.com/opencypher/cypher-for-apache-spark>





William Lyon

@lyonwj

Neo4j: Open source software that stores and queries data as nodes and relationships using the Cypher query language w/ index free adjacency.

3:44 PM - 23 Feb 2017 from [Dunlap, Seattle](#)



Reply to [@lyonwj](#)



<https://arxiv.org/pdf/1004.1001.pdf>

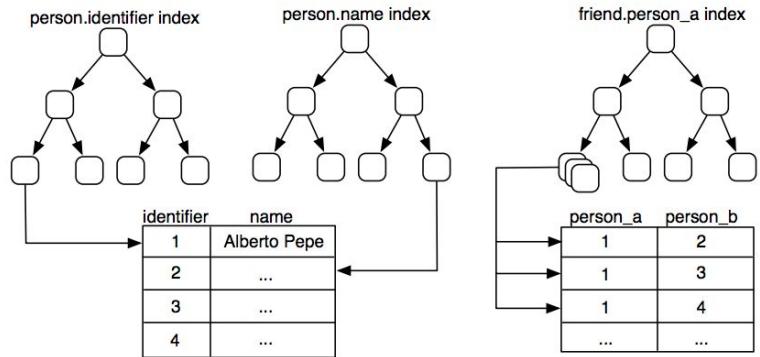


Fig. 1. A table representation of people and their friends.

# Neo4j Clustering Architecture



## Raft-based architecture

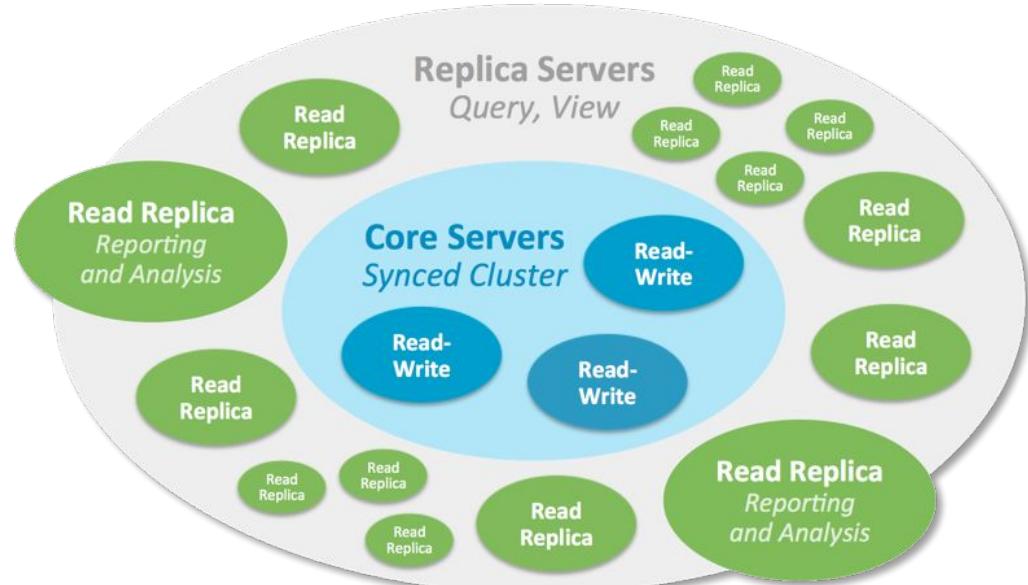
Consensus commits via “Core” servers

## Cluster-aware drivers

No need for external load balancer  
Stateful, cluster-aware sessions

## Causal Consistency

Stronger consistency model than Eventual Consistency



# Common Use Case

- Real time recommendations
- Fraud Detection
- Network & IT Management
- Social Networks
- Bill of Materials / Supply Chain
- Knowledge Graphs
- Master Data Management
- Access Management
- Microservices Analysis
- IoT
- Investigative Journalism
- ...

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.

# Russia Twitter Trolls



- **2752 Twitter accounts tied to Russia's Internet Research Agency**
- Accounts suspended by Twitter
  - Data deleted
- **What were they tweeting about?**

user id	handle
905874659358453760	10_gop
2528776985	1488reasons
2535564756	1D_Nicole_
79793901	1DRussianFNDM
1860330774	1Erik_Lee
838742761515991041	1lorenafava1
2537164155	2lauragibson
823484279787294720	2ndHalfOnion
80712878	2oliverbaker
80723578	30ToMarsFandom
3644473036	459Jisalge
2261091126	4claireevans
748870305280647168	4ever1937
4036537452	4MySquad
713665792550645760	666steverogers
2534875924	6Druz
3936735149	71bilaljamil1
2533199949	756shawn
2619792207	___Judith___D_
2556610923	___S0phia___
2595466901	_alshaw
2544663209	_Amy_Mccann_
2753352910	_AnnaSwanson
2540856370	_Becker_Gerald
2426927403	_beglov
2559290935	_Ben_Santos

# Wayback API

user id	handle
905874659358453760	10_gop
2528776985	1488reasons
2535564756	1D_Nicole_
79793901	1DRussianFNDM
1860330774	1Erik_Lee
838742761515991041	1lorenafava1
2537164155	2lauragibson
823484279787294720	2ndHalfOnion
80712878	2oliverbaker
80723578	30ToMarsFandom
3644473036	459Jisalge
2261091126	4claireevans
748870305280647168	4ever1937
4036537452	4MySquad
713665792550645760	666steverogers
2534875924	6Druz
3936735149	71bilaljamil1
2533199949	756shawn
2619792207	__Judith__D_
2556610923	__Sophia__
2595466901	_alshaw
2544663209	_Amy_Mccann_
2753352910	_AnnaSwanson
2540856370	_Becker_Gerald
2426927403	_beglov
2559290935	_Ben_Santos
2498964074	_Billy_Moyer_
2586365248	_Edward_Connor_
2644685369	_Gabriel_Pope_
2592814610	_GeorgeSchultz_
2449237434	_GregorVan
2622070817	_Howard_Good_



[http://archive.org/wayback/available?url=http://twitter.com/TEN\\_GOP](http://archive.org/wayback/available?url=http://twitter.com/TEN_GOP)



```
{  
    url: "http://twitter.com/TEN\_GOP",  
    archived_snapshots: {  
        - closest: {  
            status: "200",  
            available: true,  
            url: "http://web.archive.org/web/20170818065026/https://twitter.com/TEN\_GOP",  
            timestamp: "20170818065026"  
        }  
    }  
}
```



# Scraping Internet Archive...

Tennessee Mengetweet Semula

Tennessee @TEN\_GOP · 11 j

About a week ago a horde of Moroccans landed on the beach in Spain. This week a terror attack by a Moroccan in #Barcelona.



...

```
{"suggestion_details":{},"tweet_ids":  
"898272614224769024","scribe_component":"tweet"}>  
::before  
▼<div class="tweet js-stream-tweet js-actionable-tweet js-profile-  
popup-actionable dismissible-content  
original-tweet js-original-tweet  
  
has-cards has-content  
" data-tweet-id="898272614224769024" data-item-id=  
"898272614224769024" data-permalink-path="/TEN_GOP/status/  
898272614224769024" data-conversation-id="898272614224769024"  
data-tweet-nonce="898272614224769024-e0409e07-b9c9-416f-b4fd-  
08e3dc8781b0" data-retweet-id="898389832505929728" data-retweeter=  
"TEN_GOP" data-screen-name="TEN_GOP" data-name="Tennessee" data-  
user-id="4224729994" data-you-follow="false" data-follows-you=  
"false" data-you-block="false" data-reply-to-users-json=[  
{"id_str":"4224729994","screen_name":"TEN_GOP","name":  
"Tennessee","emojified_name":{"text":  
"Tennessee","emojified_text_as_html":"Tennessee"}}]} data-  
disclosure-type data-has-cards="true">> == $0  
::before  
►<div class="context">...</div>  
▼<div class="content">  
►<div class="stream-item-header">...</div>  
▼<div class="js-tweet-text-container">  
▼<p class="TweetTextSize TweetTextSize--normal js-tweet-text  
tweet-text" lang="en" data-aria-label-part="0">  
"About a week ago a horde of Moroccans landed on the  
beach in Spain. This week a terror attack by a Moroccan  
in "  
►<a href="/web/20170818065026/https://twitter.com/hashtag/  
Barcelona?src=hash" data-query-source="hashtag_click"  
class="twitter-hashtag pretty-link is-nav" dir="ltr">...</a>
```

# Scraping Internet Archive...

```
1  [
2    {
3      "tweet_id": 561931644785811457,
4      "screen_name": "AlwaysHungryBae",
5      "permalink": "/AlwaysHungryBae/status/561931644785811457",
6      "tweet_text": "Happy Super Bowl Sunday \n#superbowlfood pic.twitter.com/s6rwMtdLom",
7      "user_id": "2882130846",
8      "timestamp": 1422809918000,
9      "hashtags": [
10        {
11          "tag": "superbowlfood",
12          "archived_url": "/web/20150603004258/https://twitter.com/hashtag/superbowlfood?src=hash"
13        }
14      ],
15      "links": [
16        {
17          "url": "pic.twitter.com/s6rwMtdLom",
18          "archived_url": "http://web.archive.org/web/20150603004258/http://t.co/s6rwMtdLom"
19        },
20        {
21          "url": "https://pbs.twimg.com/media/B8xh2fFCQAE-vxU.jpg:large",
22          "archived_url": "/web.archive.org/web/20150603004258/https://twitter.com/AlwaysHungryBae/status/561931644785811457/photo/1"
23        }
24      ]
25    }
26  ]
```

# Scraping Internet Archive...

```
{  
    "tweet_id": 5013164470581157,  
    "screen_name": "AKeywayEnergyRes",  
    "user_id": 2031363034470581157,  
    "tweet_text": "Happy Super Bowl Sunday !\n#superbowl pic.twitter.com/skrWtD5m",  
    "user_id": 2031363034470581157,  
    "user_name": "AKeywayEnergyRes",  
    "hashtags": [  
        {"tag": "#superbowlofficial",  
         "archived_url": "http://web.archive.org/web/20160830042504/http://twitter.com/AKeywayEnergyRes/status/5013164470581157"},  
        {"tag": "#SuperBowl",  
         "archived_url": "http://web.archive.org/web/20160830042504/http://twitter.com/AKeywayEnergyRes/status/5013164470581157"},  
        {"tag": "#SuperBowlLIV",  
         "archived_url": "http://web.archive.org/web/20160830042504/http://twitter.com/AKeywayEnergyRes/status/5013164470581157"},  
        {"tag": "#SuperBowlLIV",  
         "archived_url": "http://web.archive.org/web/20160830042504/http://twitter.com/AKeywayEnergyRes/status/5013164470581157"}  
    ]  
}
```

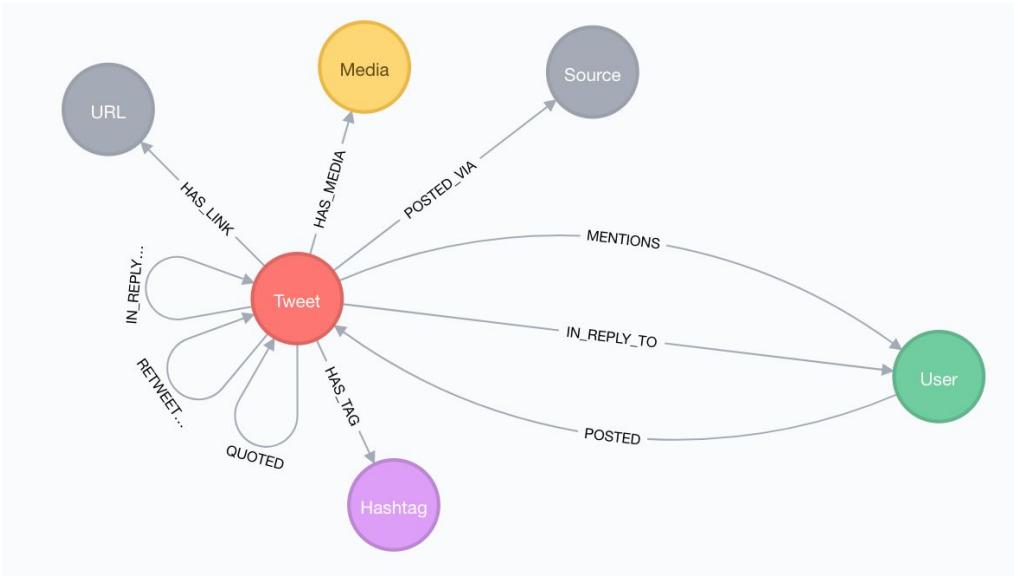


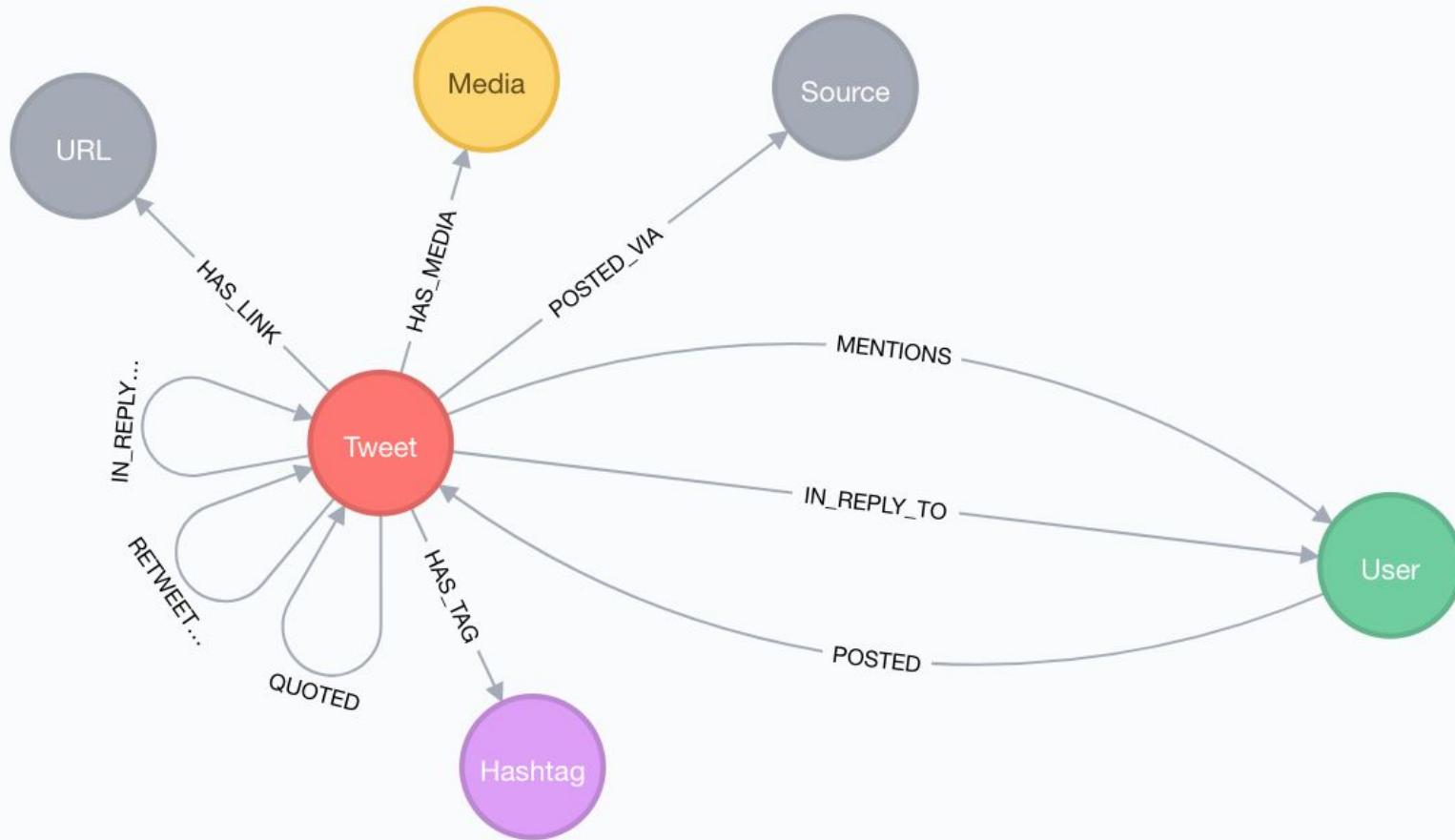
```
1 WITH $tweetArr AS tweets  
2 UNWIND tweets AS tweet  
3  
4 MERGE (u:User {user_id: tweet.user_id})  
5 ON CREATE SET u.screen_name = tweet.screen_name  
6  
7 MERGE (t:Tweet {tweet_id: tweet.tweet_id})  
8 ON CREATE SET t.text = tweet.tweet_text,  
     t.permalink = tweet.permalink  
9  
10 MERGE (u)-[:POSTED]->(t)  
11  
12 FOREACH (ht IN tweet.hashtags |  
13     MERGE (h:Hashtag {tag: ht.tag })  
14     ON CREATE SET h.archived_url = ht.archived_url  
15     MERGE (t)-[:HAS_TAG]->(h)  
16 )  
17  
18 FOREACH (link IN tweet.links |  
19     MERGE (l:Link {url: link.url})  
20     ON CREATE SET l.archived_url = link.archived_url  
21     MERGE (t)-[:HAS_LINK]->(l)  
22 )  
23 )
```

# Scraping Internet Archive...



```
1 WITH $tweetArr AS tweets
2 UNWIND tweets AS tweet
3
4 MERGE (u:User {user_id: tweet.user_id})
5 ON CREATE SET u.screen_name = tweet.screen_name
6
7 MERGE (t:Tweet {tweet_id: tweet.tweet_id})
8 ON CREATE SET t.text = tweet.tweet_text,
9     t.permalink = tweet.permalink
10
11 MERGE (u)-[:POSTED]->(t)
12
13 FOREACH (ht IN tweet.hashtags |
14     MERGE (h:Hashtag {tag: ht.tag })
15     ON CREATE SET h.archived_url = ht.archived_url
16     MERGE (t)-[E:HAS_TAG]->(h)
17 )
18
19 FOREACH (link IN tweet.links |
20     MERGE (l:Link {url: link.url})
21     ON CREATE SET l.archived_url = link.archived_url
22     MERGE (t)-[E:HAS_LINK]->(l)
23 )
```





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

# Passive Capture From Twitter API

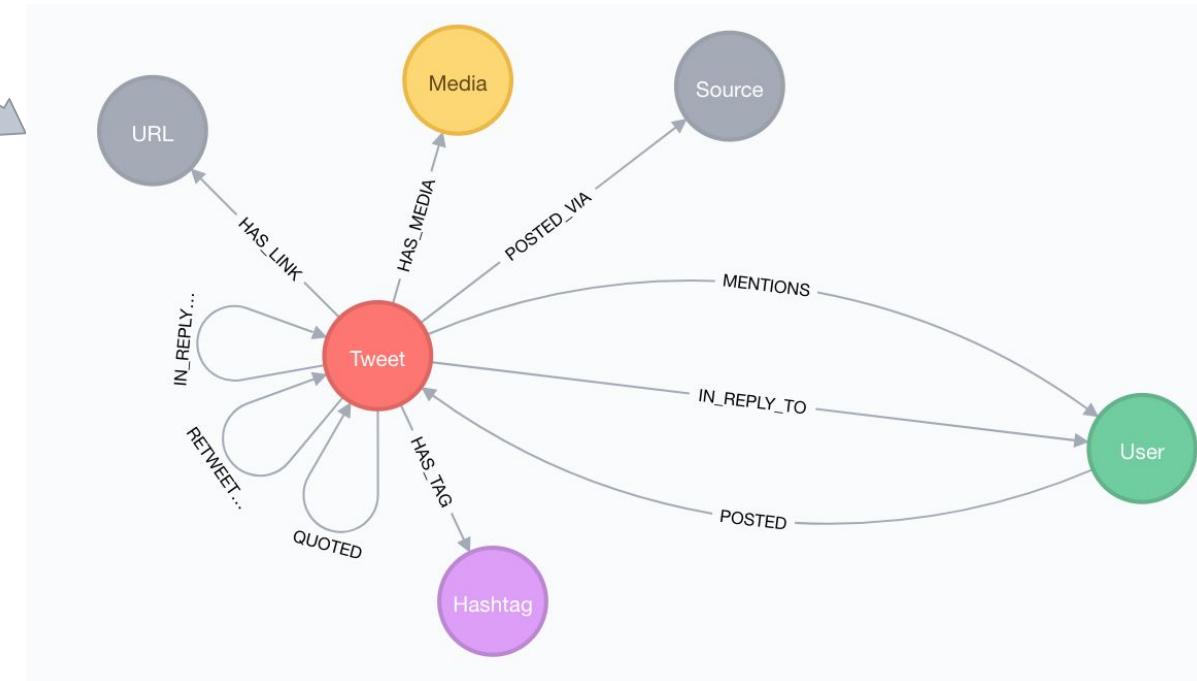


```
[  
  {  
    "contributors": null,  
    "coordinates": null,  
    "created_at": "Tue Oct 25 09:04:03 +0000 2016",  
    "display_text_range": [  
      0,  
      140  
    ],  
    "entities": {  
      "hashtags": [],  
      "symbols": [],  
      "urls": [  
        {  
          "display_url": "twitter.com/i/web/status/7-",  
          "expanded_url": "https://twitter.com/i/web/status/790841341643198464",  
          "indices": [  
            117,  
            140  
          ],  
          "url": "https://t.co/J437581mKq"  
        }  
      ],  
      "user_mentions": []  
    },  
    "extended_tweet": {  
      "display_text_range": [  
        0,  
        137  
      ],  
      "entities": {  
        "hashtags": [],  
        "symbols": [],  
        "urls": [  
          {  
            "display_url": "twitter.com/danielradosh/s-",  
            "expanded_url": "https://twitter.com/danielradosh/status/790568116023615488",  
            "indices": [  
              138,  
              161  
            ],  
            "url": "https://t.co/CAApSbra5y"  
          }  
        ],  
        "user_mentions": []  
      },  
      "full_text": "I'm sure parents concerned about the profanity (?) in Fahrenheit 451 are also Trump supporters who have no qualms about Trump's vulgarity https://t.co/CAApSbra5y"  
    },  
    "favorite_count": 0,  
    "favorited": false,  
    "filter_level": "low",  
    "geo": null,  
    "id": 790841341643198464,  
    "id_str": "790841341643198464",  
    "in_reply_to_status_id": null,  
    "in_reply_to_status_id_str": null,  
    "in_reply_to_user_id": null,  
    "in_reply_to_user_id_str": null,  
    "is_quote_status": false,  
    "lang": "en",  
    "place": null,  
    "retweet_count": 0,  
    "source": "Twitter for iPhone",  
    "text": "I'm sure parents concerned about the profanity (?) in Fahrenheit 451 are also Trump supporters who have no qualms about Trump's vulgarity https://t.co/CAApSbra5y",  
    "truncated": false,  
    "user": {  
      "contributors_enabled": false,  
      "default_profile": true,  
      "default_profile_image": false,  
      "description": "I'm a developer at a startup in San Francisco. I like to play with data and build things.",  
      "entities": {  
        "url": {  
          "url": "https://t.co/CAApSbra5y"  
        }  
      },  
      "follow_request_sent": false,  
      "followers_count": 1000,  
      "friends_count": 1000,  
      "geo_enabled": false,  
      "is_translator": false,  
      "language": "en",  
      "listed_in": {},  
      "location": "San Francisco, CA",  
      "name": "Daniel Radosh",  
      "profile_background_color": "#1DA1F2",  
      "profile_background_image": null,  
      "profile_background_tile": false,  
      "profile_image_url": "http://pbs.twimg.com/profile_images/790568116023615488/CAApSbra5y_normal.jpg",  
      "profile_image_url_https": "https://pbs.twimg.com/profile_images/790568116023615488/CAApSbra5y_normal.jpg",  
      "profile_link_color": "#000000",  
      "profile_sidebar_border_color": "#FFFFFF",  
      "profile_sidebar_fill_color": "#F0F0F0",  
      "profile_text_color": "#000000",  
      "profile_use_background_image": true,  
      "protected": false,  
      "screen_name": "danielradosh",  
      "statuses_count": 1000,  
      "time_zone": "Pacific Time (US & Canada)",  
      "url": "https://t.co/CAApSbra5y",  
      "verified": false,  
      "withheld": false  
    }  
  }  
]
```

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



```
{
  "contributors": null,
  "coordinates": null,
  "created_at": "Tue Oct 25 09:04:03 +0000 2016",
  "display_text_range": [
    0,
    140
  ],
  "entities": {
    "hashtags": [],
    "symbols": [],
    "urls": [
      {
        "display_url": "twitter.com/web/status/7",
        "expanded_url": "https://twitter.com/web/status/790841341543108464",
        "indices": [
          0,
          140
        ],
        "url": "https://t.co/J43758586q"
      }
    ],
    "user_mentions": []
  },
  "extended_tweet": {
    "display_text_range": [
      0,
      117
    ],
    "entities": {
      "hashtags": [],
      "symbols": [],
      "urls": [
        {
          "display_url": "twitter.com/danielradish/",
          "expanded_url": "https://twitter.com/danielradish/status/79056816603615468",
          "indices": [
            0,
            102
          ],
          "url": "https://t.co/CAnqGra5"
        }
      ],
      "user_mentions": []
    }
  },
  "full_text": "I'm sure parents concerned about the profanity (?) in Fahrenheit 451 are also Trump supporters who h
  ...
  "geo": null,
  "id": 790841341543108464,
  "id_str": "790841341543108464",
  "in_reply_to_status_id": null,
  "in_reply_to_status_id_str": null,
  "in_reply_to_user_id": null,
  "in_reply_to_user_id_str": null,
  "is_quote_status": false,
  "lang": "en",
  "place": null,
  "retweet_count": 0,
  "source": "Twitter for iPhone",
  "text": "About a week ago a horde of Moroccans landed on the beach in Spain. This was a terror attack by a Morocca
  ...
  "truncated": false,
  "user": {
    "contributors": null,
    "created_at": "Mon Jul 11 14:44:00 +0000 2016",
    "description": "Tennessee Mountain Service | 800.251.8771 | 1-800-251-8771 | About a week ago a horde of Moroccans landed on the beach in Spain. This was a terror attack by a Morocca
    ...
    "entities": {
      "url": "https://t.co/0D9B1B0000"
    },
    "favourites_count": 0,
    "followers_count": 0,
    "friends_count": 0,
    "geo_enabled": false,
    "id": 790841341543108464,
    "id_str": "790841341543108464",
    "is_translator": false,
    "language": "en",
    "listed_count": 0,
    "location": "Tennessee, USA",
    "name": "Tennessee Mountain Service | 800.251.8771 | 1-800-251-8771 | About a week ago a horde of Moroccans landed on the beach in Spain. This was a terror attack by a Morocca
    ...
    "profile_background_color": "000000",
    "profile_background_image_url": "http://abs.twimg.com/images/themes/theme1/bg.png",
    "profile_background_image_url_https": "https://abs.twimg.com/images/themes/theme1/bg.png",
    "profile_background_tile": false,
    "profile_image_url": "http://abs.twimg.com/images/themes/theme1/avatar_normal.png",
    "profile_image_url_https": "https://abs.twimg.com/images/themes/theme1/avatar_normal.png",
    "profile_link_color": "000000",
    "profile_sidebar_border_color": "000000",
    "profile_sidebar_fill_color": "000000",
    "profile_text_color": "000000",
    "profile_use_background_image": true,
    "screen_name": "TMS_TechSupport",
    "statuses_count": 0,
    "time_zone": null,
    "url": "http://www.tennesseemtnservice.com",
    "verified": false
  }
}
```



# Your typical Russian Twitter Troll?

# Your typical American Citizen?



@LeroyLovesUSA

Cleveland Online

@OnlineCleveland

Breaking news, weather, traffic  
and more for Cleveland. DM us  
anytime. RTs not endorsements

📍 City of Cleveland, USA

@ClevelandOnline

# Your typical Local News Publication?



Tennessee

@TEN\_GOP

Unofficial Twitter of Tennessee  
Republicans. Covering breaking  
news, national politics, foreign  
policy and more. #MAGA #2A

# Your typical Local Political Party?

@TEN\_GOP

# Your typical Russian Troll



@ClevelandOnline

# Your typical Russian Troll



@LeroyLovesUSA

# Your typical Russian Troll

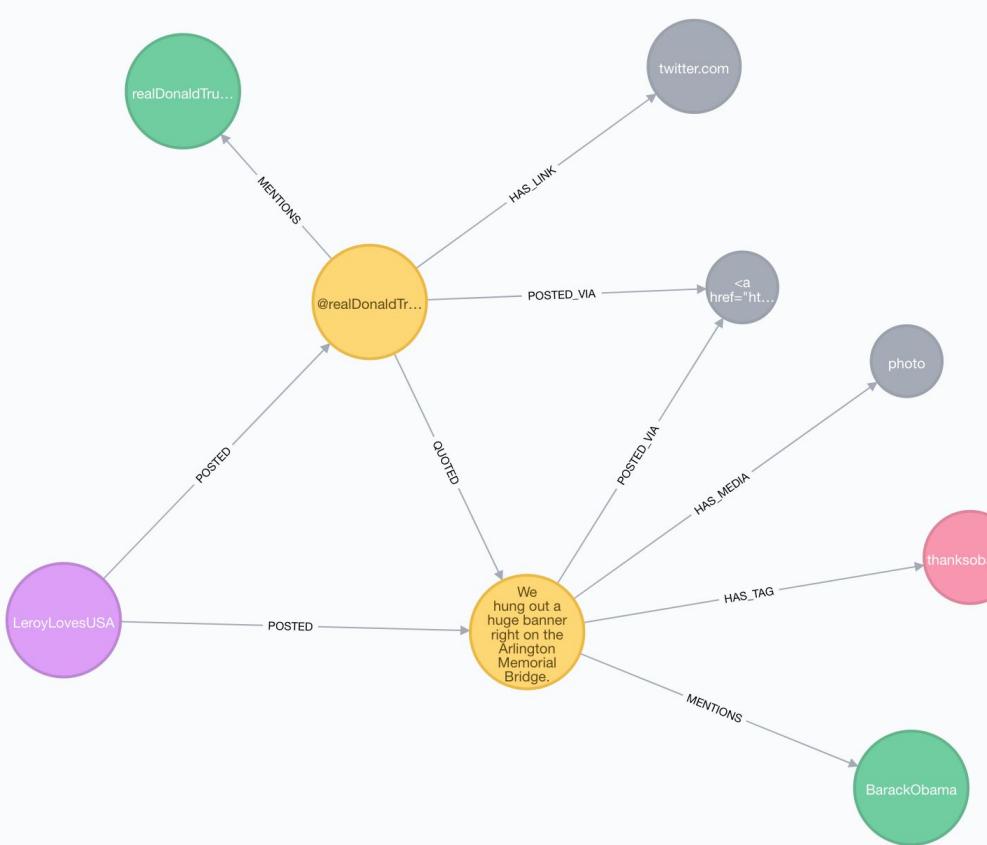


@TEN\_GOP

1 MATCH

2 (u:User {screen\_name: "LeroyLovesUSA"})-[:POSTED]->(t:Tweet)-[:HAS\_TAG]->(ht:Hashtag {key: "thanksobama"})

3 RETURN \*



**Leroy Barton** (@LeroyLovesUSA) [Follow](#)

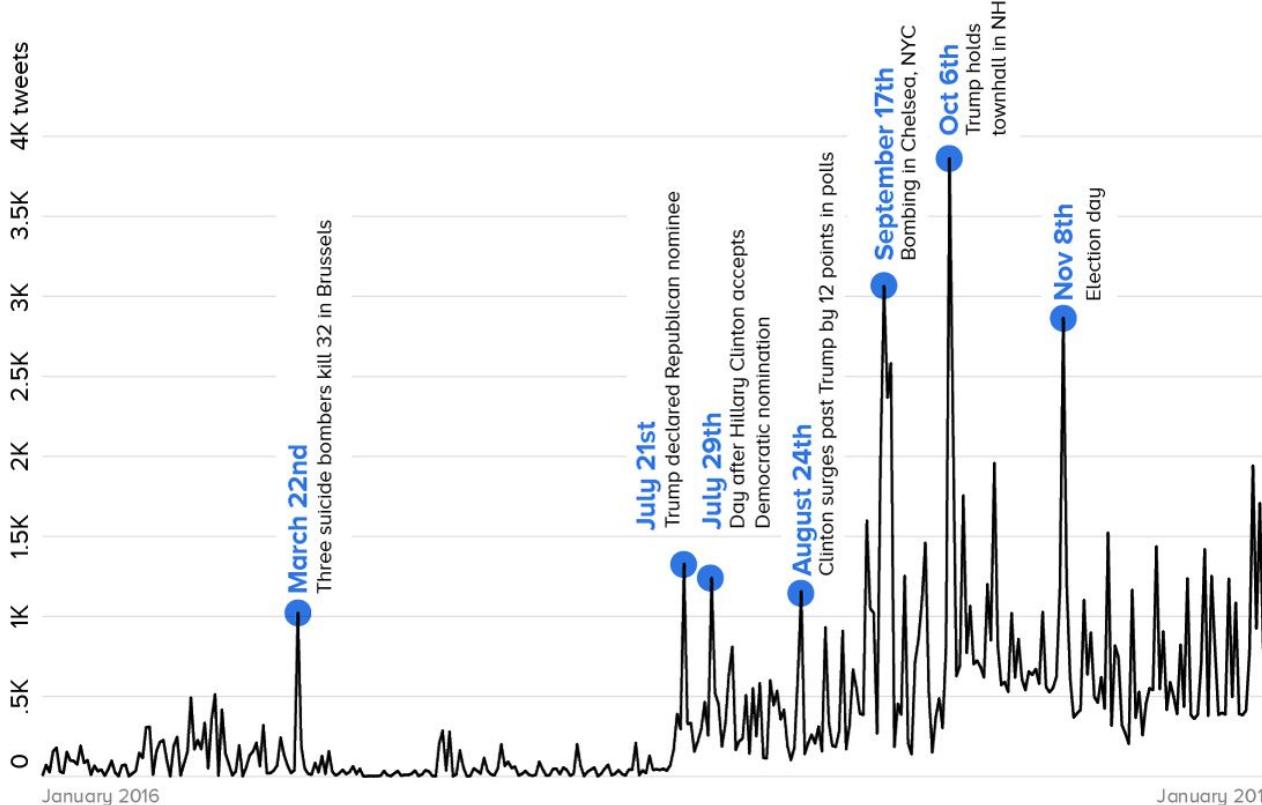
We hung out a huge banner right on the Arlington Memorial Bridge. Goodbye to murderer @BarackObama #ThanksObama

RETWEETS 323 LIKES 362

9:53 AM - 10 Nov 2016

323 362

## Russian Troll Volume Spiked During 2016 Campaign Events

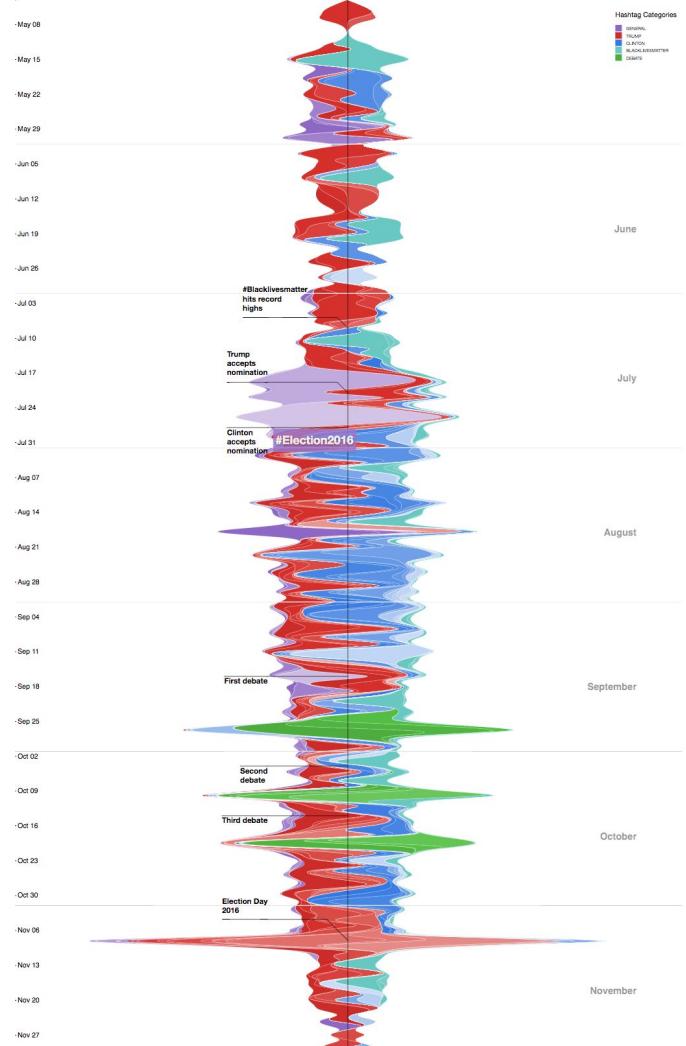


Source: Recovered Twitter API data



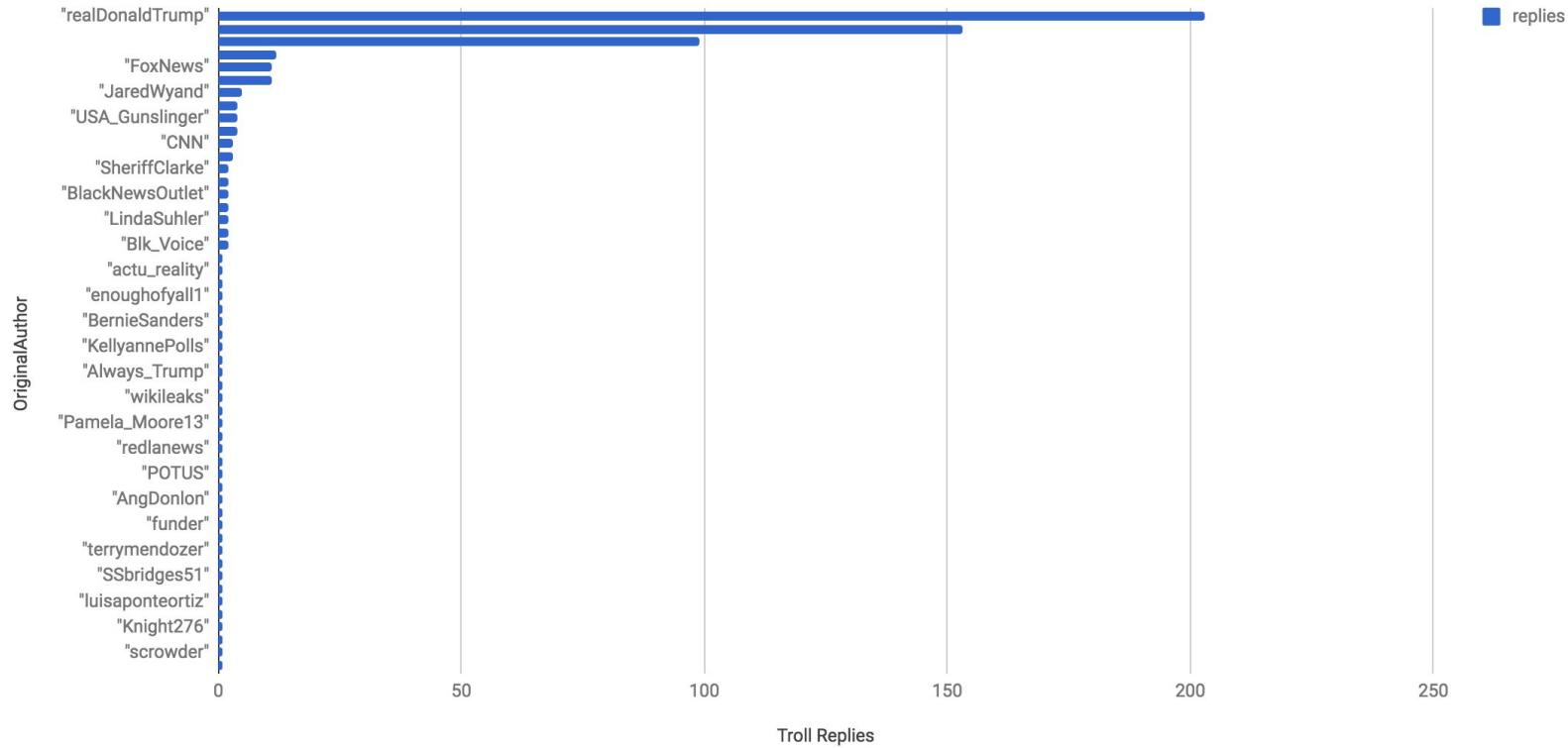
# Hashtags

- Use of hashtags to gain visibility and insert into conversation
- **@WorldOfHashtags**
  - #RejectedDebateTopics

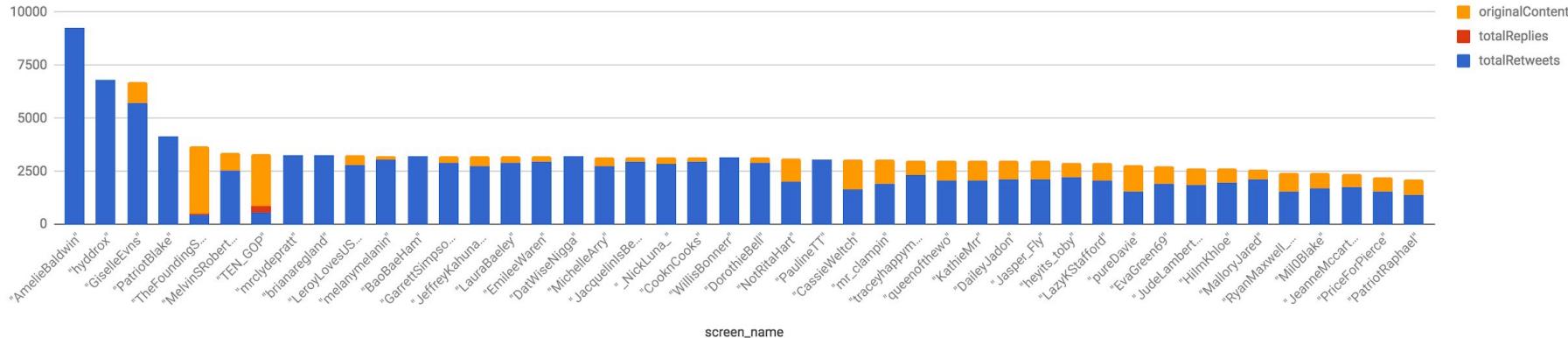


<https://www.nbcnews.com/tech/social-media/russian-trolls-went-attack-during-key-election-moments-n827176>

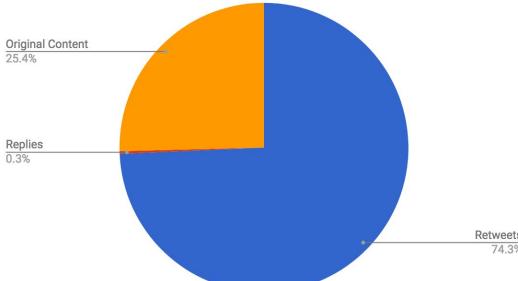
## Replies vs. Original Author



totalRetweets, totalReplies and originalContent



Summary Troll Behavior



```
1 MATCH (tr:Troll)-[:POSTED]->(tw:Tweet) WITH tr, tw
2 OPTIONAL MATCH (tw)-[:RETWEETED]-(rt:Tweet)
3 OPTIONAL MATCH (tw)-[:IN_REPLY_TO]-(irp:Tweet)
4 RETURN DISTINCT tr.screen_name AS screen_name, COUNT(tw) AS totalTweets,
5   COUNT(rt) AS totalRetweets, COUNT(irp) AS totalReplies,
6   (COUNT(tw) - (COUNT(rt) + COUNT(irp))) AS originalContent
7 ORDER BY totalTweets DESC;
```

```

1 MATCH (:Troll)-[:POSTED]->(tw:Tweet)-[:POSTED_VIA]->(s:Source)
2 RETURN DISTINCT s.name as source, COUNT(tw) as tweets
3 ORDER BY tweets DESC

```

Source	Tweets
Twitter Web Client	42684
twitterfeed	6926
TweetDeck	6409
<a href="#">Twibble.io</a>	1491
<a href="#">dlvr.it</a>	243
Twitter for iPhone	152
Twitter for Android	94
Twitter for iPad	21
Medium	17
Buffer	13
OS X	8
<a href="#">Wordpress.com</a>	6
IFTTT	5
Google	3
Tweetbot for iOS	3
Put your button on any page!	3
pourtwi	2
Twitter for Android Tablets	1
NovaPress Publisher	1

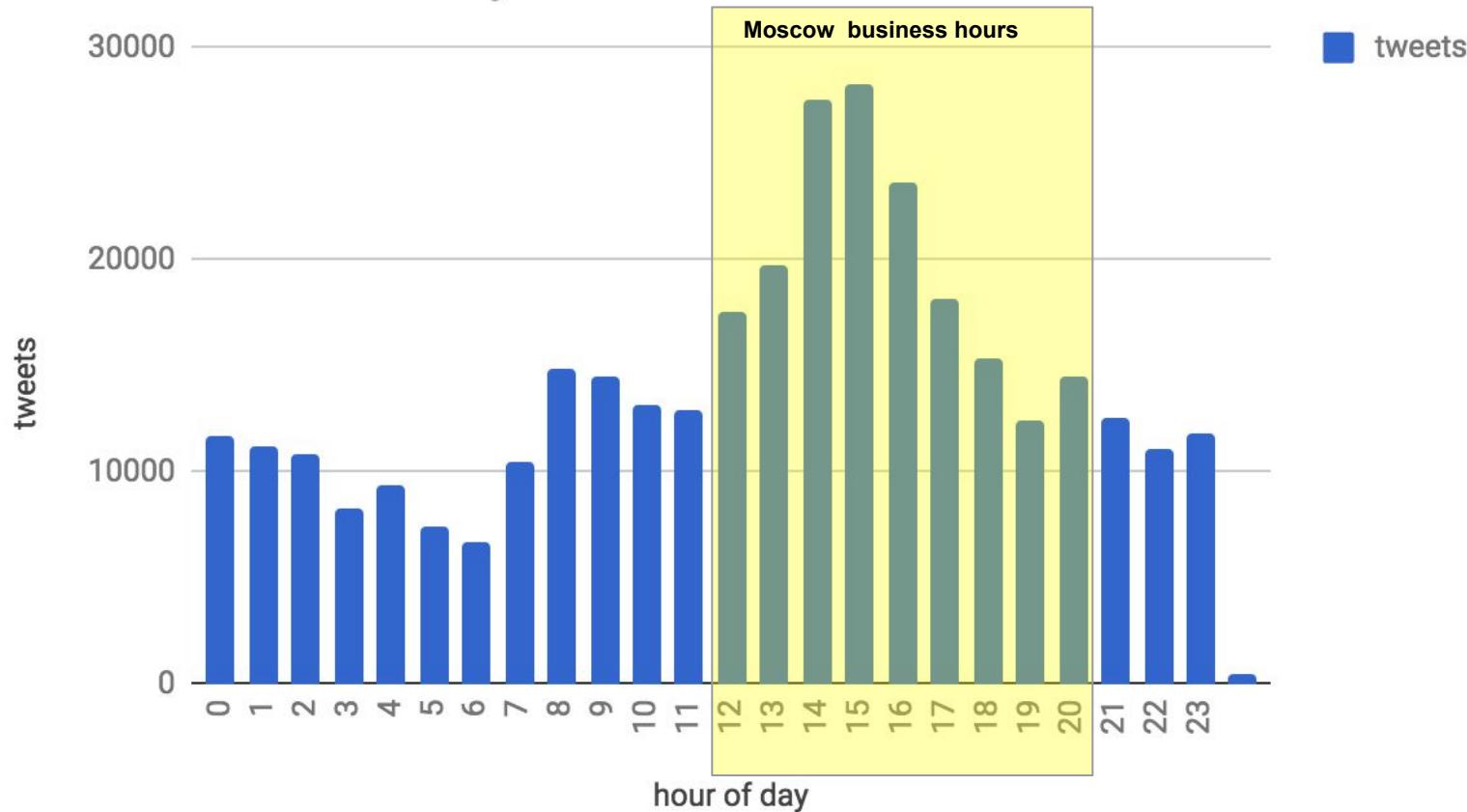
```

1 MATCH (:Troll)-[:POSTED]->(oc:Tweet)-[:HAS_LINK]-(u:URL)
2 RETURN DISTINCT u.domain as DomainOfTrollLink, COUNT(u) as LinksPosted
3 ORDER BY LinksPosted desc;

```

DomainOfTrollLink	LinksPosted
"bit.ly"	3431
"twitter.com"	1597
"twibble.io"	945
"wapo.st"	394
"www.wcvb.com"	210
"sh.st"	203
"www.nytimes.com"	136
"politics.blog.ajc.com"	104
"www.huffingtonpost.com"	98
"dailym.ai"	84
"www.cbsnews.com"	77
"cbsloc.al"	68
"politi.co"	64
"www.chron.com"	61
"www.usatoday.com"	56
"bsun.md"	56
"www.breitbart.com"	54
"www.washingtonpost.com"	54
"www.cbs8.com"	53
"www.wbaltv.com"	50
"www.nbcchicago.com"	49
"www.reuters.com"	49
"www.chicagotribune.com"	48
"www.local10.com"	47
"www.youtube.com"	44
"www.azcentral.com"	41

## tweets vs. hour of day



# Natural Language Processing

With Cypher and Neo4j



# Common NLP Tasks

- Language detection
- Part of speech tagging
- Word similarities
- Keyword extraction
- Topic extraction
- Concept hierarchy
- Sentiment analysis
- Entity extraction
- Entity merging
- Word associations
- Summarization
- Opinion mining

## Named Entity Recognition:

1 President Xi Jinping of China, on his first state visit to the United States, showed off his familiarity with American history and pop culture on Tuesday night.

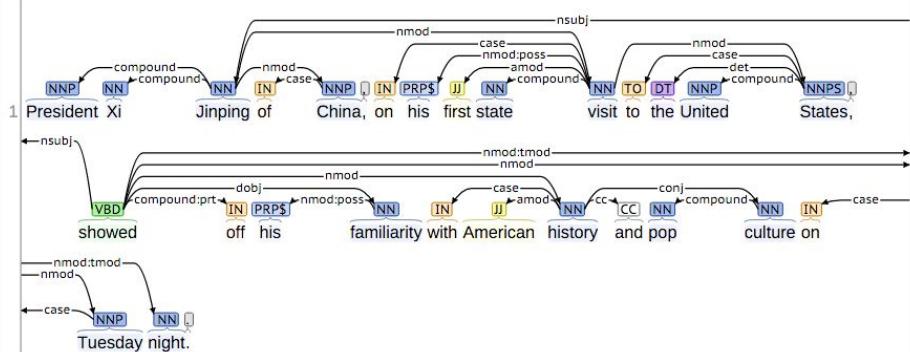
Annotations: Person, Loc, ORDINAL, Location, Misc, Date, Time

## Coreference:

1 President Xi Jinping of China, on his first state visit to the United States, showed off his familiarity with American history and pop culture on Tuesday night.

Mention: M  
Coref: M

## Basic Dependencies:



NLTK  
Textblob  
Polyglot  
TextRank



ConceptNet  
An open, multilingual knowledge graph



# NLP w/ Graph Databases

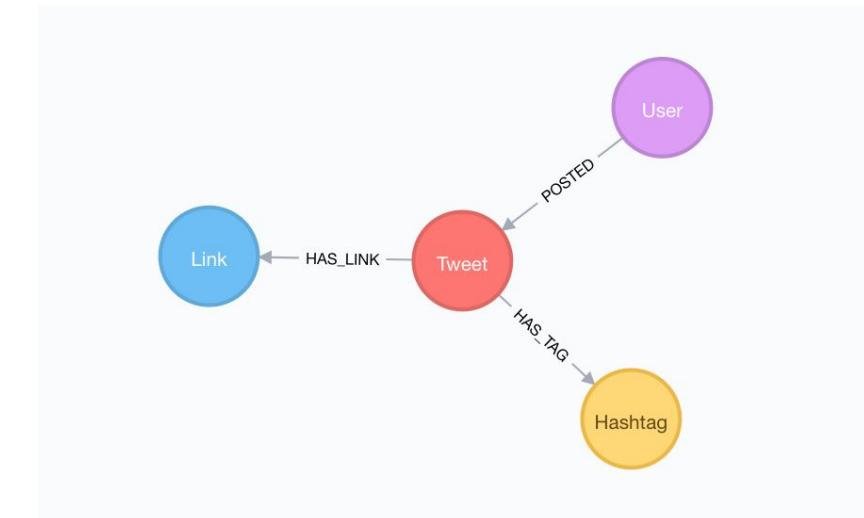
## Annotations

### Named Entity Recognition:

1 Chase Manhattan and its merger partner J.P.Morgan and Citibank, which was invo  
for Raul Salinas de Gortari, brother of a former Mexican president, to banks in Swi  
sign on.

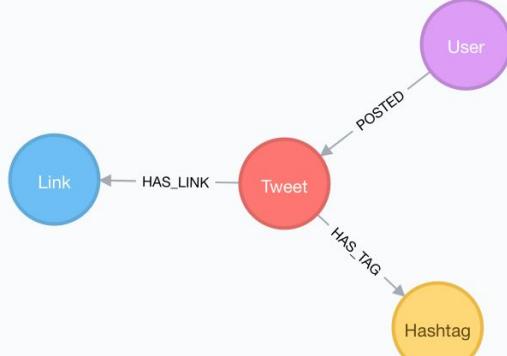
### Basic dependencies:

1 Chase Manhattan and its merger partner J.P. Morgan and Citibank, which

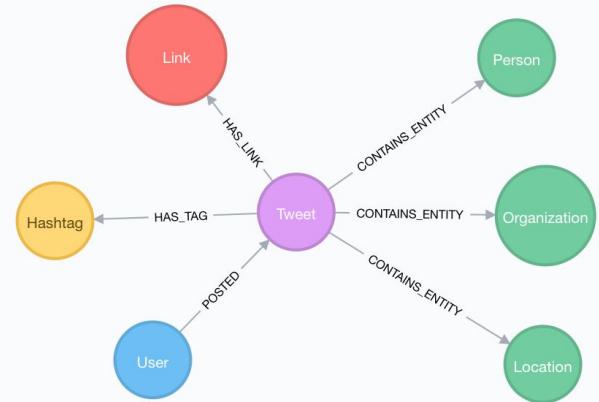


# NLP w/ Graph Databases

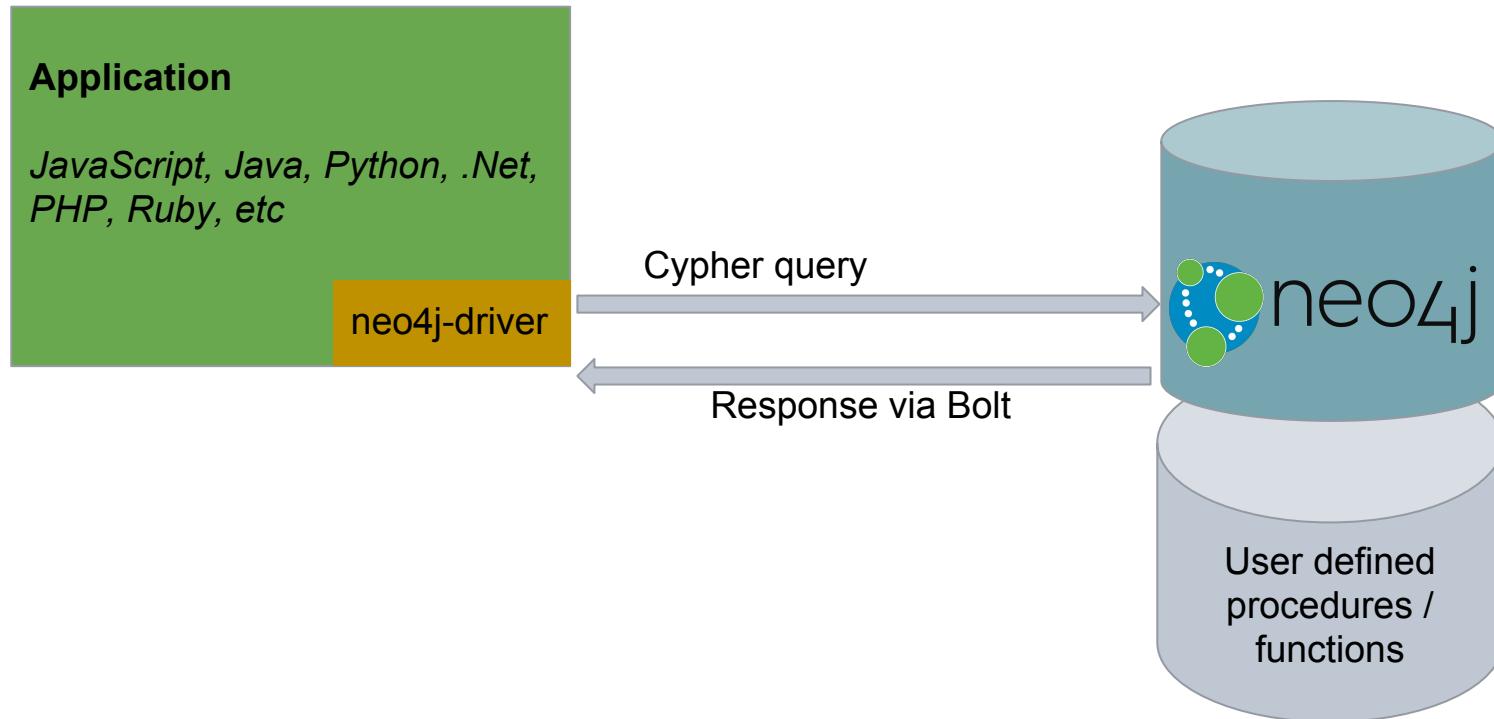
*Annotations*



*NLP  
Process*



# How to combine open source NLP tools w/ Neo4j?



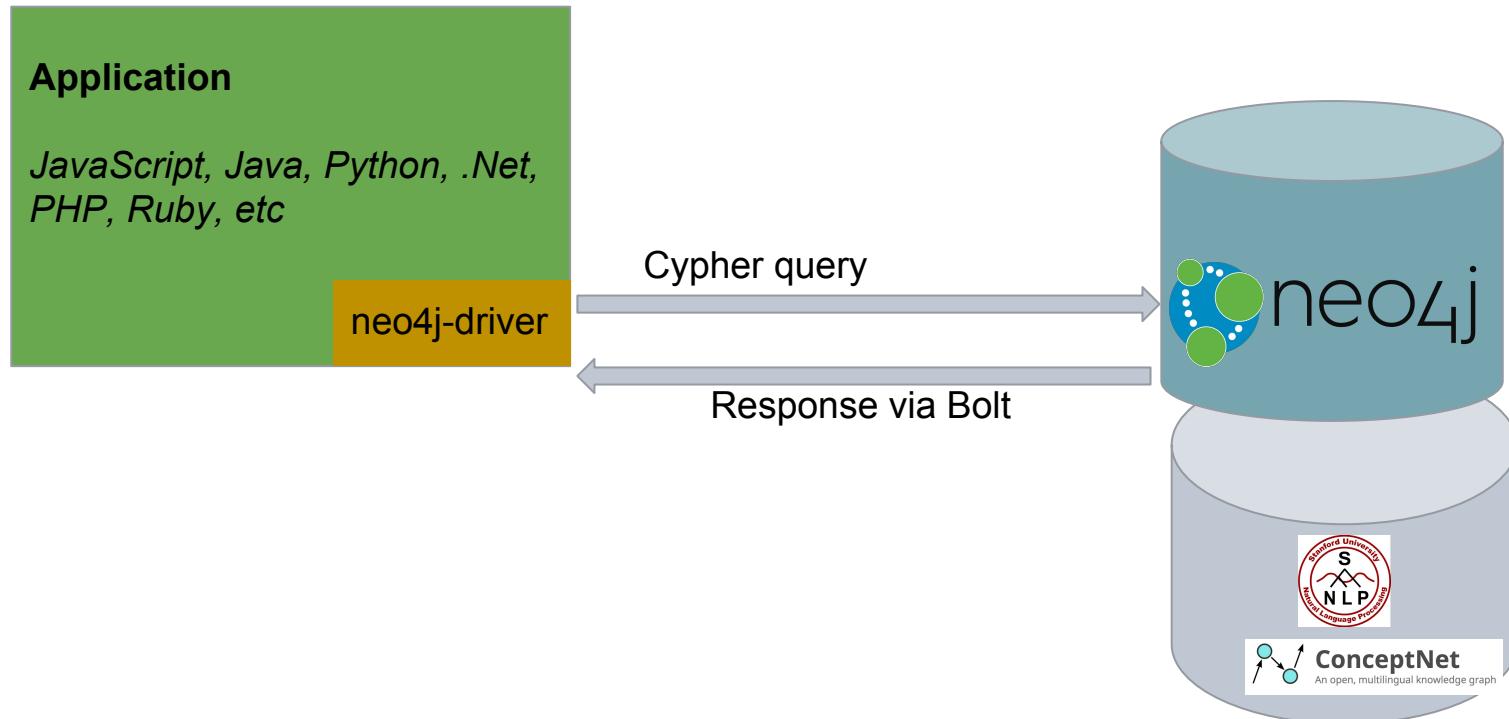
# How to combine open source NLP tools w/ Neo4j?

```
public class YourProcedureClass {  
    @Context  
    public GraphDatabaseService db; // Inject the database for easy access  
  
    @Procedure  
    @Description("<explain yourProcedure here>")  
    public Stream<OutputRecord> yourProcedure(@Name("arg1") SomeType arg1, ... )  
    {  
        // TODO: Your implementation  
    }  
}  
  
public class OutputRecord {  
    public final SomeType value; // Your result fields  
    ...  
  
    public OutputRecord(SomeType value)  
    {  
        this.value = value;  
    }  
}
```



Name of Procedure Output

# How to combine open source NLP tools w/ Neo4j?



# How to combine open source NLP tools w/ Neo4j?



## GraphAware Natural Language Processing

build passing This [Neo4j](#) plugin offers Graph Based Natural Language Processing capabilities.



```
1 MATCH (tw:Tweet {lang: "en"})  
2 CALL ga.nlp.annotate({text: tw.text, id: id(tw)})  
3 YIELD result  
4 MERGE (tw)-[:HAS_ANNOTATED_TEXT]->(result)  
5 RETURN count(result)
```

<https://github.com/graphaware/neo4j-nlp>

**PEOPLE MENTIONED**

Person	Num
"donald trump"	5676
"hillary clinton"	4792
"bill clinton"	800
"mike pence"	376
"tim kaine"	370
"micelle obama"	358
"bernie sanders"	218
"barack obama"	190
"rudy giuliani"	188
"david clarke"	170
"ted cruz"	164
"ben carson"	138
"sheriff clarke"	136
"newt gingrich"	136
"topnews obama"	128
"george soros"	114
"colin powell"	106
"topvideo trump"	88
"trump foundation"	86
"paul ryan"	86
"chelsea clinton"	84
"topvideo obama"	80
"reagan"	78

**LOCATIONS MENTIONED**

Location	Num
"white house"	760
"new york"	232
"united states"	188
"milwaukee"	142
"north carolina"	138
"munich"	126
"saudi arabia"	104
"baltimore"	104
"philippines"	84
"paris"	84
"north korea"	78
"st louis"	76
"pittsburgh"	68
"trump tower"	68
"utah"	68
"baton rouge"	56
"middle east"	56
"new york city"	56
"laos"	52
"atlantic city"	44
"new jersey"	44

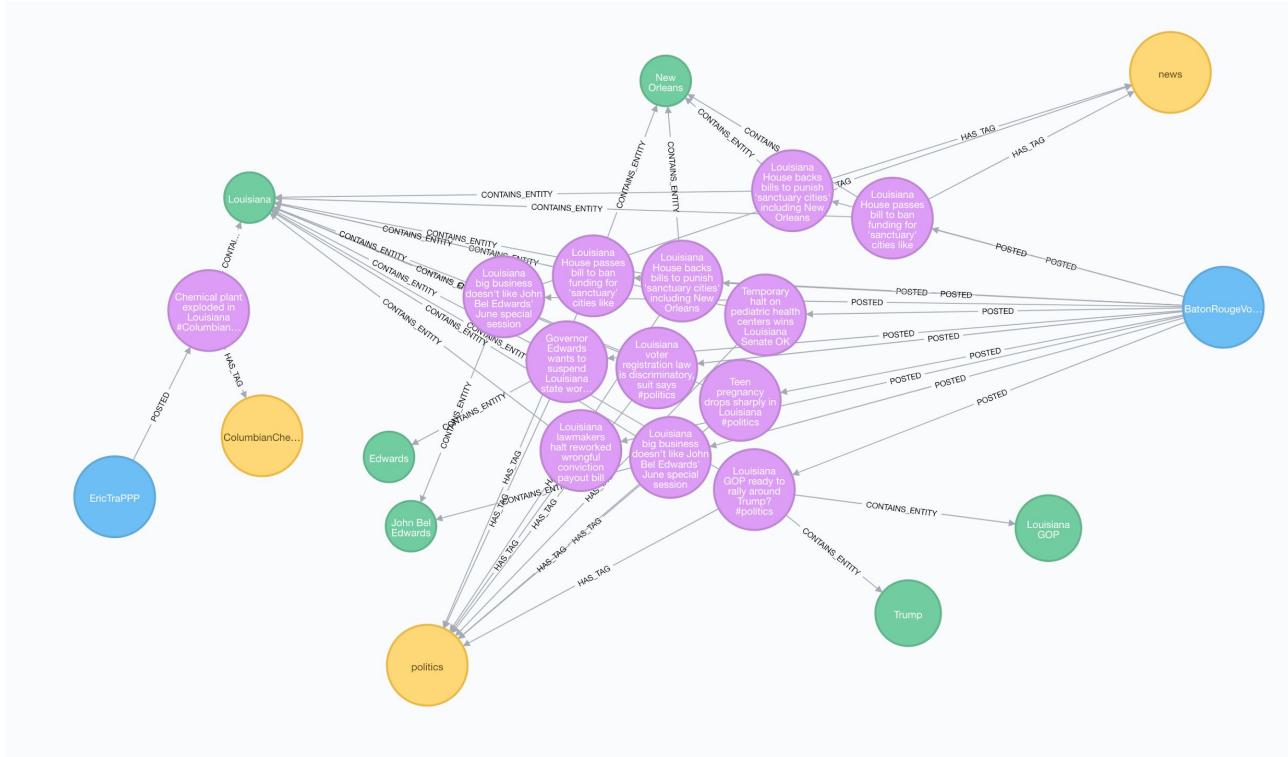
**ORGANIZATIONS MENTIONED**

Organization	Num
"white house"	760
"new york"	232
"north carolina"	138
"hillarys health"	106
"democratic party"	106
"supreme court"	100
"republican party"	98
"state department"	92
"democratic national convention"	84
"gop convention"	76
"trump tower"	68
"trump university"	58
"justice department"	56
"islamic state"	52
"secret service"	46
"walk of fame"	44
"house gop"	40
"air force"	40
"rust belt"	36
"nba"	36

```

1 // Tweets, hashtags, and entities around Louisiana
2 MATCH (u:User)-[:POSTED]->(t:Tweet)-[:CONTAINS_ENTITY]->(l:Location {name: "Louisiana"})
3 OPTIONAL MATCH (t)-[:HAS_TAG]->(ht:Hashtag)
4 OPTIONAL MATCH (t)-[:CONTAINS_ENTITY]->(e)
5 RETURN *

```



# Graph Algorithms

# Centrality

Measure of importance

## PageRank

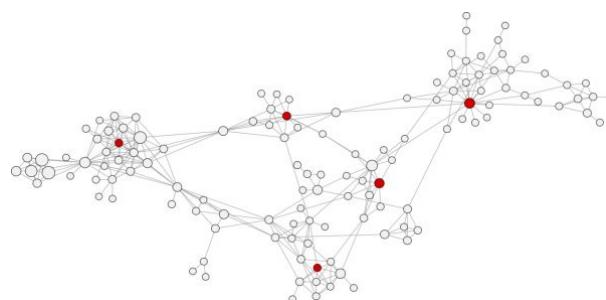
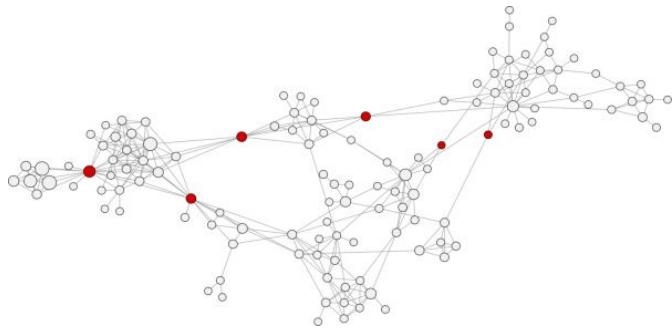
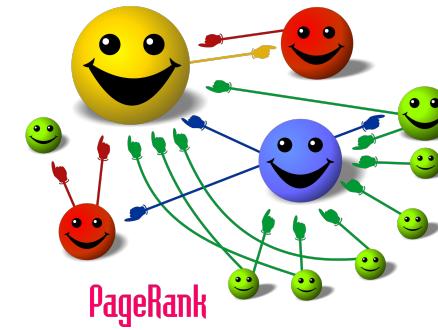
- Recursive
- Importance and number of connected nodes

## Betweenness Centrality

- Number of shortest paths connecting all pairs in the network

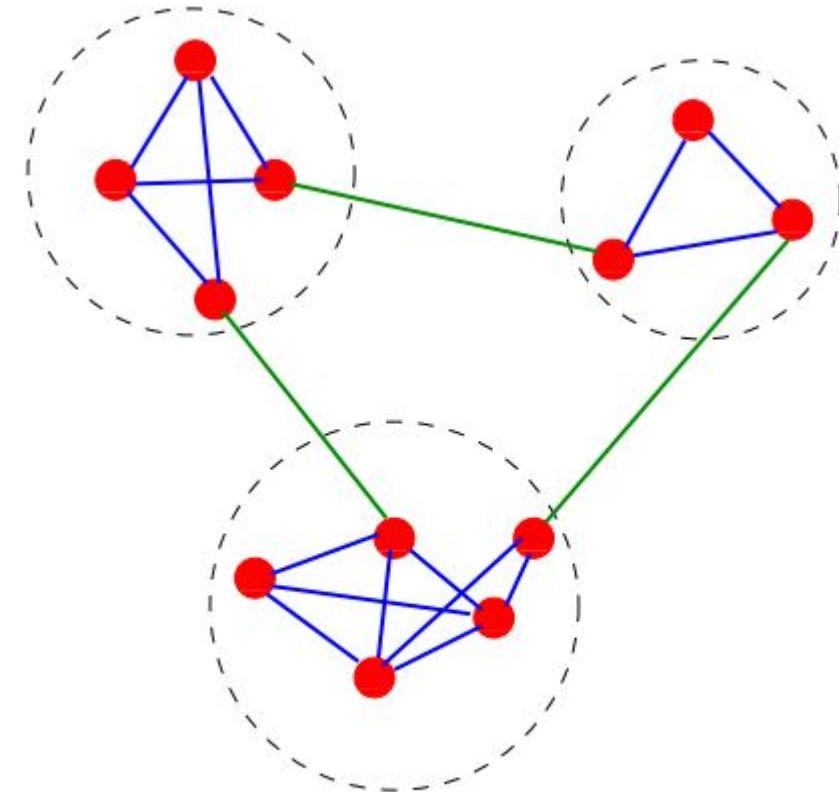
## Closeness Centrality

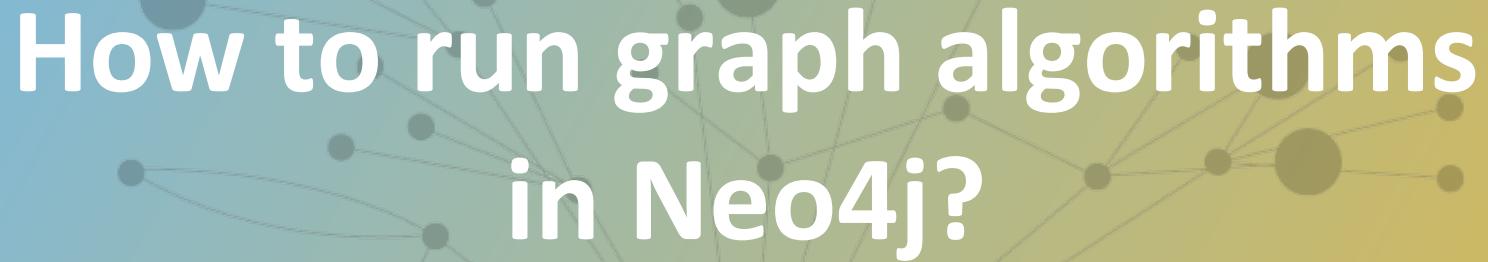
- Inverse of distance to all other nodes in the network



# Community Detection / Clustering

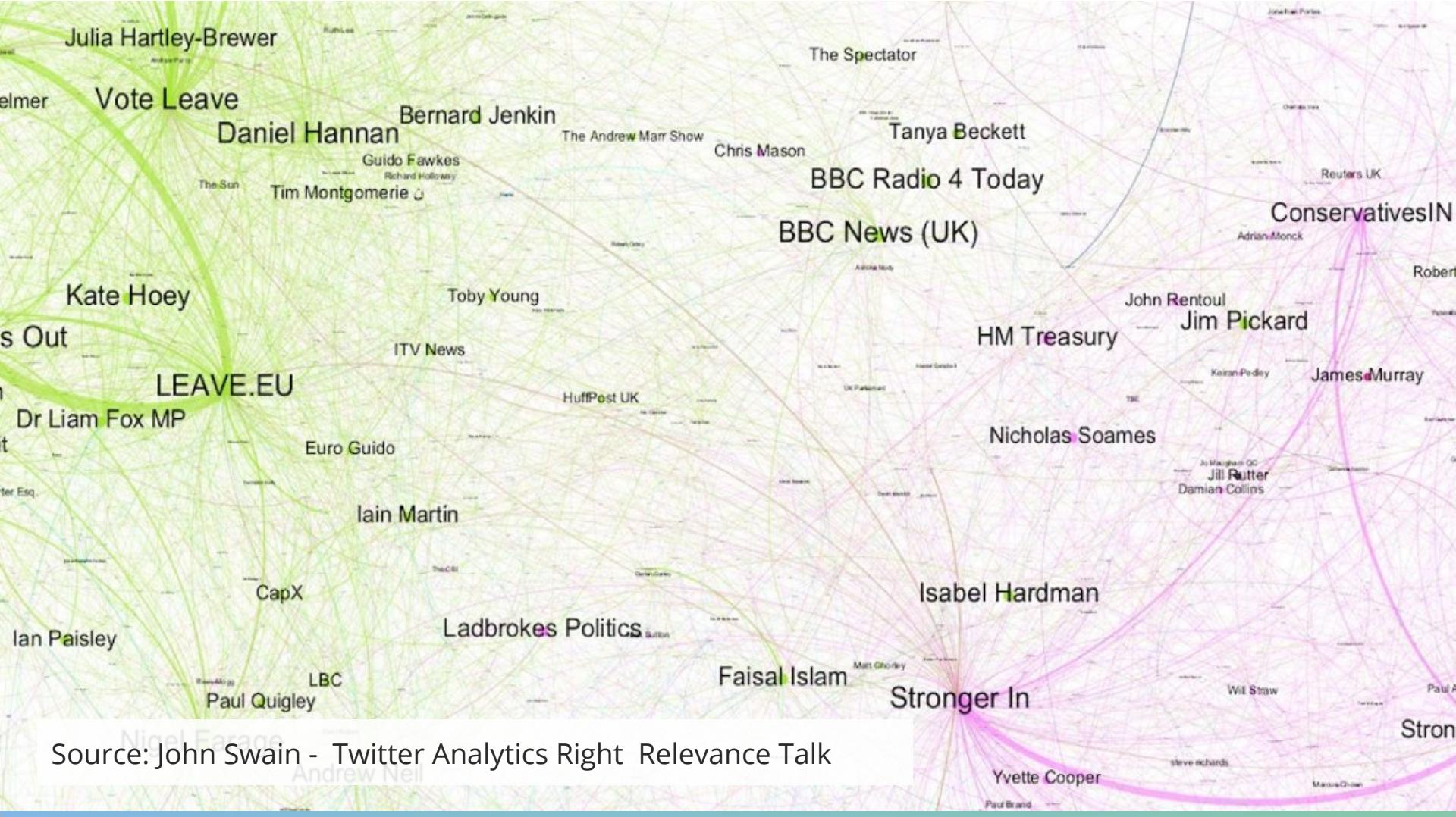
- Segment graph
- Minimize modularity
- Iterative Algorithms
  - Louvain
  - Union Find
  - Walktrap



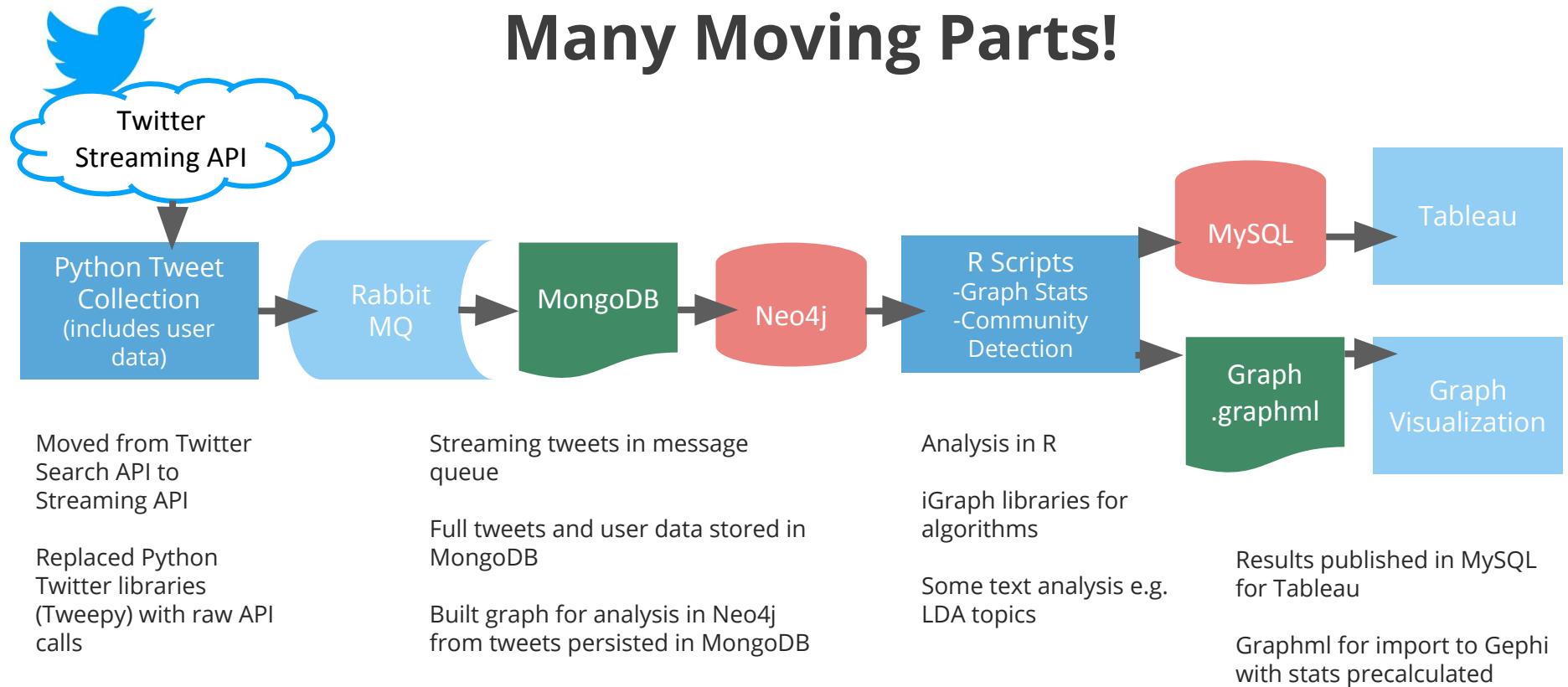


# How to run graph algorithms in Neo4j?





# Many Moving Parts!

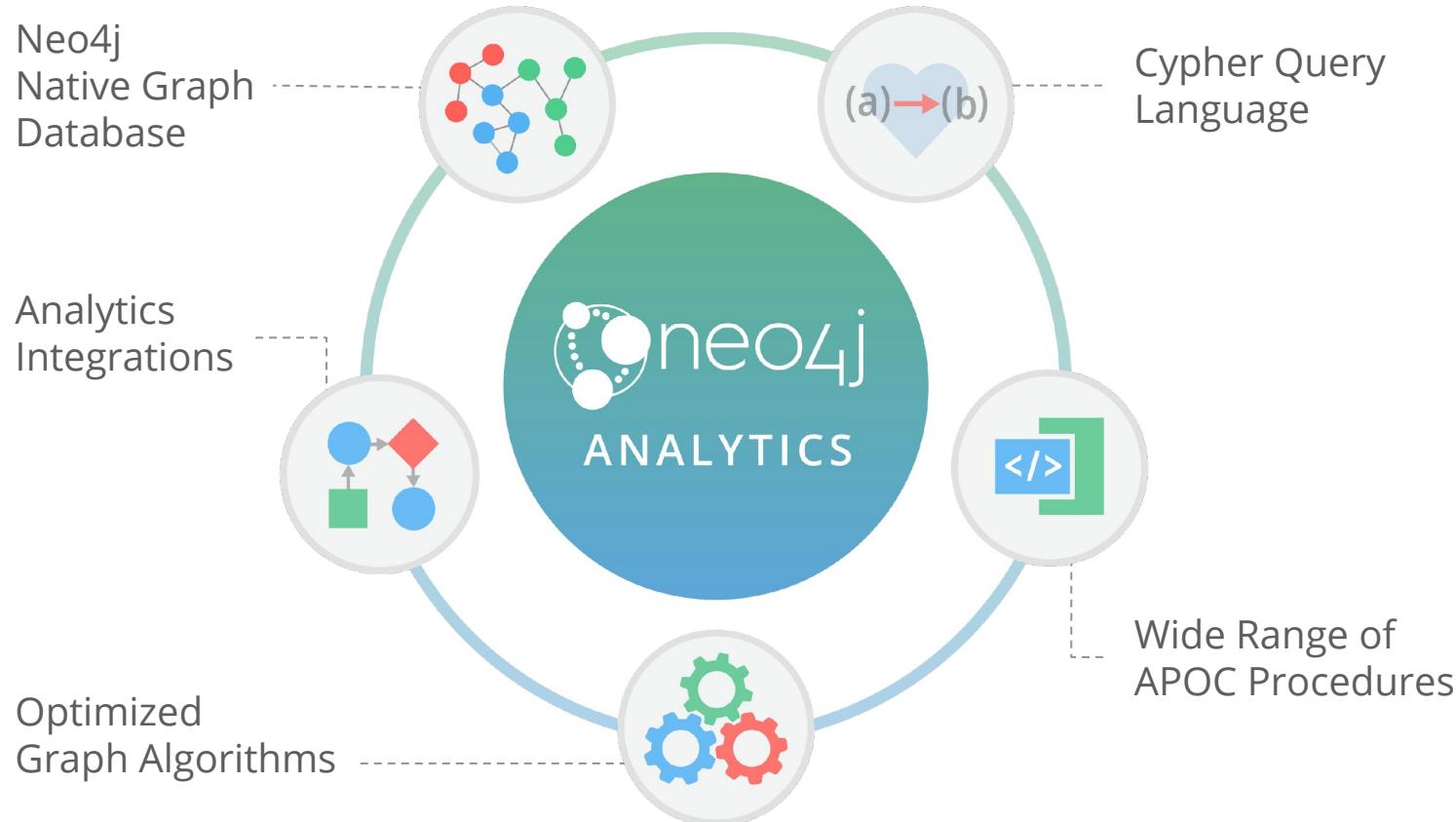


## Example Workflow Pipeline

# Our Goal



Example Workflow Pipeline



Determines the importance of distinct nodes in the network



Finds the optimal path or evaluates route availability and quality

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

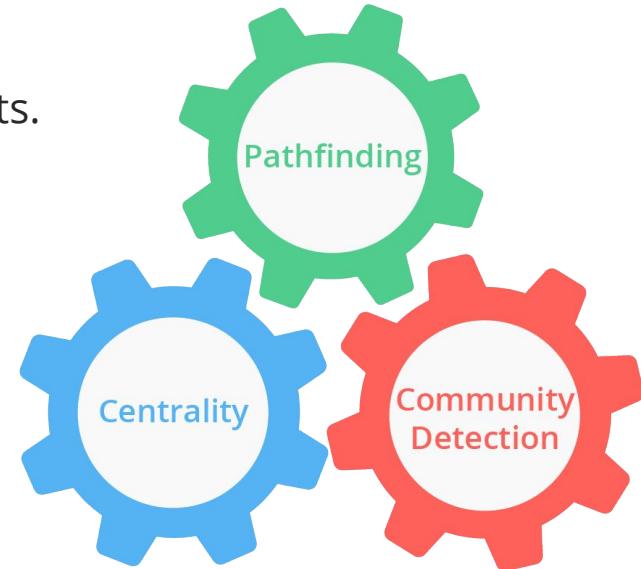
```
CALL algo.<name>('Label', 'TYPE', {conf})
```



# Cypher Projection

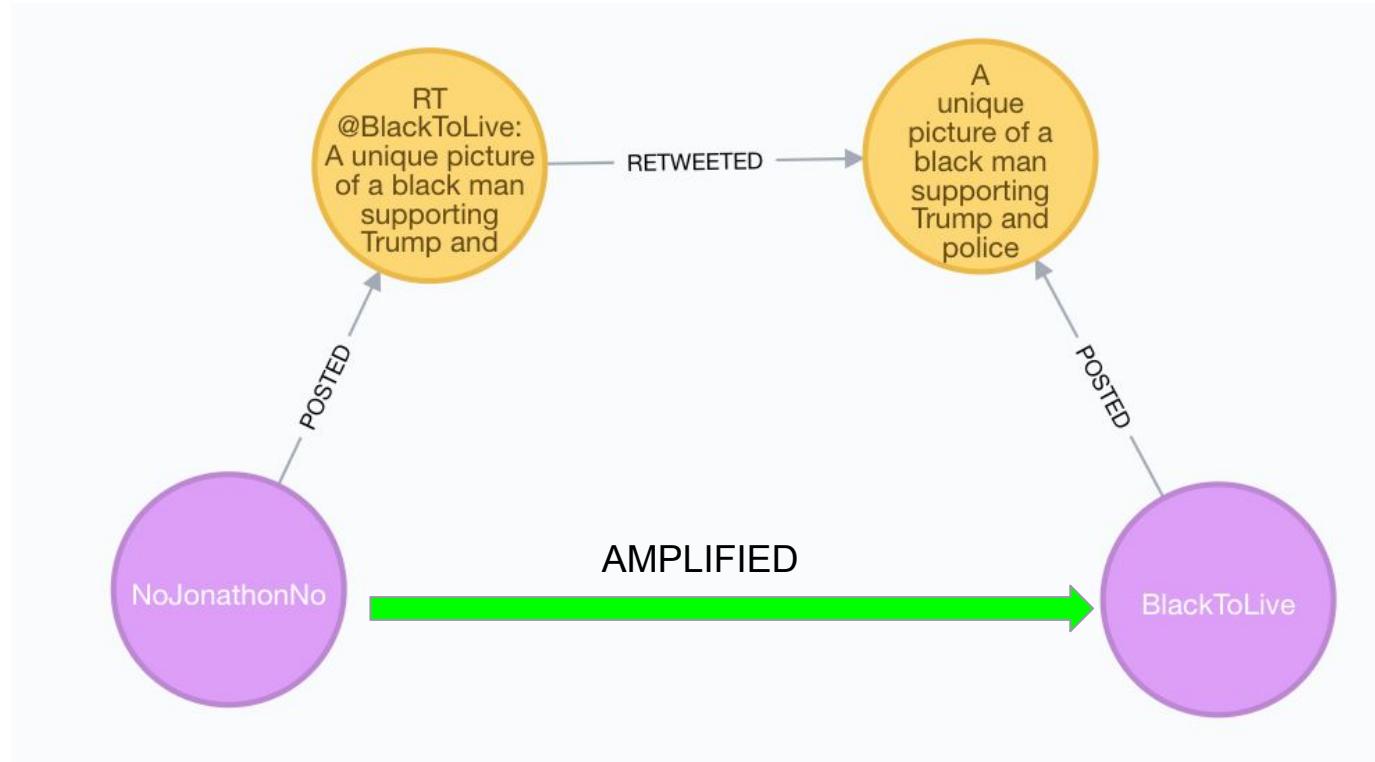
Pass in Cypher statement for node- and relationship-lists.

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



# Inferred Relationships

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

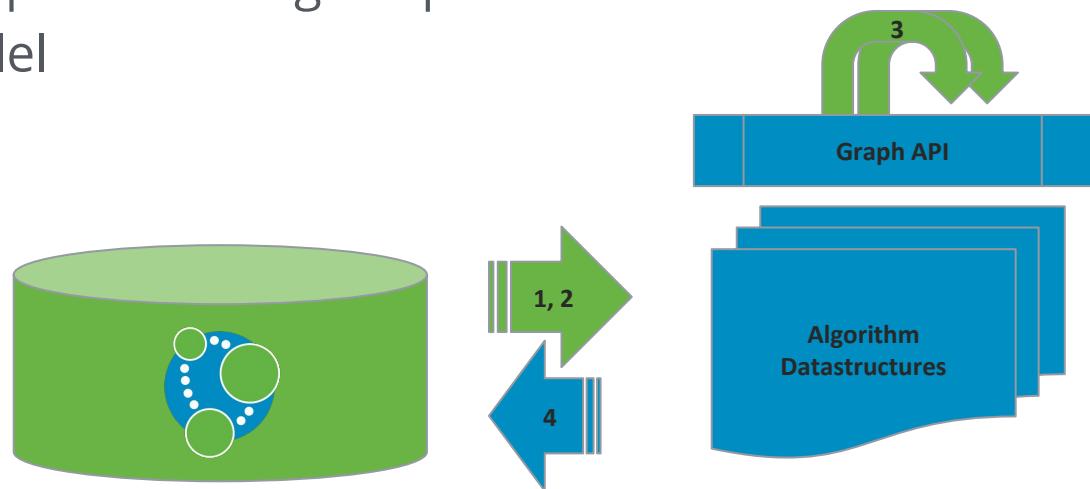


# PageRank on Inferred AMPLIFIED Graph

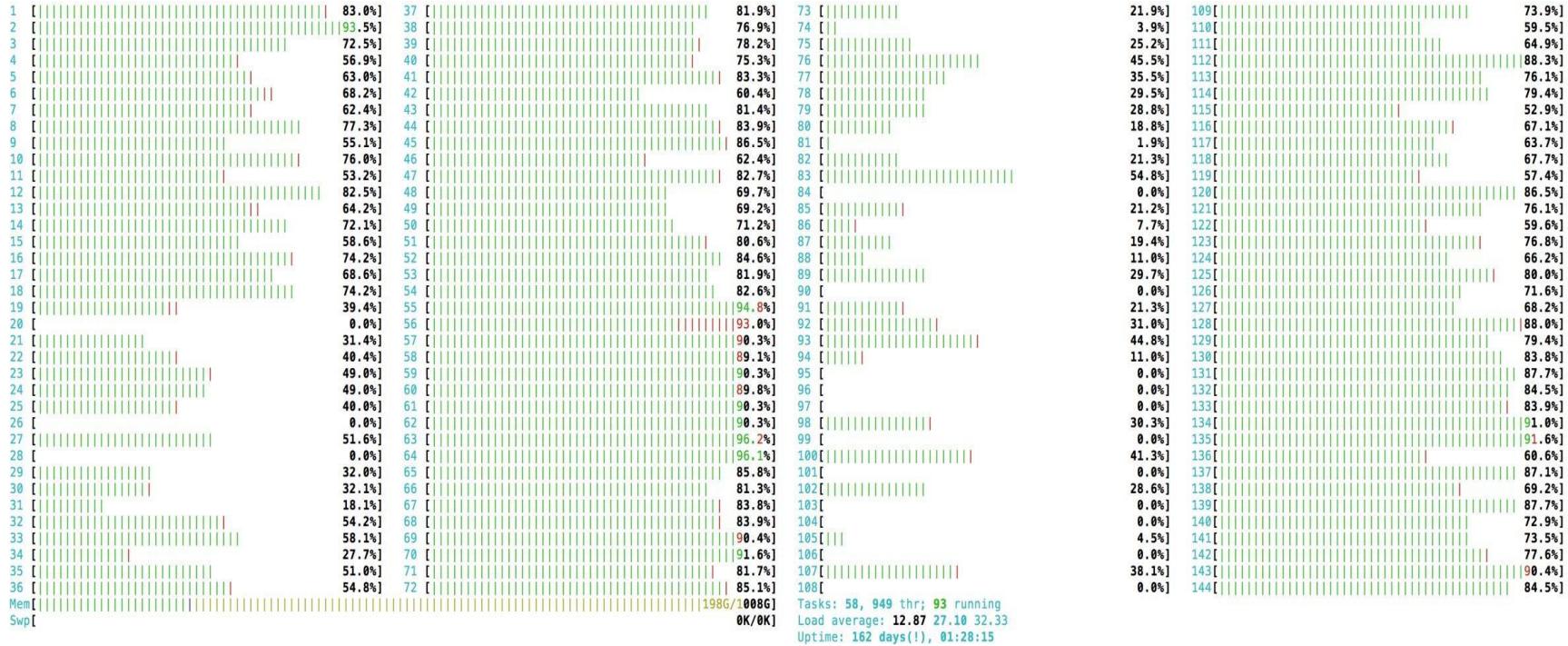
```
CALL algo.pageRank("MATCH  
  (r1:Troll)-[:POSTED]->(:Tweet)<-[:RETWEETED]-(:Tweet)<-[:POSTED]-(r2:Troll)  
  RETURN id(r2) as source, id(r1) as target", {graph: 'cypher'})
```

# Architecture

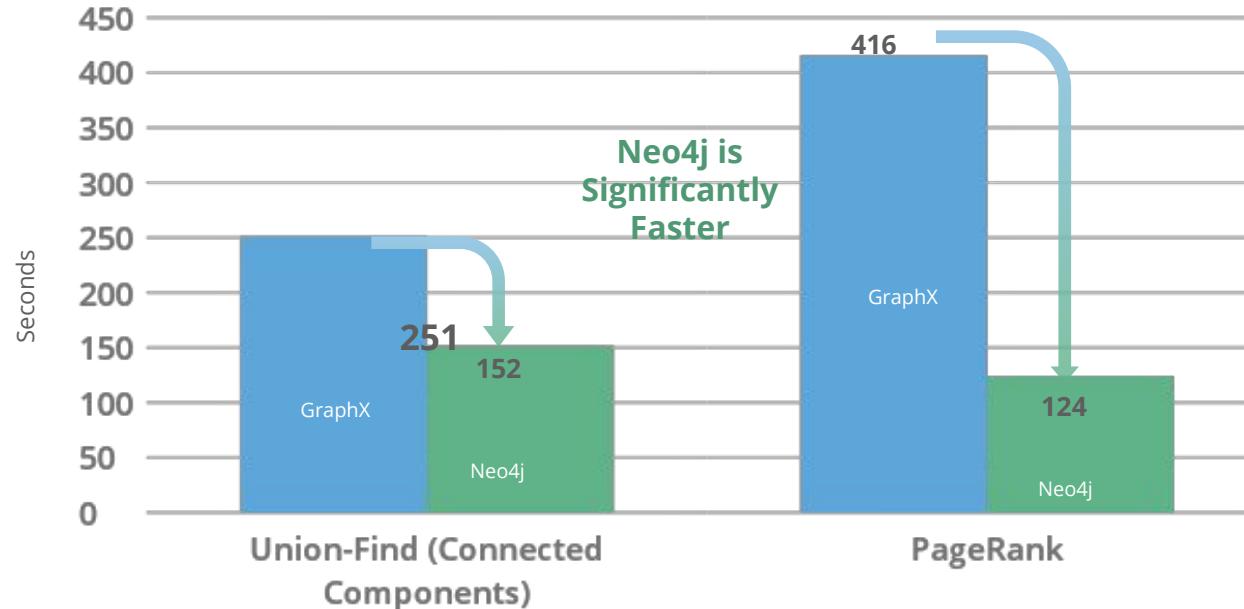
1. Load Data in parallel from Neo4j
2. Store in efficient data structures
3. Run Graph Algorithm in parallel using Graph API
4. Write data back in parallel



# Scale: 144 CPU



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



## 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, 900G RAM

```
call algo.pageRank('Account','SENT',
                   {graph:'huge',iterations:20,write:true,concurrency:20});
```

nodes	iterations	loadMillis	computeMillis	writeMillis
300000000	20	401404	6024994	47106

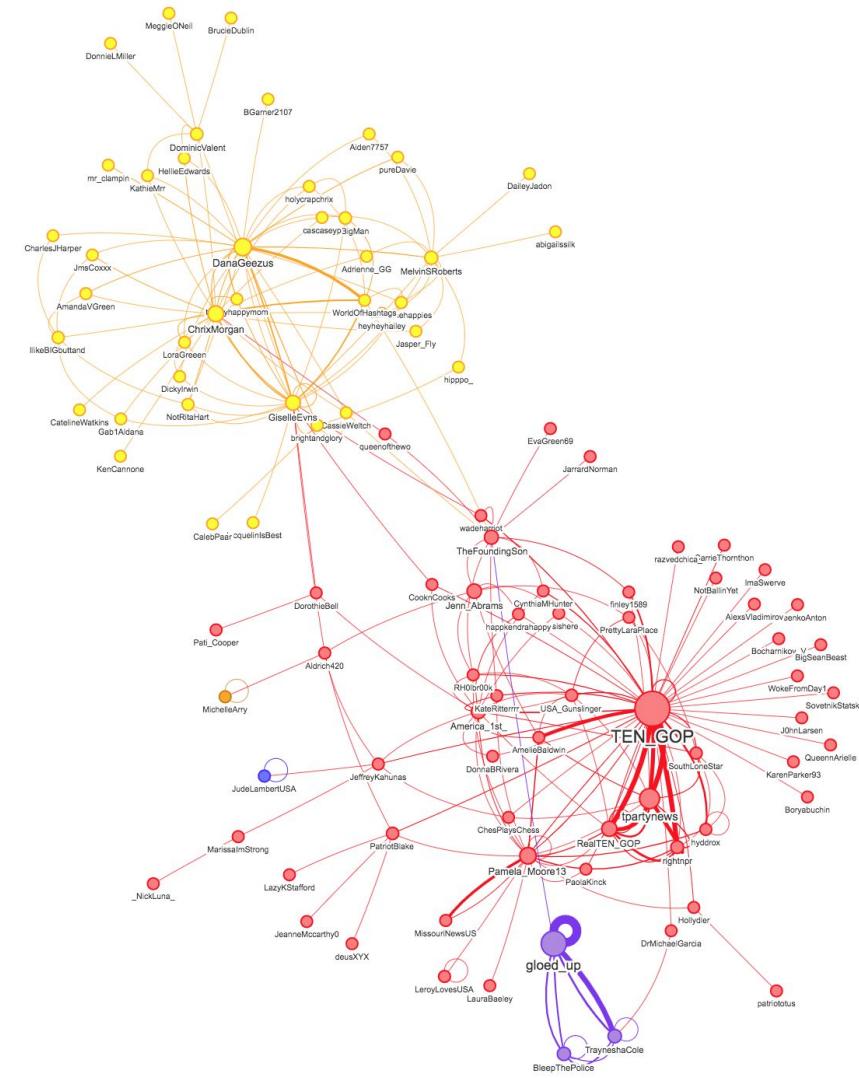
1 row 6473526 ms -> 1h 47min

# Graph Visualization

# Graph Visualization

Centrality & community detection  
AMPLIFIED relationships

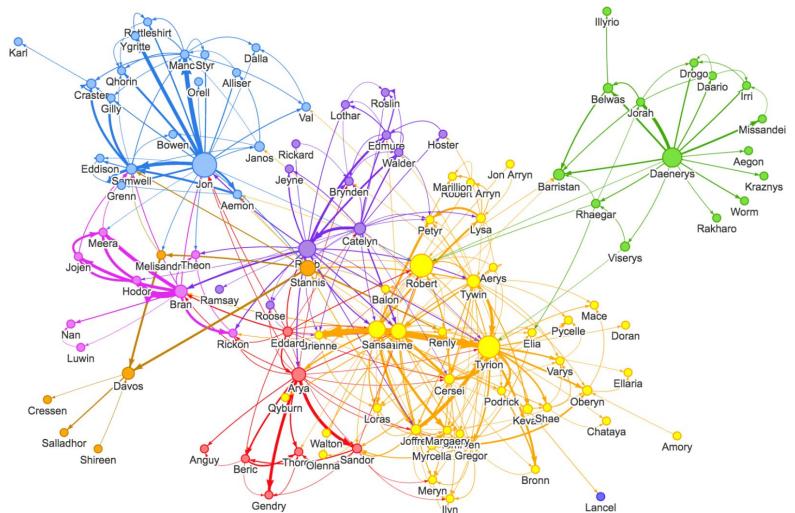
**Node size** → PageRank  
**Color** → community detection  
**Rel Thickness** → weight



# 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/johnymontana/neovis.js>

# Feb 14:



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

v.

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  
PODKOPEV,  
GLEB IGOREVICH VASILCHENKO,  
IRINA VIKTOROVNA KAVERZINA,  
and  
VLADIMIR VENKOV.

\* CRIMINAL NO.  
\* (18 U.S.C. §§ 2, 371, 1349, 1028A)

<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](http://lyonwj.com) for code, etc.
- <https://www.nbcnews.com/tech/social-media/russian-trolls-went-attack-during-key-election-moments-n827176>

# neo4jsandbox.com

neo4j

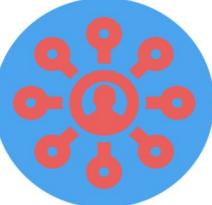
PRODUCTS SOLUTIONS PARTNERS CUSTOMERS LEARN DEVELOPERS  Search

**Launch a New Sandbox**

Each sandbox includes data, interactive guides with example queries, and sample code.

Jupyter Sandbox  Launch Sandbox	Parade Papers by ICIJ  The Parade Papers dataset and guide from the International Consortium of Investigative Journalists (ICIJ). Launch Sandbox
Neo4j 3.3  NEW Neo4j 3.3 release - Blank Sandbox Launch Sandbox	Panama Papers by ICIJ  The Panama Papers dataset and guide from the International Consortium of Investigative Journalists (ICIJ). Launch Sandbox
Recommendations  Generate personalized real-time recommendations using a dataset of movie reviews. Launch Sandbox	Network and IT Management  Dependency and root cause analysis + more for network and IT management. Launch Sandbox
Twitter  If signed into Neo4j Sandbox using Twitter, this Sandbox will allow you to Graph your Twitter network. Launch Sandbox	Legis-Graph  US Congress modeled as a Graph - bills, votes, members, and more. Launch Sandbox
Spreadsheets Grapher  Load data directly from Google Spreadsheets Launch Sandbox	Blank Sandbox  Blank Sandbox. Load your own data with LOAD CSV or create data from scratch. Launch Sandbox
Trumpworld  Explore connections in and around the Trump Administration using this dataset from BuzzFeed. Launch Sandbox	GraphConnect Schedule  GraphConnect Europe 2017 schedule graph Launch Sandbox

Russian Twitter Trolls    Get Started    **Details**    Data Model    Code    Advanced ▾ ↻



**Neo4j Browser:** <https://10-0-1-194-33031.neo4jsandbox.com/>

**Direct Neo4j HTTP:** <http://54.237.227.207:33031/browser/>

**Username:** neo4j

**Password:** recognition-bins-procurement 

**IP Address:** 54.237.227.207

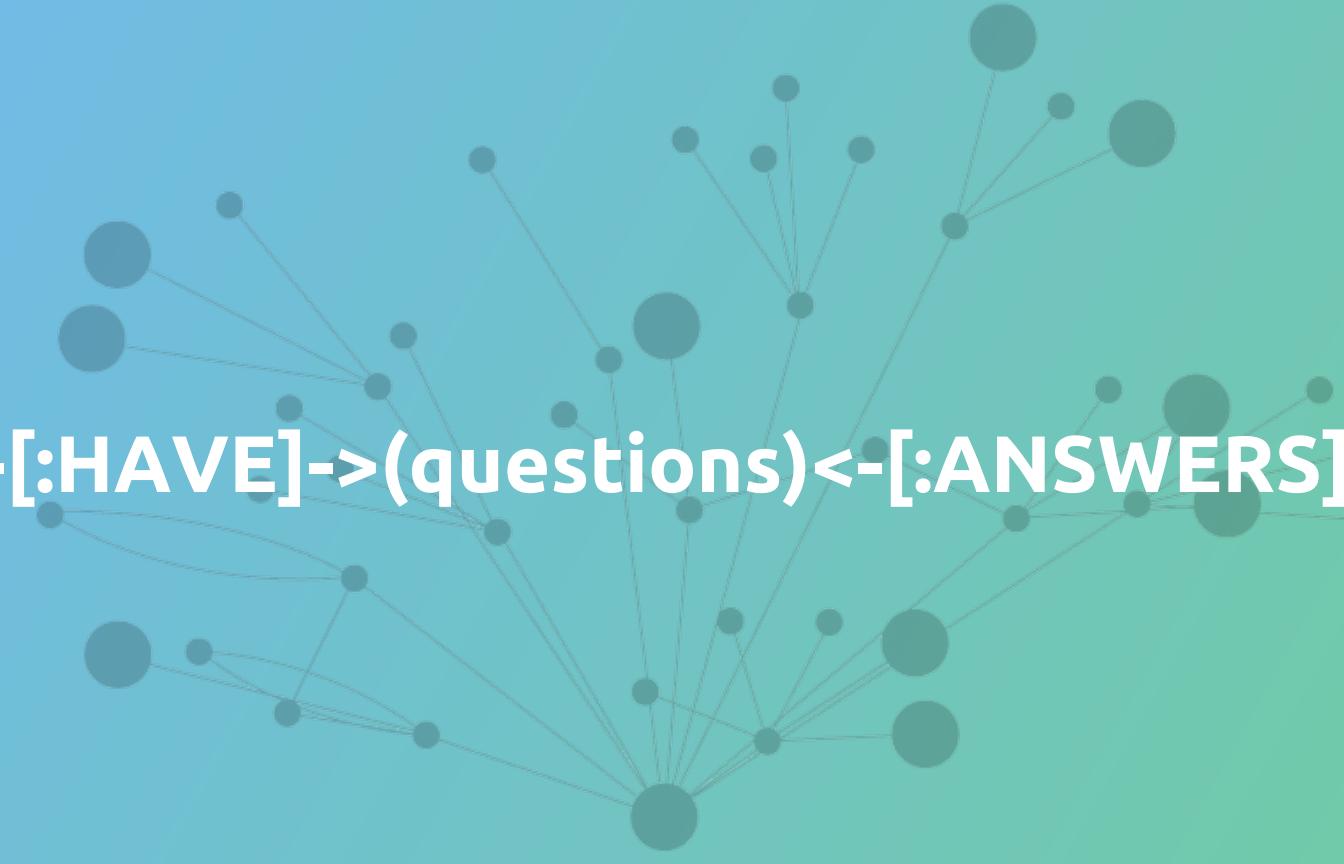
**HTTP Port:** 33031

**Bolt Port:** 33030

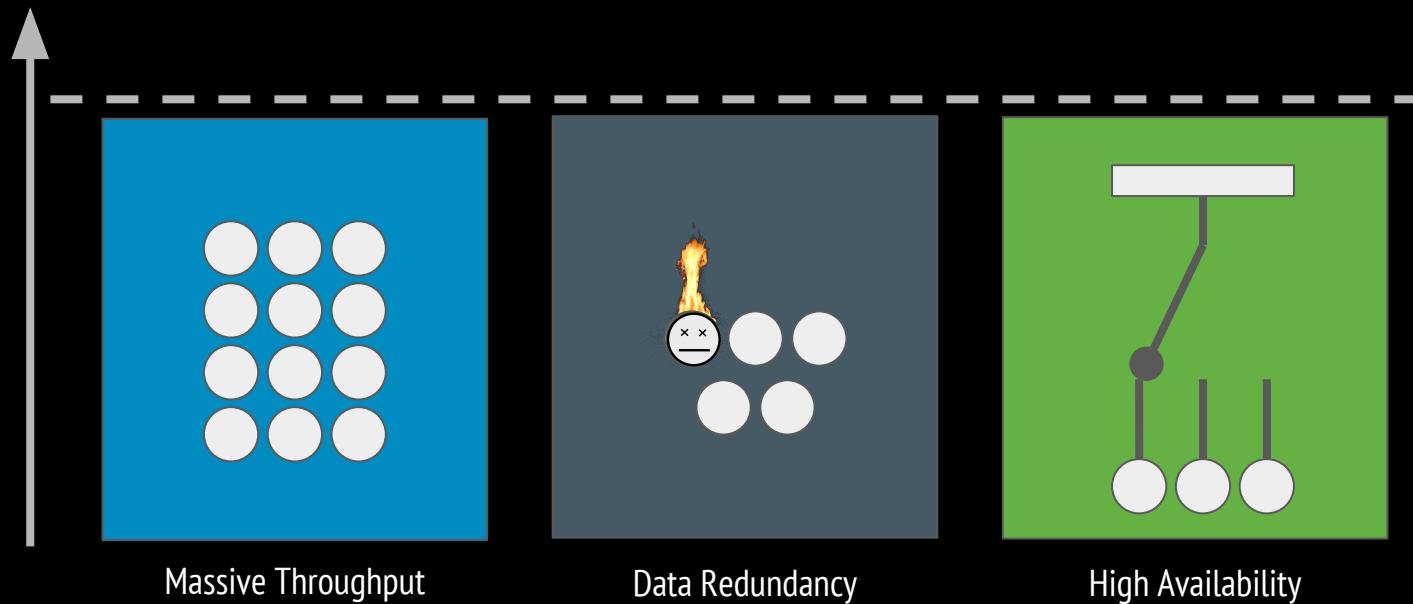
**Expires:** 8 days, 19 hours, 42 minutes

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

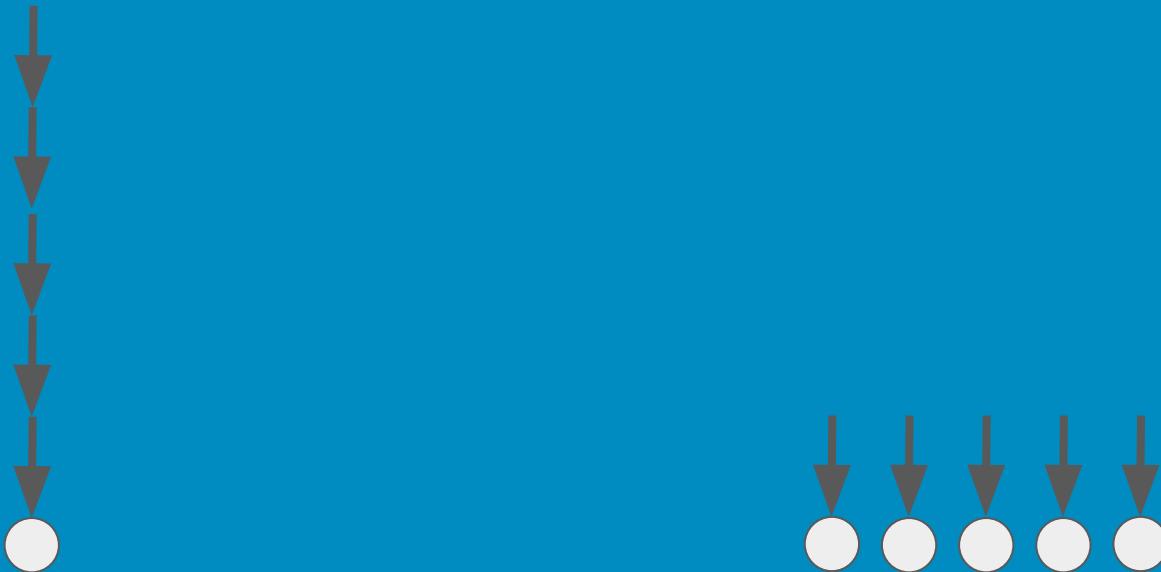
(you)-[:HAVE]->(questions)<-[ :ANSWERS ]-(will)



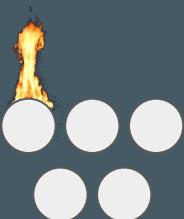
# Neo4j Causal Clustering



# Massive Throughput



# Data Redundancy



# High Availability



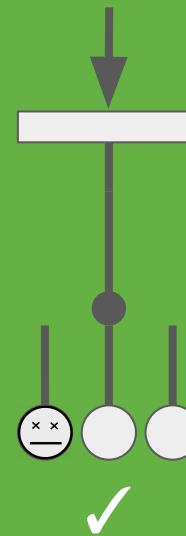
Error!  
503: Service Unavailable



# High Availability

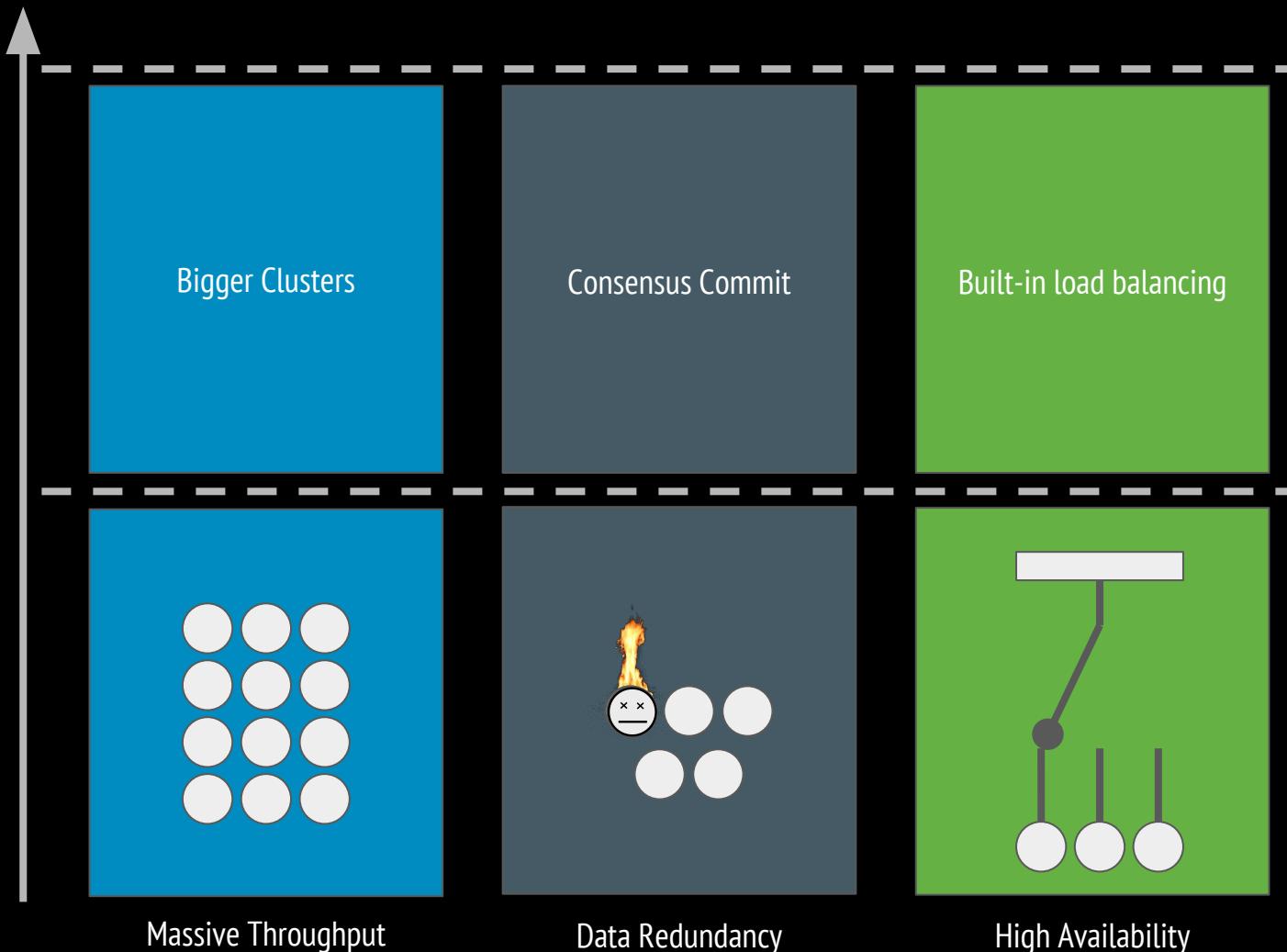


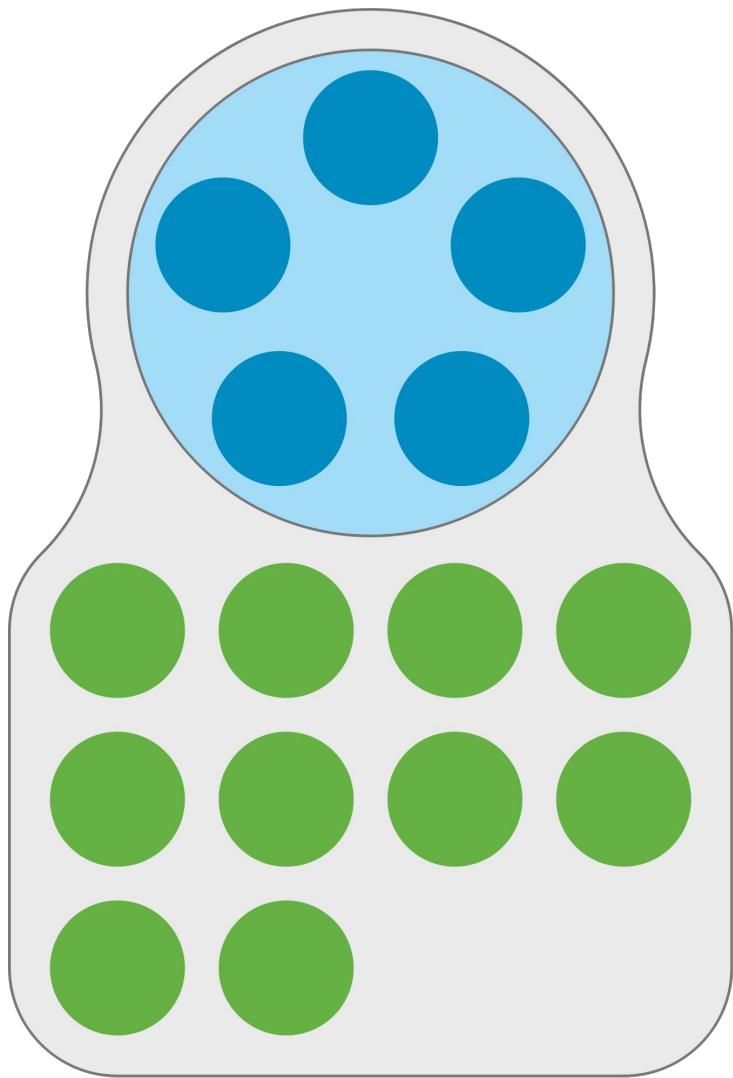
Error!  
503: Service Unavailable



 neo4j  
**3.1**  
Causal  
Clustering

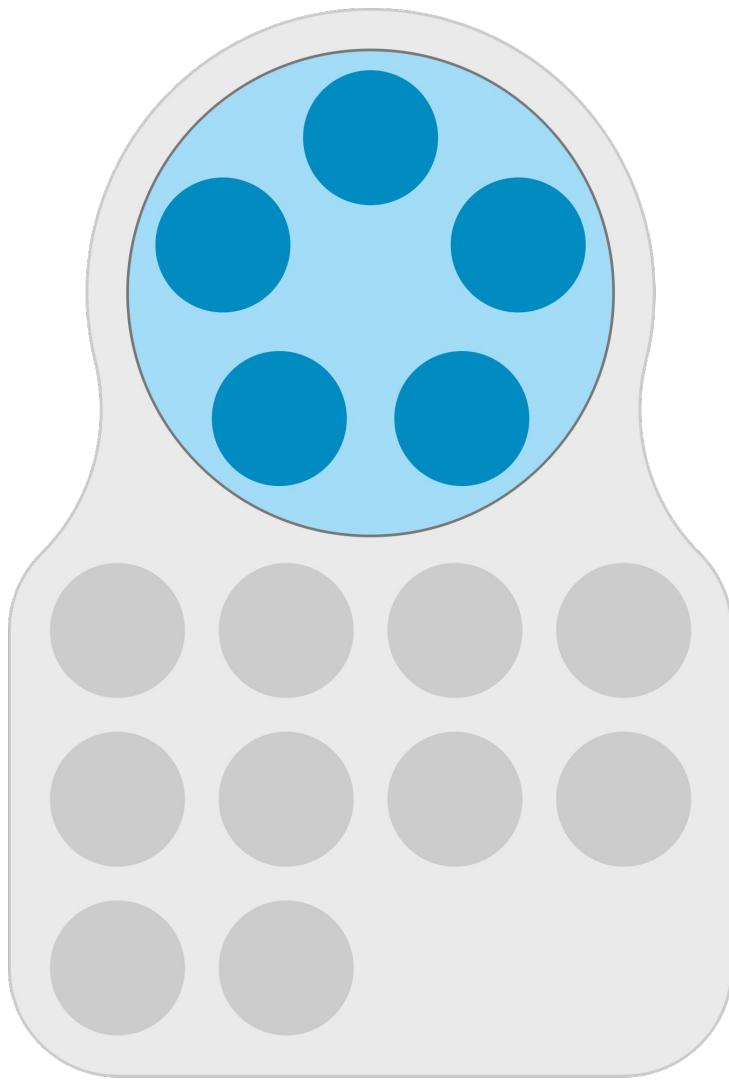
 neo4j  
**3.0**





Core

Read Replicas



# Core

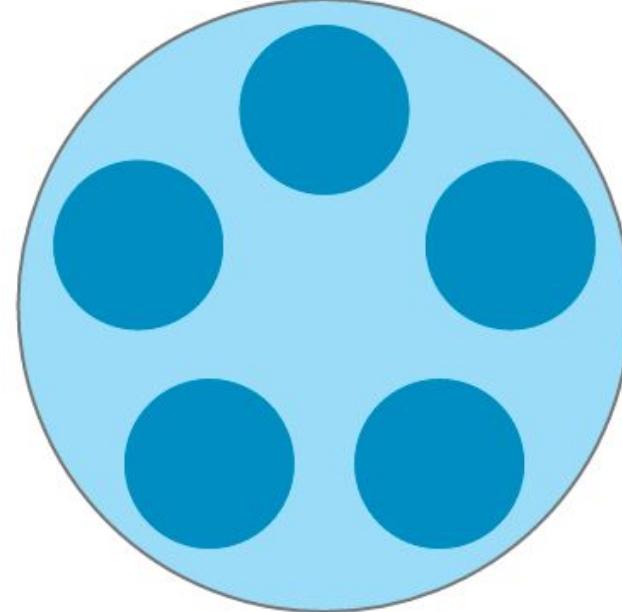
- Small group of Neo4j databases
- Fault-tolerant Consensus Commit
- Responsible for data safety

# Writing to the Core Cluster

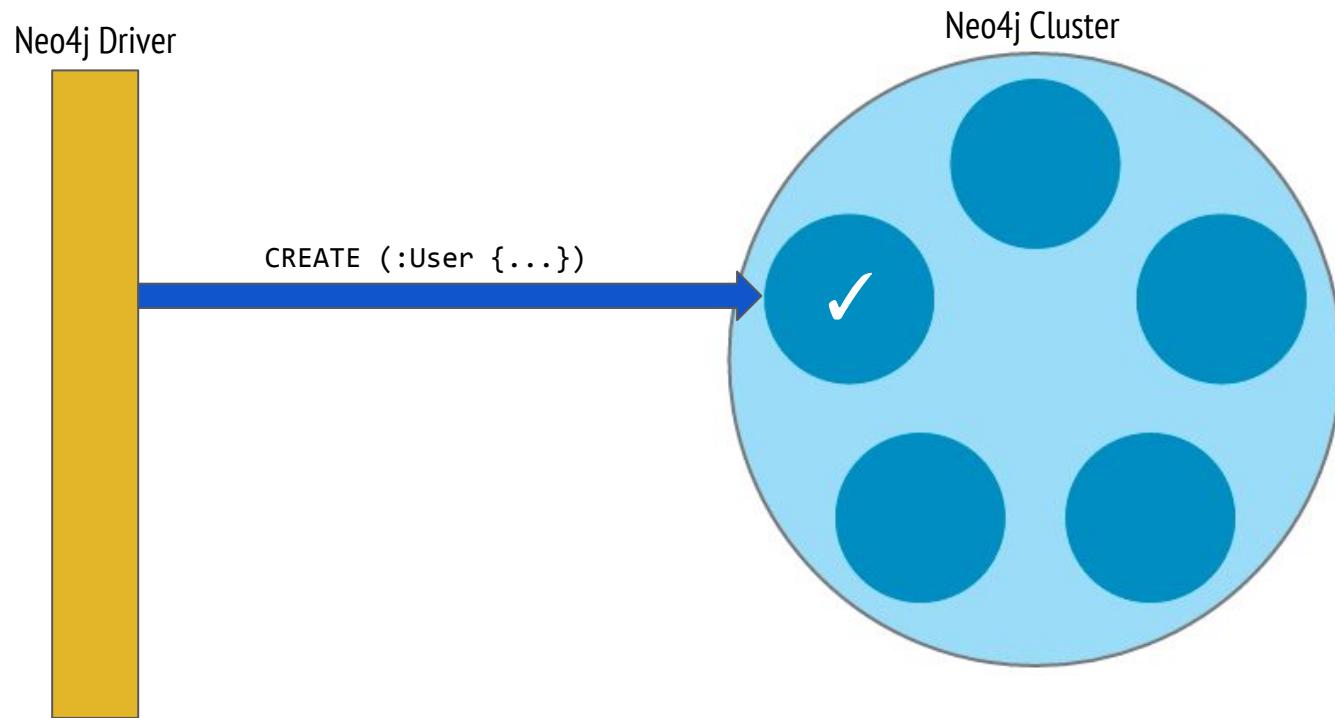
Neo4j Driver



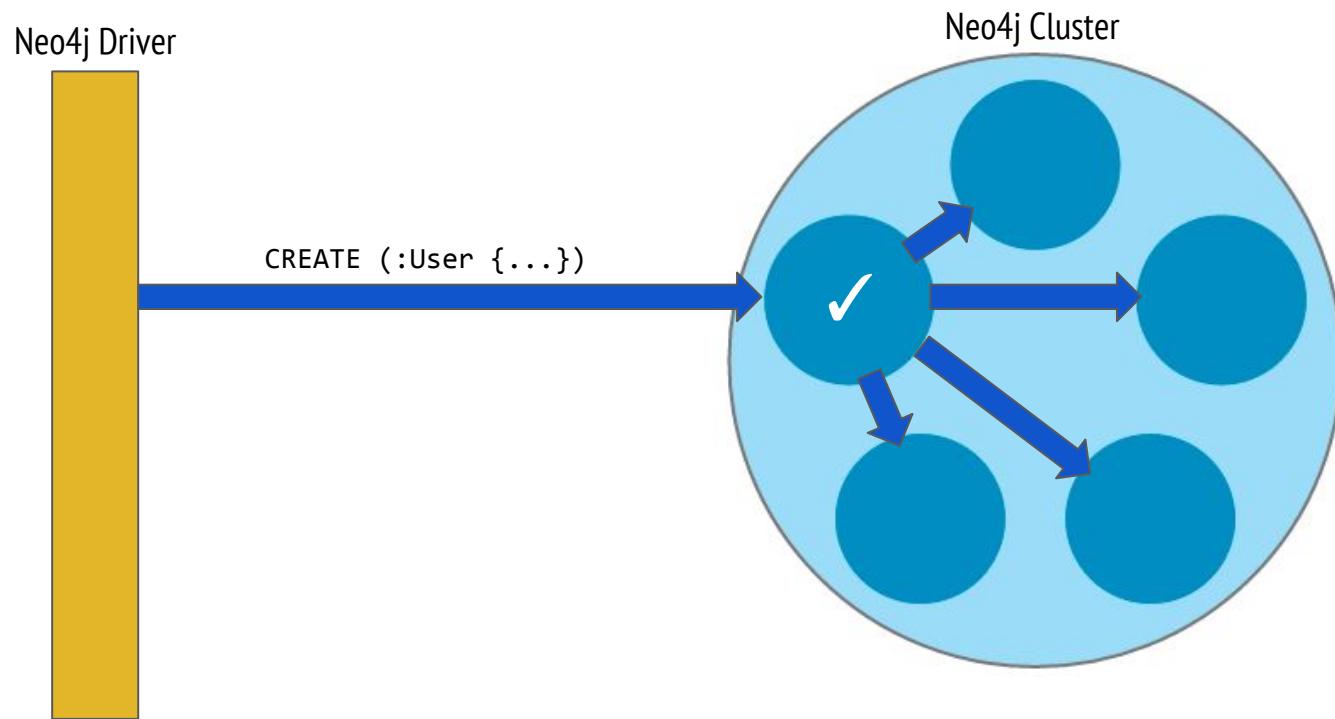
Neo4j Cluster



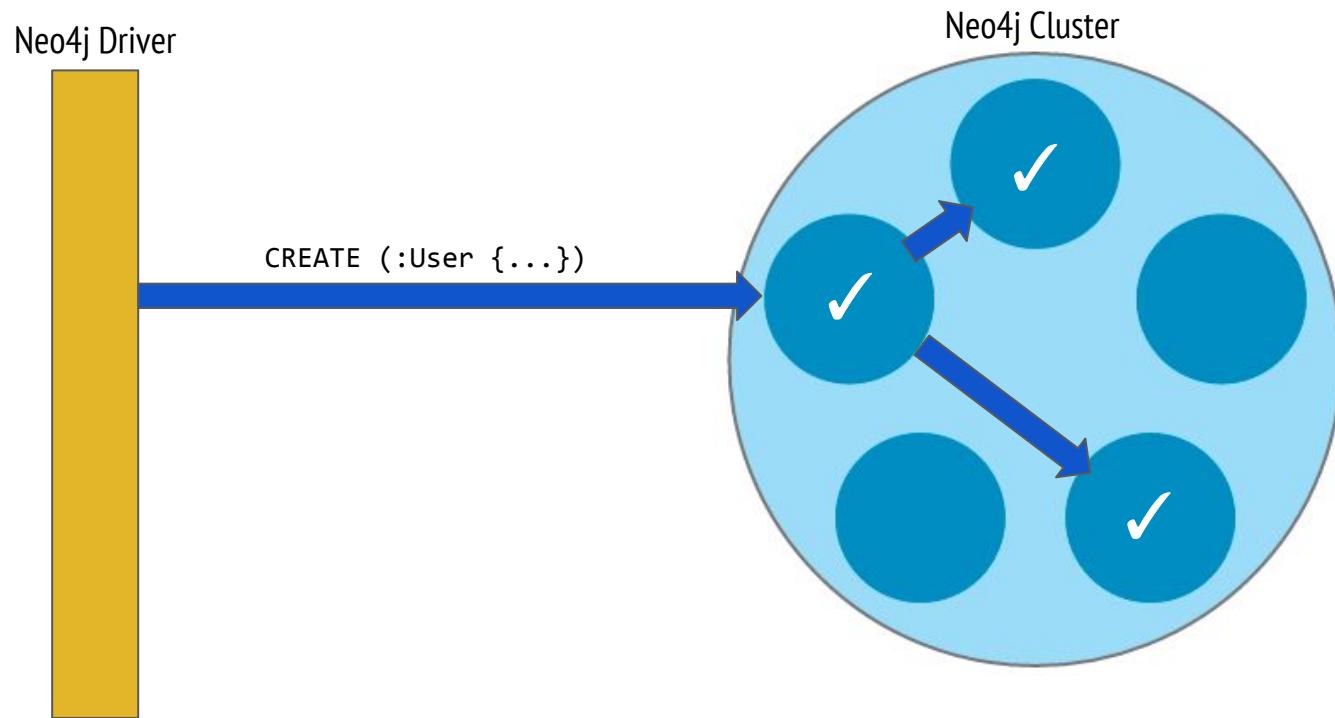
# Writing to the Core Cluster



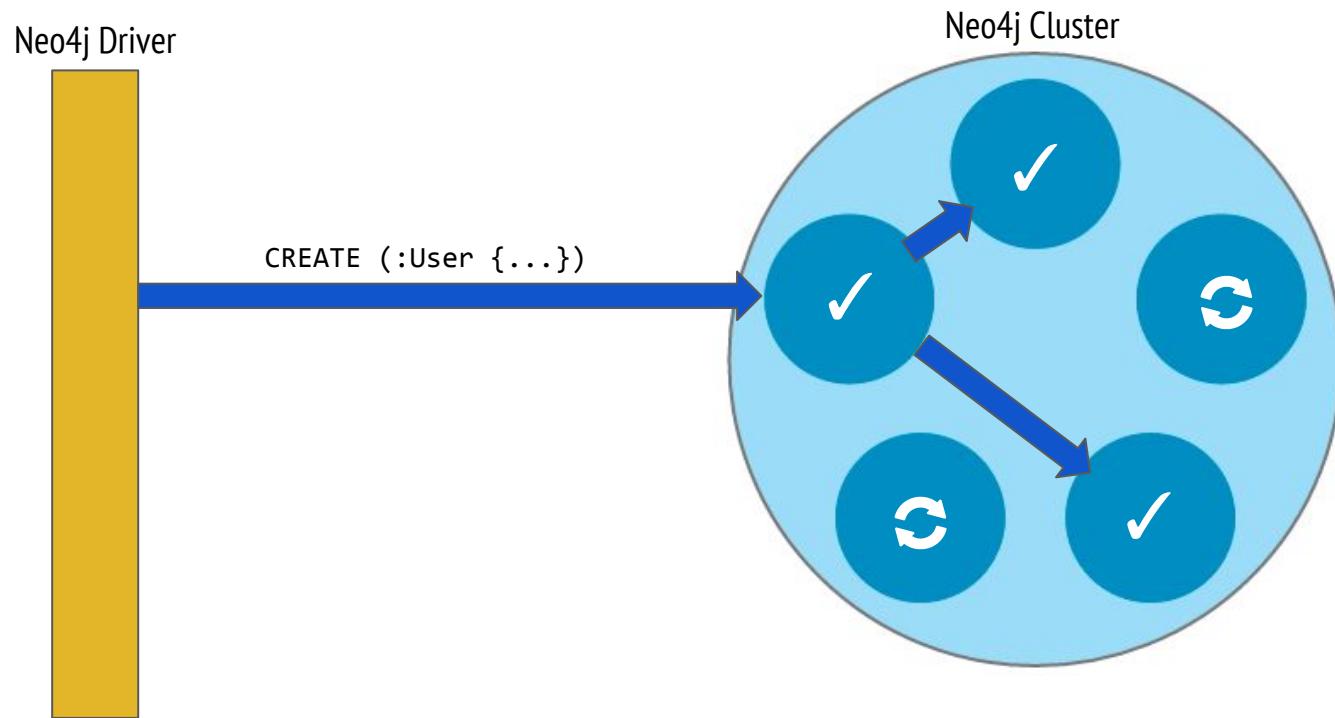
# Writing to the Core Cluster



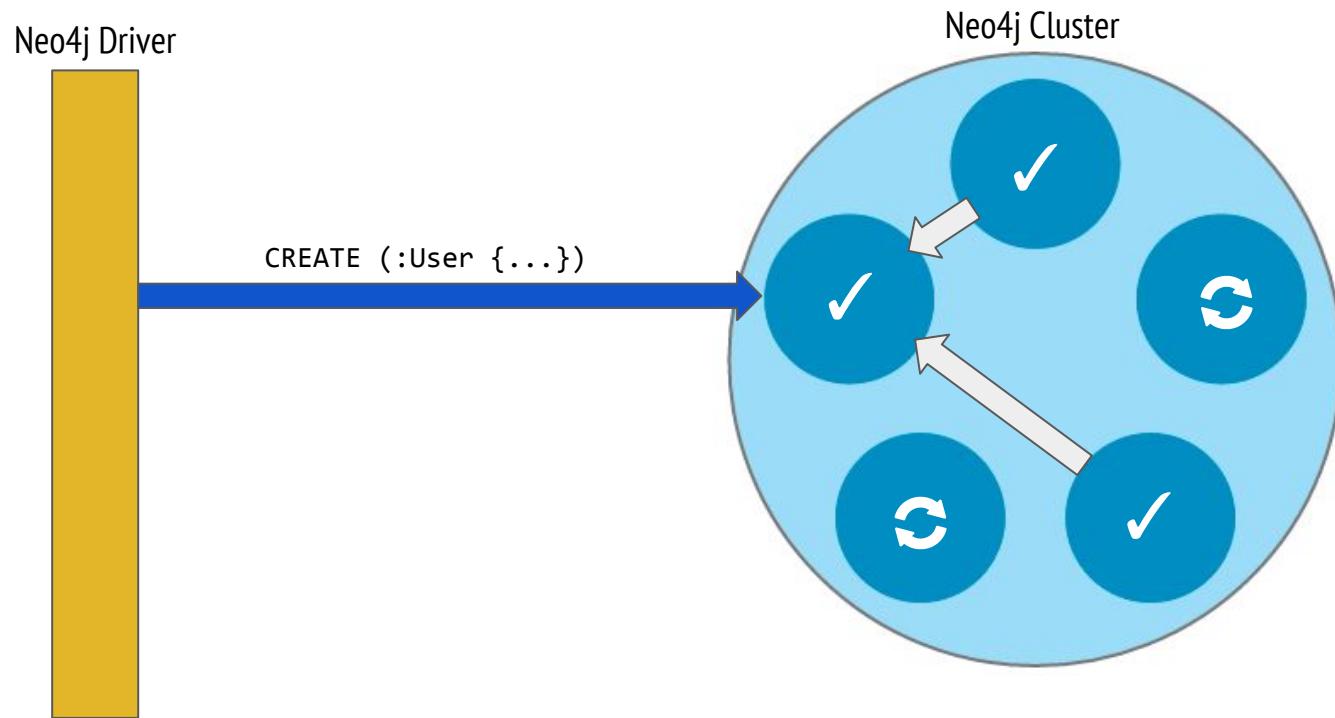
# Writing to the Core Cluster



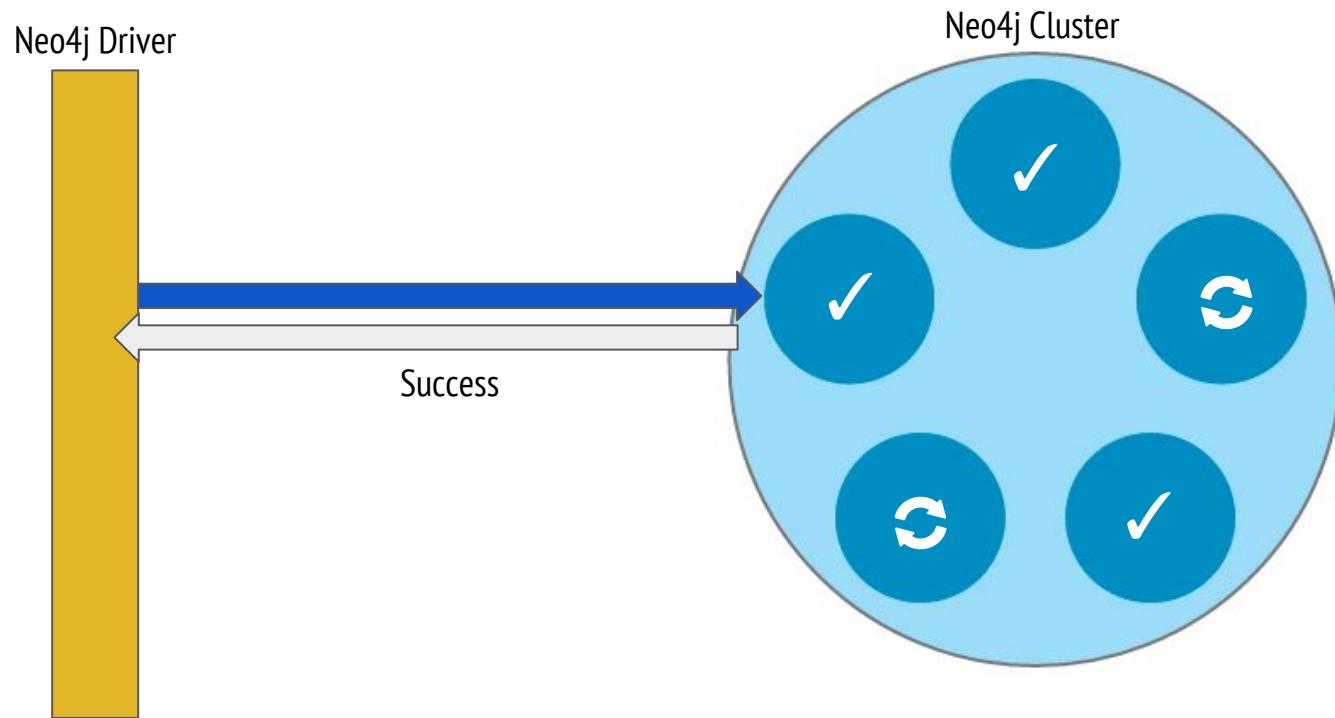
# Writing to the Core Cluster



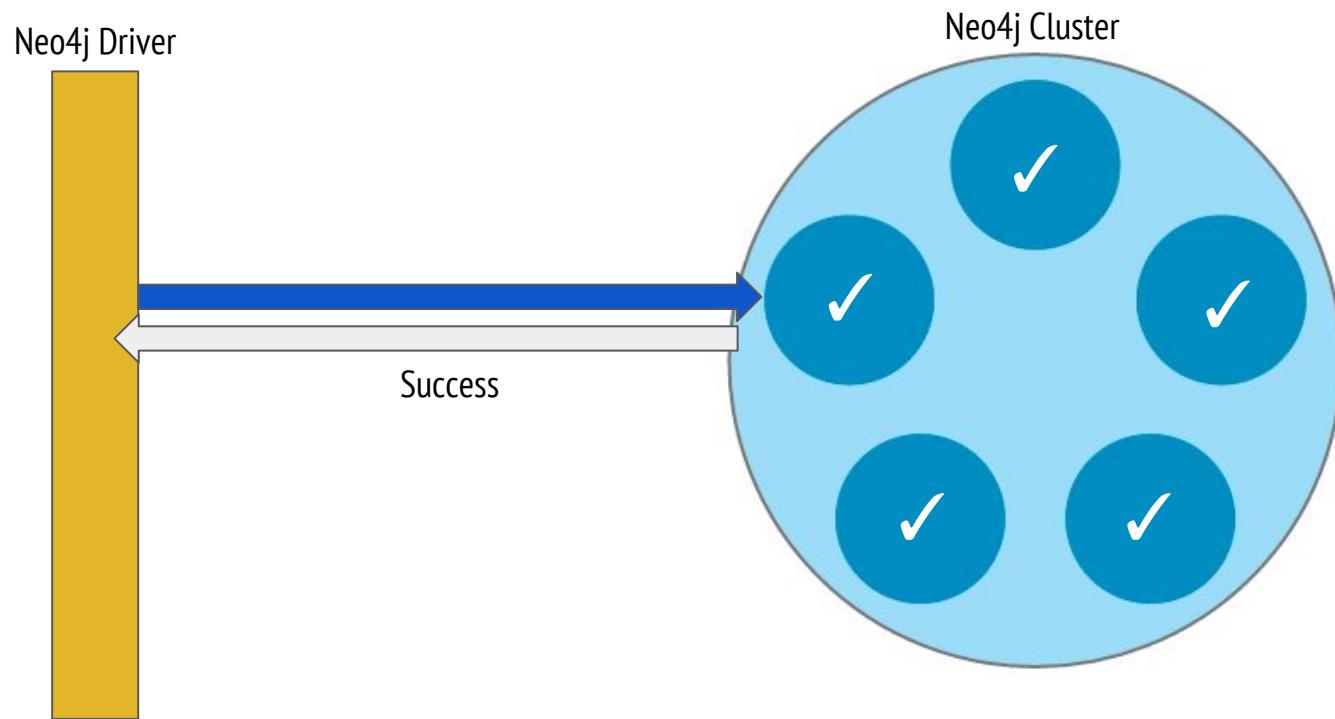
# Writing to the Core Cluster



# Writing to the Core Cluster



# Writing to the Core Cluster



# Raft Protocol

# In Search of an Understandable Consensus Algorithm (Extended Version)

Diego Ongaro and John Ousterhout  
Stanford University

## Abstract

Raft is a consensus algorithm for managing a replicated log. It produces a result equivalent to (multi-)Paxos, and it is as efficient as Paxos, but its structure is different from Paxos; this makes Raft more understandable than Paxos and also provides a better foundation for building practical systems. In order to enhance understandability, Raft separates the key elements of consensus, such as leader election, log replication, and safety, and it enforces a stronger degree of coherency to reduce the number of states that must be considered. Results from a user study demonstrate that Raft is easier for students to learn than Paxos. Raft also includes a new mechanism for changing the cluster membership, which uses overlapping majorities to guarantee safety.

## 1 Introduction

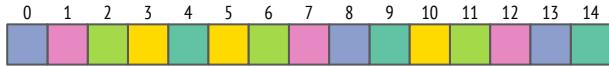
state space reduction (relative to Paxos, Raft reduces the degree of nondeterminism and the ways servers can be inconsistent with each other). A user study with 43 students at two universities shows that Raft is significantly easier to understand than Paxos: after learning both algorithms, 33 of these students were able to answer questions about Raft better than questions about Paxos.

Raft is similar in many ways to existing consensus algorithms (most notably, Oki and Liskov’s Viewstamped Replication [29, 22]), but it has several novel features:

- **Strong leader:** Raft uses a stronger form of leadership than other consensus algorithms. For example, log entries only flow from the leader to other servers. This simplifies the management of the replicated log and makes Raft easier to understand.
- **Leader election:** Raft uses randomized timers to

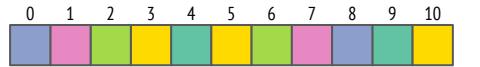
# Consensus Log → Committed Transactions → Updated Graph

Neo4j Raft implementation

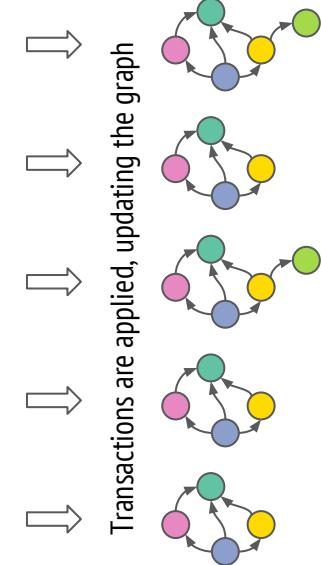


Consensus log: stores both committed and uncommitted transactions

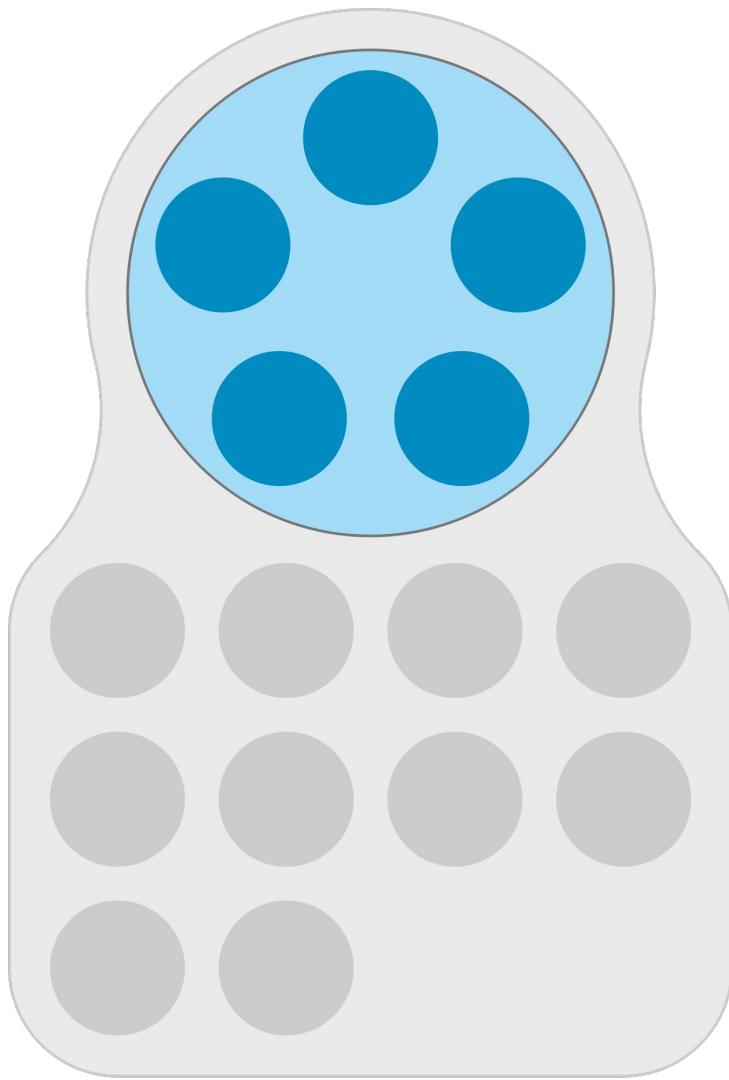
Transactions are only appended to the transaction log when committed according to Raft



Transaction log: the same transactions appear in the same order on all members

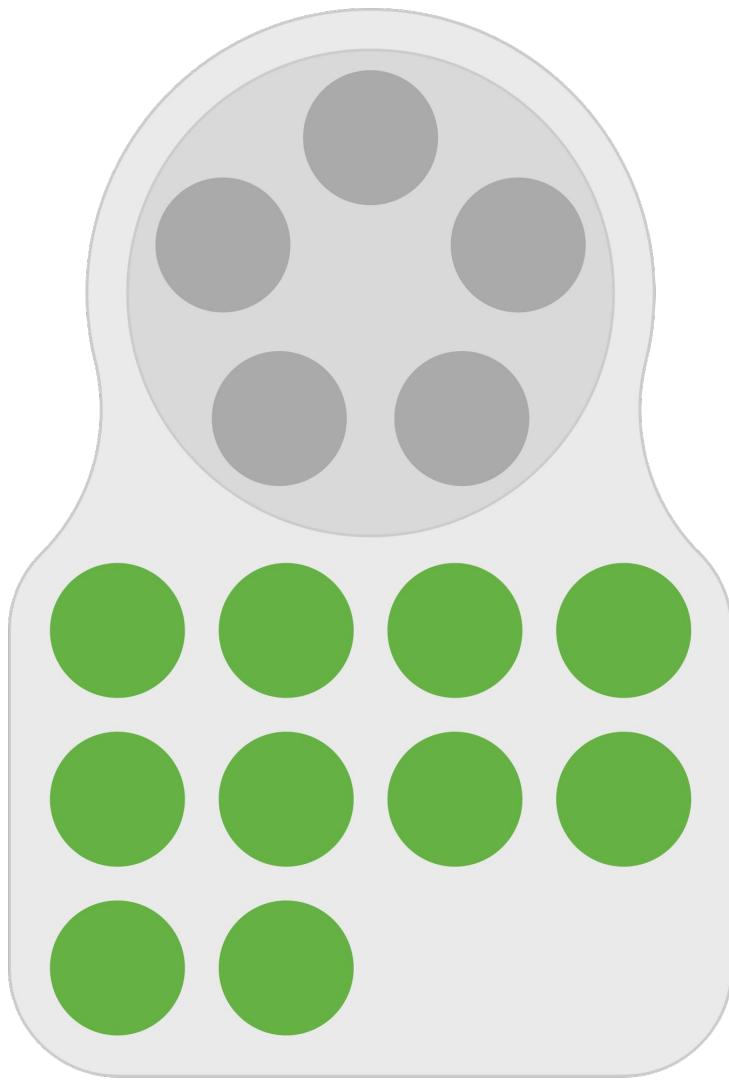


Transactions are applied, updating the graph



# Core

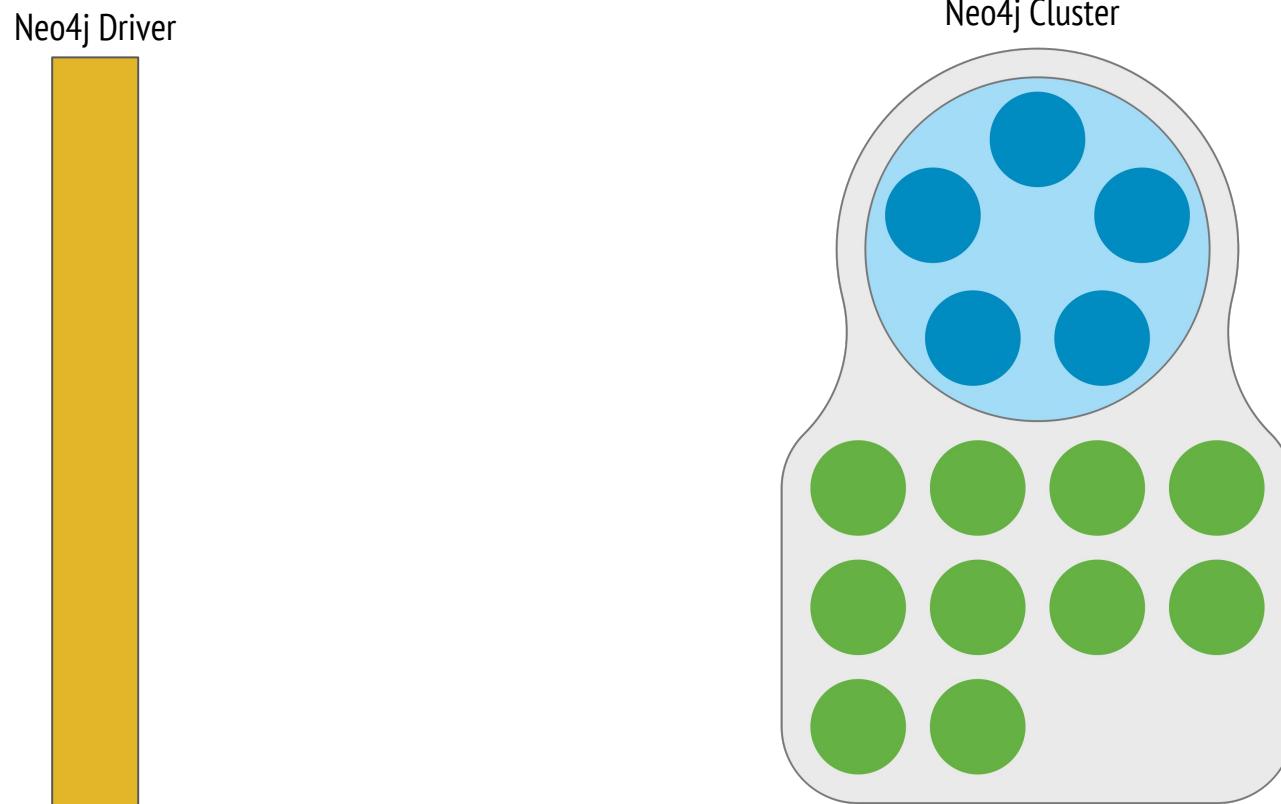
- Small group of Neo4j databases
- Fault-tolerant Consensus Commit
- Responsible for data safety



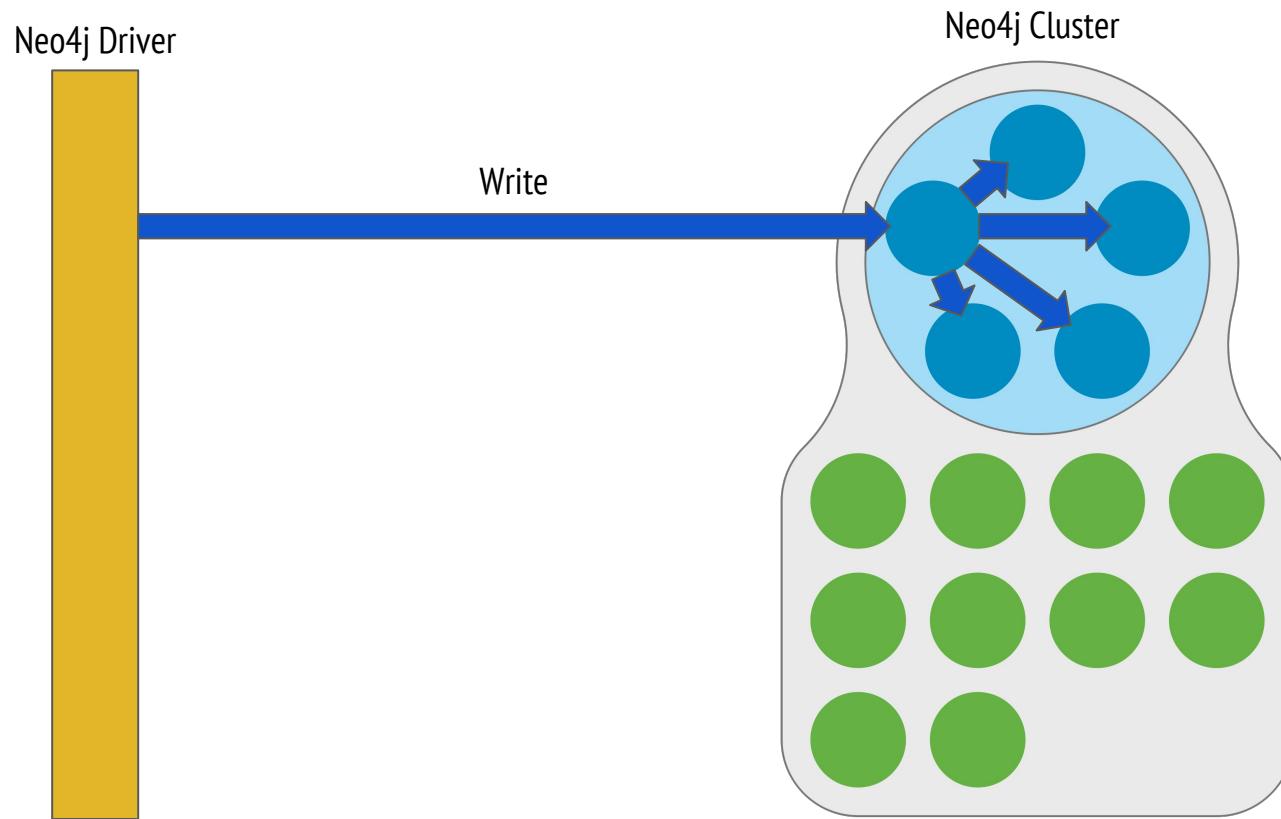
# Read Replicas

- For massive query throughput
- Read-only
- Not involved in Consensus Commit
- Disposable, suitable for auto-scaling

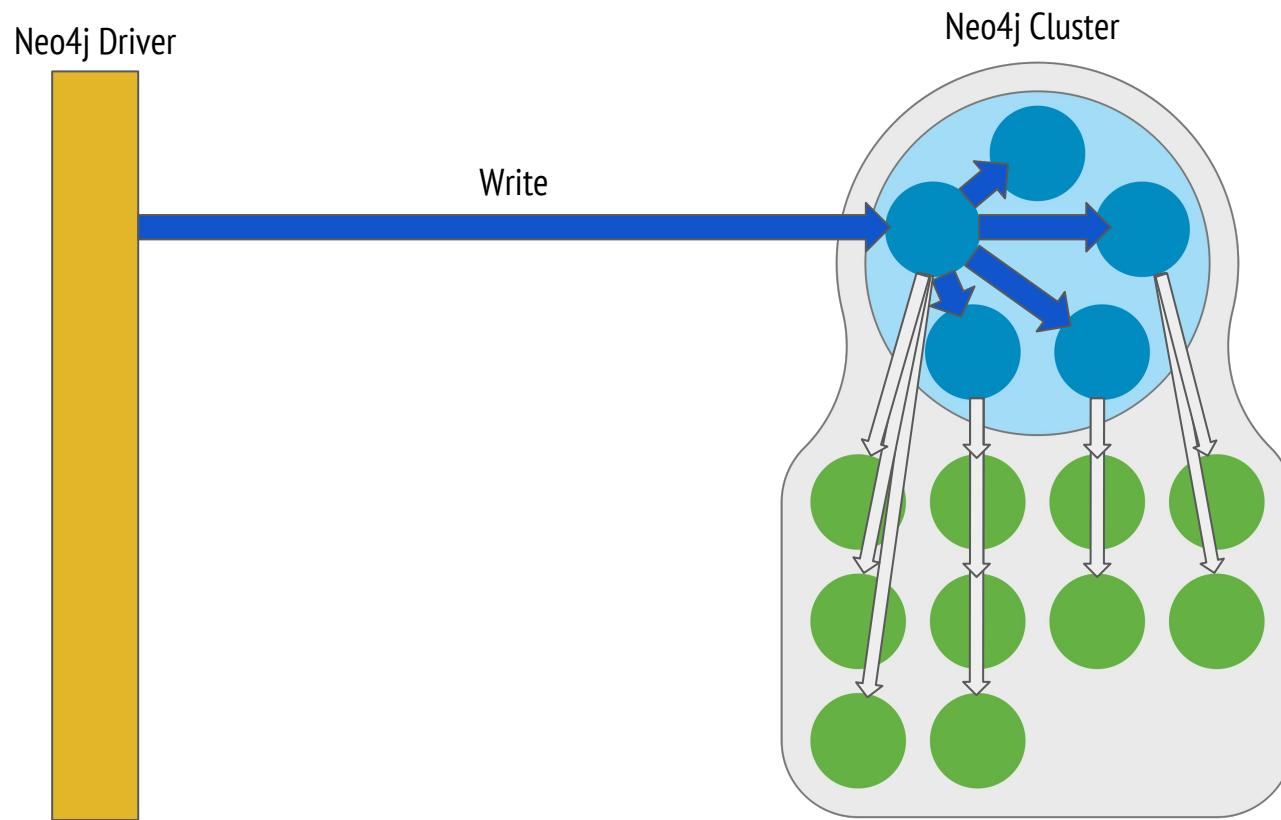
# Propagating updates to the Read Replicas



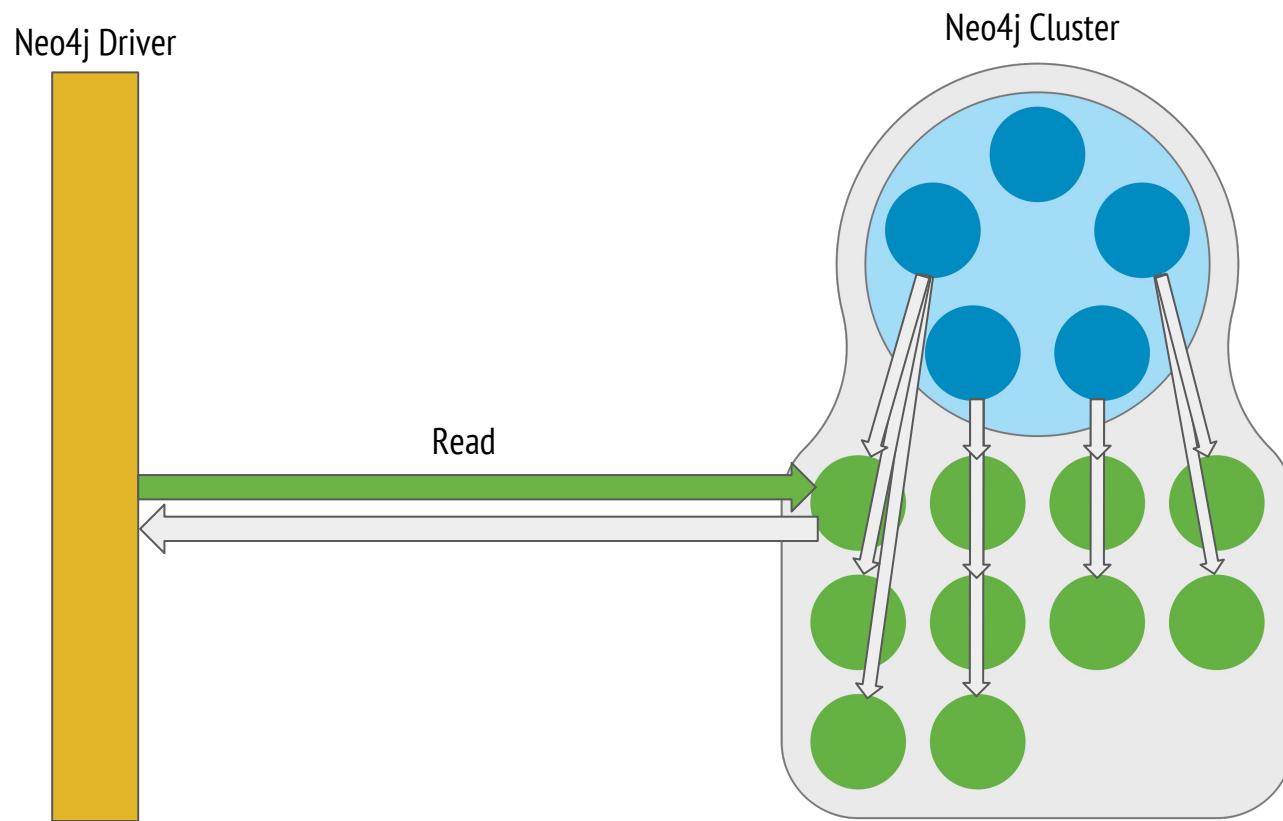
# Propagating updates to the Read Replicas

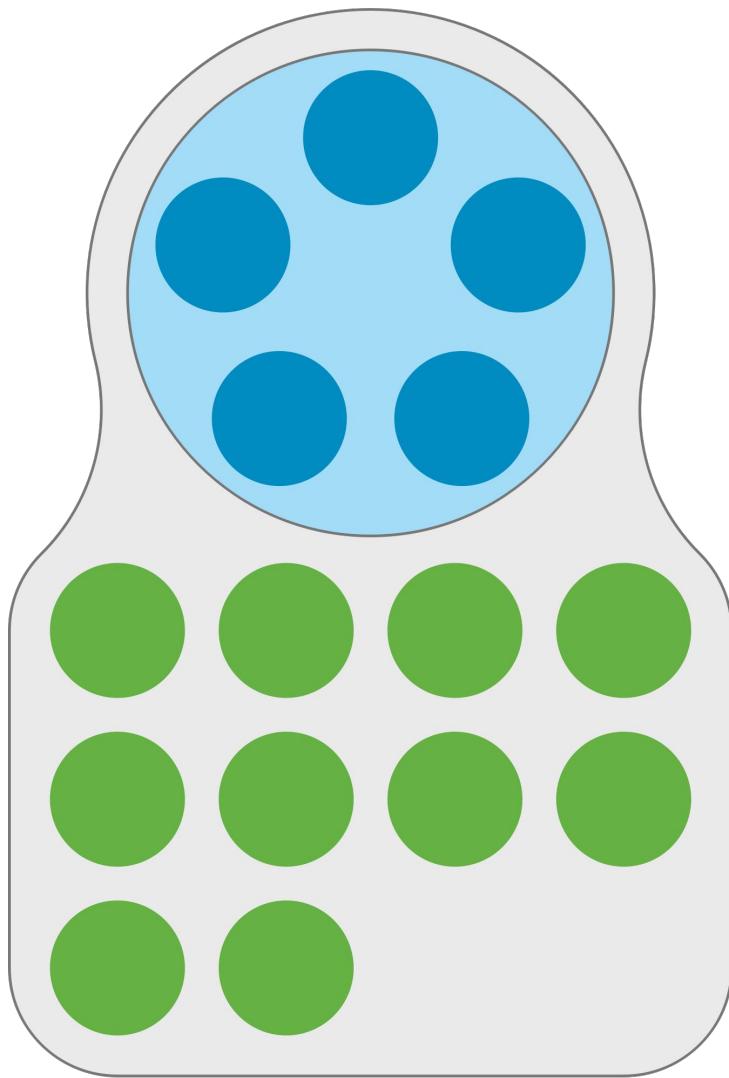


# Propagating updates to the Read Replicas



# Reading from the Edge





# Core

Updating the graph

# Read Replicas

Queries, analysis, reporting

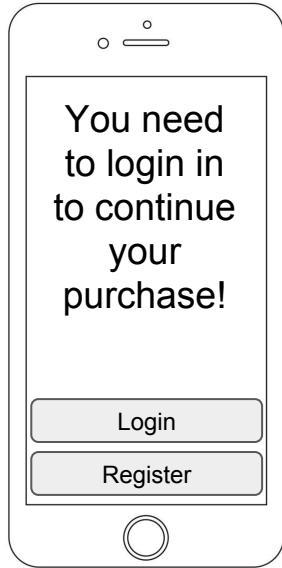
# bolt+routing://

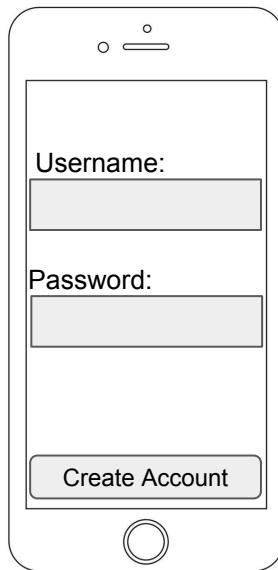
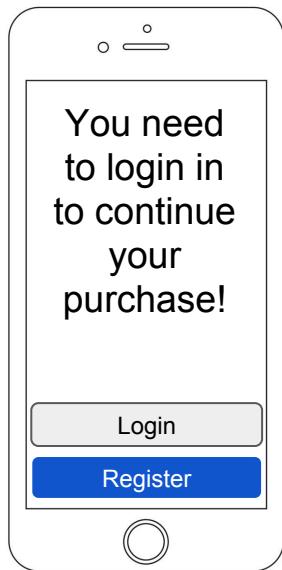
```
GraphDatabase.driver( "bolt+routing://aCoreServer" )
```

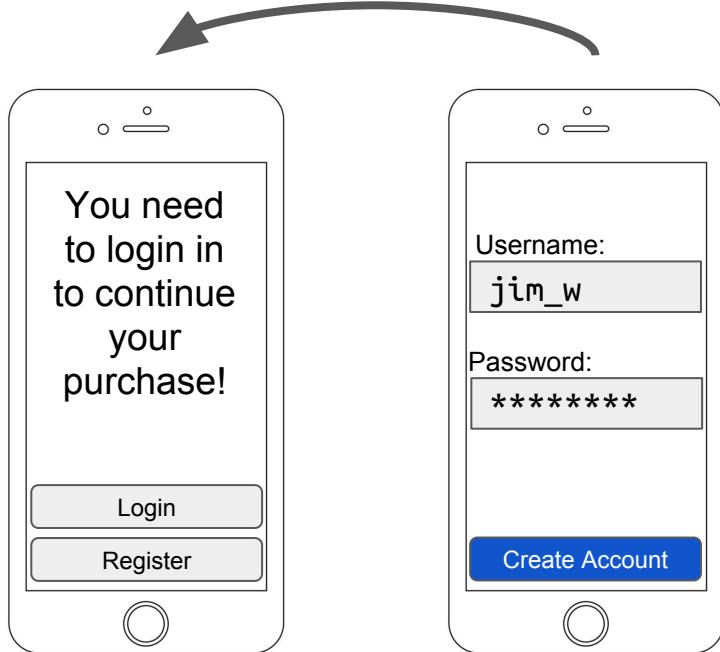


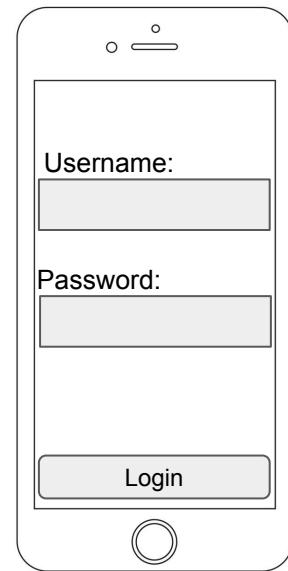
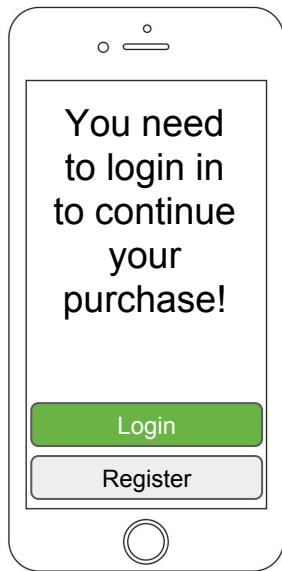
Bootstrap: specify any  
core server to route load  
across the whole cluster

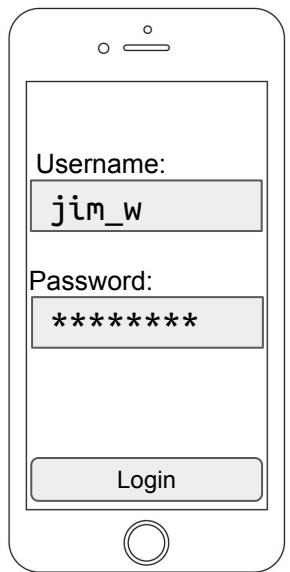
# Consistency

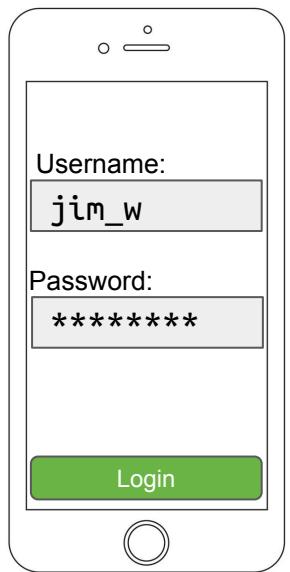




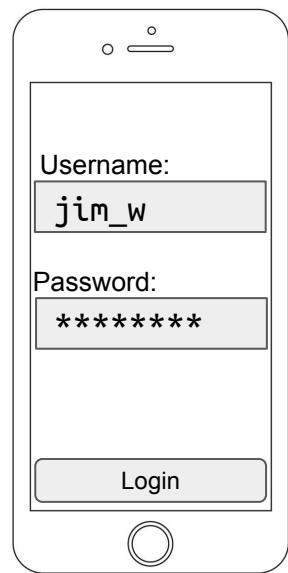




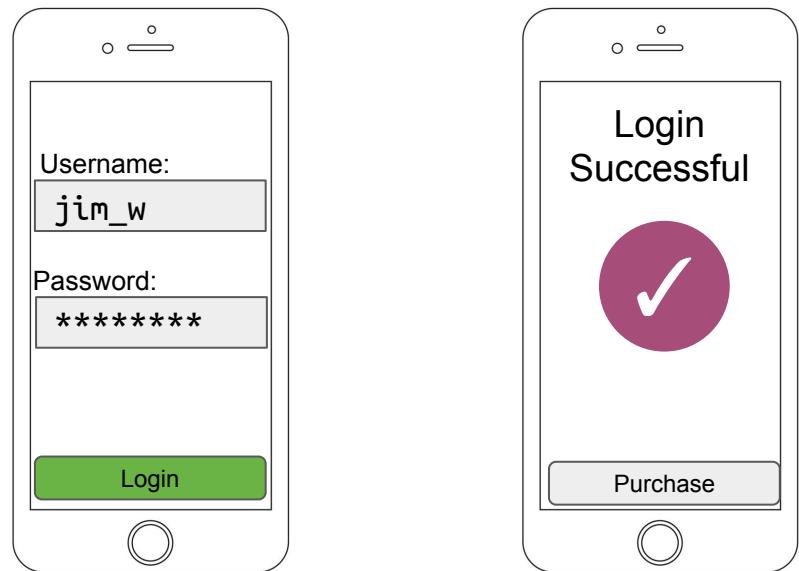




# A few moments later...



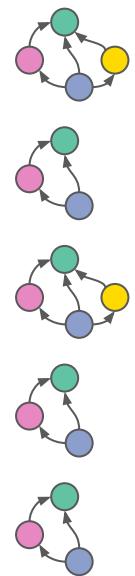
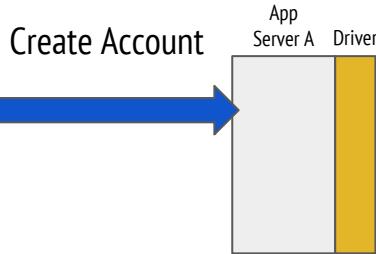
# A few moments later...



Q Why didn't this work?  
A Eventual Consistency

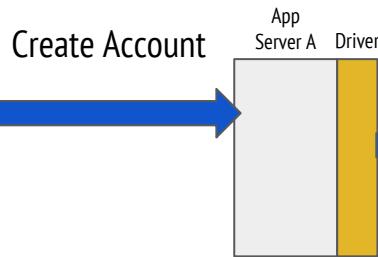


Create Account

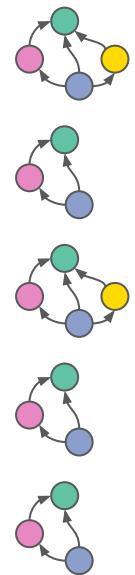
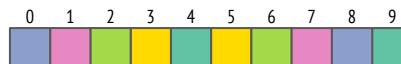
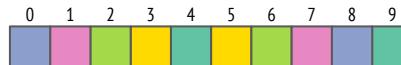


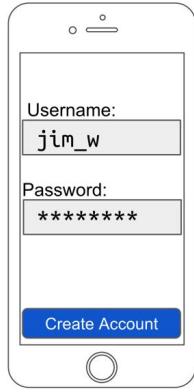


Create Account

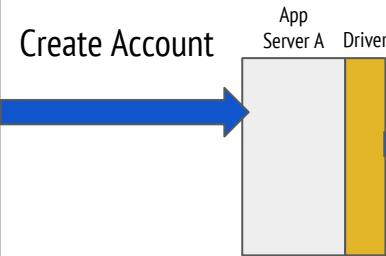


CREATE (:User)





Create Account



CREATE (:User)



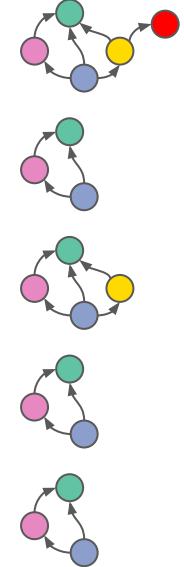
0 1 2 3 4 5 6 7 8 9 10 11

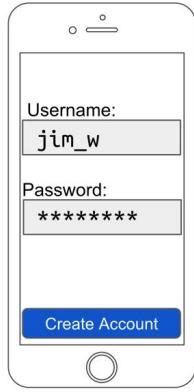
0 1 2 3 4 5 6 7 8 9

0 1 2 3 4 5 6 7 8 9 10

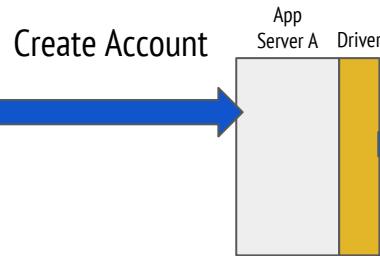
0 1 2 3 4 5 6 7 8 9

0 1 2 3 4 5 6 7 8 9





Create Account



CREATE (:User)



0 1 2 3 4 5 6 7 8 9 10 11

Blue	Pink	Green	Yellow	Teal	Yellow	Blue	Pink	Green	Teal	Yellow	Red
------	------	-------	--------	------	--------	------	------	-------	------	--------	-----

0 1 2 3 4 5 6 7 8 9

Blue	Pink	Green	Yellow	Teal	Yellow	Green	Pink	Blue	Teal
------	------	-------	--------	------	--------	-------	------	------	------

0 1 2 3 4 5 6 7 8 9 10 11

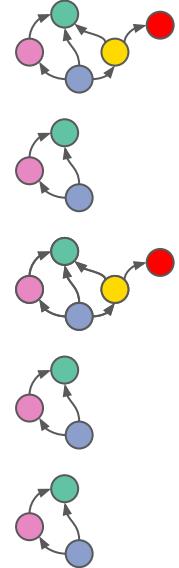
Blue	Pink	Green	Yellow	Teal	Yellow	Green	Pink	Blue	Yellow	Red
------	------	-------	--------	------	--------	-------	------	------	--------	-----

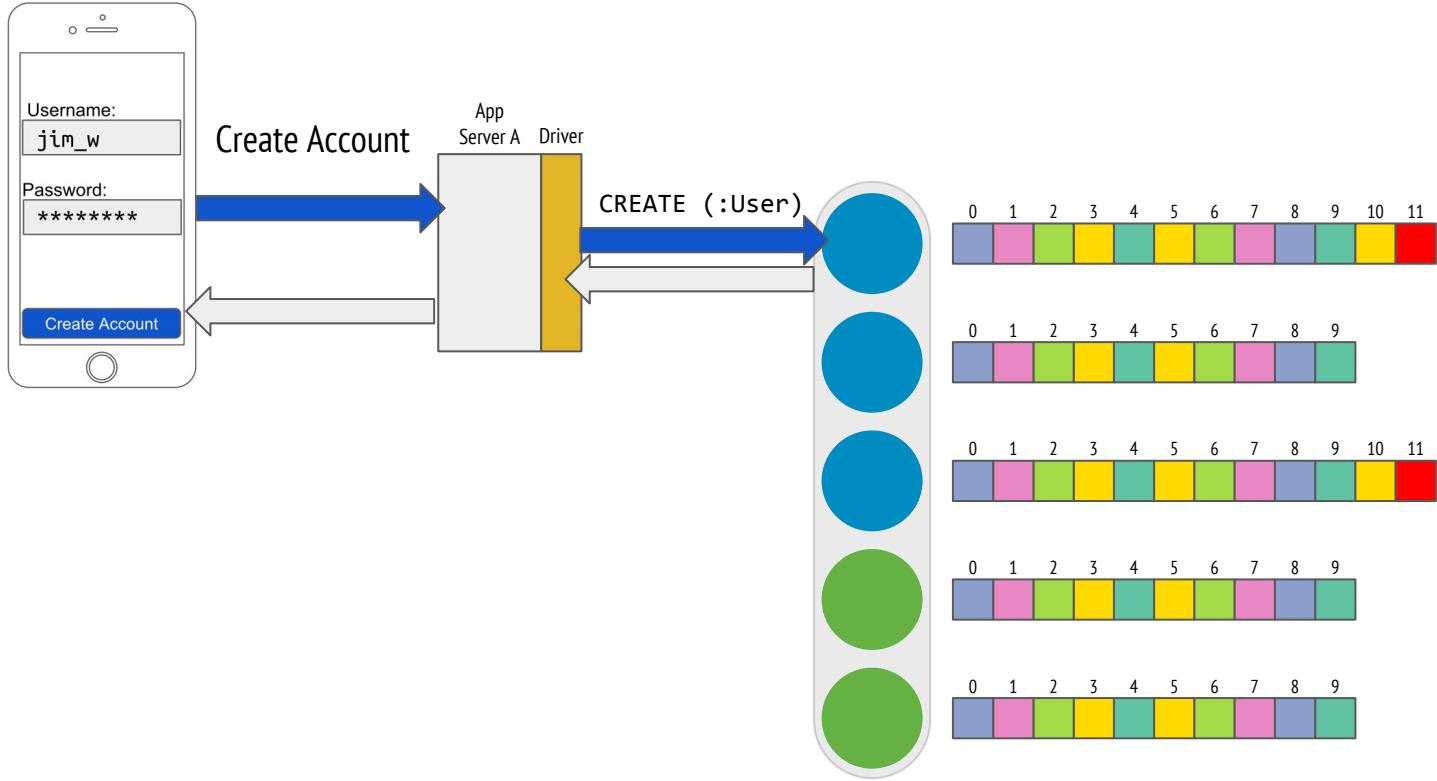
0 1 2 3 4 5 6 7 8 9

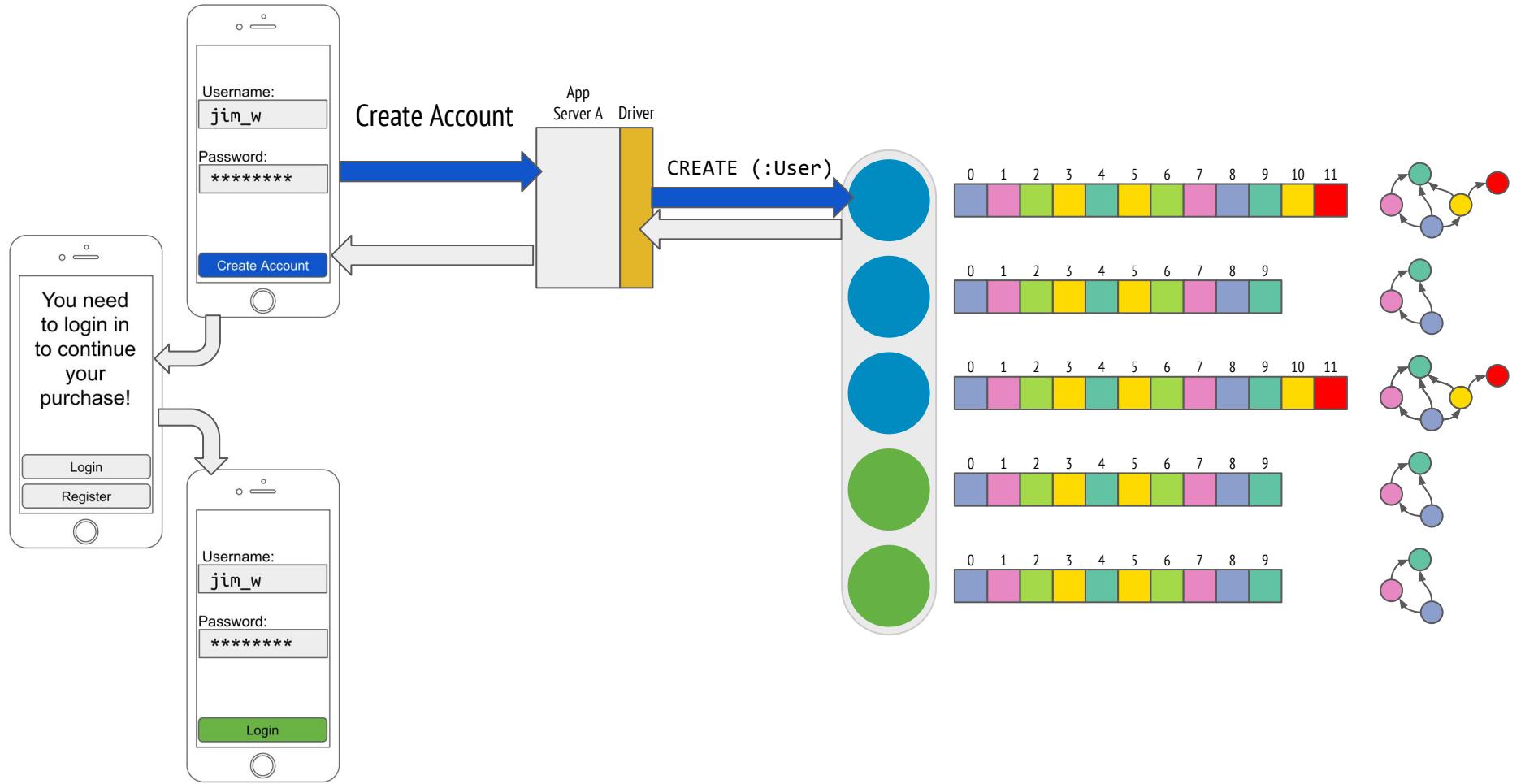
Blue	Pink	Green	Yellow	Teal	Yellow	Green	Pink	Blue	Teal
------	------	-------	--------	------	--------	-------	------	------	------

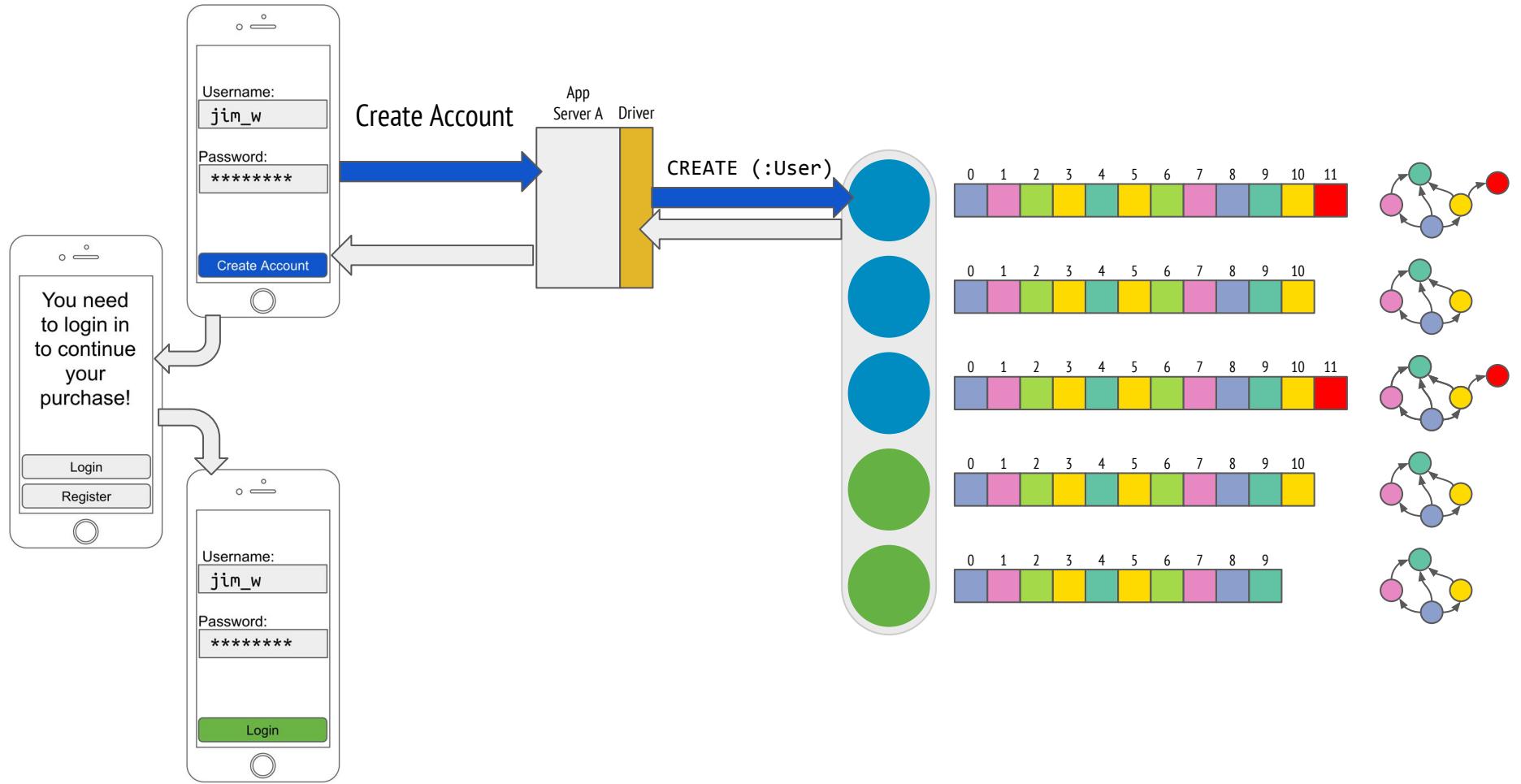
0 1 2 3 4 5 6 7 8 9

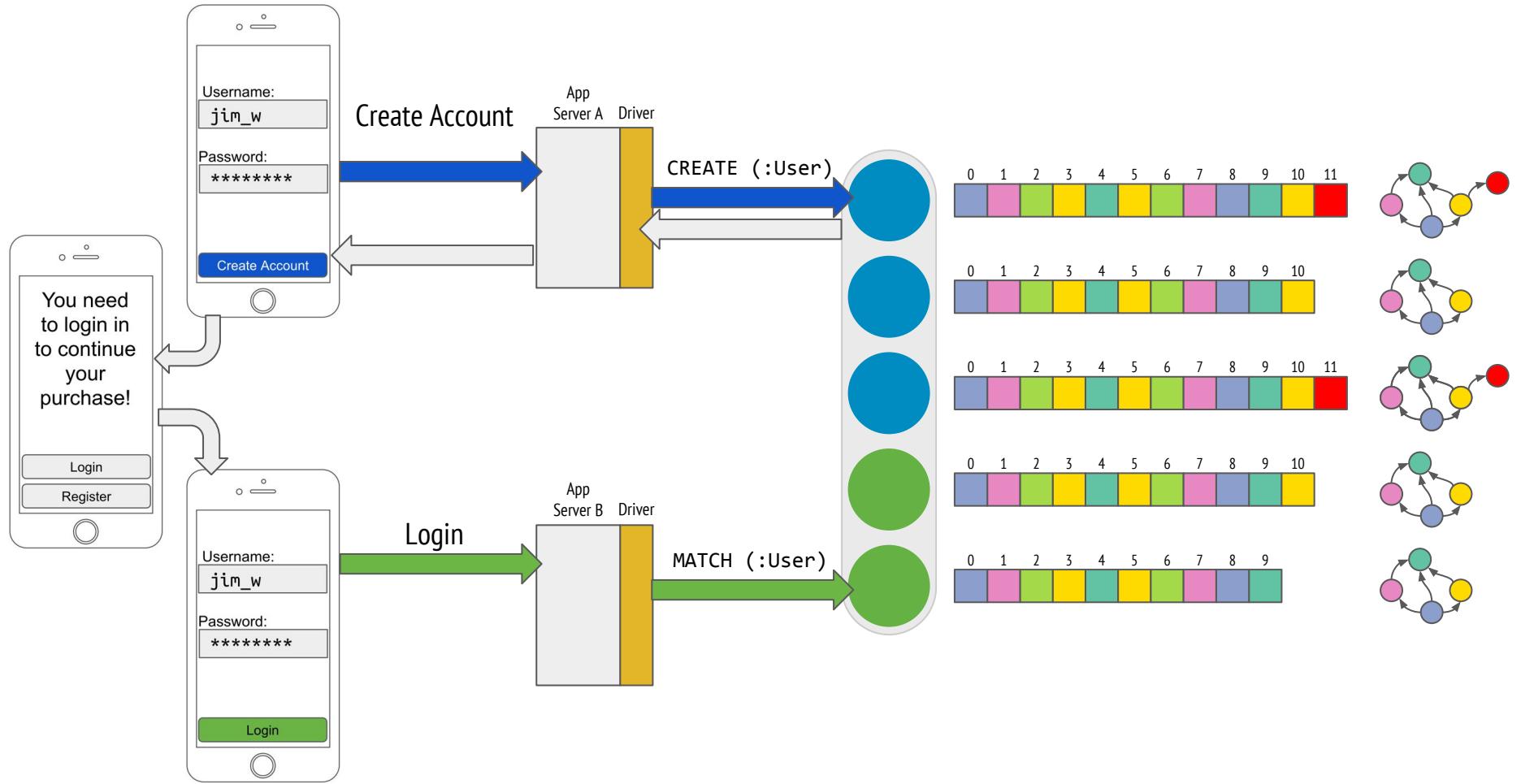
Blue	Pink	Green	Yellow	Teal	Yellow	Green	Pink	Blue	Teal
------	------	-------	--------	------	--------	-------	------	------	------

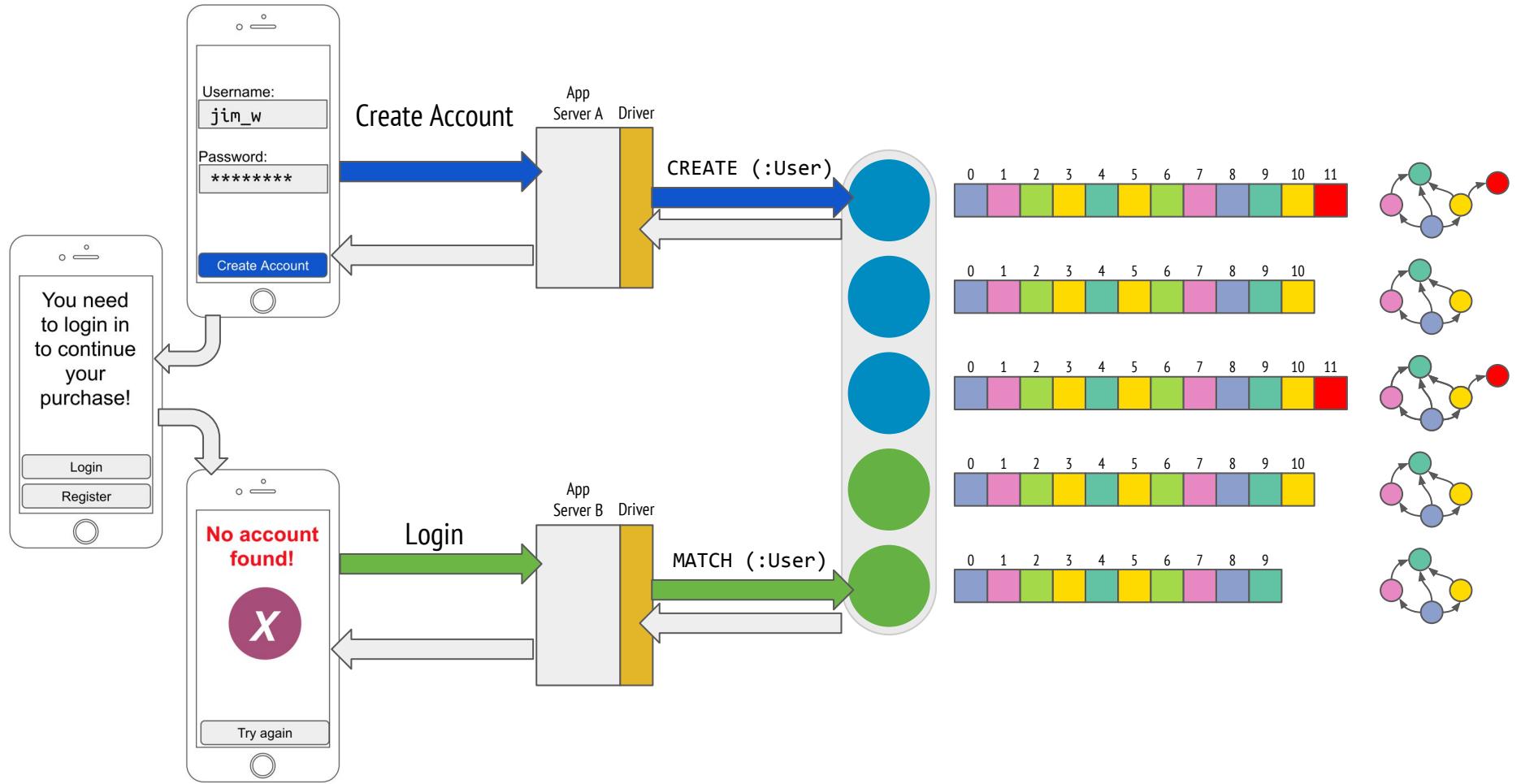






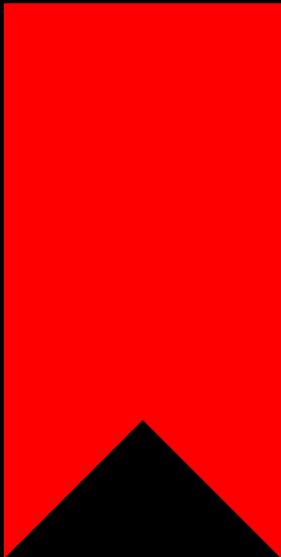




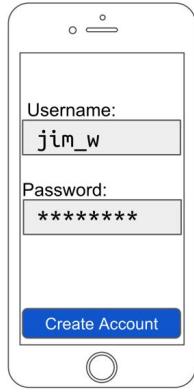


Let's try again, with Causal Consistency

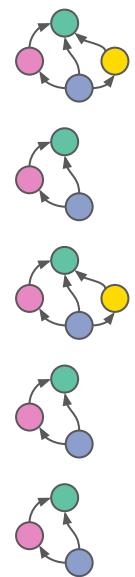
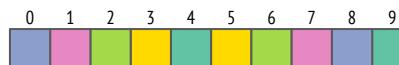
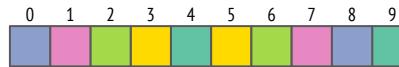
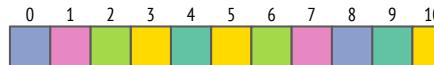
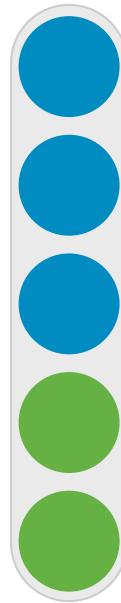
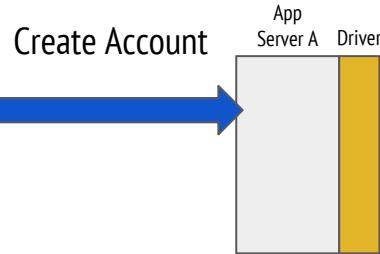
# Bookmark



- Session token
- String (for portability)
- Opaque to application
- Represents ultimate user's most recent view of the graph
- More capabilities to come

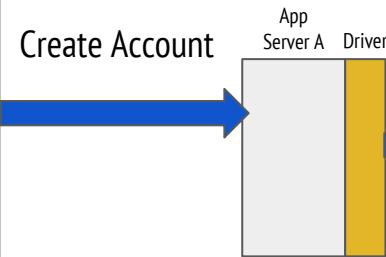


Create Account

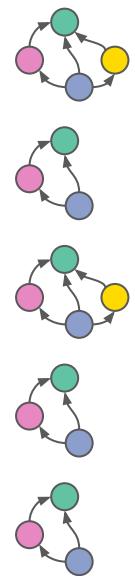
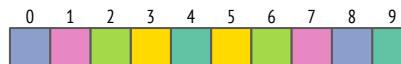


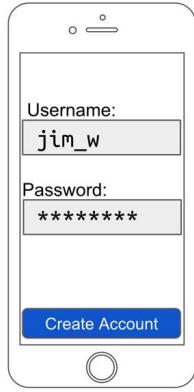


Create Account

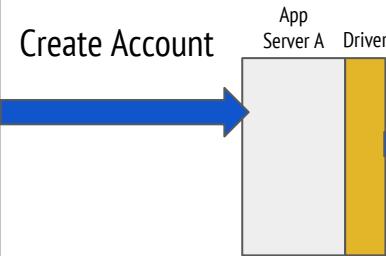


CREATE (:User)





Create Account



CREATE (:User)



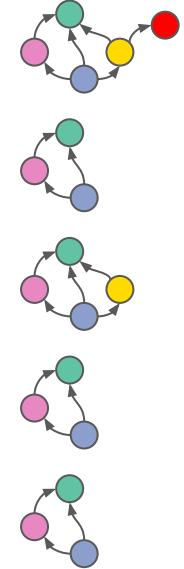
0 1 2 3 4 5 6 7 8 9 10 11

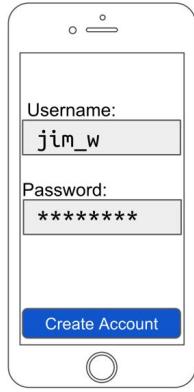
0 1 2 3 4 5 6 7 8 9

0 1 2 3 4 5 6 7 8 9 10

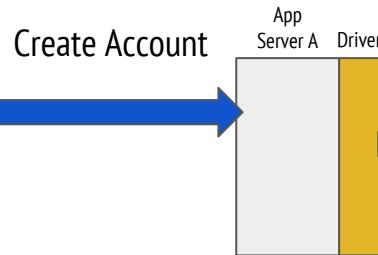
0 1 2 3 4 5 6 7 8 9

0 1 2 3 4 5 6 7 8 9





Create Account



CREATE (:User)



0 1 2 3 4 5 6 7 8 9 10 11

Blue	Pink	Green	Yellow	Teal	Yellow	Blue	Pink	Green	Teal	Yellow	Red
------	------	-------	--------	------	--------	------	------	-------	------	--------	-----

0 1 2 3 4 5 6 7 8 9

Blue	Pink	Green	Yellow	Teal	Yellow	Green	Pink	Blue	Teal
------	------	-------	--------	------	--------	-------	------	------	------

0 1 2 3 4 5 6 7 8 9 10 11

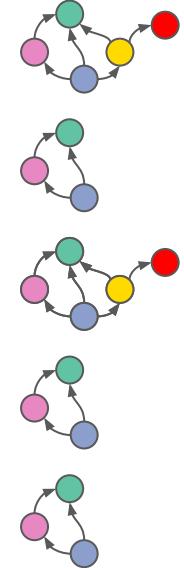
Blue	Pink	Green	Yellow	Teal	Yellow	Green	Pink	Blue	Teal	Yellow	Red
------	------	-------	--------	------	--------	-------	------	------	------	--------	-----

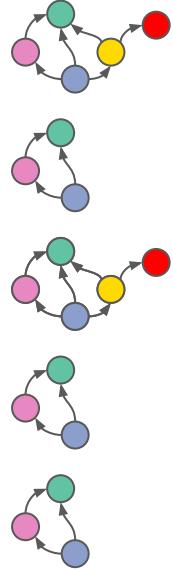
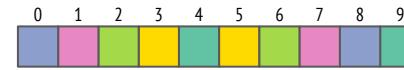
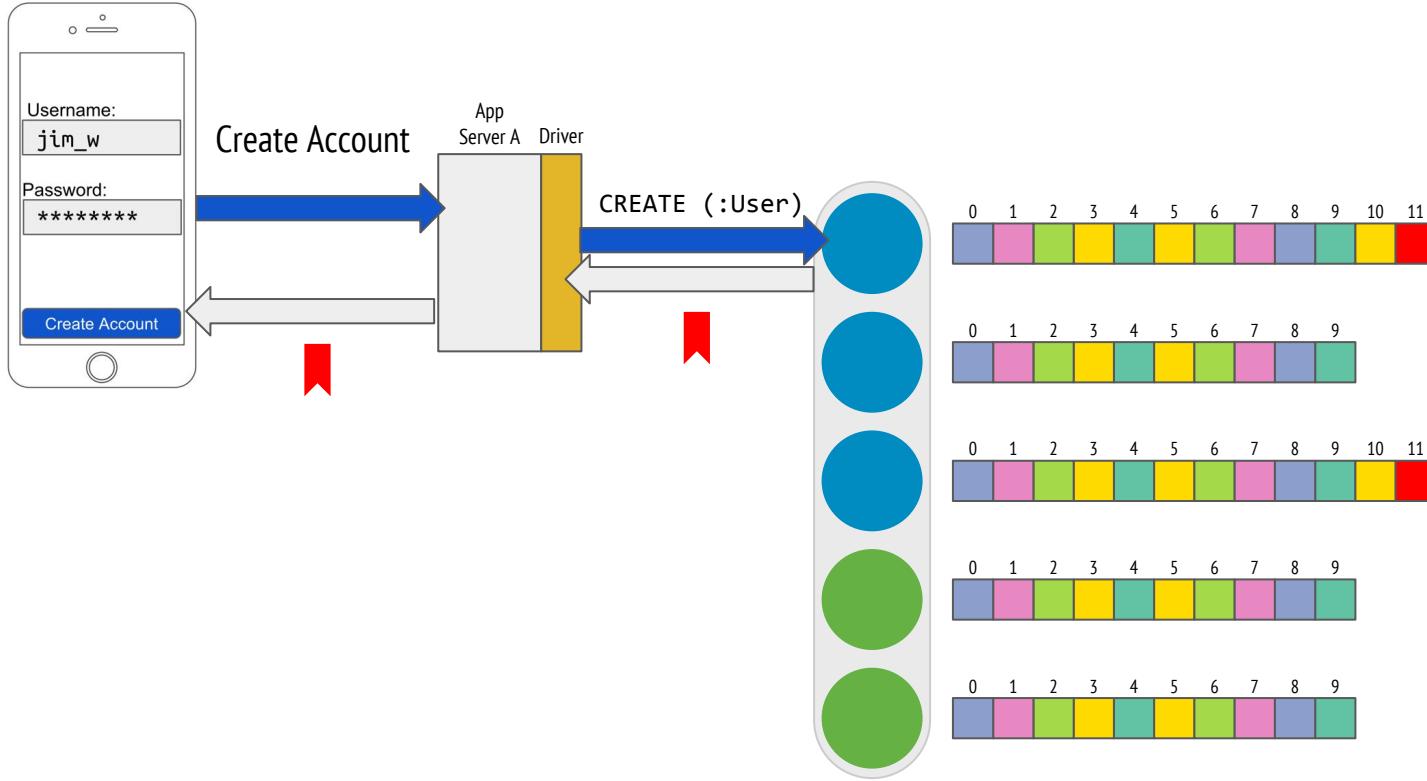
0 1 2 3 4 5 6 7 8 9

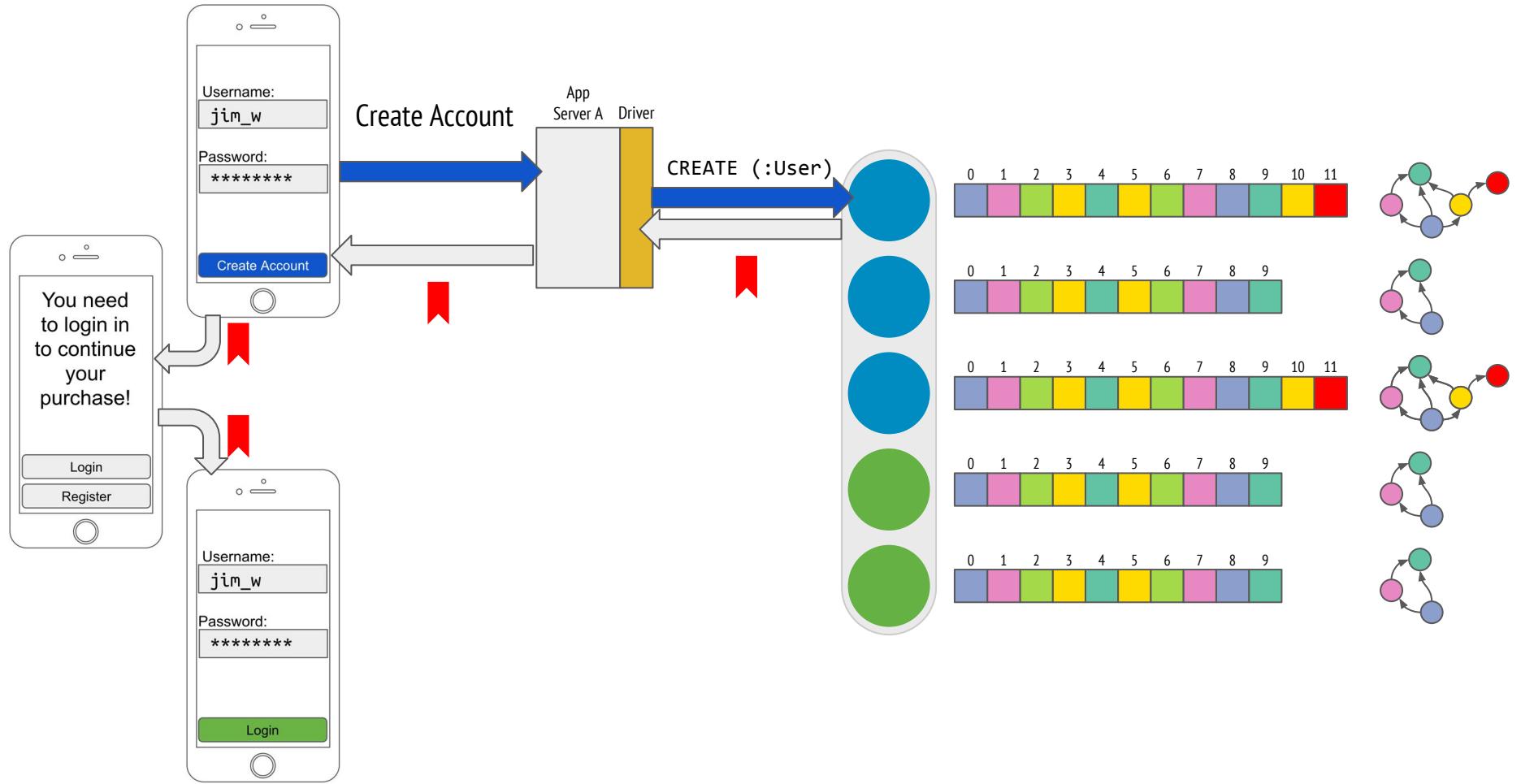
Blue	Pink	Green	Yellow	Teal	Yellow	Green	Pink	Blue	Teal
------	------	-------	--------	------	--------	-------	------	------	------

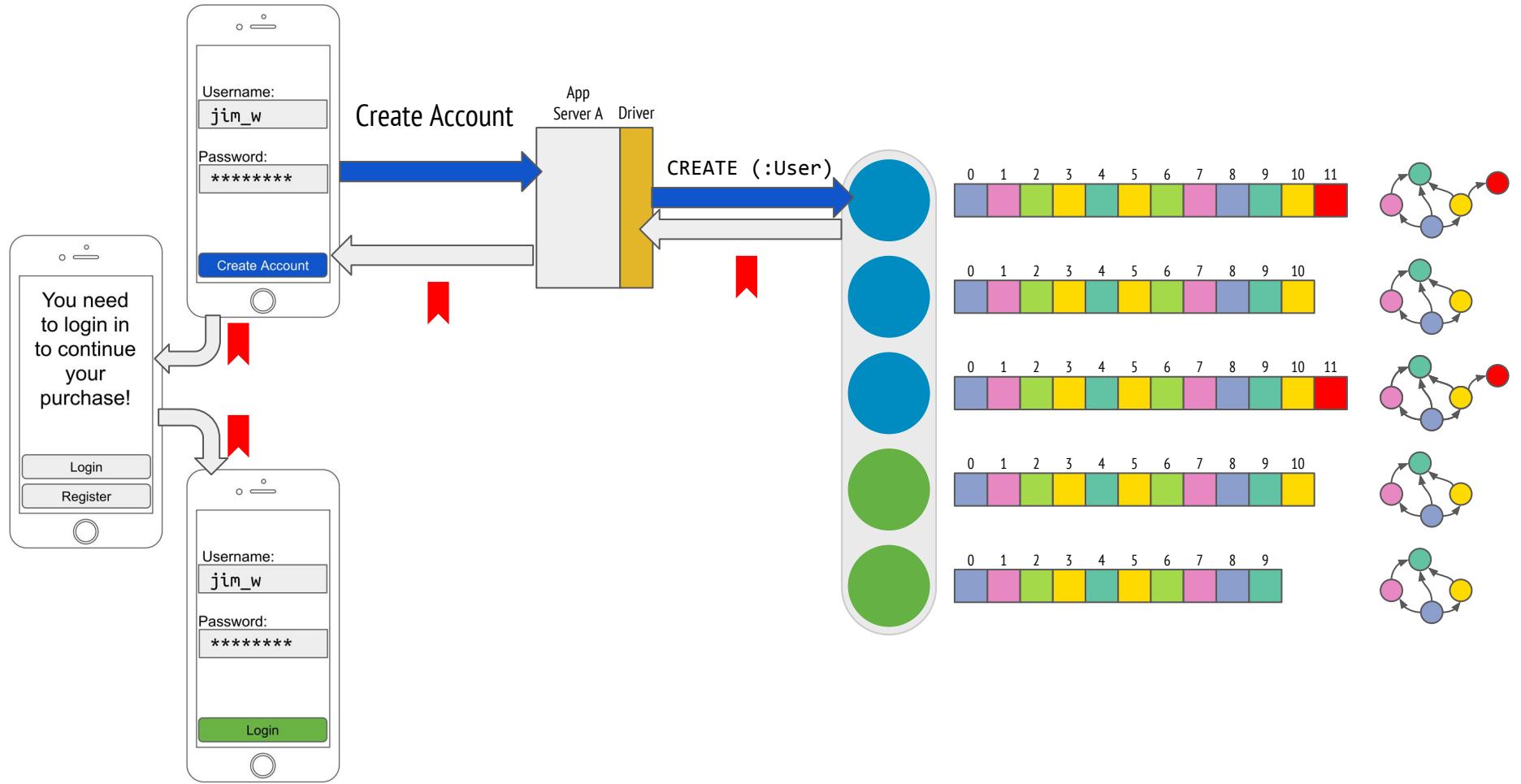
0 1 2 3 4 5 6 7 8 9

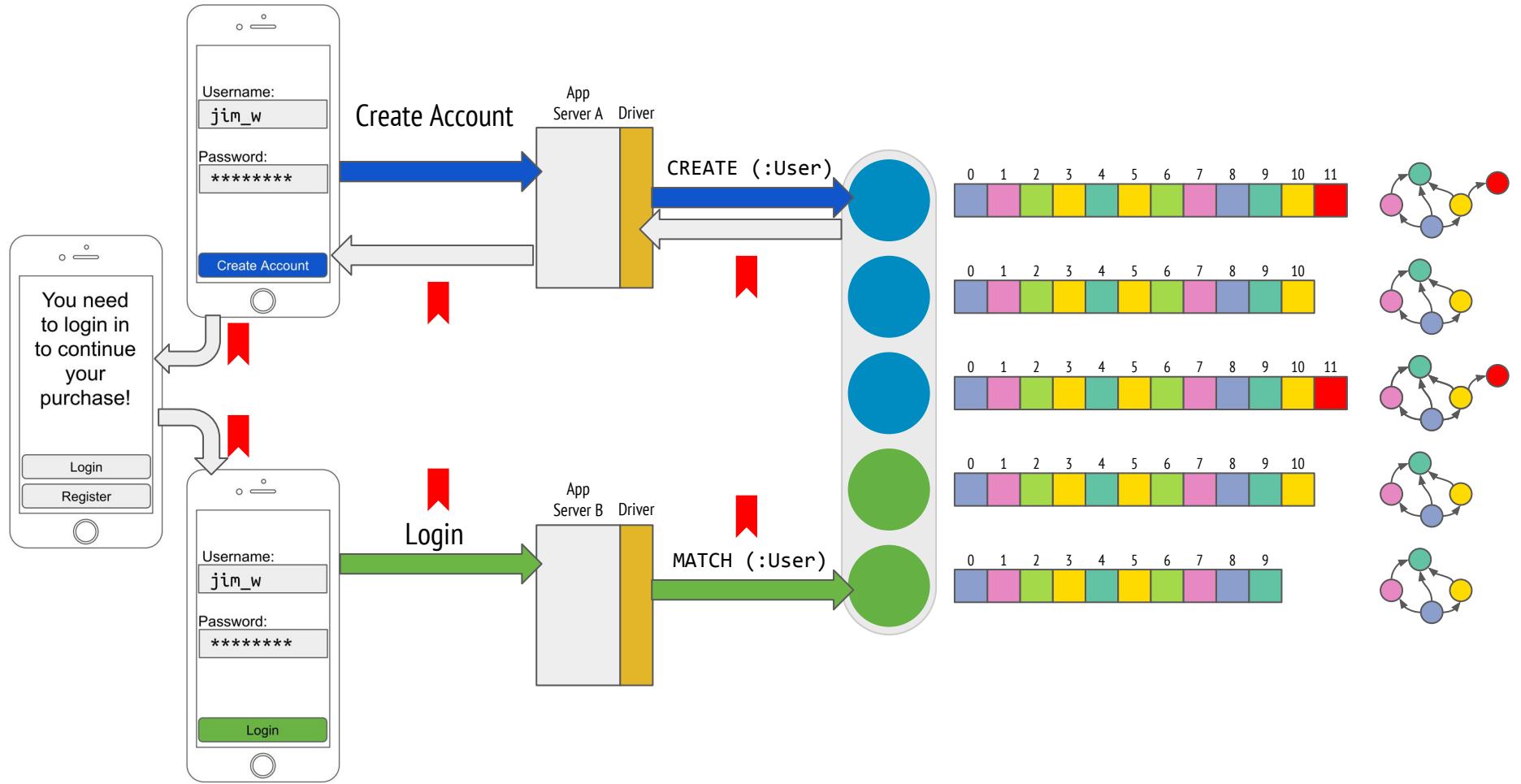
Blue	Pink	Green	Yellow	Teal	Yellow	Green	Pink	Blue	Teal
------	------	-------	--------	------	--------	-------	------	------	------

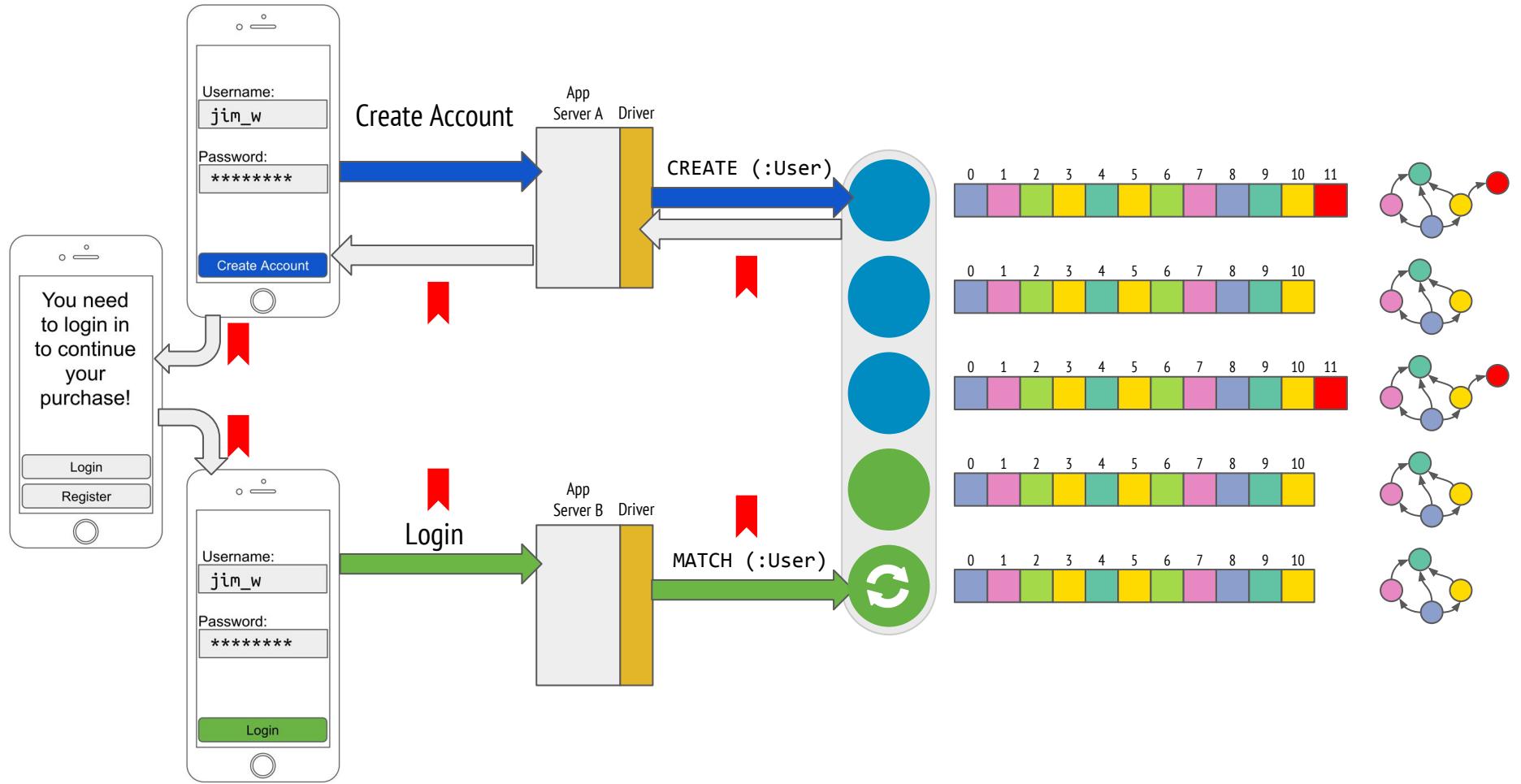


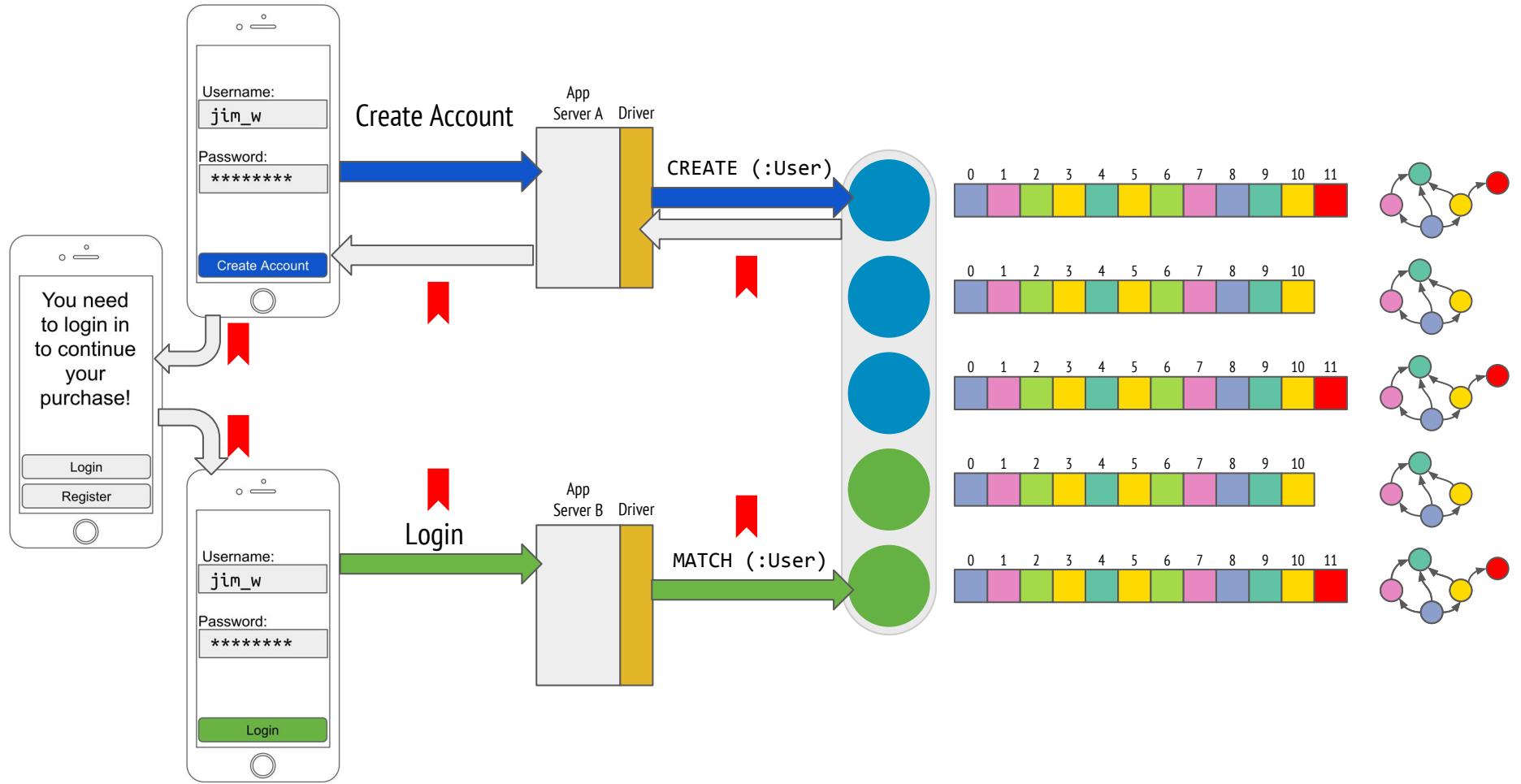


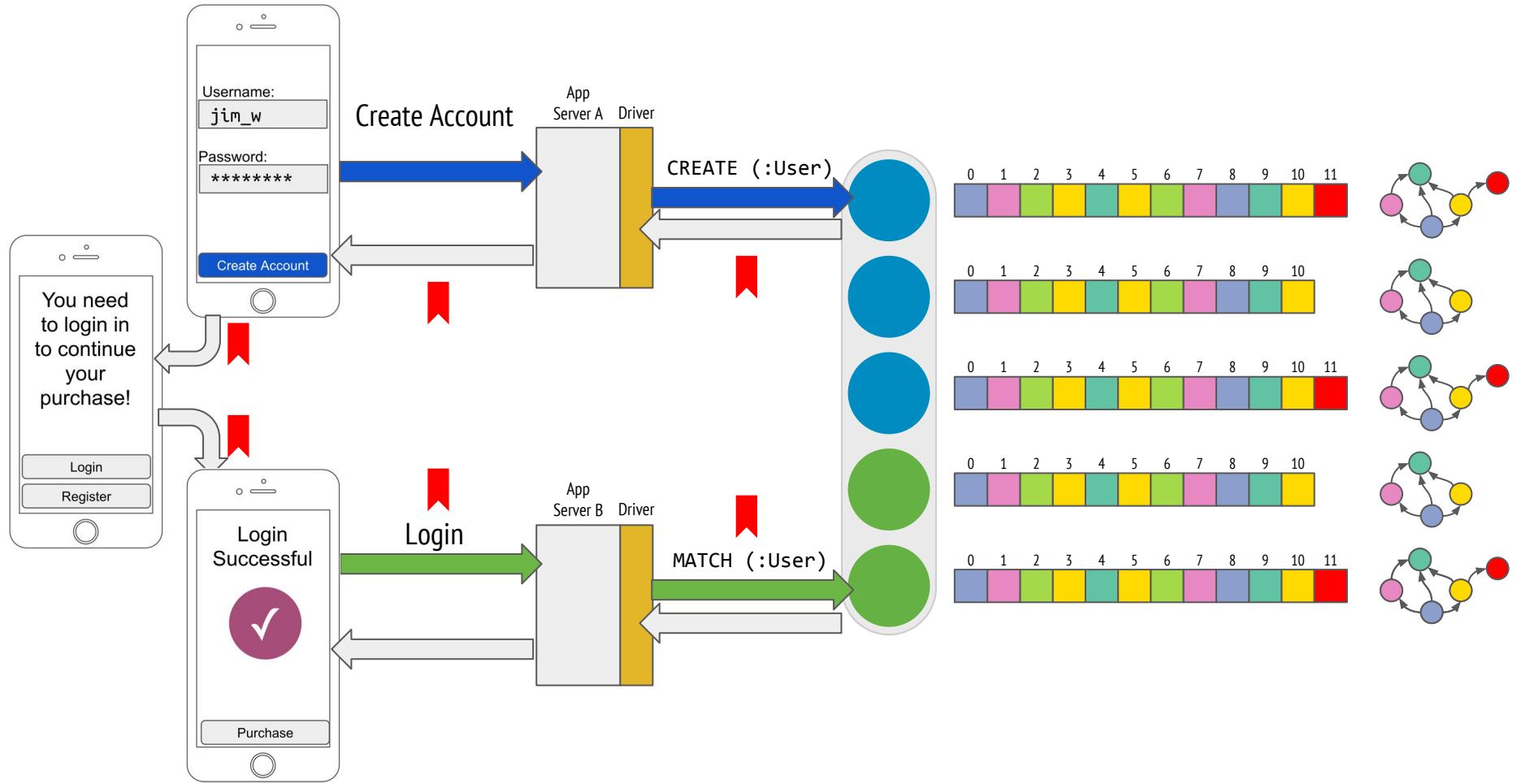












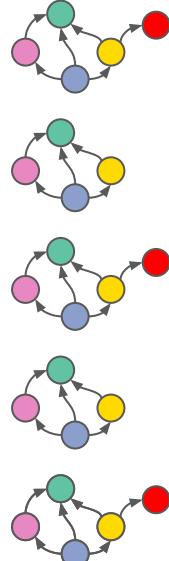
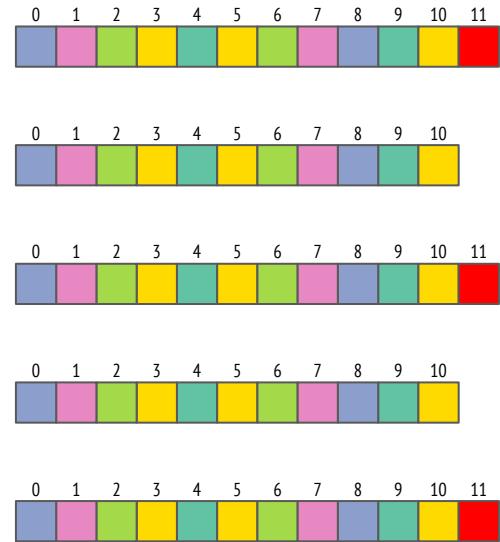
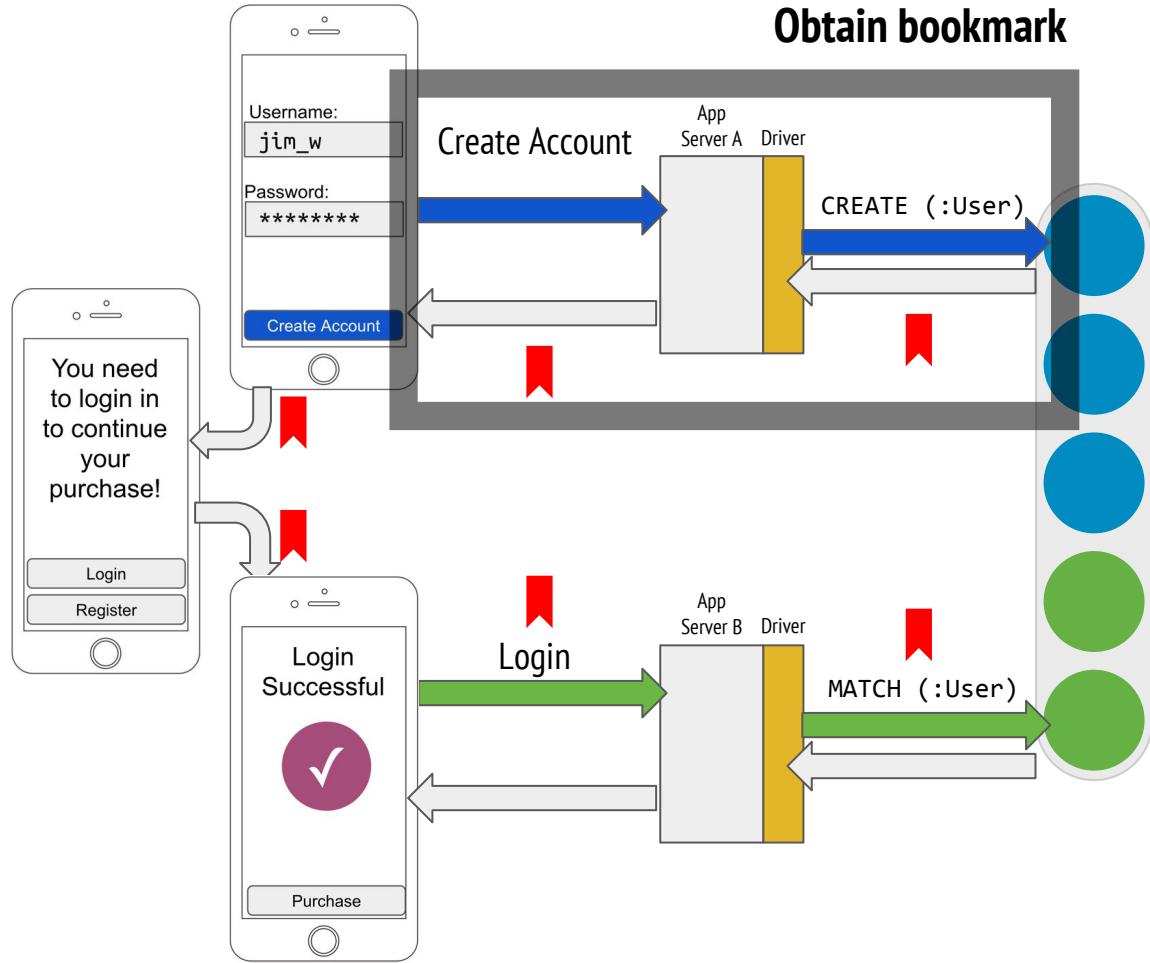
# Obtain bookmark

```
try ( Session session = driver.session( AccessMode.WRITE ) )
{
    try ( Transaction tx = session.beginTransaction() )
    {
        tx.run( "CREATE (user:User {userId: {userId}, passwordHash: {passwordHash}})",
                parameters( "userId", userId, "passwordHash", passwordHash ) );

        tx.success();
    }

    String bookmark = session.lastBookmark();
}
```

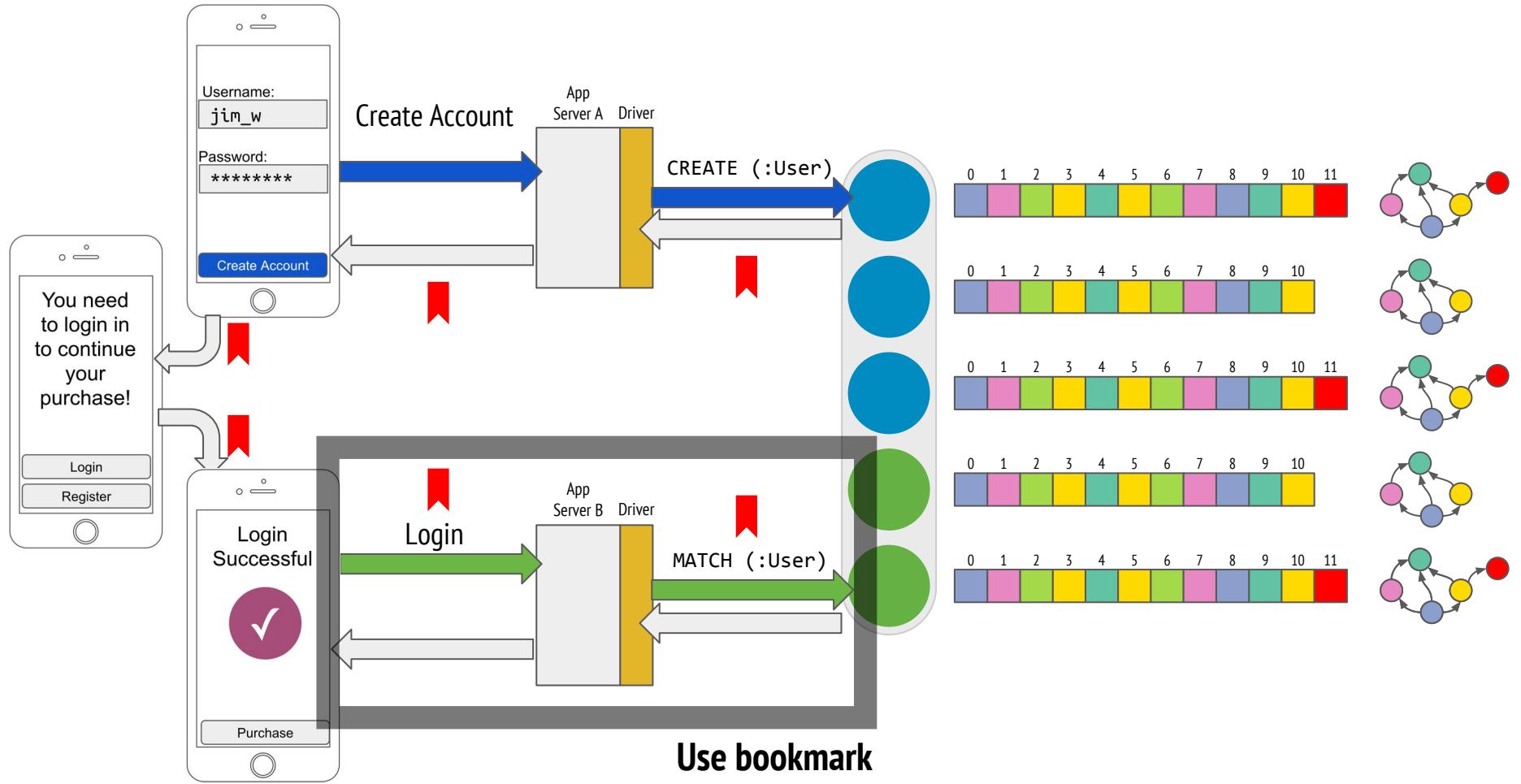
# Obtain bookmark

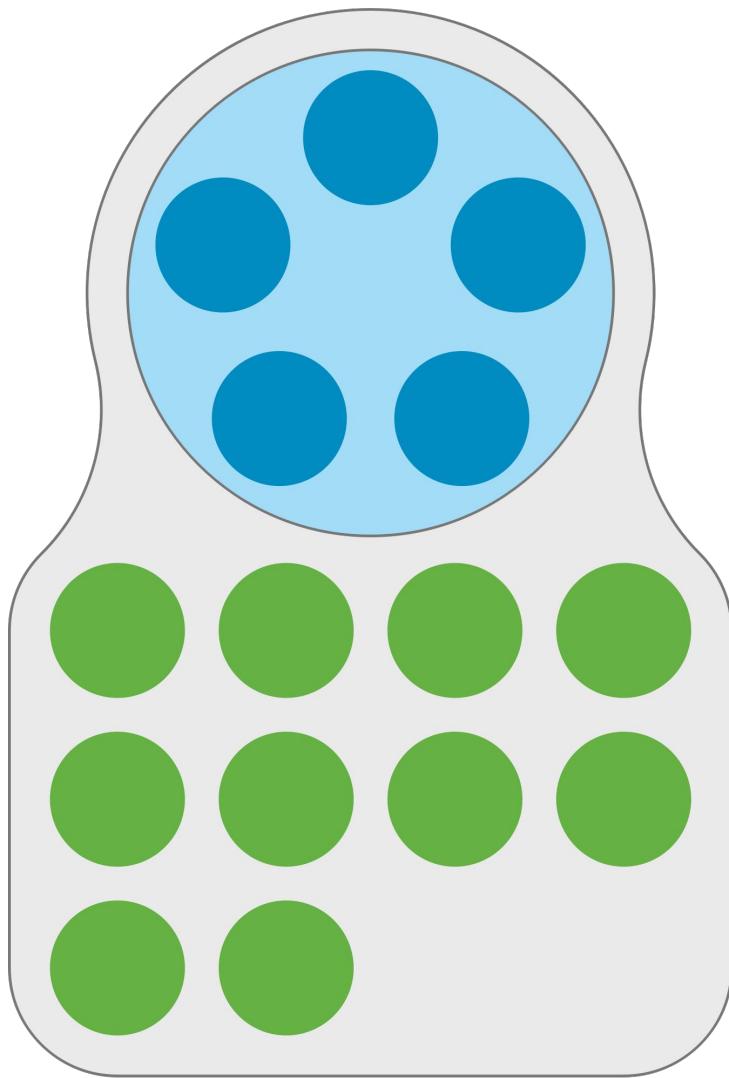


# Use a bookmark

```
try ( Session session = driver.session( AccessMode.READ ) )
{
    try ( Transaction tx = session.beginTransaction( bookmark ) )
    {
        tx.run( "MATCH (user:User {userId: {userId}}) RETURN *",
                parameters( "userId", userId ) );

        tx.success();
    }
}
```





# Core

Updating the graph  
Continuous operation

# Read Replicas

Queries, analysis, reporting  
At large scale

Questions?