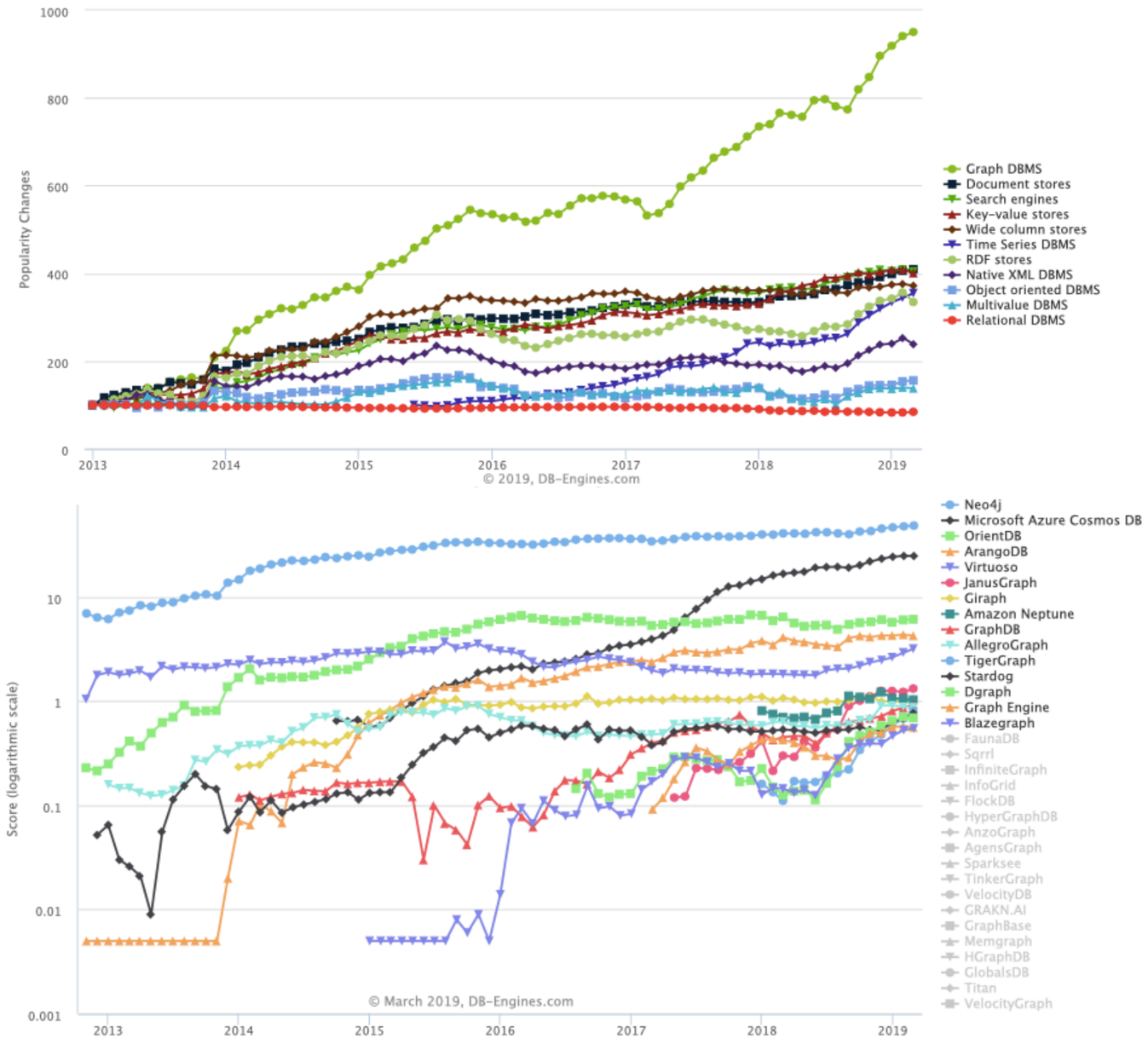
NEO4J GRAPH DATABASE WITH TIMED OPERATIONS

UTM 3252 BIG DATA MANAGEMENT SYSTEMS AND TOOLS

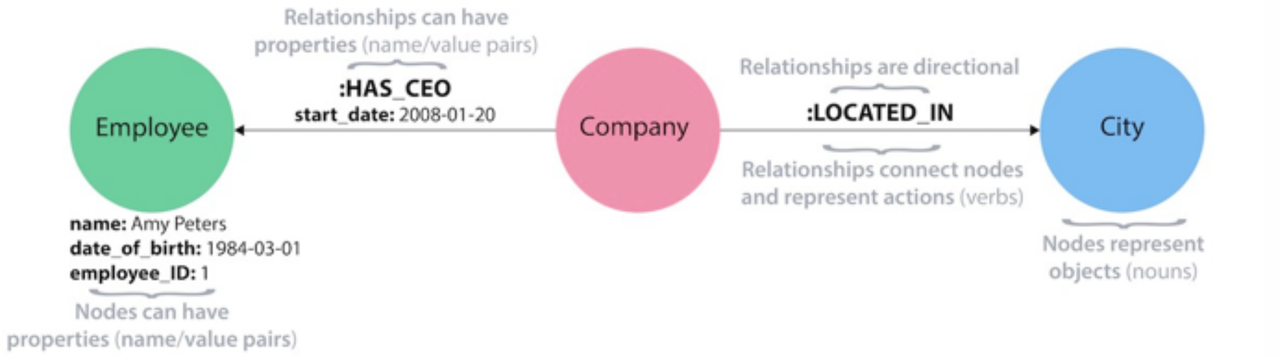
PRESENTED BY: ASHOK MISTRY

DB-ENGINES.COM - POPULARITY TREND OF GRAPH DBMS



- Graph DBMS getting mainstream adoption.
- Graph databases used in many applications:
  - fraud detection
  - recommendation engines
  - managing social media
  - knowledge graph
- Used extensively in investigations of the Panama Papers.

CYPHER LANGUAGE




```
()              //anonymous node (no label or variable) can refer to any node in the database
(c:Company)     //using variable c and label Company
(:Company)      //no variable, label Company
(work:Company)  //using variable work and label Company

//data stored with this direction
CREATE (c:Company)-[:HAS_CEO]->(e:Employee)
CREATE (c:Company {name: “Bell Canada”})-[rel:HAS_CEO]->(e:Employee {name: “Amy Peters"})

//query relationship backwards will not return results
MATCH (c:Company)<-[:HAS_CEO]-(e:Employee)

//better to query with undirected relationship unless sure of direction
MATCH (c:Company)-[:HAS_CEO]-(e:Employee)

//return all values
MATCH (e:Employee)
RETURN e
```

- A graph model is composed of two elements: Node and a Relationship.
- Each node represents an entity (a person, place, thing), think noun. A node contains properties (key-value pair).
- Each relationship represents how the two nodes are connected, think verb. Relationships have direction and can have properties.

NEO4J AND SELECTION OF DATASET

Neo4J is an open-source, NoSQL, ACID-compliant native graph database. Available for free download as a Desktop and Community Edition. A commercial Enterprise Edition includes backups, clustering, and failover capabilities.

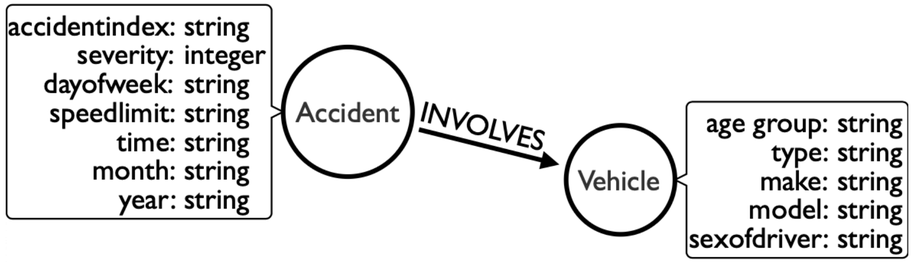
The dataset used is the UK Road Safety consisting of accidents and the vehicles from 2005 to 2016 obtained from Kaggle. The original data comes from the Open Data website of the UK government, where they have been published by the Department of Transport. This dataset was chosen because of its large number of rows AND columns.

Consists of two files:

- **Accident_Information.csv** (672.77 MB), 2047256 rows, 34 columns
- **Vehicle_Information.csv** (614.57 MB), 2177205 rows, 24 columns

The two files can be linked through the **Accident_Index** column, a unique traffic accident identifier.

RESULTS



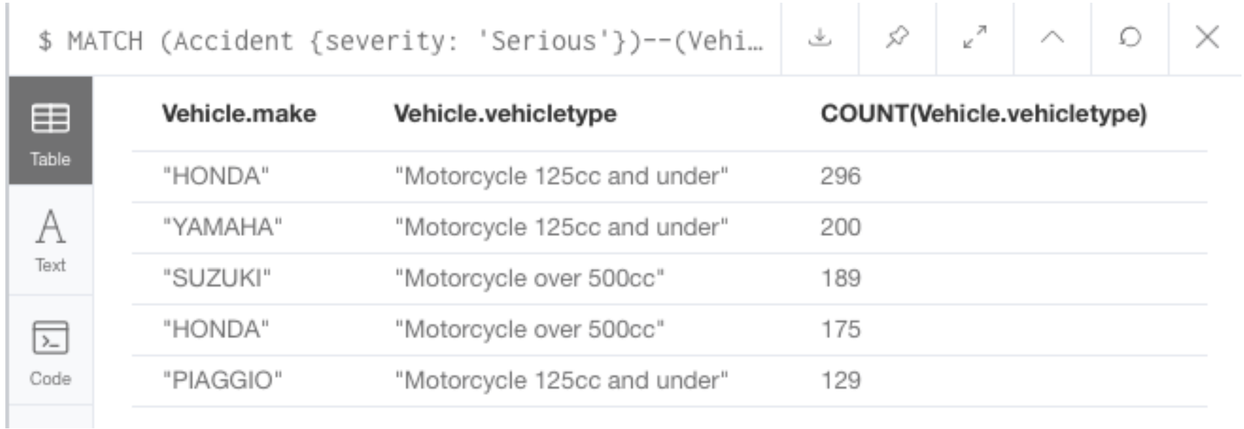
PostgreSQL: SQL

```
SELECT "make", "Vehicle_Type", COUNT("Vehicle_Type")

FROM vehicle

INNER JOIN accident ON vehicle."Accident_Index" =
accident."Accident_Index"

WHERE "Vehicle_Type" LIKE 'Motorcycle%' AND
"Sex_of_Driver" = 'Female' AND "Accident_Severity" =
'Serious'

GROUP BY "make", "Vehicle_Type"

ORDER BY COUNT("Vehicle_Type") DESC LIMIT 5
```

Successfully run. Total query runtime: 9 secs 852 msec.
5 rows affected.

NEO4J: Cypher

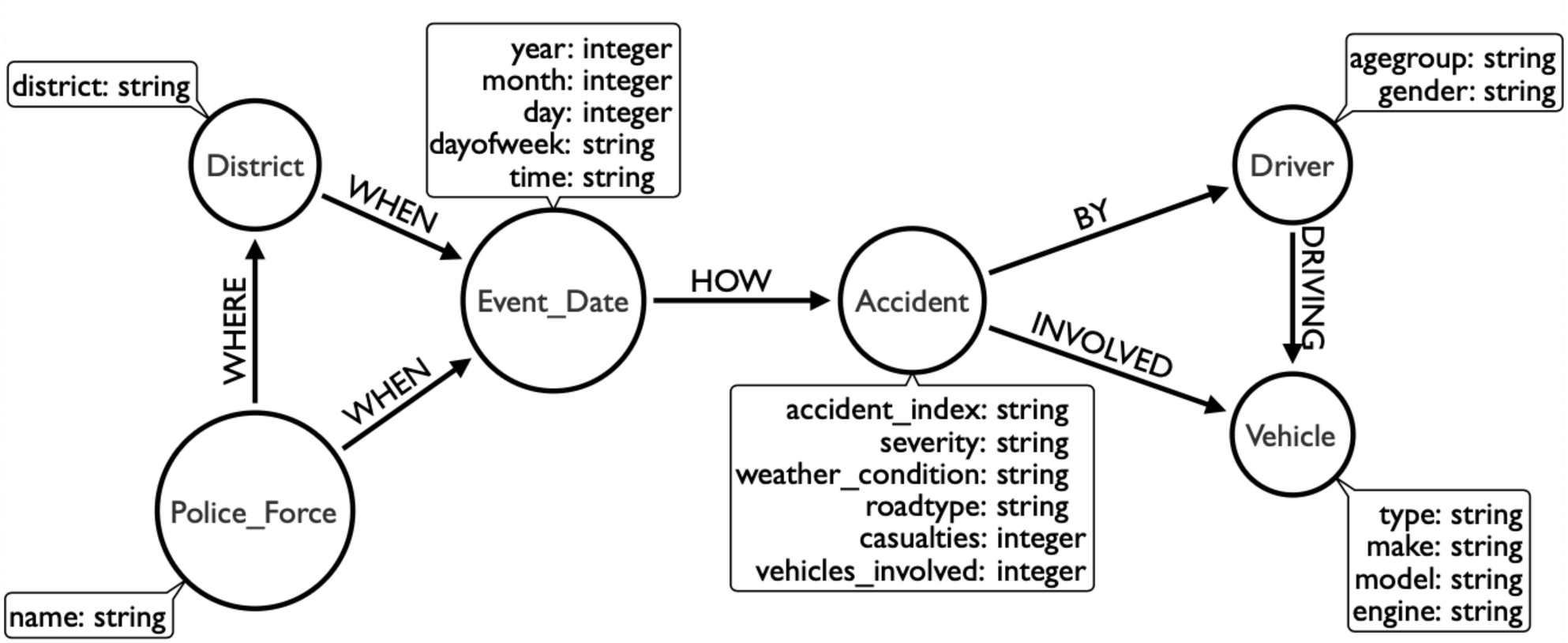
```
MATCH (a:Accident {severity: 'Serious'})-[:INVOLVES]->(v:Vehicle)

WHERE v.vehicletype STARTS WITH 'Motorcycle' AND v.sexofdriver = 'Female'

RETURN v.make, v.vehicletype, COUNT(v.vehicletype)

ORDER by COUNT(v.vehicletype) DESC limit 5
```



Vehicle.make	Vehicle.vehicletype	COUNT(Vehicle.vehicletype)
"HONDA"	"Motorcycle 125cc and under"	296
"YAMAHA"	"Motorcycle 125cc and under"	200
"SUZUKI"	"Motorcycle over 500cc"	189
"HONDA"	"Motorcycle over 500cc"	175
"PIAGGIO"	"Motorcycle 125cc and under"	129

Started streaming 5 records after 3050 ms and completed after 3050 ms.

ENHANCED ROAD SAFETY MODEL

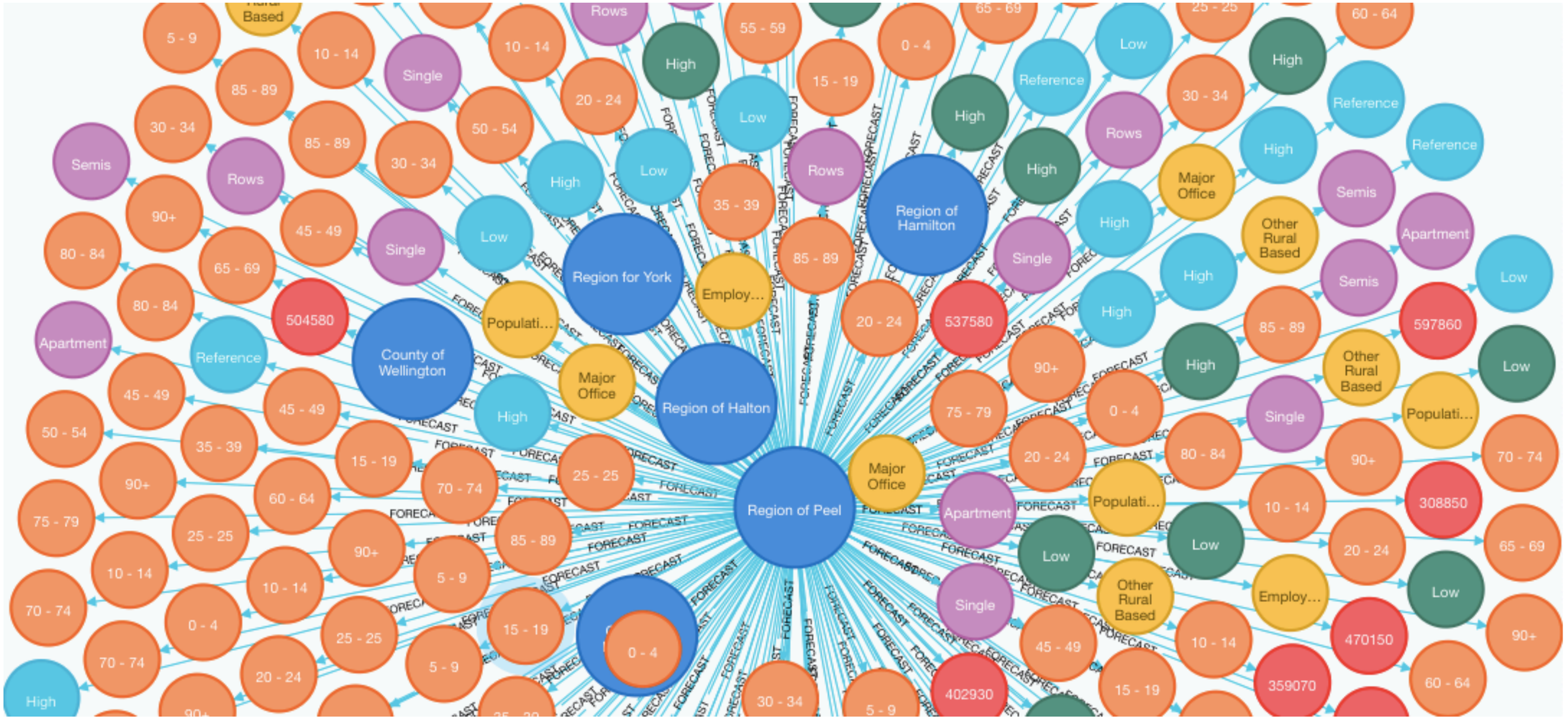


Created with arrow tool from http://www.apcjones.com/arrows/

CONCLUSION

- IMPLEMENTING A PROJECT REQUIRES FORE THOUGHT, GOOD DESIGN WORK AND BETTER PLANNING.
- DESIGN WILL CHANGE AS NEW NODES AND RELATIONSHIPS ARE INTRODUCED.
- EASY TO REMOVE NODES AND RELATIONSHIPS WITH THE MATCH DETACH AND DELETE, BUT TAKES LONG TO DO.
- GRAPH-BASED QUERIES ARE FASTER AS RELATIONSHIPS ARE ALREADY PRE-ESTABLISHED DURING THE CREATE AND MERGE OPERATION. WITH SQL MANY JOIN OPERATIONS ARE COSTLY AND ADDS TO THE RUNTIME.
- CYPHER QUERY LANGUAGE MAKES WRITING QUERIES SIMPLE AND EASY TO UNDERSTAND.
- THERE WERE PROBLEMS WHEN USING LARGE DATASETS AND NEO4J WITH CONNECTION DROPPING DUE TO PERFORMANCE, LACK OF MEMORY AND LOW LATENCY. RESEARCH OTHER GRAPH DATABASES WITH BETTER SCALABILITY.

ANY QUESTIONS?



Greater Golden Horseshoe Growth Forecast DataSet