neo4j graphsummit Breda

Architecting Innovative Graph Applications

March 13 2024

Neo4j Field Engineering

Kees Vegter kees.vegter@neo4j.com
Niels de Jong niels.dejong@neo4j.com

Goal of Today

Demonstrate (with YOUR HELP :-)), how to build a Graph Application

Agenda

1. Logistics
2. Introduction
3. Use Case Explanation
4. Building the Solution

BREAK (15min) Around 15:30

5. Discussion: How to Improve the Use Case Further On
6. Final Q&A
7. Goodbye

Logistics

First things first ...

Logistics

WIFI Access:	**Name:** neo4j | **Password:** graphsummit24
Restrooms:	Out of the room and to ☒ the left ☐ the right ☐ Straight ahead
Chargers:	☒ under ☐ on top ☐ under the table in first row
Material of the workshop:	https://github.com/kvegter/gsummit2024

Introduction

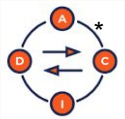
A short overview of the Neo4j Product

Neo4j Native Graph Database - What is it?

Neo4j is a NORMAL Database (DB)

- Comparable with other DBs like Postgres, MySQL, MariaDB, Oracle, HANA DB, etc\
- Securly save, insert, update & query data
- Backup & Recovery
- Highly Scalable

Neo4j is “ACID Compliant” and transaction save



- **ACID** (Atomicity, Consistency, Isolation, Durability)
- Data is stored consistently on transaction level without any losses during an outage
- **Most relational DB’s are ACID compliant, too**

Neo4j is used 80%+ for OLTP Workloads

- > 80% of Neo4j Customers using the database as normal OLTP data store
- Data is stored, changed, deleted and queried
- With the same or even better performance compared to relational database systems

* Source: Medium.com - Link to the Article

What is the Value Using a Native Graph Database?

How data is stored on disk

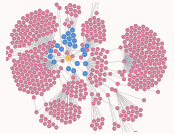
- Data is already stored as connected
- Data is efficiently placed on-disk using “Index-free Adjacency”
- A GDB stores **Nodes** and **Relations** instead of Rows & Columns
- **Semantics** may be built into the data!

How to query the data

```
(:Product)-[:CONTAINS]->(:Part)
```

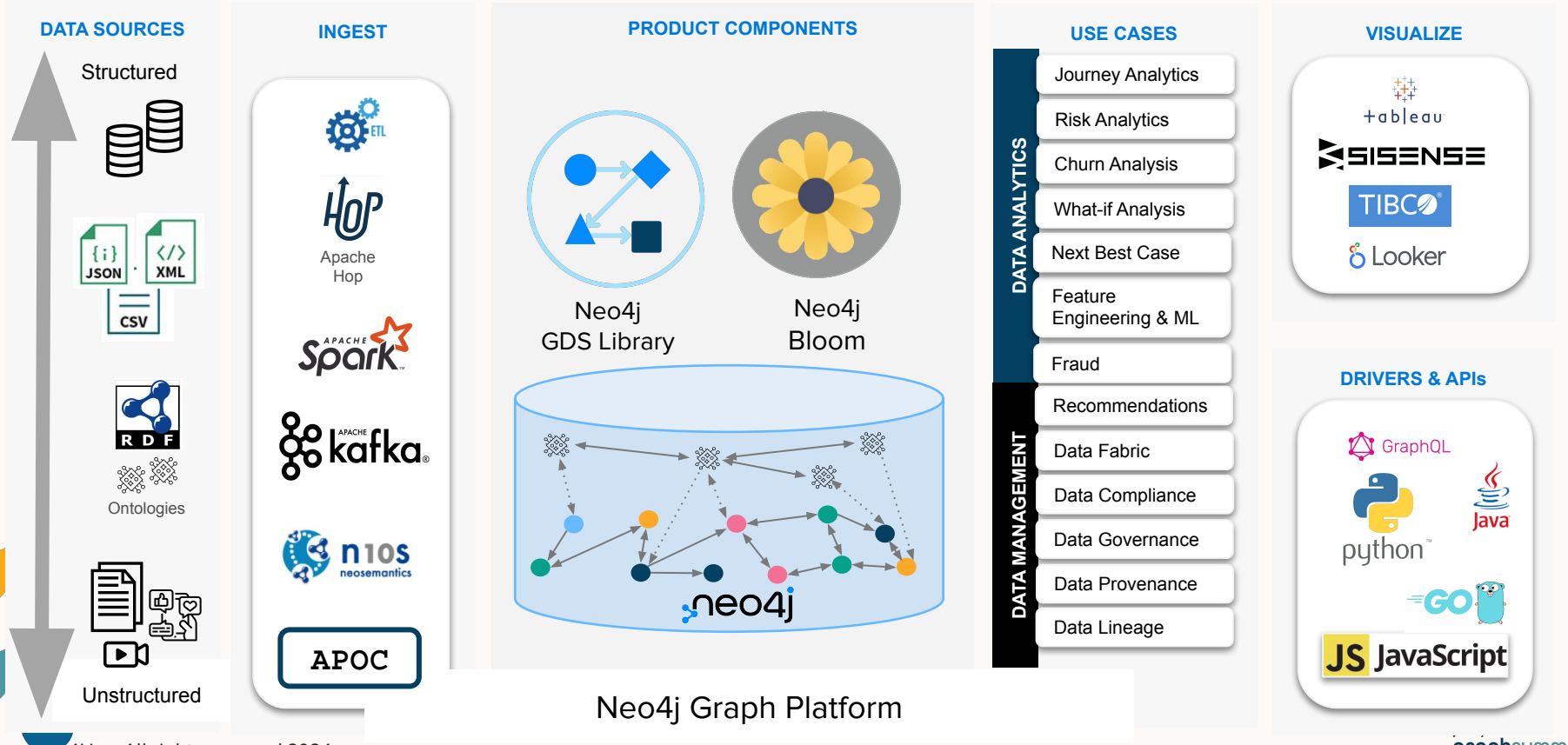
- Cypher Query Language vs. SQL (ISO -> GQL)
- Simple, lesser lines of code, easier to read and maintain
- Queries possible, that span 100+ Hops in the Graph, which is comparable with SQL Joins over 100+ Tables!

Storing complex data networks & semantics



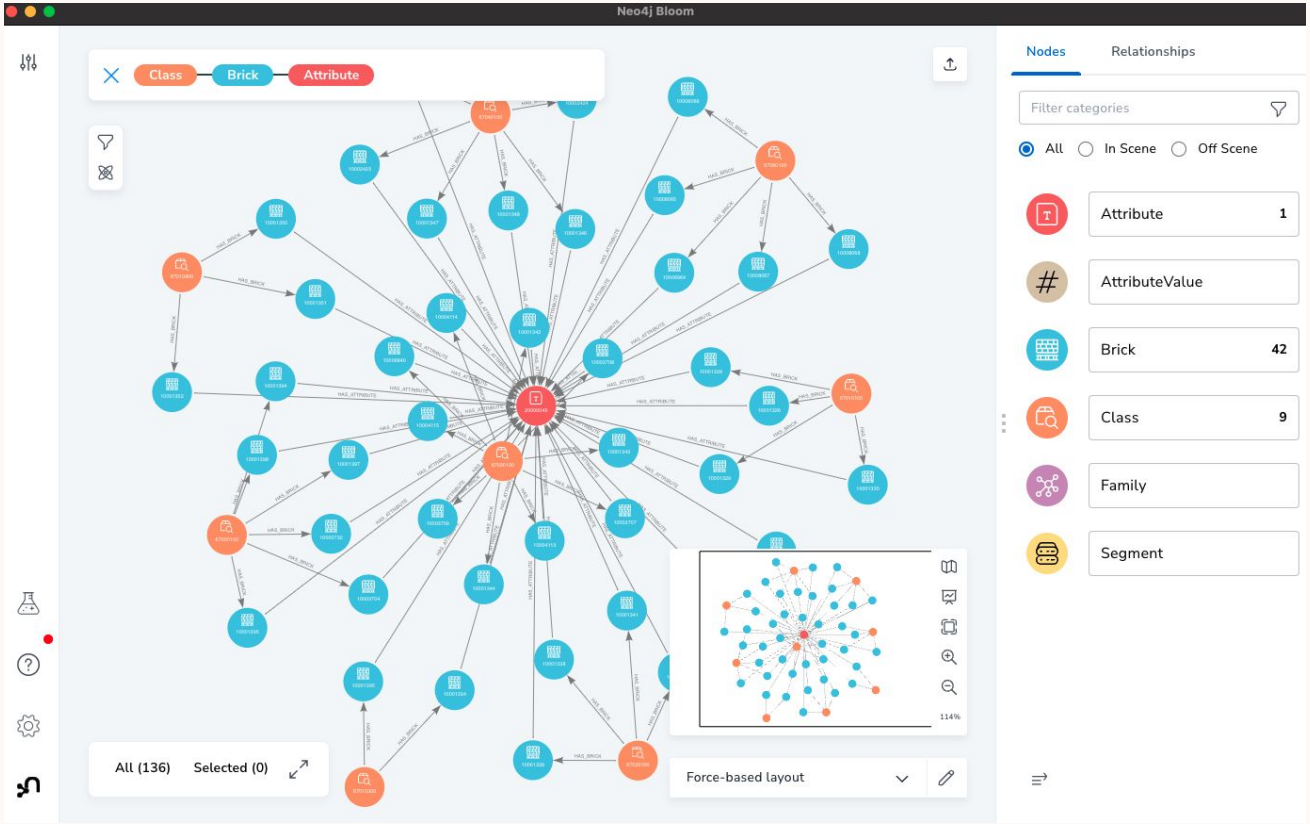
- Storing and analyzing complex relations between data
- Analyzing data from traditionally siloed data sources
- Extendable with data science algorithms and AI

Neo4j - Part of a Complete Ecosystem

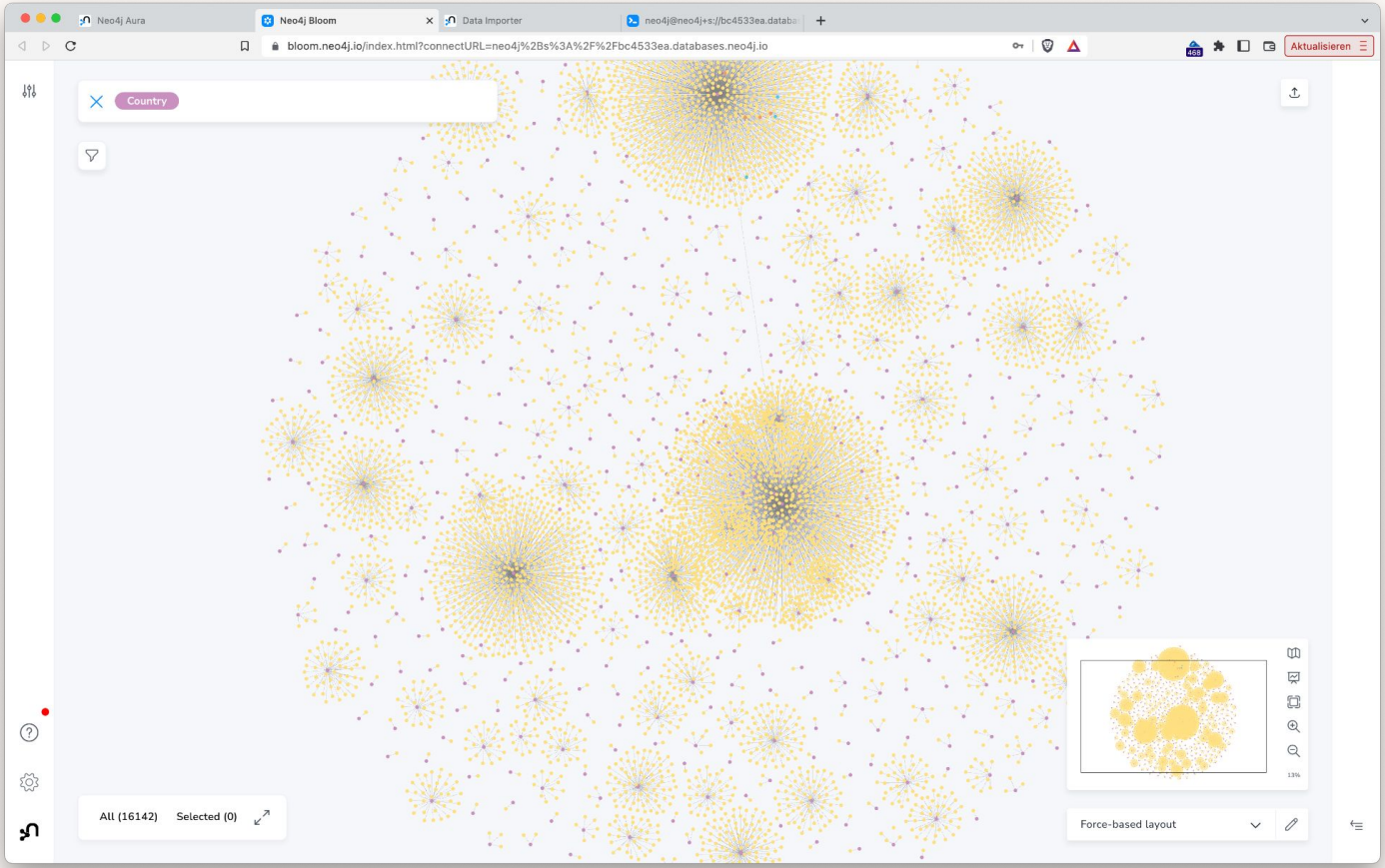


Neo4j Graph Platform

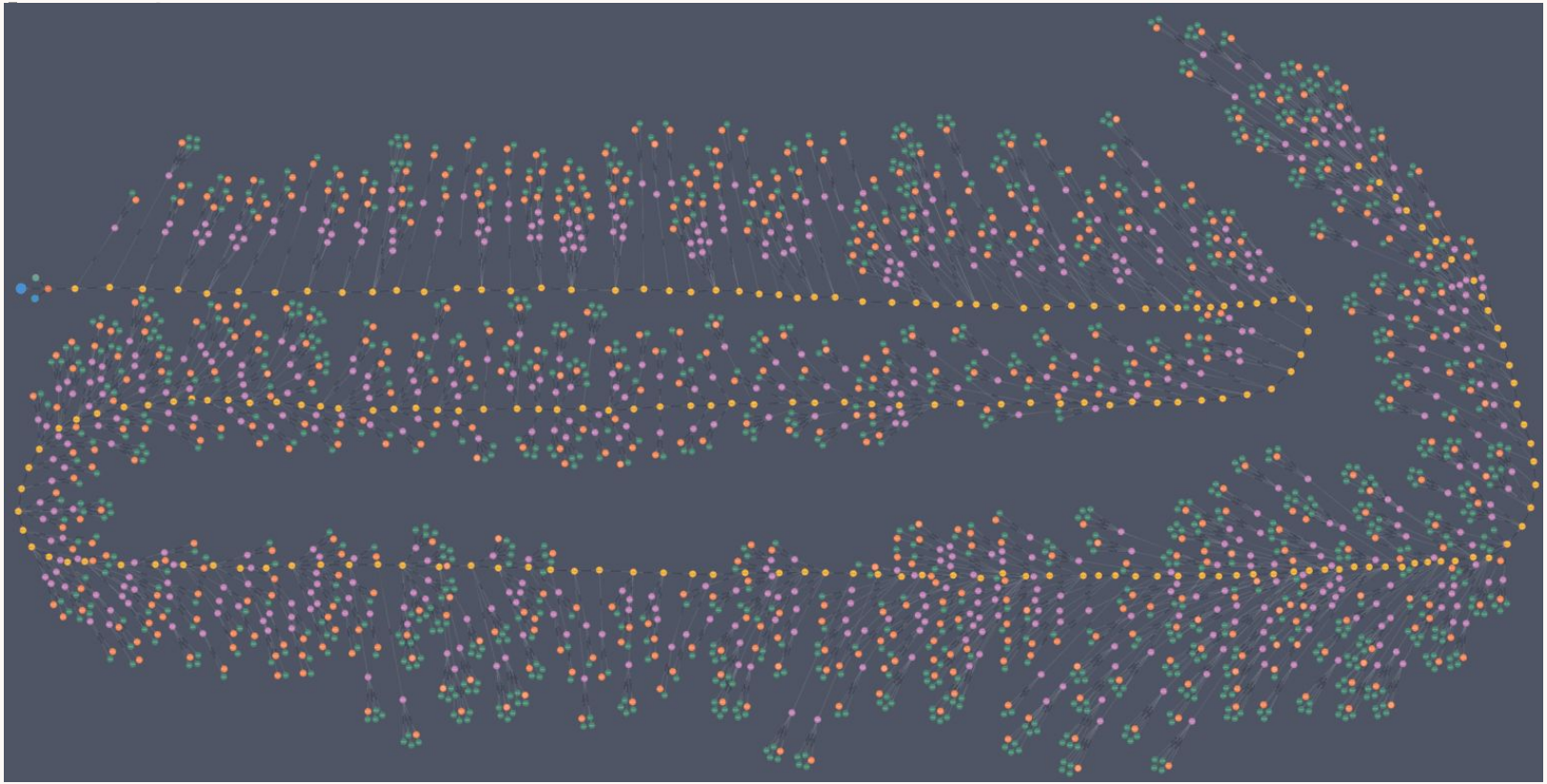
This is a Graph …



This is a Graph, as well...



...and this is also a Graph! Guess, what Graph could



* https://www.graphable.ai/blog/patient-journey-mapping/

Labeled Property Graph Model

Nodes

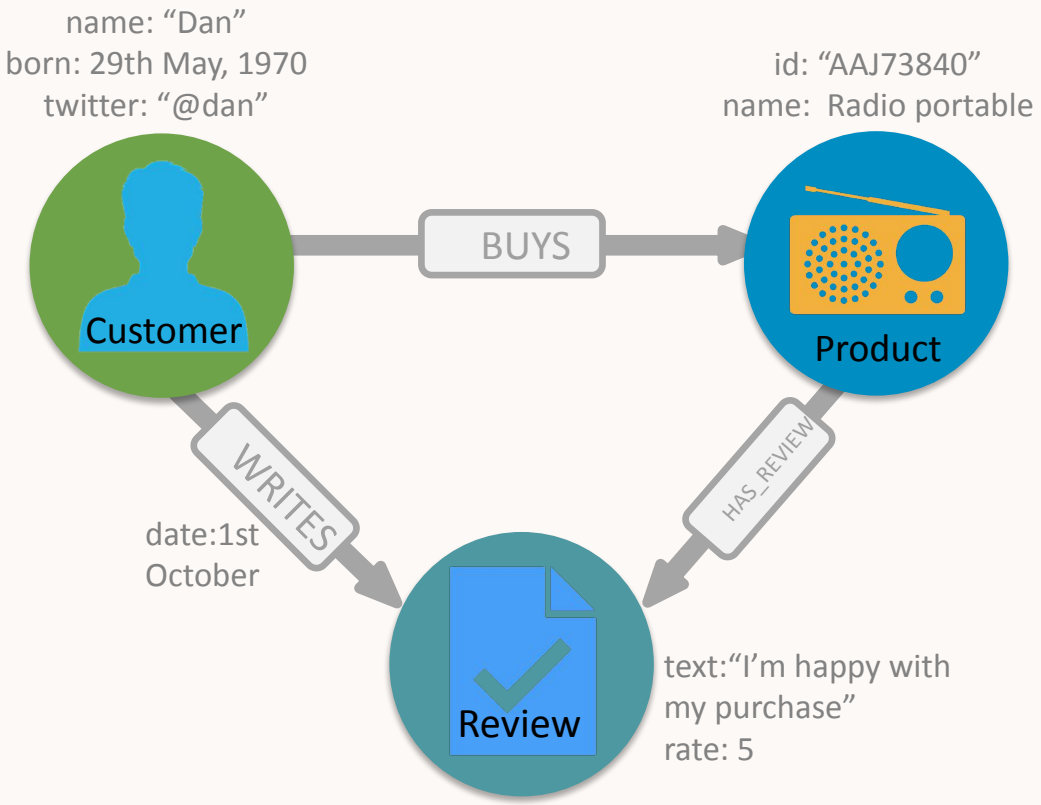
- Can have ***Labels*** to classify Nodes
- Labels have *native indexes*

Relationships

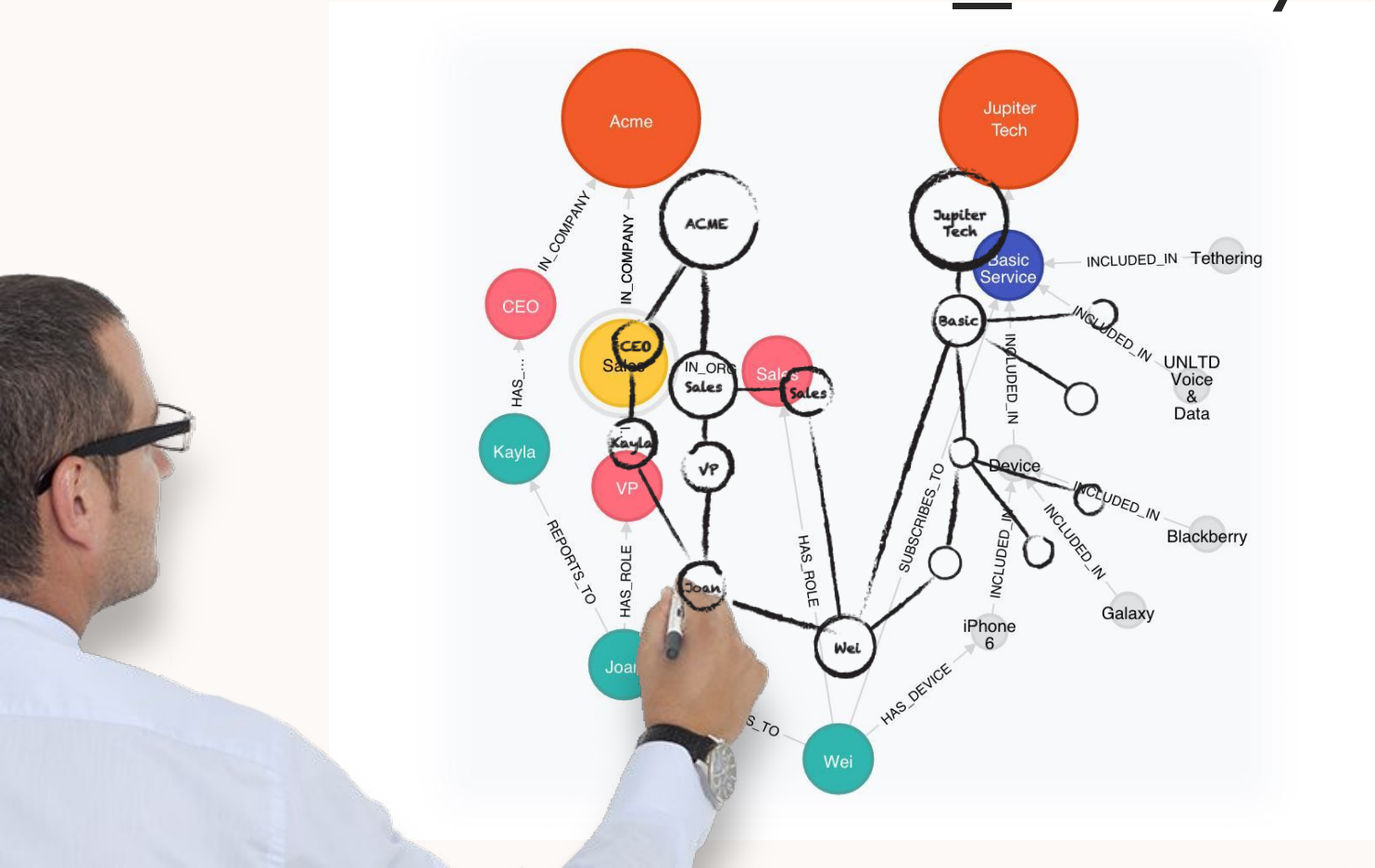
- Connect nodes by Type and Direction

Properties

- Attributes of Nodes & Relationships
- Stored as Name/Value pairs
- Can have indexes and composite indexes



The Whiteboard Model Is the Physical Model



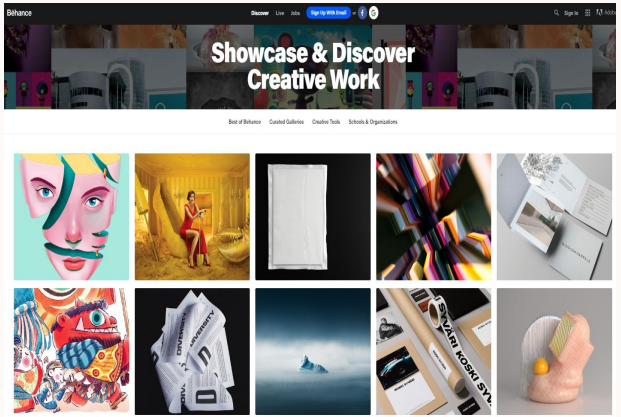
The Graph Problem Problem

Many organizations don’t realize that they have a graph problem

Adobe

Adobe Behance

Social Network of 10M
Graphic Artists



Background

- Social network of 10M graphic artists
- Peer-to-peer evaluation of art and works-in-progress
- Job sourcing site for creatives
- Massive, millions of updates (reads & writes) to Activity Feed
- *150 Mongos to 48 Cassandras to 3 Neo4j's!*

Business Problem

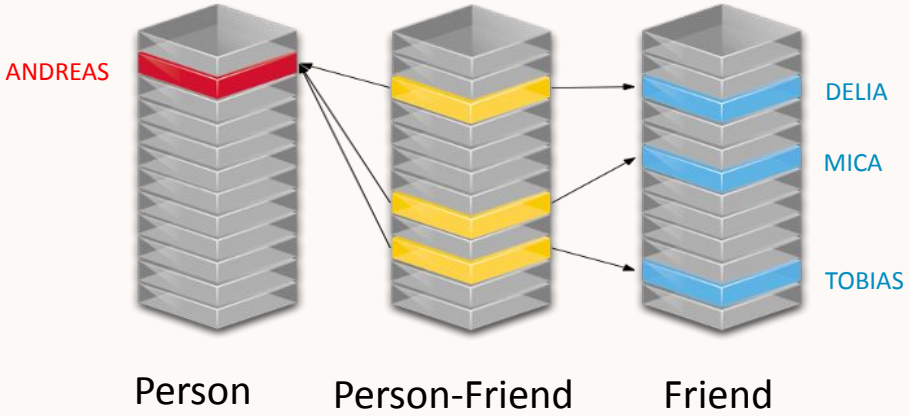
- Artists subscribe, appreciate and curate "galleries" of works of their own and from other artists
- Activities Feed is how everyone receives updates
- 1st implementation was 150 MongoDB instances
- 2nd implementation shrunk to 48 Cassandras, but it was still too slow and required heavy IT overhead

Solution and Benefits

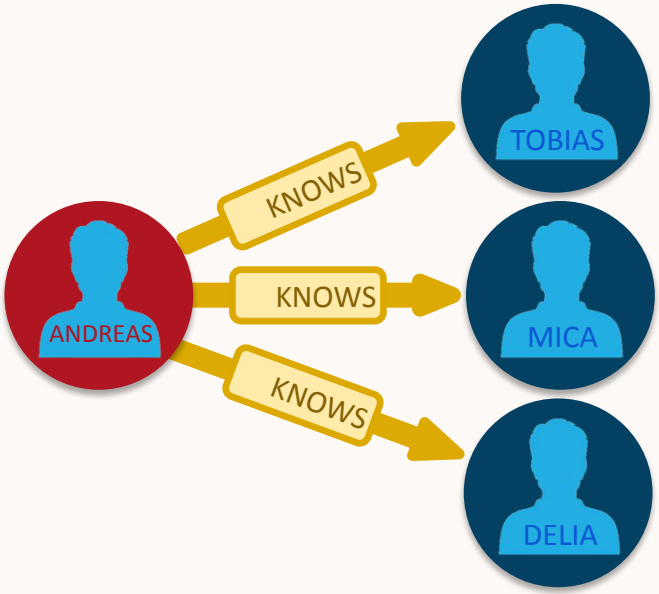
- 3rd implementation shrunk to 3 Neo4j instances
- Saved over $500k in annual AWS fees
- Reduced data footprint from 50TB to 40GB
- Significantly easier to introduce new features like, "New projects in your Network"

RDBMS vs Graph Model

RDBMS Model



Graph Model



Graph Data Science

Data science when ***relationships matter***

Queries
(Cypher)

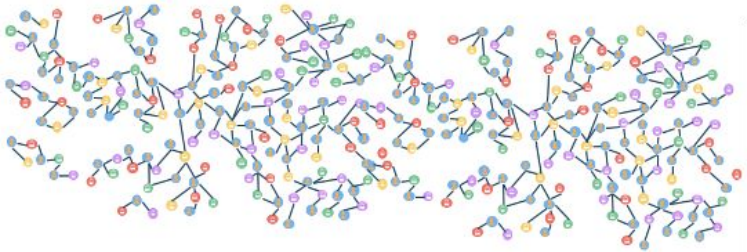
Fast, local decisioning and pattern matching

Graph Algorithms
(e.g. Neo4j graph data science library)

and iterations

Local Pattern ⟵ ⟶ **Global Computation**

You know what you're looking for and making a decision

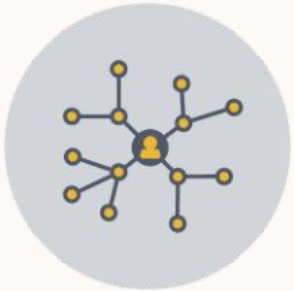


You're learning the overall structure of a network, updating data, and predicting

Graph Algorithms Verticals

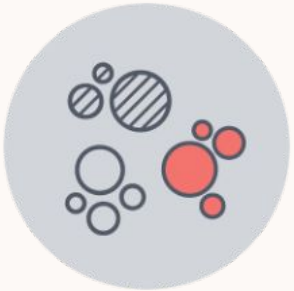
Pathfinding and Search

Finds optimal paths or evaluates route availability and quality.



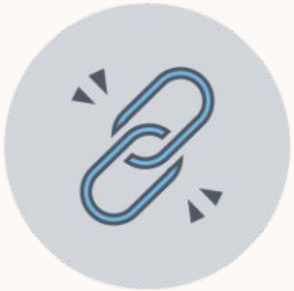
Centrality

Determines the importance of distinct nodes in the network.



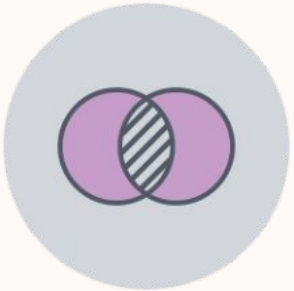
Community Detection

Detects group clustering or partition.



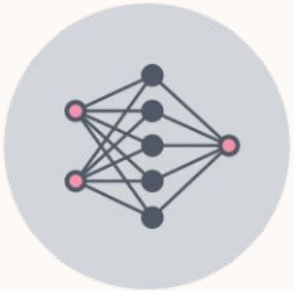
Heuristic Link Prediction

Estimates the likelihood of nodes forming a future relationship.



Similarity

Evaluates how alike nodes are by neighbors and relationships.



Embeddings

Learns graph topology to reduce dimensionality for ML

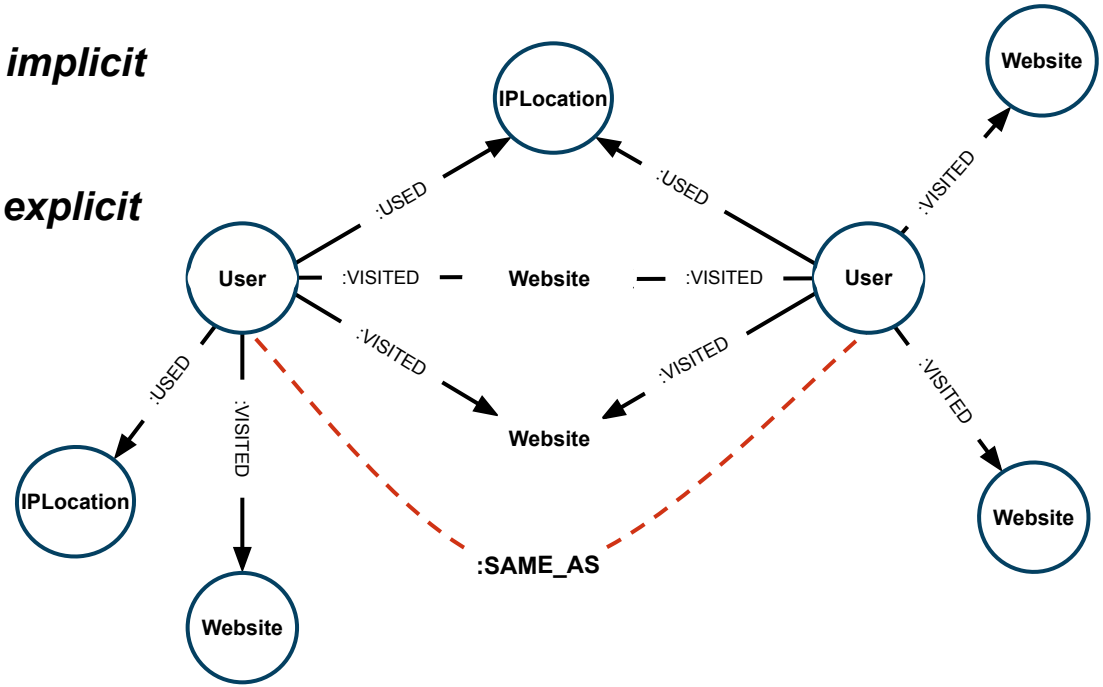
65+ graph algorithms in Neo4j

Knowledge Graphs

Graphs....Grow!

Graphs allows you to make implicit relationships....

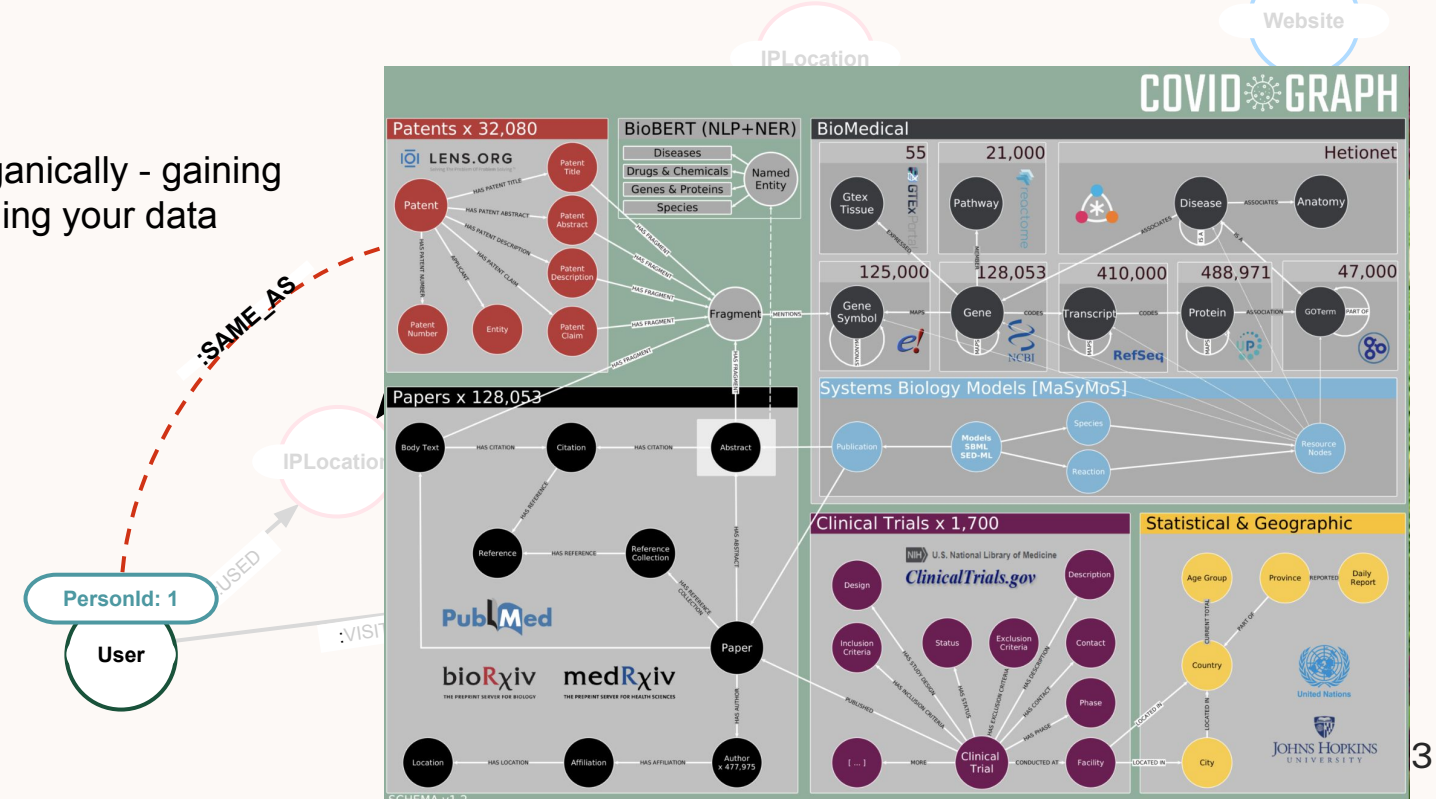
....explicit



Knowledge Graphs….Grow!

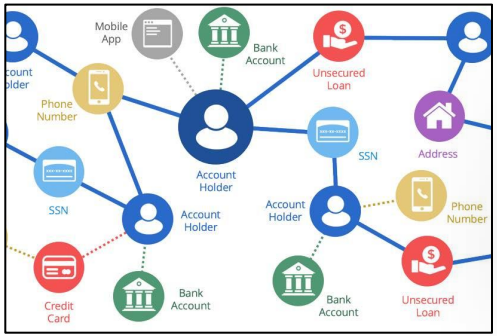
…and can then group similar nodes…and create a new graph from the explicit relationships…

A graph **grows** organically - gaining insights and enriching your data



What can you do with a knowledge graph?

Financial Domain

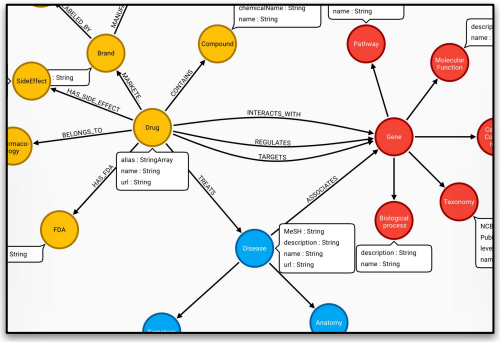


How many flagged accounts are in the applicant's network **4+ hops out**?

How many **login / account variables in common?**

Add these metrics to your approval process

Life Sciences

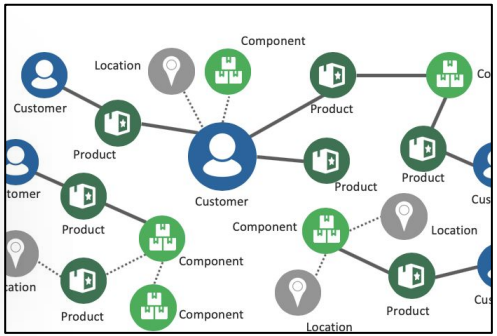


What **completes the connections** from genes to diseases to targets?

What genes can be reached **4+ hops out** from a known drug target?

What **mechanisms in common** are there between two drugs?

Marketing and Recommendations

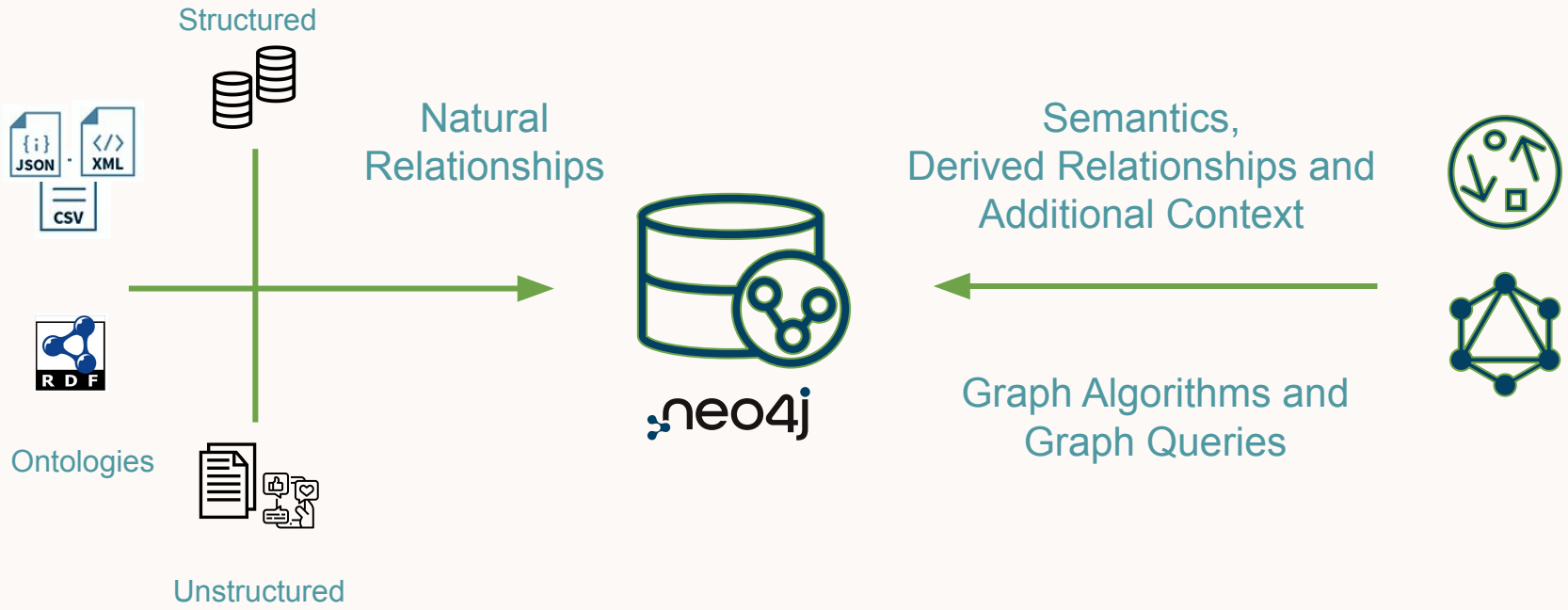


Collaborative filtering: users who bought X, also bought Y (**open-ended pattern matching**)

What items make you more likely to buy additional items **in subsequent transactions**?

Traverse hierarchies - what items are similar **4+ hops out**?

Knowledge Graphs



Large Language Models

...

Customer Operations

Marketing & Sales

LLMs save time and money

Software Engineering

R&D

75% of GenAI value will come from these four areas

The economic potential of generative AI: The next productivity frontier, McKinsey & Company, June 2023.

Outdated Public Sources

Hallucination

LLMs Challenges...

Probabilistic

Lack of Enterprise Domain Knowledge

Improve Accuracy and Specificity

Security and Privacy

Ground LLMs in Neo4j's Knowledge Graph

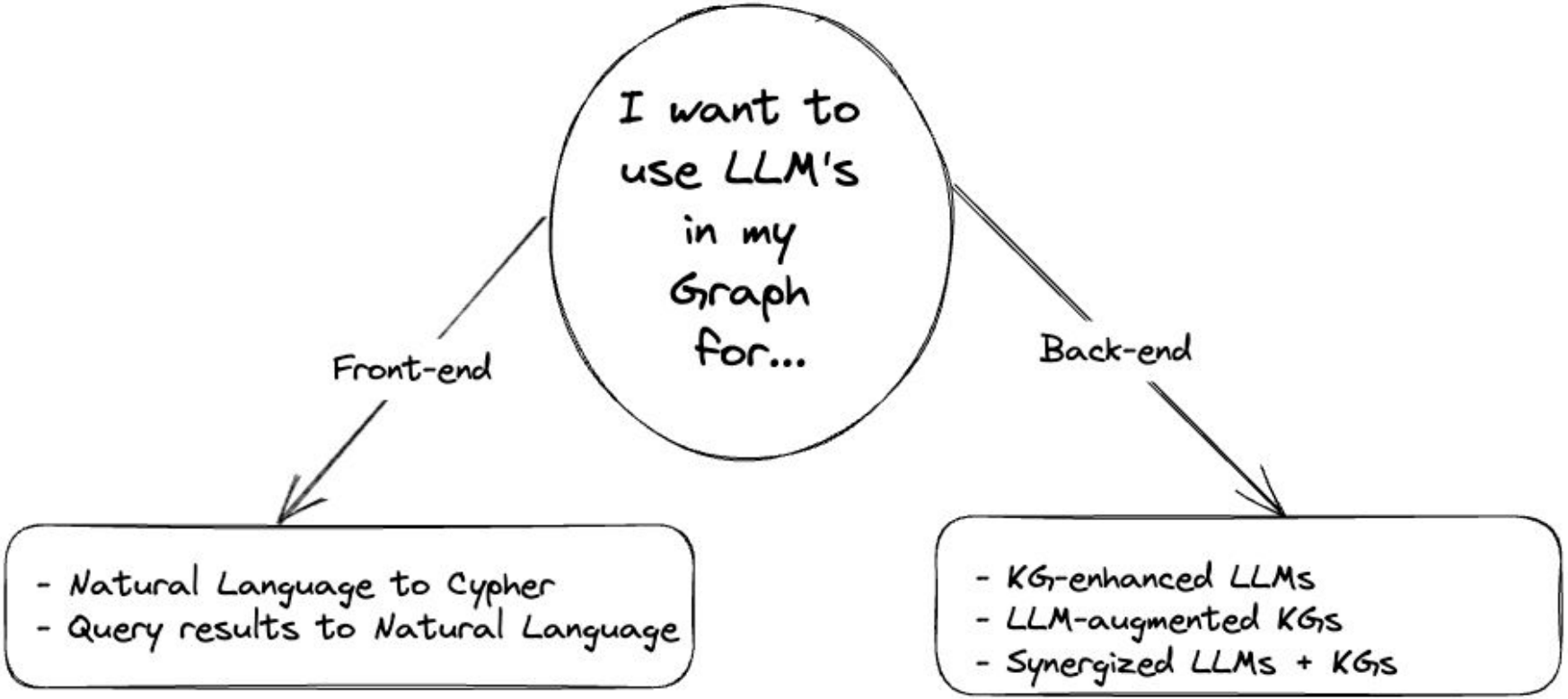
Probabilistic

Deterministic Facts

Reliability

Explainability

Setting the context - From a user point of view



About Indexes

Indexes are used to 'land' in the Graph as quickly as possible

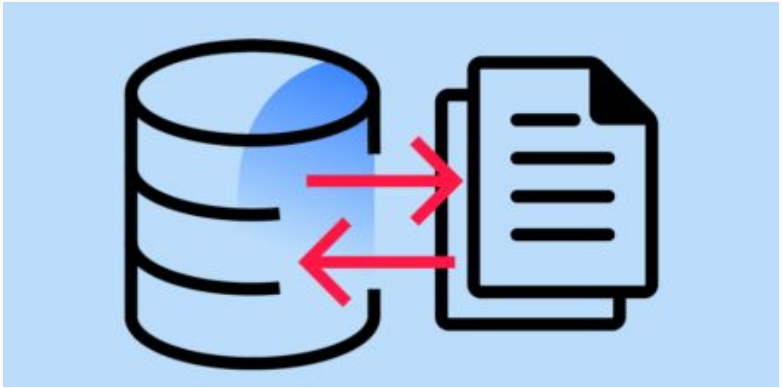
Vector Example

```
//
// Question input
//
with "Heb ik recht op een ov jaarkaart?" as vraag
//
// Question to embedding with OpenAI
//
WITH genai.vector.encode(vraag,"OpenAI",{token: $openaikey, model: $model } ) as embedding
//
// Vector search with the question-embedding as search parameter
//
CALL db.index.vector.queryNodes('tekst-embeddings', 5, embedding) YIELD node, score
//
// Now based on the nodes found in the vector index we can retrieve the context of the found nodes
//
MATCH p=(node)<-[*]-(:Wet)
RETURN score, node.tekst as text, p
```

Data Insertion & Loading

Data Insertion & Loading into Neo4j

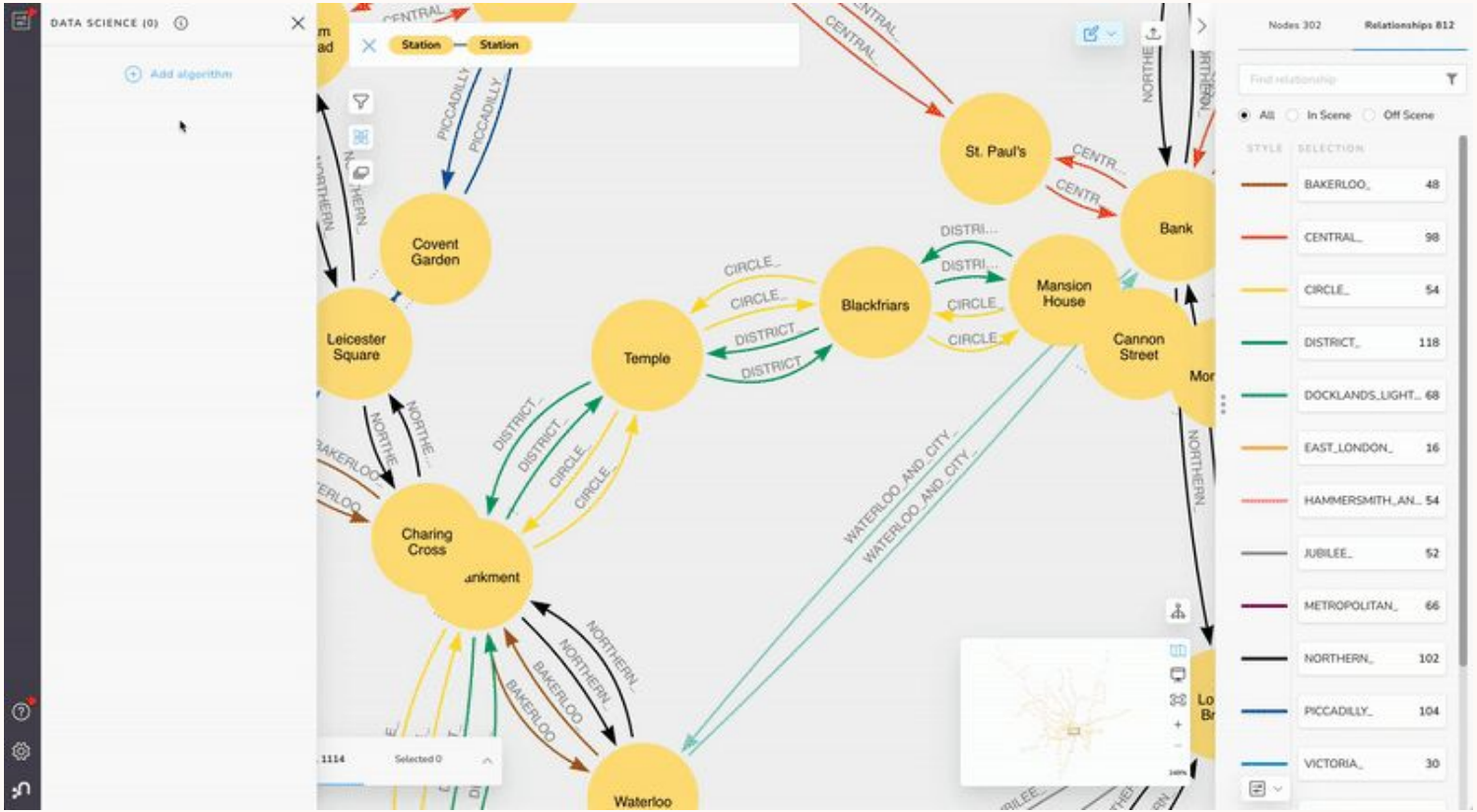
- **>>neo4j-admin** import Command
- Cypher CSV Loader
- APOC
- ETL Tools (Apache Hop,…)
- Driver Connections
- 3rd Party Platforms (Kafka, Spark)



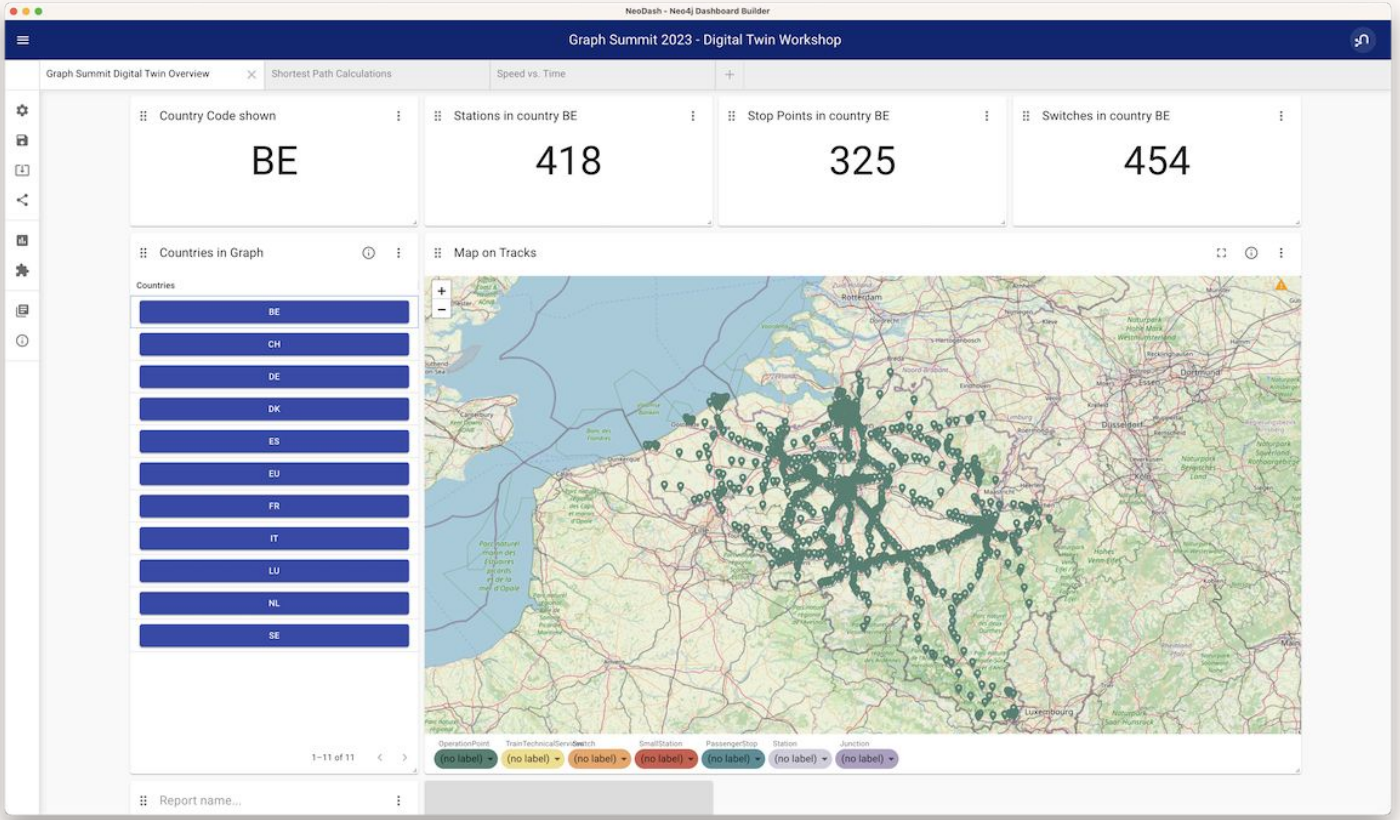
More about **Data loading, etc.** on the Neo4j Documentation

Data Visualization

Neo4j Bloom - Neo4j Visualisation Platform



NeoDash - Dashboarding with Your Graph Data



Use Case Explanation

Digital Twin - An Overview

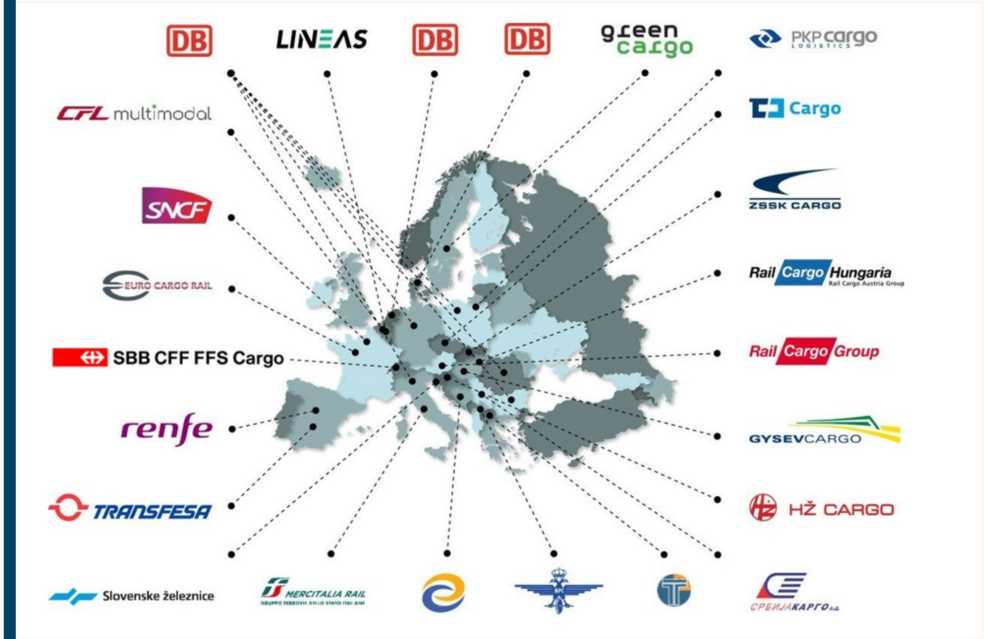
What is a Digital Twin?

A **Digital Twin** is a digital representation of an intended or actual real-world physical product, system, or process (a *physical twin*) that serves as the effectively indistinguishable digital counterpart of it for practical purposes, such as simulation, integration, testing, monitoring and maintenance.

* It has been done before

EU Railway Network - Track & Trace

- Challenge: **Legacy technology could not** track and analyze train journeys
- Solution: **Neo4j Knowledge Graph**
- **Identify and avoid bottlenecks (DB)**



Why do we need a Digital Twin? (few reasons)

Improved efficiency

It can optimize its operations and reduce costs by simulating different scenarios and making data-driven decisions.

Enhanced safety

It can identify potential hazards and test safety measures to improve safety for passengers and employees.

Predictive maintenance

It can monitor asset condition in real-time, predict maintenance needs, and increase asset lifespan.

Improved customer experience

A digital twin can simulate disruptions and help proactively address issues to enhance the customer experience and increase satisfaction.

Another reason why a digital twin might be helpful

https://www.euronews.com/travel/2023/02/21/unspeakable-botch-spain-spends-258-million-on-trains-that-are-too-big-for-its-tunnels

TRAVEL NEWS

‘Unspeakable botch’: Spain spends €258 million on trains that are too big for its tunnels

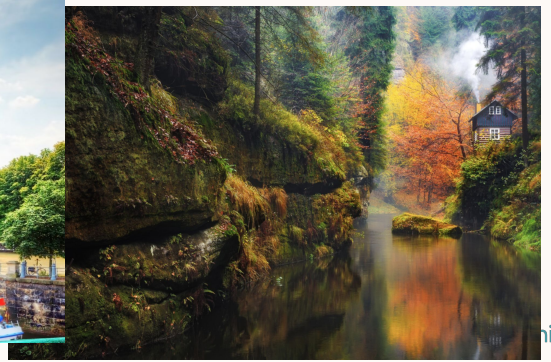
Two senior officials from Renfe and Adif have been dismissed over the error. – Copyright Canva

By **Angela Symons** • Updated: 21/02/2023

Spain has spent €258 million on **trains** that are too big to fit in its rail network’s tunnels.

After the blunder was exposed by local newspaper El Comercio in late January, two **transport** bosses were fired.

POI's Included



Unleash the power of the Graph

Abstracting from this Use Case

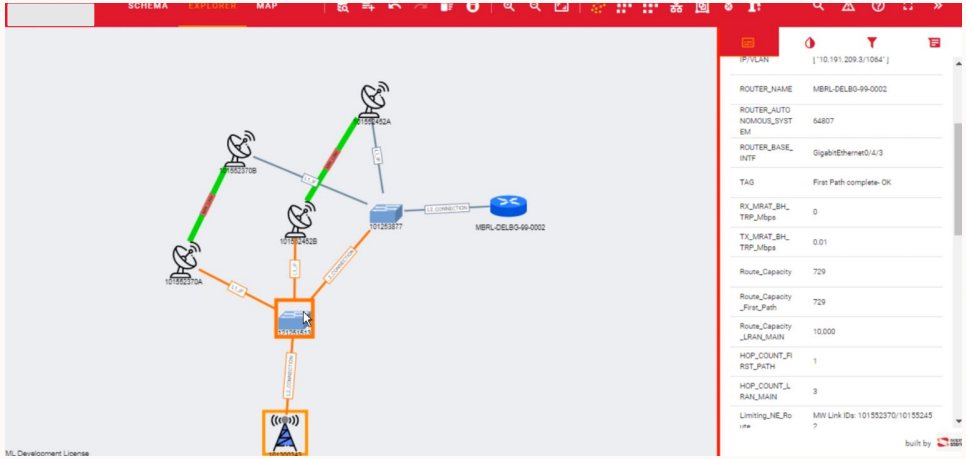
Digital Twin and Networks are everywhere

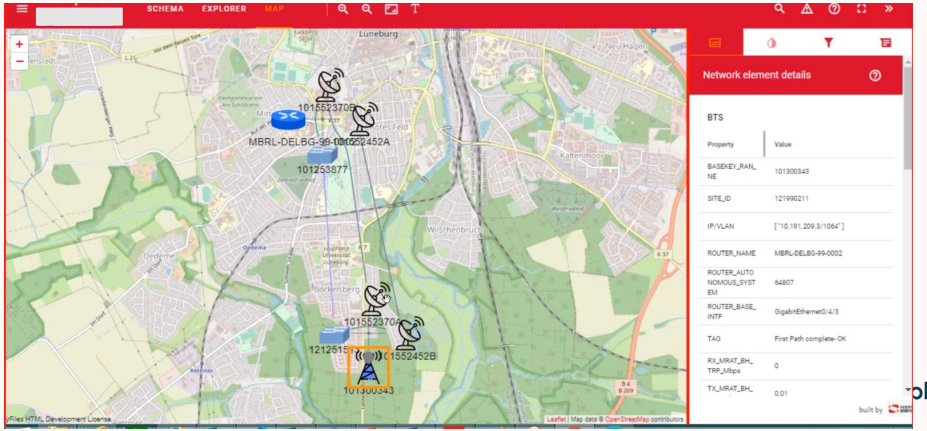
Many other use cases are networks, too!

- Supply Chain
- Mobile Phones Networks
- Power Grids / Gas Grids / Fibre Networks
- Social Networks
- Patient Journeys
- etc

Mobile Network Operations - Large Mobile Provider

- Digital Twin of Mobile Network
- Planning and operation
  - Low signal strength of tower
  - Moving equipment around
- Simulation of possible failure scenarios
- Repair & maintenance support
  - Supply Chain Impact





"We wanted to create a solution that exploits artificial intelligence, integrates current and historical AIS positions as well as multiple data feeds and APIs. Such an effort would require a world-class infrastructure. That's why we selected Neo4j."

David Levy
Chief Marketing Officer
OrbitMI

Customer Case Study: Logistics and Supply Chain

Plan maritime routes based on distances, costs, and internal logic.

Results:

- **Subsecond** maritime routes planning
- Reduce global carbon emissions

  60,000 tons
- **12-16M ROI** for OrbitMI customers

OrbitMI

The Sample Dataset

Sample Data: Railway Network Digital Twin Dataset

Provided Data:

- Consists of a Railway **Track** Network + **Operation Point (OP)**
- **It includes attributes like:**
  - Tracks called **Sections**,
  - Stations and other **OPs**,
  - Geo location information,
  - **Section** length and speed,
  - other parameters

- Add on from us: **Point of Interest** (POI) along tracks

- **Format**: CSV → Generated from XML
- Origin of Data: https://data-interop.era.europa.eu/search

The Data Explained - In an Easy Way

“The data can be seen as a highway that has ramps throughout its entire path.

Each ramp can be an **Operation Point** (OP) like a Station, a Switch, etc.

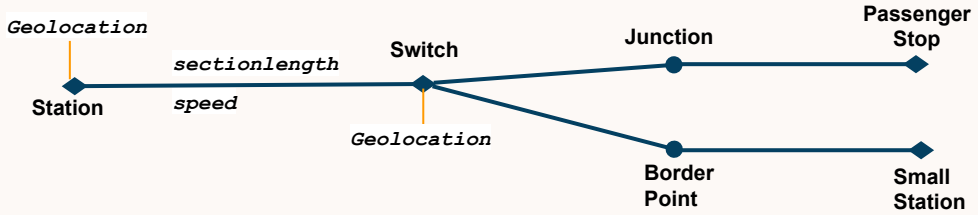
Tracks have a length and a speed associated. **Tracks have a start point and an endpoint and will be inter-connecting Operation Points**.

An Operation Point has a name and is referred to by a relationship with a country code.

Operation Points (OP) - Data Explanation (Nodes)

CSV Header titles:

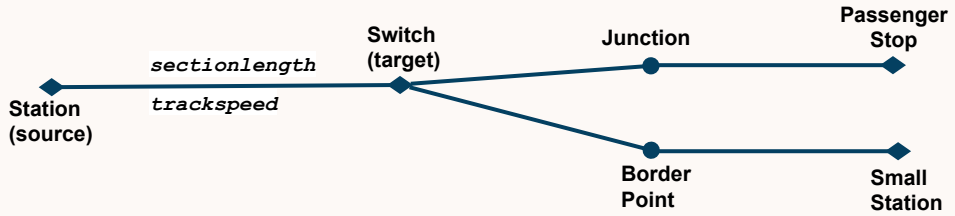
- **id**: the internal number of the OP
- **extralabel**: the kind of OP we deal with, e.g. Station, Junction, Switch, etc.
- **name**: the name of a OP
- **latitude**: of the OP
- **longtitude**: of the OP



Section Connection Data Explanation (Relationships)

CSV Header titles:

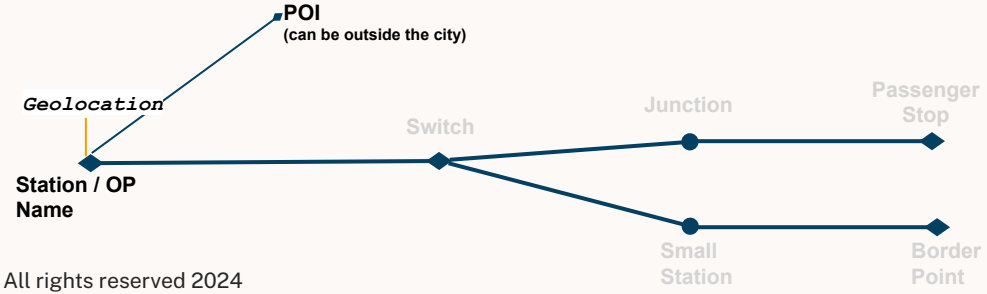
- **source**: start OP for this section
- **target**: end OP for this section
- **sectionlength**: the length in km of that section
- **trackspeed**: max speed allowed on that section



Point of Interest (POI) Data Explanation

CSV Header titles:

- **CITY:** City the POI is in or close by
- **POI_DESCRIPTION:** A short description of of the POI
- **LINK_FOTO:** a short name
- **LINK_WEBSITE:** is the name of the track corresponding to the shortcut
- **LAT:** Latitude of the POI
- **LONG:** Longitude of the POI
- **SECRET:** True if not well know, False if well known (e.g. Berlin)



Data Modeling - Writing down Questions & Answers

Data Modeling - The Questions are the Input

1. How long is the **journey** in KM from **station** X to station Y?
2. How long in KM is an alternative route if a station on my journey is closed?
3. Do I have alternative routes if a **station** in my journey is down?
4. What is the *shortest distance* in KM between **stations** X and Y?
5. Can I **geolocate** stations in my static rail network?
6. What are the stations expecting the most traffic?
7. What **POI**s are along my route?

Note: Aggregations are also possible, but we focus on the “graphy” queries here!

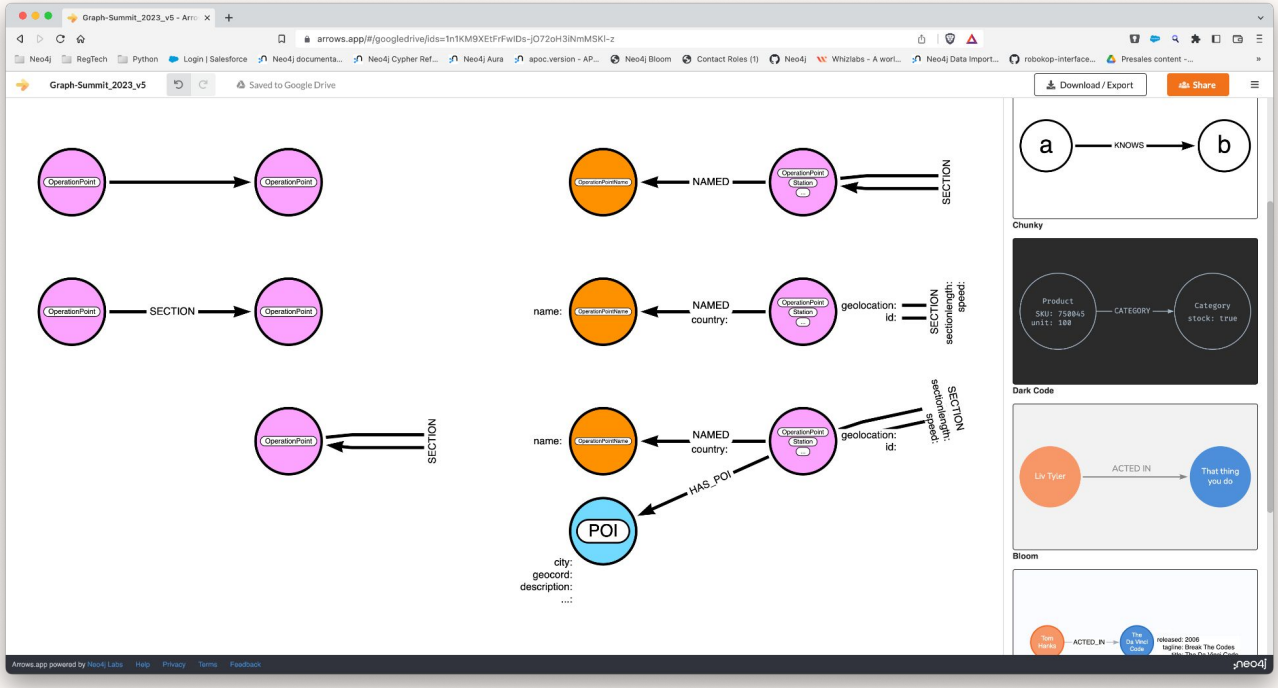
Expected Answers

1. Your journey is 30 stations and 239km long
2. Your alternative route is 36 stations and 271km long
3. The shortest route between X and Y is XXX km
4. The geo location of your station is Lat / Long
5. The station with the most traffic is … ?
6. The following POIs are along your route …

Data Modeling
Building the Model

Tools for Graph Data Modeling

Arrows Tool → https://arrows.app



Data Modeling Using the Questions as Input

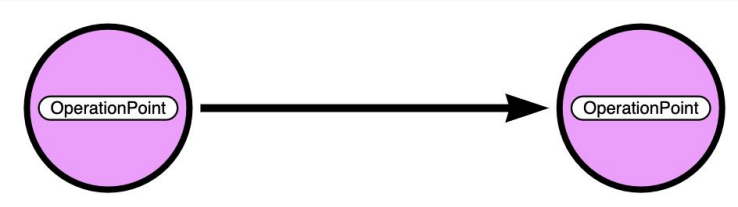
Remember those questions?

1. How long is the **journey** in KM from **station** X to station Y?
2. How long in KM is an alternative route if a station on my journey is closed?
3. Do I have alternative routes if a **station** in my journey is down?
4. What is the *shortest distance* in KM between **stations** X and Y?
5. Can I **geolocate** stations in my static rail network?
6. What are the stations expecting the most traffic?
7. What **POI**s are along my route?

Blue → Nodes / Properties? **Orange** → Relationship Types

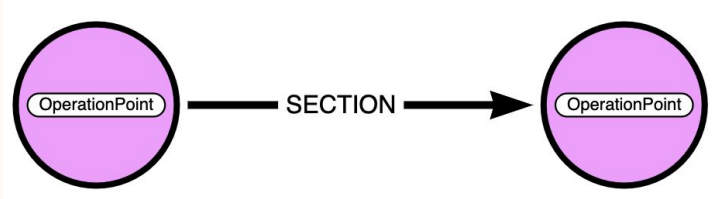
Building a First Data Model 1/6

- Nouns in your questions are the nodes, verbs are the relationships
- Nodes have a **Label**, Relationships MUST have a **Type**
- Properties should help to uniquely identify Nodes and/or answer questions



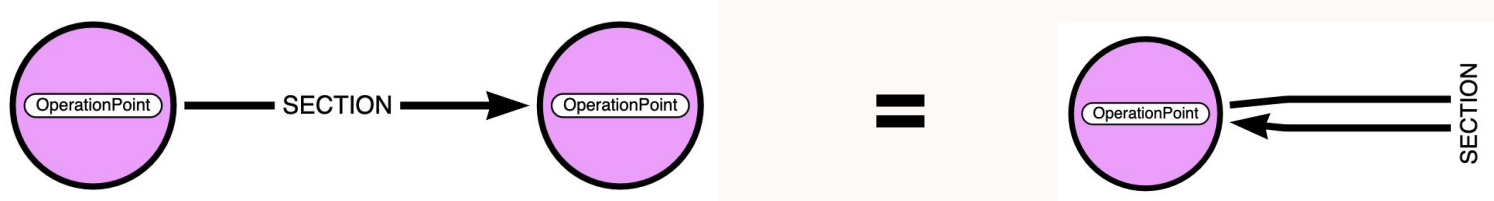
Building a First Data Model 2/6

- Nouns in your questions are the nodes, verbs are the relationships
- Nodes have a **Label**, Relationships MUST have a **Type**
- Properties should help to uniquely identify Nodes and/or answer questions



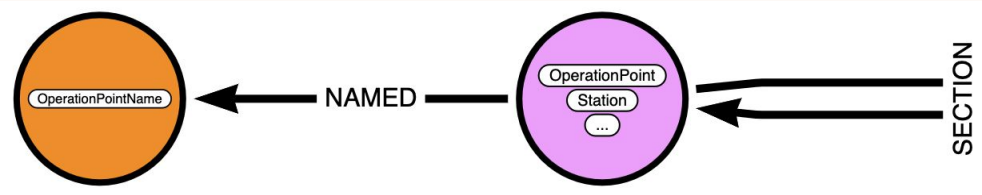
Building a First Data Model 3/6

- Nouns in your questions are the nodes, verbs are the relationships
- Nodes have a **Label**, Relationships MUST have a **Type**
- Properties should help to uniquely identify Nodes and/or answer questions



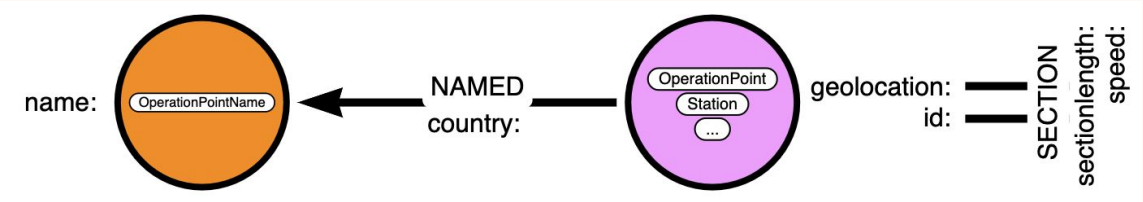
Building a First Data Model 4/6

- Nouns in your questions are the nodes, verbs are the relationships
- Nodes have a **Label**, Relationships MUST have a **Type**
- Properties should help to uniquely identify Nodes and/or answer questions



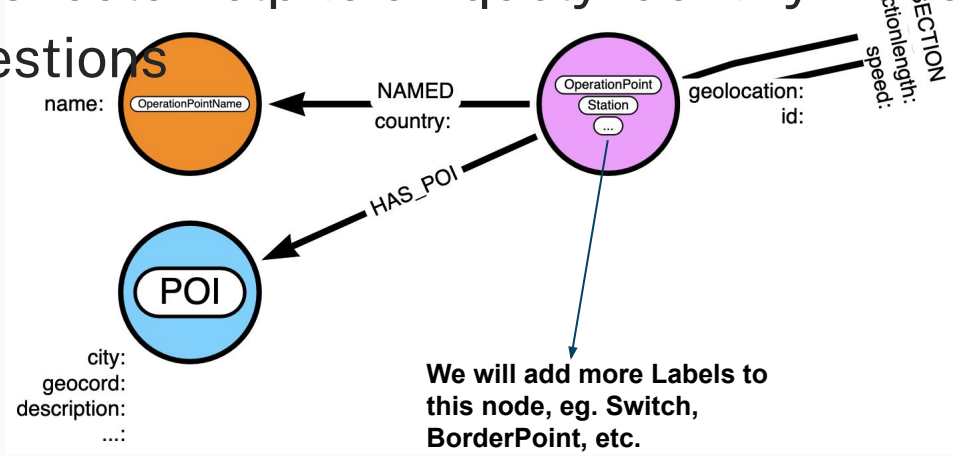
Building a First Data Model 5/6

- Nouns in your questions are the nodes, verbs are the relationships
- Nodes have a **Label**, Relationships MUST have a **Type**
- Properties should help to uniquely identify Nodes and/or answer questions



Building a First Data Model 6/6

- Nouns in your questions are the nodes, verbs are the relationships
- Nodes have a **Label**, Relationships MUST have a **Type**
- Properties should help to uniquely identify Nodes and/or answer questions



Workshop Time

Let’s build the solution together …

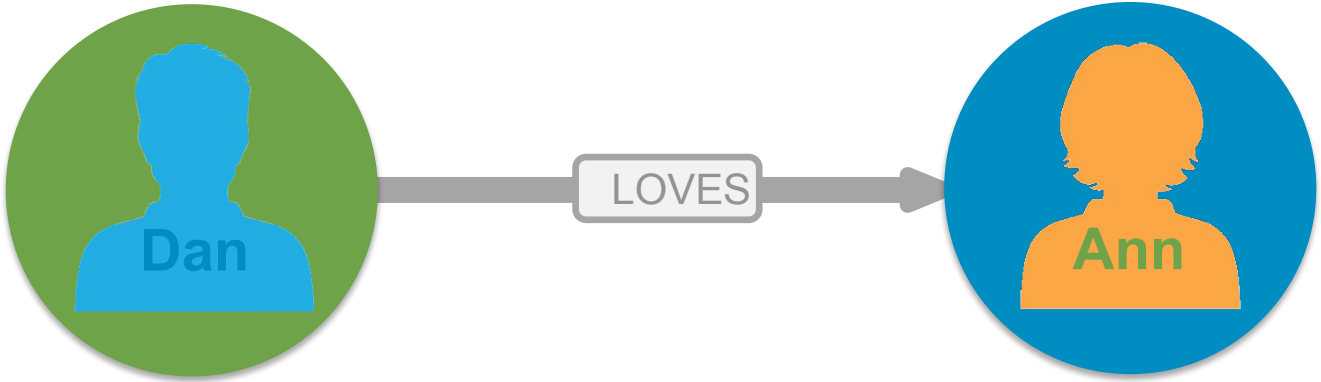
Let's Do it Together ...

1. Create a Neo4j Graph instance via any of: https://console.neo4j.io/
    a. Neo4j Aura
        i. You can spin up a Free aura instance here. If you already have one Aura Free db in use and don't want to clear it can use the following option for this workshop
    b. Neo4j Sandbox use a "Blank Sandbox"
    c. Neo4j Desktop
        i. If you are using Neo4j Desktop, you will need to ensure that APOC is added to any graph you create. Installation instructions can be found here.

2. Open the Github page at: https://github.com/kvegter/gsummit2024
    a. `load-all-data.cypher`

(Basic) Cypher Hands-On

Cypher: Powerful and Expressive Query Language



NODE RELATIONSHIP NODE

```
CREATE (:Person { name:“Dan”} ) -[:LOVES]-> (:Person { name:“Ann”} )
```

LABEL PROPERTY LABEL PROPERTY

MATCH <PATTERN> RETURN <DATA>

Cypher Check Up

```
// What is this Query Doing?
MATCH (a:Person)-[:FRIEND_OF]->(b:Person)<-[:FRIEND_OF]-(c:Person)
WHERE a.name = “Kees Vegter”
AND c.name = “Marco De Luca“
RETURN b.name as Name, b.id as FriendID
```

Cypher Hands-on Questions

```
// Find the 50 Operation Points, Ordered By Neo4j internal id’s
```

Cypher Hands-on Questions

```
// Find the 50 Operation Points, Ordered By Neo4j internal id's
MATCH (op:OperationalPoint)
RETURN op
ORDER BY id(op)
LIMIT 50;
```

Cypher Hands-on Questions

```
// Find the 50 Operation Points, Ordered By Neo4j internal id’s
MATCH (op:OperationalPoint)
RETURN op
ORDER BY id(op)
LIMIT 50;

// Find All Operation Points Names in the city of Amsterdam
```

Cypher Hands-on Questions

```
// Find the 50 Operation Points, Ordered By Neo4j internal id's
MATCH (op:OperationalPoint)
RETURN op
ORDER BY id(op)
LIMIT 50;

// Find All Operation Points Names in the city of Amsterdam
MATCH (op:OperationalPointName)
WHERE op.name CONTAINS 'Amsterdam'
RETURN op.name;
```

Cypher Hands-on Questions

```
// Find the Station id's for Berlin and Paris
// HINT: Use the pattern (OperationalPointName)<-[:NAMED]-(Station)
```

Cypher Hands-on Questions

```
// Find the Station id's for Berlin and Amsterdam
// HINT: Use the pattern (OperationalPointName)<-[:NAMED]-(Station)


MATCH (n:OperationalPointName)<-[r:NAMED]-(station:Station)
WHERE n.name STARTS WITH 'Berlin'
RETURN n.name, station.id

MATCH (n:OperationalPointName)<-[r:NAMED]-(station:Station)
WHERE n.name STARTS WITH 'Amsterdam'
RETURN n.name, station.id
```

Cypher Hands-on Questions

```
// Find the Shortest path between Frankfurt and Amsterdam
(Frankfurt-id: DE000FF, Amsterdam-id: NLASD)
TIP: use shortestPath

MATCH (op1:OperationalPoint WHERE op1.id = 'NLASD')
MATCH (op2:OperationalPoint WHERE op2.id = 'DE000FF')
MATCH sp=shortestPath((op1)-[:SECTION*]-(op2))
RETURN sp;
```

Cypher Hands-on Questions

```
// Find the Shortest path between Amsterdam (POI) and Berlin (POI)


MATCH(a:POI), (b:POI)
WHERE toLower(a.city) CONTAINS "amsterdam"
AND toLower(b.city) CONTAINS "berlin"
WITH a, b
MATCH p = shortestPath((a)-[*]-(b))
RETURN p
```

Build your own Graph Solution

with NeoDash

Go to the github page here and follow instructions:

How Could this “Mini Digital Twin” be Enhanced?

Extending the Graph is easy, e.g. with further track information:

- For example with possible **pace on track**, **track quality metrics**, **mobile network coverage**, etc.
- Sensor information about switches to quickly find problems and re-route traffic
- The more related information is added, the more use cases could benefit from the digital twin

Going even further:

- Additional information about **trains running on the tracks** enable additional benefit:
  - Track and schedule optimization
  - Simulation of new track (to grow traffic)

Q & A

Thanks for Graphing with us!

Contact us:
Kees.Vegter@neo4j.com