Graph Databases and Graph Visualizations

Ben Doan
March 24, 2021

About Me

That's Me!

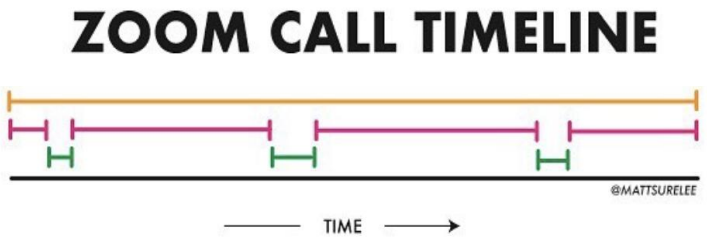
- Data Scientist and Full Stack Developer at Greenzone (Ambit Group)
- Working with distributed computing infrastructure for big data analytics and **graph databases**
- Projects include anti-human trafficking and child exploitation counter-intelligence
- McIntire Class of 2017, Block 2

Email: kdoan@greenzoneinc.com || knd9uz@virginia.edu

What are Graphs?

The term “graph” can refer to one of two things

Pictorial representation or a diagram that represents data or values in an organized manner.

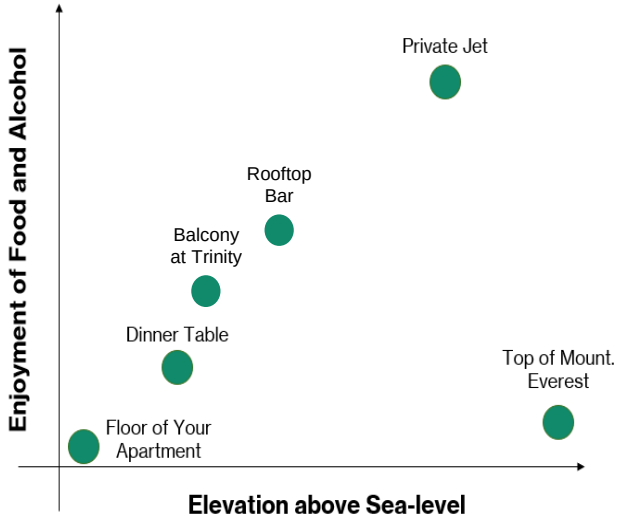


Me talking

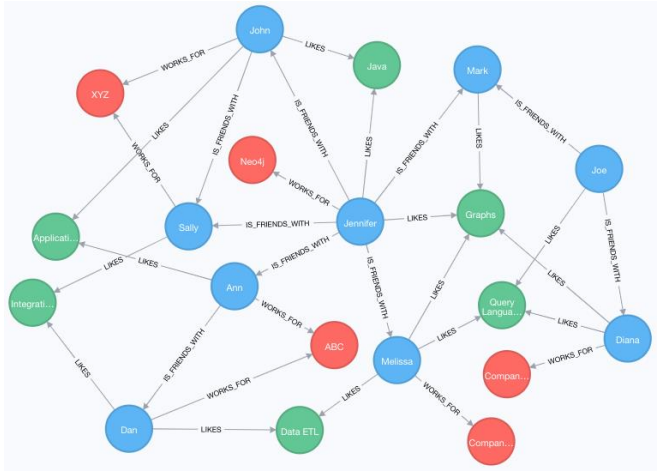
Them talking

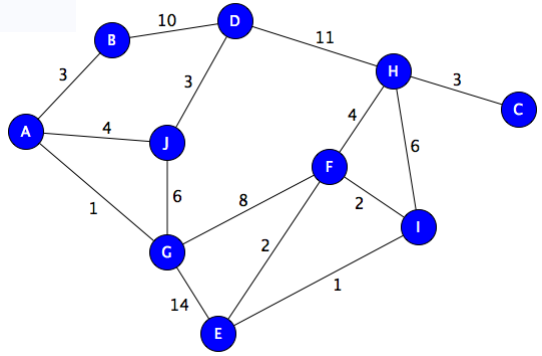
Me staring at my own stupid face

Source: @mattsurelee



Mathematical structures made up of *vertices* (*nodes or points*) which are connected by *edges* (*links or edges*).





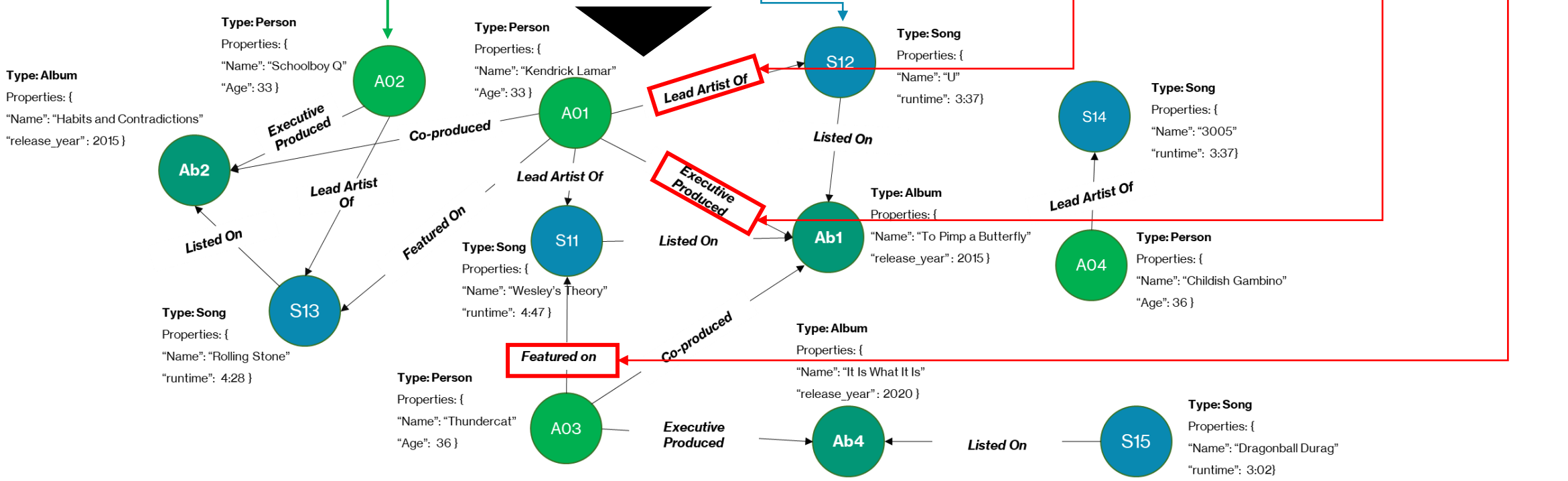
Source: Neo4j

What are Graph Databases?

A graph database is a database that uses a **vertex and edge** structure for data representation and querying, instead of the traditional **row and column format.** This is done by converting the **rows in a table to vertexes** and certain **key columns to edges**

Artist_id	name	age	record_label_id (FK)
A01	Kendrick Lamar	33	Top Dawg Entertainment
A02	Schoolboy Q	34	Top Dawg Entertainment
A03	Thundercat	36	Universal Music
A04	Childish Gambino	37	Universal Music

Song_id	Song_name	runtime	release_date	artist_id (FK)	album_name (FK)	produced_by (FK)	features (FK)
S11	Wesley's Theory	4:47	3/15/2015	A01	To Pimp a Butterfly	[A01, A03]	[A03]
S12	U	4:28	3/15/2015	A01	To Pimp a Butterfly	[A01]	
S13	Rolling Stone	3:37	1/11/2011	A02	Habits and Contradictions	[A02]	[A01]
S15	Dragonball Durag	3:02	2/17/2020	A03	It Is What It Is	[A03]	



Why Use a Graph?

Traditional relational database structures (RDBs) or RDB-like structures are generally rigid, which is good for strict control over entities and attributes, but **bad** for modeling links or relationships

RDB ER Diagram: Music Labels & Artists

Song table

Song_id	Song_name	artist_id	runtime	album_id
S11	Wesley's Theory	A01	4:47	Ab1
S12	U	A01	4:28	Ab1
S13	Rolling Stone	A02	3:37	Ab2
S14	3005	A04	3:54	Ab3
S15	Dragonball Durag	A03	3:02	Ab4

Album Table

id	name	release_year	artist_id
Ab1	To Pimp a Butterfly	2015	A01
Ab2	Habits and Contradictions	2012	A02
Ab3	Because the Internet	2013	A04
Ab4	It Is What It Is	2020	A03

Entities and Things

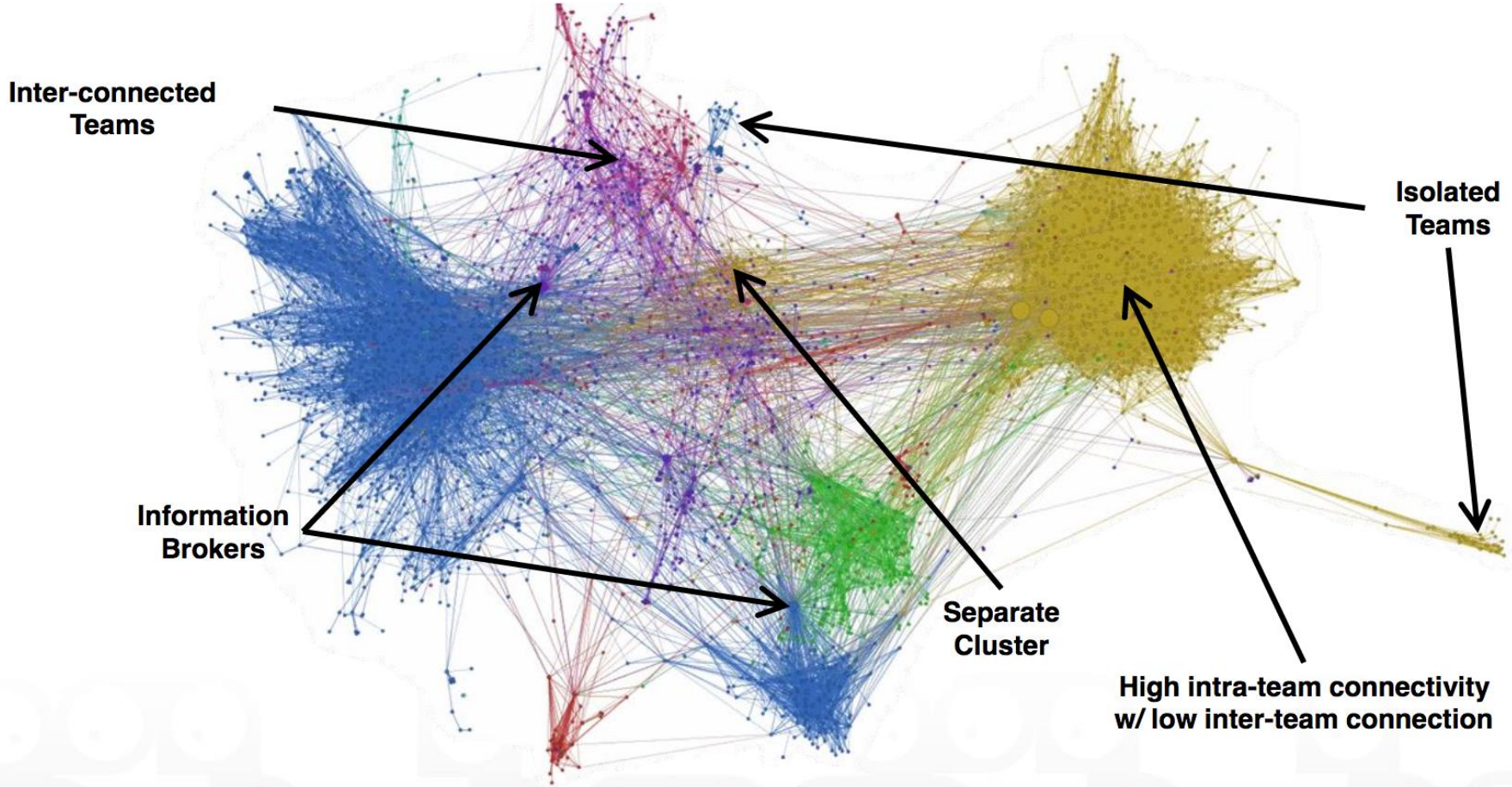
- In RDBs, entities or "things" are represented as rows in tables
- Characteristics of "things" are columns represented as columns
- Types of "things" are captured in the name of the table

Relationships between Entities

- Relationships between entities or "things" are represented by **foreign keys**
- This makes RDBs more **rigid** when it comes to modeling the links or relationships between things, and it makes visualizing the relationships between entities more difficult

Why Use a Graph?

Whereas traditional row and column data is good for row/column-wise aggregations, graphs are great when it comes to visualizing and analyzing **networks and relationships within and between datasets**

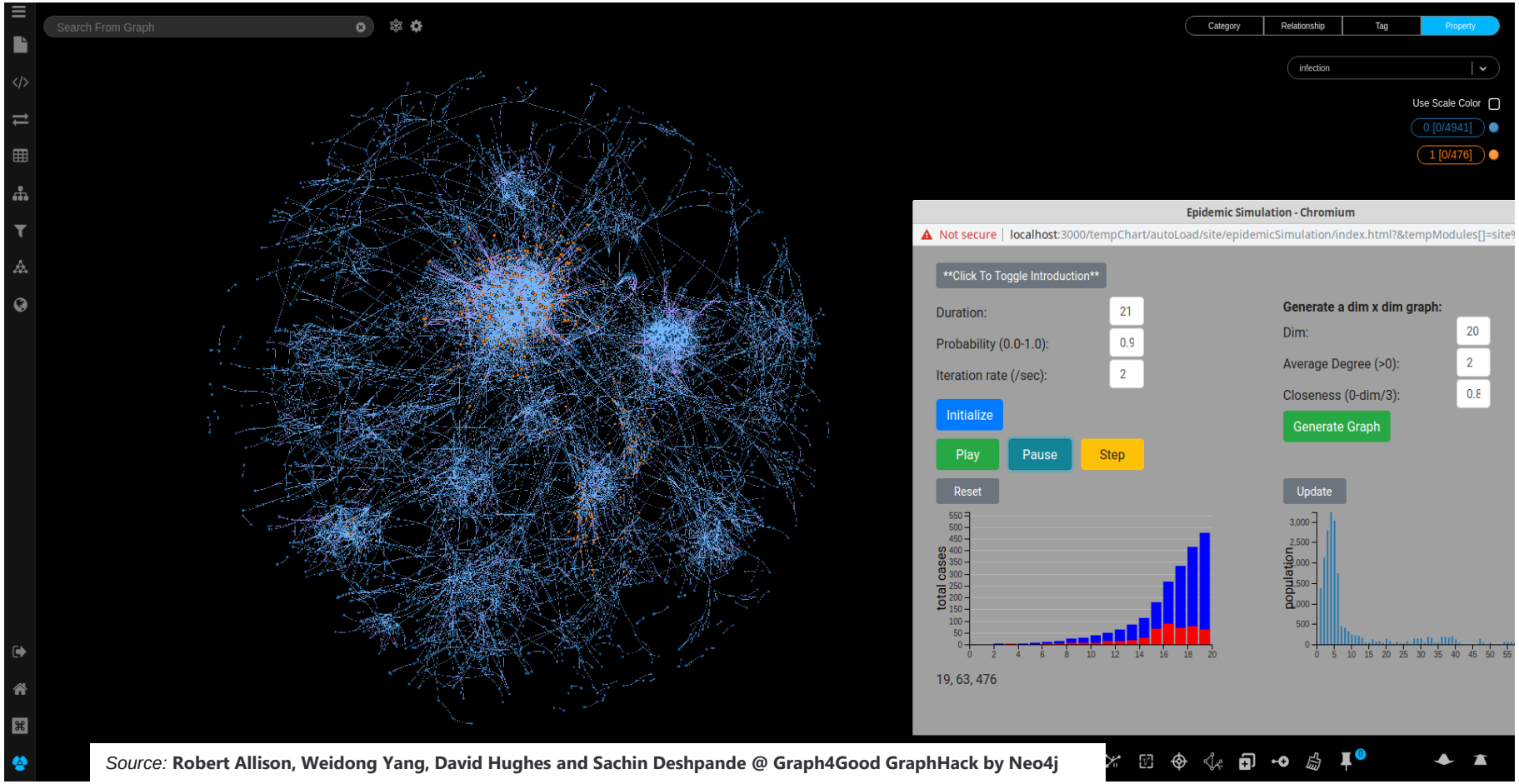


IBM Team Collaboration Graph shows the social network analysis of its team's code commits and comments, to identify key information brokers, isolated teams, and interconnected teams

Source: Jason Plurad, IBM Big Data Meet-Up, 2017

Why Use a Graph?

Whereas traditional row and column data is good for row/column-wise aggregations, graphs are great when it comes to visualizing and analyzing **networks and relationships within and between datasets**



Source: **Robert Allison, Weidong Yang, David Hughes and Sachin Deshpande @ Graph4Good GraphHack by Neo4j**

Covid Epidemic simulator built during the Neo4j Graph4Good competition, which uses graph networks and social media contact points to project the spread of COVID-19 under various conditions

Graph Database Demonstration: Personal Spotify Artist Networks

The following process demonstrates how we convert raw data provided an API service to a graph-based format and visualization

Key Steps

Raw Data	Intermediate Tables	Vertex and Edge Transition	Graph Creation	Visualization
Extract data from source backend (usually a database, sometimes a csv file) *For this example, we will hit the Spotify database, via it's open source API*	*Temporarily load the data into a table (with rows and columns) so we can see which columns are **foreign keys** that need to be transitioned to edges*	*Convert the **rows to vertexes**, and the **foreign key columns to edges***	*Load the new vertexes and edge data into a separate graph database backend*	*Pull the edges and vertexes into a visualization tool*

Data Ingest

Data Cleaning and Transformation

Data Loading

Data Visualization

Toolkit

Graph Database Demonstration

Common Graph Tools

Graph tools are growing across multiple industries and consist of both open-source and managed services

Graph Databases

neo4j

An industry leading suite of graph tools for graph databases, data cleaning, graph transitions and visualizations

JanusGraph

Leading open-source distributed graph database, used for extremely large graphs that need to be stored on a cluster

Amazon Neptune

Graph database service managed by Amazon Web Services (AWS)

Querying Tools

Apache Gremlin is an open source framework for easily querying graphs, using constructs called traversals

neo4j Cypher

Cypher is the native query language for neo4j, with SQL-like syntax

Visualization Tools

Gephi

KeyLines by Cambridge Intelligence

tableau

Cytoscape

Want to Learn More?

Anyone interested in learning more about graph tools can check out the following sources

Prerequisites:

- Basic understanding of functional (Python) and object-oriented programming (Java/JavaScript)

Graph Database Documentation

- Neo4j: https://neo4j.com/developer/
- JanusGraph: https://docs.janusgraph.org/
- Apache Tinkerpop (for graph queries): https://tinkerpop.apache.org/

Datasets:

- https://snap.stanford.edu/data/ (< Includes a Twitter memetracker graph)
- http://networkrepository.com/
- https://graphchallenge.mit.edu/data-sets

Questions