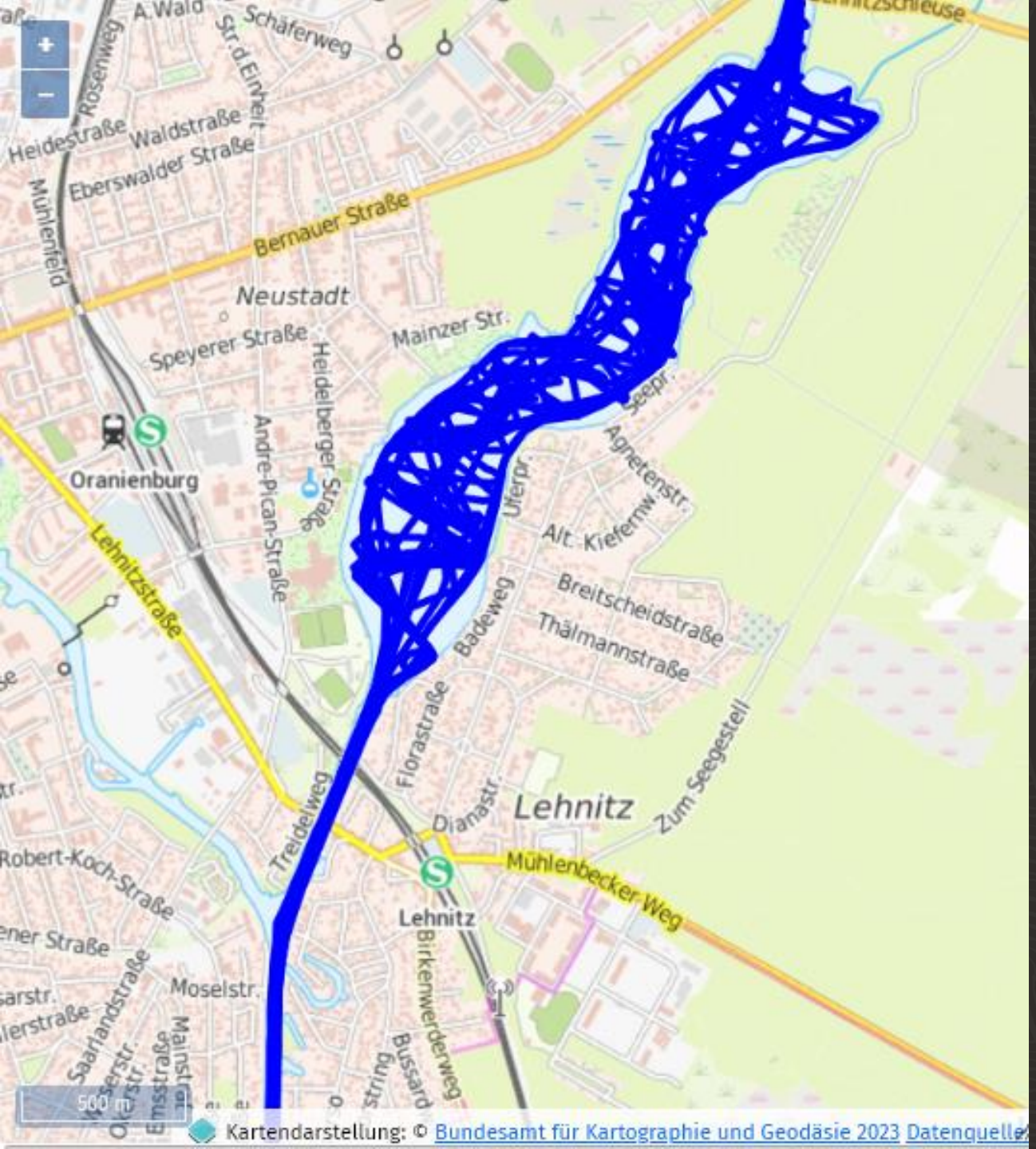
ORACLE

Graph Algorithmen praktisch nutzen

Karin Patenge | karin.patenge@oracle.com
Senior Principal Product Manager | Oracle Spatial and Graph Technologies
DOAG Conference 2023 | Nuremberg | November 21-24



About me

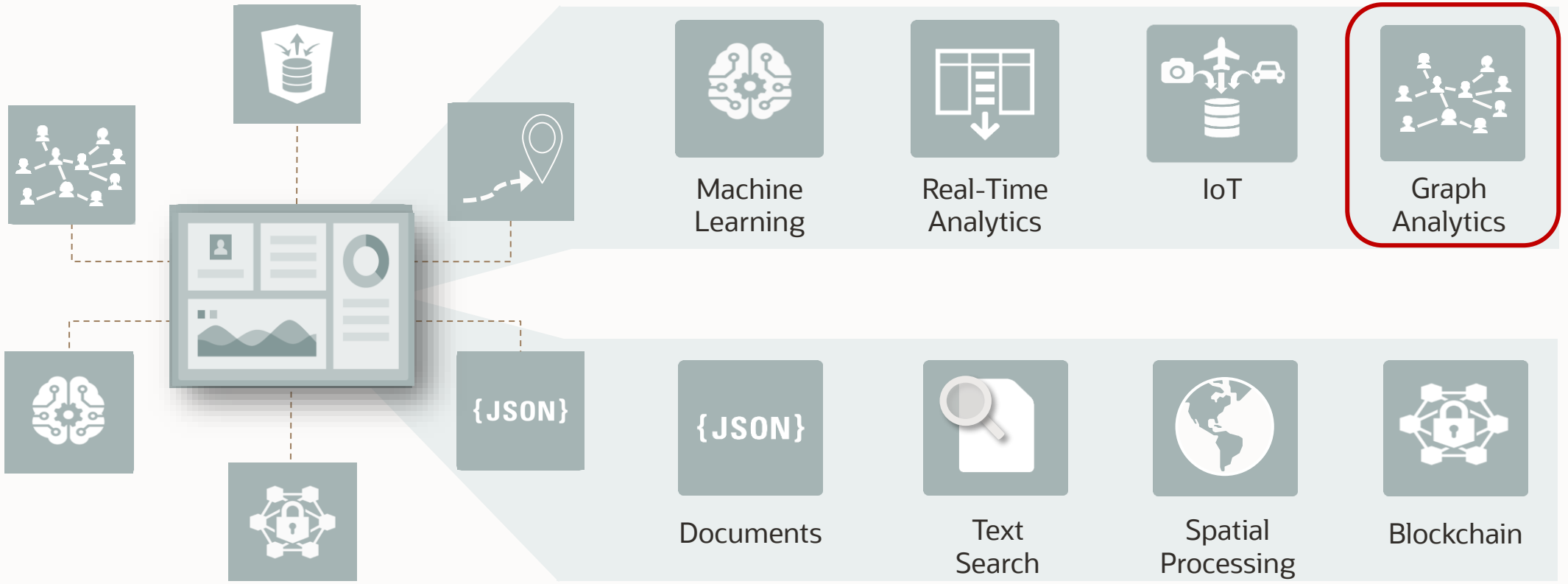
- MSc. in Computer Science (Studies in Romania and Germany)
- Postgraduate studies in Geoinformatics (Austria)
- Worked for Oracle since 2007, many years as a Solution Engineer
- Joined the Oracle Spatial & Graph Product Management Team in 2022
- > 30 years in IT

Agenda

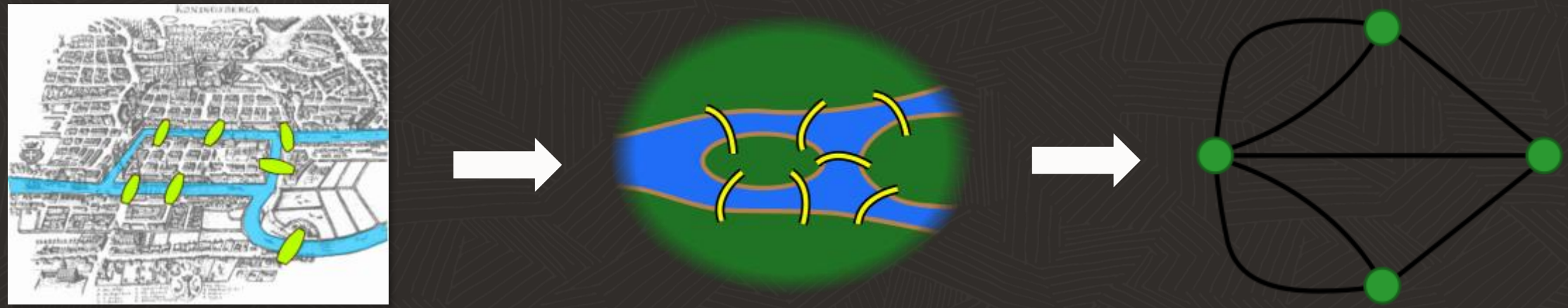
- Where we come from
- The Graph Model & Graph Analytics
- Demo
- Takeaways

Where we come from

Modern Apps Need To Generate Value From Data in New Ways

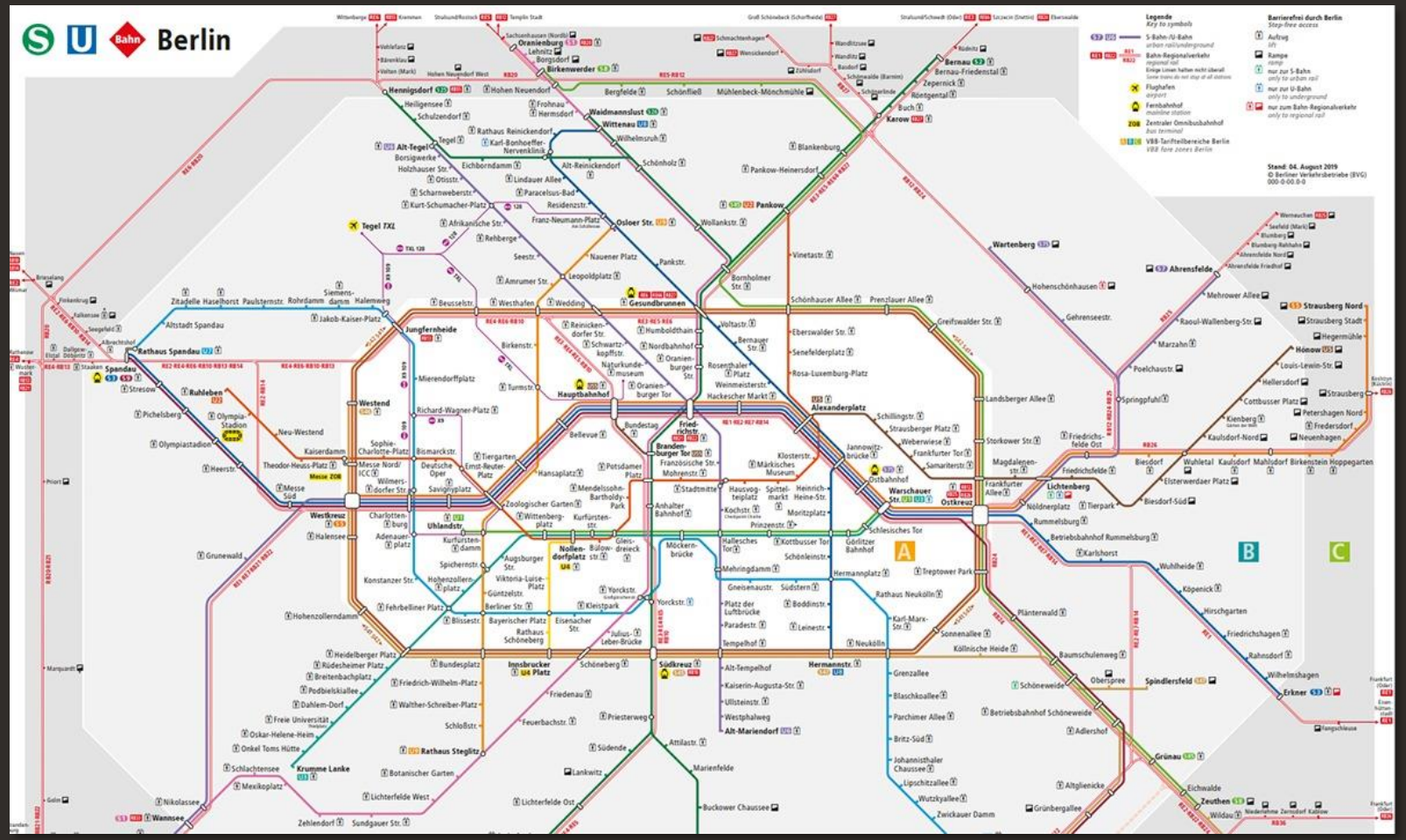


Everything can be a graph



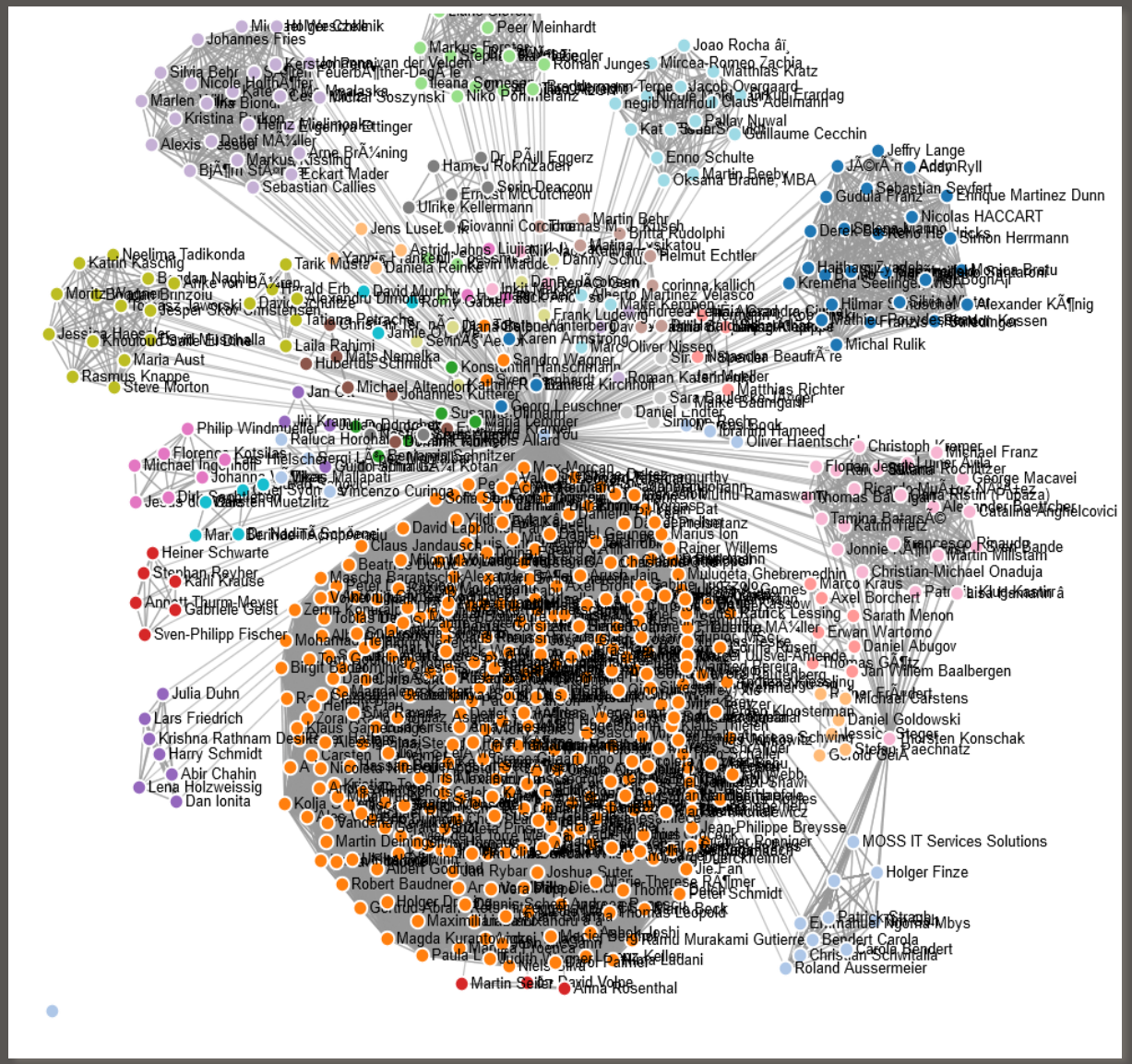
Source: Wikipedia

Everything can be a graph



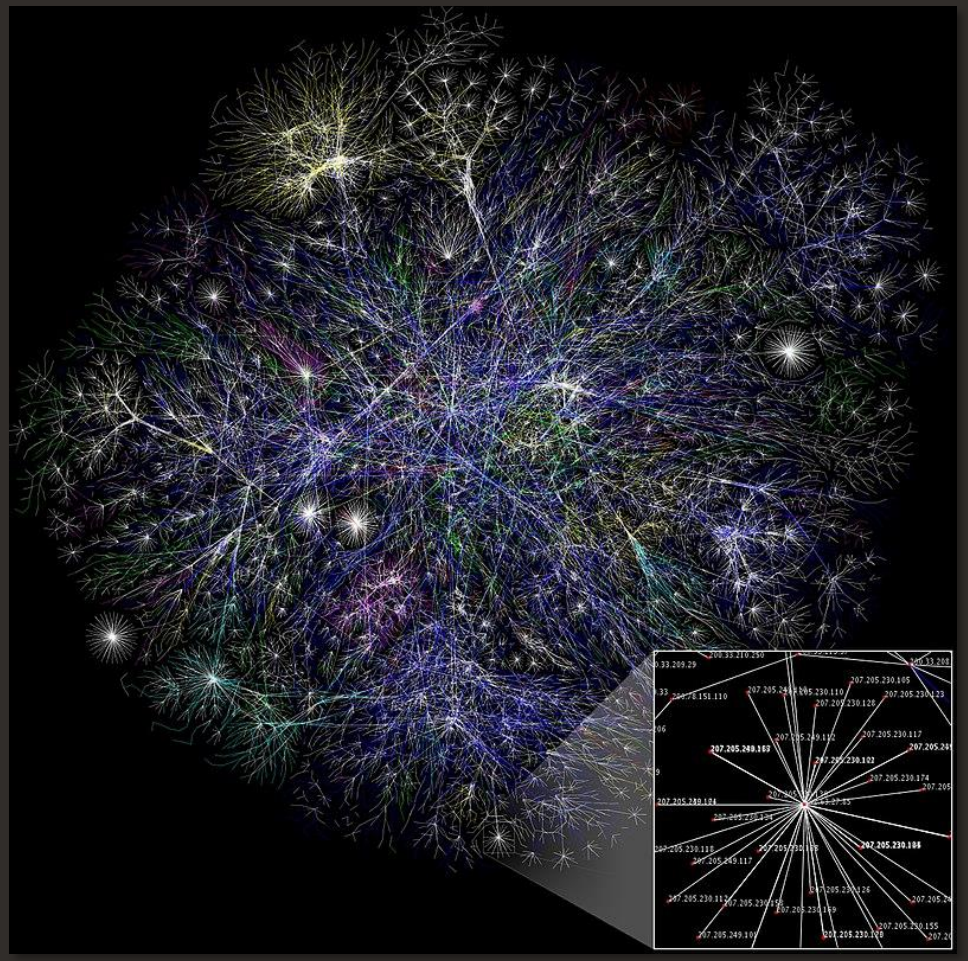
Source: BVG

Everything can be a graph



Data Source: LinkedIn
Visualization: thanh-to.github.io/linkedin-connections-visualization.html

Everything can be a graph



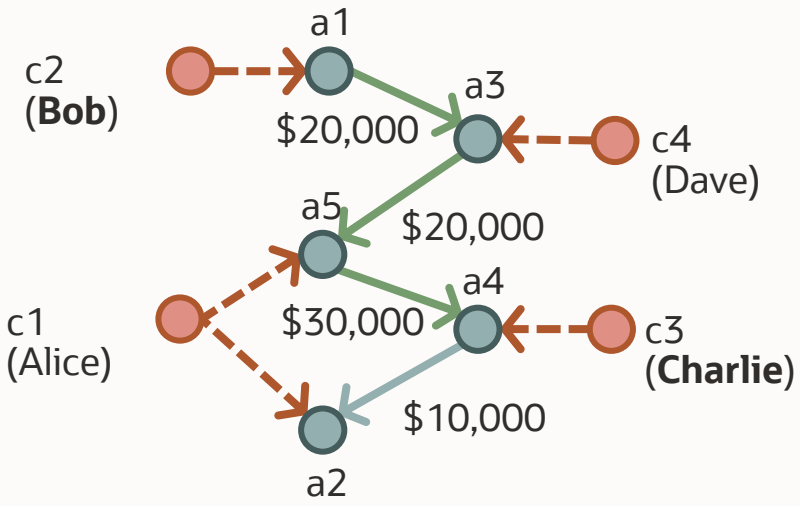
Source: Wikipedia

When are Graphs useful?

Business Problem	Graph Analytics using Pattern Matching and Graph Algorithms
Identifying fraud in financial transactions in relational tables Tables updated with new accounts and transactions every second **10 accounts are being added, are any of them connected to known fraud accounts?**	• Identify a pattern of cycles in cash transfer relationships • Identify accounts through which many transactions flow • Query for anomalous patterns
Find indirect connections in BOMs and other structures in manufacturing **Can a phone camera be built this week? Can a laptop screen be built this week?**	• Identify whether a component can be built at a given time by traversing all the connected components. • New components and new connections lead to new indirect connections
Identify customer communities for retention and product recommendation **Find customers with similar buying patterns. Estimate risk of others leaving if the current customer does.**	• Graph analytics of customer behavior to identify clusters

Use Cases for Graph Models and Graph Analytics

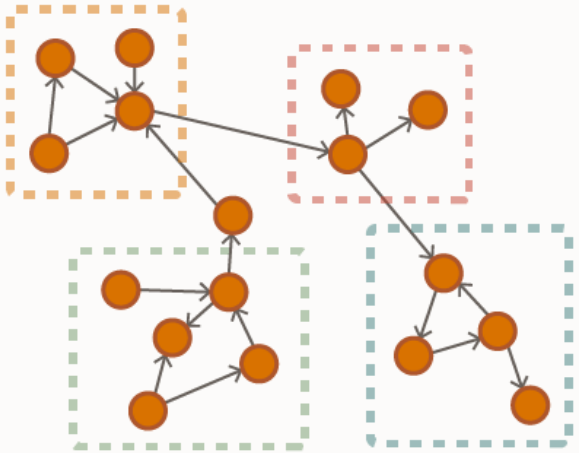
*Is there **any money flow** between Bob & Charlie?*



Graph-based queries:

- Fast traversals
- Path finding
- Identify patterns
- Extract subgraphs

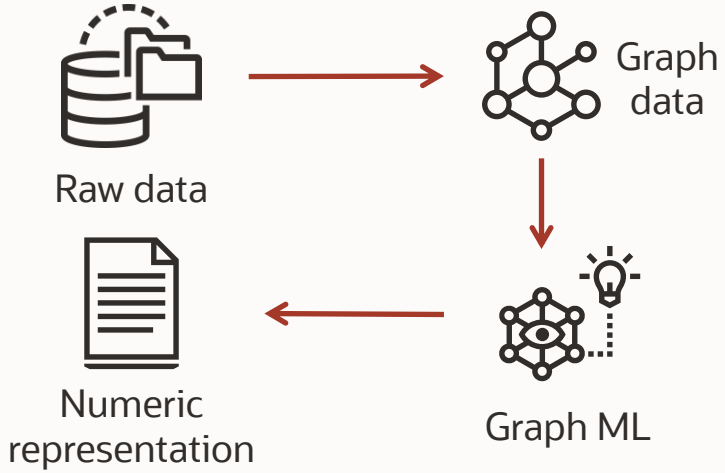
*How to identify the **clusters** of transactions?*



Graph algorithms:

- Community Detection
- Ranking and Centrality
- Paths and Connectivity
- Link Prediction, Similarity

*How to detect **similar structures**?*

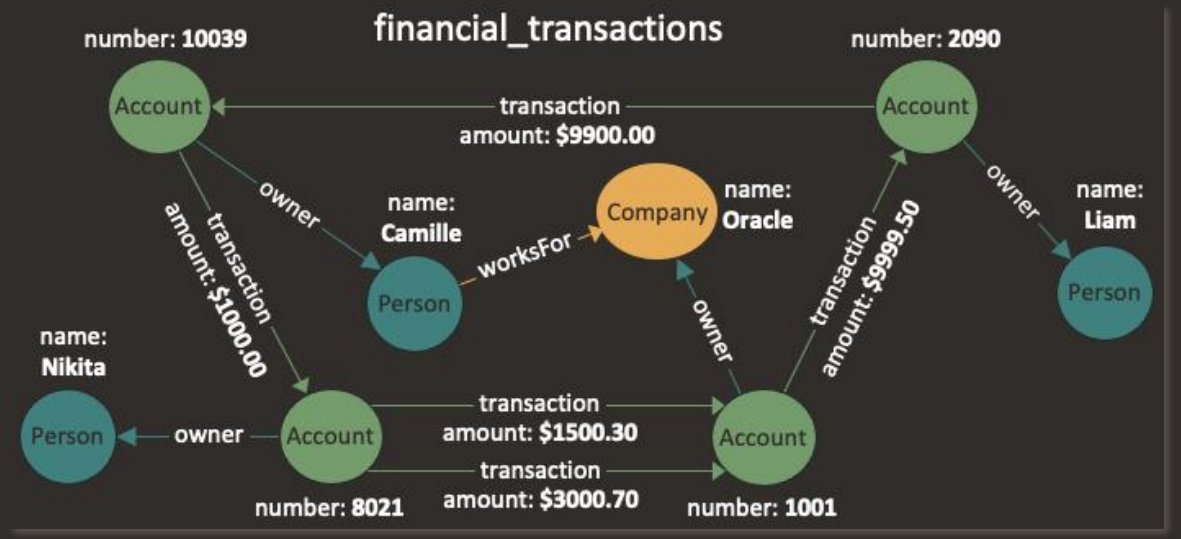


Graph machine learning:

- Unsupervised vector representations of nodes, edges, graphs
- Graph-based supervised classification and regression
- Anomaly detection

Graph Models & Graph Analytics

The (Property) Graph Model



Source: pgql-lang.org

A representation of the real world focusing on relationships using

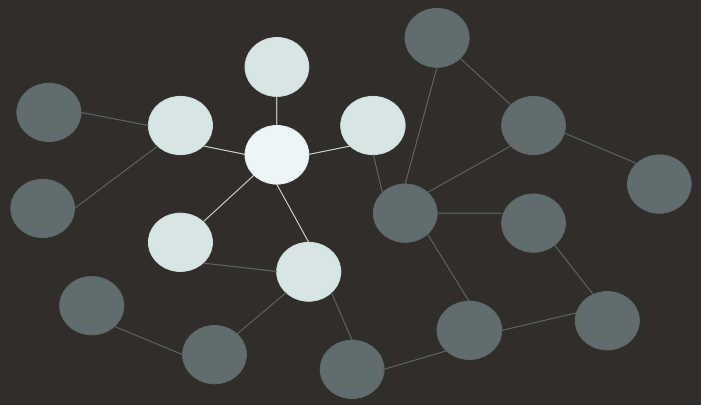
- **Vertices** (nodes) representing a *Thing*
- **Edges** representing relationships between vertices
- **Properties** expressed as key-value pairs attached to vertices and/or edges
- **Labels** representing the type of vertex or edge

→ *Labeled* Property Graph

Graph Analytics

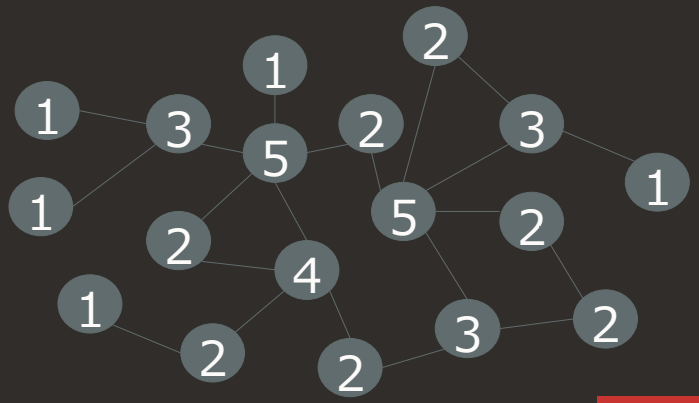
Using a Graph Query Language

- Path finding
- Fast traversals
- Search for surrounding nodes
- Identify patterns
- Extract subgraphs

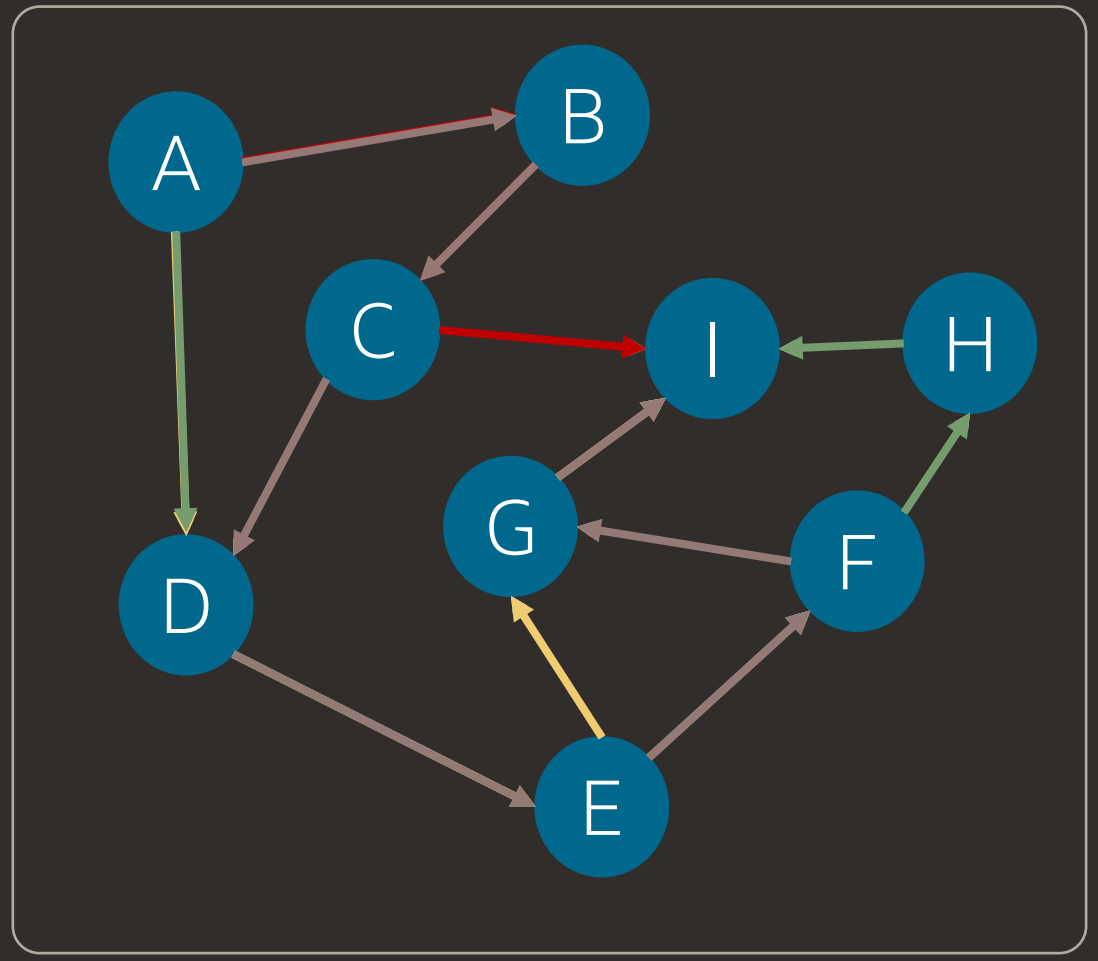


Using Graph Algorithms

- Rank the importance of nodes
- Detect components and clusters
- Detect and evaluate the structure of communities
- Shortest paths
- Predict links
- Find similarities or outliers



Is A connected to I, and if so, how?



Traverse the graph along its paths, starting at node A and trying to reach node I.

Graphs make it easier to answer queries related to connectivity.

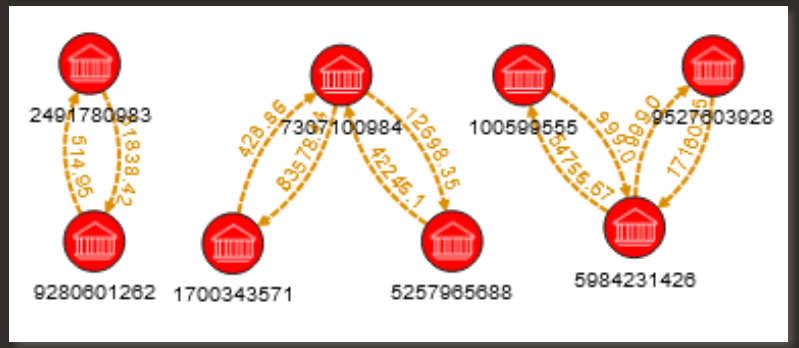
Graphs operate on data as nodes (aka vertices) and edges, instead of as rows and columns.

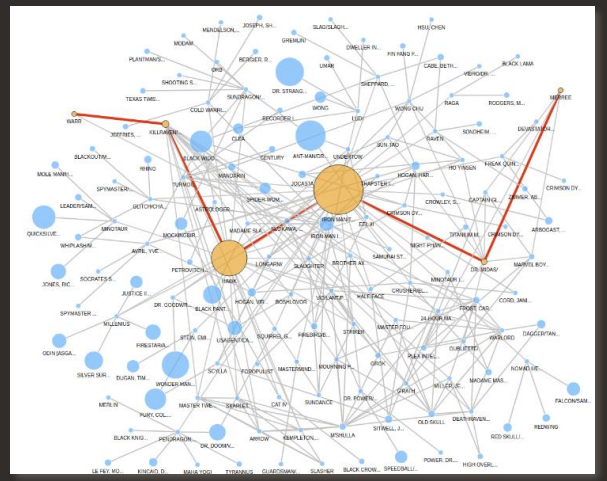
Graphs are intuitive. They align with the way we think.

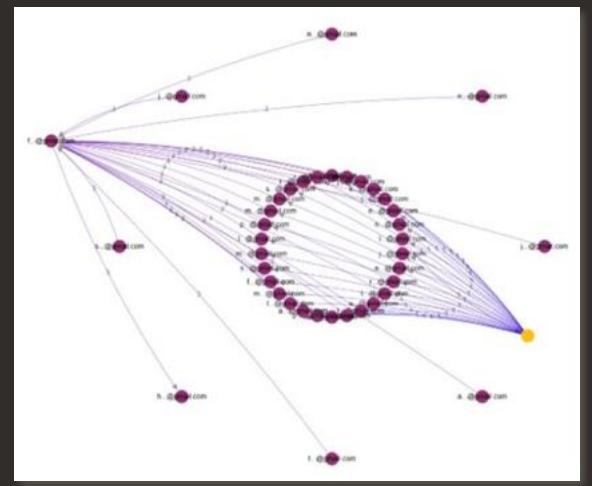
Seeing your data as graphs brings a new paradigm of thinking about what is important within your data.

Detect Patterns in Connected Data

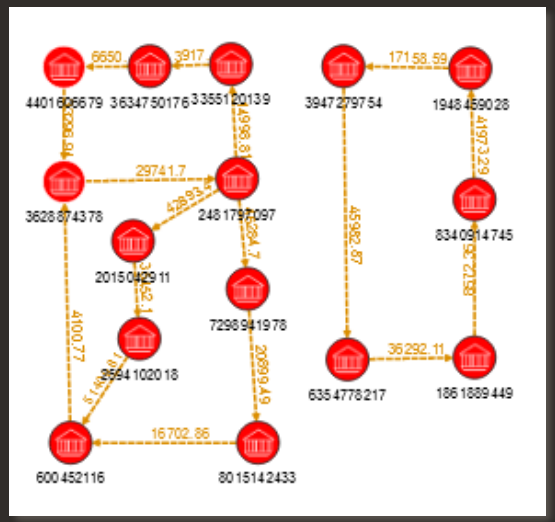
Cycles, paths, more complex patterns

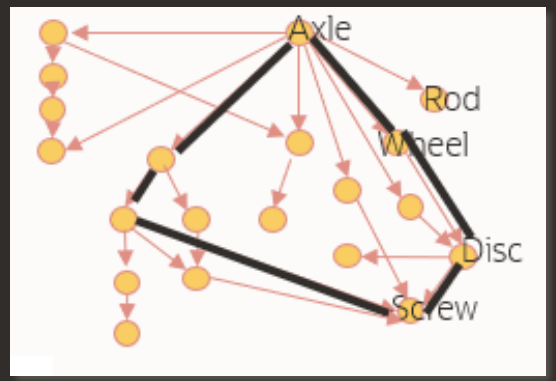


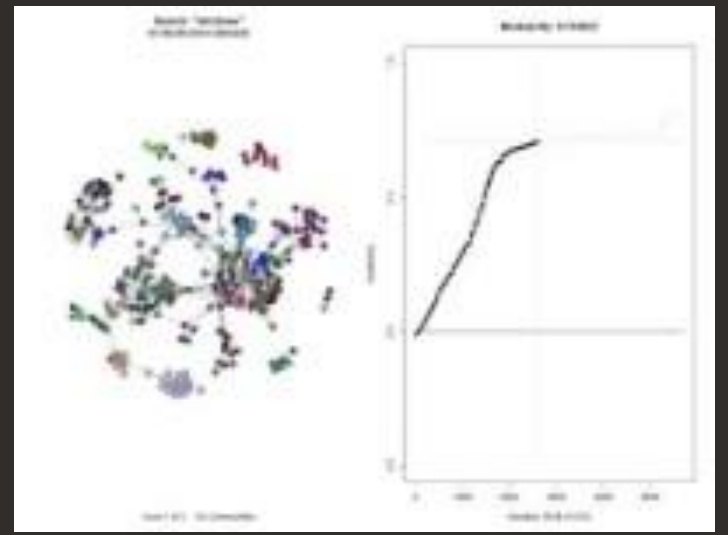




Images courtesy: Paysafe







www.youtube.com/watch?v=dGa-TXpoPz8

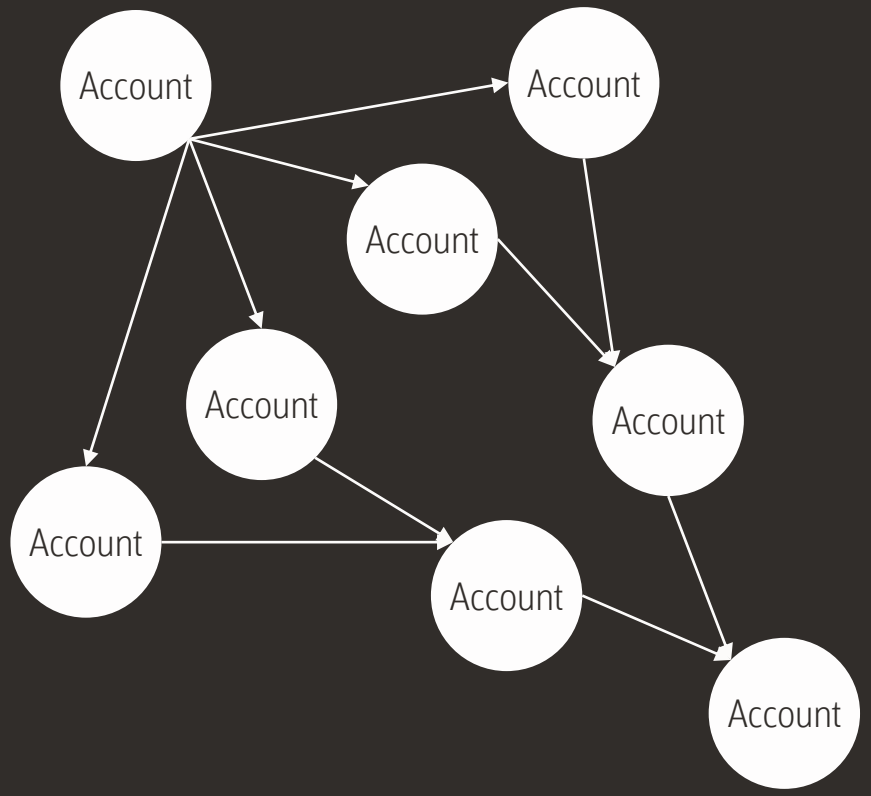
Fraud Prevention in Financial Services

Graph Analytics using a Graph Query Language

Manage large networks of interconnected entities

Detect suspicious patterns in financial transactions, e.g.:

- Indirect connections to suspicious/fraudulent accounts
- Circular money flows, indicative of money laundering
- Layering (hiding money flows in a large number of small, indirect transactions)



BOM (Bill-of-material) Analysis in Manufacturing

Graph Analytics using a Graph Query Language

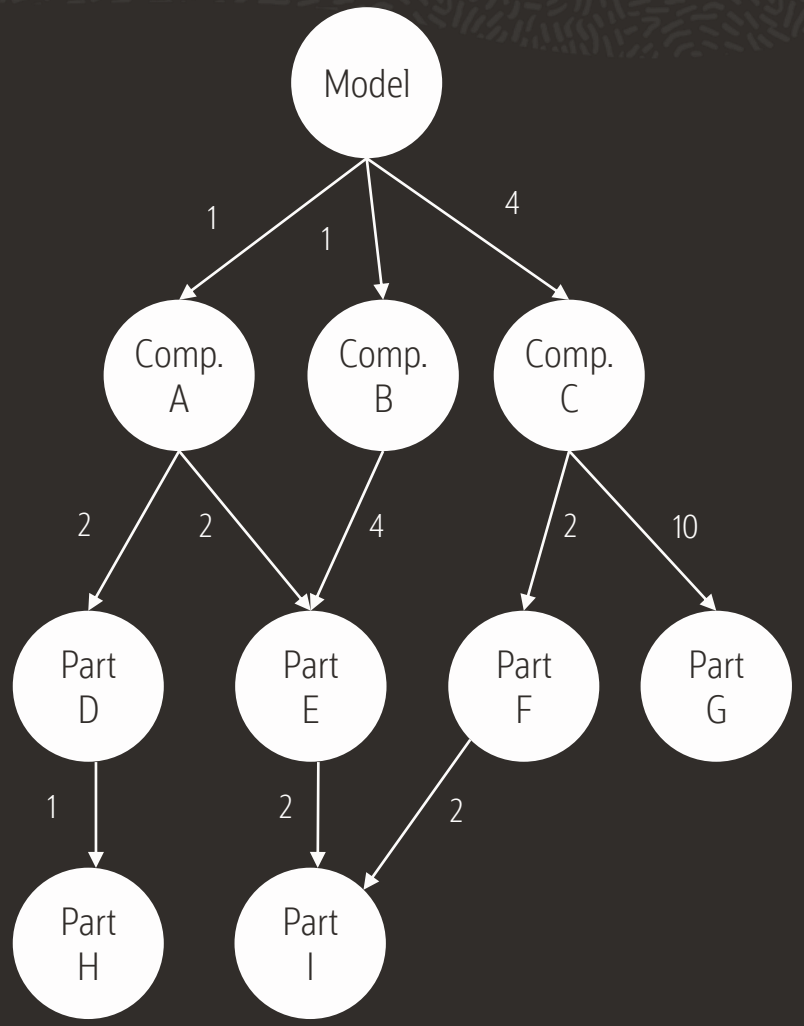
Manage complex hierarchies of parts and components

Analyze nested relationships using very compact queries, e.g.:

- Aggregate required parts for a product variant
- Analyze dependencies on a given part

Similar use cases:

- Data lineage
- GDPR compliance
- Supply Chains



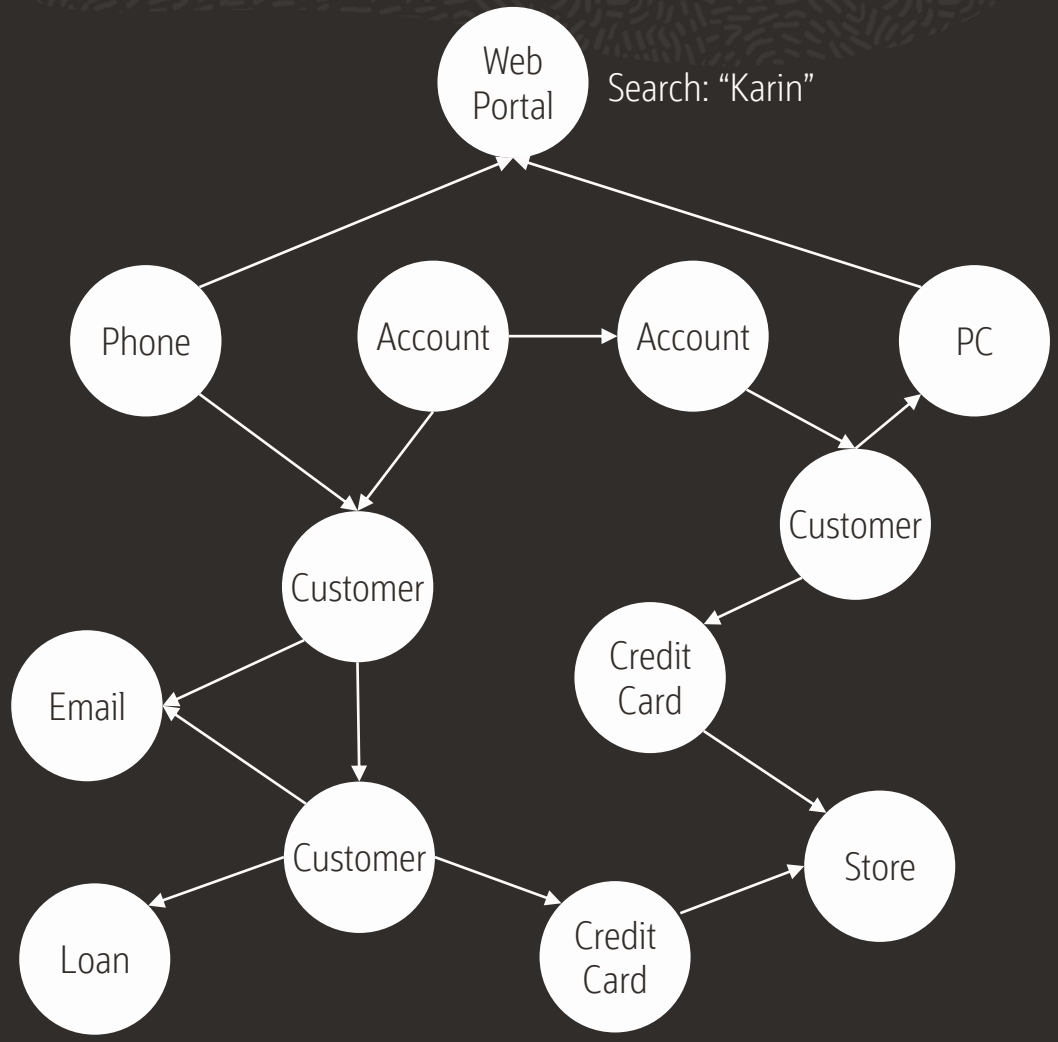
Customer 360 Analysis in e-Commerce

Graph Analytics using Graph Algorithms

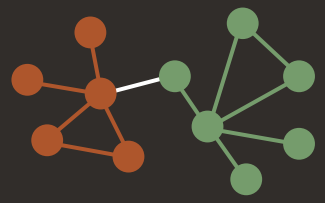
Organize customer data in a data lake or data warehouse

Analyze relationships through graph algorithms, e.g.:

- Associate entities and disambiguate data ("Entity resolution")
- Identify important entities and measure influence
- Determine communities for targeted marketing or risk management
- Predict links, e.g., for product recommendation

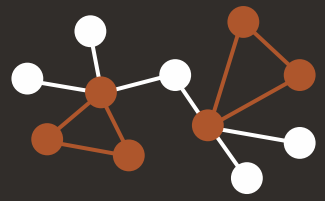


Built-In Graph Algorithms



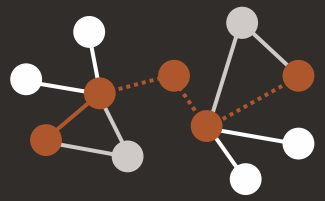
Detecting communities

Strongly Connected Components, Weakly Connected Components, Label Propagation, Louvain, Conductance Minimization, Infomap



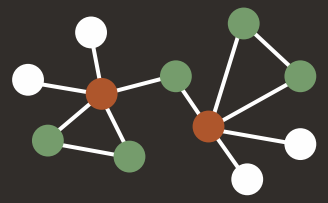
Topology analysis

Conductance, Cycle Detection, Degree Distribution, Eccentricity, K-Core, LCC, Modularity, Reachability Topological Ordering, Triangle Counting, Bipartite Check, Partition conductance



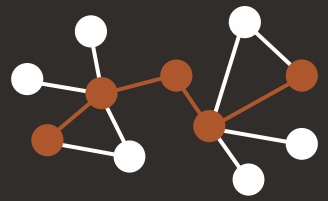
Link prediction and others

Twitter Whom-to-follow, SALSA, Adamic-Adar Index



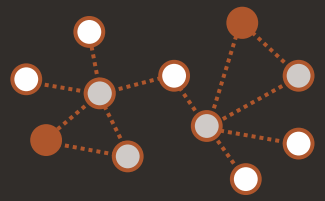
Ranking and walking

PageRank, Personalized PageRank, Degree Centrality, Closeness Centrality, Vertex Betweenness Centrality, Eigenvector Centrality, HITS, Minimum Spanning-Tree (Prim's), Breadth-First Search, Depth-First Search, Random Walk with Restart



Pathfinding

Shortest Path (Bellman-Ford, Dijkstra, Bidirectional Dijkstra), Fattest Path, Compute Distance Index, Enumerate Simple Paths, Filtered and Unfiltered Fast Path Finding, Hop Distance



Machine Learning

DeepWalk, Supervised GraphWise, Unsupervised GraphWise, Pg2Vec, Matrix Factorization, GNNExplainer

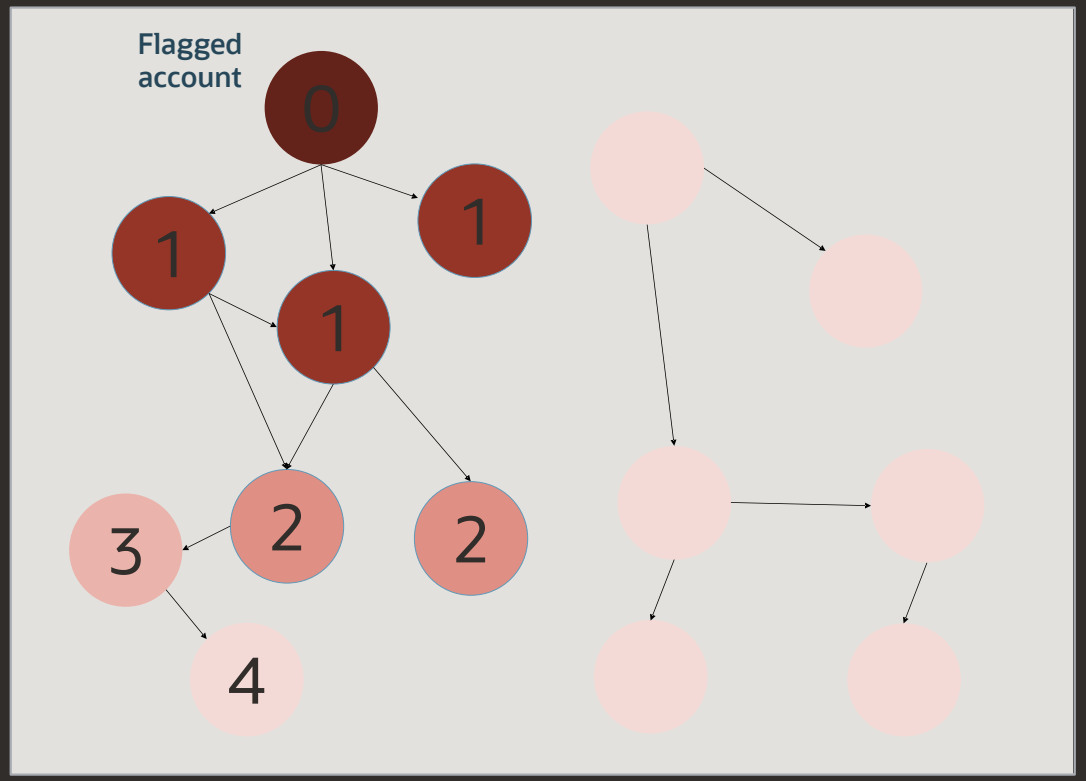
github.com/oracle-samples/pgx-samples/tree/master/built-in-algorithms

Hop Distance

Analytics using Graph Algorithms

Ranking algorithm

- Proximity measure – number of hops from a given vertex
- Useful risk indicator, if one account is confirmed to be fraudulent

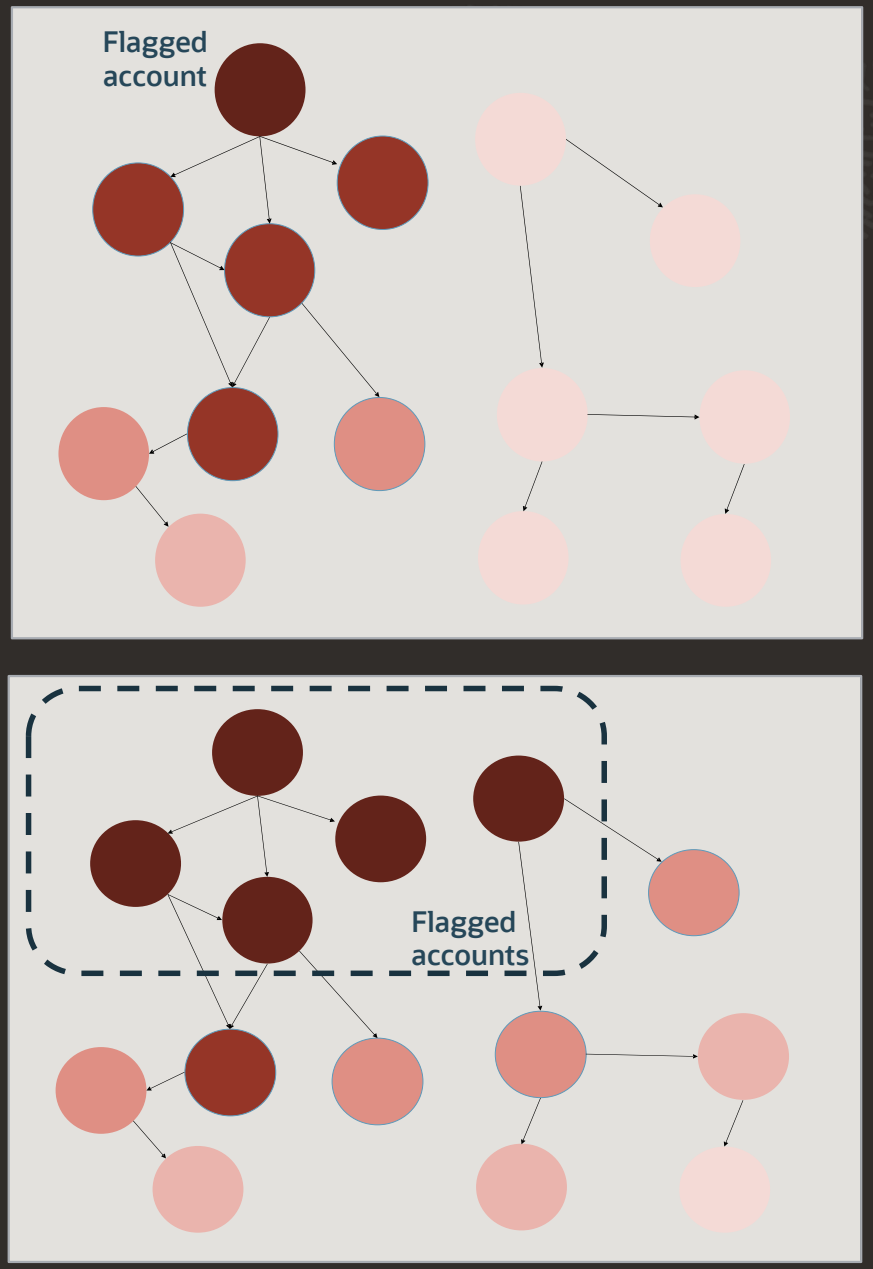


Personalized Pagerank (PPR)

Analytics using Graph Algorithms

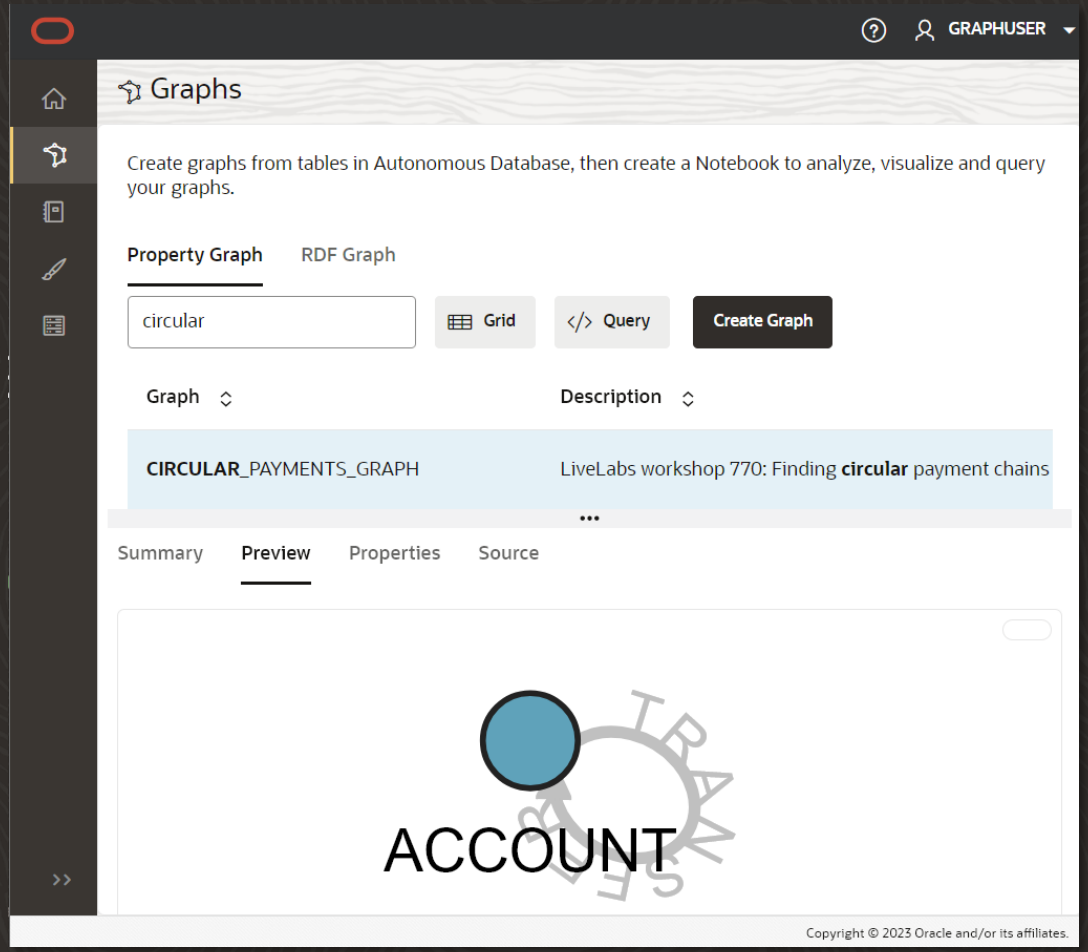
Ranking algorithm

- Proximity measure, starting from a given vertex or a set of vertices
- Probability of reaching a given vertex with a random walk starting at the given vertex
- More granular measure than hop-distance



Demo: Graph Analytics

Use case: Insights into Financial Transactions using Graph Model and Graph Analytics



Summary

1. (Knowledge) Graphs provide a powerful way to represent and visualize your data, enable novel analytics approaches and uncover useful insights.

2. Graphs can be obtained from text documents automatically by using modern natural language processing techniques such as fine-tuning language models.

3. Annotating documents is necessary to teach language models your desired graph schema, but pre-trained transformer models reduce the annotation efforts greatly.