

# Leveraging Graph Data Science with Neo4j

## Advanced Insights and Applications

Ali Balaj

Business Research Centre

December 1, 2024

# Table of Contents

- 1 Introduction to Graph Visualization
- 2 Bloom
- 3 NetworkX
- 4 Gephi

# The Importance of Visualization in Data Analysis

- Simplifies complex data, revealing trends and patterns.
- Transforms numbers into clear, intuitive insights.
- Enhances communication with impactful visuals.
- Supports informed decision-making by making data accessible.
- "A picture is worth a thousand numbers" – visuals drive better decisions.

# Graph Visualization Tools

- **Neo4j Bloom:** Intuitive graph exploration for real-time queries.
- **NetworkX:** Flexible Python library for graph creation and analysis.
- **Gephi:** Advanced tool for large-scale network visualization.
- **Power BI:** Business intelligence with integrated graph visualizations.

# Tool Capabilities

- **Neo4j Bloom:** Ideal for interactive graph exploration (e.g., fraud detection).
- **NetworkX:** Python-based graph creation, ideal for algorithmic tasks.
- **Gephi:** Best for exploring large networks and community detection.
- **Power BI:** Integrates graphs with dashboards for business analytics.

# Comparison of Visualization Tools

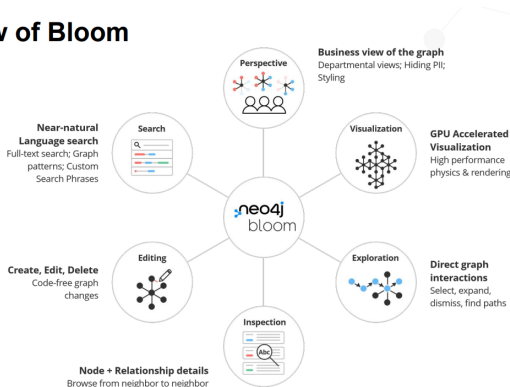
- **Neo4j Bloom:** Real-time graph exploration with database connectivity.
- **NetworkX:** Customizable Python tool for graph analysis and visualization.
- **Gephi:** Powerful for large networks with clustering and layout support.
- **Power BI:** Combines graphs with business intelligence and dashboards.
- Use the right tool based on network size, task complexity, and audience.

# Why use Neo4j Bloom? Isn't Neo4j's built-in visualization already sufficient?

- **User-Friendly Interface:** Intuitive drag-and-drop features for easier graph exploration.
- **Advanced Visualization:** Custom styling, grouping, and node/relationship appearance for better clarity.
- **Contextual Search:** Semantic search for finding patterns using natural language queries.
- **Scalability:** Optimized for large and complex graphs with better zoom, pan, and layout controls.
- **Storytelling:** Create engaging presentations and animations to guide data exploration.
- **Collaboration:** Share visualizations with non-technical users and stakeholders.

# Exploring Neo4j Bloom Capabilities

## Overview of Bloom



Source: Neo4j

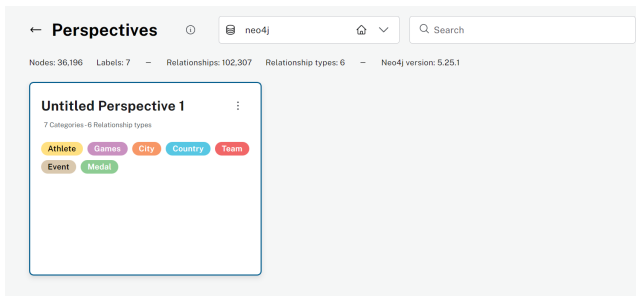


# Image and Data Source Attribution

- The images in this presentation are sourced from the training material of Neo4j Bloom, a tool for interactive graph visualization and exploration.
- Screenshots were captured by myself to illustrate specific features and configurations of Bloom.
- The dataset used for these examples is sourced from Kaggle: 120 Years of Olympic History - Athletes and Results.

# Interactive Data Visualization with Neo4j Bloom

1. To create stunning visualizations, you need data! Launch the Neo4j browser and load the dataset using the provided query. You can find it in the repository at: [Click here!](#)
2. Rename the Perspective and verify Bloom's connection to the database. Let us discuss the Bloom environment.



# Exploring the Settings in Neo4j Bloom

## Settings



Node query limit: 1100

120

Search timeout  
(seconds)

15

30

60

120

240

OFF

Logout timeout: 60 (minutes)



Use classic search



Experimental features



Show restore Scene dialog



Auto-select results



Graph layout compatibility mode



Case insensitive search and suggestions



Property tooltips



Perspective auto-sync

## Product Analytics



Product usage

This data helps us prioritize features and improvements. No personal information is collected or sent.



Crash reports

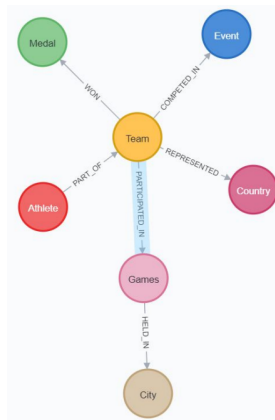
Crash reports allow us to quickly diagnose and fix problems in the software. No personal information is collected or sent.

Bloom provides an intuitive and interactive environment to visualize and explore graph data stored in Neo4j. By configuring various settings, users can tailor the visualization experience to suit their needs, enabling better insights and understanding of the data relationships.

# Introduction to the Kaggle Olympics Data Set

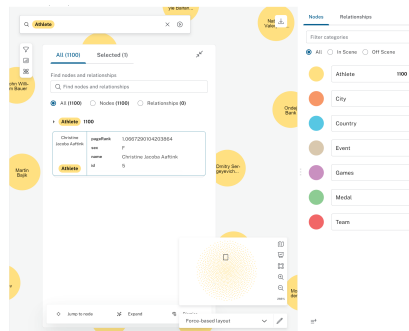
## Kaggle Olympics Data Set:

- Summer and Winter Olympics from 1896 to 2016
- Source:  
[www.kaggle.com/heesoo37/120-years-of-olympic-history-athletes-and-results](https://www.kaggle.com/heesoo37/120-years-of-olympic-history-athletes-and-results)
- We're only filtering on the Winter Games



# Explore the selected object using the card list

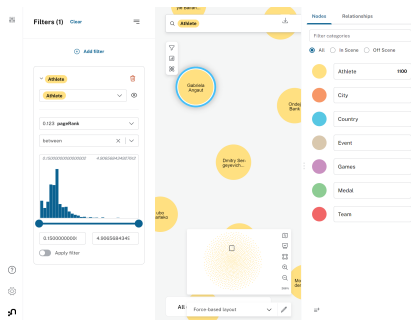
- Choose a suggested category and execute it, such as **athlete**.
- The number of loaded graph nodes depends on the settings discussed in the previous slides. Adjust the setting to load **1,000 athletes**.
- Select an athlete and retrieve more information using the **card list**.



# Getting insights by creating a filter

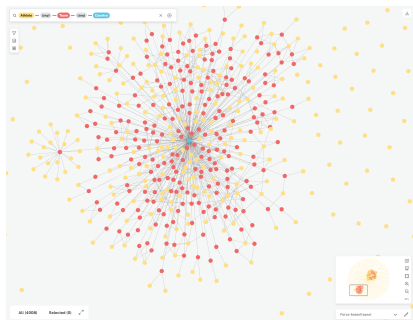
Getting insights by creating a filter allows us to narrow down and focus on specific data, making it easier to analyze relationships and patterns in the graph.

- Create a filter to narrow down the data.
- Check the PageRank distribution of the athletes.
- Create a filter to focus on athletes with a PageRank index between 1 and 1.5.
- Apply a filter to include only female athletes.



# Exploring the Map: Athletes, Teams, and Countries

- Choose the athletes' team and country to visualize the relationships on the map.
- Recognize the clustering patterns among athletes based on their teams and countries.
- Explore the map for a more detailed visualization of data.
- Switch to presentation mode for a clearer view and interaction.
- What additional insights can you gain from the legend bar? Export the results and clear the scene!



# Simple Search in Bloom

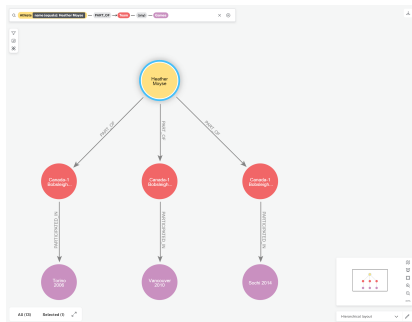
- Select athletes whose names end with a letter of your choice.
- Focus on the selected node by using the "Dismissing Other Nodes" option.
- Expand all details about the athlete and their team using the "Expand All" option.
- Export the results for further analysis or sharing.





# Exercise: Where Has Heather Moyse Competed in Winter Games?

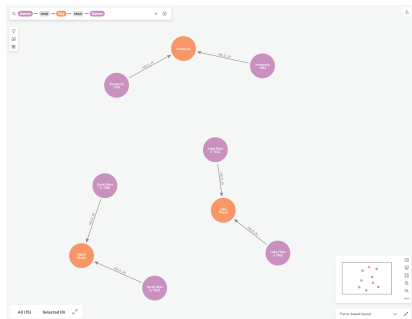
- Where has Heather Moyse participated in the Winter Games?





# Finding Multiple Occurrences

- Perform the search query as demonstrated in the screenshot.
- Explore multiple occurrences:  
As illustrated in the screenshot, the city of Innsbruck hosted the Winter Games in two different years, 1964 and 1976.



# Creating Search Phrases in Bloom

Search phrase \*

Find the Athlete called \$name1 \$name2

Description

Finding an Athlete based on first and last name

Cypher query \*

```
MATCH (node:Athlete) Where node.name CONTAINS $name1  
and node.name CONTAINS $name2 RETURN node
```

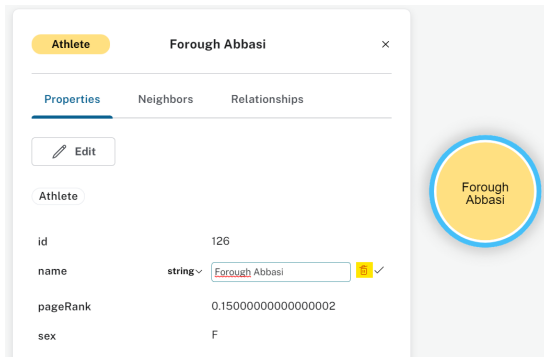
- Go to the perspective drawer.
- Under "Search Phrase," create a new search phrase.
- Execute the search phrase in the search bar.
- Save the results for further analysis.

# Editing Data in Bloom

- Bloom provides basic editing options, such as:
  - Editing property values.
  - Duplicating nodes.
  - Creating new nodes.
  - Creating relationships between nodes.
  - Changing node labels.
- Editing is only possible if you have write access to Neo4j. Without it, these features are unavailable.
- Constraints while editing in Bloom:
  - Relationships can only be created from existing relationship types.
  - You can delete relationships, nodes, or properties.
  - New property keys cannot be created; you must use existing ones.
  - Properties can only be added to existing categories/labels.

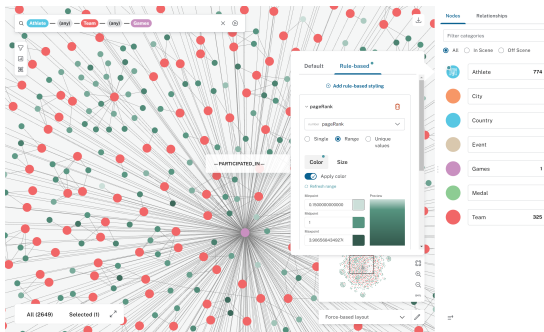
# Editing in Bloom vs. Neo4j Browser

- Utilizing the Neo4j Browser for editing is recommended, as Bloom is primarily optimized for visualization and exploration rather than data modification.
- Summary of editing capabilities in Bloom:
  - Create, edit, or delete nodes, relationships, and properties.
  - Limited to existing relationship types, property keys, and labels.



# Rule-Based Visualization of Athlete Nodes in Winter Sports

- Choose a ski or winter sport icon for athletes.
- Create a rule-based node visualization:
  - Node size or color based on PageRank values.
- Explore the visualization options.
- Export the result as a PNG for further documentation.



# Introduction to NetworkX

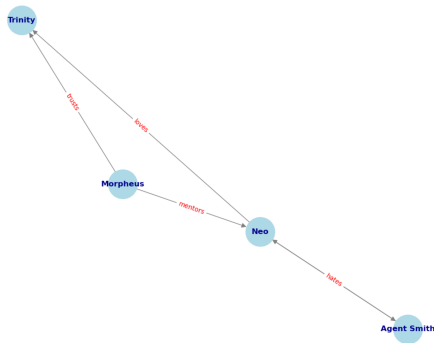
- NetworkX is a Python library for creating, analyzing, and visualizing networks and graphs.
- Supports a variety of graph types:
  - Undirected and Directed Graphs.
  - Weighted Graphs.
  - Multigraphs (graphs with multiple edges between nodes).
- Applications:
  - Social network analysis.
  - Pathfinding and optimization.
  - Centrality measures and graph algorithms.



# NetworkX

- Steps to create and visualize a graph using NetworkX:
  - 1 Import the necessary libraries for graph visualization.
  - 2 Initialize a graph object.
  - 3 Add nodes and define relationships (edges) between them.
  - 4 Customize and display the graph visualization.

Directed Relationships Between Matrix Characters



[Click here for the full code](#)

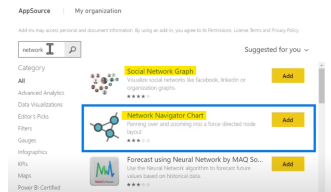
# Gephi – Network Visualization Tool

- Gephi is a powerful open-source software for exploring and visualizing graphs and networks.
- Key Features:
  - 1 Supports large-scale graph visualization and analysis.
  - 2 Provides a rich set of layout algorithms (e.g., Force Atlas, Yifan Hu).
  - 3 Offers metrics like PageRank, Betweenness Centrality, etc.
  - 4 Allows rule-based customization of node/edge size, color, and style.

► [Download Gephi Here](#)

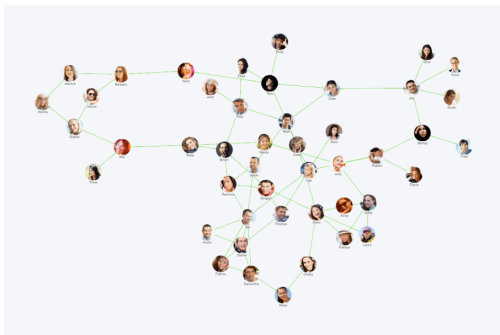
# Getting Started with Power BI

- Steps to Create a Graph Visualization in Power BI:
  - 1 Download and install Power BI Desktop from Power BI's official website.
  - 2 Import data from supported sources (e.g., Excel, SQL Server, or online services).
  - 3 Clean and transform the data using Power Query Editor.
  - 4 Explore **custom visualizations** from the App Store.
  - 5 Publish reports to the Power BI Service for sharing and collaboration.



# Custom Visualizations in Power BI

- Power BI allows the use of custom visualizations beyond the default options.
- Steps to use custom visualizations:
  - 1 Go to the Power BI App Store by clicking "More Visualizations" in the Visualizations Pane.
  - 2 Search for a desired custom visualization (e.g. Social Network Graph or Network Navigator Chart).
  - 3 Use it like any other chart by assigning fields and adjusting settings.



**Thank you!**

Any questions?