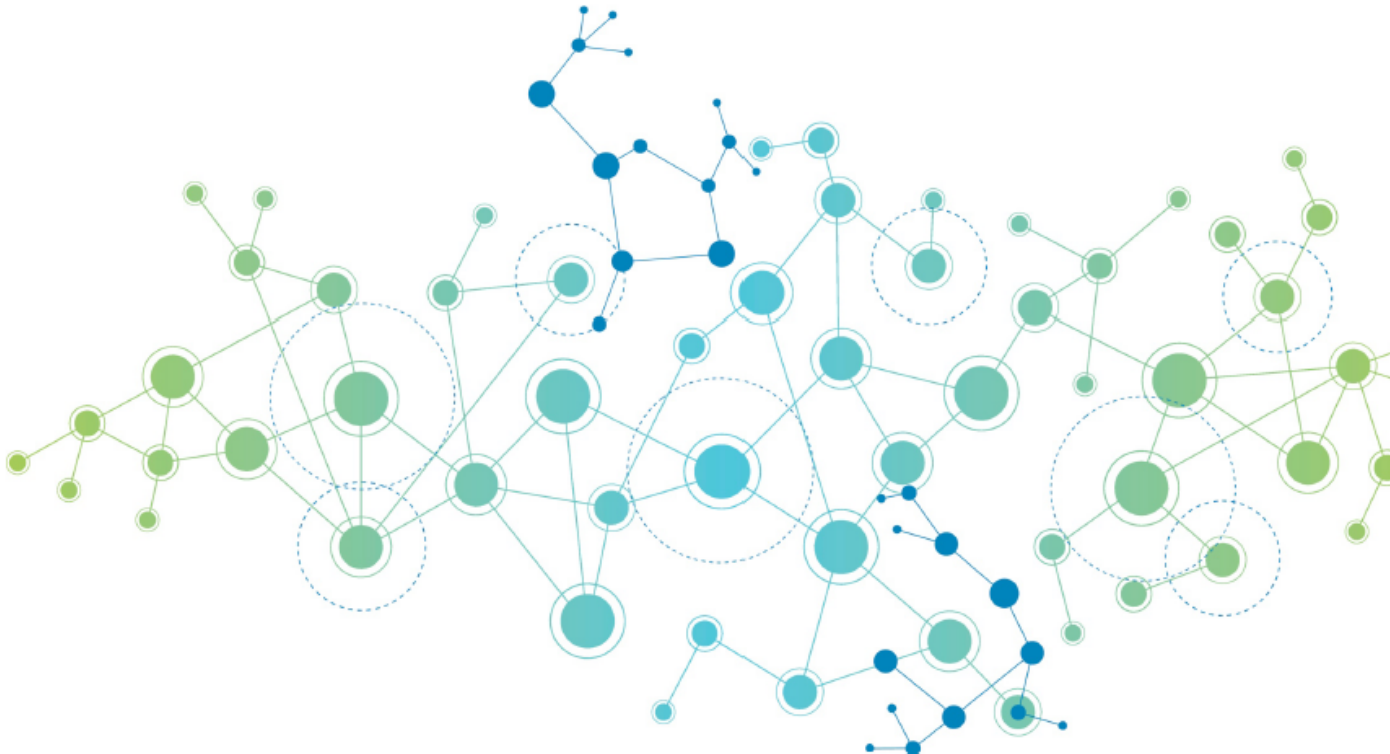




A graph database.



Óscar García Hinde

# Contenidos.

- Qué es una Graph DB?
- Graph DBs vs Relational Dbs.
- Introducción a Neo4j.
- Tutorial.
- Discusión.

# ¿Qué es una graph database?

- Base de datos que estructura la información en forma de **grafo**:
  - **Nodos**: elementos almacenados en el grafo, como personas, cuentas, documentos...
  - **Aristas**: líneas que conectan los nodos, representando interrelaciones.
  - **Propiedades**: atributos específicos de los nodos (y aristas) como nombre, nacionalidad, rol...

# Graph DBs vs Relational DBs.

- Las **DBs relacionales** no contienen la idea de relaciones fijas entre datos de forma explícita.
- Esto complica la implementación de ciertas consultas basadas precisamente en las relaciones entre datos.

*¿Cuál es esa peli sobre submarinos con el actor que salía en esa otra peli con ese otro actor que era el prota de “El Fugitivo”...?*

# Graph DBs vs Relational DBs.

## Propiedades:

- Son muy eficientes cuando gestionan datos asociativos.
- Pueden escalar de una forma más natural en el caso de conjuntos de datos grandes.
- Son capaces de adaptarse mejor a situaciones ad-hoc en escenarios poco estables.
- Son muy potentes en consultas y operaciones con forma de grafo: ruta más corta entre nodos, detección de comunidades, etc.

# Introducción a Neo4j.

## Información general:

- **Neo4j** es un sistema de gestión y visualización de Graph DBs.
- Es la Graph DB **más popular** del mundo según [www.db-engines.com](http://www.db-engines.com)
- Versión 1.0 en febrero de 2010.
- Versión 3.1.0 en diciembre de 2016.
- **Open Source.**

# Introducción a Neo4j.

## Compatibilidades:

- Integración a través de drivers con:
  - Java
  - JavaScript
  - Python
- Integración con Spark (y otros) a través del protocolo Bolt.

# Introducción a Neo4j.

## Estructura de datos:

- Todo se guarda como nodo, arista o atributo.
- Cada nodo y arista pueden poseer cualquier atributo.
- Se pueden aplicar etiquetas para facilitar las consultas.
- Se puede indexar.



# Introducción a Neo4j.

## Cypher Query Language:

- Es un lenguaje declarativo inspirado en SQL.
- Trata de describir explícita y visualmente patrones en grafos mediante **sintaxis ascii-art**.
- Permite definir **qué acciones** queremos ejecutar sin definir **cómo** el programa tiene que hacerlo.

# Introducción a Neo4j.

ASCII-Art syntax:

Cypher using relationship 'likes'



Cypher

(a) -[:LIKES]-> (b)

# Introducción a Neo4j.

## Visualización:

- Cualquier herramienta de visualización de grafos que utilice datos **JSON**:
  - D3.js, three.js, sigma.js o Alchemy.js.
- **Screencast**: herramienta de visualización de la aplicación browser de Neo4j.

# TUTORIAL TIME!

