

Neo4J

Mikel Egaña Aranguren

mikel-egana-aranguren.github.io

mikel.egana@ehu.eus



Neo4J

<https://github.com/mikel-egana-aranguren/ABD>



Neo4J

Base de datos más importante en Property Graphs (No RDF)

Almacenaje nativo optimizado para grafos: muy eficiente (ej. Stardog basado en RockDB)

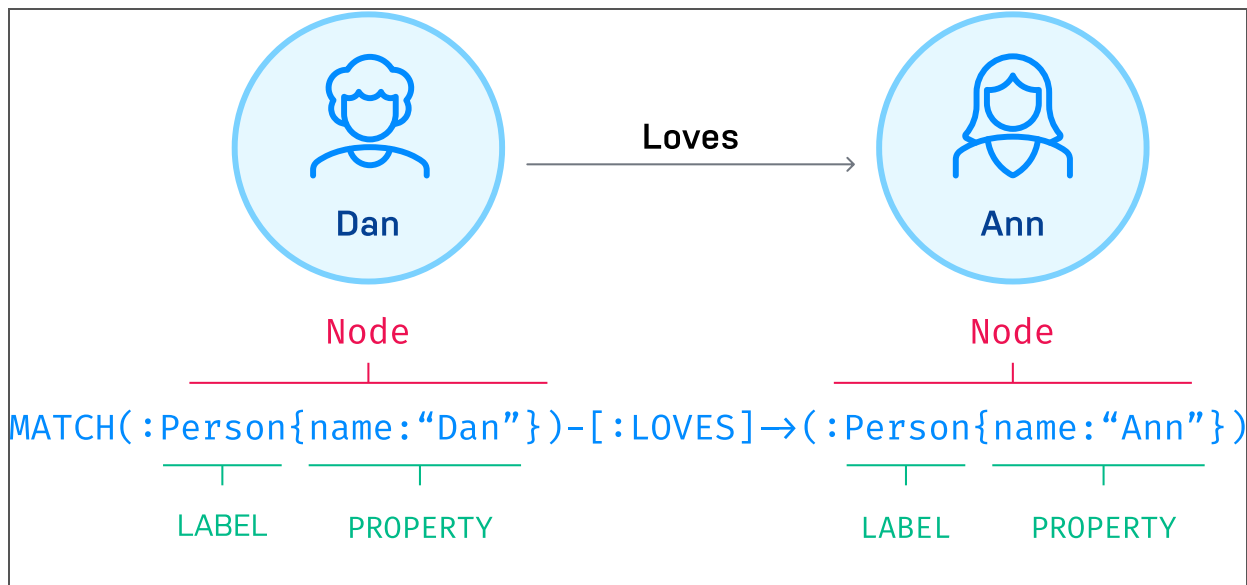
Modelo de datos propio (No estándar)

Modelo Neo4J

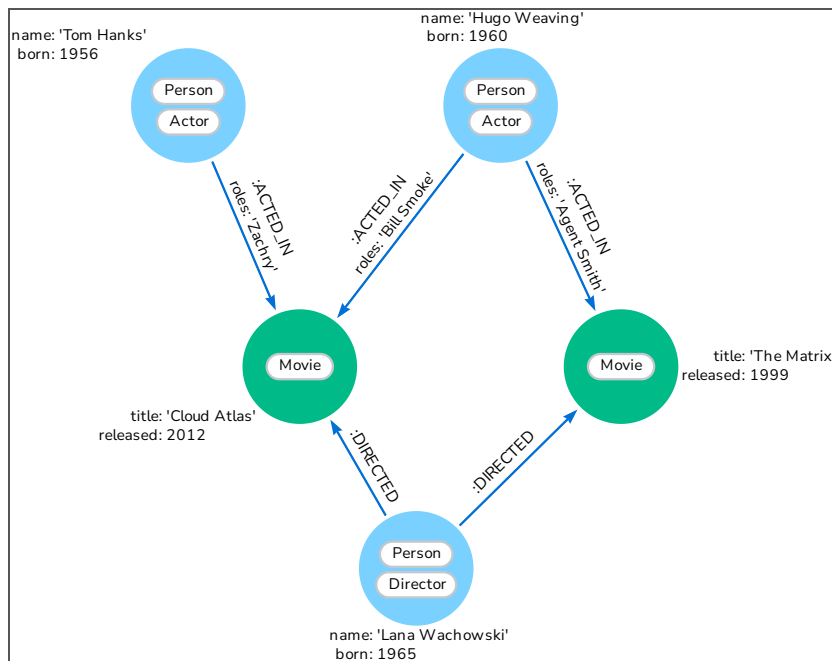
Nodos o relaciones:

- Labels (Type en RDF)
- Properties: parejas clave valor
- No tienen URI

Modelo Neo4J



Modelo Neo4J



Consulta Neo4J

Cypher: lenguaje de consulta, creación y modificación de grafos

APOC (Awesome Procedures On Cypher): extiende Cypher con funciones

Cypher: crear datos

Crear un nodo: **CREATE (n:Persona {nombre: "Juan", edad: 30})**

Crear una relación: **CREATE (n:Persona {nombre: "Juan"})-[:AMIGO_DE]->
(m:Persona {nombre: "Marta"})**

Cypher: consultar datos

Seleccionar nodos: **MATCH (n:Persona) RETURN n**

Patrones de relación: : **MATCH (n:Persona)-[:AMIGO_DE]->(m) RETURN n, m**

Cypher: actualizar datos

Actualizar un nodo: **MATCH (n:Persona {nombre: "Juan"}) SET n.edad = 31**
RETURN n

Cypher: eliminar datos

Eliminar un nodo: **MATCH (n:Persona {nombre: "Juan"}) DELETE n**

Cypher: otras cláusulas

MATCH (n:Persona) RETURN n.nombre, n.edad ORDER BY n.edad DESC

MATCH (n:Persona) RETURN n.nombre ORDER BY n.edad DESC LIMIT 10

MATCH (n:Persona) WHERE n.edad > 30 RETURN n

MATCH (n:Persona) RETURN count(n)