

Ejemplo de BD en grafo: Neo4j

Máster en Business Analytics y Big Data
Bases de Datos No Convencionales



Contenidos de la sesión

- Introducción y características
- Modelo de datos
- Operaciones CRUD
- Estrategias de distribución
- Recursos y enlaces

¿Qué es Neo4j?

- Creada por Neo Technology, Inc.
- Sigue un modelo de grafo de propiedades etiquetado.
- Disponible en sistemas Linux, Windows y OS X
- Incorpora:
 - *Drivers* para multitud de lenguajes de programación
 - Un lenguaje declarativo de manipulación de datos: Cypher
 - Una API REST para consultar/modificar los datos de la base de datos.

Características de Neo4j

- Modelo de datos basado en grafos:
 - Permite definir propiedades (atributos) en los nodos y las relaciones.
 - Permite definir los tipos de los nodos (no tan *schemaless*).
 - Permite definir índices.
- Usa un sistema de transacciones ACID.
- Las operaciones se realizan a través de *graph traversing*.
- Permite *sharding*.
- Gestión de réplicas vía *master-slave* asíncrona



Modelo de datos

Modelo de datos

Modelo Relacional

Base de datos

Tabla, vista

Fila

Columna

Clave primaria

Clave foránea

Combinación
(*join*)



Neo4j

Grafo

Etiqueta de Nodo

≈ Nodo

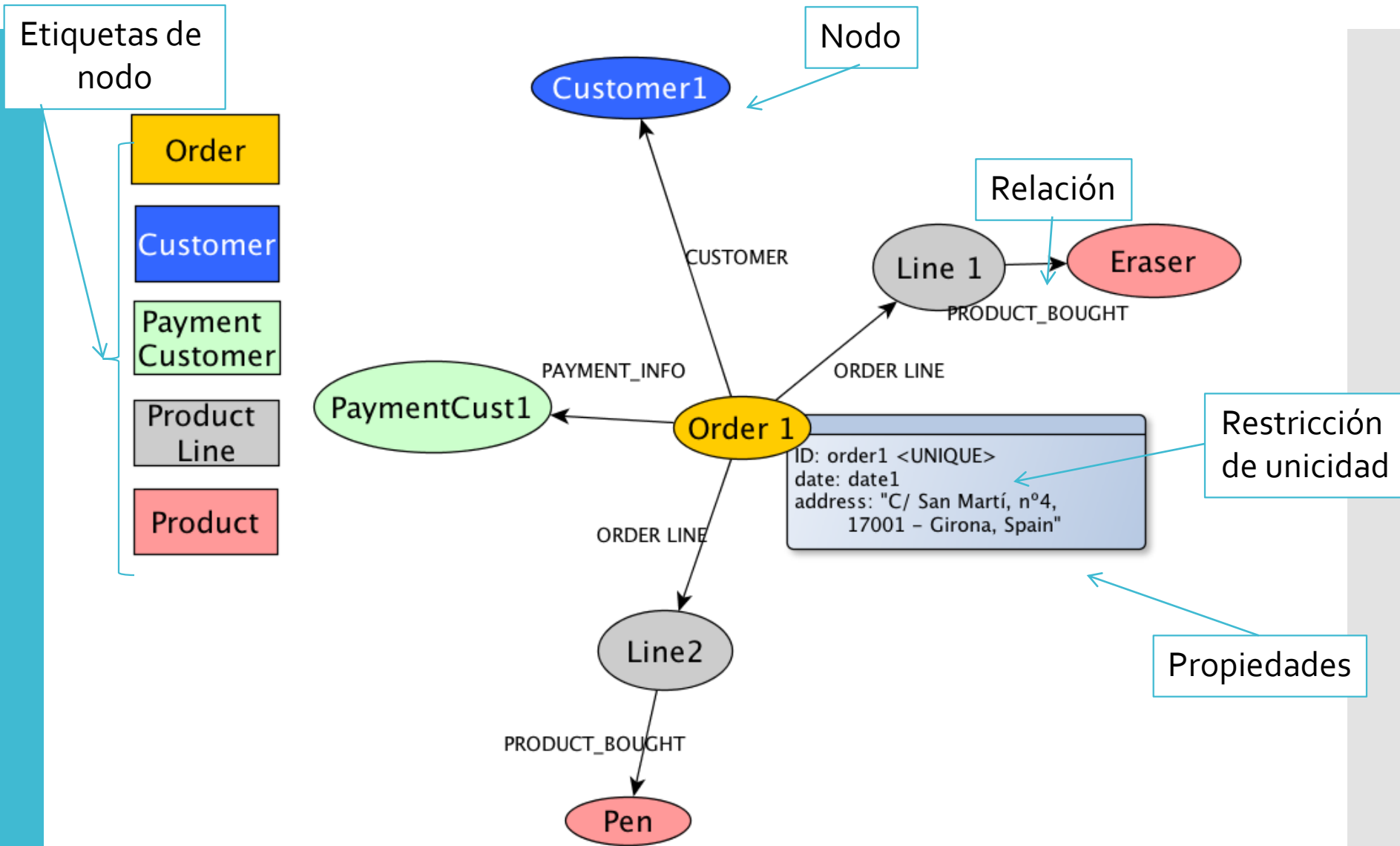
Propiedad

Propiedad + restricción de
unicidad

relación

Graph traversing

Modelo de datos: Ejemplo



Índices

- Se indexan las propiedades de los grafos:
 - Permiten utilizar elementos del esquema de datos.

Ejemplo: Crear un índice sobre la propiedad *country* de los nodos de tipo *Person*

- Los índices se crean de forma asíncrona.
- Existe otro tipo de índices, denominados *Legacy Index*
 - Permite definir índices personalizados.
 - Actualmente no se pueden realizar mediante Cypher.



Operaciones CRUD

Consultas básicas

- MATCH <patrón> RETURN <elementos>
- 1. MATCH (n) RETURN n
- 2. MATCH (m:Movie) RETURN m
- 3. MATCH (m:Movie) RETURN m.title
- 4. MATCH (m:Movie)-->other RETURN m, other LIMIT 100
- 5. MATCH (m:Movie)<--other RETURN m, other LIMIT 100
- 6. MATCH (m:Movie)--other RETURN m, other LIMIT 100
- 7. MATCH (m:Movie)-[a:ACTED_IN]-(p:Person)RETURN m.title, p.name, a.roles

Aplicar filtros a las consultas

- Cláusula WHERE <condición>
- Cláusula DISTINCT <lista elementos>
 1. MATCH (n) WHERE n.title='V for Vendetta' RETURN n
 2. MATCH (n:Movie) WHERE n.title='V for Vendetta' RETURN n
 3. MATCH (n:Movie)--(other) WHERE n.title='V for Vendetta' RETURN n,other
 4. MATCH (m:Movie)--(p:Person) WHERE m.title='The Matrix' OR m.title='The Matrix revolutions' OR m.title='The Matrix Reloaded' RETURN p.name
 5. MATCH (m:Movie)--(p:Person) WHERE m.title IN ['The Matrix', 'The Matrix revolutions', 'The Matrix Reloaded'] RETURN distinct p.name
 6. MATCH (m:Movie)--(p:Person) WHERE m.title=~'.*Matrix.*' RETURN p.name
 7. MATCH (m:Movie)--(p:Person) WHERE m.title=~'.*Matrix.*' RETURN DISTINCT p.name

Transacciones

- Satisface las propiedades ACID:
 - Gestión de concurrencia vía reservas en escrituras a nivel de nodos y relaciones
 - Nivel de aislamiento por defecto *read committed*:
 - Siempre se leen datos confirmados.
 - Pueden aparecer lecturas no repetibles.
 - Se puede evitar reforzando el nivel de aislamiento manualmente

Creación de nodos

- Creación de nodos:
 - CREATE (nodo [:Tipo][{Propiedades}])
RETURN nodo
1. CREATE (p:Person)
 2. CREATE (p:Person { name : "Alex"}) RETURN p
 3. CREATE (s:Subject { title : "NoSQL", credits: "4.5 cr"})

Creación de relaciones

- Creación de relaciones:

- MATCH <patrón> WHERE <condiciones>
CREATE <datos relación>

1. MATCH (p:Person), (s:Subject) WHERE p.name="Alex"
AND s.title="NoSQL"
CREATE p-[r:TEACHES
{dedication : "100%", semester : "March 2014"}
]->s

Eliminación de nodos y relaciones

- No se puede eliminar un nodo si tiene relaciones asociadas:
 - MATCH <patrón> DELETE <lista elementos>
 1. MATCH (p:Person) WHERE p.name="Alex"
DELETE p
 2. MATCH (p:Person)-[r]-(s) WHERE p.name="Alex"
DELETE p, r

Modificación de propiedades

- Añadir/modificar/borrar propiedades:
 - MATCH ... [SET propiedad=valor] [REMOVE propiedad]
1. MATCH (p:Person) WHERE p.name="Robin Williams"
SET p.rating=5
 2. MATCH (p:Person) WHERE p.name="Robin Williams"
REMOVE p.rating
 3. MATCH (p:Person) WHERE p.name="Robin Williams"
SET p.rating=3, p.oscarsOwned=1

Modificación de etiquetas: Tipos

- Añadir/borrar la etiqueta de un nodo o una relación:
 - MATCH ... {SET | REMOVE} {nodo | relación} :Tipo1:....TipoN
1. MATCH (p:Person)-[r:ACTED_IN]->(f)
SET p :Actor
 2. MATCH (p:Actor) WHERE p.name="Robin Williams"
REMOVE p :Actor

Otras operaciones

- Gestión de índices:
 - <CREATE | DROP> INDEX ON :Type(property)
 1. CREATE INDEX ON :Person(name)
 2. MATCH (p:Person) USING INDEX p:Person(name)
WHERE p.name= 'Robin Williams' RETURN n
- Definición de restricciones de integridad:
 - <CREATE | DROP> CONSTRAINT ON
(variable:Type) ASSERT <variable.propiedad>
IS UNIQUE
 3. CREATE CONSTRAINT ON (p:Subject)
ASSERT p.title IS UNIQUE

Otras operaciones

■ Cálculo de agregados:

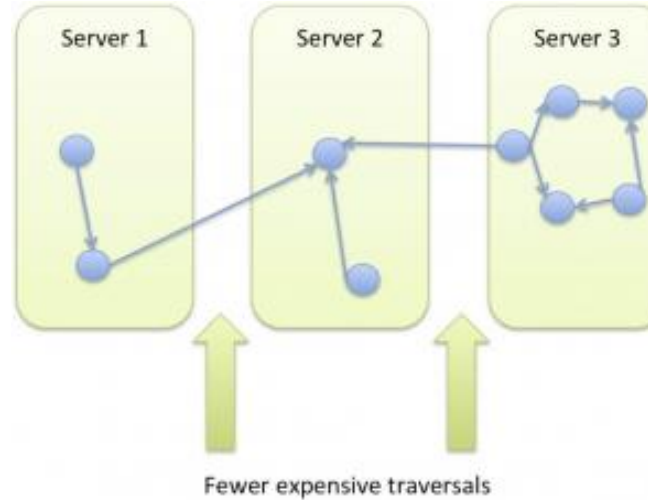
1. MATCH (p:Person)
RETURN p.born, count(*)
2. MATCH (p:Person)--(f:Movie)
RETURN DISTINCT f.title, count(*)



Estrategias de distribución

Distribución de datos

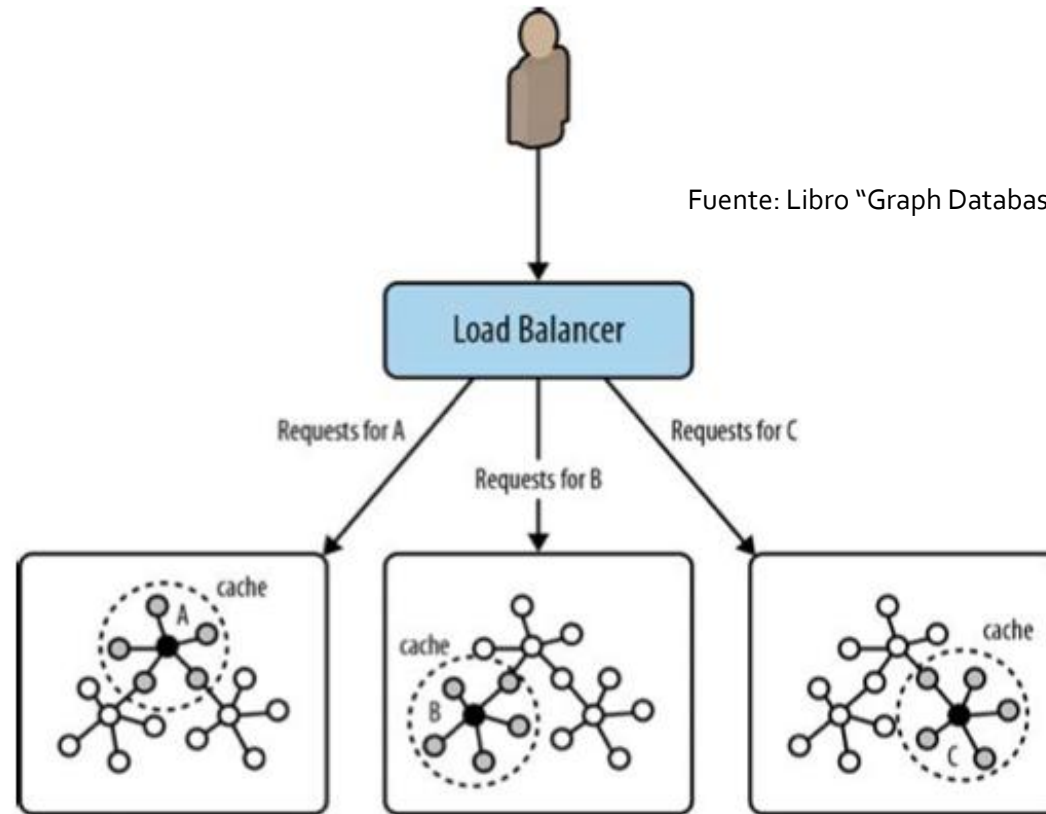
- La correcta fragmentación en los modelos de grafo es complicada, debido a:
 - La mutabilidad de los grafos
 - El alto número de relaciones entre nodos
 - Debe evitarse que la navegación por el grafo atraviese un gran número de servidores.



Fuente: <http://jim.webber.name/2011/02/on-sharding-graph-databases/>

Arquitectura de distribución

- Graph sharding:
 - Facilita la escalabilidad horizontal.
 - No se fragmentan los datos en distintos servidores, sino en sus cachés.





Recursos y enlaces

Lecturas

- Página oficial de Neo4j: <http://www.Neo4j.org/>
- Documentación oficial: <http://docs.Neo4j.org/>
- Reference Card de Cypher: <http://docs.Neo4j.org/refcard/2.1.1/>
- Ian Robinson, Jim Webber & Emli Eifrem (2013). Graph Databases. O'Reilly.
- E. Redmond, J. Wilson (2012). Seven Databases in Seven Weeks: A Guide to Modern Databases and the NoSQL Movement. Pragmatic Bookshelf.
- P.J. Sadalage & M. Fowler. (2013). NoSQL Distilled. A brief Guide to the Emerging World of Polyglot Persistence, Pearson Education.