

# Arquitecturas de Bases de NoSQL

II Congreso de Datos Abiertos y Metadatos

**CODAM 2022**

**DATOS ABIERTOS  
EN LA CIENCIA** Ecuador



26 al 28 de octubre de 2022 / Modalidad Híbrida

Miguel Ponce  
UOC/SRI, Quito,  
mponce1@uoc.edu

ORGANIZADORES

[data-lat.org](http://data-lat.org)



AUSPICIANTES

**CS&S** Code for Science & Society

**Chan  
Zuckerberg  
Initiative**

GORDON AND BETTY  
**MOORE**  
FOUNDATION

# SUMARIO/CONTENIDO:

- Introducción
- Mongo DB
- Neo 4j
- Conclusiones
- Referencias
- Agradecimientos

Octubre, 2022

ORGANIZADORES

[data-lat.org](http://data-lat.org)



AUSPICIANTES

**CS&S** Code for Science & Society

**Chan  
Zuckerberg  
Initiative**

GORDON AND BETTY  
**MOORE**  
FOUNDATION

# Origen del término

En 1998 aparece por primera vez la referencia NoSQL( Carlo Strozzi)

En 2009 re-aparece el término, se utiliza de forma generalizada, se referirse a:

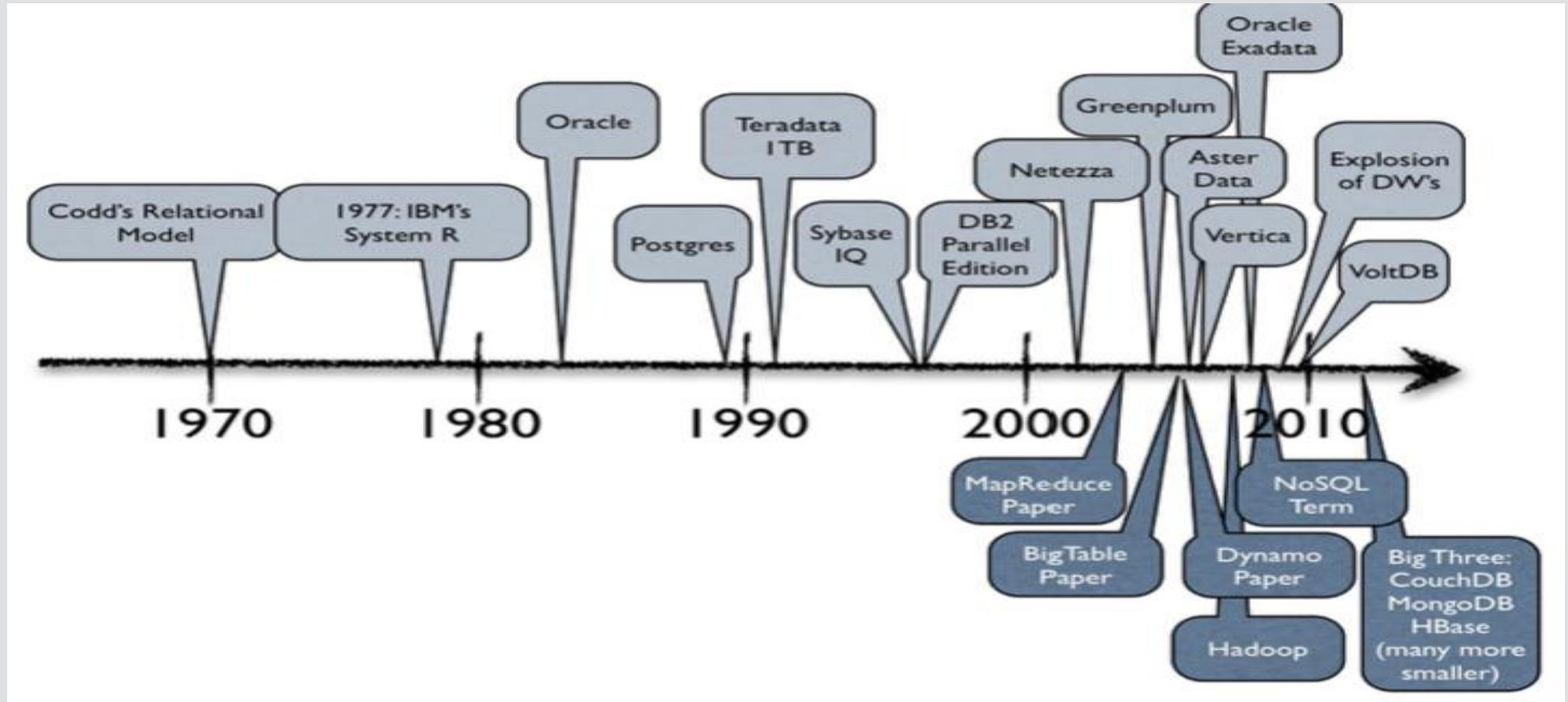
- SGBD, no relacionales
- Distribuidos

¿NoSQL es polémica?

# Hechos relevantes

- El año 2000 el profesor Eric A. Brewer(CAP).
- En el año 2002 se prueba formalmente su validez(MIT).
- Aparecen nuevas necesidades en las empresas y con ellas nuevos proyectos para soportarlas.
- Entre el año 2000 y 2010 aparecen los primeros proyectos NoSQL(1989).

# Hechos relevantes



# Características de NoSQL

- ❑ No ofrecen SQL como lenguaje estándar.
- ❑ Esquema flexible (schemaless)
- ❑ No garantizan las propiedades ACID al completo.
- ❑ Favorecen la escalabilidad.
- ❑ Ofrecen solución a los inconvenientes y rigidez del modelo relacional.
- ❑ Distribuidas/De código abierto

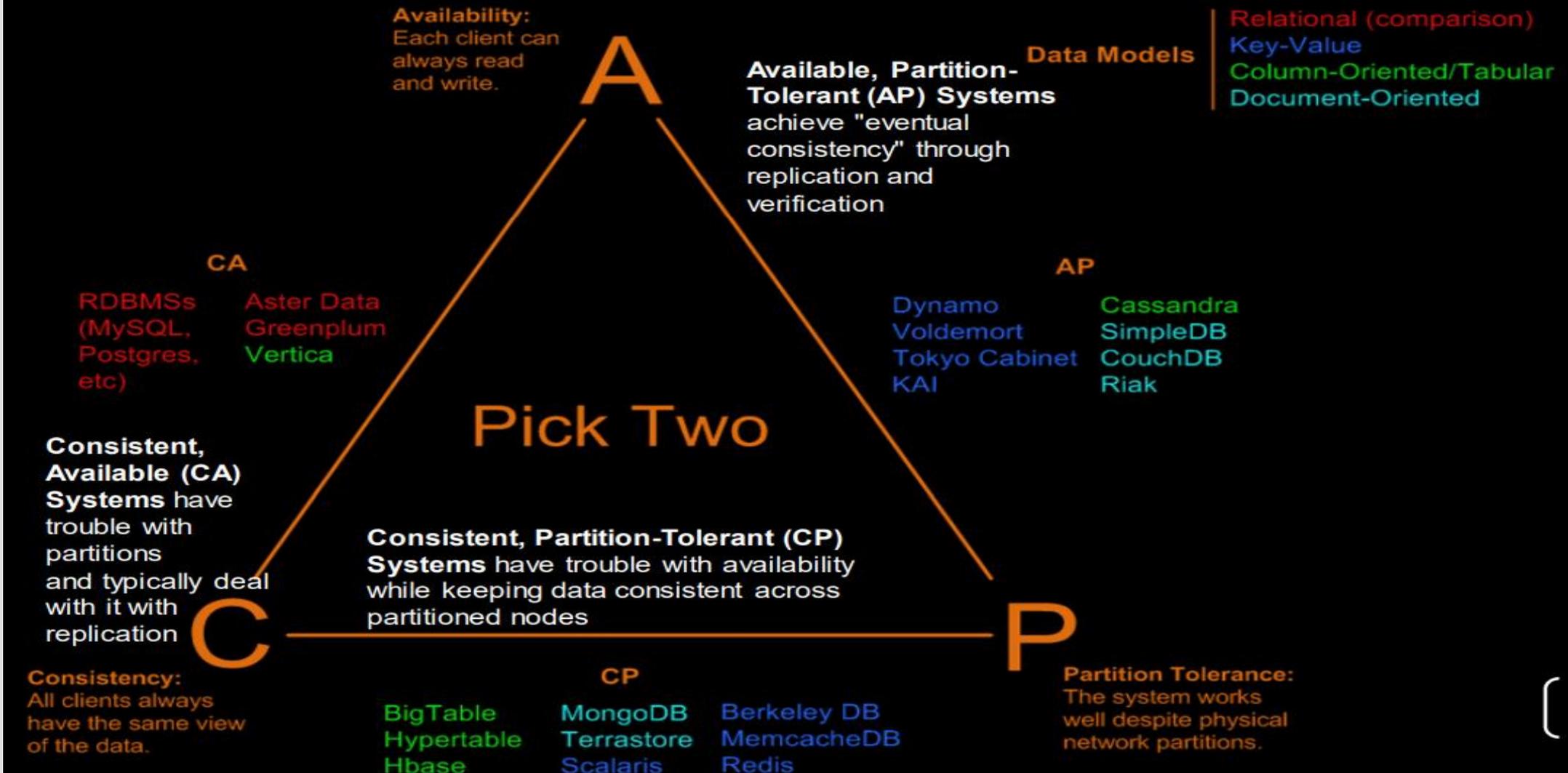
# Las aplicaciones y los datos

- ❑ Datos con estructura variable.
- ❑ Múltiples orígenes de datos(formato).
- ❑ Grandes volúmenes de datos
- ❑ Flexibilidad/rendimiento.
- ❑ Procesamiento en “tiempo real”.

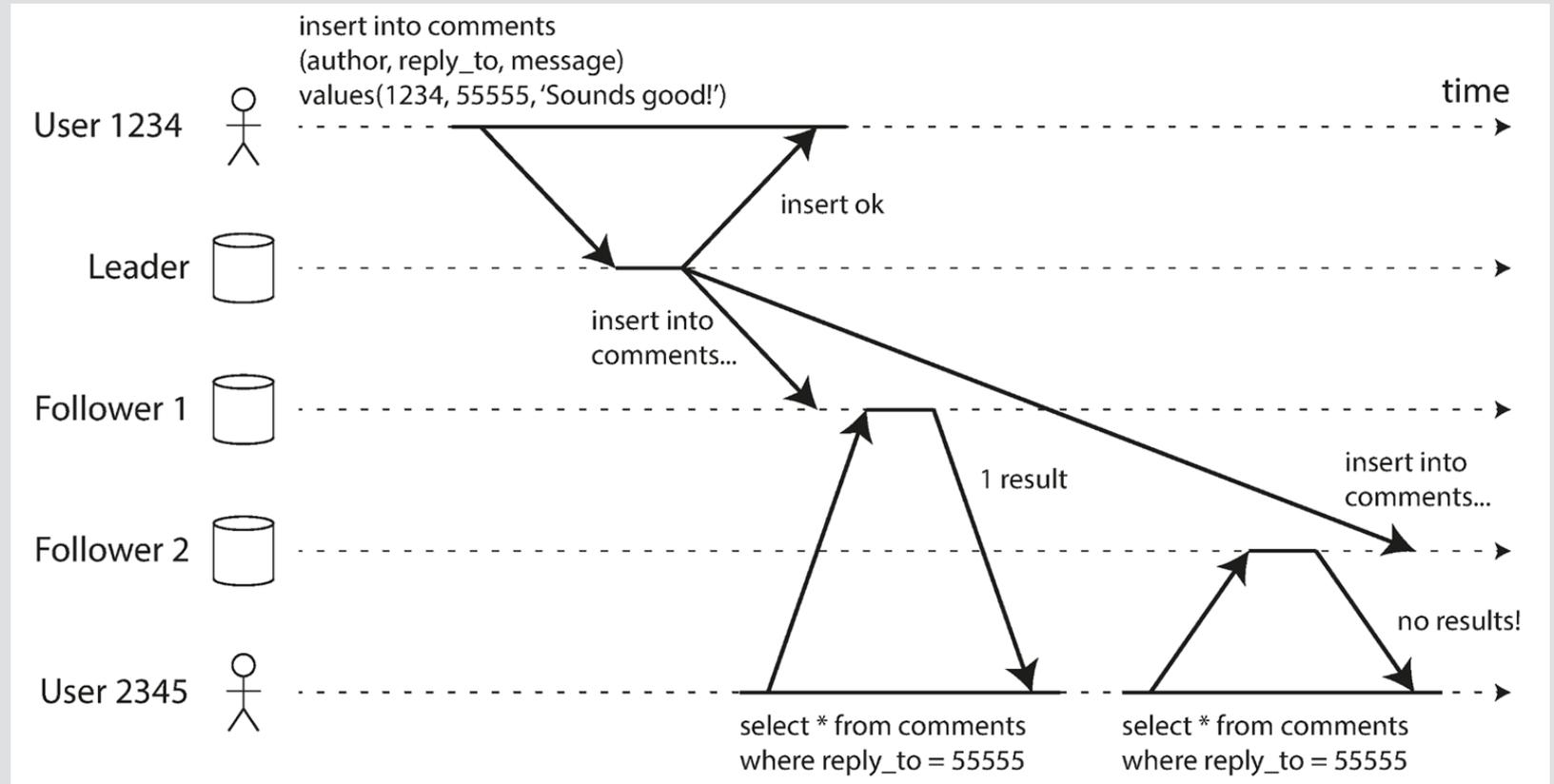
# Teorema CAP(Eric A. Brewer) - 2/3

- ❑ Consistencia(C): Todos los nodos deben garantizar la misma información al mismo tiempo(I,U,D).
- ❑ Disponibilidad(A): Independientemente si uno de los nodos se ha caído, el sistema debe seguir en funcionamiento(R/W).
- ❑ Tolerancia al Particionado de Red(P): El sistema debe estar disponible así existan problemas de comunicación.

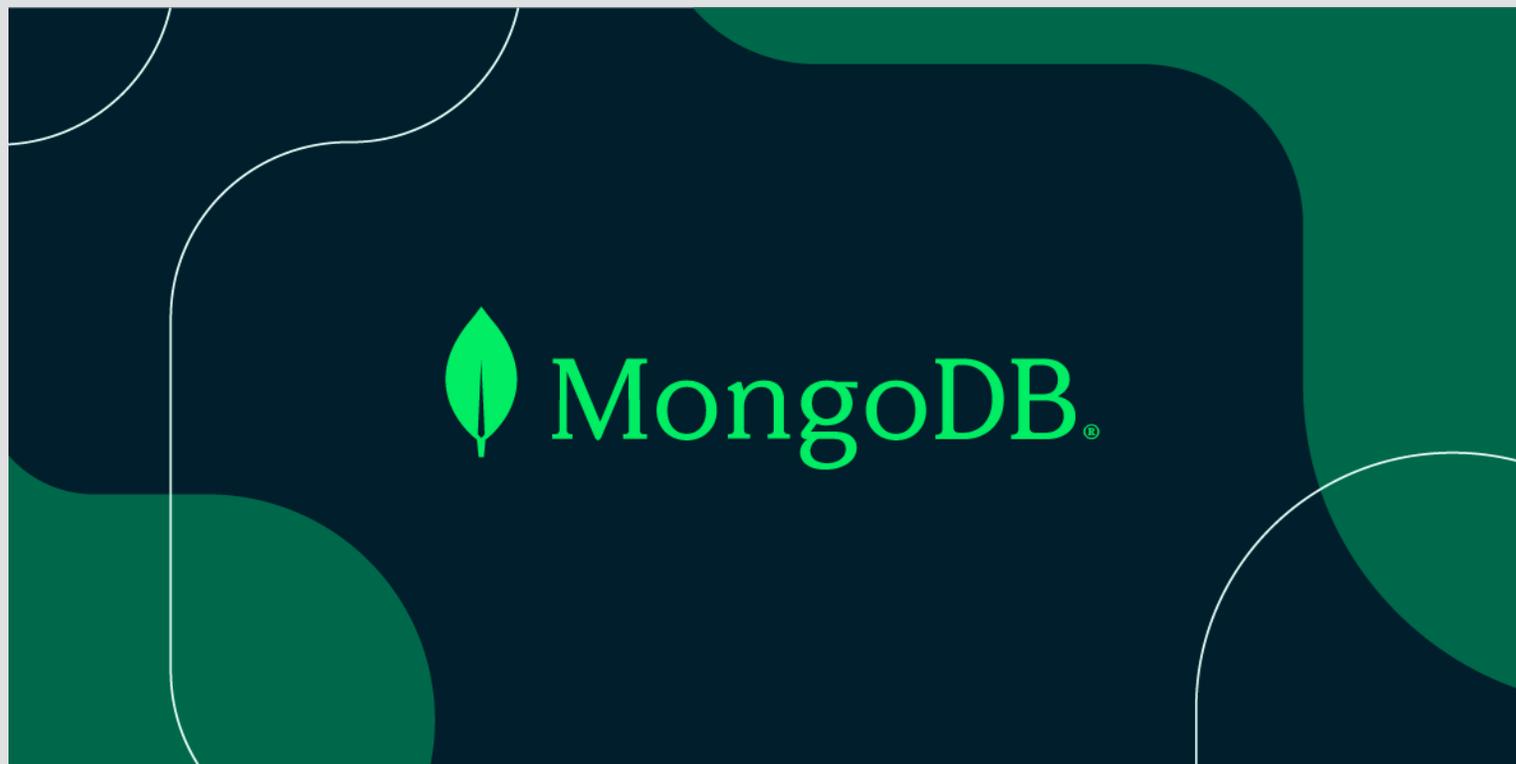
# Visual Guide to NoSQL Systems



# Lectura recomendada



# Introducción



Octubre, 2022

ORGANIZADORES

[datalat.org](https://datalat.org)



AUSPICIANTES



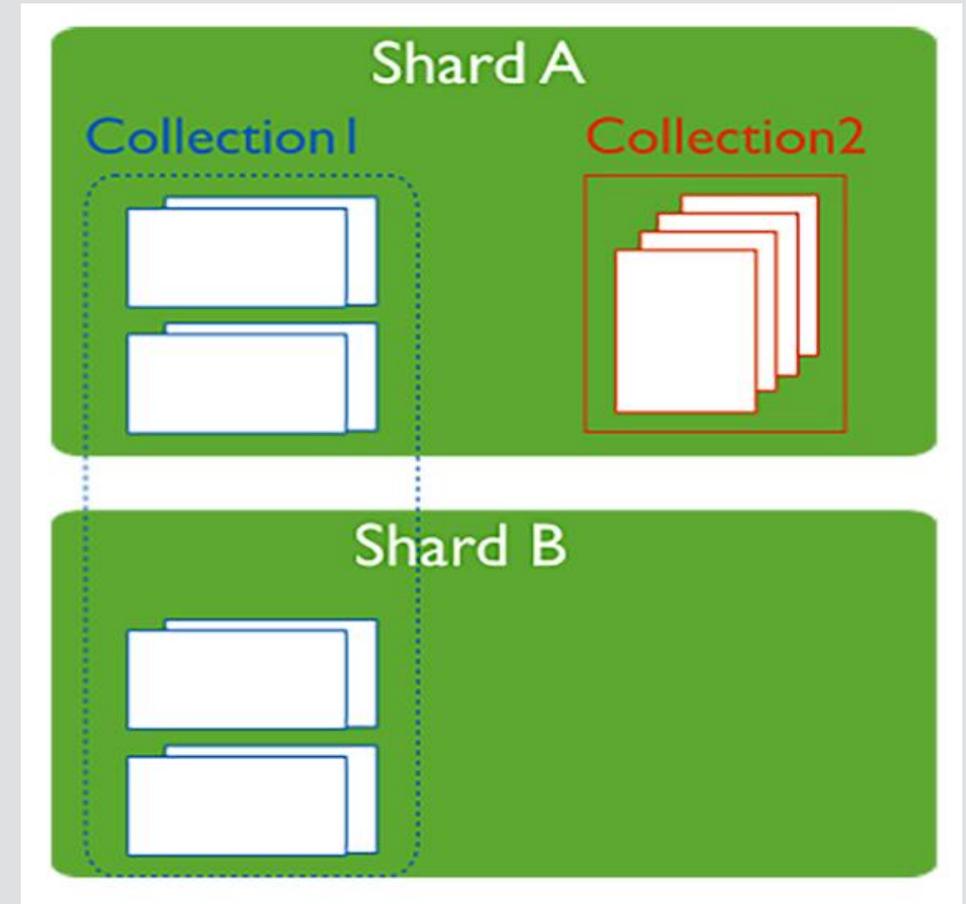
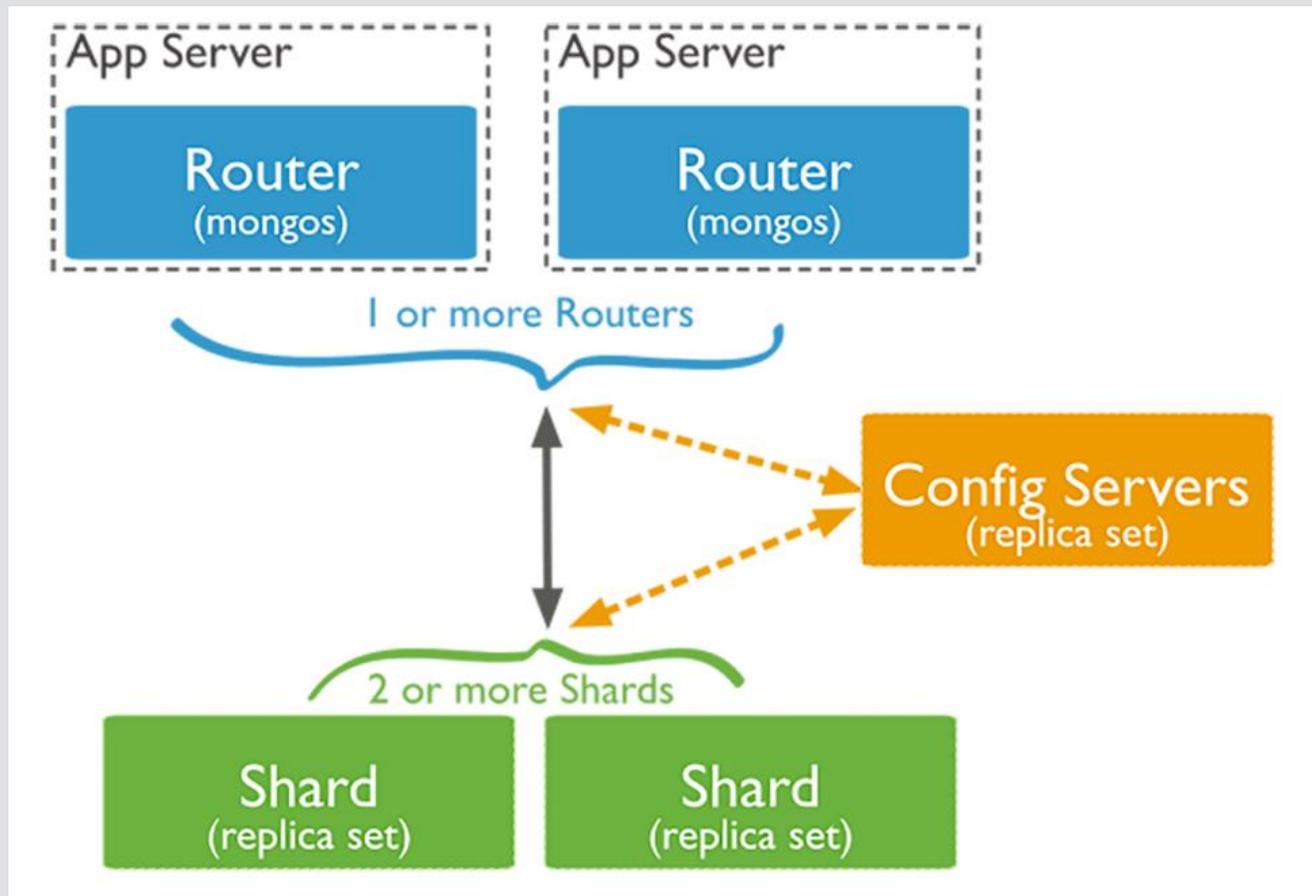
# Introducción

- ❑ BD de documentos.
- ❑ Modelo de consultas e indexación avanzado.
- ❑ Utiliza una arquitectura basada en agregados.
- ❑ Almacena datos en documentos flexibles(JSON).
- ❑ Indexación y agregación.
- ❑ Distribuida en su núcleo(escalabilidad/flexibilidad).
- ❑ MongoDB es de uso gratuito.

# Sharding

- ❑ Método para distribuir datos entre varias máquinas.
- ❑ MongoDB usa fragmentación para conjuntos de datos muy grandes y operaciones de alto rendimiento.

# Sharding



RDBMS		MongoDB
Database	➔	Database
Table, View	➔	Collection
Row	➔	Document (BSON)
Column	➔	Field
Index	➔	Index
Join	➔	Embedded Document
Foreign Key	➔	Reference
Partition	➔	Shard

# Actividad recomendada



RECOMMENDED

M001

## MongoDB Basics

Set up your MongoDB database and learn to search, create, and analyz...

[View Course](#)



**ALLISON B.**

Awarded by

**MongoDB, Inc.**

as a

**MongoDB Certified DBA**  
Associate Level

LICENSE # 000-000-000  
*MongoDB 4.2*

ISSUE DATE 2021-01-01  
*This certification has no expiration*

Verify this professional certification at  
[https://university.mongodb.com/certification/verify\\_certificate](https://university.mongodb.com/certification/verify_certificate)



Octubre, 2022

ORGANIZADORES

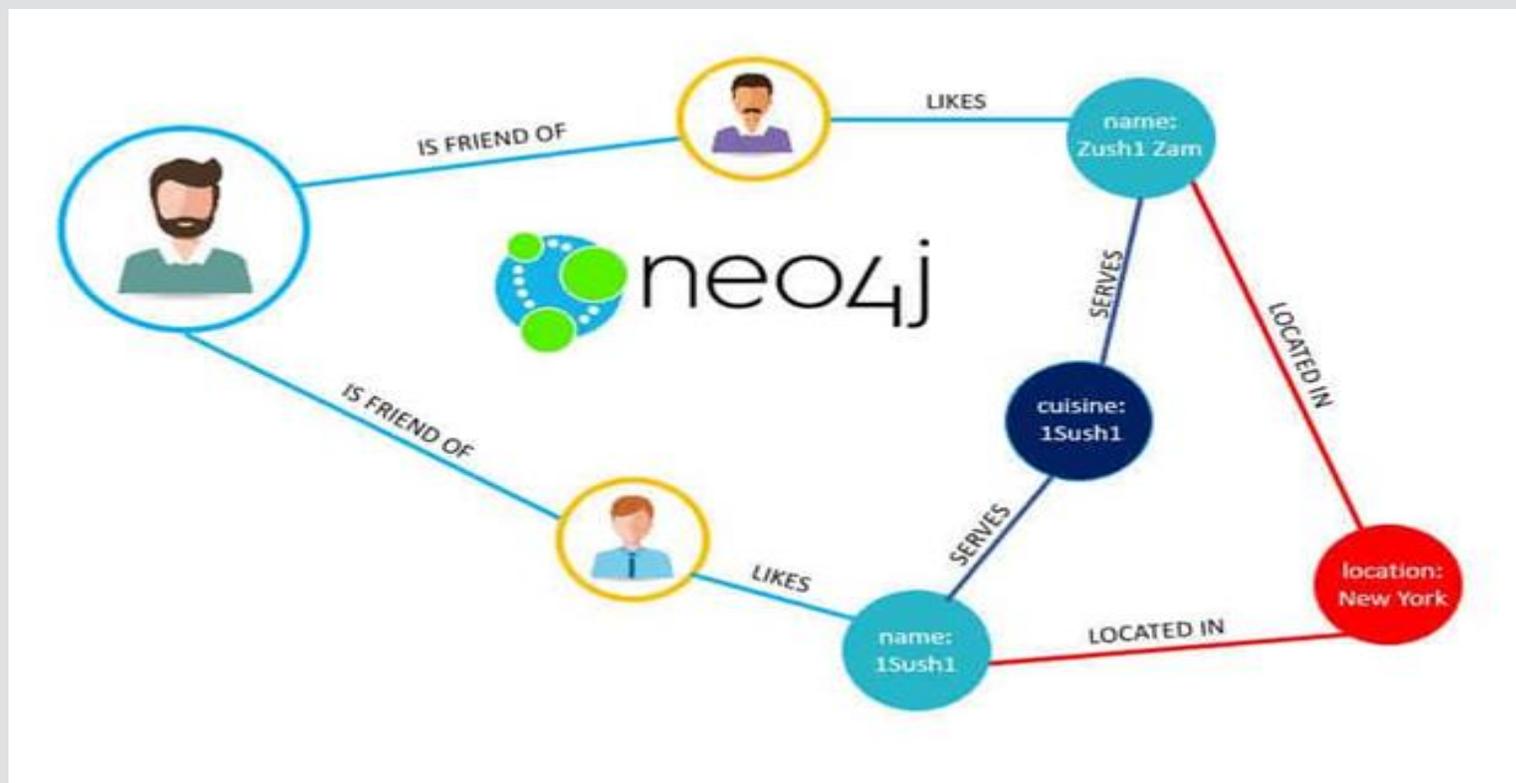
[datalat.org](https://datalat.org)



AUSPICIANTES



# Neo4j



Octubre, 2022

ORGANIZADORES

[data-lat.org](http://data-lat.org)



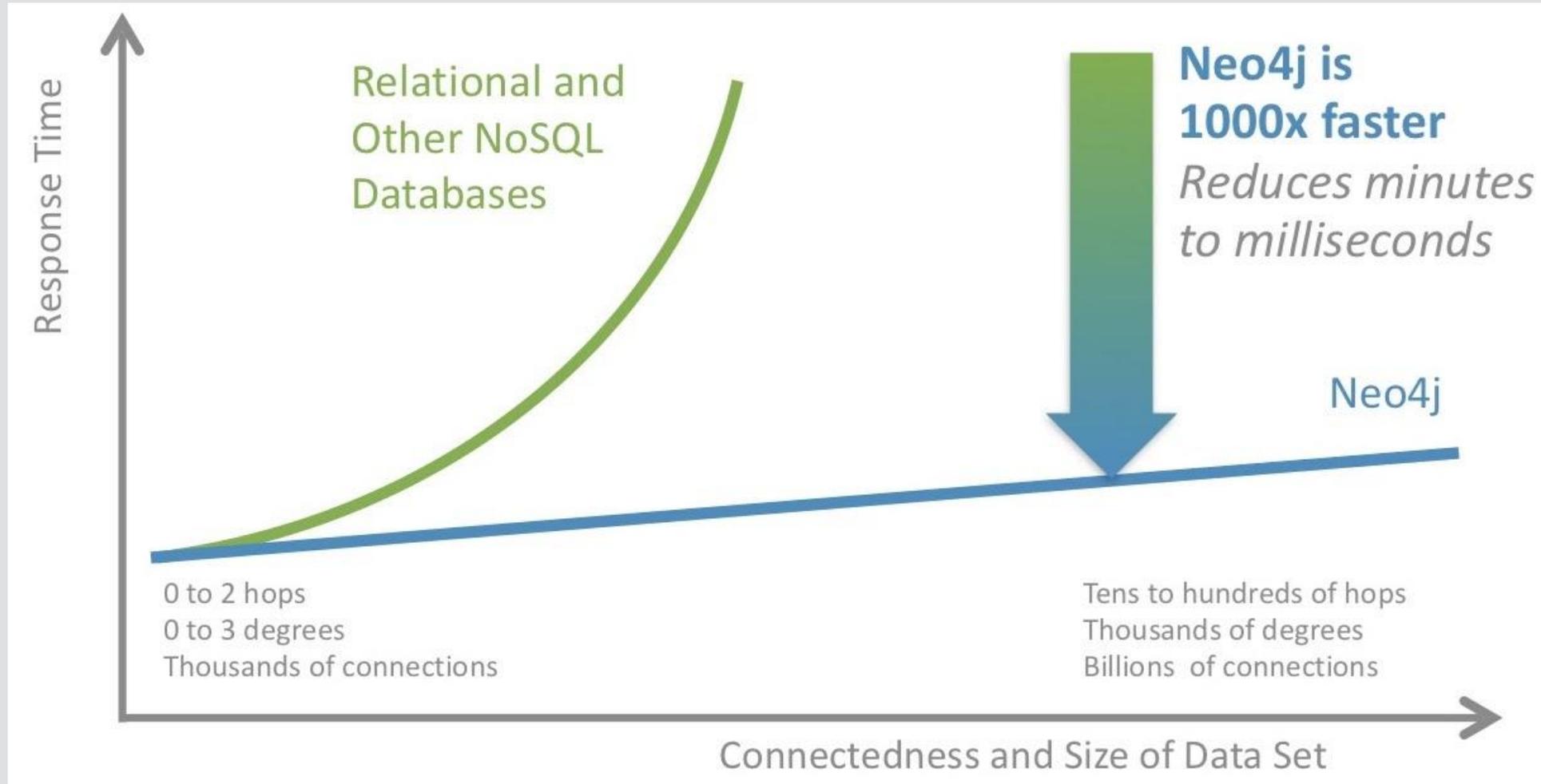
AUSPICIANTES



# Introducción

- ❑ BD de grafos.
- ❑ Se basa en la teoría de grafos.
- ❑ Modelo basado en búsqueda heurística y otros .
- ❑ Utiliza una arquitectura basada en nodos y relaciones.
- ❑ Ofrece una versión de uso gratuita.

# Comparativa



# Componentes

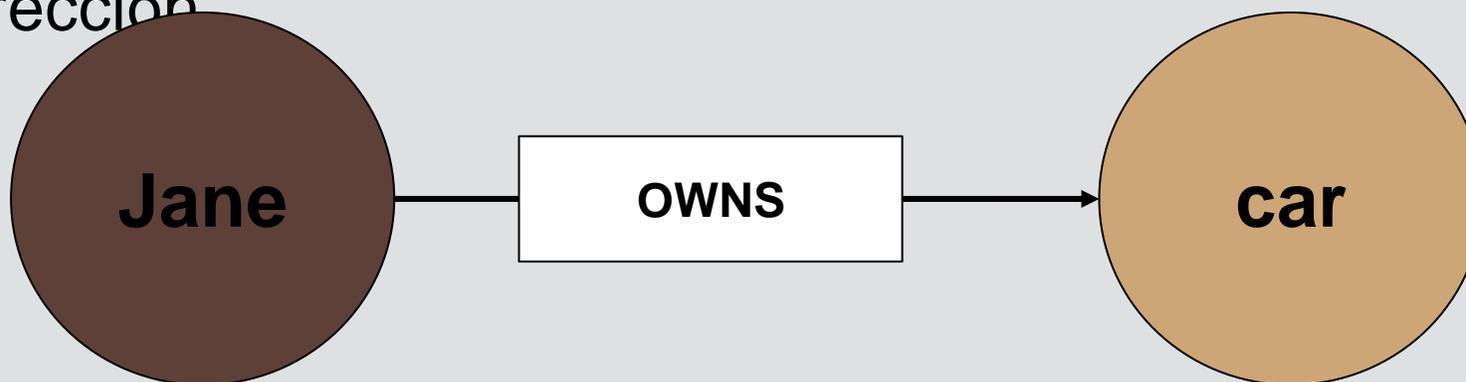
## **Nodo** (Vertex)

El principal elemento de datos a partir del cual se construyen los grafos.

## **Relación** (Edge)

Un enlace entre dos nodos.

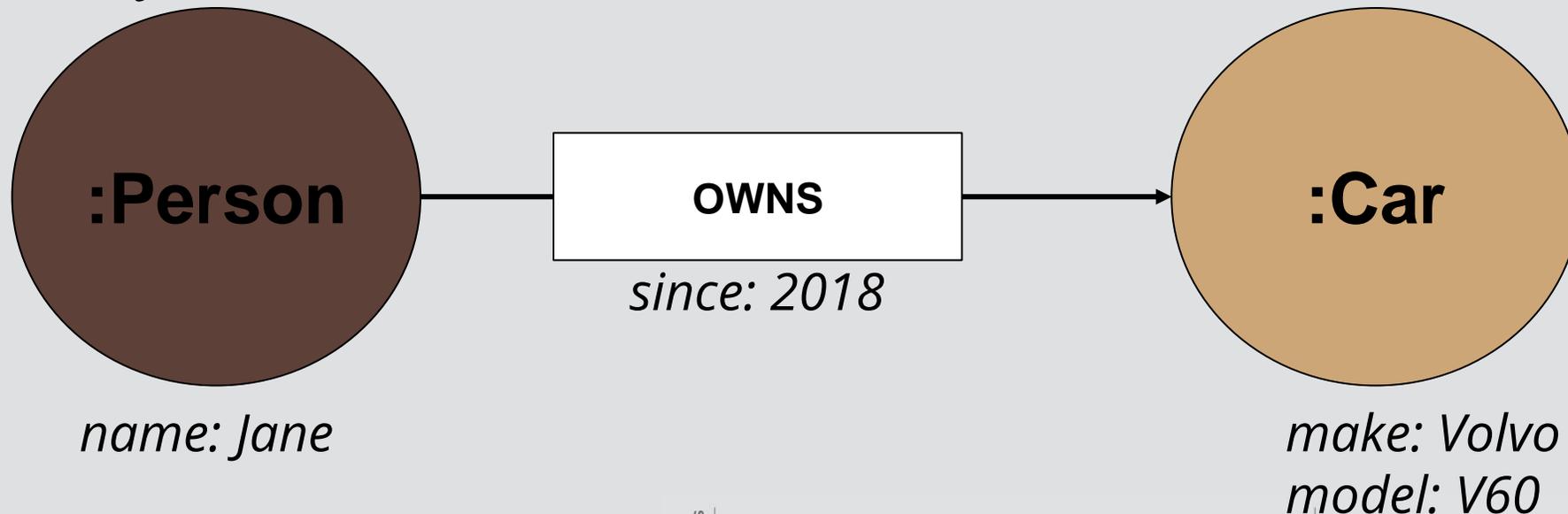
Tiene dirección



**Nota: Se permite un nodo sin relaciones.**

# Etiquetado/Propiedades

- ❑ Definir categoría de nodo (opcional)
- ❑ Puede tener más de uno
- ❑ Enriquecer un nodo o una relación
- ❑ ¡No hay necesidad de valores nulos!



# Algoritmos por categoria



## Pathfinding & Search

- Parallel Breadth First Search & Depth First Search
- Shortest Path
- Single-Source Shortest Path
- All Pairs Shortest Path
- Minimum Spanning Tree
- A\* Shortest Path
- Yen's K Shortest Path
- K-Spanning Tree (MST)
- Random Walk



## Centrality / Importance

- Degree Centrality
- Closeness Centrality
- CC Variations: Harmonic, Dangalchev, Wasserman & Faust
- Betweenness Centrality
- Approximate Betweenness Centrality
- **PageRank**
- Personalized PageRank
- ArticleRank
- Eigenvector Centrality



## Community Detection

- Triangle Count
- Clustering Coefficients
- **Weakly Connected Components**
- Strongly Connected Components
- **Label Propagation**
- **Louvain Modularity**
- K1 coloring
- Modularity optimization



## Similarity

- **Node Similarity**
- Euclidean Distance
- Cosine Similarity
- Jaccard Similarity
- Overlap Similarity
- Pearson Similarity



## Link Prediction

- Adamic Adar
- Common Neighbors
- Preferential Attachment
- Resource Allocations
- Same Community
- Total Neighbors

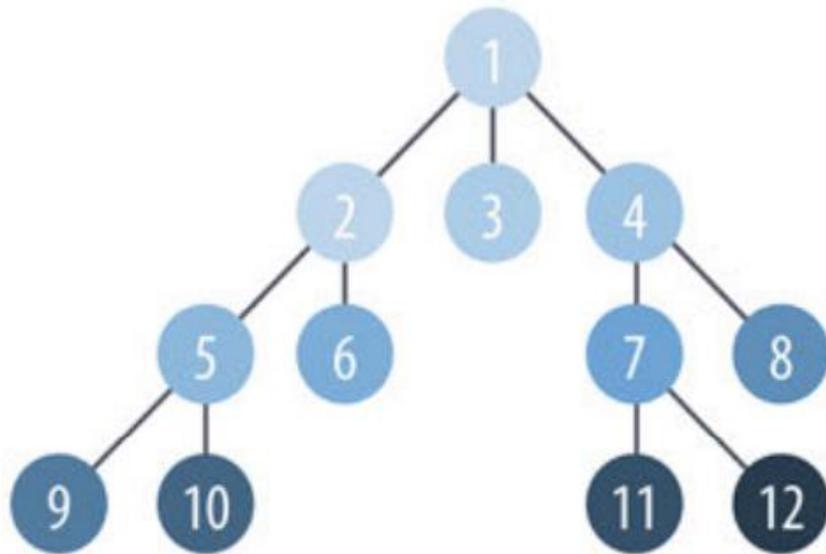
...and also:

- Random graph generation
- One hot encoding

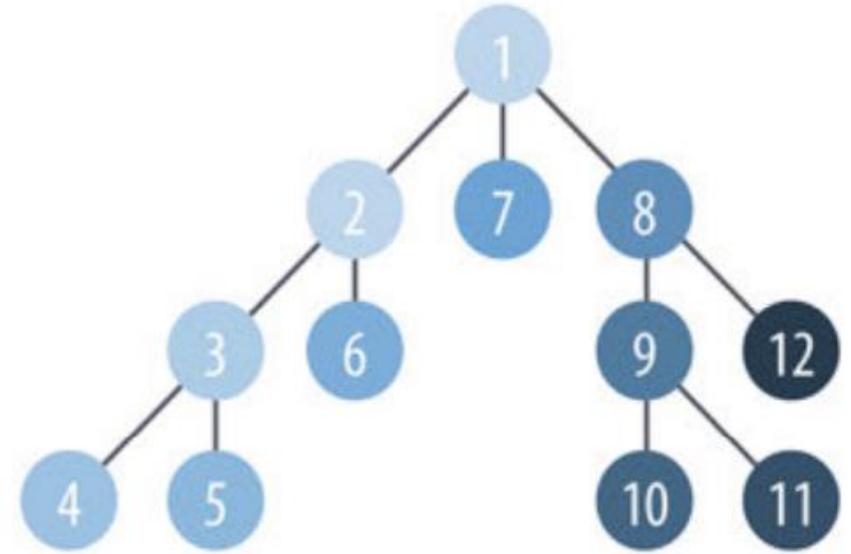


# Tipos Búsquedas

## Graph Search Algorithms



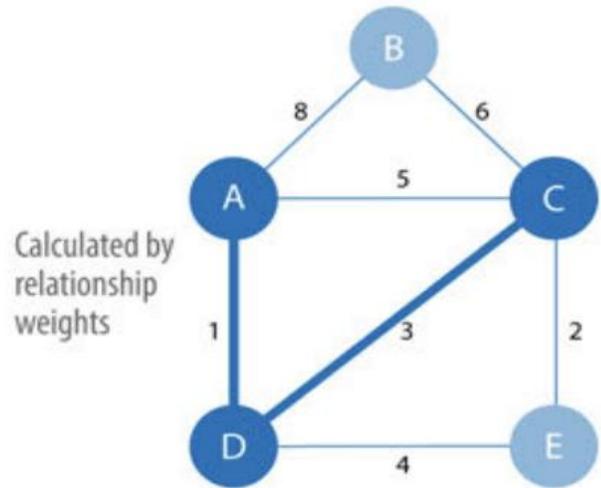
**Breadth First Search**  
Visits nearest neighbors first



**Depth First Search**  
Walks down each branch first

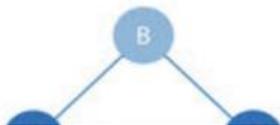
# Tipos Búsquedas

## Pathfinding Algorithms



### Shortest Path

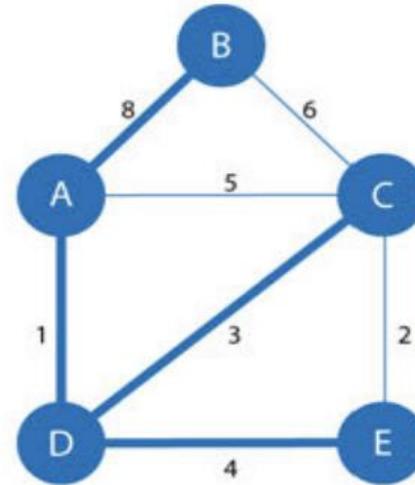
Shortest path between 2 nodes (A to C shown)



(A, B) = 8  
 (A, C) = 4 via D  
 (A, D) = 1  
 (A, E) = 5 via D  
 (B, C) = 6  
 (B, D) = 9 via A or C  
 And so on...

### All-Pairs Shortest Paths

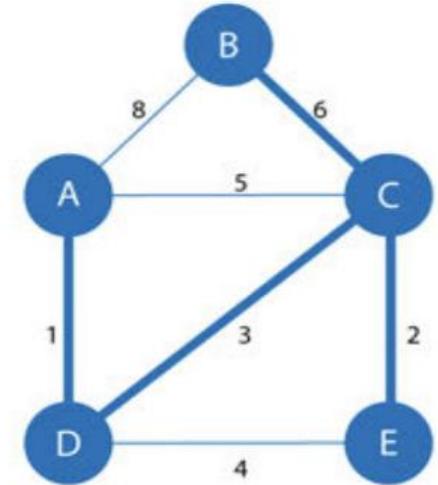
Optimized calculations for shortest paths from all nodes to all other nodes



### Single Source Shortest Path

Shortest path from a root node (A shown) to all other nodes

Traverses to the next unvisited node via the lowest cumulative weight from the root



### Minimum Spanning Tree

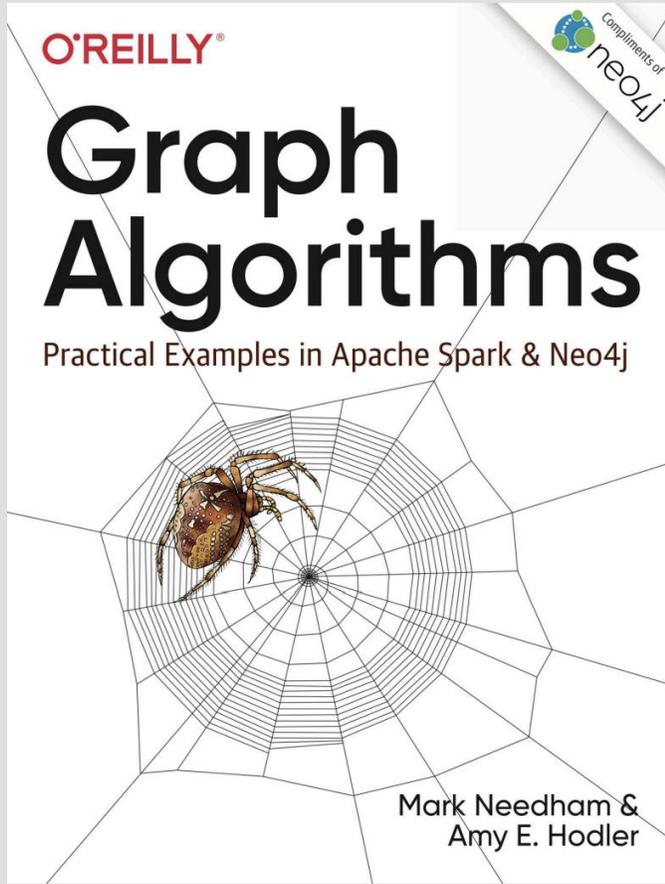
Shortest path connecting all nodes (A start shown)

Traverses to the next unvisited node via the lowest weight from any visited node

Figura 2. Comparativa entre el modelo relacional y el modelo basado en grafos de Neo4j



# Actividad recomendada



THE FASTEST PATH TO GRAPH

## Neo4j AuraDB

Neo4j's fully managed cloud service: The zero-admin, always-on graph database for cloud developers.

Start Free

Octubre, 2022

ORGANIZADORES

[datalat.org](http://datalat.org)



AUSPICIANTES

**CS&S** Code for Science & Society

**Chan Zuckerberg Initiative**

GORDON AND BETTY  
**MOORE**  
FOUNDATION

# Conclusiones

Octubre, 2022

ORGANIZADORES

[data-lat.org](https://data-lat.org)



AUSPICIANTES



# Referencias

<https://learning.oreilly.com/library/view/graph-algorithms/9781492047674/>

<https://neo4j.com/developer/download-materials/>

<https://www.mongodb.com/docs/manual/introduction/>

<https://www.ccs.neu.edu/home/kathleen/classes/cs3200/20-NoSQLMongoDB.pdf>

UOC/M2.852 - Arquitecturas de bases de datos no tradicionales



Octubre, 2022

ORGANIZADORES

[datalat.org](https://datalat.org)



AUSPICIANTES



CO-ORGANIZADORES



ALIADOS DE DIFUSIÓN



ORGANIZADORES



AUSPICIANTES



Octubre, 2022