

# Introducción a Bases de Datos NoSQL

Bases de Datos 2022

# Base de Datos NoSQL

Las Bases de datos NoSQL son **bases de datos no relacionales**.

- Proveen **modelos de datos diferentes** al modelo relacional (tablas)
  - Modelo de Documentos, Grafos, ...
- Proveen **lenguajes de consultas distintos** a SQL
  - MQL, Ciper, CQL, ...

Emergen a fines de la década de 2000 como respuesta a nuevas necesidades de las empresas

- Necesidad de almacenar, acceder y procesar grandes volúmenes de datos
- Limitaciones de la bases de datos relacionales tradicionales

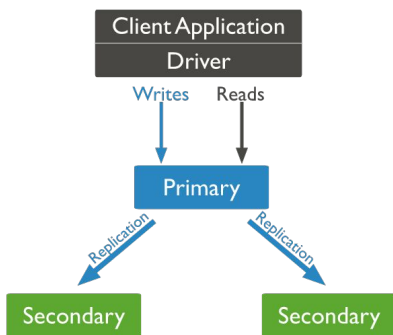
# Características de NoSQL

## Esquemas Flexibles

- No imponen un esquema predefinido
  - Datos semiestructurados o no estructurados
- Estructuras de datos polimórficas

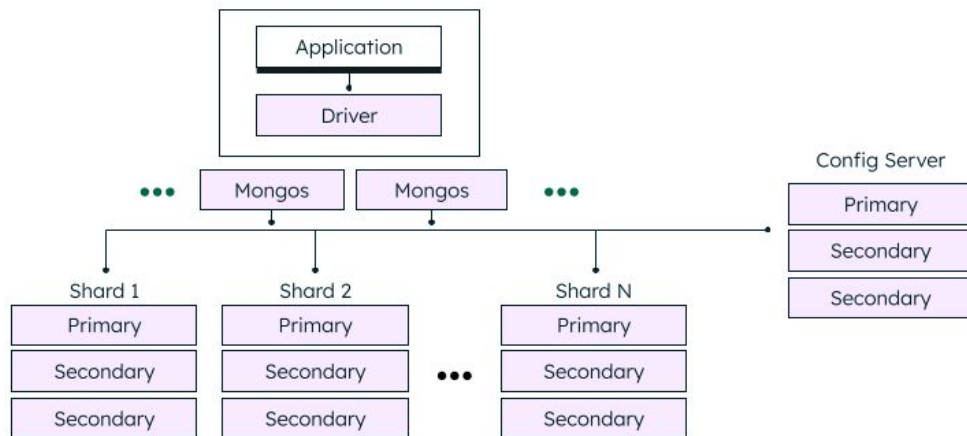
## Alta disponibilidad

- Réplica de los datos en varios nodos



## Escalabilidad horizontal

- Partición de los datos (sharding)



# Características de NoSQL

## Integridad Referencial

- Usualmente no proveen FKs
- Delegan la responsabilidad al desarrollador

## JOINS

- Usualmente no proveen JOINS o
- Recomendado sólo para consultas de análisis

## Desnormalización

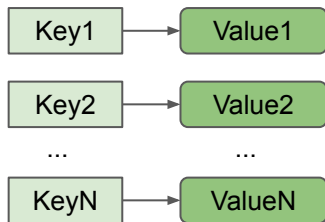
- Es una práctica usual tener duplicados

## Métodos de Diseño Query-driven

- Las **consultas** son requeridas para diseñar modelos de datos optimizados
- Los **requisitos no funcionales** (NFRs) de la aplicación son necesarios para el diseño
- Los NFRs son considerados para la selección de la BD más adecuada

# Tipos de BD NoSQL: Base de datos Clave-Valor

## Modelo de datos



## Características

- Esquema de datos super flexibles
- In-memory
- Tipos de datos
  - **Memcached**: Strings
  - **Redis**: Strings, Hashes, Lists, Sets, Sorted Sets

## Lenguaje de consulta: Redis

> **sadd** venues "Olympic Stadium" "Nippon Budokan" "Tokyo Stadium"

> **sscan** venues 0 match \*

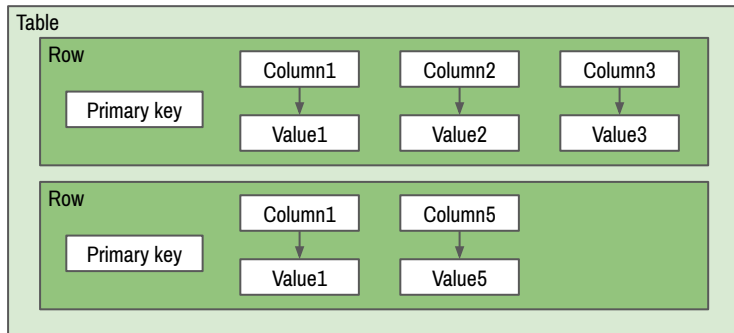
> **sismember** venues "Eiffel Tower"

## Casos de usos

- Almacenar datos en **cache** (caching)
- Gestión de **sesiones**
- Deduplicate

# Tipos de BD NoSQL: Base de datos Wide-column

## Modelo de datos



## Lenguaje de consulta: CQL

```
ALTER TABLE users ADD todo map<timestamp, text>;  
UPDATE users SET todo = { '2012-9-24' : 'enter mordor',  
                          '2014-10-2 12:00' : 'throw ring into mount doom' }  
WHERE user_id = 'frodo';  
  
SELECT user_id, todo FROM users WHERE user_id = 'frodo';
```

## Características

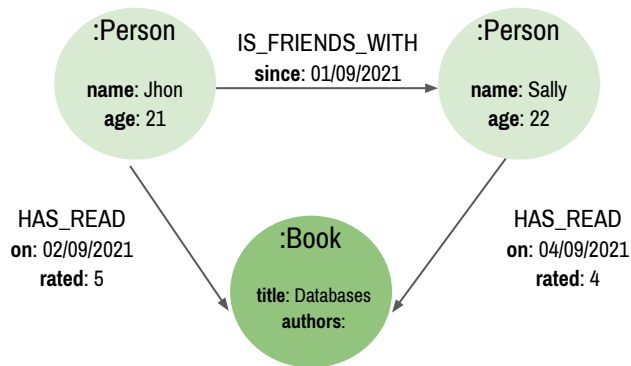
- Agregar columnas con zero downtime
- Tipos de datos
  - Cassandra: incluye native types, collection types, user-defined types, tuple types

## Casos de usos

- Las **escrituras** superan a las lecturas
- **Logs** de transacciones
- **IoT**

# Tipos de BD NoSQL: Base de datos de Grafos

## Modelo de datos



## Lenguaje de consulta: CypHer

*//find the top 3 people who have the most friends*

```
MATCH (p:Person)-[r:IS_FRIENDS_WITH]-(other:Person)
```

```
RETURN p.name, count(other.name) AS numberOfFriends
```

```
ORDER BY numberOfFriends DESC
```

```
LIMIT 3
```

## Características

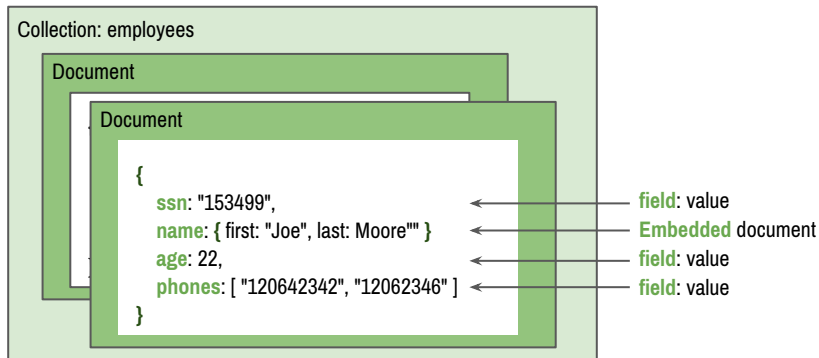
- Estructuras de grafos: nodos y relaciones
- Tipos de datos
  - Neo4j: incluye property types, composite types

## Casos de usos

- Redes sociales
- Análisis y detección de fraudes
- Sistema de recomendaciones

# Tipos de BD NoSQL: Base de datos de Documentos

## Modelo de datos



## Lenguaje de consulta: MQL

```
//find the employees whose last name starts with letter 'M'

db.employees.find(

  { name.last: /^M/},           //Query filter

  { ssn: 1, name: 1, age: 1 }   //Project fields

).sort( { age: -1 } ).limit( 10 )
```

## Características

- Estructuras de datos similares a JSON
- Relaciones obvias usando arrays y documentos anidados
- Tipos de datos
  - MongoDB: [incluye](#) numbers, strings, date, arrays, objects

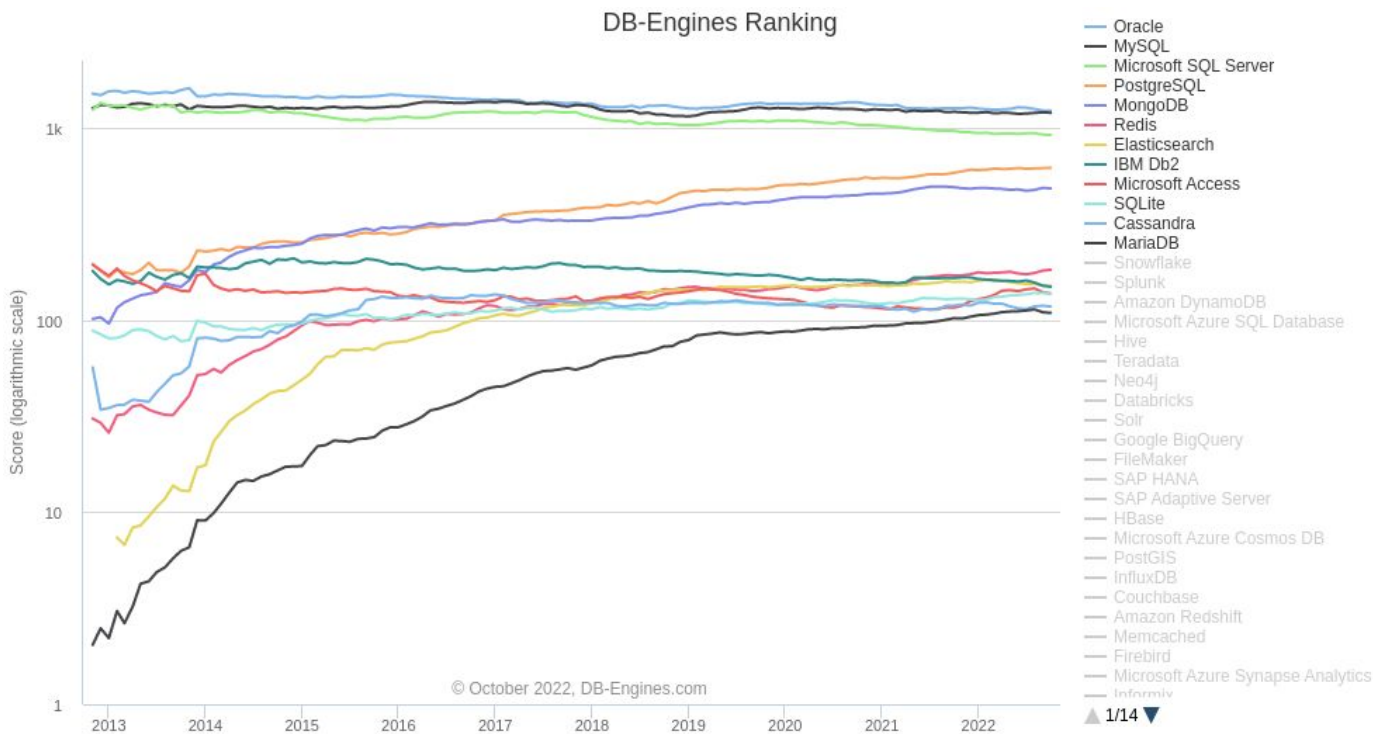
## Casos de usos

- De proposito general
- Catalogos
- IoT
- Gestión de contenidos



# DB-Engines Ranking

➤ 397 bases de datos en el [ranking](#)



# ¿Cómo elegir la base de datos adecuada?



# Polígloa Persistencia

## Polyglot Persistence within a single application

