

Ing. Víctor A. Fuentes T., MSc.

# Base de Datos NoSQL (No Relacionales)

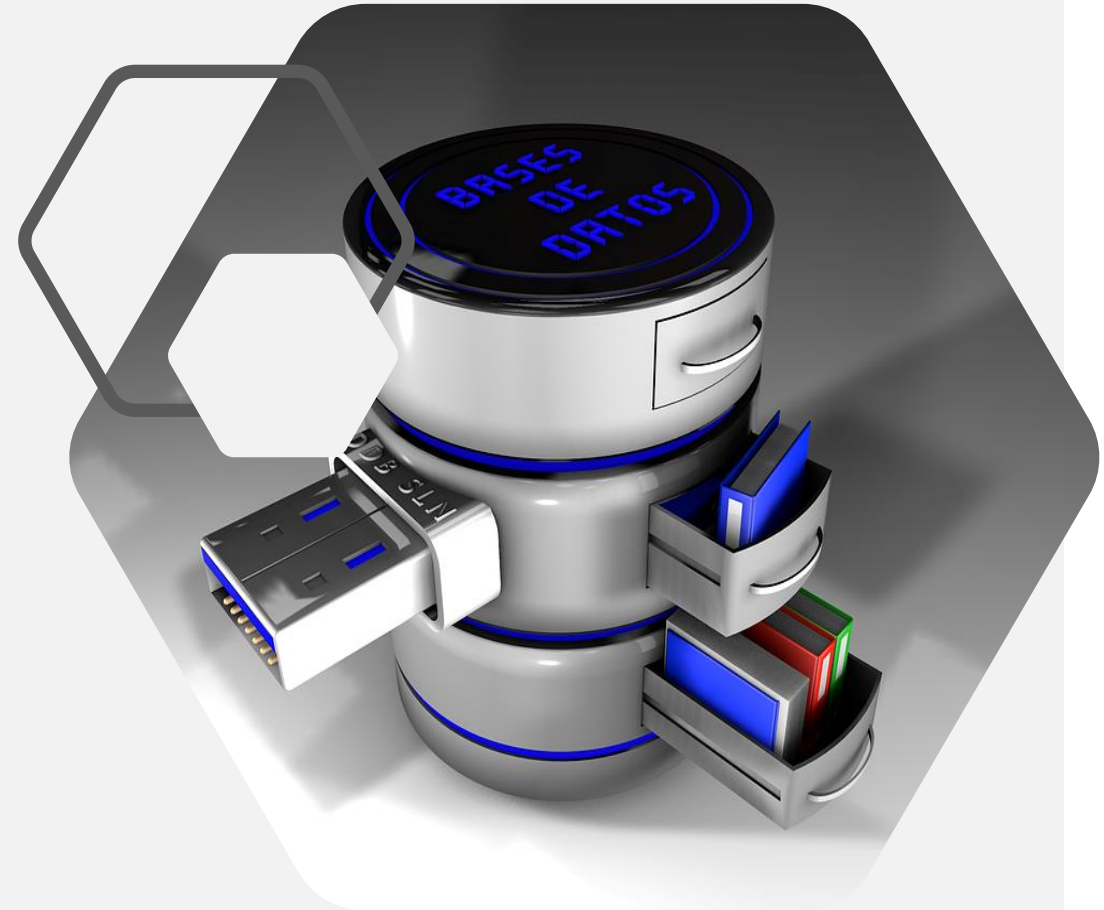


# Base de Datos – Not Only SQL (NoSQL)

Surgen de la necesidad de gestionar grandes cantidades de información y se distinguen porque no cumplen con el tradicional esquema entidad-relación.

Son más flexibles, ya que suelen permitir almacenar información en otros formatos como clave-valor similar a tablas Hash, Mapeo de Columnas, Documentos o Grafos.

Las bases de datos NoSQL se puede definir como “un conjunto de conceptos que permite el procesamiento rápido y eficiente de conjuntos de datos con un enfoque en el rendimiento, la confiabilidad y la agilidad”



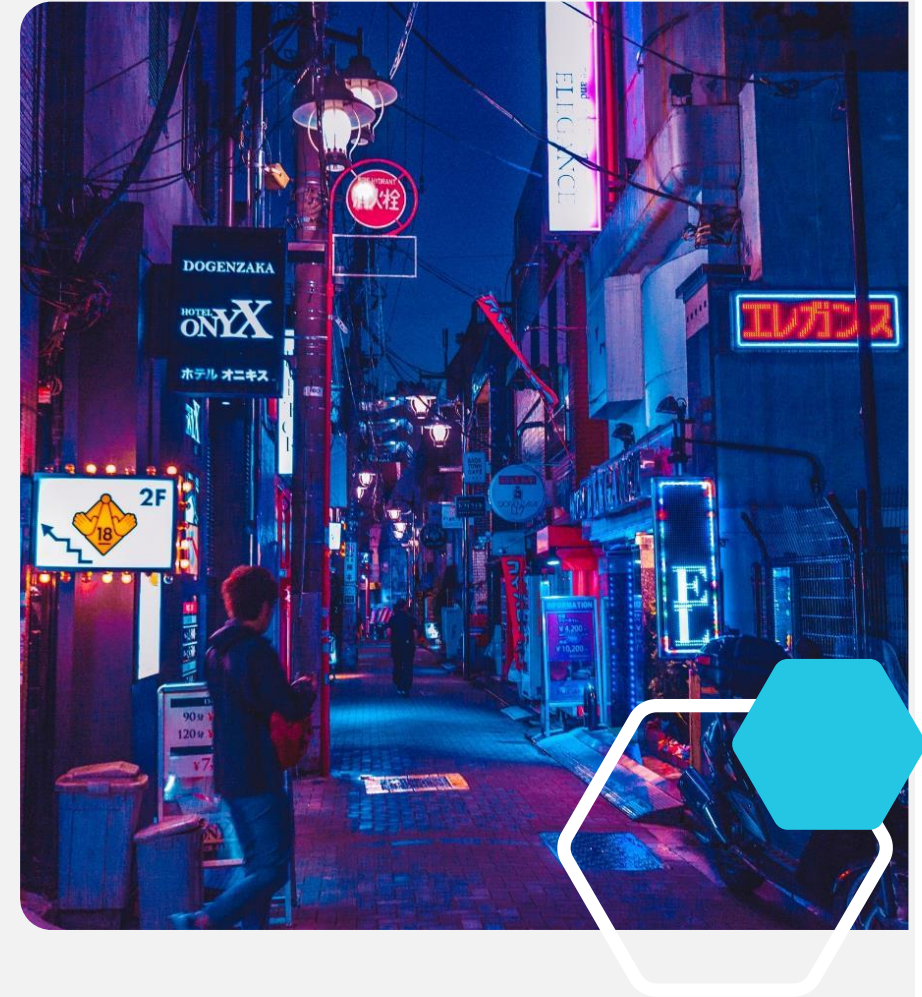


# Aspectos Sobresalientes

**Tamaño y cantidad de la información:** el crecimiento exponencial de archivos y de la información provoca que las consultas sean más veloces lo que genera cambios en la arquitectura lo que ha hecho más llamativo el uso de tecnologías NoSQL.

**Velocidad:** con la insaciable necesidad de los usuarios de obtener resultados inmediatos de sus consultas, el software en la nube, el uso de streaming ha desafiado a las bases de datos tradicionales y ha aparecido un cambio de modelo con el uso de las bases de datos NoSQL.

**Falta de innovación:** Las bases de datos tradicionales fueron creadas y adaptadas a una época pasada, el auge de los grandes volúmenes de información, la necesidad imperante de escalamiento, desligar el hardware del modelo de datos es lo que induce a la innovación trayendo como resultado soluciones más actuales y deseables como es la tecnología NoSQL.



# Características del Modelo NoSQL

Rick Cattell (2011), propone seis características comunes que tienen los sistemas NoSQL:

1. **Escalabilidad Horizontal** es una característica potente que tiene la capacidad de adaptarse a la demanda de rendimiento para el manejo de altos volúmenes de información, lo que implica tener diversos servidores conocidos como nodos, trabajando como un todo.
2. **Capacidad de replicar y distribuir datos en los servidores**, permitiendo a usuarios de todo el mundo acceder a los mismos datos garantizando el rendimiento, disponibilidad y seguridad de los datos.
3. **Interfaz o protocolo de nivel de llamada simple**, permitiendo simplicidad y eficiencia.
4. **Soporte para modelos de consistencia más débil**, es decir, el control transaccional de ACID (Atomicity, Consistency, Isolation, Durability) deja de ser tan importante lo que da como resultado pocas garantías sobre la consistencia de las transacciones. En su lugar implementa BASE (ver más adelante).
5. **Hacen uso eficiente de los índices distribuidos y de la RAM**, por lo que las organizaciones pueden hacer una transición a nuevas herramientas tecnológica de manera eficiente sin afectar el almacenamiento de datos.
6. **No se basan en un diseño rígido** ya que estos tienen la facilidad de establecer dinámicamente nuevos atributos o esquemas de registros de datos.



# Teorema CAP

Este término se escuchó por primera vez en el año 2000 por el profesor Eric Brewer quien sostenía que no se puede garantizar de forma simultánea la consistencia, la disponibilidad y la tolerancia al particionado de red en un sistema computacional distribuido.

El teorema de CAP es un concepto de que un sistema de almacenamiento distribuido solo puede tener 2 de los 3: consistencia, disponibilidad y tolerancia al particionado de red.





# Teorema CAP

**CONSISTENCIA.** El cliente percibe que un conjunto de operaciones ha ocurrido todas a la vez, es decir, los datos se mueven de un estado correcto a otro estado correcto, sin posibilidad de que los lectores puedan ver valores diferentes.

**DISPONIBILIDAD.** El sistema permanece operativo el 100% del tiempo y cada operación debe terminar en una respuesta prevista. La palabra clave aquí es cada. Cada nodo (a cada lado de una partición de red) debe poder responder en un período de tiempo razonable.

**TOLERANCIA AL PARTICIONADO DE RED.** El sistema aún necesita funcionar incluso cuando algunos nodos del sistema no pueden comunicarse con otros nodos del sistema.



# Teorema CAP

CA (Consistency – Availability): el sistema siempre estará disponible y responderá a las peticiones, la información o los datos procesados serán consistentes. Pero se permite una pérdida de comunicación entre nodos (particionado de red).

AP (Availability – Partition Tolerance): el sistema siempre estará disponible y responderá a las peticiones, aunque existan problemas de comunicación entre nodos (particionado de red), los datos procesados pueden ser inconsistentes.

CP (Consistency – Partition Tolerance): El sistema siempre garantizará la consistencia, aunque se pierda la comunicación entre nodos (particionado de red) y no se asegura que el sistema responda a la disponibilidad.



# BASE (Basically Available, Soft State, Eventually Consistent).



BASE (disponibilidad básica, estado suave, consistencia eventual) es una filosofía de diseño de sistema de datos que favorece la disponibilidad sobre la consistencia de las operaciones, y se desarrolló como alternativa para desarrollar arquitecturas de datos con mayor escalabilidad y accesibilidad permitiendo de esta manera más oportunidades para las empresas en expansión o que desean adquirir hardware para sus operaciones de datos. (Cattell, 2011)

- **Disponibilidad básica:** indica que el sistema hace garantía de disponibilidad, en términos del teorema de CAP, el sistema parecerá funcionar la mayor parte del tiempo y habrá una respuesta a cualquier solicitud. La respuesta aún podría ser "falla" en la obtención de los datos solicitados, o los datos pueden estar en un estado inconsistente o cambiante.
- **Estado suave:** indica que el estado del sistema puede cambiar con el tiempo, incluso sin entrada. Esto se debe al eventual modelo de coherencia.
- **Consistencia eventual:** indica que el sistema se volverá consistente con el tiempo, dado que el sistema no recibe información durante ese tiempo.



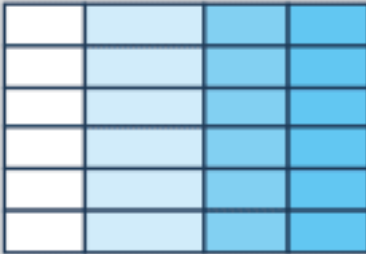


# Categorías de BD NoSQL

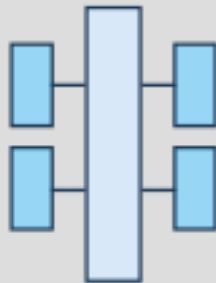
# Distribución por tipo de Base de Datos SQL vs NoSQL

## SQL

### Relational

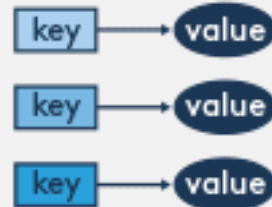


### Analytical (OLAP)



## NoSQL

### Key-Value



### Column-Family



### Graph



### Document



# Tipo Clave (Llave) Valor

Son útiles para operaciones sencillas, que se basan solo en los atributos llave.

Su objetivo es facilitar el acceso a datos, por lo general, estos sistemas proporcionan un mecanismo de persistencia y funcionalidad adicional como: replicación, control de versiones, bloqueo, transacciones, clasificación. Además, la interfaz del cliente proporciona inserciones, eliminaciones y búsquedas de índices. (Cattell, 2011)

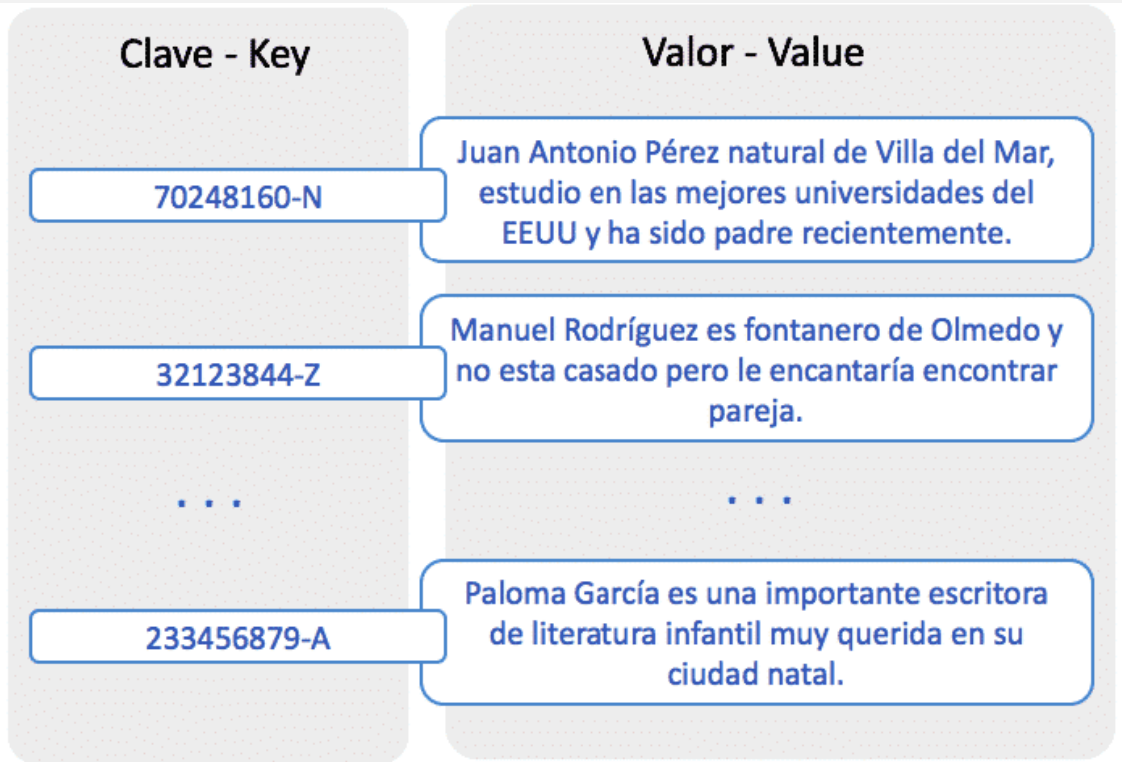
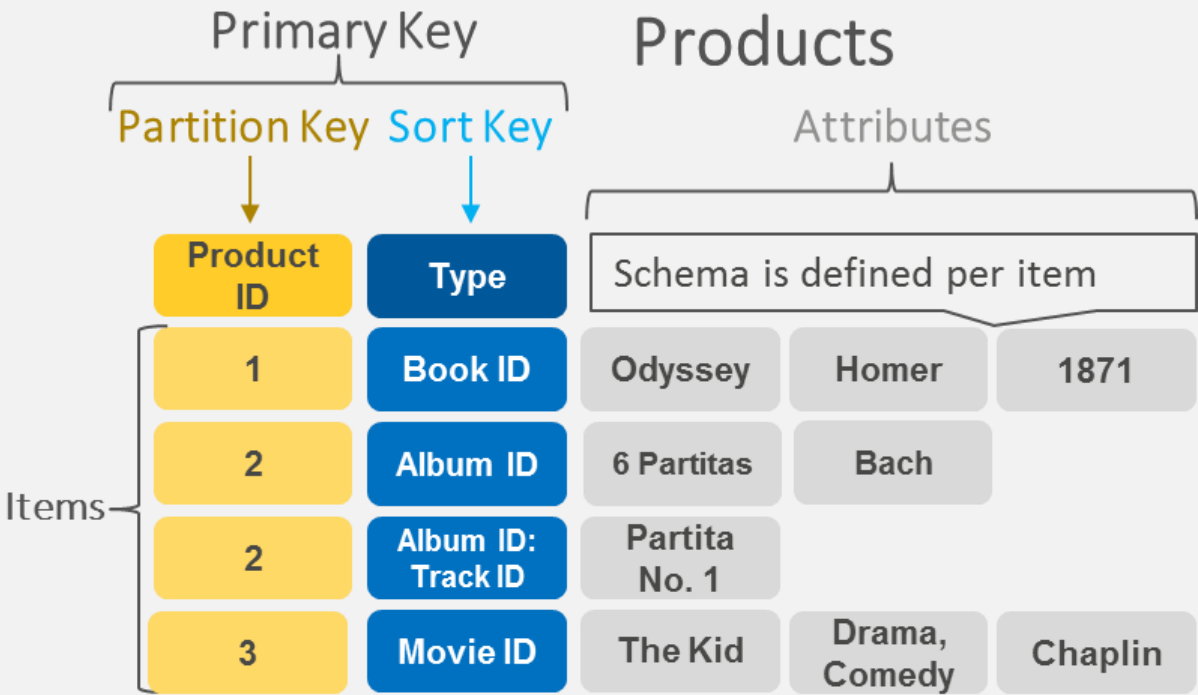
Estas bases de datos sacrifican la consistencia de los datos para obtener escalabilidad y como punto débil, la falta de un esquema hace mucho más difícil interpretar los datos. (Ameya Nayak, 2013)

Algunos ejemplos son: Riak, Voldemort, Redis, Amazon DynamoDB, entre otros.





# Tipo Clave (Llave) Valor



# Basadas en Documentos

Las bases de datos basadas en documentos se refieren a aquellas que almacenan sus datos en forma de documentos y dentro de estos documentos, posibilitando anidar otros documentos relacionados, como si se tratase de carpetas físicas. (Pollo Cattaneo, López Nocera, & Daián Rottoli, 2014)

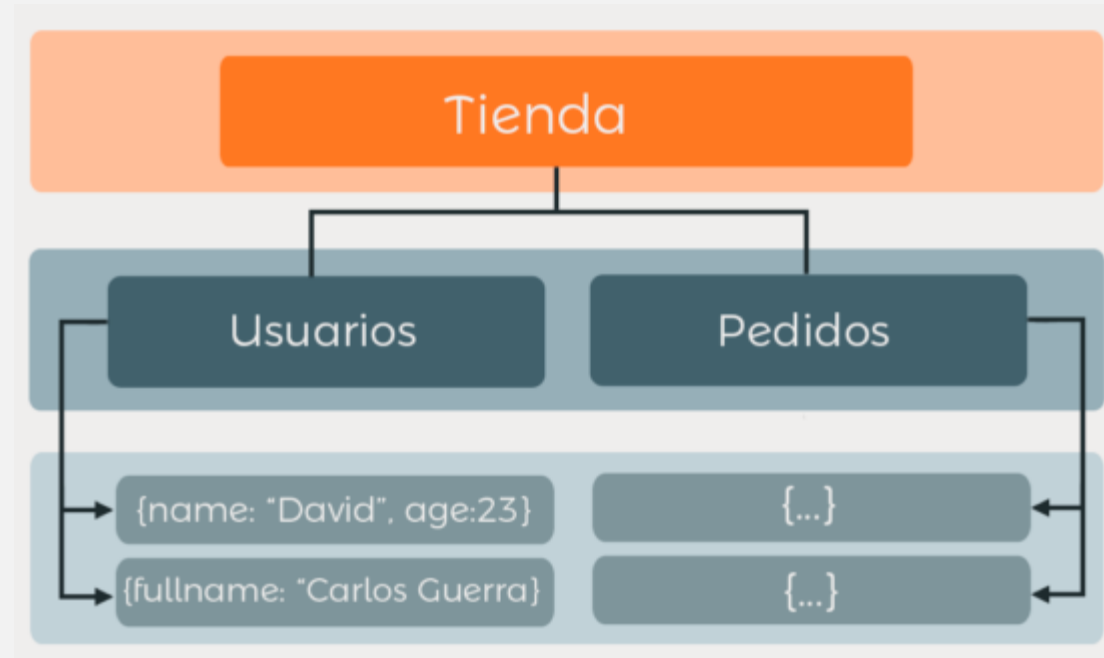
Las bases de datos basadas en documentos ofrecen un gran rendimiento y opciones de escalabilidad horizontal y son similares a los registros en bases de datos relacionales, pero son mucho más flexibles, ya que no tienen esquema. (Ameya Nayak, 2013)

A diferencia de las bases de datos tipo llave-valor, estos sistemas usualmente admiten índices secundarios y múltiples tipos de documentos (objetos) por base de datos, y documentos o listas anidados. (Cattell, 2011)

Algunos de los ejemplos más destacados de este tipo de base de datos son: MongoDB, CouchDB, SimpleDB, entre otros.

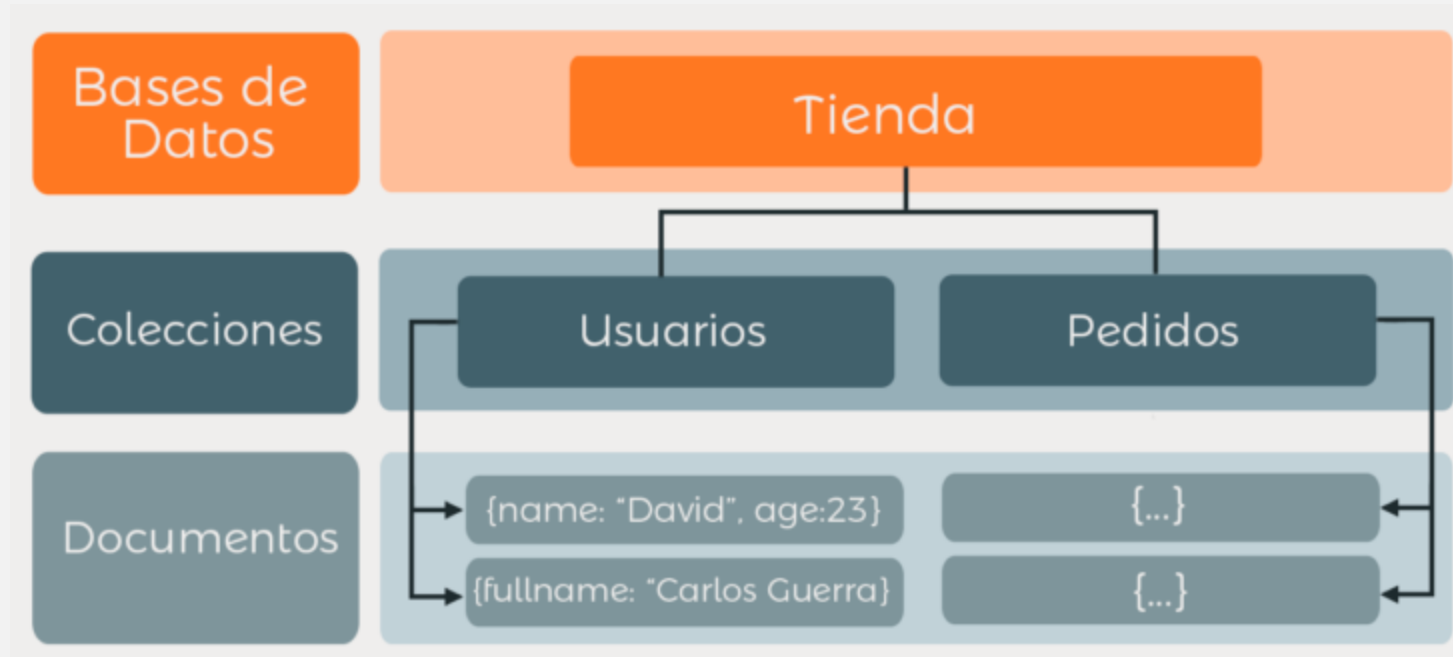


# Modelo Tradicional

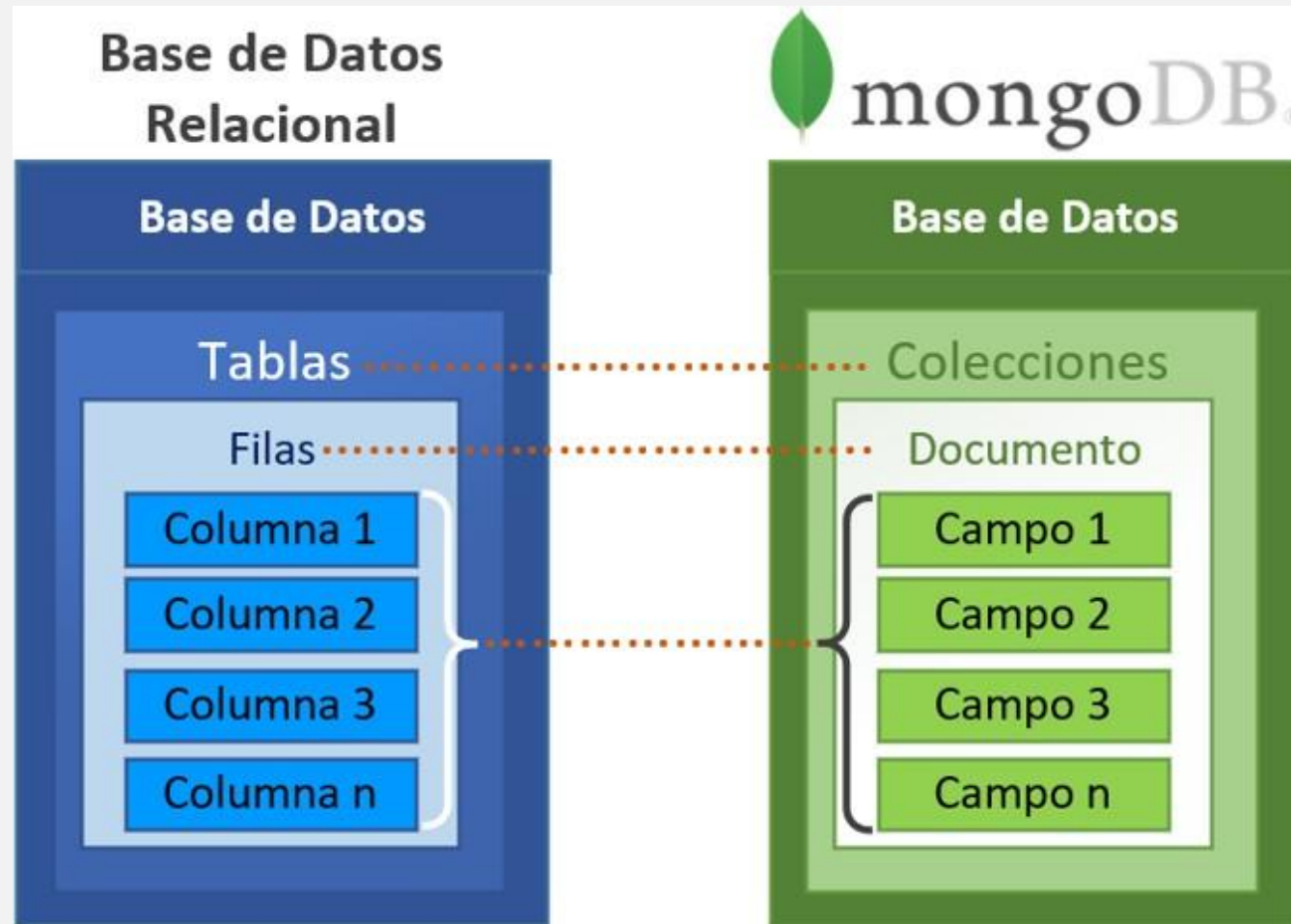




# Basada en Documentos



# Basada en Documentos



# Basada en Documentos

## Relational

ID	first_name	last_name	cell	city	year_of_birth	location_x	location_y
1	'Mary'	'Jones'	'516-555-2048'	'Long Island'	1986	'-73.9876'	'40.7574'

ID	user_id	profession
10	1	'Developer'
11	1	'Engineer'

ID	user_id	name	version
20	1	'MyApp'	1.0.4
21	1	'DocFinder'	2.5.7

ID	user_id	make	year
30	1	'Bentley'	1973
31	1	'Rolls Royce'	1965

## MongoDB

```
{  
  first_name: "Mary",  
  last_name: "Jones",  
  cell: "516-555-2048",  
  city: "Long Island",  
  year_of_birth: 1986,  
  location: {  
    type: "Point",  
    coordinates: [-73.9876, 40.7574]  
  },  
  profession: ["Developer", "Engineer"],  
  apps: [  
    { name: "MyApp",  
      version: 1.0.4 },  
    { name: "DocFinder",  
      version: 2.5.7 }  
  ],  
  cars: [  
    { make: "Bentley",  
      year: 1973 },  
    { make: "Rolls Royce",  
      year: 1965 }  
  ]  
}
```





# Orientadas a Columnas

Las bases de datos orientada a columnas usan el concepto de espacio de llaves, que es como un esquema en los modelos relacionales.

Este espacio de llaves, contienen todas las familias de columnas, que luego contienen filas, que luego contienen columnas, pero no almacenan los datos en tablas, es decir, son en realidad un almacén híbrido de filas/columnas que almacenan los datos en arquitecturas distribuidas masivamente, proporcionando mayor flexibilidad. (Ameya Nayak, 2013)

Las bases de datos orientadas a columnas son recomendadas para grandes almacenes de datos cuya lectura es más frecuente que su escritura, además, ofrecen alta escalabilidad, son apropiadas para aplicaciones analíticas o estadísticas, y para la minería de datos. (Castro Romero, González Sanabria, & Callejas Cuervo, 2012)

Algunos ejemplos de bases de datos orientadas a columnas son Cassandra y Big Table.



# Orientadas a Columnas

Base de datos relacional

NombreProveedor	ContactoProveedor	Telefono	Producto	Precio
Frutas Gutierrez	Antonio Gutierrez	607454545	Pera	2,05 €
Frutas Gutierrez	Antonio Gutierrez	607454545	Platano	1,87 €
Frutas Gutierrez	Antonio Gutierrez	607454545	Manzana	2,14 €
Frutas Gutierrez	Antonio Gutierrez	607454445	Naranja	1,36 €
Frutas Gutierrez	Antonio Gutierrez	607454545	Limon	1,04 €
Hortalizas del Sur	Guillermo Morales	652854874	Pimiento	0,50 €
Hortalizas del Sur	Guillermo Morales	652854874	Calabaza	1,30 €
Hortalizas del Sur	Guillermo Morales	652884874	Naranja	1,50 €
Hortalizas del Sur	Guillermo Morales	652854874	Pera	3,50 €
Hortalizas del Sur	Guillermo Morales	652854874	Aguacate	5,00 €
Azucarera Sevillana	Rodrigo Mendez	622525885	Azucar(kg)	3,00 €
Azucarera Sevillana	Rodrigo Mendez	622525885	Limon	1,00 €
Azucarera Sevillana	Rodrigo Mendez	622525885	Naranja	1,70 €
Huevos La Puebla	Araceli Arnedo	652879137	Huevo	0,10 €
Huevos La Puebla	Araceli Arnedo	652879137	Pera	1,98 €
Huevos La Puebla	Araceli Arnedo	652879137	Naranja	1,30 €
Huevos La Puebla	Araceli Arnedo	652879137	Arroz (Kg)	1,20 €
Arroces La Cigala	Maria Alvarez	677889922	Arroz (Kg)	1,45 €
Arroces La Cigala	Maria Alvarez	677889922	Naranja	2,45 €
Arroces La Cigala	Maria Alvarez	677889924	Limon	1,30 €



Base de datos orientada a columna

Datos de Proveedor			Datos de Producto	
Nombre	Contacto	Telefono	Nombre	Precio
Frutas Gutierrez	Antonio Gutierrez	607454545	Pera	2,05 €
Frutas Gutierrez	Antonio Gutierrez	607454545	Platano	1,87 €
Frutas Gutierrez	Antonio Gutierrez	607454545	Manzana	2,14 €
Frutas Gutierrez	Antonio Gutierrez	607454445	Naranja	1,36 €
Frutas Gutierrez	Antonio Gutierrez	607454545	Limon	1,04 €
Hortalizas del Sur	Guillermo Morales	652854874	Pimiento	0,50 €
Hortalizas del Sur	Guillermo Morales	652854874	Calabaza	1,30 €
Hortalizas del Sur	Guillermo Morales	652884874	Naranja	1,50 €
Hortalizas del Sur	Guillermo Morales	652854874	Pera	3,50 €
Hortalizas del Sur	Guillermo Morales	652854874	Aguacate	5,00 €
Azucarera Sevillana	Rodrigo Mendez	622525885	Azucar(kg)	3,00 €
Azucarera Sevillana	Rodrigo Mendez	622525885	Limon	1,00 €
Azucarera Sevillana	Rodrigo Mendez	622525885	Naranja	1,70 €
Huevos La Puebla	Araceli Arnedo	652879137	Huevo	0,10 €
Huevos La Puebla	Araceli Arnedo	652879137	Pera	1,98 €
Huevos La Puebla	Araceli Arnedo	652879137	Naranja	1,30 €
Huevos La Puebla	Araceli Arnedo	652879137	Arroz (Kg)	1,20 €
Arroces La Cigala	Maria Alvarez	677889922	Arroz (Kg)	1,45 €
Arroces La Cigala	Maria Alvarez	677889922	Naranja	2,45 €
Arroces La Cigala	Maria Alvarez	677889924	Limon	1,30 €

# Orientadas a Grafos

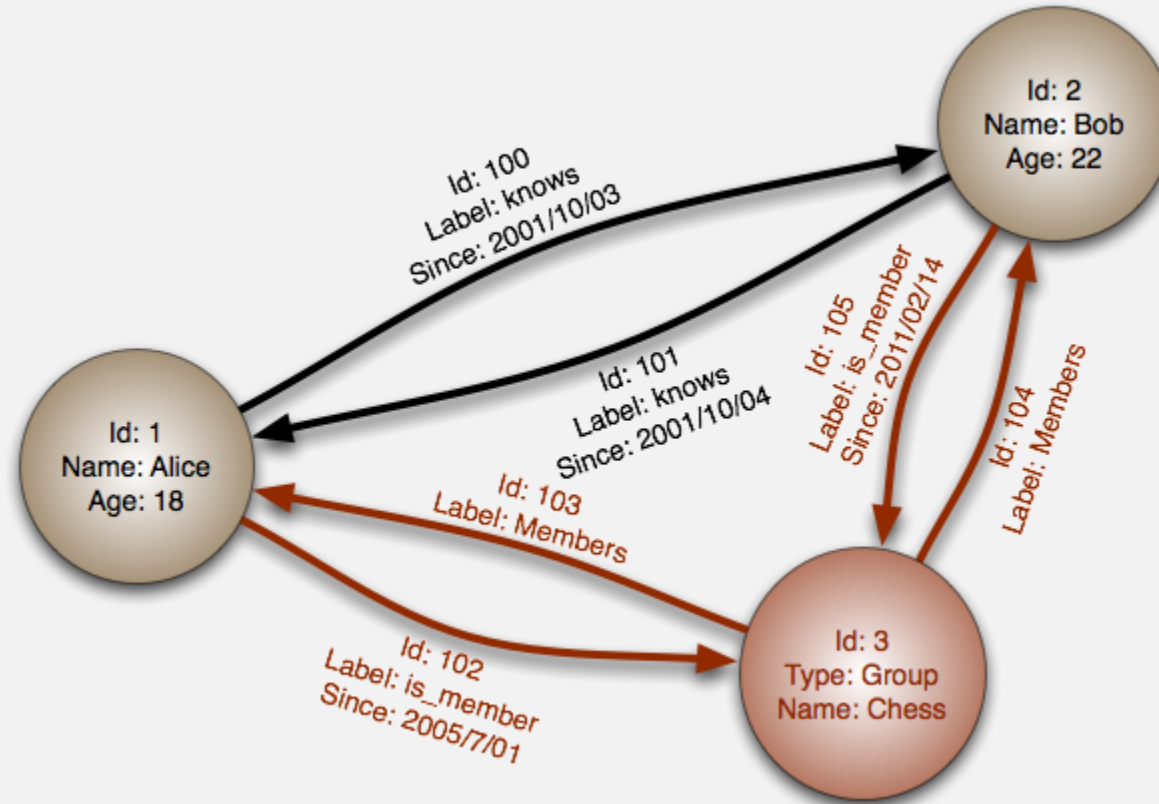
Las bases de datos orientadas a grafos analizan las relaciones que existen entre los registros de una base de datos para representar la información en un grafo, donde los nodos son los datos como tal y las aristas que unen a los nodos son las relaciones que hay entre ellos, dicho grafo se utiliza para recorrer la base de datos de manera eficiente. (Sepúlveda Cruz, 2017)

Son herramientas poderosas para analizar rápidamente la información que el usuario desea consumir fácilmente. Por ejemplo, en [LinkedIn](#) las conexiones de primer, segundo o tercer grado, se obtiene de los resultados del gráfico de red profesional del sitio de la red social construido en una base de datos de grafos. [También, Facebook, Instagram y Twitter utilizan estas bases de datos orientadas a grafos.](#)





# Orientadas a Grafos



# En Resumen...

## SQL

Cuando el volumen de mis datos no crece o lo hace poco a poco.

Cuando las necesidades de proceso se pueden asumir en un sólo servidor.

Cuando no tenemos picos de uso del sistema por parte de los usuarios más allá de los previstos.

## NoSQL

Cuando el volumen de mis datos crece muy rápidamente en momentos puntuales.

Cuando las necesidades de proceso no se pueden prever.

Cuando tenemos picos de uso del sistema por parte de los usuarios en múltiples ocasiones.



# Cuándo usarlas...







# ¿Preguntas?

👤 Ing. Víctor A. Fuentes T.

📞 +507 5603673

✉ victor.fuentes@utp.ac.pa

🌐 [www.fisc.utp.ac.pa](http://www.fisc.utp.ac.pa)