Unidad 4 - Bases de datos noSQL

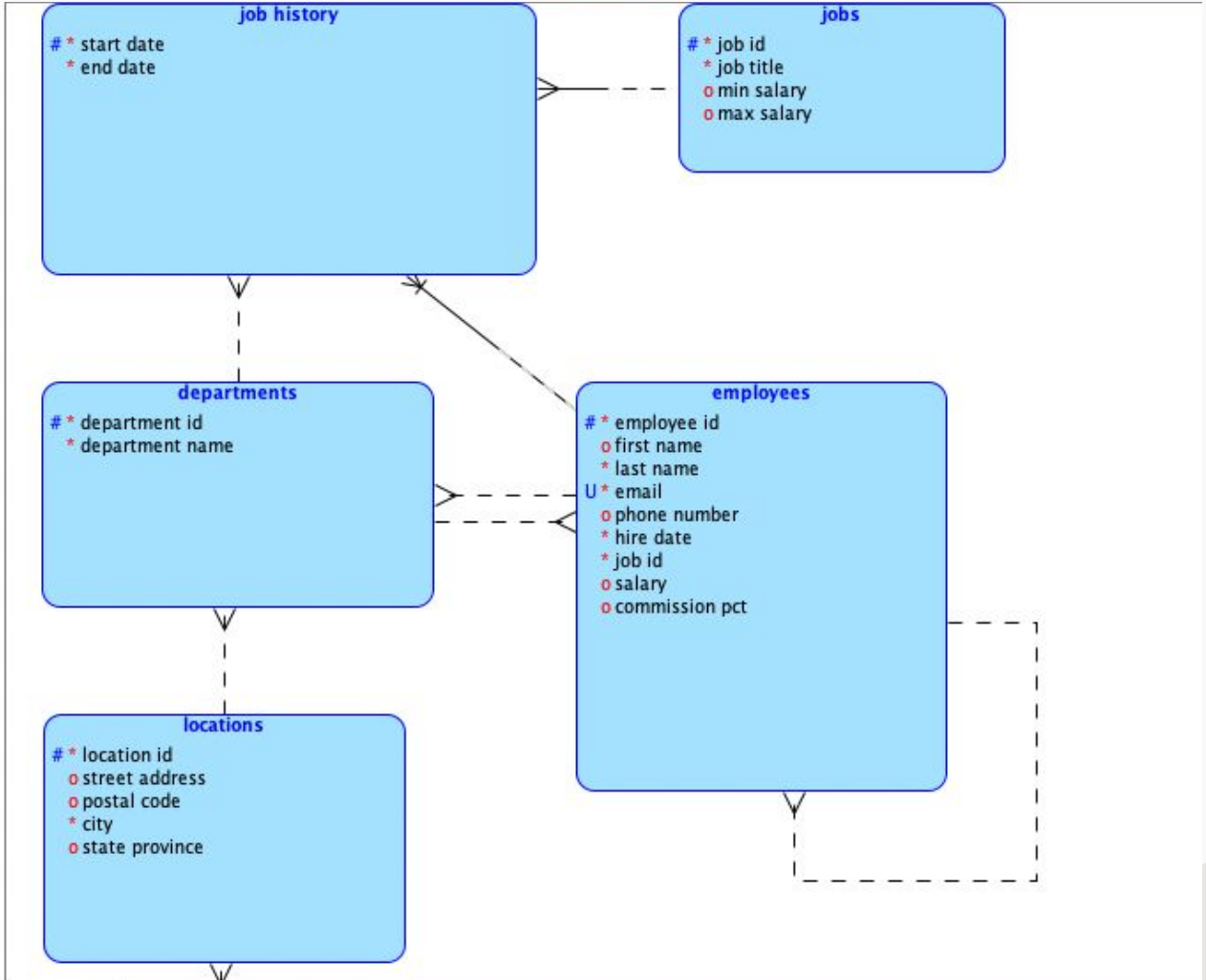
- Describir el concepto de bases de datos NoSQL, sus principios y comprender las diferencias fundamentales con respecto a las bases de datos relacionales.
- Explicar las características y propiedades clave de las bases de datos NoSQL, como la escalabilidad horizontal, la flexibilidad del esquema, la alta disponibilidad y la tolerancia a fallos, en los principales tipos de bases de datos NoSQL, como las bases de datos clave-valor, orientadas a documentos, orientadas a columnas y orientadas a grafos.
- Describir la arquitectura general de las bases de datos NoSQL, identificando los componentes principales y su interacción, así como sus características de replicación y distribución de datos.

Introducción

Desde los años 80's la mayoría de los sistemas organizacionales han utilizado gestores relacionales:

- Aplicaciones de bancos, bibliotecas, … Sistemas operacionales que manejan datos transaccionales.
- CRM (Customer Relationship Management)
- ERP (Enterprise Resource Planning)
- Sistemas académicos
- Etc.

MODELO DE DATOS RELACIONAL



Employees

```
1  select * from employees;
```

EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DATE	JOB_ID	SALARY	COMMISSION_PCT	MANAGER_ID	DEPARTMENT_ID
100	Steven	King	SKING	515.123.4567	17-JUN-03	AD_PRES	24000	-	-	90
101	Neena	Kochhar	NKOCHHAR	515.123.4568	21-SEP-05	AD_VP	17000	-	100	90
102	Lex	De Haan	LDEHAAN	515.123.4569	13-JAN-01	AD_VP	17000	-	100	90
103	Alexander	Hunold	AHUNOLD	590.423.4567	03-JAN-06	IT_PROG	9000	-	102	60
104	Bruce	Ernst	BERNST	590.423.4568	21-MAY-07	IT_PROG	6000	-	103	60
105	David	Austin	DAUSTIN	590.423.4569	25-JUN-05	IT_PROG	4800	-	103	60

```
1  select * from Departments;
```

DEPARTMENT_ID	DEPARTMENT_NAME	MANAGER_ID	LOCATION_ID
10	Administration	200	1700
20	Marketing	201	1800
30	Purchasing	114	1700
40	Human Resources	203	2400
50	Shipping	121	1500

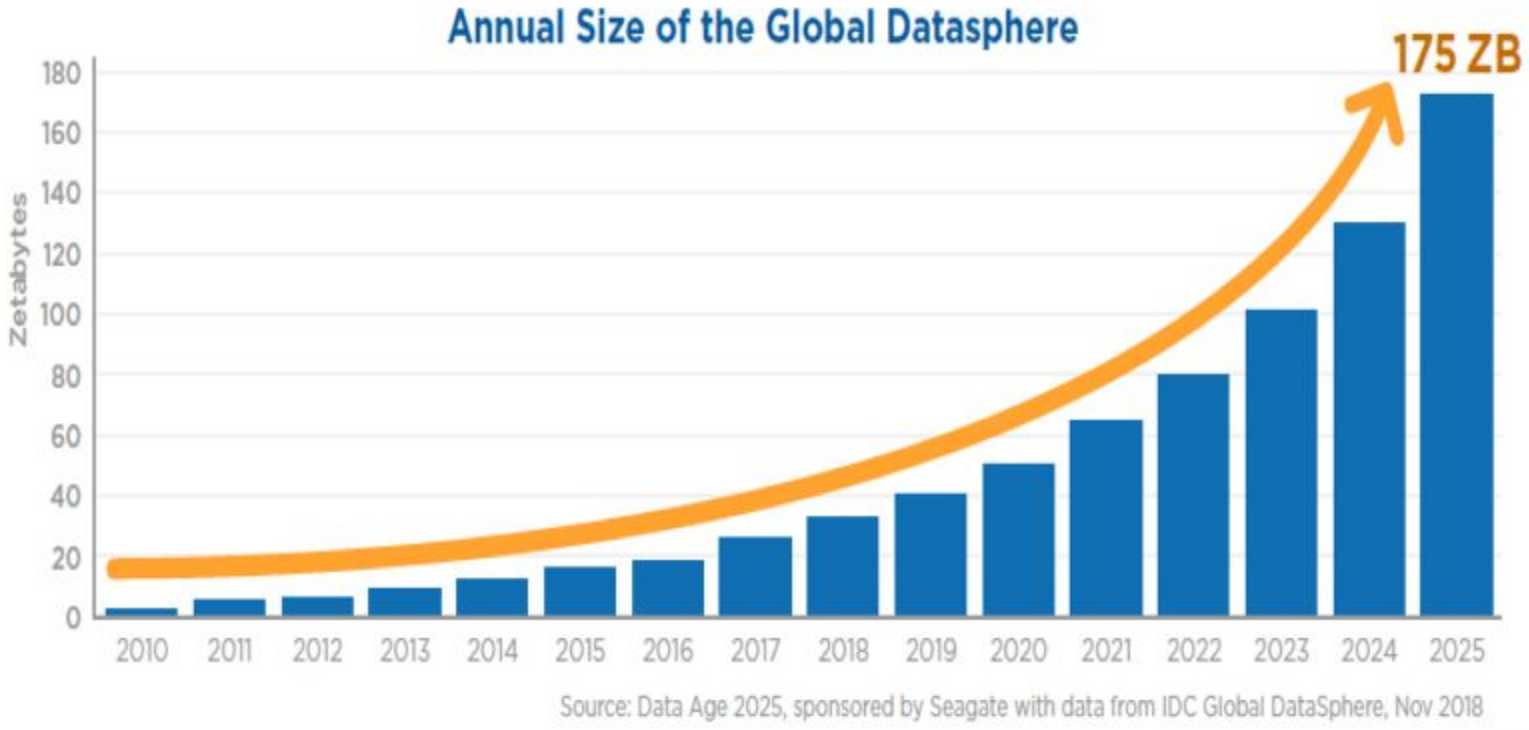
ttp://www.geeksengine.com/database/sample/northwind-categories.ph

Tomado de: Oracle Live HR Objects and Data For Live SQL [link]

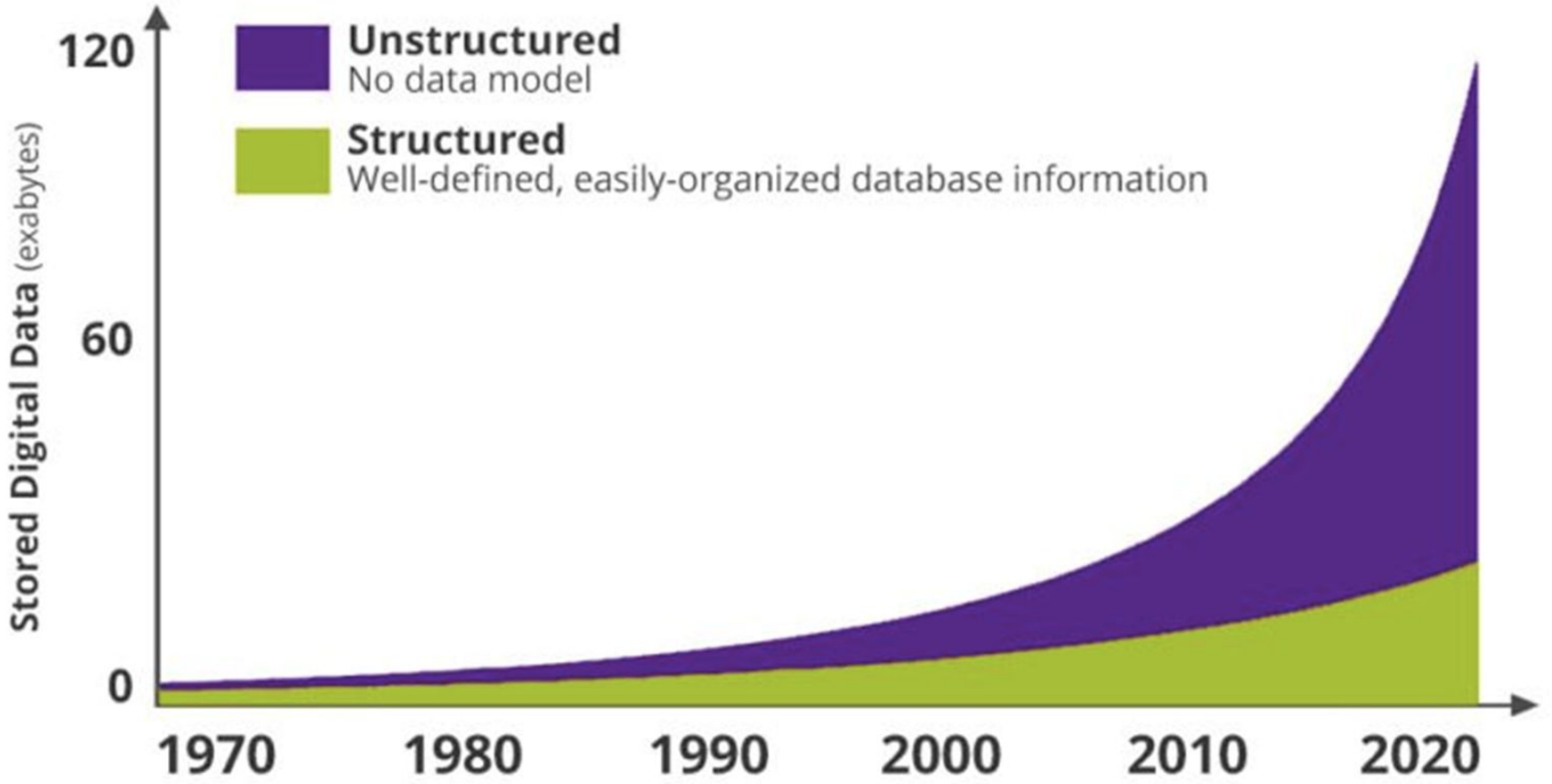
INTRODUCCIÓN A NOSQL

Las BD relacionales (R-DBMS) se caracterizan por:

- Utilizar un modelo de datos simple (basado en tablas y relaciones entre tablas).
- Ofrecer herramientas para garantizar la integridad de datos y la consistencia de la información (ACID).
- Utilizar un lenguaje de consulta estándar, simple y potente.
- Proporcionar utilidades para asegurar el acceso, manipulación y la privacidad de los datos.
- Ofrecer utilidades para la auditoría y recuperación de datos.
- Garantizar la independencia del esquema lógico y físico.



IDC Datasphere growth - **IDC REPORT FIGURE**



Tomado de: https://www.komprise.com/glossary_terms/unstructured-data/

¿Qué es NoSQL?

Es una categoría general de sistemas de gestión de bases de datos que difiere de los RDBMS en diferente modos:

No tienen esquemas, no permiten JOINs, no intentan garantizar ACID y escalan horizontalmente

Tanto las bases de datos NoSQL como las relacionales son tipos de Almacenamiento Estructurado.

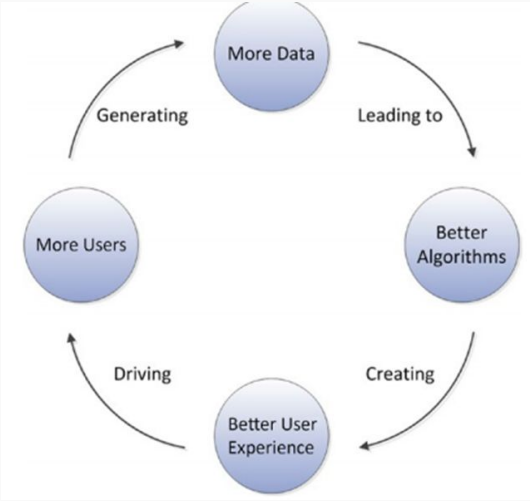
- El término surge en 1998 por Carlo Strozzi. NoSQL - A relational database management system. Carlo Strozzi
- Resucitado en 2009 por Eric Evans

Por qué surgen?

- Necesidad de manipulación de información no estructurada y semi-estructurada.
- Manejo de grandes volúmenes de datos.
- Tradicionalmente los sistemas relacionales asumen modelo computacional centralizado.

Big Data

- Internet de las cosas (IoT)
  - Carros
  - Electrodomésticos
  - Teléfono
  - Sensores de velocidad
  - Sensores de temperatura
- Logs de aplicaciones (clicks de páginas web, acciones del usuario, envío de mensajes,…)



Ciclo Virtuoso de Big Data (Harrison, 2015)

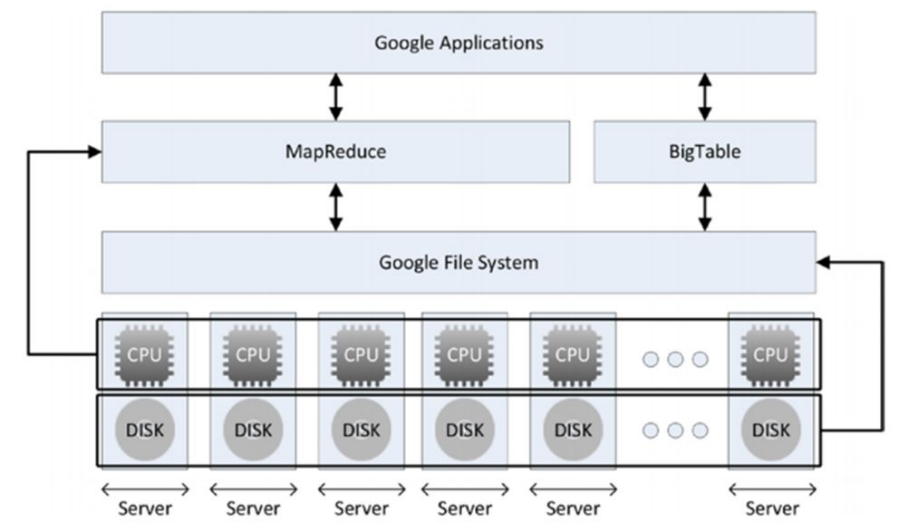
Big Data

Gartner (2012): «Son activos de información caracterizados por su alto volumen, velocidad y variedad, que demandan soluciones innovadoras y eficientes de procesado para la mejora del conocimiento y toma de decisiones en las organizaciones».

Actividad

Actividad de aprendizaje:
Investigar los siguientes y explicar la imagen de la derecha:

- Google File System (GFS)
- Map Reduce
- Big Table



¿QUÉ TANTA INFORMACIÓN ES BIG DATA?

- Tengo que procesar 1’000.000 de archivos de música de 4M c/u, es big data?
- Diariamente mi aplicación genera 1 Gb de datos, a patir de cuándo se considera Big Data?
- Tengo una app que cada hora genera 1 Gb de datos, se considera Big Data?

No hay unicidad, pero se puede hablar de Big Data como conjuntos de datos que van desde 30-50 Terabytes a varios Petabytes.

- 1 kibibyte = 1024 B = 2^{10} bytes.
- 1 kilobyte = 1000 B = 10^3 bytes.

1 KiloByte (KB) = 1,000 Bytes
1 MegaByte (MB) = 1,000 KB
1 GigaByte (GB) = 1,000 MB
1 TeraByte (TB) = 1,000 GB
1 PetaByte (PB) = 1,000 TB
1 ExaByte (EB) = 1,000 PB
1 ZettaByte (ZB) = 1,00 EB
1 YottaByte (YB) = 1,000 ZB
1 XeraByte (XB) = 1,000 YB

Características de Big Data

- 7V'S DE

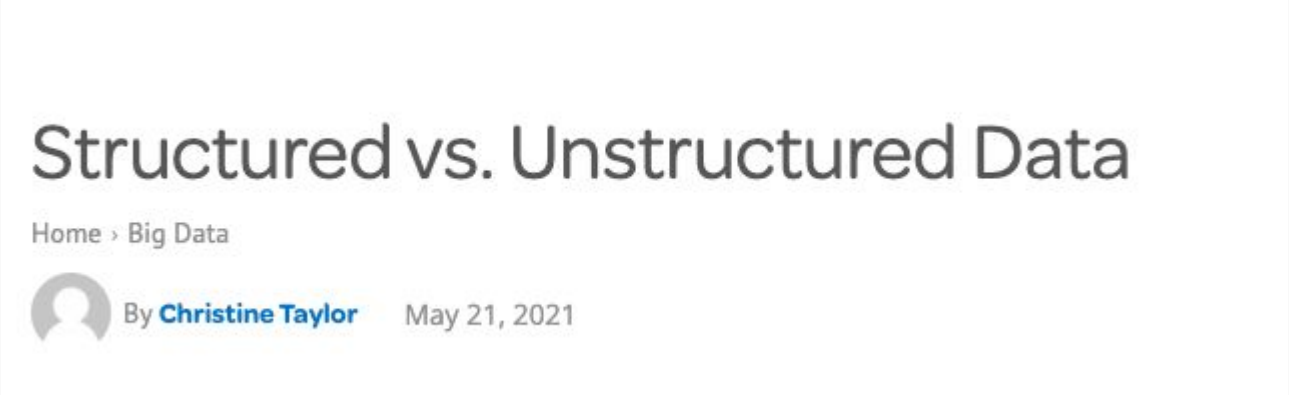
  BIG DATA

- Volumen de información
- Velocidad de datos
- Variedad de los datos
- Veracidad de los datos
- Viabilidad
- Visualización de los datos
- Valor de los datos

https://www.iic.uam.es/innovacion/big-data-infografia-7-v/

ACTIVIDAD

- Leer artículo Structured vs. Unstructured Data



TIPOS DE DATOS

- Datos estructurados. Aquellos que se presentan en un formato o esquema bien definido y que poseen campos fijos. Son hojas de cálculo, archivos, bases de datos tradicionales provenientes de CRM, ERP, etc.
- Datos semiestructurados. No tienen formato definido, pero sí contienen etiquetas u otros marcadores con el fin de clasificar los elementos de los mismos. En esta categoría encontramos textos con etiquetas XML y HTML.
- Datos no estructurados. Los más numerosos. Son datos de tipo indefinido, almacenados principalmente como documentos u objetos sin estructura fija ni bajo ningún patrón concreto. Pueden ser generados por máquinas y personas. Son archivos de audio, vídeo, fotografía y formatos de texto libre como emails, SMS, artículos, WhatsApp, etc.

El 80 % de la información relevante para un negocio se origina en forma no estructurada, principalmente en formato texto.

Joyanes (2013)

EJEMPLOS DE FUENTES DE DATOS

Fuentes	Descripción	Tipo de dato
Web and social	Engloba todo contenido proveniente de las páginas web e información obtenida de redes sociales.	•Publicaciones en Facebook. •Feeds de Twitter. •Contenido de páginas web. •Publicaciones en blogs.
Machine-to-Machine (M2)	Machine (M2M) Hace referencia a la comunicación entre máquinas. Son aquellas tecnologías que habilitan la conectividad entre dispositivos a través de sensores.	•Lecturas RFID. •Señales GPS. •Temperatura. •Variables meteorológicas.
Big transaction data	Datos de transacciones provenientes de centralitas de telefonía, atención al cliente, banca, etc.	•Registros detallados de llamadas (CDR). •Mensajería. •Registros de facturación.
Biometrics	Esta información tiene gran relevancia en sectores como la seguridad, gobiernos, servicios de inteligencia, etc.	•Reconocimiento facial. •ADN. •Huellas digitales. •Escaneo ocular.
Human generated	Datos generados por las personas en su día a día.	•Email. •Registros médicos. •Notas de voz. •Multas. Documentos electrónicos.

Fuente: https://www.marketing-xxi.com/big-data-aplicaciones-gestion-dato-distintas-etapas-funnel-conversion/big-data

METADATOS

- Los metadatos proporcionan información sobre cada conjunto de datos. Por ejemplo, el tamaño, el esquema de una base de datos, el formato, la última hora de modificación, las listas de control de acceso, el uso, etc.
- El uso de metadatos permite la gestión de una plataforma y arquitectura de data lake escalable, así como gobierno de datos.
- Los metadatos se suelen almacenar en un catálogo central para proporcionar a los usuarios información sobre los conjuntos de datos disponibles.

¿QUÉ OCURRE EN UN MINUTO EN INTERNET?

- Revise qué ocurre en un minuto en Internet, para que tengamos claridad de la gran variedad de datos y volumen que se puede utilizar como fuentes para los análisis.

https://influencermarketinghub.com/internet-real-time-social-media-statistics/

INTRODUCCIÓN A NOSQL

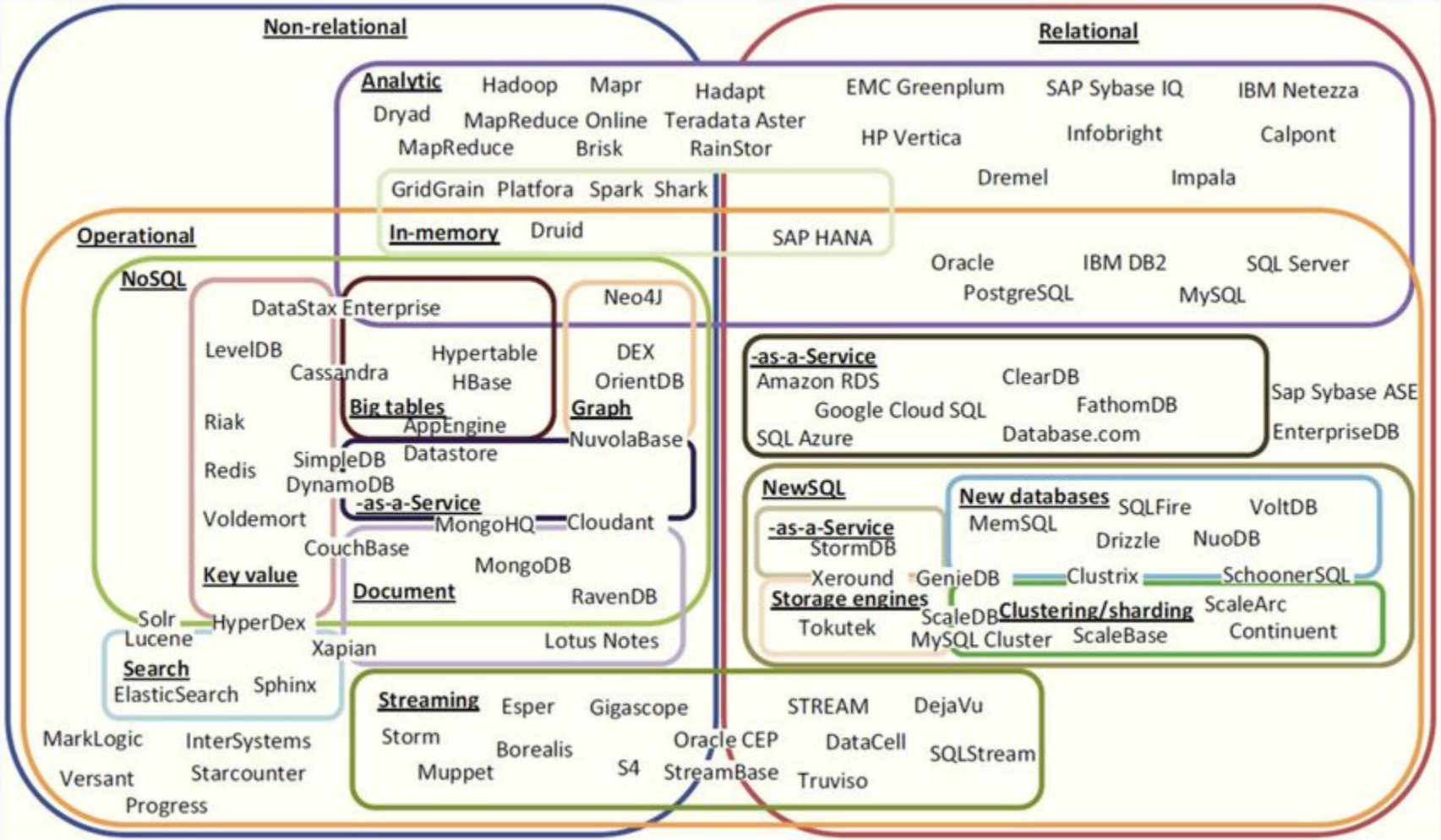
- Aplicaciones como Facebook, Amazon, Google necesitaban dar servicio a miles de usuarios concurrentes y responder millones de preguntas diarias y la tecnología relacional no ofrecía ni el nivel de escalabilidad ni el rendimiento adecuado.

Los RDBMS presentan varios cuellos de botella:

- Gestión de Logs.
- Control de concurrencia.
- Protocolos de transacciones distribuidas.
- Administración de buffers.
- Interfaces CLI (JDBC, ODBC, etc.).

SOLUCIÓN:

- Implementar modelos alternativos que se ajusten a lo que realmente se necesita.
- Google, Facebook, Amazon diseñan su propia solución.



Fuente: [Korflatis et. al (2016)]

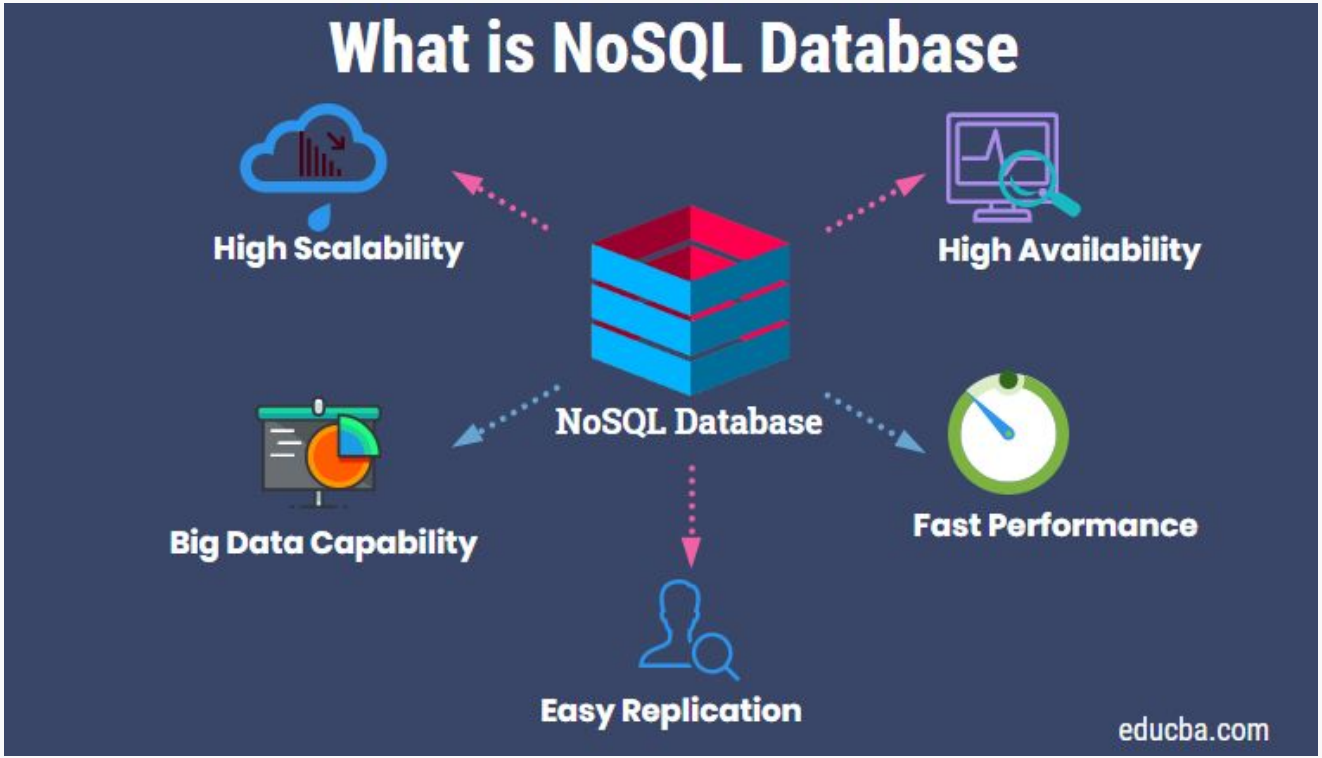
DB ENGINE RANKING

415 systems in ranking, October 2023

Rank Oct 2023	Rank Sep 2023	Rank Oct 2022	DBMS	Database Model	Score Oct 2023	Score Sep 2023	Score Oct 2022
1.	1.	1.	Oracle	Relational, Multi-model	1261.42	+20.54	+25.05
2.	2.	2.	MySQL	Relational, Multi-model	1133.32	+21.83	-72.06
3.	3.	3.	Microsoft SQL Server	Relational, Multi-model	896.88	-5.34	-27.80
4.	4.	4.	PostgreSQL	Relational, Multi-model	638.82	+18.06	+16.10
5.	5.	5.	MongoDB	Document, Multi-model	431.42	-8.00	-54.81
6.	6.	6.	Redis	Key-value, Multi-model	162.96	-0.72	-20.41
7.	7.	7.	Elasticsearch	Search engine, Multi-model	137.15	-1.84	-13.92
8.	8.	8.	IBM Db2	Relational, Multi-model	134.87	-1.85	-14.79
9.	9.	↑ 10.	SQLite	Relational	125.14	-4.06	-12.66
10.	10.	↓ 9.	Microsoft Access	Relational	124.31	-4.25	-13.85
11.	11.	↑ 13.	Snowflake	Relational	123.24	+2.35	+16.51
12.	12.	↓ 11.	Cassandra	Wide column, Multi-model	108.82	-1.24	-9.12
13.	13.	↓ 12.	MariaDB	Relational, Multi-model	99.66	-0.79	-9.65
14.	14.	14.	Splunk	Search engine	92.37	+0.98	-2.28
15.	15.	↑ 16.	Microsoft Azure SQL Database	Relational, Multi-model	80.93	-1.80	-4.03
16.	16.	↓ 15.	Amazon DynamoDB	Multi-model	80.91	+0.00	-7.44
17.	17.	↑ 20.	Databricks	Multi-model	75.82	+0.64	+18.21
18.	18.	↓ 17.	Hive	Relational	69.18	-2.65	-11.42
19.	19.	↓ 18.	Teradata	Relational, Multi-model	58.56	-1.77	-7.51
20.	20.	↑ 22.	Google BigQuery	Relational	56.57	+0.11	+4.12

https://db-engines.com/en/ranking

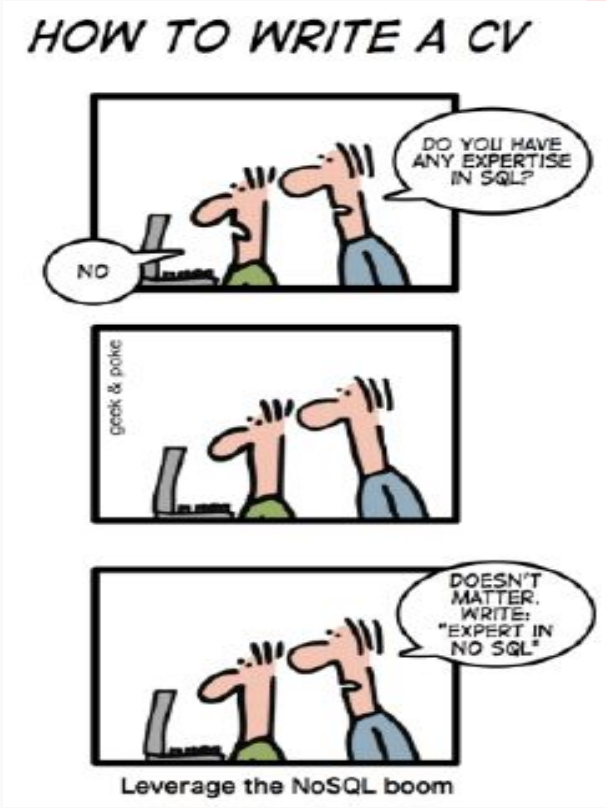
CARACTERÍSTICAS



Source: https://www.educba.com/what-is-nosql-database/

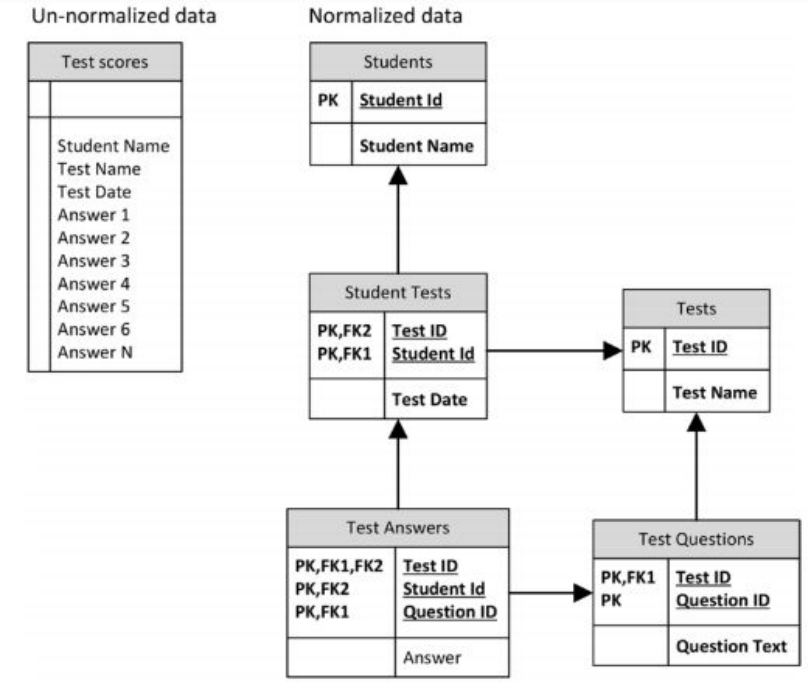
Definición

- ***NoSQL*** *is a set of concepts that allows the rapid and efficient processing of data sets with a focus on performance, reliability, and agility*. Dan McCreary - Ann Kelly



Principios

- Tanto las bases de datos NoSQL como las relacionales son tipos de Almacenamiento Estructurado.
- La principal diferencia radica en cómo guardan los datos.



PRINCIPIOS DE NOSQL

NoSQL se basa en 4 principios:

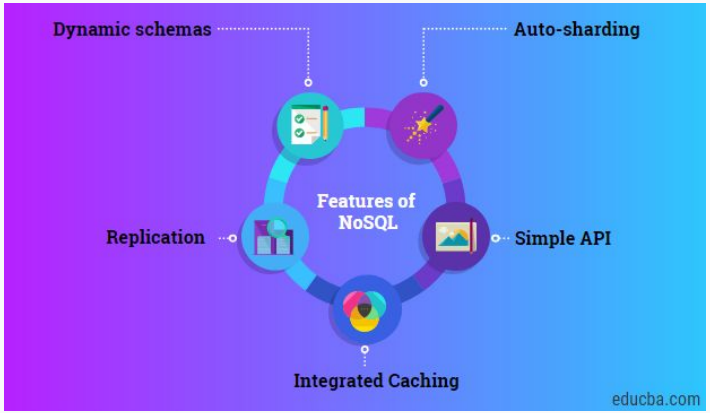
- El control transaccional ACID no es importante.
- Los JOINs tampoco lo son. En especial los complejos y distribuidos. Se persigue la desnormalización.
- Algunos elementos relacionales son necesarios y aconsejables: claves (keys).
- Gran capacidad de escalabilidad y de replicación en múltiples servidores.

CARACTERÍSTICAS PRINCIPALES

- Se ejecutan en máquinas con pocos recursos (clusters)
- No tiene esquemas fijos
- Fáciles de usar en clusters de balanceo de carga convencionales facilitan **escalabilidad horizontal**
- No genera cuellos de botella
- Tienen propiedades ACID en un nodo del clúster y son "**eventualmente consistentes**" en el clúster.
- Las consultas se realizan principalmente por key o índice (otro tipo de consultas son muy costosas).
- Las consultas complejas se realizan mediante una infraestructura de procesamiento externo tal como **MapReduce**.

DIFERENCIAS CON LAS SQL

- Modelo de datos relacional vs no relacionales
- SQL vs lenguajes dinámicos
- Almacenamiento en tablas vs estructuras flexibles.
- Info no estructurada (Json) vs transacciones con varias filas, JOINs.
- Arquitectura distribuida vs centralizada.
- Escalabilidad horizontal vs vertical



COMPUTACIÓN DISTRIBUIDA

Clustering: un clúster o racimo de computadoras consiste en un grupo de computadoras de relativo bajo costo conectadas, unidas entre sí, normalmente por una red de alta velocidad y que se comportan como si fuesen una única computadora (balanceo de carga).

- Alto rendimiento
- Alta disponibilidad
- Balanceo de carga
- Escalabilidad

Teorema CAP

- Teorema de Brewer: “es imposible para un sistema computacional distribuido ofrecer simultáneamente las siguientes tres garantías”:

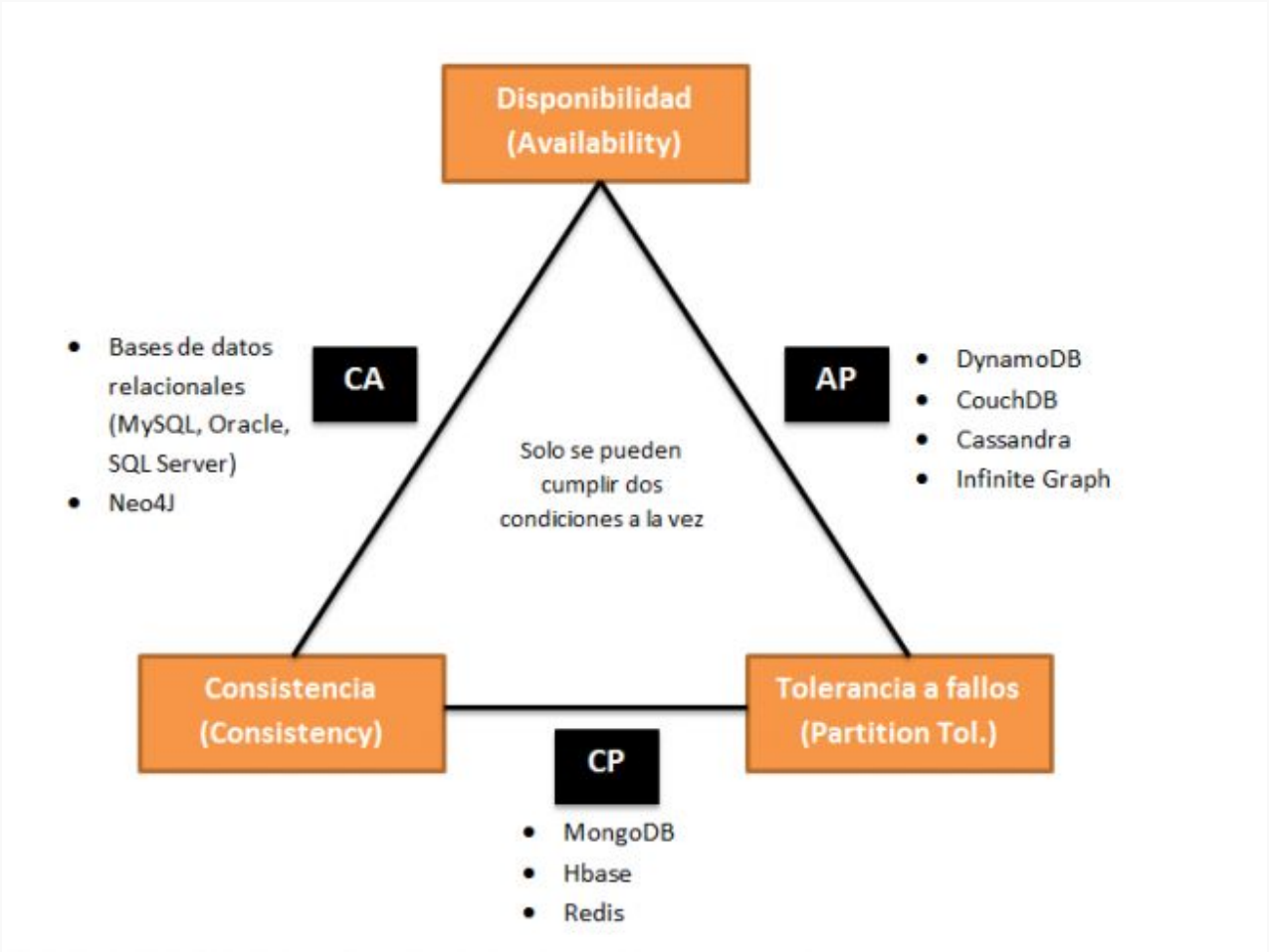
  Consistencia *(Consistency)* – todos los nodos ven los mismos datos al mismo tiempo.

  Disponibilidad *(Availability)* – garantiza que cada petición recibe una respuesta.

  Tolerancia a la partición *(Partition)* – el sistema continua funcionando a pesar de que fallen algunos nodos.

- Equivalente a:

  “You can have it good, you can have it fast, you can have it cheap: pick two.”


Disponibilidad (Availability)
Consistencia (Consistency)
Tolerancia a fallos (Partition Tol.)
Solo se pueden cumplir dos condiciones a la vez
CA
Bases de datos relacionales (MySQL, Oracle, SQL Server)
Neo4J
AP
DynamoDB
CouchDB
Cassandra
Infinite Graph
CP
MongoDB
Hbase
Redis

TEOREMA CAP

Figure 6.2 CAP Theorem: when partitioning your database, you need to choose between availability and consistency.

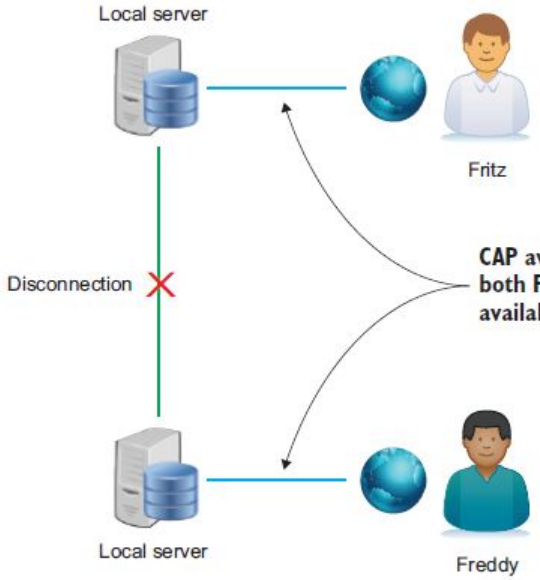


Figure 6.3 CAP Theorem: if nodes get disconnected, you can choose to remain available, but the data could become inconsistent.

ACID VS BASE

- En las BD relacionales, estamos familiarizados con las transacciones ACID, que garantizar la consistencia y estabilidad de las operaciones pero requieren bloqueos sofisticados:

  ACID = **A**tomicidad, **C**onsistencia, (**I**solation) aislamiento y **D**urabilidad

- Las BBDD NoSQL son repositorios de almacenamiento más optimistas , siguen el **modelo BASE**:

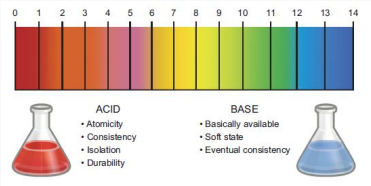
  Basic availability – funciona la mayoría del tiempo incluso ante fallos gracias al almacenamiento distribuido y replicado

  Soft-state – no tienen porque ser consistentes sus réplicas en todo momento.

  - El programador puede verificar esa consistencia.

  Eventual consistency – la consistencia se da pasado cierto tiempo

- **BASE es una alternativa flexible a ACID** para aquellas bases de datos que no requieren un manejo estricto de las transacciones.



Revisar más detalle en: https://queue.acm.org/detail.cfm?id=1394128

Arquitectura de las NoSQL

- A menudo ofrecen sólo **garantías de consistencia débiles**, como por ejemplo *eventual consistency*, o transacciones restringidas a elementos de datos simples
- Emplean una **arquitectura distribuida**, donde los datos se guardan de modo redundante en distintos servidores, a menudo usando tablas hash distribuidas.
- Suelen ofrecer **estructuras de datos sencillas** como arrays asociativos o almacenes de pares clave-valor.

APLICACIONES

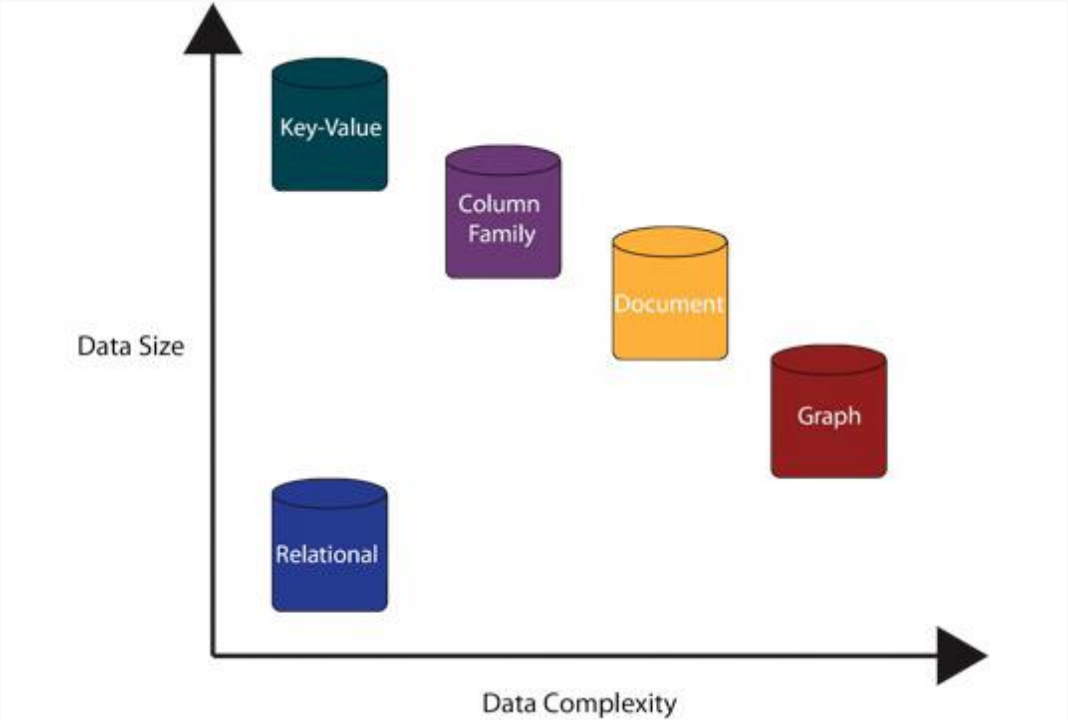
- Las aplicaciones son más complejas, el programador debe ser consciente del comportamiento de las transacciones soportadas por la tecnología escogida.
- Deben definir si aceptan los datos inconsistentes o si se definirá consistencia en el tiempo.

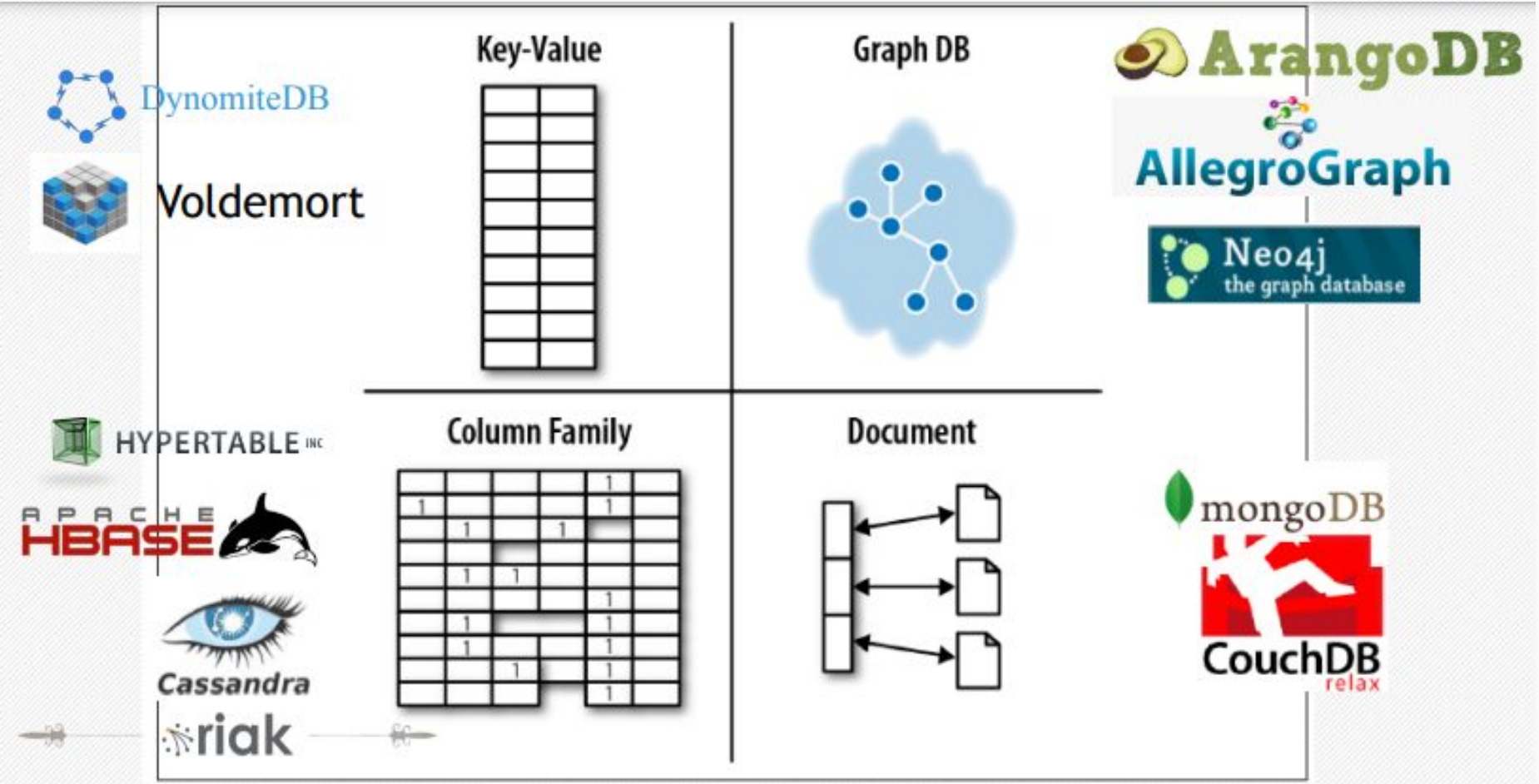
APLICACIONES

- Su uso es adecuado en:
  - Manejo de grandes volúmenes
  - Frecuencia alta de accesos de lectura y escritura
  - Cambios frecuentes en los esquemas de datos
  - Y que no requieran propiedades ACID

TIPOS DE BD NOSQL

- Clave-valor
- Orientadas a documentos
- Orientadas a columnas
- Orientadas a grafos
- Orientadas a objetos





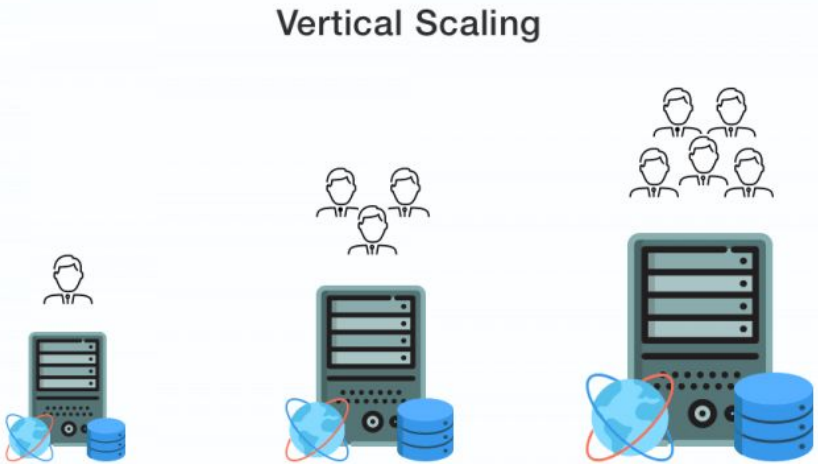
CONCLUSIONES

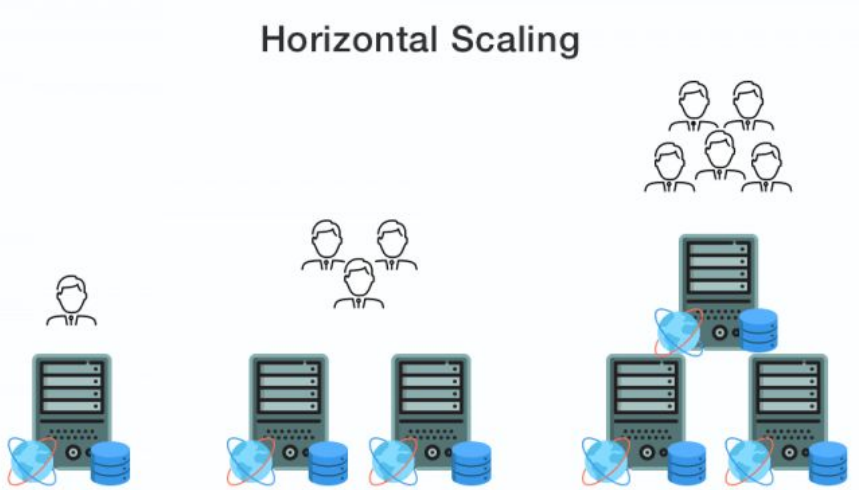
- Se recuperan los datos más rápidamente que en RDBMS, consultas más limitadas. La complejidad se traslada a la aplicación.
- Si se requiere manejo transaccional y garantía de ACID es mejor usar RDBMS.
- NoSQL no es adecuado para consultas complejas (… a excepción de las Orientadas a grafos, soportan consultas muy complejas).
- La tendencia es a que convivan diferentes tipos de BD. Persistencia políglota.

Referencias

- Eric Brewer, "Towards Robust Distributed Systems”
- https://people.eecs.berkeley.edu/~brewer/cs262b-2004/PODC-keynote.pdf
- Base: An Acid Alternative. https://queue.acm.org/detail.cfm?id=1394128
- NoSQL http://nosql-database.org/
- Bases de datos NoSQL. Introducción. Marta Zorrila, Diego Garcia. Enero 2017.
- https://ocw.unican.es/pluginfile.php/2396/course/section/2473/Tema%201.%20NoSQL%20introduccio%CC%81n.pdf
- Harrison et al. Next Generation Databases: NoSQL, NewSQL, and Big Data. 2015. Apress.
- Aguilar, L. J. (2016). Big Data, Análisis de grandes volúmenes de datos en organizaciones. Alfaomega Grupo Editor.

TIPOS DE ESCALABILIDAD





Tomado de: https://www.grapheverywhere.com/escalabilidad-de-bases-de-datos/

MAP REDUCE

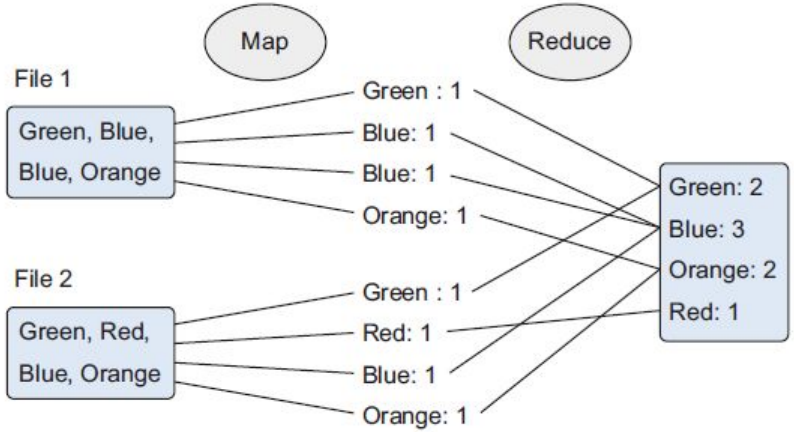
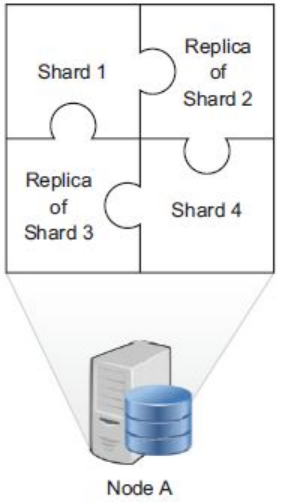


Figure 5.3 A simplified example of a MapReduce flow for counting the colors in input texts

Tomado de: Cielen, D., Meysman, A., & Ali, M. (2016). Introducing Data Science.

CONSISTENCIA EVENTUAL



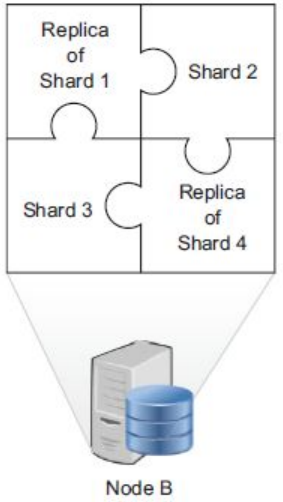


Figure 6.5 Sharding: each shard can function as a self-sufficient database, but they also work together as a whole. The example represents two nodes, each containing four shards: two main shards and two replicas. Failure of one node is backed up by the other.

Tomado de: Cielen, D., Meysman, A., & Ali, M. (2016). Introducing Data Science.

BALANCEO DE CARGA

- Centralizado
- Semi-distribuido
- Completamente distribuido