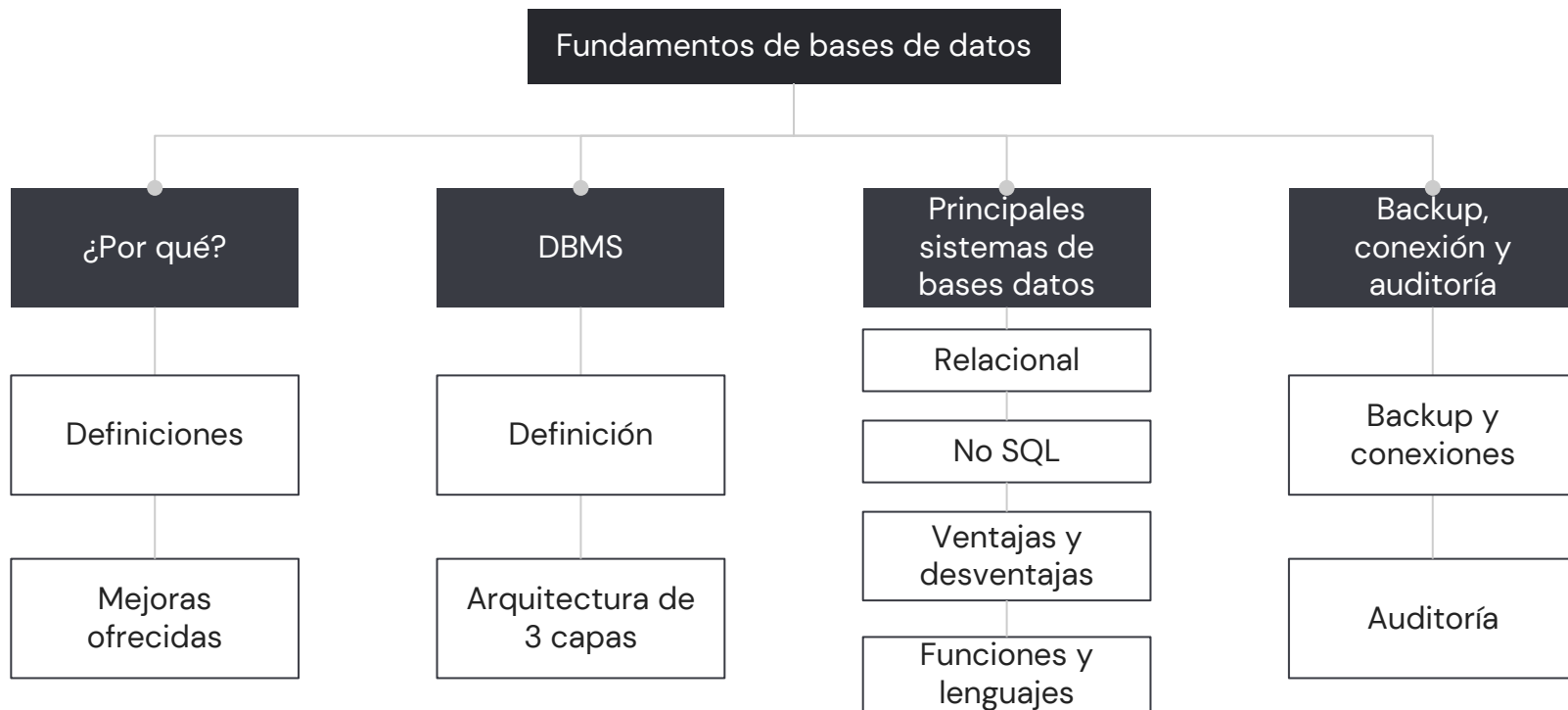


UNIDAD 1. MACHINE LEARNING PARA LA CIENCIA DE
DATOS

Bases de Datos y Aplicaciones

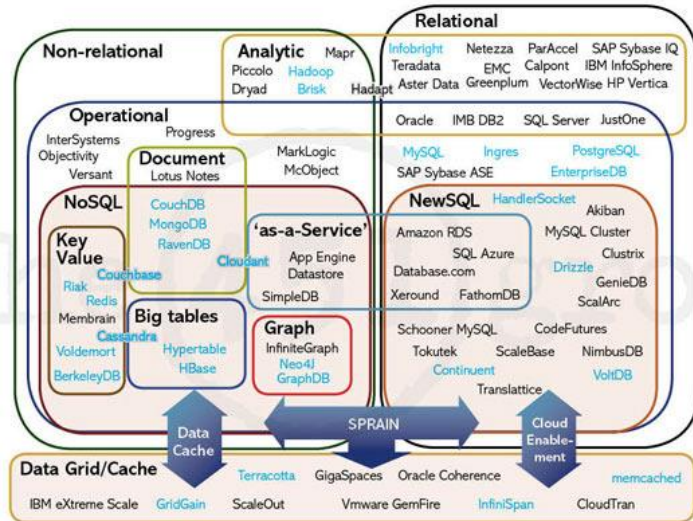
MAPA DE CONCEPTOS



Introducción a las Bases de datos

Tipos de bases de datos

Tipos de Bases de datos



SQL: Colección de tablas que almacena un conjunto específico de datos estructurados.

NoSQL: variedad de modelos en datos, incluidos los formatos de clave-valor, documento, columna y grafos.

NewSQL: NewSQL tiene una escalabilidad como NoSQL y garantiza estructura ACID de RDBMS

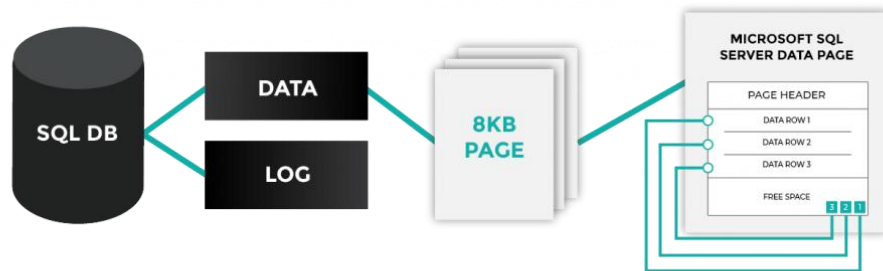
Analytical: almacena y administra big data usando Business Intelligence (BI).

As a service: servidor que utiliza una aplicación de base de datos que proporciona servicios de base de datos a otros programas informático.

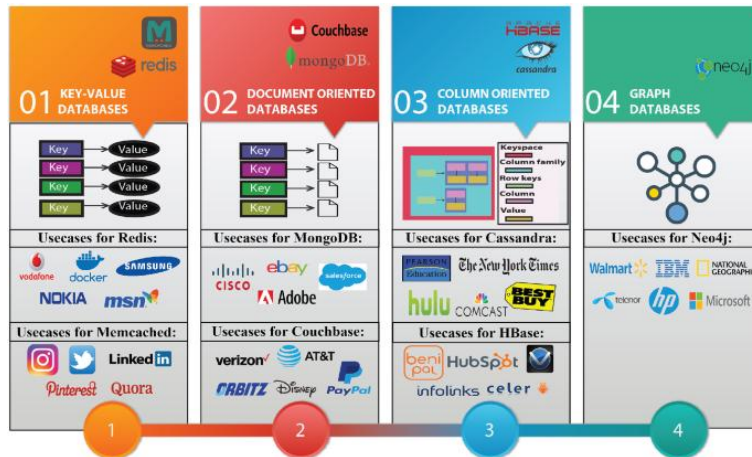
Tipos de Bases de datos - SQL

Se basa en el modelo de datos relacionales, que almacena datos en forma de filas (tupla) y columnas (atributos), y juntos forman una tabla (relación). Utiliza SQL para almacenar, manipular y mantener los datos. Inventado por EF Codd en 1970.

Algunos ejemplos de bases de datos relacionales son MySQL, Microsoft SQL Server, Oracle, etc.



Tipos de Bases de datos - NoSQL



- ✓ Es un tipo de base de datos que se utiliza para almacenar una amplia gama de conjuntos de datos. No solo almacena datos en forma tabular sino de varias maneras diferentes.
- ✓ Surgió cuando aumentó la demanda de creación de aplicaciones modernas, creado por Carlo Strozzi en 1998.
- ✓ NoSQL tiene una amplia variedad de tecnologías en respuesta a las demandas: **Key-value (Riak, Redis), Documentos (CouchDB, Mongo DB), Grafos (Neo4j, GraphDB) y Wide-column (Cassandra, HBase)**

Una definición mejorada

La base de datos es un conjunto de datos no redundantes, almacenados en un soporte informático, organizados en forma independiente de su utilización y accesibles simultáneamente por distintos usuarios y aplicaciones.



Características importantes

No sean redundantes	Se refiere a la eficiencia en su almacenamiento
Están almacenados en soporte informático :	Se refiere a la necesidad de contar con medios físicos de almacenamientos que garanticen seguridad
Están organizados en forma independiente de su uso	No se necesita saber cómo están almacenados para poder acceder a los datos
Pueden ser accedidos en forma simultánea	Se refiere a la flexibilidad de poder utilizar un sistema específico

Mejoras ofrecidas por Bases de datos

Sistema DBMS

Las bases de datos clásicas están administradas por un sistema particular adecuado para tal fin denominado **Sistema de Gestión de Bases de Datos (DBMS: Data Base Management System)**



Algunas mejoras ofrecidas

✓ **Independencia:** de la representación de los datos con respecto a su almacenamiento.

✓ **Eficiencia:** se utilizan técnicas específicas para acelerar la consulta y utilización de los datos almacenados en la base de datos.

✓ **Integridad:** se ofrecen diversos niveles de protección frente a fallos.

✓ **Seguridad:** complejo sistema para otorgar permisos de tal forma de garantizar el acceso a quien corresponda.

Algunas mejoras ofrecidas

✓ **Centralización:** permite que sean administrados de forma centralizada.

✓ **Acceso concurrente:** el acceso por parte de múltiples usuarios es complejo, y existen muchas situaciones donde los datos podrían corromperse. El DBMS provee mecanismos para evitar estas situaciones.

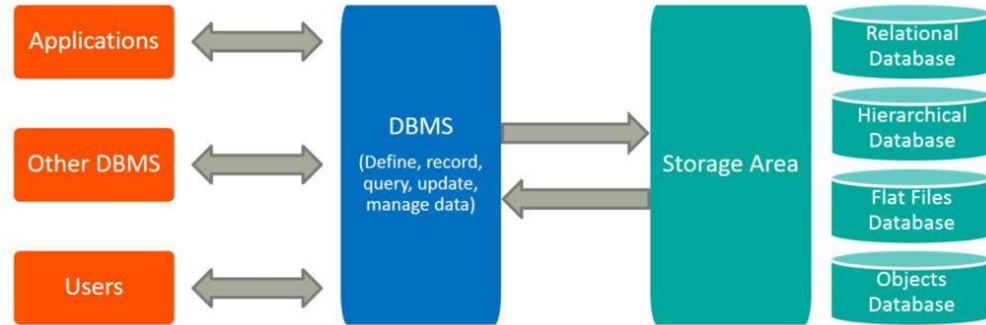
✓ **Reusabilidad:** situaciones comunes de acceso y uso y utilizarlas a través de distintas aplicaciones.

Sistemas de gestión de bases de datos

Data Base Management System

Solución tecnológica utilizada para optimizar y administrar el almacenamiento y la recuperación de datos de las bases de datos.

Ofrece un enfoque sistemático para administrar bases de datos a través de una interfaz para usuarios





PARA RECORDAR

Es un software específicamente diseñado para **definir, manipular y utilizar la información** que contienen las bases de datos.

Permite realizar las tareas de administración para garantizar su operatividad y además mantener las características de integridad y seguridad de los datos

Principales tipos de sistemas de gestión de **Bases de datos**

Tipos de bases de datos comunes

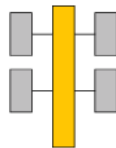


SQL Database

Relational

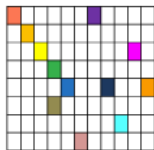


Analytical (OLAP)



NoSQL Database

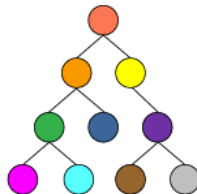
Column-Family



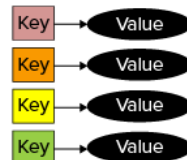
Graph



Document



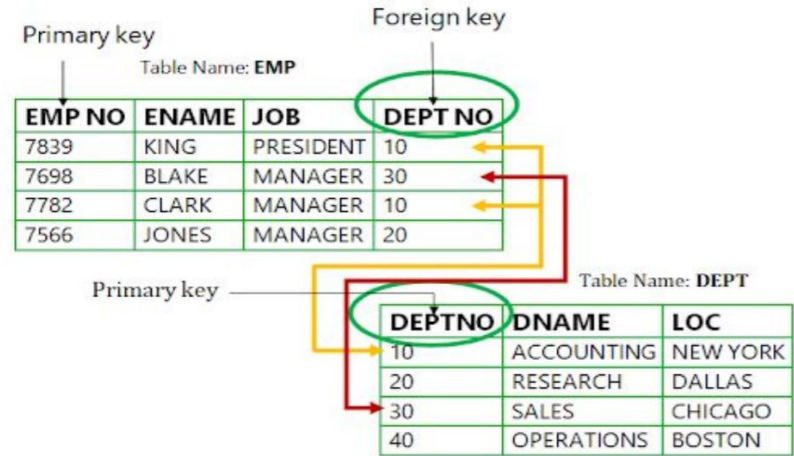
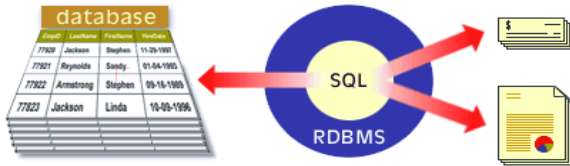
Key-Value



Relacional

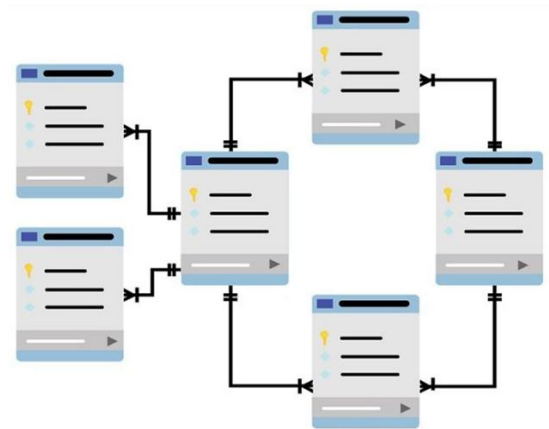
Relacionales

Responden al **Modelo de Datos relacional** propuesto por **Edward Frank Codd en 1970**, tal cual como podemos observar en la siguiente imagen:



Relacionales

- ✓ Representan **los datos como un conjunto de tablas en dos dimensiones, formadas por filas y columnas**. Cada fila representa una forma de relación entre un conjunto de valores y está identificada de forma única.
- ✓ A menudo, **las bases de datos relacionales contienen múltiples tablas relacionadas entre sí**, lo cual permite que la información pueda estar almacenada de una forma más compacta.

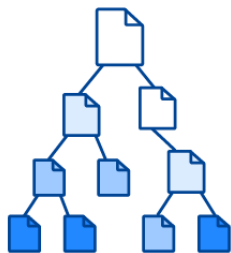


No SQL

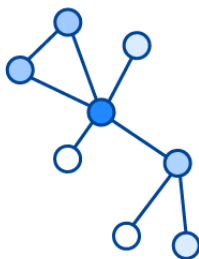
No Relacionales

Modelo propuesto por **Carlo Strozzi en 1998**, como una base de datos "relacional" de código abierto y liviana que no usa SQL, desarrollado en principio para datos web (**no estructurados**) y por la necesidad de un procesamiento más rápido.

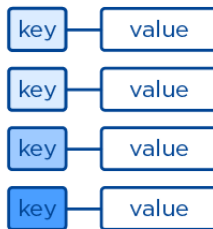
Document



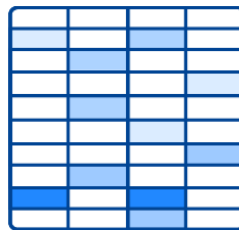
Graph



Key-Value



Wide-column



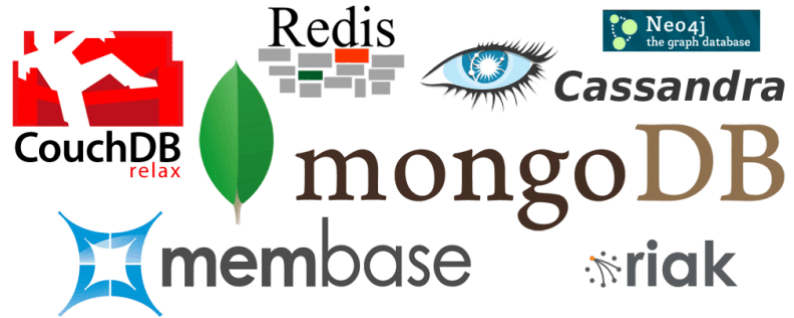
No Relacionales

Engloba muchos subtipos que comparten como característica fundamental **no estar organizados en tablas**, y gracias a ello, pueden ser accedidos mediante otras formas acordes a su uso específico.

🗃️ ¡Aunque aclaramos que muchas de ellas también pueden ser accedidas por SQL!

Algunos ejemplos

- ✓ **Orientadas a objetos:** almacenan los objetos tal cual como son utilizados en los lenguajes de programación orientados a objetos.
- ✓ **Orientadas a gráficos:** guardan los datos como nodos y relaciones, y pueden ser consultados de forma automatizada.
- ✓ **Orientadas a documentos:** guardan conjuntos de texto muy grandes (los documentos), y este texto puede estar a su vez estructurado dentro de cada documento.



Ventajas y desventajas de bases de datos relacionales y No relacionales

Ventajas y desventajas bases SQL

Ventajas	Desventajas
Simplicidad del modelo: Muy simple, no requiere consultas complejas	Mantenimiento: difícil por acumulación de datos en el tiempo
Fácil uso: usuarios pueden acceder/recuperar fácilmente la información requerida en segundos sin caer en la complejidad.	Costo: se generan costos fijos y variables por mantenimiento
Precision: bien definidas y organizadas, no duplicados.	Almacenamiento físico: requiere mucha memoria física.

Ventajas y desventajas bases SQL

Ventajas	Desventajas
Integridad de datos: brindan coherencia en todas las tablas.	Poca escalabilidad: los datos no son escalables en diferentes servidores de almacenamiento físico
Normalización: se divide la información en partes manejables para reducir el tamaño del almacenamiento	Estructura compleja: sólo puede almacenar datos en forma tabular, dificultando representación compleja.
Colaboración: muchos usuarios interactuando al tiempo	Reducción de performance en tiempo: mayor complejidad
Integridad y Seguridad: Sistemas medianamente confiables	Menor tiempo de respuesta: muchos datos poca eficiencia

Ventajas y desventajas bases No SQL

Ventajas	Desventajas
Modelo flexible: puede almacenar y combinar cualquier tipo de datos, tanto estructurados como no estructurados	Falta de estandarización: No existe un estándar que defina reglas y roles de las bases de datos NoSQL.
Modelo de datos en evolución: permite actualizar dinámicamente el esquema para evolucionar con los requisitos cambiantes sin interrupciones.	Algunos problemas de backup: No está del todo desarrollado este ámbito en este tipo de bases de datos.
Fácil escalamiento: pueden escalar para adaptarse a cualquier tipo de crecimiento de datos manteniendo un bajo costo.	Consistencia: NoSQL prioriza la escalabilidad y el rendimiento, pero cuando se trata de la consistencia de los datos no es tan eficiente.

Ventajas y desventajas bases No SQL

Ventajas	Desventajas
Alto performance: gran rendimiento, medido en términos de rendimiento y latencia (retraso entre la solicitud y la respuesta real).	Difícil mantenimiento: pueden llegar a ser costosos y requerir de personal especializado
Acceso libre: no requieren tarifas de licencia costosas y pueden ejecutarse en hardware económico, lo que hace que su implementación sea rentable.	Poco nivel de madurez: Son relativamente más nuevas que las bases relacionales por ende tienen todavía mucho por mejorar.