

Tarea 1

Unidad 3

Sistemas de Big Data

Mecanismos de
Almacenamiento en
Big Data

Índice

Bases de datos relacionales	3
Sistemas de archivos distribuidos	4
Almacenes de objetos en la nube	6
Bases de datos NoSQL	7
Comparativa	9
Bibliografía	10

Bases de datos relacionales

Es un tipo de base de datos que cumple con el modelo relacional. Funciona por un sistema de clave-valor. Almacenan los datos de manera estructurada y relacionada entre sí.

Cada fila representa un registro, y cada registro un atributo de una entidad. La relación entre entidades se establece mediante claves primarias y foráneas.

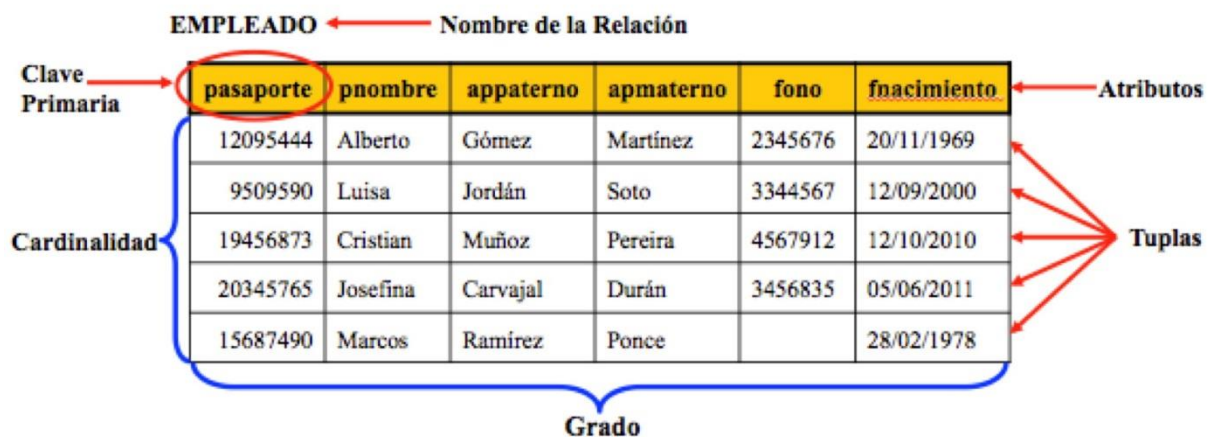
Algunas características:

- Evita la duplicidad.
- Garantiza la integridad referencial.
- Transacciones ACID.

Para guardar los datos necesitamos usar un SGBD (Sistema gestor de base de datos).

Ventajas	Desventajas
Evita duplicidad	Malos para manejar datos gráficos
Se puede bloquear el acceso a datos mientras se están actualizando, para evitar conflictos	Rigidez en la estructura, no se pueden crear subfilas
Integridad de los datos	Escalabilidad limitada
Permite mantener la uniformidad de los datos (mismos datos en todo momento). Transacciones ACID	Rendimiento negativo a la hora de segmentar en distintas tablas los datos

Imagen de la estructura básica de una tabla de una base de datos relacional:



En cuanto a tipos de despliegue, tenemos:

- **On-premise:** La base de datos se instala y gestiona en local.
- **En la nube:** Usando plataformas como AWS o Azure, donde se tiene menos control de datos, pero es más simple y económico.
- **Híbrido:** Donde se combinan los 2 despliegues anteriormente mencionados.

Algunos ejemplos de campos de aplicación podrían ser:

- **Sistemas ERP y CRM:** Gestiones de compras, inventario.. etc.
- **Sistemas de gestión de la salud:** Gestiones de pacientes, historiales.. etc.
- **Campo de la educación/finanzas:** Administración de cuentas, alumnos.. etc.

Sistemas de archivos distribuidos

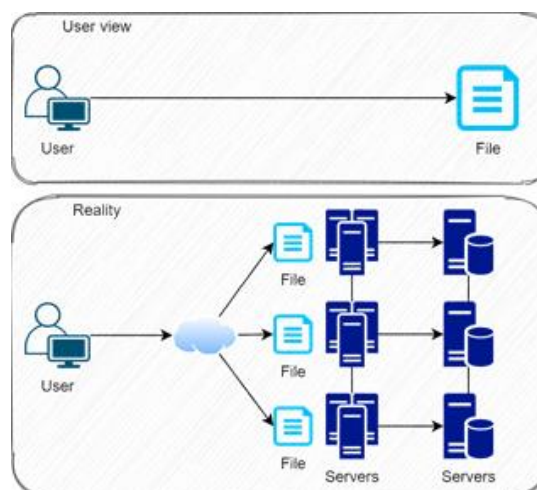
Es un sistema de almacenamiento que permite a los usuarios acceder a los archivos.

El funcionamiento es básicamente que almacena y accede a archivos distribuidos en múltiples servidores, mientras se presenta al usuario como un único sistema de archivos.

Algunas características:

- Transparencia de acceso
- Transparencia de ubicación: los usuarios verán un único espacio de nombres para todos los archivos de datos
- Bloqueo de archivos
- Cifrado de datos
- Compatibilidad con protocolos

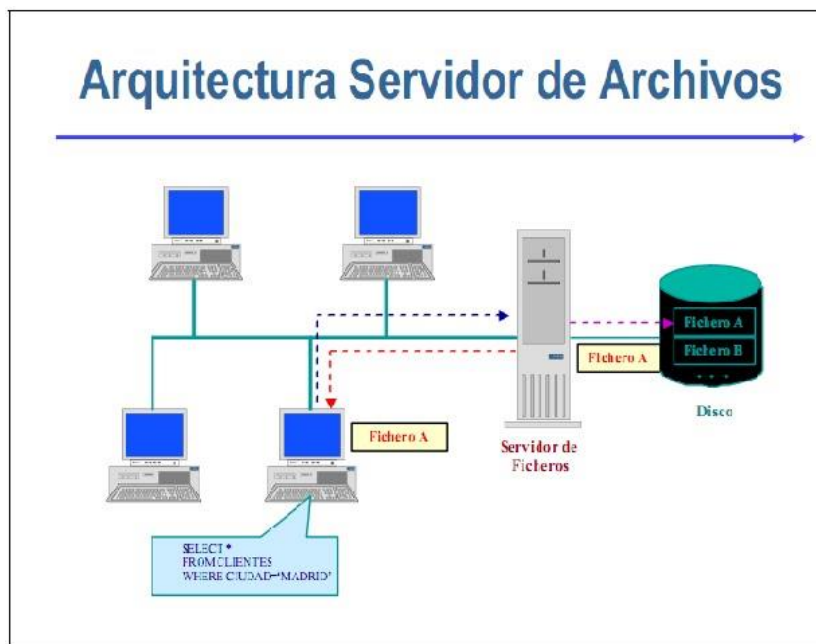
Imagen de la transparencia de acceso:



En cuanto a su arquitectura, existen:

- **Nodos de datos:** Suelen ser los servidores de una red distribuida
- **Nodos de metadatos:** Contiene información de los directorios, se suele distribuir para evitar cuellos de botella
- **Clientes:** Proporciona una interfaz que permite a los clientes interactuar con el sistema.
- **Red de comunicación:** Conecta los nodos y permite la transferencia de datos.

Imagen visual de la arquitectura de archivos:



Ventajas	Desventajas
Resiliencia de los datos	Mayor complejidad
Compartir información de manera rápida, fácil y eficiente	Redundancia de los datos
Escalabilidad sencilla	Costoso al inicio

En cuanto al despliegue, se encuentran:

- Entornos locales/privados
- En la nube
- Sistemas de alta disponibilidad

Algunos ejemplos de uso:

- Big Data
- Machine Learning
- Simulaciones

Almacenes de objetos en la nube

Es una arquitectura diseñada para manejar grandes cantidades de datos no estructurados (Por ejemplo: correos electrónicos, archivos multimedia, audio..etc.)

Los bloques de datos se mantienen juntos como un objeto y se colocan en un entorno de datos como grupo de almacenamiento.

Cuando se desee acceder, usarán un identificador único y los metadatos para encontrar el objeto.

Se puede buscar objetos y acceder a ellos mediante APIs de RESTful, HTTP, HTTPS.

Algunos beneficios de ello pueden ser:

- Elasticidad
- Escalabilidad
- Fácil uso e integración

En cuanto a limitaciones tenemos que:

- Hay un límite en cuanto al tamaño de los archivos
- Latencia variada, ya que accedes desde la nube
- Riesgos de seguridad

Caso de uso: Archivar grandes cantidades de archivos multimedia en la nube y añadir metadatos personalizados para facilitar su recuperación.

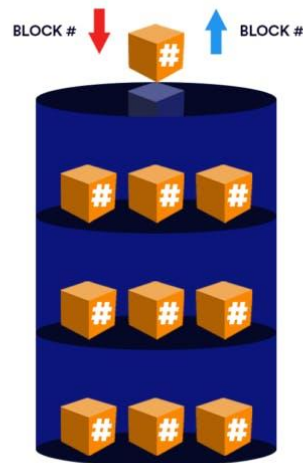
Otro podría ser el respaldo y recuperación de datos.

Algunos proveedores de servicios en la nube podrían ser

- AWS
- Google Cloud
- Microsoft Azure

Imagen representativa de los almacenes de objetos en la nube

BLOCK STORAGE



Bases de datos NoSQL

Son bases de datos que no utilizan el modelo tradicional de tablas relacionales.

Están diseñadas para ser más flexibles, escalables y manejar grandes volúmenes de datos no estructurados o semi estructurados.

Los principales tipos de bases de datos NoSQL:

- **Base de datos de documentos:**

Se almacenan en documentos (JSON, XML). Cada documento puede tener una estructura diferente y se identifica mediante clave única.

Un ejemplo de este tipo de base de datos podría ser MongoDB.

Características destacadas:

- Flexibilidad
- Escalabilidad horizontal
- Consultas eficientes

Donde más se utilizan son en:

- Aplicaciones web
- Análisis de grandes volúmenes

- **Bases de datos clave-valor:**

Se almacenan como pares de clave-valor. Cada elemento tiene una clave única asociada a un valor.

Un ejemplo de este tipo de base de datos podría ser Redis.

Características destacadas:

- Simplicidad
- Escalabilidad horizontal
- Alto rendimiento en operaciones simples

Donde más se utilizan son en:

- Almacenamiento de sesiones, caché o memoria
- Sistemas de caché

- **Bases de datos de grafos:**

Se representan como un conjunto de nodos, aristas y propiedades. Son especialmente útiles para representar relaciones complejas entre entidades.

Un ejemplo de este tipo de base de datos podría ser Neo4J

Características destacadas:

- Relaciones complejas
- Escalabilidad

Donde más se utilizan son en:

- Redes sociales
- Motores de recomendación

- **Bases de datos en columnas:**

En lugar de almacenar los datos en filas (como las bases de datos relacionales), se almacenan en columnas, lo que facilita el acceso rápido de grandes volúmenes de datos que requieren leer columnas específicas.

Un ejemplo de este tipo de base de datos podría ser Apache Cassandra

Características destacadas:

- Optimización para lectura masiva
- Escalabilidad

Donde más se utilizan son en:

- Almacenamiento de grandes volúmenes de datos históricos
- Análisis de datos en tiempo real o masivos

Imagen representativa de como sería cada tipo de base de datos nosql

Columnar:



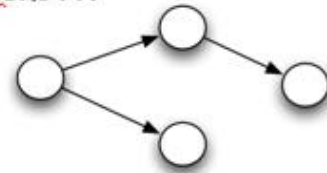
Documental:

```
[ "user" : {  
  "id": "124",  
  "name": "Emmanuel",  
  "addresses" : [  
    { "city": "Paris", "country": "France" },  
    { "city": "Atlanta", "country": "USA" }  
  ]  
}]
```

Key-value:

key	value
123	Address@23
126	"Booya"

Graph:



Comparativa

Diferencias entre las bases de datos relacionales y no relacionales (SQL y NoSQL)

Características	SQL	NoSQL
<u>Modelo de datos</u>	Tabla con filas y columnas, estricto.	Documentos, clave-valor, columnas, grafos. Flexible
<u>Escalabilidad</u>	Horizontal	Vertical
<u>ACID</u>	Soporte ACID	Generalmente no soporta ACID
<u>Consistencia</u>	CAP (Consistencia, Disponibilidad y Tolerancia a particiones)	Eventual, depende de la implementación
<u>Integridad de los datos</u>	Alta, restricciones con claves foráneas y validación de datos	Menos controlado
<u>Flexibilidad</u>	Baja flexibilidad	Alta flexibilidad

<u>Despliegue en la nube</u>	Infraestructura dedicada o virtualizada	Servicio de nube completamente gestionado
<u>Casos de uso</u>	Aplicaciones transaccionales, ERP	Big Data, redes sociales, IoT
<u>Análisis de datos</u>	Sistemas OLAP, BI	Big Data, Hadoop
<u>Aplicaciones Web y Móviles</u>	Aplicaciones de tamaño mediano con datos estructurados	Aplicaciones que manejan grandes volúmenes de datos no estructurados

Bibliografía

Base de datos relacional

<https://ayudaleyprotecciondatos.es/bases-de-datos/relacional/>

<https://azure.microsoft.com/es-es/resources/cloud-computing-dictionary/what-is-a-relational-database>

<https://appmaster.io/es/blog/ejemplos-del-mundo-real-de-bases-de-datos-relacionales>

<https://click-it.es/infraestructura-on-premise-cloud-o-hibrida-como-elegir-la-mejor-opcion-para-tu-negocio/>

Sistemas de archivos distribuidos

<https://www.nutanix.com/es/info/distributed-file-systems>

<https://ilimit.com/blog/importancia-arquitectura-distribuida/#:~:text=Una%20arquitectura%20distribuida%20se%20caracteriza,realiza%20con%20un%20%C3%BAnico%20nodo.>

<https://www.atlassian.com/es/microservices/microservices-architecture/distributed-architecture>

<https://iberasync.es/transparencia-de-acceso-y-escala-en-sistemas-distribuidos/>

https://es.wikipedia.org/wiki/Sistema_de_archivos_distribuido

<https://iboysoft.com/es/wiki/sistema-de-archivos-distribuidos.html>

Almacenes de datos en la nube

<https://cloud.google.com/learn/what-is-object-storage?hl=es-419>

<https://massive.io/es/transfencia-de-archivos/almacenamiento-de-archivos-vs-almacenamiento-de-objetos/#object-storage>

<https://aws.amazon.com/es/what-is/object-storage/>

<https://www.computerweekly.com/es/consejo/Comprendiendo-el-almacenamiento-de-objetos-vs-el-almacenamiento-de-bloques-para-la-nube>

<https://forum.huawei.com/enterprise/en/block-storage-vs-object-storage/thread/690226202436059136-667213859733254144>

Base de datos NoSQL

<https://www.ibm.com/es-es/topics/nosql-databases>

<https://cloud.google.com/discover/what-is-nosql?hl=es-419>

<https://es.wikipedia.org/wiki/NoSQL>

<https://aws.amazon.com/es/nosql/>

<https://aws.amazon.com/es/nosql/key-value/>

<https://www.unir.net/revista/ingenieria/nosql-vs-sql/>

<https://ilimit.com/blog/base-de-datos-sql-nosql/>

<https://www.astera.com/es/knowledge-center/sql-vs-nosql/>

<https://www.coursera.org/mx/articles/nosql-vs-sql>

<https://gltaboada.github.io/tgdbook/conceptos-y-tipos-de-bases-de-datos-nosql-documental-columnar-clavevalor-y-de-grafos.html>

<https://www.ionos.es/digitalguide/hosting/cuestiones-tecnicas/base-de-datos-columnar/>