



**Σavante**

**2**

BASE DE DATOS

**Fundamentos de las bases de datos**

# ÍNDICE

<b>/ 1. Introducción y contextualización práctica</b>	<b>4</b>
<b>/ 2. Definición y elementos de las bases de datos</b>	<b>5</b>
2.1. Concepto de base de datos	5
2.2. Elementos que componen una base de datos	5
2.3. Bases de datos según su uso	6
<b>/ 3. Bases de datos según el modelo de datos. Estandarizadas</b>	<b>6</b>
3.1. Base de datos relacionales	6
3.2. Bases de datos orientadas a objetos	7
3.3. Bases de datos objeto-relacionales	7
<b>/ 4. Bases de datos según el modelo de datos. Especializadas</b>	<b>7</b>
4.1. Bases de datos jerárquicas	7
4.2. Bases de datos en red	8
<b>/ 5. Bases de datos centralizadas y distribuidas</b>	<b>9</b>
5.1. Bases de datos centralizadas	9
5.2. Bases de datos distribuidas	9
5.3. Ventajas e inconvenientes de las distribuidas	10
5.4. Diseño de las bases de datos distribuidas	10

# ÍNDICE

<b>/ 6. Arquitectura de una base de datos distribuida</b>	<b>11</b>
6.1. Enfoques de definición de la arquitectura	11
6.2. Esquemas de la arquitectura de referencia	11
6.3. Implementaciones múltiples y federadas	12
6.4. Hardware y software de una BD distribuida	12
<b>/ 7. Fragmentación de la información</b>	<b>13</b>
<b>/ 8. Distribución de la información</b>	<b>14</b>
8.1. Algoritmos de distribución	15
8.2. Objetivos de la distribución de datos	15
<b>/ 9. Seguridad y recuperación en las bases de datos distribuidas</b>	<b>16</b>
<b>/ 10. Caso práctico 1:“Base de datos para tienda y almacén”</b>	<b>17</b>
<b>/ 11. Caso práctico 2. “¿Qué modelo usar?”</b>	<b>18</b>
<b>/ 12. Resumen y resolución del caso práctico de la unidad</b>	<b>19</b>
<b>/ 13. Bibliografía</b>	<b>21</b>

# OBJETIVOS

*Entender el concepto de base de datos y sus aplicaciones.*

*Conocer los tipos de bases de datos y sus diferencias.*

*Identificar los componentes clave de una base de datos y profundizar en sus aplicaciones.*

*Estudiar las características y funcionalidades de las bases de datos distribuidas.*

## / 1. Introducción y contextualización práctica

Como se vio en el tema 1, para la mayoría de los procesos que realizamos en nuestro día a día, es esencial un buen almacenamiento y una óptima gestión de los datos. Las bases de datos fueron creadas exactamente para esto.

En la década de los sesenta, varias compañías comenzaron a usar discos magnéticos para el almacenamiento de datos, lo que dio origen a esta tecnología. Posteriormente, se desarrollaron los archivos, que únicamente podían manejarse de forma secuencial. Los programas de esa época usaban aplicaciones muy específicas y rara vez interactuaban entre sí. Era común que una empresa utilizara un programa para gestionar las ventas y otro diferente para manejar los inventarios, y esto provocaba errores y falta de coherencia. La solución a este problema fue el uso de bases de datos. Estas permiten trabajar con diversos tipos de archivos interrelacionados. Además, el crecimiento exponencial de las TIC en los últimos años ha hecho que esta tecnología sea accesible no solo en el ámbito empresarial, sino también a nivel general.



Fig.1. El uso de BBDD está presente en nuestra vida cotidiana.

Este tema establecerá los conceptos básicos sobre las bases de datos, lo cual es esencial para poder profundizar y ampliar conocimientos en los temas siguientes.

A continuación, se presenta un caso práctico a través del cual podremos aproximarnos de forma aplicada a la teoría de este tema.

Escucha el siguiente audio donde planteamos la contextualización práctica de este tema. Encontrarás su resolución en el apartado 'Resumen y resolución del caso práctico'.



Audio Intro. "Pensemos en el diseño de nuestra base de datos"

<https://on.soundcloud.com/8vkVhrvTyA88k7VY7>



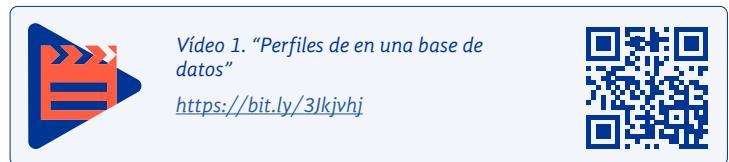
## / 2. Definición y elementos de las bases de datos

Antes de las bases de datos, la información se gestionaba con archivos independientes para cada sistema, lo que generaba problemas (que corrigen las bases de datos) como el aumento de la necesidad de espacio de almacenamiento, la poca flexibilidad en el manejo de los datos, los datos duplicados entre diferentes sistemas, las dificultades para la gestión de la información, las incongruencias en los datos de los diferentes sistemas o la incompatibilidad al tratar los datos desde fuera del sistema.

### 2.1. Concepto de base de datos

Las bases de datos son **herramientas que permiten estructurar el almacenamiento de datos, relacionarlos y ayudar a su gestión, consulta y manipulación, garantizando siempre su integridad**.

Otro concepto clave es el de **modelo de base de datos**, que **describe cómo se almacena la información y cómo se interrelaciona, definiendo así la arquitectura de la base de datos**.



### 2.2. Elementos que componen una base de datos

Son los siguientes:

- **Entidades**: Son los objetos sobre los que se guarda información, los cuales tienen características distintas entre sí. Por ejemplo, en una tienda, las entidades pueden ser los clientes.
- **Tablas**: Organizan la información a guardar de cada entidad. Por ejemplo, los datos de cada cliente (nombre, DNI, dirección, etc.).
- **Campos**: Identifican los diferentes tipos de datos y se representan como columnas en la tabla. Por ejemplo, el campo 'DNI' en la tabla 'clientes'.
- **Registros**: Un registro es una fila en la tabla. Por ejemplo, en la tabla de clientes, cada registro representa un cliente.

La siguiente tabla muestra un ejemplo de la tabla de clientes:

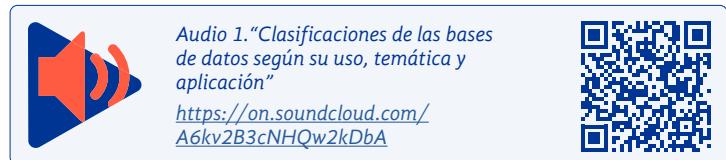
ID	DNI	NOMBRE	APELLIDOS	LOCALIDAD
1	54621345D	Esther	Rodríguez Méndez	Pontevedra
2	85542549E	Javier	Ramírez Lorenzo	Zamora
3	24581392F	Silvia	Castro Gutiérrez	Ourense

Tabla 1. Tabla de ejemplo con datos de clientes.

## 2.3. Bases de datos según su uso

Según el uso que hagamos de ellas, podemos clasificarlas en:

- **Individual:** Es la base de datos utilizada por una persona que asume todos los roles de usuario, el de administrador, programador, etc.
- **Compartida:** Es aquella en la que múltiples usuarios acceden a la misma información. Por ejemplo, la base de datos de una pyme.
- **Bancos de datos:** Almacenan grandes volúmenes de datos específicos desarrollados para consulta. Por ejemplo, [BASE](#).
- **Pública:** Esta tipología permite el acceso a cualquier usuario que cuente con los medios técnicos para hacerlo. Por ejemplo, la base de datos del catastro.



## / 3. Bases de datos según el modelo de datos. Estandarizadas

Destacan por su alta estandarización, robustez y el uso de normas aceptadas como SQL, garantizando interoperabilidad, consistencia y facilidad de uso y mantenimiento.

### 3.1. Base de datos relacionales

**Almacenan datos en tablas con filas y columnas que se relacionan entre sí.** Su estructura asegura la integridad de los datos. La simplicidad, flexibilidad y el uso de SQL han facilitado su expansión, permitiendo el manejo eficiente de **grandes volúmenes de datos y consultas complejas**.

En la siguiente imagen podemos ver un ejemplo de dos tablas y cómo se relacionan los datos de ambas.

Matrícula	Marca	Modelo	Propietario	ID	Nombre
7891GHR	Seat	León	1	1	Maria
1897DTR	Ford	Focus	2	2	Daniel
6375JKR	Mercedes	Clase E	3	3	Esther

Fig. 2. Modelo de base de datos relacional.

### 3.2. Bases de datos orientadas a objetos

Surgieron en los años 90 con la programación orientada a objetos. **Almacenan información como objetos definidos por clases**, que incluyen atributos y métodos. También permiten almacenar objetos multimedia como vídeos, imágenes y audios.

Clase	Objeto	Datos o atributos
Vídeos	Vídeo 1	Título: "Sample Vídeo 1" Formato: "mp4" Duración: "10 min" Resolución: "1920x1080" Datos: [binary data]

Tabla 2. Modelo de base de datos orientada a objetos.

### 3.3. Bases de datos objeto-relacionales

Combinan características de las bases de datos orientadas a objetos y las relacionales. Mejoran las relacionales añadiendo ventajas de las orientadas a objetos. Fabricantes como Oracle, Microsoft e IBM las prefieren por su capacidad para mejorar productos y reducir costes. Sus principales aportes incluyen tipos de datos avanzados, procedimientos y funciones definidos por usuarios, y la capacidad de realizar consultas recursivas.

En este audio explicaremos las bases de datos multidimensionales:



Audio 2. "Bases de datos multidimensionales"  
<https://on.soundcloud.com/7tYwG67KcCZ4eZ8j6>



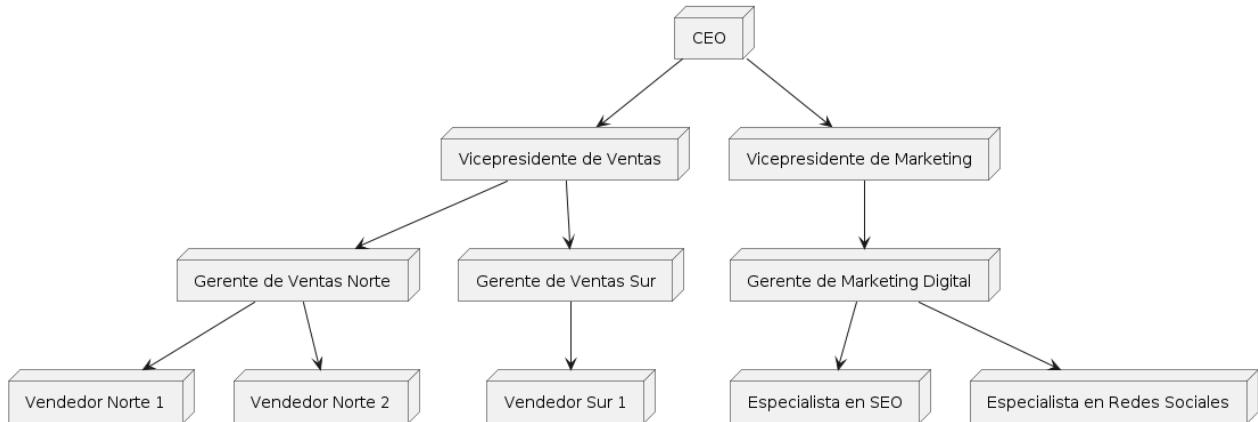
## / 4. Bases de datos según el modelo de datos. Especializadas

Como hemos mencionado anteriormente, existen diferentes maneras de clasificar las bases de datos. En este apartado nos centraremos en las que podríamos calificar, en base a su modelo de datos, como estructuradas y más rígidas.

### 4.1. Bases de datos jerárquicas

Las bases de datos de organización jerárquica son las más antiguas de esta clasificación. **Surgen con la aparición de los discos magnéticos**, que permitieron el acceso a la información de una forma aleatoria, no secuencial como hasta entonces.

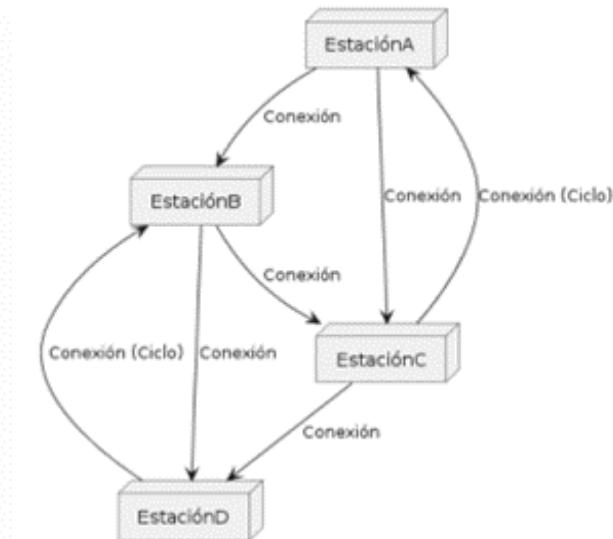
En ellas, la información se almacena **en forma de listas o árbol y ramas**, manteniendo una **relación padre-hijo**. En ellas cada nodo se caracteriza por tener un único parente, mientras que puede ser parente de múltiples hijos, como se muestra en esta imagen:



*Fig.3. Modelo de base de datos jerárquica de los puestos de una empresa.*

## 4.2. Bases de datos en red

Son una evolución de las bases de datos de organización jerárquica. La principal diferencia entre ambas es que las **relaciones** en estas pueden ser **circulares**. Es decir, un nodo puede tener varios padres y ser a la vez parente de alguno de ellos, como podemos ver en la siguiente figura.



*Fig.4. Modelo de base de datos en red.*

En este audio explicaremos las bases de datos deductivas y NoSQL:



Audio 3. "Bases de datos deductivas y NoSQL"  
<https://on.soundcloud.com/wU53CypSysrXg7t98>



## / 5. Bases de datos centralizadas y distribuidas

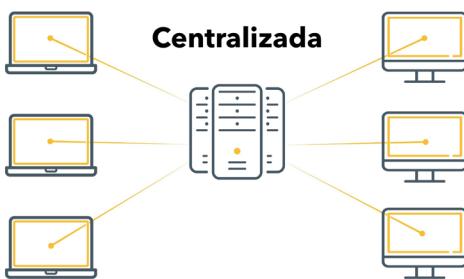
Ante todo, es importante entender que la diferenciación entre bases de datos centralizadas y distribuidas es otra forma de categorizar las bases de datos según la ubicación de la información.

### 5.1. Bases de datos centralizadas

Las bases de datos centralizadas se distinguen, principalmente, por estar **localizadas en un solo dispositivo o máquina**, que puede ser de diferentes tipos.

Aunque su aplicación es diversa, suelen encontrarse en empresas o incluso en sistemas domésticos de un solo usuario.

Esto significa que, a pesar de estar centralizadas, pueden manejar grandes volúmenes de datos con un alto nivel de rendimiento.



*Fig.5. Arquitectura de base de datos centralizada.*

### 5.2. Bases de datos distribuidas

Las bases de datos distribuidas aparecieron en los años 70 para **gestionar datos a gran escala en múltiples ubicaciones**, enfrentando inicialmente desafíos de coordinación y consistencia de datos. Gracias a los avances tecnológicos y en protocolos, su uso se extendió, garantizando la disponibilidad de datos en múltiples centros de datos alrededor del mundo, en empresas como Amazon y Google. Actualmente, son fundamentales para muchas aplicaciones esenciales y se prevé que la inteligencia artificial y el aprendizaje automático aceleren su desarrollo futuro.

A diferencia de las bases de datos centralizadas, estas se distinguen por que **la información no se encuentra en un solo dispositivo, sino que se distribuye entre varios nodos**.

**Las bases de datos distribuidas pueden describirse como conjuntos de bases de datos interconectadas por redes de comunicación**, a través de las cuales los usuarios pueden acceder a la información del sistema.

Se trata de multiplicar los dispositivos o nodos que controlan la base de datos colocándolos en ubicaciones diversas y facilitando el intercambio y acceso de datos entre ellos a través de la red.

#### 🔗 Enlaces de interés...

Te dejamos el siguiente enlace de interés, donde encontrarás un vídeo que muestra las principales características de una base de datos distribuida:  
<https://www.youtube.com/watch?v=32USrHr36uU>

## 5.3. Ventajas e inconvenientes de las distribuidas

Las bases de datos distribuidas proporcionan múltiples ventajas a distintos niveles.

**A nivel técnico:**

- Aumentan la confiabilidad, disponibilidad y eficiencia en el acceso a la información.
- Son más modulables y flexibles.
- Son fácilmente escalables añadiendo nuevos nodos.
- Optimizan y mejoran el rendimiento al compartir recursos entre nodos.
- Funcionan con autonomía de cada nodo, sin dependencia de un único nodo.
- Permiten el acceso concurrente desde distintas ubicaciones geográficas.

**A nivel económico:** Logran la optimización de los costes de explotación, administración y crecimiento de la red.

**A nivel de estructura u organizativo:**

- Facilitan la adaptabilidad a la institución a la que sirven.
- Ayudan a la fusión o división de compañías y organizaciones.
- Proporcionan independencia local.

Sin embargo, presentan diversos inconvenientes. En el siguiente audio explicaremos algunos:



Audio 4. "Inconvenientes de las bases de datos distribuidas"  
[https://on.soundcloud.com/  
 ZWCJWFMDXHqncjZV6](https://on.soundcloud.com/ZWCJWFMDXHqncjZV6)



## 5.4. Diseño de las bases de datos distribuidas

Al diseñar una base de datos distribuida, es necesario tomar decisiones sobre aspectos relativos a:

- **Cómo se lleva a cabo la fragmentación** de la base de datos.
- **Qué fragmentos** de la base de datos **se replicarán**.
- **Cuál será la ubicación** de estos fragmentos y sus réplicas asociadas.

La replicación mejora la disponibilidad de datos, pero replicar todo en cada nodo puede reducir la eficiencia. Sin replicación, la disponibilidad se ve afectada. Es esencial encontrar un equilibrio y definirlo en el esquema de replicación, indicando qué objetos se replican, su ubicación y cómo se manejan las actualizaciones. En esta imagen se muestran los tres aspectos fundamentales del diseño.

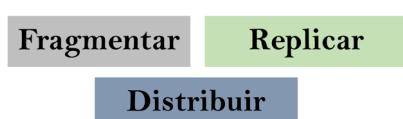


Fig.6. Tres aspectos básicos del diseño de una BBDD distribuida.

Seleccionar dónde almacenar un fragmento depende de factores como **rendimiento, disponibilidad y recursos**. La **replicación permite**:

- **Distribuir la carga de trabajo.**
- **Mejorar el rendimiento de las consultas.**
- **Aumentar la disponibilidad.**

Las bases de datos distribuidas incluyen todas las fases del diseño de una base de datos convencional.

## / 6. Arquitectura de una base de datos distribuida

La estructura del sistema de una base de datos distribuida está determinada por su arquitectura. Por lo tanto, **para definir la arquitectura, es esencial identificar los componentes del sistema, las funciones específicas de cada uno, cómo se interconectan entre sí, etc.**

### 6.1. Enfoques de definición de la arquitectura

Existen tres enfoques distintos para ello:

- **Basado en componentes:** Se definen los componentes del sistema y sus relaciones. Cada componente añade funcionalidades distintas al sistema.
- **Basado en funciones:** Se identifican los tipos de usuarios y la funcionalidad que el sistema ofrece a cada tipo.
- **Basado en datos:** Se clasifican los tipos de datos y se definen las unidades que utilizarán, según su clasificación.

### 6.2. Esquemas de la arquitectura de referencia

En la siguiente figura, se puede ver la arquitectura de referencia de una base de datos distribuida. En ella, se distinguen varias partes:

- **Esquema global:** Se definen todos los datos de la base igual que si fuese distribuida.
- **Esquema de fragmentación:** Define cómo se definen a nivel lógico los datos.
- **Esquema de asignación:** Indica cómo se distribuyen los fragmentos, es decir, dónde se guarda cada uno de ellos.

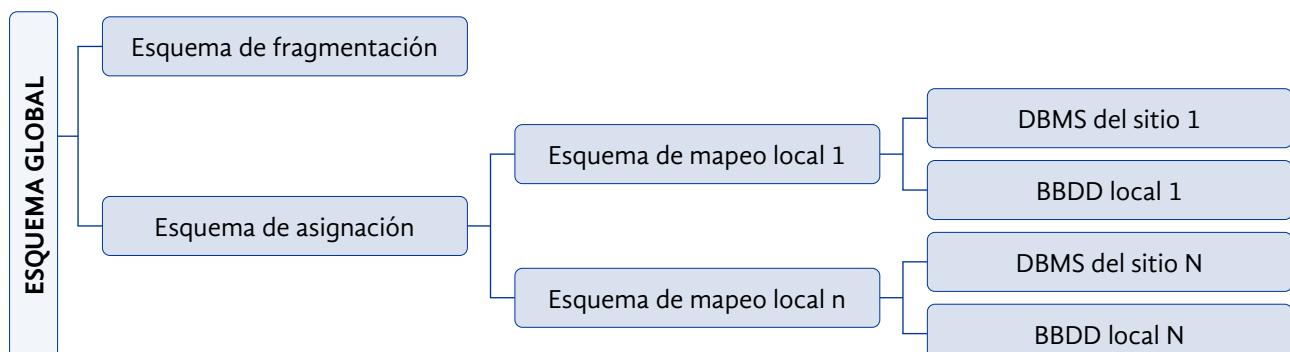


Fig.7. Arquitectura de referencia de una base de datos distribuida.

Se pueden identificar dos tipos de sistemas de bases de datos distribuidas:

- **Homogéneo:** Combina varios recursos de datos distribuidos en diferentes sitios que se comunican entre sí. No hay usuarios globales.
- **Heterogéneo:** Administra diversos sistemas de bases de datos en los nodos locales, que se conectan mediante software. Existen tanto usuarios globales como locales.

### Enlaces de interés...

En este enlace, encontrarás un vídeo donde puedes ampliar información sobre la arquitectura de las bases de datos distribuidas. Es un contendio muy útil para entender mejor su funcionamiento y sus ventajas:  
<https://www.youtube.com/watch?v=aHfvo1RGXTA>

## 6.3. Implementaciones múltiples y federadas

Las BDD tienen dos tipos de implementaciones:

- **Implementaciones múltiples:** Varias instancias completas o fragmentadas se distribuyen en nodos distintos. Cada nodo puede funcionar independientemente, pero todos colaboran en un sistema cohesivo. Algunas de sus características son:
  - » **Replicación de datos:** Los datos se duplican en varios nodos, mejorando la disponibilidad y la tolerancia a fallos.
  - » **Escalabilidad:** Fácil adición de nodos para manejar más datos y transacciones.
  - » **Optimización de consultas:** Procesamiento de consultas desde el nodo más cercano o menos cargado.
- **Implementaciones federadas:** Cada nodo tiene su propia base de datos autónoma y se conecta con otros nodos mediante una federación. Estos nodos pueden ser heterogéneos, lo que significa que pueden utilizar diferentes sistemas de gestión de bases de datos (DBMS) y estar ubicados en diversos entornos. Algunas de sus características son:
  - » **Autonomía local:** Cada nodo mantiene su propia base de datos y sus políticas de gestión.
  - » **Interoperabilidad:** Los nodos pueden ser heterogéneos, pero comunicarse y trabajar juntos mediante la federación.
  - » **Consulta global:** Las consultas pueden implicar a múltiples nodos para obtener una vista unificada.
  - » **Coordinación y consistencia:** Los mecanismos de federación coordinan los nodos y garantizan la consistencia de los datos.

## 6.4. Hardware y software de una BD distribuida

El HW y el SW de una BDD se componen de:

- **Nodos o servidores (HW):** Almacenan la información.
- **DDBMS (SW):** Sistema de administración de BDD.
- **DTM (SW):** Administrador de transacciones distribuidas.

En esta figura podemos ver los componentes hardware y software.

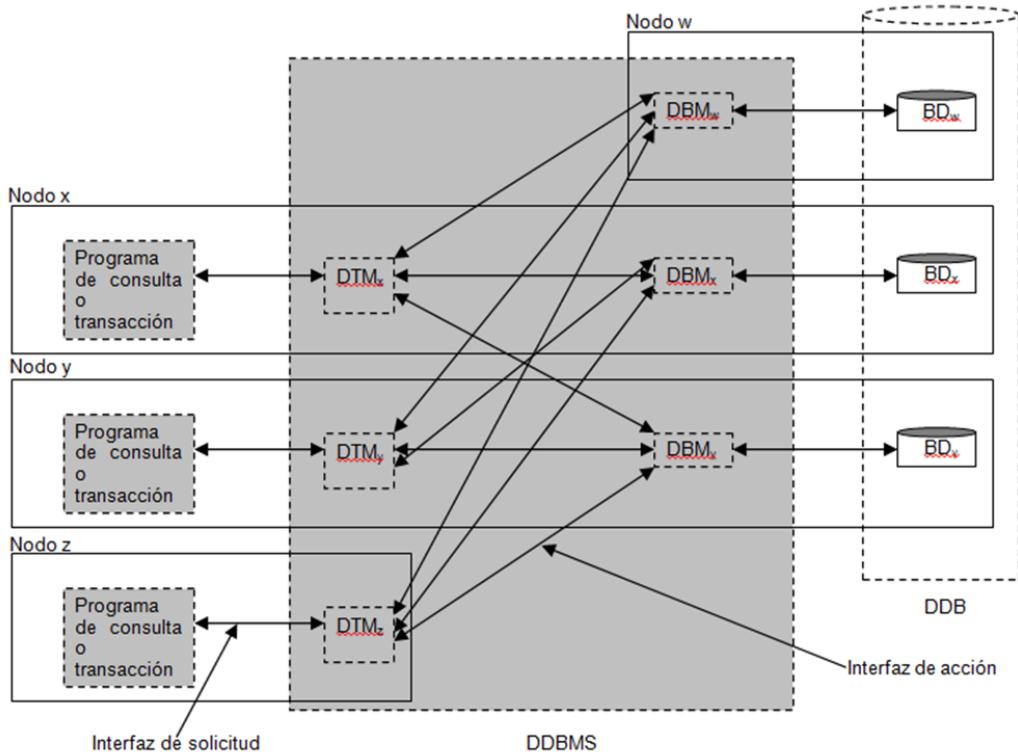


Fig.8. Hardware y software de una BDD.

En el siguiente audio, explicaremos cómo se comportan estos componentes hardware y software y cómo interactúan entre ellos:



*Audio 5. "Componentes hardware y software de las bases de datos distribuidas"*

<https://on.soundcloud.com/fSwcMj1DxclGSe7>



## / 7. Fragmentación de la información

Como se mencionó anteriormente, en las bases de datos distribuidas, la información está repartida entre varios sitios, lo que significa que suele estar fragmentada.

Por lo tanto, conviene dividir la información de manera adecuada para evitar problemas al recuperarla.

Es fundamental, en este contexto, considerar el **grado de fragmentación**, que se refiere a cuántas veces se dividen los datos. Un grado de fragmentación demasiado alto o demasiado bajo no estaría permitido, ya que ambos extremos harían inviable su gestión. El grado de fragmentación debe ser intermedio y dependerá del uso previsto de la base de datos, así como de la arquitectura y las aplicaciones utilizadas. Por todo esto, el grado de fragmentación es un concepto vital para la ejecución eficiente de consultas en la base de datos.

Podemos distinguir varios tipos de fragmentación de la información:

- **Fragmentación horizontal:** Se aplica directamente sobre los registros o filas de las tablas, dividiendo la información en fragmentos que incluyen varias de estas filas. Es más simple de realizar que la fragmentación vertical, ya que su ejecución es menos compleja.
- **Fragmentación vertical:** Se distingue por que cada fragmento resultante de dividir la información contiene una serie de atributos del origen, los cuales se repiten en todos ellos.

El objetivo de esta técnica es permitir que las aplicaciones de usuario se ejecuten en un solo fragmento, optimizando las consultas y consiguiendo un mejor tiempo de respuesta.

En esta tabla podemos ver un ejemplo de fragmentación vertical:

<b>Id empleado</b>	<b>Nombre</b>	<b>Salario</b>	<b>Localidad</b>	<b>Departamento</b>
132	María	2.500	Teruel	Dirección
245	Juan	2.400	Oviedo	I+D
334	Sofía	2.700	Pamplona	RR.HH.
<b>Id empleado</b>	<b>Nombre</b>	<b>Salario</b>		
132	María	2.500		
245	Juan	2.400		
334	Sofía	2.700		
<b>Id empleado</b>	<b>Localidad</b>	<b>Departamento</b>		
132	Teruel	Dirección		
245	Oviedo	I+D		
334	Pamplona	RR.HH.		

Tabla 3. Ejemplo de fragmentación vertical.

- **Fragmentación híbrida o mixta:** Combina las dos técnicas anteriores. Puede ser HV (se aplica primero horizontal sobre los datos y luego vertical sobre los fragmentos) o VH (se aplica primero vertical sobre los datos y luego horizontal sobre los fragmentos) o también puede aplicarse de forma directa.

### Enlaces de interés...

En el siguiente enlace, encontrarás un vídeo donde puedes ampliar información sobre la fragmentación. Se muestran los principios sobre los que se crean los diferentes tipos (horizontal, vertical y mixta):  
[https://www.youtube.com/watch?v=QRFS2OnN\\_dc](https://www.youtube.com/watch?v=QRFS2OnN_dc)

## / 8. Distribución de la información

Las bases de datos distribuidas pueden configurarse de tres formas:

- **Nodo réplica:** La información se duplica en cada nodo.
- **Nodo particionado:** La información se distribuye entre los nodos sin duplicación.
- **Nodo híbrido:** Combina las dos anteriores y es la más usada.

## 8.1. Algoritmos de distribución

Existen múltiples algoritmos de distribución que **permiten determinar la ubicación de los datos**, aunque son necesariamente dependientes de la metodología de fragmentación elegida. Algunos de ellos son los que podemos ver en la siguiente imagen.

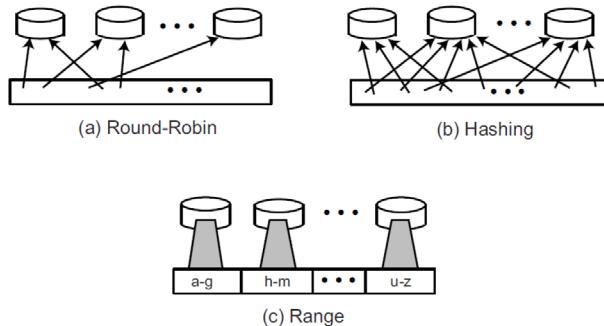


Fig.9. Algoritmos de distribución.

Estos algoritmos deben valorar los requisitos y características del sistema para poder determinar qué distribución optimiza el rendimiento y funcionamiento de la base de datos distribuida.

Los **aspectos más relevantes** a tener en cuenta son:

- El rendimiento que debe tener el sistema.
- La consistencia y disponibilidad de datos del sistema.
- La cantidad y longitud de los registros, además de la complejidad de las relaciones entre las distintas entidades.
- El tipo de transacciones a realizar y los atributos principales a usar.

## 8.2. Objetivos de la distribución de datos

Durante la etapa de diseño de la distribución de datos, es necesario no perder de vista los objetivos explicados en la siguiente tabla:

OBJETIVO	DESCRIPCIÓN
Localidad	Acercar la información a las aplicaciones que la consumirán para reducir latencias.
Balanceo de carga	Distribuir la carga de trabajo proporcionalmente a las capacidades de cada nodo, para no saturar ninguno de ellos mientras otros apenas trabajan.
Escalabilidad	Facilitar el crecimiento del sistema añadiendo más nodos y recursos de manera sencilla.
Tolerancia a fallos	Al estar los datos distribuidos, en caso de fallo de un nodo, sigue habiendo más nodos disponibles.

Tabla 4. Objetivos de la distribución de datos.

## / 9. Seguridad y recuperación en las bases de datos distribuidas

La seguridad, el control de la concurrencia y la recuperación de la información son **cruciales para garantizar el acceso seguro y la continuidad del servicio en caso de fallos**. En la siguiente tabla se desarrollan estos conceptos:

OBJETIVO	DESCRIPCIÓN
<b>Control de concurrencia y recuperación</b>	<ul style="list-style-type: none"> <li><b>Consistencia de copias de datos:</b> Deben sincronizarse lo más inmediatamente posible para seguir operando si falla un nodo.</li> <li><b>Gestión de fallos en nodos y sus enlaces de comunicación:</b> Actualización de los datos al restablecerse un nodo tras una caída.</li> <li><b>Confirmación (commit) distribuida en dos fases:</b> El nodo coordinador pregunta a los nodos participantes si están listos para confirmar la transacción. Si es así, ordena confirmar, si no, ordena deshacer (rollback).</li> <li><b>Interbloqueo distribuido:</b> Previene bloqueos mutuos entre procesos reservando todos los recursos necesarios antes de iniciar la transacción.</li> </ul>
<b>Medidas de seguridad</b>	<ul style="list-style-type: none"> <li><b>Cifrado de datos:</b> Impide que los datos sean legibles si no se cuenta con las claves necesarias.</li> <li><b>Seguridad en la red:</b> Firewall, VPN y VPC que protegen la comunicación entre nodos.</li> <li><b>Auditorías constantes:</b> Verifican los accesos y actividades realizados sobre las bases de datos.</li> </ul>

Tabla 5. Control de concurrencia, seguridad y recuperación en las BBDD.

Por otro lado, la recuperación de información es fundamental para garantizar la continuidad del servicio en caso de fallos o desastres.

Algunas de las **estrategias de recuperación** para BBDD son:

- Replicación geográfica:** Se replican los datos en varios lugares para asegurar la disponibilidad y la tolerancia a fallos.
- Copias de seguridad incrementales:** Son *backups* que solo guardan los cambios desde la última copia de seguridad completa, con lo que se ahorra tiempo y espacio.
- Comutación por error:** En caso de que un nodo falle, otro nodo asume sus funciones sin interrupción relevante en el servicio. Esto mantiene la continuidad del mismo.

Además, las bases de datos distribuidas se enfrentan a diversos **desafíos** relacionados con la **seguridad**. Algunos de ellos son:

- **Latencia de red:** Compromete el rendimiento y la seguridad al afectar a la consistencia de los datos, la sincronización de las transacciones y la disponibilidad del sistema. Estas vulnerabilidades pueden ser explotadas por atacantes. Usar caché y optimizar el enrutamiento de red puede ser una solución.
- **Gestión de transacciones:** Implica coordinar las transacciones distribuidas para garantizar que todas las operaciones se completen con éxito o que ninguna se lleve a cabo (atomicidad).

## / 10. Caso práctico 1:“Base de datos para tienda y almacén”

**Planteamiento:** Continuando con el enfoque del tema 1, vamos a asistir a nuestro amigo en la creación de una base de datos simple que facilite la gestión de su tienda. Antes de implementarla, es necesario diseñarla.

**Nudo:** Como diseñadores de la base de datos, el primer paso será definir su arquitectura. Para ello, debemos decidir qué entidades, tablas, registros y campos incluiremos en este diseño inicial. Debemos recordar que el objetivo principal es desarrollar un sistema de base de datos que ayude en la gestión del almacén, así como en las operaciones de compras y ventas.

**Desenlace:** A continuación, se muestra una propuesta de diseño (que posteriormente podría ampliarse o potenciarse para cubrir otras necesidades).

- Entidades: Almacén, cliente y proveedor.
- Tablas:
  - » Almacén: *Stock* y movimientos.
  - » Clientes: Ventas y clientes.
  - » Proveedor: Compras y proveedores.
- Campos:
  - » *Stock*: ID producto, concepto, unidades, proveedor.
  - » Movimientos: ID producto, concepto, unidades.
  - » Ventas: ID producto, concepto, unidades vendidas, precio, fecha venta, cliente.
  - » Clientes: ID cliente, nombre, apellidos, dirección, teléfono.
  - » Compras: ID producto, concepto, unidades compradas, precio, fecha compra, proveedor.
  - » Proveedores: ID proveedor, nombre, apellidos, dirección, teléfono.

A continuación, podrás ver un diseño de base de datos para la tienda de informática.

STOCK	MOVIMIENTOS	VENTAS
<b>ID producto</b> <b>Concepto</b> <b>Unidades</b> <b>Proveedor</b>	<b>ID producto</b> <b>Concepto</b> <b>Unidades</b>	<b>ID producto</b> <b>Concepto</b> <b>Unidades</b> <b>Precio</b> <b>Fecha</b> <b>Cliente</b>
CLIENTES	COMPRAS	PROVEEDORES
<b>ID cliente</b> <b>Nombre</b> <b>Apellidos</b> <b>Dirección</b> <b>Teléfono</b>	<b>ID producto</b> <b>Concepto</b> <b>Unidades</b> <b>Precio</b> <b>Fecha</b> <b>Proveedor</b>	<b>ID proveedor</b> <b>Nombre</b> <b>Apellidos</b> <b>Dirección</b> <b>Teléfono</b>

Tabla 6. Diseño de base de datos para la tienda de informática.

## / 11. Caso práctico 2. “¿Qué modelo usar?”

**Planteamiento:** Como seguimos ayudando a nuestro amigo con su tienda, después de considerar cómo crear la base de datos en el caso práctico 1, aún tenemos dudas sobre qué modelo de datos sería más adecuado utilizar.

**Nudo:** Tras analizar las características principales de todos los modelos de datos disponibles, debemos seleccionar el que mejor se adapte a nuestro caso específico y explicar las razones que justifiquen nuestra elección.

**Desenlace:** La elección del modelo de datos depende principalmente de las capacidades que aporte la base de datos (lo que se pueda llegar a hacer con ella) y de las funcionalidades que los usuarios requieran. Como nuestro amigo necesita organizar su negocio, parece más lógico empezar con un modelo relacional tradicional, ya que es sencillo y rápido de implementar y construir. A medida que surjan necesidades adicionales, este modelo podría evolucionar a un modelo objeto-relacional para ofrecer más funcionalidades.

Además, dado que dispone únicamente de una tienda, parece más sencillo optar por una base de datos centralizada. Con toda esta información, podemos llegar a las conclusiones que se muestran en la siguiente tabla:

EJEMPLO	DE	TABLA
<b>Jerárquica</b>		Obsoleta. Demasiado rígido.
<b>En red</b>		En desuso.
<b>Relacional</b>		Amplio soporte y adecuado para el objetivo perseguido.
<b>Orientada a objetos</b>		Parece compleja para lo que se quiere hacer, aunque sería viable.
<b>Objeto-relacional</b>		Possible para futuras funcionalidades más avanzadas.
<b>Multidimensional</b>		No tiene sentido aplicarla en este caso por complejidad.
<b>Deductiva</b>		No tiene sentido en este caso.
<b>NoSQL</b>		Este tipo de bases de datos no cumplen el requisito de tener una estructura clara.

Tabla 7. Toma de decisiones del modelo de datos adecuado.

## / 12. Resumen y resolución del caso práctico de la unidad

En este tema hemos visto conceptos básicos de las bases de datos, así como sus elementos, usos y tipos según el modelo de datos.

También hemos analizado las diferencias entre BBDD centralizadas y distribuidas, desarrollando conceptos clave como diseño, arquitectura y seguridad de estas últimas.

A continuación podrás ver el esquema resumen del tema.

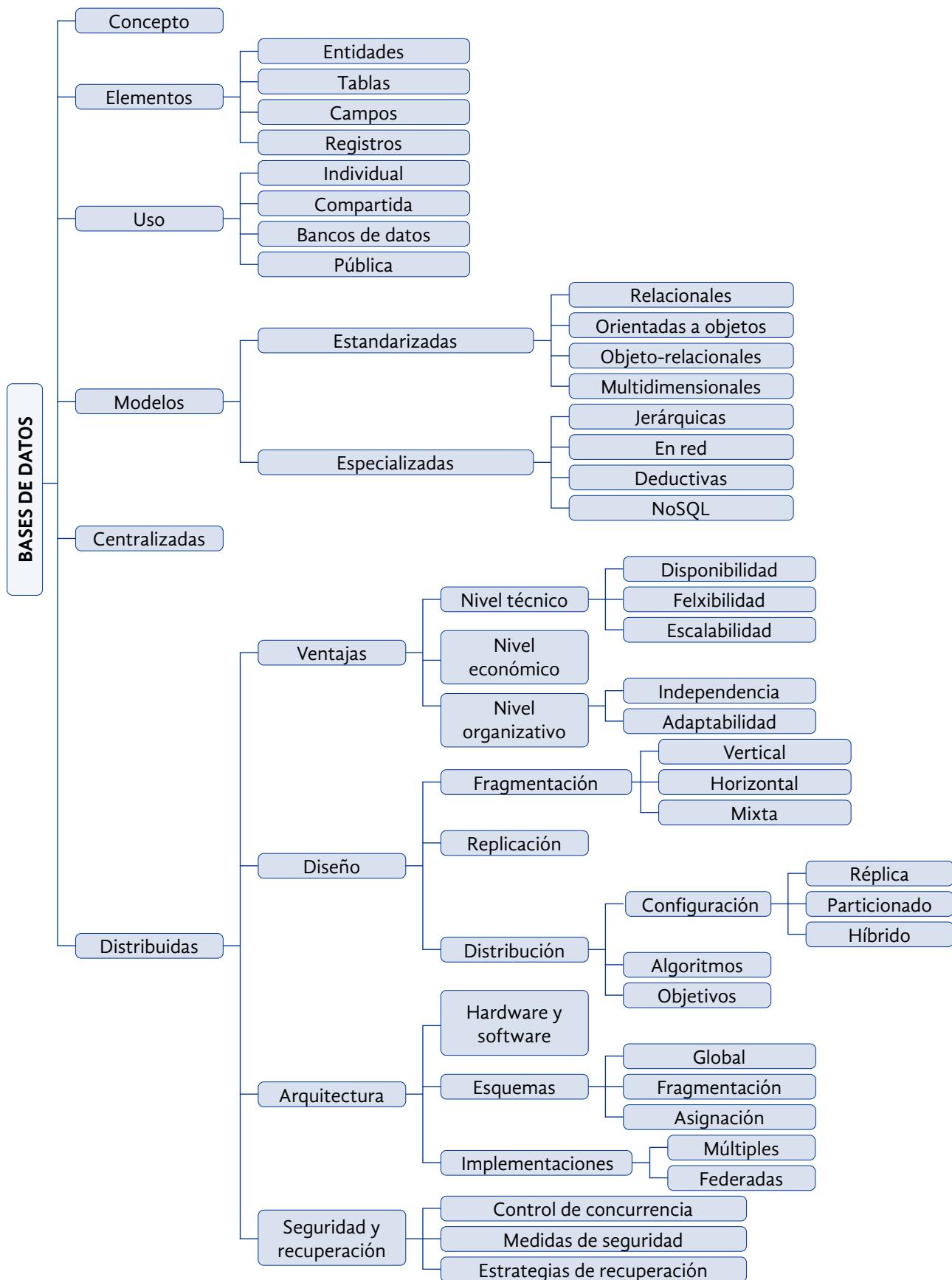


Fig.10. Esquema resumen del tema.

## Resolución del caso práctico de la unidad

En el caso práctico inicial, se planteaban cuestiones sobre el proyecto de base de datos de la tienda de informática de nuestro amigo. Algunas posibles respuestas a esas preguntas serían:

- **¿Necesitaremos más información de la que ya tenemos? Si es así, ¿cuál concretamente?** En cualquier diseño, especialmente de base de datos, es fundamental partir de la información adecuada, para lo que es importante tratar con varios perfiles de usuarios que tenga la aplicación, con el fin de conocer las necesidades de cada uno. Es muy importante definir cuál será el medio de alojamiento, qué tipo de modelo de datos se aplicará, qué datos que necesitaremos para construir la base de datos (ver caso práctico 1), etc.
- **¿Cómo vamos a construir la base de datos?** A esta pregunta le hemos dado respuesta en los casos prácticos 1 y 2, en los que hemos definido, en gran medida, la arquitectura que tendrá nuestra base de datos y la información que vamos a recoger en la misma. En pasos anteriores, hemos definido el medio físico y cómo se accederá a la información.
- **¿Vamos a aplicar algún modelo en concreto?** Dadas las características y necesidades que queremos cubrir, parece lógico plantear la base de datos como una base de datos relacional.
- **¿Optaremos por una base de datos centralizada o distribuida?** Como nuestro amigo tiene una única tienda, una base de datos centralizada será suficiente para empezar.

## / 13. Bibliografía

Sánchez, G. C. (2001). *Sistemas gestores de bases de datos*. Paraninfo.

Elmasri, R. y Navathe, S. (2007). *Fundamentos de sistemas de bases de datos*. Addison Wesley.

López, I., Castellano, M. J. y Ospino, J. (2011). *Bases de datos*. Garceta.

Oppel, A. (2009). *Databases A Beginner's Guide*. McGraw-Hill Education.

Querydatabase (2012, 3 de febrero). *Bases de datos distribuidas*:

<https://querydatabase.wordpress.com/2012/01/31/video-bases-de-datos-distribuidas/>

Rodríguez, N. (2022, 27 de abril). Blockchain vs Base de datos: Entendiendo las diferencias entre las dos. *101 Blockchains*.

<https://101blockchains.com/es/blockchain-vs-base-de-datos/>