

Modelo orientado a objetos

Este modelo agrupa la información en paquetes relacionados entre sí, los datos de cada uno de los registros se combinan en un solo objeto con todos sus atributos. Gracias a esto, toda la información estará disponible en el objeto ya que los datos quedan agrupados más no distribuidos en diferentes tablas. En los objetos no solo se guardan los atributos sino también los métodos. Los objetos son unidades concretas de una clase abstracta lo que crea una jerarquía de clases y subclases, las subclases adoptan las propiedades de las clases superordinarias y las complementan con sus propios atributos.

El modelo de base de datos con el que se va a trabajar depende mucho del uso que queramos darle, este tipo de modelo es adecuado si se trabaja con lenguajes de programación orientados a objetos. Aquí presento algunas de sus ventajas y desventajas:

Ventajas	Desventajas
Los conjuntos de datos se guardan más fácil.	El uso de las bases de datos orientadas a objetos no está muy extendido.
Los códigos de identificación se asignan automáticamente a cada objeto.	En algunas situaciones, la gran complejidad puede tener problemas de rendimiento.
Funciona bien con lenguajes de POO.	La comunidad que se dedica a mejorar este modelo es pequeña.

Modelos NoSQL

Las bases de datos NoSQL son un modelo de bases de datos caracterizada al almacenamiento de información en colecciones en lugar de tablas. A diferencia de las bases de datos relacionales las bases de datos NoSQL almacenan la información en otra estructura mucho más versátil y ligera llamada colección. Las colecciones permiten almacenar documentos generando una estructura heterogénea, por lo que se puede decir que es muy flexible.

Clave-Valor

Son las más sencillas de todas y almacenan datos en una organización de pares de clave y valor.

Sus casos de uso:

- Almacenamiento en caché: Destaca en situaciones de almacenamiento en caché en las que el acceso rápido a los datos es crucial para el rendimiento.

- Perfiles de usuario: Gestiona información de los usuarios como nombre de usuario, correo electrónico y sus preferencias.
- Almacenamiento de la sesión: Son eficaces para gestionar los datos de sesión.
- Análisis en tiempo real: La alta velocidad de acceso a los datos hace que las bases de datos clave-valor sean adecuadas para escenarios analíticos.
- Catálogo de productos: Ofrecen una forma sencilla y eficaz de gestionar catálogos de productos, sobre todo en situaciones que los productos tienen una variedad de atributos diferentes.

Ventajas	Desventajas
Simplicidad Modelo de datos sencillo que reduce la complejidad.	Capacidad de consulta limitada Al carecer de capacidad de consultas avanzadas las hacen menos adecuadas para escenarios de consulta y análisis complejos.
Alto rendimiento Está optimizada para operaciones rápidas de lectura y escritura.	Problemas de integridad de los datos Muchas no aplican el mismo nivel de coherencia e integridad referencial que las bases de datos relacionales tradicionales.
Escalabilidad Permiten la adición de nodos para gestionar un mayor volumen de datos y tráfico.	Curva de aprendizaje del paradigma NoSQL Adaptarse a NoSQL puede suponer una curva de aprendizaje intimidatoria para quienes están acostumbrados a SQL.
Flexibilidad Un diseño sin esquema admite diversas estructuras de datos en una única base de datos.	

Documentales

Almacenan datos en colecciones, siendo los elementos de cada una de esas colecciones documentos que contienen pares clave/valor.

Sus casos de uso:

- Una base de datos sobre libros: Representa una relación entre libros y autores en diferentes tablas con sus ID's correspondientes.
- Gestión de contenido: Los archivos son almacenados como un solo documento y la base de datos es más fácil de mantener a medida que el servicio evoluciona con el tiempo.

Ventajas	Desventajas
Sin esquema No existen restricciones en el formato y estructura del almacenamiento de datos.	Limitaciones en la comprobación de coherencia Los listados pueden presentar inconsistencias a la hora de hacer búsquedas.
Creación y mantenimiento más rápidos Se requiere un mantenimiento mínimo una vez que crea el documento.	Debilidades en la atomicidad Las operaciones que afectan a múltiples colecciones no son atómicas.
Sin claves externas Los documentos pueden ser independientes entre sí.	Seguridad Los dueños de este tipo de modelo, debe prestar mucha atención con la vulnerabilidad de la web app.
Formatos abiertos Proceso de compilación limpio.	
Control de versiones integrado A medida que sus documentos crecen en tamaño, también pueden crecer en complejidad. Este control de versiones, disminuye los conflictos generados.	

Grafos

Está compuesto por documentos que se relacionan entre sí y que permiten representar relaciones complejas entre los datos.

Sus casos de uso:

- Las empresas pueden aprovechar la posibilidad de obtener información acerca de las búsquedas del historial, de clics y de otros comportamientos del usuario y crear perfiles completos de los usuarios.
- En gestión de cadenas de suministros se pueden monitorizar procesos, desde diseño hasta las ventas.
- Análisis de riesgos, detección de fraudes y búsqueda de errores.
-

Ventajas	Desventajas
Velocidad Esta velocidad depende únicamente del número de relaciones concretas, no del conjunto de datos.	Consultas Las consultas que no están pensadas para investigar relaciones entre nodos pueden ser menos eficientes.
Resultados Los resultados son en tiempo real.	Distribución La distribución puede ser complicada y requieren técnicas de partición y replicación.

Presentación La presentación intuitiva y resumida de las relaciones.	Complejidad A medida que crece la complejidad entre datos y sus relaciones, la gestión y optimización puede volverse más compleja.
Estructura Las estructuras son flexibles y ágiles.	

Columnas

Almacena tablas de datos por columna en lugar de por fila.

Sus casos de uso:

- Los tamaños de los conjuntos de datos son grandes, por lo que la compresión proporciona importantes ahorros en almacenamiento.
- Las consultas tienden a realizar agregaciones en grandes cantidades de registros.
- Baja frecuencia de actualización en comparación con las operaciones de lectura.

Ventajas	Desventajas
Compresión Las altas compresiones 10x o más reducen costos de almacenamiento.	Actualizaciones Modificar y eliminar datos es más lento y complejo.
Rendimiento de consulta Acceder sólo a las columnas necesarias hace que las consultas sean rápidas.	Transacciones La falta de bloqueo a nivel de fila dificulta las transacciones.
Vectorización Las operaciones individuales en vectores de columna el paralelismo de la CPU.	Gastos generales Se agregan complejidades para inserciones, eliminaciones y mecanismos de bloqueo.
Flexibilidad Se pueden agregar nuevos índices analíticos sin afectar los datos.	No apto para OLTP (Proceso de transacciones en línea) Gastos generales e incapacidad para realizar consultas puntuales rápidas.
Escalabilidad Escalar utilizando almacenamiento distribuido.	
Disponibilidad Los motores de replicación y consulta distribuida proporcionan alta disponibilidad.	
Analítica	

Se adapta perfectamente a cargas de trabajo de análisis con agregaciones grandes de conjuntos de datos.	
---	--

Referencias

[1] “Base de datos orientada a objetos: el secreto mejor guardado de los modelos de bases de datos”. IONOS Digital Guide. Accedido el 15 de agosto de 2025. [En línea]. Disponible: <https://www.ionos.mx/digitalguide/hosting/cuestiones-tecnicas/base-de-datos-orientada-a-objetos/>

[2] “Bases de datos NoSQL: qué son, tipos y ventajas | Blog de Arsys”. Arsys. Accedido el 15 de agosto de 2025. [En línea]. Disponible: <https://www.arsys.es/blog/bases-de-datos-nosql-que-son-tipos-y-ventajas>

[3] “Columnar Databases (Use Cases, Examples, Advantages & Disadvantages)”. DatabaseTown. Accedido el 15 de agosto de 2025. [En línea]. Disponible: <https://databasetown.com/columnar-databases/>

[4] “Base de datos clave-valor | Conceptos | Couchbase”. Couchbase. Accedido el 15 de agosto de 2025. [En línea]. Disponible: <https://www.couchbase.com/es/resources/concepts/key-value-database/>

[5] “Document Database {Definition, Features, Use Cases}”. Knowledge Base by phoenixNAP. Accedido el 15 de agosto de 2025. [En línea]. Disponible: <https://phoenixnap.com/kb/document-database>

[6] “Graph database”. IONOS Digital Guide. Accedido el 15 de agosto de 2025. [En línea]. Disponible: <https://www.ionos.mx/digitalguide/hosting/cuestiones-tecnicas/graph-database/>

[7] “Base de datos orientada a grafos: ventajas y desventajas | Tokio School”. Tokio School. Accedido el 15 de agosto de 2025. [En línea]. Disponible: <https://www.tokioschool.com/noticias/ventajas-desventajas-base-datos-grafos>