

1 Modelo Orientado a Objetos

En la actualidad no existe una definición estandarizada del modelo, sin embargo, de manera muy general un OODBMS debe cumplir con 2 características[1]:

- Ser un DBMS: Debe incluir características de cualquier DBMS, como persistencia, administración de almacenamiento secundario, concurrencia, recuperación y facilidad de consultas personalizadas.
- Ser consistente con los lenguajes de programación orientados a objetos: Se trata de objetos complejos, identidad de objetos, encapsulación, herencia, sobreescritura y sobrecarga y completa capacidad computacional .

Ventajas[1]: No debería ser necesario el uso de más de un lenguaje de programación a diferencia de un modelo relacional donde se utiliza uno para la aplicación y otro para la base de datos. La mayoría de los proyectos de desarrollo son más cortos y requieren menos personas.

Desventajas[1]: La ingeniería de software orientada a objetos requiere mucho tiempo de análisis. Carecen de un modelo de datos común y de fundamentos formales, además de que su comportamiento en seguridad y manejo de transacciones no están a la altura de los programas actuales de administradores de bases de datos.

Casos de uso:

- Aplicaciones científicas: Es utilizado en investigaciones científicas debido al gran volumen de datos y la creación de modelos complejos
- Sistemas de Información Geográfica (SIG): Es perfecto para manejar datos espaciales complejos, como mapas, coordenadas, y relaciones espaciales entre objetos geográficos.
- Aplicaciones multimedia: Gracias a su capacidad para manejar datos complejos, se utiliza en sistemas que requieren manipular grandes cantidades de multimedia.

2 Modelo Clave-Valor

Almacena datos como un conjunto de pares clave-valor en los que una clave sirve como un identificador único.

Ventajas[2]: Son modelos flexibles, permiten almacenar toda la información en una sola tabla. Soportan tipos de datos complejos, como matrices diccionarios anidados y multimedia.

Desventajas[2]: No admiten consultas complejas. Existen limitaciones en cuanto a la cantidad de datos que se pueden filtrar y clasificar antes de acceder a ellos. Cualquier persona puede modificar el esquema en el programa de base de datos.

Casos de uso[2]:

- Administración de sesiones: Cada sesión de usuario tiene un identificador único. Solo la clave principal consulta los datos de la sesión, lo que encaja muy bien con el modelo.
- Carro de compras: Pueden manejar el escalado de grandes cantidades de datos y volúmenes extremadamente altos de cambios de estado, mientras se atiende a millones de usuarios simultáneamente a través del procesamiento y el almacenamiento distribuidos.
- Almacenamiento en caché: Se puede usar una base de datos de clave-valor para almacenar datos temporalmente y recuperarlos más rápidamente.

3 Modelo Documental

Se trata de una base de datos que almacena información en documentos. Los documentos almacenan datos en pares de campo-valor. Los valores pueden ser de varios tipos y estructuras, como cadenas, números, fechas, matrices u objetos. Los documentos se pueden almacenar en formatos como JSON, BSON, y XML[3].

Ventajas[3]: Es un modelo intuitivo, lo que lo hace fácil y rápido para los desarrolladores. Se trata de un modelo flexible que es capaz de escalar horizontalmente. Las bases de datos de documentos tienen una API o lenguaje de consulta.

Desventajas[3]: Muchas bases de datos documentales no admiten transacciones ACID de varios documentos.

Casos de uso[3]:

- Analítica operativa
- Procesamiento de pagos
- Gestión y personalización de datos de clientes

4 Modelo Orientado a Grafos

Los grafos y las bases de datos orientadas a grafos proporcionan modelos de grafos para representar las relaciones en los datos. Gracias a estas, los usuarios pueden hacer consultas transversales basadas en conexiones y aplicar algoritmos de grafos a fin de encontrar patrones, rutas, comunidades, factores influyentes, puntos únicos de fallo y otras relaciones[4].

Ventajas[4]: Permite un análisis eficiente a gran escala con cantidades masivas de datos. Hace posible analizar relaciones y comportamientos entre datos en gráficas, haciendo posible entender cosas difíciles de ver con otros métodos

Casos de uso[4]:

- Los grafos le permiten explorar y descubrir conexiones y patrones en redes sociales, IdC, big data, almacenes de datos y también datos de transacciones complejas para múltiples casos de uso comercial.
- Blanqueo de dinero: Permiten ubicar criminales a partir de transacciones entre entidades.

5 Modelo Columnar

Las bases de datos columnares se orientan en columnas, por cada entrada hay una columna. Los datos de entrada están dispuestos uno debajo del otro[5].

Ventajas[5]: Son eficientes para operaciones de evaluación de grandes volúmenes de datos. Son capaces de comprimir datos.

Desventajas[5]: En aplicaciones transaccionales son más lentas que un modelo de filas.

Casos de uso:

- Se utilizan cuando es necesario analizar grandes cantidades de datos, como en investigaciones.

6 Referencias

- [1] D. Kanaguisco, “Modelo Orientado a Objetos,” Unam.mx, 2017. https://repositorio-uapa.cuaed.unam.mx/repositorio/moodle/pluginfile.php/2731/mod_resource/content/1/UAPA-Modelo-Orientado-Objetos/index.html
- [2] “¿Qué es una base de datos clave-valor? - Explicación de las bases de datos y los pares clave-valor - AWS,” Amazon Web Services, Inc., 2024. <https://aws.amazon.com/es/nosql/key-value/#topic-1> (accessed Aug. 17, 2025).
- [3] “Document Database - NoSQL,” MongoDB, 2025. <https://www.mongodb.com/es/resources/basics/databases/document-databases>
- [4] “What is a Graph Database?,” Oracle.com, Oct. 30, 2019. <https://www.oracle.com/mx/autonomous-database/what-is-graph-database/#graph-database-advantages> (accessed Aug. 18, 2025).
- [5] “Base de datos columnar,” IONOS Digitalguide. <https://www.ionos.mx/digitalguide/hosting/cuestiones-tecnicas/base-de-datos-columnar/>