



# 1. MODELO ORIENTADO A OBJETOS

## 1.1 DEFINICIÓN DE MODELO ORIENTADO A OBJETOS

La orientación a objetos representa el mundo real y resuelve problemas a través de objetos, ya sean tangibles o digitales. Este paradigma tecnológico considera un sistema como una entidad dinámica formada de componentes. Un sistema sólo se define por sus componentes y la manera en que éstos interactúan. [1]

Una base de datos orientada a objetos es un sistema de bases de datos que puede trabajar con datos complejos, es decir, objetos que reflejan aquellos objetos que se utilizan en el lenguaje de programación orientado a objetos. [2]

## 1.2 ELEMENTOS DE MODELO ORIENTADO A OBJETOS [2]

### **Objeto**

Funcionan a modo de entidades dentro de la base de datos orientada a objetos. Pueden ser personas o cosas del mundo real que ejecutan determinadas tareas. El objeto es persistente y generalmente no se modifica. Un ejemplo de objeto puede ser la entidad «Libro 1».

### **Clase**

Es la agrupación de todos los objetos que poseen características y comportamientos similares en las bases de datos orientadas a objetos.

### **Atributos y métodos**

Los atributos son propiedades del objeto. Los métodos son la forma en la que se comportan las propiedades de dicho objeto en las bases de datos orientadas a objetos.

### **Patrones**

Los patrones permiten acceder a los objetos complejos de un modelo de base de datos orientado a objetos. Implican algunos principios que se verán a continuación.

## 1.3 CARACTERÍSTICAS DE MODELO ORIENTADO A OBJETOS [2]

### **Polimorfismo**

Es la capacidad que tiene un objeto para adoptar múltiples formas. En una BDOO esta característica hace que el mismo código de programa funcione con diferentes tipos de datos.



## Herencia

Dentro de modelo de base de datos orientado a objeto, crea relaciones jerárquicas entre distintas clases que permiten que muchas partes del código puedan reciclarse.

## Encapsulación

Es la capacidad de agrupar diferentes datos y objetos en una sola clase u objeto mayor, para poder brindar seguridad al sistema realizado.

## Abstracción

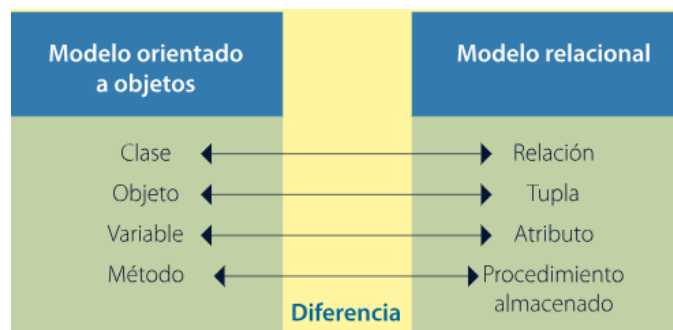
La encapsulación del modelo de base de datos orientado a objetos hace que el proceso de abstracción en una base de datos orientada a objetos sea posible, ya que elimina toda la información no esencial y permite representar solo las características de datos relevantes para la funcionalidad necesaria.

La abstracción permite simplificar los datos modelados y a su vez posibilita la reutilización de esos datos de los elementos orientado a objetos.

### 1.4 UTILIDAD [1]

Los manejadores de bases de datos orientados a objetos deben tomar en cuenta las siguientes operaciones:

- Ser capaces de definir sus propios tipos de datos.
- El tamaño de los datos puede ser muy grande.
- La duración de las transacciones puede ser muy larga.
- Recuperar rápidamente objetos complejos.
- Lenguajes de consulta de objetos, un ejemplo es OQL (Object Query Language).
- Mecanismos de seguridad basados en la noción de objeto.
- Funciones para definir reglas deductivas.





## BIBLIOGRAFÍA

- [1] E. D. K. Hernández, “Modelo Orientado a Objetos”, Unam.mx. [En línea].  
Disponible en:  
[https://programas.cuaed.unam.mx/repositorio/moodle/pluginfile.php/782/mod\\_resource/content/8/contenido/index.html](https://programas.cuaed.unam.mx/repositorio/moodle/pluginfile.php/782/mod_resource/content/8/contenido/index.html). [Consultado: 02-feb-2024].
- [2] “¿Qué es una base de datos orientada a objetos?”, *KeepCoding Bootcamps*, 29-sep-2022. .



## 2. MODELOS NoSQL (CLAVE- VALOR, DOCUMENTALES, GRAFOS)

### 2.1 CONCEPTOS

**Base de datos documental:** Una base de datos documental es una de las principales variantes de las bases de datos no relacionales o NoSQL. Se caracterizan por almacenar la información en registros, cada uno de los cuáles funciona como una unidad autónoma de información.

Como su propio nombre indica, las bases de datos orientadas a documentos utilizan documentos para el almacenamiento de todos los registros y los datos asociados a ellos. Cada uno de estos registros puede almacenar distintos tipos de datos. A su vez, los documentos que contienen los registros pueden tener diferentes formatos, desde archivos JSON o XML hasta documentos de texto. [1]

**Base de datos columnares:** Una base de datos en columnas es un tipo especializado de sistema de administración de bases de datos (DBMS) diseñado para manejar de manera eficiente cargas de trabajo de análisis de lectura intensiva, particularmente en el contexto de big data y almacenamiento de datos. Esta arquitectura de base de datos difiere de los sistemas tradicionales de bases de datos relacionales basadas en filas (RDBMS) al almacenar datos en forma de columna. [2]

**Base de datos clave/valor:** Esta modalidad de base de datos, key-value database o store en inglés, se basa en una tabla de tan solo dos columnas. En una de ellas se guarda un valor y en la otra, una clave que representa una característica identificativa única. Un valor puede ser sencillo, como una cadena de caracteres o un número entero, o pueden ser objetos complejos (un documento también puede ocupar el lugar de un valor, aunque, entonces se hablaría de una base de datos de documentos). En las bases de datos se pueden incluir también referencias a archivos, así como a tuplas (conjunto de valores). [3]

**Base de datos grafos:** Una base de datos orientada a grafos es una plataforma especializada y de un solo propósito para crear y manipular grafos. Los grafos contienen nodos, bordes y propiedades que se utilizan para representar y almacenar datos de una forma que no permiten las bases de datos relacionales. [4]

### 2.2 CARACTERISTICAS DE LAS BASES DE DATOS NO SQL



- Libre de esquemas – no se diseñan las tablas y relaciones por adelantado, además de permitir la migración del esquema.
- Proporcionan replicación a través de escalado horizontal.
- Este escalado horizontal se traduce en arquitectura distribuida
- Generalmente ofrecen consistencia débil
- Hacen uso de estructuras de datos sencillas, normalmente pares clave/valor a bajo nivel
- Suelen tener un sistema de consultas propio (o SQL-like)
- Siguen el modelo BASE (Basic Availability, Soft state, Eventual consistency) en lugar de ACID (Atomicity, Consistency, Isolation, Durability) [5]

El modelo BASE consiste en:

- Basic Availability – el sistema garantiza disponibilidad, en términos del teorema CAP.
- Soft state – el estado del sistema puede cambiar a lo largo del tiempo, incluso sin entrada. Esto es provocado por el modelo de consistencia eventual.
- Eventual consistency – el sistema alcanzará un estado consistente con el tiempo, siempre y cuando no reciba entrada durante ese tiempo. [5]

## 2.3 VENTAJAS

- A diferencia de las bases de datos relacionales, las bases de datos NoSQL están basadas en key-value pairs
- Algunos tipos de almacén de bases de datos NoSQL incluyen diferentes tipos de almacenes como por ejemplo el almacén de columnas, de documentos, de key value store, de gráficos, de objetos, de XML y otros modos de almacén de datos.
- Algunos tipos de almacén de bases de datos NoSQL incluyen almacenes de columnas, de documentos, de valores de claves, de gráficos, de objetos, de XML y otros modos de almacén de datos.
- Podría decirse que las bases de datos NoSQL de código abierto tienen una implementación rentable. Ya que no requieren las tarifas de licencia y pueden ejecutarse en hardware de precio bajo. [6]

## 2.4 DESVENTAJAS

- La mayoría de las bases de datos NoSQL no admiten funciones de fiabilidad, que son soportadas por sistemas de bases de datos relacionales. Estas



características de fiabilidad pueden resumirse en: “atomicidad, consistencia, aislamiento y durabilidad.” Esto también significa que las bases de datos NoSQL, que no soportan esas características, ofrecen consistencia para el rendimiento y la escalabilidad.

- Con el fin de apoyar las características de fiabilidad y coherencia, los desarrolladores deben implementar su propio código, lo que agrega más complejidad al sistema.
- Esto podría limitar el número de aplicaciones en las que podemos confiar para realizar transacciones seguras y confiables, como por ejemplo los sistemas bancarios.
- Otras formas de complejidad encontradas en la mayoría de las bases de datos NoSQL, incluyen la incompatibilidad con consultas SQL. Esto significa que se necesita un lenguaje de consulta manual, haciendo los procesos mucho más lentos y complejos. [6]

## **2.5 Casos de uso**

Las bases de datos NoSQL se utilizan de forma generalizada en aplicaciones web en tiempo real y big data, ya que sus principales ventajas son los elevados niveles de escalabilidad y disponibilidad.

Las bases de datos NoSQL también son la opción preferida entre los desarrolladores, ya que propician por su propia naturaleza un desarrollo ágil dada su rápida adaptación a los requisitos en constante cambio.

Las bases de datos NoSQL permiten almacenar los datos de forma más intuitiva y fácil de entender, o más cercanas a la forma en que las aplicaciones utilizan los datos: no necesitan realizar tantas transformaciones cuando almacenan o recuperan datos utilizando interfaces API de NoSQL. Además, las bases de datos NoSQL pueden aprovechar al máximo la nube para evitar por completo el tiempo de inactividad. [7]



## BIBLIOGRAFIA

- [1] F. Tablado, “Bases de datos documentales. ¿Qué es? Tipos y ejemplos”, Ayuda Ley Protección Datos, 10-sep-2020. [En línea]. Disponible en: <https://ayudaleyprotecciondatos.es/bases-de-datos/documentales/>. [Consultado: 02-feb-2024].
- [2] “Base de datos columnar”, Appmaster.io. [En línea]. Disponible en: <https://appmaster.io/es/glossary/base-de-datos-columnar>. [Consultado: 02-feb-2024].
- [3] “Bases de datos clave-valor”, IONOS Digital Guide, 10-mar-2020. [En línea]. Disponible en: <https://www.ionos.mx/digitalguide/hosting/cuestiones-tecnicas/bases-de-datos-clave-valor/>. [Consultado: 02-feb-2024].
- [4] “¿Qué es una base de datos orientada a grafos?”, *Oracle.com*. [En línea]. Disponible en: <https://www.oracle.com/mx/autonomous-database/what-is-graph-database/>. [Consultado: 02-feb-2024].
- [5] “5.1 Conceptos y tipos de bases de datos NoSQL (documental, columnar, clave/valor y de grafos)”, *Github.io*, 03-nov-2023. [En línea]. Disponible en: <https://gltaoada.github.io/tgddbbook/conceptos-y-tipos-de-bases-de-datos-nosql-documental-columnar-clavevalor-y-de-grafos.html>. [Consultado: 02-feb-2024].
- [6] S. Martín, “Bases de datos NoSQL: Guía con las ventajas y desventajas”, Pandora FMS - The Monitoring Blog, 20-abr-2017. [En línea]. Disponible en: <https://pandorafms.com/blog/es/bases-de-datos-nosql/>. [Consultado: 02-feb-2024].
- [7] “¿Qué es NoSQL?”, *Oracle.com*. [En línea]. Disponible en: <https://www.oracle.com/mx/database/nosql/what-is-nosql/>. [Consultado: 02-feb-2024].