

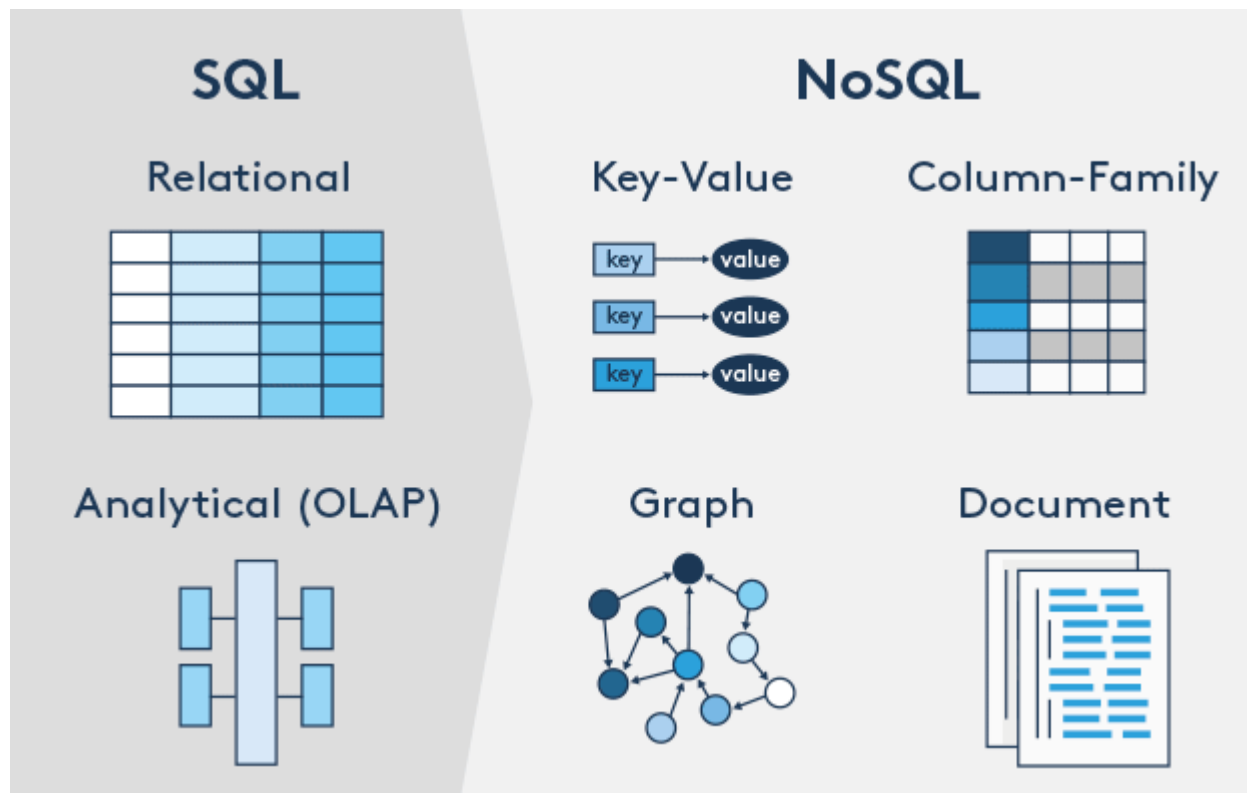


Universidad Nacional  
Autónoma de México

Facultad de Ingeniería



# Tarea #1: Modelo Orientado a Objetos y Modelo No SQL



Bases de Datos

Semestre 2023-1 Grupo 01

Profesor: Ing. Fernando Arreola Franco

Zuriel Zárate García

# Modelo Orientado a Objetos

En el Modelo Orientado a Objetos, se busca implementar la orientación a objetos a las bases de datos. Esto significa que, en lugar de utilizar tipos de datos establecidos, podemos crear nuestros propios tipos de datos para facilitar la implementación de las reglas de negocio en la fase de diseño. Parece ser una tecnología relativamente nueva y en desarrollo, ya que los manejadores que se utilizan carecen de algunas formalidades en sus fundamentos y presentan ciertos problemas. Cabe mencionar que sus manejadores se denominan como “*Sistemas de administración de bases de datos orientadas a objetos*”, o simplemente *OODBMS*.

Algunas de las ventajas y desventajas del modelo orientado a objetos son:

## Ventajas

- Puede utilizarse un sólo lenguaje de programación, que funcione para la aplicación y para el manejo de la base de datos. Por otra parte, los *OODBMS* suelen utilizar un lenguaje de consultas (OQL).
- Simplifican el manejo de imágenes y archivos multimedia en general, documentos, gráficos, etc.
- Facilitan el apoyo a la toma de decisiones debido a la complejidad de abstracción que implica la orientación a objetos.
- El tamaño de los datos puede ser muy grande.
- Se pueden recuperar objetos complejos con rapidez.
- Tienen tendencias que podrían ser muy útiles en el futuro: La implementación de bases de datos con pura lógica de programación, mejor implementación de algoritmos de inteligencia artificial, etc.
- Su uso puede disminuir el costo en personal y de tiempo invertido en el desarrollo de una aplicación específica.

## Desventajas

- La duración de las transacciones puede llegar a ser muy larga.
- Tecnología que sigue en una fase de mejoras.
- El comportamiento de las transacciones y la seguridad puede variar en un sentido negativo.
- Falta de estandarización.

## Modelos No SQL

Los modelos NoSQL hacen referencia al lenguaje SQL, utilizado en bases de datos relacionales. Esto quiere decir que los modelos no SQL son bases de datos no relacionales, sino que se fundamentan en otros modelos. A continuación, listamos algunos tipos de modelos NoSQL:

- De búsqueda: Consiste en mejorar la búsqueda de datos utilizando índices. Por lo general, se utilizan para el análisis de datos y su visualización en tiempo real.
- En memoria: Se enfocan en mejorar el tiempo de respuesta al mantener los datos en memoria principal. Su implementación se suele dar en aplicaciones de compras, videojuegos, sesiones en tiempo real, análisis de datos, etc.
- Documentales: Los desarrolladores de aplicaciones suelen utilizar documentos JSON o XML para almacenar los datos que necesitan de manera sencilla e intuitiva, además de que suele ser más rápido. Este modelo facilita mucho el uso de este tipo de documentos.
- Orientadas a grafos: Se basan en la teoría de grafos y suelen utilizarse en aplicaciones cuyos datos están altamente relacionados. Este modelo es muy reciente y sigue en una fase de mejoras, aunque ha mostrado tener un gran potencial, ya que se han implementado en redes sociales (para mejorar la publicidad), detección de fraude, motores de recomendaciones, etc.
- Clave-Valor: Es un modelo en el que se pueden dividir los datos en simples columnas de una clave identificadora y un valor. Esto hace que el tiempo de respuesta sea muy corto, ya que sólo se necesita la clave única para recuperar un dato de forma casi inmediata. Se suelen utilizar en aplicaciones de compras, cuyos artículos son más fáciles de identificar de esta manera.

A continuación, veremos algunas ventajas y desventajas de los modelos NoSQL:

## Ventajas

- Su implementación es altamente flexible, ya que se tienen varios modelos para utilizarse y adaptarse a las necesidades de una aplicación.
- Su rendimiento es excelente con grandes volúmenes de datos.
- Los DBMS suelen incluir API's y tipos de datos para implementar fácilmente los modelos.
- Son altamente escalables debido a su flexibilidad.

## Desventajas

- Su aumento de rendimiento y escalabilidad implica una disminución en las propiedades *ACID* (Atomicity, Consistency, Isolation, Durability), lo que disminuye su fiabilidad.
- La fiabilidad, coherencia y consistencia suelen pesar más sobre los desarrolladores de la base de datos.
- Debido a la disminución de propiedades *ACID*, el número de aplicaciones en las que puede funcionar de manera correcta y óptima disminuye. Generalmente funcionan mejor en aplicaciones de muchísimos datos.
- No incluye un lenguaje de consulta como SQL, por lo que los procesos de consulta quedan a cargo de los desarrolladores y, por lo anterior, suelen ser más tardados y complejos.

\*

# Referencias

- [1] AWS, *¿Qué es NoSQL? Bases de datos no relacionales con excelente rendimiento y modelos de datos flexibles*, Consultado el 21 de agosto de 2022. Recuperado de: <https://aws.amazon.com/es/nosql/>, 2022.
- [2] CUAED, *Modelo orientado a objetos*, Universidad Nacional Autónoma de México. Consultado el 21 de agosto de 2022. Recuperado de: [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), 2017.
- [3] S. Martín, *Bases de datos NoSQL: Guía definitiva*, Pandora FMS. Consultado el 21 de agosto de 2022. Recuperado de: <https://pandorafms.com/blog/es/bases-de-datos-nosql/>, 2021.