

# Tarea 1

Alan Dunzz Llampallas

2 de enero del 2024

## 1 Modelo orientado a objetos

- Descripción: Es un modelo de bases de datos basado en el paradigma de la programación orientada a objetos. Dentro de este contexto, una base de datos es un conjunto de objetos almacenados que pueden ser accedidos por múltiples usuarios y aplicaciones. Su definición se hace mediante el Lenguaje de Definición de Objetos (ODL) Un objeto es una instancia autocontenida de una entidad de interés del mundo real y cuenta con un identificador único. Los objetos cuentan con operaciones, que están dispuestas de entradas y salidas. Asimismo, tienen propiedades, que corresponden a sus atributos y las relaciones. El otro componente básico además del objeto es la literal, que es un valor específico que puede ser booleano, entero, flotante, etc.
- Ventajas: Mayor capacidad de modelado al permitir capturar estado y comportamiento de un objeto y poder aplicar el concepto de herencia, donde unos objetos adquieren características de otros. Ampliabilidad al poder construir nuevos tipos de datos a partir de los existentes y agrupar propiedades comunes a clases e incluirlas en una superclase. Permite desarrollar aplicaciones más avanzadas. Tienen mejor rendimiento en comparación con las bases de datos tradicionales.
- Desventajas: No se cuenta con un modelo de datos universalmente aceptado para sus manejadores. Al ser un modelo relativamente nuevo, aún no se tiene tanta experiencia como con los modelos tradicionales. La mayoría de sus modelos carecen de una base teórica, a comparación de las bases de datos relacionales, que cuentan con una base sólida y un estándar aprobado.
- Casos de uso: Aplicaciones más complejas donde se emplean lenguajes de programación orientados a objetos, de tal forma que se puede especificar la estructura de los objetos (estado) y operaciones que pueden aplicarse en éstos (comportamientos). Utiliza acceso navegacional, por lo que es más adecuada para despieces y consultas recursivas.

## 2 Modelos NoSQL

Inicialmente se refería a bases de datos que no emplean el Lenguaje de Consulta Estructurado (SQL). Actualmente abarca a bases de datos que integran ventajas de modelos NoSQL con características de las bases de datos SQL. El diseño de estas bases de datos se originó de la necesidad de tener modelos que pudieran escalarse para ser utilizados en aplicaciones Web con grandes volúmenes de datos y garantizar un acceso rápido a la información. Los datos no cuentan con las características ACID (Atomicidad, Consistencia, Aislamiento y Durabilidad).

### 1. Clave-valor

- Descripción: Está basado en la forma en que la programación en ensamblador accede a memoria. Los almacenamientos key-value asocian una clave única (key) al valor que se quiere guardar (value). La forma de acceder a la información presenta también cierta similitud con un diccionario. Algunos ejemplos son Cassandra y DynamoDB
- Ventajas: Se trata de un modelo que permite acceder a la información muy rápido. Se tiene una alta escalabilidad y buen rendimiento.

- Desventajas: No sigue el paradigma ACID. No se verifica la integridad de los datos ni referencias externas.
- Casos de uso: Se utiliza cuando se tienen volúmenes de información de gran tamaño, característicos del Big Data. Cuando la aplicación requiere acceder a los datos con gran velocidad.

## 2. Documentales

- Descripción: Bases de datos en las cuales la información es almacenada como un listado de documentos estructurados. Dentro de cada uno de estos documentos se pueden ingresar campos junto con los valores asociados a estos. Algunos ejemplos son MongoDB y CouchDB. Los documentos guardados pueden ser en formatos textuales como XML, JSON o BSON o formato binarios como PDF o Word. Los documentos son almacenados dentro de una serie de contenedores llamados colecciones y tienen una clave única dentro de ellos.
- Ventajas: Son rápidos al recuperar la información de un documento en específico. Cuentan con un esquema de datos de gran flexibilidad.
- Desventajas: Son lentos si se buscan todos los documentos de un determinado campo. Abarcan un mayor espacio.
- Casos de uso: Son particularmente útiles cuando se llevan a cabo transacciones o datos enviados desde dispositivos móviles con JSON y se desea almacenar la información de estos movimientos.

## 3. Orientadas a grafos

- Descripción: Su principal característica es que organizan la información en grafos dirigidos. Esto se logra dividiendo la información en trozos más básicos llamados chunks y estableciendo relaciones entre ellos. Se utilizan nodos y aristas en vez de entidades y relaciones. Algunos ejemplos son Neo4J, Infinite Graph y Allegro Graph.
- Ventajas: Son óptimas para hacer operaciones de consultas. Son extremadamente rápidas. Su rendimiento no se deteriora al crecer la base de datos. Se tiene gran flexibilidad para manipular el conjunto de datos.
- Desventajas: No se tiene un lenguaje de consulta estandarizado. Puede ser difícil la escalabilidad.
- Casos de uso: Este tipo de bases de datos es particularmente útil cuando se tienen múltiples contactos y relaciones. Un caso particularmente importante es el de las redes sociales.

# 3 Referencias

1. Piqueres, J.(2007) "Bases de datos orientadas a objetos". IES San Vicente. [En línea]. Disponible en: <https://iessanvicente.com/colaboraciones/>
2. Joyanes, L. (2019) "Inteligencia de negocios y analítica de datos". Editorial Alfaomega. [En línea]. Disponible en: [https://libroweb.alfaomega.com.mx/book/1007/free/data/contenidos\\_cap6.pdf](https://libroweb.alfaomega.com.mx/book/1007/free/data/contenidos_cap6.pdf)
3. Herranz, R. (2014). "Bases de datos NoSQL: Arquitectura y ejemplos de aplicación". Departamento de Informática. Universidad Carlos III de Madrid. [En línea]. Disponible en: <https://core.ac.uk/download/pdf/44310803.pdf>