



BASES DE DATOS TAREA

Estudiante: Sandra Susana Pérez Gutiérrez

Grupo: 3

Fecha: 2 de Octubre del 2020

sandraconnors271198@gmail.com

Semestre 2021-1

Investigar que es un modelo orientado a objetos, modelo objeto relacional y un modelo NoSQL.





Contents

1	Modelo orientado a objetos	4
1.1	Ventajas	4
1.2	Desventajas	4
2	Modelo objeto relacional	5
2.1	Ventajas	5
2.2	Desventajas	5
3	Modelo NoSQL	6
3.1	Ventajas	6
3.2	Desventajas	6
4	Referencias	7



1 Modelo orientado a objetos

Una base de datos orientada a objetos es aquella que implementa dentro de un sistema de información la representación de datos en forma de objetos. Esto mediante la aplicación de programación orientada a objetos. Estas bases de datos se diferencian de las bases de datos relacionales clásicas, debido a que no responden a un sistema de tablas para registrar su información.

La mayoría de las bases de objetos ofrecen lenguajes de consultas para encontrar los objetos que pertenecen a la base. Para ello se utiliza un enfoque de programación declarativa. El acceso a los datos en estas bases es más rápido debido a que los objetos pueden ser recuperados de forma directa sin búsqueda específica.

1.1 Ventajas

- Lenguajes de Consulta: a partir de la programación declarativa se recupera todo el objeto sin necesidad de una búsqueda a un atributo en específico, esto hace una recuperación directa a todo un objeto sin necesidad de una búsqueda en específico.

- Persistencia transparente: Se refiere a la posibilidad de la base de datos de utilizar uno o varios objetos a través del lenguaje de programación para obtener análisis o manipulación de la data, es decir, por medio de las relaciones de los objetos da un análisis completo de los datos.

- Transacciones ACID: Estas bases de datos tienen un alto nivel de confiabilidad en la ejecución de transacciones ACID (atomicidad, consistencia, aislamiento y durabilidad). Esto significa que las transacciones se cumplen o terminan únicamente cuando no existen conflictos con los datos. También los usuarios pueden abortar las acciones en caso de que se presente algún error en el sistema de la base de datos.

- Almacenamiento en cache: Las bases de datos orientadas a objetos tienen un mecanismo de almacenamiento en cache que crea replicas parciales de la base de datos. El almacenamiento en cache es un método de almacenamiento y resguardo de datos que permite minimizar el acceso al disco o el flujo de tráfico de la red. Esto permite que las aplicaciones accedan a objetos desde la memoria del programa en lugar de utilizar el almacenamiento en disco.

1.2 Desventajas

- Carecen de madurez en el desarrollo: esto lleva a un mayor riesgo de sostenibilidad de la base de datos en el largo plazo. No hay criterios claros de estandarización, ya que estas apenas están reemergiendo en el mercado digital.

- Poca experiencia y documentación sobre proyectos que las implementen. Esto pudiese extender los tiempos de desarrollo de un proyecto con este modelo de bases de datos.

- Gastos de capacitación adicional para los equipos de desarrolladores ya que no existe una familiaridad en los entornos del mercado digital con el paradigma de objetos.



2 Modelo objeto relacional

El modelo de base de datos objeto-relacional integra los conceptos de la tradicional base de datos relacional y los conceptos de paradigma de objetos que se utiliza en la programación orientada a objetos (POO). El objetivo de este concepto es poder aplicar la tecnología madura de bases de datos relacionales sobre la organización de los datos complejos es decir datos de texto e imagen, mapas, datos en el rango de audio etc. Las bases de datos Objeto-relacional son compatibles con estos objetos de datos y las operaciones de mayor complejidad.

Una base de datos orientada a objetos es una base de datos que incorpora todos los conceptos importantes del paradigma de objetos:

- Encapsulación: Propiedad que permite ocultar la información al resto de los objetos, impidiendo así accesos incorrectos o conflictos.

- Herencia: Propiedad a través de la cual los objetos heredan comportamiento dentro de una jerarquía de clases.

- Polimorfismo: Propiedad de una operación mediante la cual puede ser aplicada a distintos tipos de objetos.

Los objetos en la base de datos constan de dos partes:

1. Atributos: Datos, meta-datos e información de localización.
2. Métodos: Son funciones cuyos parámetros son los atributos del objeto.

2.1 Ventajas

- En bases de datos orientadas a objetos, los usuarios pueden definir operaciones sobre los datos como parte de la definición de la base de datos. Una operación (llamada función) se especifica en dos partes. La interfaz (o signatura) de una operación incluye el nombre de la operación y los tipos de datos de sus argumentos (o parámetros). La implementación (o método) de la operación se especifica separadamente y puede modificarse sin afectar la interfaz. Los programas de aplicación de los usuarios pueden operar sobre los datos invocando a dichas operaciones a través de sus nombres y argumentos, sea cual sea la forma en la que se han implementado. Esto podría denominarse independencia entre programas y operaciones.

- Genera la posibilidad de guardar objetos más complejos en una sola tabla con referencias a otras relaciones, con lo que se acerca más al paradigma de programación orientada a objetos.

2.2 Desventajas

- No todos los SGDB (sistemas gestor de bases de datos) soportan las extensiones objeto-relacional.



3 Modelo NoSQL

Las bases de datos NoSQL están diseñadas específicamente para modelos de datos específicos y tienen esquemas flexibles para crear aplicaciones modernas. Las bases de datos NoSQL son ampliamente reconocidas porque son fáciles de desarrollar, por su funcionalidad y el rendimiento a escala. Son bases de datos no relacionales que utilizan una variedad de modelos de datos para acceder y administrar datos. Estos tipos de bases de datos están optimizados específicamente para aplicaciones que requieren grandes volúmenes de datos, baja latencia y modelos de datos flexibles, lo que se logra mediante la flexibilización de algunas de las restricciones de coherencia de datos en otras bases de datos.

3.1 Ventajas

Las bases de datos NoSQL se adaptan perfectamente a muchas aplicaciones modernas, como dispositivos móviles, web y juegos, que requieren bases de datos flexibles, escalables, de alto rendimiento y altamente funcionales para proporcionar excelentes experiencias de usuario.

- Flexibilidad: las bases de datos NoSQL generalmente ofrecen esquemas flexibles que permiten un desarrollo más rápido y más iterativo. El modelo de datos flexible hace que las bases de datos NoSQL sean ideales para datos semiestructurados y no estructurados.

- Escalabilidad: las bases de datos NoSQL generalmente están diseñadas para escalar usando clústeres distribuidos de hardware en lugar de escalar añadiendo servidores caros y sólidos. Algunos proveedores de la nube manejan estas operaciones en segundo plano, como un servicio completamente administrado.

- Alto rendimiento: la base de datos NoSQL está optimizada para modelos de datos específicos y patrones de acceso que permiten un mayor rendimiento que el intento de lograr una funcionalidad similar con bases de datos relacionales.

- Altamente funcional: las bases de datos NoSQL proporcionan API altamente funcionales y tipos de datos que están diseñados específicamente para cada uno de sus respectivos modelos de datos.

3.2 Desventajas

- No todas las bases de datos no relacionales contemplan la propiedad de atomicidad.

- Algunas consultas con el lenguaje SQL (el más extendido en el mundo) no se pueden traducir en consultas NoSQL.

- No hay una estandarización entre todos los SGDB no relacionales existentes.



4 Referencias

-Graph everywhere, " *Bases de datos NoSQL — Bases de datos orientadas a objetos*", <https://www.grapheverywhere.com/es/de-datos-orientadas-objetos/>, 30 de Septiembre del 2020.

-Enzyme advising group, " *Ejemplos de base de datos: arquitecturas , ventajas y desventajas*", Albert Ramoneda, <https://blog.enzymeadvisinggroup.com/ejemplos-de-base-de-datos>, 30 de Septiembre del 2020.