



Tarea 1

MODELO ORIENTADO A OBJETOS

En una base de datos orientada a objetos, los componentes se almacenan como objetos y no como datos, tal y como hace una base relacional, cuya representación son las tablas.

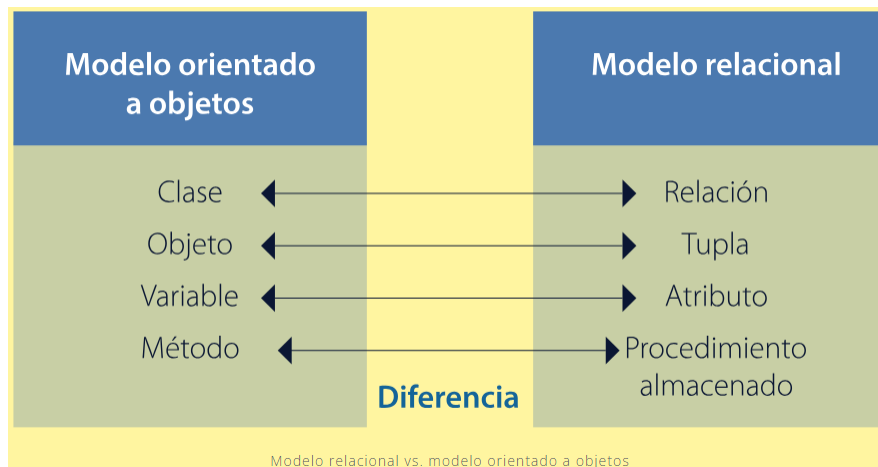
Los administradores de base de datos (DBMS por sus siglas en inglés) evolucionan con el afán de satisfacer nuevos requerimientos tecnológicos y de información. Aunque los DBMS relacionales (RDBMS) son actualmente líderes del mercado y brindan las soluciones necesarias a las empresas comerciales, existen aplicaciones que necesitan funciones con las que no cuentan.

Las CAD/CAM, los sistemas multimedia, como los geográficos y de medio ambiente, los de gestión de imágenes y documentos y los de apoyo a las decisiones necesitan de modelos de datos complejos, difíciles de representar como tuplas de una tabla.

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.

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.



Las principales características de la orientación a objetos son:

- Cada objeto tiene un nombre, atributos y operaciones.
- Es una tecnología para producir modelos que reflejen un dominio de negocio y utiliza la terminología propia de tal dominio.
- Cuenta con cinco conceptos subyacentes: objeto, mensajes, clases, herencia y polimorfismo.
- Un objeto tiene un estado, un comportamiento y una identidad.
- Los mensajes brindan comunicación entre objetos.
- Las clases son un tipo de plantilla usada para definir objetos, los cuales son instancias del mundo real.

MODELOS NoSQL

NoSQL, también conocido como "no solo SQL", "no SQL", es un enfoque para el diseño de bases de datos que permite el almacenamiento y la consulta de datos fuera de las estructuras tradicionales que se encuentran en las bases de datos relacionales. Si bien aún puede almacenar datos que se encuentran dentro de los sistemas de gestión de bases de datos relacionales (RDBMS), los almacena de manera diferente a un RDBMS.

En lugar de la estructura tabular típica de una base de datos relacional, las bases de datos NoSQL albergan datos dentro de una estructura de datos, como un documento JSON. Dado que este diseño de base de datos no relacional no requiere un esquema, ofrece escalabilidad rápida para gestionar grandes conjuntos de datos normalmente no estructurados.

NoSQL también es un tipo de base de datos distribuida, lo que significa que la información se copia y almacena en varios servidores, que pueden ser remotos o locales. De esta manera, se garantiza la disponibilidad y la confiabilidad de los datos. Si algunos de los datos se desconectan, el resto de la base de datos puede continuar ejecutándose.

TIPOS DE BASES DE DATOS NoSQL

- Almacén de pares clave valor

Este modelo de datos sin esquema se organiza en un diccionario de pares clave-valor, donde cada elemento tiene una clave y un valor. La clave podría ser algo similar que se encuentra en una base de datos SQL, como un ID de carrito de compras, mientras que el valor es una matriz de datos, como cada artículo individual en el carrito de compras de ese usuario.

VENTAJAS:

- **Escalabilidad:** Como todas las consultas de los usuarios requieren la interacción de datos, las bases de datos a menudo pueden convertirse en un obstáculo para el rendimiento de las aplicaciones. Muchas bases de datos de clave-valor ofrecen soporte integrado para funciones de escalado avanzadas. Se escalan horizontalmente y distribuyen automáticamente los datos entre los servidores para reducir los obstáculos en un solo servidor.
- **Facilidad de uso:** Las bases de datos de clave-valor siguen el paradigma orientado a objetos que permite a los desarrolladores mapear objetos del mundo real directamente a objetos de software. En lugar de mapear sus objetos de código a varias tablas subyacentes, los ingenieros pueden crear pares clave-valor que coincidan con sus objetos de código. Esto hace que el uso de las tiendas de clave-valor sea más intuitivo para los desarrolladores.
- **Rendimiento:** Las bases de datos de clave-valor procesan operaciones de lectura-escritura constantes con llamadas al servidor de bajo costo. La latencia mejorada y el tiempo de respuesta reducido ofrecen un mejor rendimiento a escala. Se basan en estructuras simples de una sola tabla, en lugar de en múltiples tablas interrelacionadas. A diferencia de las bases de datos relacionales, las bases de datos de clave-valor no tienen que realizar uniones de tablas que consumen muchos recursos, lo que las hace mucho más rápidas.

DESVENTAJAS:

- **Ausencia de consultas complejas:** Como las bases de datos de clave-valor no admiten consultas complejas, los desarrolladores deben solucionar este problema en el código. Las operaciones de datos se realizan principalmente mediante términos de lenguaje de consulta simples como “get”, “put” y “delete”. Existen limitaciones en cuanto a la cantidad de datos que se pueden filtrar y clasificar antes de acceder a ellos.
- **Mala administración del esquema:** El diseño de la tienda de clave-valor no impone un esquema a los desarrolladores. Cualquier persona puede modificar el esquema en el programa de base de datos. Los equipos de desarrollo deben planificar el modelo de datos de forma sistemática para evitar problemas a largo plazo. La falta de un esquema estricto también significa que la aplicación es responsable de la interpretación adecuada de los datos que consume, lo que a menudo se denomina “esquema de lectura”.

CASOS DE USO:

- **Administración de sesiones:** Una aplicación orientada a la sesión, como una aplicación web, inicia una sesión cuando un usuario inicia sesión en una aplicación y está activo hasta que el usuario cierra la sesión o esta expira. Durante este período, la aplicación almacena todos los atributos de la sesión del usuario en la memoria principal o en una base de datos. Cada sesión de usuario tiene un identificador único. Solo la clave principal consulta los datos de la sesión nunca, por lo que un almacén clave-valor rápido se adapta mejor a los datos de la sesión.
- **Carro de compras:** un sitio web de puede recibir miles de millones de pedidos en segundos. Las bases de datos de clave-valor 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. Los almacenes de clave-valor también tienen redundancia incorporada, lo cual puede manejar la pérdida de nodos de almacenamiento.
- **Almacenamiento en caché:** Las aplicaciones de redes sociales pueden almacenar datos a los que se accede con frecuencia, como el contenido de las fuentes de noticias. Los sistemas de almacenamiento en caché de datos en memoria también utilizan almacenes de clave-valor para acelerar las respuestas de las aplicaciones.

- Almacén de documentos

Las bases de datos de documentos almacenan datos como documentos. Pueden ser útiles para gestionar datos semiestructurados y, por lo general, los datos se almacenan en formatos JSON, XML o BSON. Esto mantiene los datos juntos cuando se usan en aplicaciones, lo que reduce la cantidad de conversión necesaria para usarlos.

VENTAJAS:

- **Facilidad de desarrollo:** Los documentos JSON se asignan a objetos, un tipo de datos común en la mayoría de los lenguajes de programación. Al crear aplicaciones, los desarrolladores pueden crear y actualizar documentos de forma flexible directamente desde el código. Esto significa que dedican menos tiempo a crear modelos de datos de antemano. Por lo tanto, el desarrollo de aplicaciones es más rápido y eficiente.
- **Esquema flexible:** Una base de datos orientada a documentos le permite crear varios documentos con diferentes campos dentro de la misma colección. Esto puede resultar útil cuando se almacenan datos no estructurados, como correos electrónicos o publicaciones en redes sociales. Sin embargo, algunas bases de datos de documentos ofrecen validación de esquemas, por lo que puede imponer algunas restricciones a la estructura.
- **Rendimiento a escala:** Las bases de datos de documentos ofrecen capacidades de distribución integradas. Puede escalarlas horizontalmente en varios servidores sin afectar al rendimiento, lo que también es rentable. Además, las bases de datos de documentos proporcionan tolerancia a errores y disponibilidad a través de la replicación integrada.

DESVENTAJAS:

- No utilizan el lenguaje SQL como lenguaje principal de consulta, aunque sí lo pueden

usar de apoyo. Es decir, al contrario que las bases relacionales, no existe un lenguaje estandarizado para la creación de estas bases de datos.

- No siempre pueden garantizar las propiedades ACID de atomicidad, consistencia, integridad y durabilidad.
- No tienen una gran comunidad detrás y existen mucha menos información acerca de estas bases de datos.
- Los índices pueden ocupar mucha memoria RAM, sobre todo en las bases documentales que manejan un gran volumen de datos.

CASOS DE USO:

- **Administración de contenido:** Una base de datos de documentos es una excelente opción para aplicaciones de administración de contenido, como blogs y plataformas de video. Con una base de datos de documentos, cada entidad que rastrea la aplicación se puede almacenar como un único documento. La base de datos de documentos es una manera más intuitiva para que un desarrollador actualice una aplicación a medida que evolucionan los requisitos. Además, si el modelo de datos necesita cambiar, solo se deben actualizar los documentos afectados.
- **Catálogos:** en una aplicación de e-commerce, los diferentes productos generalmente tienen diferentes números de atributos. La administración de miles de atributos en bases de datos relacionales no es eficiente y afecta al rendimiento de lectura. Al utilizar una base de datos de documentos, los atributos de cada producto se pueden describir en un solo documento para que la administración sea fácil y la velocidad de lectura sea más rápida. Cambiar los atributos de un producto no afectará a otros.
- **Gestión de sensores:** Los datos del sensor suelen llegar como un flujo continuo de valores variables. Debido a problemas de latencia, es posible que algunos objetos de datos estén incompletos, duplicados o que falten. Además, debe recopilar un gran volumen de datos antes de poder filtrarlos o resumirlos para su análisis.

Los almacenes de documentos son más convenientes en este caso. Puede almacenar rápidamente los datos del sensor tal como están, sin limpiarlos ni ajustarlos a esquemas predeterminados. También puede escalarlo según sea necesario y eliminar documentos completos una vez que se hayan realizado los análisis.

- Almacén de grafos

Este tipo de base de datos normalmente alberga datos de un gráfico de conocimientos. Los elementos de datos se almacenan como nodos, bordes y propiedades. Cualquier objeto, lugar o persona puede ser un nodo. Un borde define la relación entre los nodos. Las bases de datos de gráficos se utilizan para almacenar y gestionar una red de conexiones entre elementos dentro del gráfico.

VENTAJAS:

- **Flexibilidad:** El esquema y la estructura de los modelos de grafos pueden cambiar con sus aplicaciones. Los analistas de datos pueden agregar o modificar las estructuras de grafos existentes sin afectar a las funciones existentes. No es necesario modelar los dominios por adelantado.
- **Rendimiento:** El rendimiento de la base de datos de grafos mejora en varios órdenes de magnitud cuando se consultan relaciones. El rendimiento se mantiene constante incluso cuando aumenta el volumen de datos de los grafos.
- **Eficiencia:** Las consultas de grafos son más cortas y más eficientes al generar los mismos informes en comparación con las bases de datos relacionales. Las tecnologías de grafos aprovechan los nodos enlazados. Recorrer las uniones o relaciones es un proceso muy rápido, ya que las relaciones entre nodos no se calculan en el momento de la consulta, sino que se mantienen en la base de datos.

DESVENTAJAS:

- **Escalabilidad:** Es difícil de escalar, por estar diseñado para arquitecturas con un solo servidor.
- Los índices pueden ocupar mucha memoria RAM, sobre todo en las bases documentales que manejan un gran volumen de datos.
- **Baja comunidad:** Al ser un tipo de BBDD bastante reciente, la comunidad aún no es muy grande y, por lo tanto, cuesta encontrar soporte a los distintos problemas que se pueden dar.
- **Inexistencia de lenguaje estandarizado:** Cada plataforma de BDOG tiene un lenguaje de peticiones diferente por lo que no hay un lenguaje estandarizado para realizar peticiones a diferencia de las bases de datos relacionales con el SQL (Structure Query Language).
- **Ineficiencia para peticiones transaccionales:** las bases de datos de grafos no funcionan bien para peticiones transaccionales a diferencia de las BBDD relacionales.

CASOS DE USO:

- **Detección de fraude:** Puede utilizar relaciones en bases de datos de grafos para procesar transacciones financieras casi en tiempo real. Con consultas de grafos rápidas, puede detectar que un comprador potencial está utilizando la misma dirección de correo electrónico y la misma tarjeta de crédito incluidas en un caso de fraude conocido. Las bases de datos de grafos también pueden ayudarlo a detectar fraudes a través de patrones de relación, como varias personas asociadas a una dirección de correo electrónico personal o varias personas que comparten la misma dirección IP pero residen en ubicaciones físicas diferentes.
- **Motores de recomendaciones:** Puede almacenar relaciones de grafos entre categorías de información como intereses de clientes, amigos e historial de compras. Puede utilizar una base de datos de grafos de alta disponibilidad para hacer recomendaciones de productos a un usuario basándose en qué productos han comprado otras personas con

intereses e historiales de compra similares. También puede identificar a las personas que tienen un amigo en común pero que aún no se conocen y hacerles una recomendación de amistad.

- **Descubrimiento de patrones:** Las bases de datos de grafos son ideales para descubrir relaciones complejas y patrones ocultos en los datos. Por ejemplo, una empresa de redes sociales utiliza una base de datos de grafos para distinguir entre cuentas de bots y cuentas reales. Analiza la actividad de la cuenta para descubrir conexiones entre las interacciones de la cuenta y la actividad de los bots.

Referencias

AWS, ¿Qué es una base de datos de clave-valor?. [Online]. Available: <https://aws.amazon.com/es/nosql/key-value/> [Accessed Feb. 09, 2024].

AWS, ¿Qué es una base de datos de documentos?. [Online]. Available: <https://aws.amazon.com/es/nosql/document/> [Accessed Feb. 09, 2024].

AWS, ¿Qué es una base de datos de grafos?. [Online]. Available: <https://aws.amazon.com/es/nosql/graph/> [Accessed Feb. 10, 2024].

ayudaley, Bases de datos documentales. ¿Qué es? Tipos y ejemplos. [Online]. Available: <https://ayudaleyprotecciondatos.es/bases-de-datos/documentales/> [Accessed Feb. 09, 2024].

CUAED, Modelo Orientado a Objetos. [Online]. Available: https://programas.cuaed.unam.mx/repositorio/moodle/pluginfile.php/782/mod_resource/content/8/contenido/index.html [Accessed Feb. 08, 2024].

IBM, ¿Qué son las bases de datos NoSQL?. [Online]. Available: <https://www.ibm.com/mx-es/topics/nosql-database> [Accessed Feb. 08, 2024].

IONOS, Graph database: bases de datos para una interconexión eficiente, 2019. [Online]. Available: <https://www.ionos.mx/digitalguide/hosting/cuestiones-tecnicas/graph-database/> [Accessed Feb. 10, 2024].

R. Cañadas, (2022, En. 18). Base de datos orientada a grafos o graph database. [Online]. Available: <https://goo.su/liU51w> [Accessed Feb. 10, 2024].