

Lectura 6/7

Estudiante: Mario Lara Molina

¿De qué forma se diferencia el BASE consistency model del ACID?

En las bases NoSQL los modelos de consistencia de datos pueden variar de los usados en bases SQL, los dos modelos más comunes son ACID y BASE, ambos con sus ventajas y desventajas.

El modelo ACID también es utilizado en bases de datos SQL, por lo que algunos desarrolladores están familiarizados con este concepto, la clave de ACID garantiza un ambiente seguro en el que se pueden operar datos. El acrónimo corresponde a:

- Atómico: Todas las transacciones se completan en su totalidad o son devueltas a un estado anterior (roll back).
- Consistencia: Al momento de completar una transacción, la base de datos es estructuralmente sólida.
- Isolada: Las transacciones no deben contender con otras, el acceso a los datos es moderado por la base de datos con el fin de que las transacciones se ejecuten secuencialmente.
- Duradero: Los resultados de una transacción son permanentes, incluso si ocurre algún error.

Las propiedades de ACID permiten que cuando se completa una transacción, los datos son consistentes y estables en el disco.

La consistencia en las escrituras puede ser algo maravilloso para los desarrolladores, pero requiere procesos de bloqueo y patrones de mucho peso para la mayoría de casos de uso. En tecnologías NoSQL se utiliza ACID para garantizar la seguridad y consistencia de los datos almacenados.

Para la mayoría de dominios y casos de uso, las transacciones de ACID son más pesimistas de lo que se requiere, en el mundo NoSQL, las transacciones ACID son pasadas de moda ya que algunas bases de datos perdieron algunos requerimientos por consistencia inmediata, frescura de datos y exactitud para poder obtener beneficios como escalabilidad y resiliencia.

El acrónimo BASE corresponde a:

- Disponibilidad básica (Basic Availability): La base de datos funciona bien la mayoría del tiempo.
- Estado suave (Soft State): Los almacenes no necesitan ser consistentes con las escrituras, pero las réplicas deben ser consistentes mutuamente todo el tiempo.
- Consistencia eventual (Eventual consistency): Los almacenes exhiben consistencia hasta algún punto muy lejano.

Las propiedades de BASE son menos estrictas que las de ACID, sin embargo, no hay diferencias de uno a uno establecidas entre ambos modelos de datos. Un almacén de BASE valora la disponibilidad pero no ofrece consistencia de los datos replicados que se escriben en el momento, por otro lado, BASE ofrece más libertad que ACID, por lo que los datos son consistentes con el tiempo.

El modelo BASE se utiliza para almacenamientos agregados, incluyen familias de columnas, valores llave y almacenes de documentos.

¿Explique porque no es recomendable modelar una base de datos orientada

a grafos mediante una base de datos relacional?

El modelado de datos es un proceso de abstracción, el cual comienza con las necesidades del usuario o negocio y transforma esas necesidades en una estructura para almacenar y organizar información.

Con bases de datos relacionales el proceso de modelado es algo complicado, ya que se requiere de un modelo lógico el cual relacione todas las tablas, a su vez que un modelo físico. Por otro lado, modelar con una base de datos orientada a grafos es muy sencillo, al dibujar el sistema con círculos y cajas se puede elaborar un grafo que funciona como base de datos con pocas líneas de código.

Las bases de datos relacionales no son diseñadas para tener cambios constantes en su estructura debido a sus esquemas rígidos y al proceso de modelado complejo, por otro lado, las bases de datos orientadas a grafos solucionan estos problemas.

Por lo que no es recomendable modelar una base de datos orientada a grafos mediante una base de datos relacional, al tener que cumplir con consultas complejas en poco tiempo, cosa que con las bases de datos relacionales se complica demasiado, además de que el proceso de modelar en una base de datos relacional implica la creación de tablas intermedias para realizar el enlace entre tablas, lo que hace que el proceso de búsqueda sea más lento y las relaciones no sean del todo eficientes.

¿Qué es una base de datos orientada a grafos? Explique casos de uso.

Una base de datos orientada a grafos es relativamente más sencilla que una base de datos, un grafo está compuesto por dos elementos, un nodo y una relación. Cada nodo representa una entidad (persona, lugar, cosa, categoría, o cualquier tipo de dato) y cada relación representa cómo dos nodos están relacionados.

Una base de datos orientada a grafos se ajusta a un contexto donde se tienen presupuestos, cronogramas, estándares corporativos y competidores.

- **Rendimiento:** El volumen de datos definitivamente va a crecer con el tiempo, con bases de datos tradicionales las consultas relacionales comienzan a resultar ineficientes cuando el número y la profundidad de las relaciones aumenta. Por otro lado, el rendimiento de una base de datos de grafos se mantiene constante conforme los datos crecen.
- **Flexibilidad:** Con bases de datos de grafos, la parte de IT y arquitectos de datos pueden seguir el ritmo del negocio, porque la estructura y el esquema de un modelo de datos de grafos funciona como solución y cambio industrial. Por lo que el equipo no debe modelar exhaustivamente el dominio con anticipación, simplemente pueden añadirlo a las estructuras existentes sin poner en peligro las funcionalidades actuales.
- **Agilidad:** Desarrollar con bases de datos de grafos se puede trabajar perfectamente con prácticas de desarrollo ágiles, lo que permite que la aplicación evolucione al mismo tiempo que los requerimientos cambiantes del negocio.

En una base de datos orientada a grafos, las relaciones toman el primer lugar en la jerarquía de prioridad, esto significa que la aplicación no necesita inferir conexiones de datos usando llaves foráneas o procesamiento fuera de banda.

Existen dos propiedades importantes de las bases de datos orientadas a grafos:

- Almacenamiento: Algunas bases de datos orientadas a grafos usan almacenamiento "nativo" que es específicamente diseñado para almacenar y manejar grafos, mientras que otras usan bases de datos relacionales u orientadas a objetos, lo que es más lento que un acercamiento de almacenamiento "nativo".
- Procesamiento: Procesamiento de grafos nativo es la manera más eficiente de procesar datos dentro de un grafo, debido a la conexión física de los nodos en cada base de datos, mientras que el procesamiento de grafos no nativo realiza las operaciones de crear, leer, actualizar y eliminar de manera diferente.

Las bases de datos orientadas a grafos son extremadamente útiles en manejar conjuntos de datos muy grandes.