Trabajo Práctico

Laboratorio

Integrantes
- Ivan Donato Ngua
- Daniel Seoane
- Lucas Bonanni
-

Docente
Juan Lopreiato

Fecha de entrega
27/10/2022

Guía de contenido

Introducción

Mientras una base de datos relacional está optimizada para almacenar filas de datos, normalmente para aplicaciones transaccionales, una base de datos en columnas está optimizada para lograr una recuperación rápida de columnas de datos, normalmente en aplicaciones analíticas.

Las bases de datos columnares almacenan datos en registros de manera que puedan contener un gran número de columnas dinámicas. A diferencia de base de datos relacionales que están optimizadas para almacenar filas de datos.

Las bases de datos columnares se introdujeron por primera vez en 1970 en productos como Model 204 y ABABAS, desde 2004 han tenido una evolución constante para implementaciones comerciales. Hoy en día su desarrollo y aplicación ha resultado en una gran competencia y variedad de opciones en bases de datos columnares.

Por último, hay que destacar que desde su diseño las bases columnares están pensadas para reducir la escala de clústeres distribuidos en hardware de bajo costo. Para aumentar el desempeño. Lo que los hace una de las primeras opciones en cuanto a procesamiento de Big Data y para almacenamiento de datos.

Ventajas

Las bases de datos basadas en columnas son creadas para la velocidad, trabajan de una forma que permite omitir los datos irrelevantes para el análisis y leer de inmediato lo que se busca. De esta manera las consultas de agregación se vuelven especialmente rápidas. Las principales ventajas son:

Escalabilidad

La mayor ventaja que tiene este tipo de bases de datos es la escalabilidad, principalmente para almacenamiento de big data. Con la habilidad para esparcirse en múltiples nodos dependiendo de la escala de la base de datos, soporta procesamiento en paralelo, lo que significa que soporta varios procesadores para que trabajen en el mismo set de datos.

Compresión

Generalmente, estos motores también son buenos guardando y comprimiendo estos datos para ahorrar espacio de almacenamiento. Los almacenes de columnas son muy eficientes en la compresión y/o partición de datos. La compresión permite que las operaciones en columna, como MIN, MAX, SUM, COUNT y AVG, se realicen muy rápidamente. Debido a su estructura, las bases de datos en columnas funcionan particularmente bien con las consultas de agregación (como SUM, COUNT, AVG, etc.).

Rapidez

El tiempo de carga es mínimo, las consultas realizadas se ejecutan rápidamente, ya que están diseñadas para contener enormes cantidades de datos y que sean prácticas para el análisis de datos.

Desventajas

Diseño de índices para el esquema

Es muy difícil hacer el diseño efectivo de un esquema.

Mecanismos de seguridad

No tiene mecanismos de seguridad como si tiene las bases de datos relacionales donde se puede restringir por esquemas, vistas, etc.

Procesamiento transaccional en línea

Las bases de datos columnares no son eficientes en el procesamiento transaccional en línea, tanto como para el procesamiento de datos analíticos. Lo que quiere decir que no son muy buenos actualizando transacciones, pero están diseñados para analizarlas.

Carga incremental de datos

Mientras las cargas incrementales no son imposibles, las bases de datos columnares no son las más eficientes para ello, porque para eso primero se tendría que escanear la columna para identificar las filas y se tiene que escanear nuevamente para buscar datos modificados o que necesitan ser muy escritas.

Consulta específica de filas

Para poder consultar una fila específica es necesario agregar un paso extra que es identificar las filas y después leer los datos. Y lleva más tiempo obtener un dato individual dispersado en múltiples columnas comparado con acceder a registros agrupados en una columna.

Para qué se utiliza

La arquitectura de base de datos columnar ha sido llamada el futuro de la inteligencia de negocios por Business Insider porque permite el manejo de queries instantaneos de carácter analitico que casi todos los emprendimientos dependen crucialmente a la hora de tomar decisiones de negocios. Estas bases proveen acceso sencillo a los elementos más relevantes, lo cual justamente incrementa la velocidad de un query incluso en una base conteniendo millones de registros. Se cuenta con un análisis mucho más sencillo de la data en general.

Basada en líneas y se utiliza sobre todo cuando hay que realizar muchas transacciones rápidamente, en muchos campos de aplicación (por ejemplo, pero no exclusivamente, en la investigación) los datos pasan por evaluaciones continuas. Esto es mucho más rápido con sistemas basados en columnas: la razón de esto es que se requiere acceder menos al disco duro. Los datos de una categoría se almacenan muy próximos entre sí. Si se desea leer y evaluar un registro de datos, basta con cargar un bloque; no es necesario leer la base de datos completa.

Ejemplos de usos reales

Netflix

Netflix es uno de los principales valedores de las bases de datos NoSQL ya que utilizan Cassandra para su capa de persistencia. Desarrollada en primera instancia por Facebook y luego continuada por la Fundación Apache, Cassandra se caracteriza por ser distribuida, escalable y ofrecer un gran rendimiento... características esenciales para aguantar las tremendas cifras de Netflix: más de 1 millón de peticiones de escritura por segundo.

Apple

Hbase – (Maps, Siri, iAd, iCloud, and more, often related to Hadoop deployments)
Cassandra – (Maps, iAd, iCloud, iTunes, and more)

Black rock

Para ayudar a impulsar nuestra plataforma de gestión de inversiones Aladdin. En esta charla daré una visión general de nuestro uso de Cassandra, con énfasis en cómo gestionamos la multi-tenencia en nuestra infraestructura de Cassandra.

Spotify

Spotify utilizan Kafka para la recopilación de registros, Storm para el procesamiento de eventos en tiempo real, Crunch para la ejecución de trabajos map-reduce por lotes en Hadoop y Cassandra para almacenar los atributos del perfil del usuario y los metadatos sobre entidades como listas de reproducción, artistas, etc.

Motores de ejemplo

- Apache Cassandra
- Redshift
- MaríaDB ColumnStore
- Apache Hadoop Hbase

Bases de datos columnares en AWS

Amazon Web Services (AWS) proporciona una variedad de opciones de base de datos columnares para los desarrolladores. Puede operar su propio almacén de datos no relacional en columnas en la nube en Amazon EC2 y Amazon EBS, trabajar con proveedores de soluciones de AWS, o aprovechar los servicios de base de datos columnares totalmente gestionados. Amazon Redshift es un almacén de datos orientado a columnas, rápido y totalmente administrado a escala de petabytes que permite analizar todos los datos de forma sencilla y rentable utilizando las herramientas de inteligencia empresarial existentes.
Amazon Redshift consigue un almacenamiento eficiente y un rendimiento óptimo a través de una combinación de procesamiento paralelo de forma masiva, almacenamiento de datos en columnas y esquemas de codificación de compresión de datos muy específicos y eficientes.

Bases de datos columnares en Amazon EC2 o Amazon EMR

Los desarrolladores pueden instalar las bases de datos orientadas a columnas que elijan en Amazon EC2 y Amazon EMR, lo que significa que evitan la fricción del aprovisionamiento de la infraestructura, al tiempo que les permite acceder a diferentes motores de bases de datos columnares estándar.

Apache Cassandra

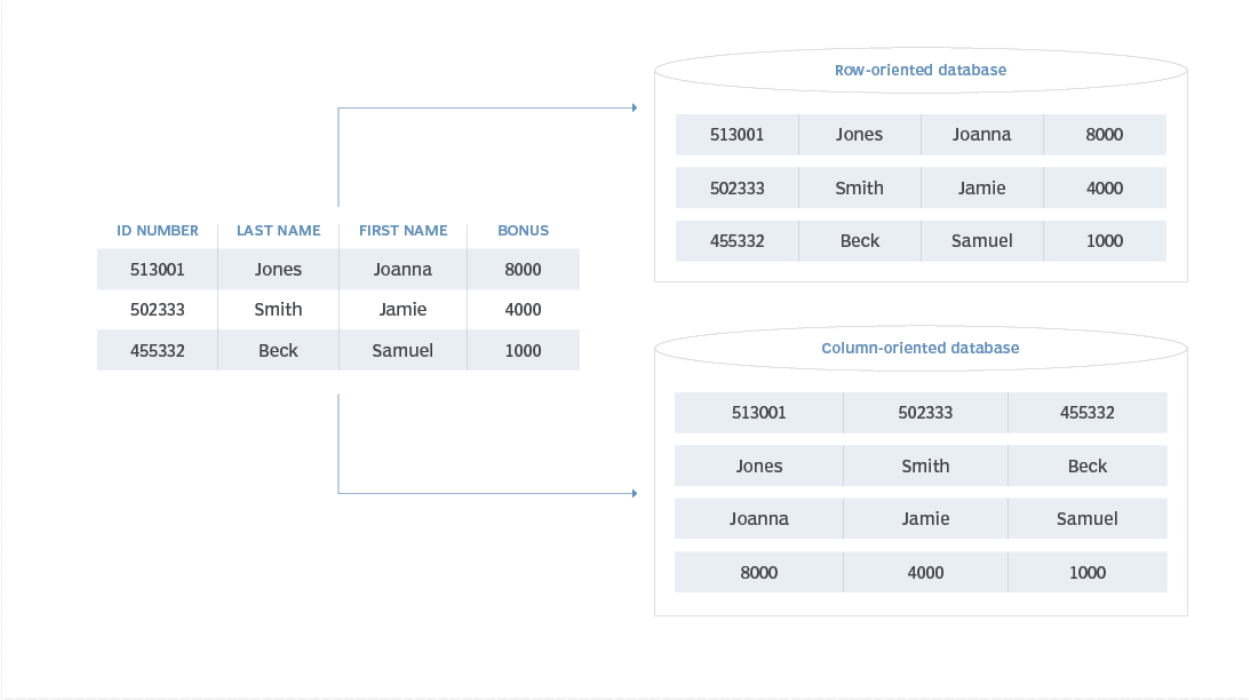
Cassandra es una base de datos orientada a columnas de código abierto diseñada para gestionar grandes cantidades de datos en muchos servidores comerciales. A diferencia de una tabla en una base de datos relacional, las diferentes filas en la misma tabla (familia de columna) no tienen que compartir el mismo conjunto de columnas.

La variedad de aplicaciones para las que puede emplearse Cassandra base de datos la configuran como una opción a tener en cuenta por las organizaciones que se interesan por el internet de las cosas y las aplicaciones relacionadas, el rastreo y la monitorización de la actividad de los usuarios en su interacción con los productos o servicios, la analítica social media y los motores de recomendación. Debido a sus posibilidades, resulta muy recomendable para empresas del sector retail que cuentan con e-commerce o quieren utilizarla para sus apps o catálogos, contribuyendo a mejorar el nivel de soporte.

Diferencias con otras DDBB

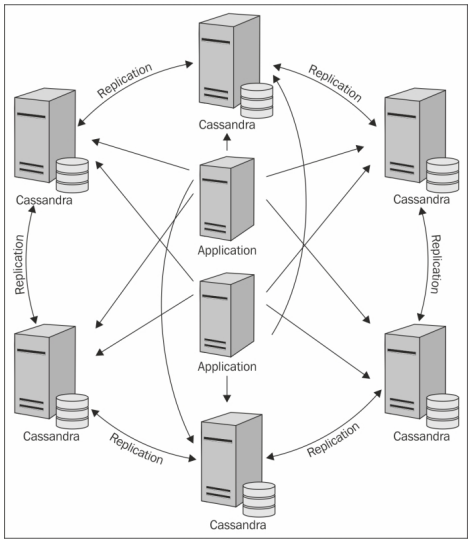
La diferencia entre una base de datos columnares y una base de datos relacional es que la base de datos relacional está optimizada para almacenar filas de datos, generalmente realizando operaciones transaccionales. En cambio, una base de datos en columnas está optimizada para lograr una recuperación rápida de columna de datos.
Las bases de datos tradicionales almacenan data secuencialmente de manera orientada a filas. En este caso, data similar no está junta o una al lado de la otra, incrementando el tiempo para acceder a información y a unidades de almacenamiento.

Diferentemente en las bases de datos columnares, los nombres de todos los empleados se encuentran en serie uno al lado del otro, todos los nombres en la columna “NOMBRE” y los nombres en la columna “DEPARTMENT” están almacenados uno detrás del otro. Esto simplifica el proceso de extracción de información que sea similar ya que la data guardada en la columna entera es agrupada y almacenada al mismo tiempo.



Escalabilidad

Las bases de datos columnares están diseñadas para reducir la escala utilizando clúster distribuido de hardware de bajo costo para aumentar el desempeño. En otras palabras, están orientadas a escalar principalmente de manera horizontal que vertical.



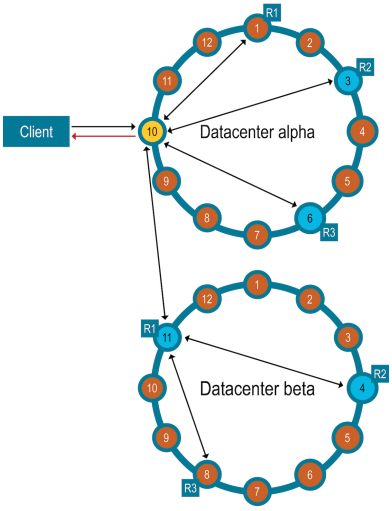
Las siguientes imágenes son usando Casandra con múltiples volúmenes Amazon Elastic Block Store (Amazon EBS). Estos volúmenes se utilizan con instancias Amazon EC2 y ofrecen una baja latencia para trabajar con grandes volúmenes de datos.

Figure 3: Single EBS Volume

Figure 4: Multiple EBS Volumes for Data

En una estructura distribuida de Cassandra todos los nodos juegan el mismo rol (no hay un nodo principal o máster).

Cassandra tiene nodos semillas (seed nodes) que son consultados por un nuevo nodo que quiere unirse a la red de Cassandra (una nueva instancia).
Cada nodo se encarga de una partición de datos.
Luego la estructura se puede expandir múltiples regiones.



Bibliografía

Bekker, Alex. 2018. “Cassandra Performance: The Most Comprehensive Overview You'll Ever See.” ScienceSoft. https://www.scnsoft.com/blog/cassandra-performance.

Cockcroft, Adrian, and Denis Sheahan. n.d. “Benchmarking Cassandra Scalability on AWS — Over a million writes per second | by Netflix Technology Blog.” Netflix TechBlog. Accessed November 8, 2022. https://netflixtechblog.com/benchmarking-cassandra-scalability-on-aws-over-a-million-writes-per-second-39f45f066c9e.

Mishra, Kinshuk, and Matt Brown. 2015. “Personalization at Spotify using Cassandra - Spotify Engineering.” Spotify Engineering. https://engineering.atspotify.com/2015/01/personalization-at-spotify-using-cassandra/.

“¿Qué es una base de datos columnar? – AWS.” n.d. AWS. Accessed November 8, 2022. https://aws.amazon.com/es/nosql/columnar/.