

Resumen: "Bigtable: un sistema de almacenamiento distribuido para datos estructurados"

Fecha de entrega: 30/08/22 3:58 pm

Estudiante: David José Espinoza Soto - 2016012024

Instrucciones: Del documento adjunto implementar un resumen de máximo 5 páginas.

Introducción

Bigtable es un sistema de almacenamiento distribuido para administrar datos estructurados que está diseñado para escalar a gran escala, actualmente es un sistema muy utilizado en proyectos de Google (+70). A comprobado ser una solución flexible y de alto rendimiento para requerimientos como manejo de grandes cantidades de datos o de latencia (que van desde el procesamiento masivo de back-end hasta servicio de datos en tiempo real).

Bigtable ha logrado varios objetivos como amplia aplicabilidad (wide applicability), escalabilidad (scalability), alto rendimiento (high performance) y alta disponibilidad (high availability).

Bigtable no admite un modelo de datos relacional completo; en su lugar, proporciona a sus usuarios un modelo de datos simple que admite el control dinámico sobre el diseño y el formato de los datos, y permite a los usuarios razonar sobre las propiedades de localidad de los datos representados en el almacenamiento subyacente. Los datos se dexan usando nombres de filas y columnas que pueden ser cadenas arbitrarias. Bigtable también trata los datos como cadenas no interpretadas, aunque los clientes suelen serializar varias formas de datos estructurados y semiestructurados en estas cadenas. Los usuarios pueden controlar la localidad de sus datos mediante elecciones cuidadosas en sus esquemas.

Modelo de Datos

Un Bigtable es un mapa ordenado multidimensional persistente, distribuido y disperso. El mapa está indexado por una clave de fila, una clave de columna y una marca de tiempo; cada valor en el mapa es una matriz de bytes no interpretada.

Rows

Cada llave de cada fila en una tabla son cadenas arbitrarias (tienen un tamaño de hasta 64 KB) . Cada lectura o escritura de datos bajo una sola llave en la fila es atómica (independientemente de la cantidad de columnas diferentes que se leen o escriben en la fila), esta decisión de diseño que facilita a los suarios razonar sobre el comportamiento del sistema en presencia de actualizaciones concurrentes en la misma fila. Bigtable mantiene los datos en orden lexicográfico por llave de fila. El rango de filas de una tabla se particiona dinámicamente. Cada rango de filas se denomina "tablet", que es la unidad de distribución y equilibrio de carga. Como resultado, las lecturas de rangos de filas cortos son eficientes y generalmente requieren comunicación con solo una pequeña cantidad de máquinas.

Column Families

Las claves de columna se agrupan en conjuntos denominados familias de columnas, que forman la unidad básica de control de acceso. Todos los datos almacenados en una familia de columnas suelen ser del mismo tipo (comprimimos juntos los datos de la misma familia de columnas). Se debe crear una familia de columnas antes de que los datos se puedan almacenar bajo cualquier clave de columna en esa familia; una vez que se ha creado una familia, se puede utilizar cualquier clave de columna dentro de la familia. Nuestra intención es que el número de familias de columnas distintas en una tabla sea pequeño (cientos como máximo) y que las familias rara vez cambien durante la operación. Por el contrario, una tabla puede tener un número ilimitado de columnas. El calificador es el nombre del sitio de referencia; el contenido de la celda es el texto del enlace. El control de acceso y la contabilidad tanto del disco como de la memoria se realizan a nivel de familia de columnas.

Timestamps

Cada celda de Bigtable puede contener varias versiones de los mismos datos; estas versiones están indexadas por una marca de tiempo. Las marcas de tiempo de Bigtable son números enteros de 64 bits. Pueden ser asignados por Bigtable, en cuyo caso representan "tiempo real" en microsegundos, o ser asignados explícitamente por aplicaciones cliente. Las aplicaciones que necesitan evitar colisiones deben generar marcas de tiempo únicas por sí mismas. Las diferentes versiones de una celda se almacenan en marca de tiempo decreciente o decreciente, de modo que las versiones más recientes se pueden leer primero. Para que la administración de datos versionados sea menos onerosa, admitimos dos configuraciones por familia de columnas que le indican a Bigtable que recopile automáticamente las versiones de las celdas.

El cliente puede especificar que solo se conserven las últimas n versiones de una celda, o que solo se conserven las versiones lo suficientemente nuevas (por ejemplo, solo se conserven los valores que se escribieron en los últimos siete días).

API

La API de Bigtable proporciona funciones para crear y eliminar tablas y familias de columnas. También proporciona funciones para cambiar los metadatos de grupos, tablas y familias de columnas, como los derechos de control de acceso.

Las aplicaciones cliente pueden escribir o eliminar valores en Bigtable, buscar valores de filas individuales o iterar sobre un subconjunto de datos en una tabla. Los clientes pueden iterar sobre varias familias de columnas y existen varios mecanismos para limitar las filas, las columnas y las marcas de tiempo producidas por un escaneo.

Bigtable admite transacciones de una sola fila, que se pueden usar para realizar secuencias atómicas de lectura, modificación y escritura en los datos almacenados en una clave de una sola fila. También permite que las celdas se utilicen como contadores de enteros. Finalmente, Bigtable admite la ejecución de secuencias de comandos proporcionadas por el cliente en los espacios de direcciones de los servidores.

Building Blocks

Bigtable se basa en varias otras piezas de la infraestructura de Google. Bigtable utiliza el sistema de archivos de Google distribuido (GFS) [17] para almacenar archivos de registro y datos. Un clúster de Bigtable generalmente opera en un grupo compartido de máquinas que ejecutan una amplia variedad de otras aplicaciones distribuidas, y los procesos de Bigtable a menudo comparten las mismas máquinas.

con procesos de otras aplicaciones. Bigtable depende de un sistema de administración de clústeres para programar trabajos, administrar recursos en máquinas compartidas, lidiar con fallas de máquinas y monitorear el estado de las máquinas. El formato de archivo SSTable de Google se usa internamente para almacenar datos de Bigtable. Una SSTable proporciona un mapa inmutable ordenado y persistente de claves a valores, donde tanto las claves como los valores son cadenas de bytes arbitrarias. Se proporcionan operaciones para buscar el valor asociado con una clave específica Para aparecer en OSDI 2006 3 y para iterar sobre todos los pares clave/valor en un rango de clave específico. Internamente, cada SSTable contiene una secuencia de bloques (por lo general, cada bloque tiene un tamaño de 64 KB, pero esto es configurable). Se utiliza un índice de bloque (almacenado al final de SSTable) para ubicar bloques; el índice se carga en la memoria cuando se abre SSTable. Se puede realizar una búsqueda con una sola búsqueda de disco: primero encontramos el bloque apropiado realizando una búsqueda binaria en el índice en memoria y luego leyendo el bloque apropiado del disco. Opcionalmente, una SSTable se puede mapear completamente en la memoria, lo que nos permite realizar búsquedas y escaneos sin tocar el disco.