[Opcional] Bases de datos de columnas anchas

Estado: Traducido automáticamente del Inglés

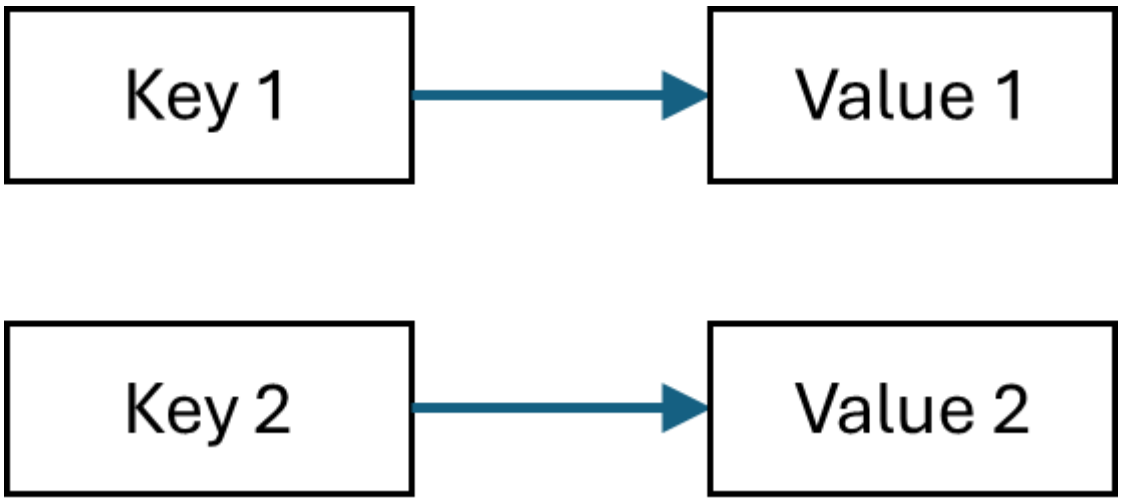
Traducido automáticamente del Inglés

Información:

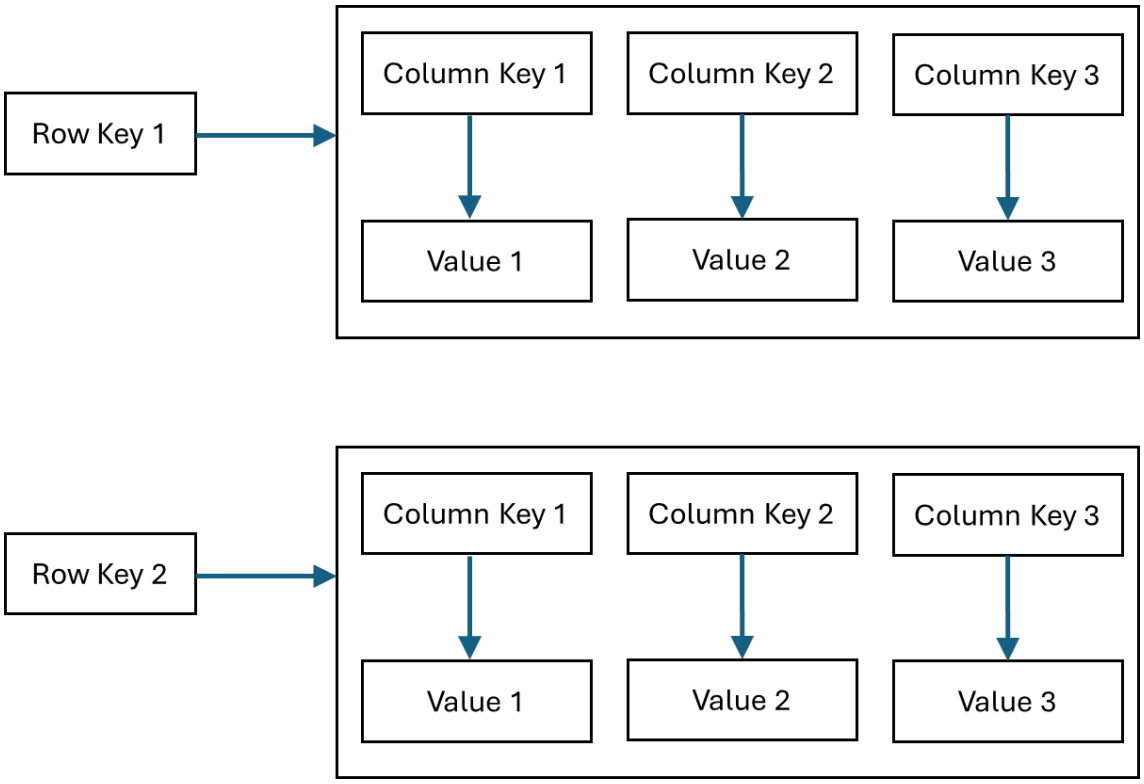
Este elemento incluye contenido que aún no se tradujo a tu idioma preferido.

Una base de datos de columnas anchas es un tipo de base de datos NoSQL que es como una combinación de una base de datos relacional y un almacén de documentos. Estructura la parte de valor de un par clave-valor en más pares clave-valor.

En una base de datos clave-valor, los datos se representan de la siguiente manera:

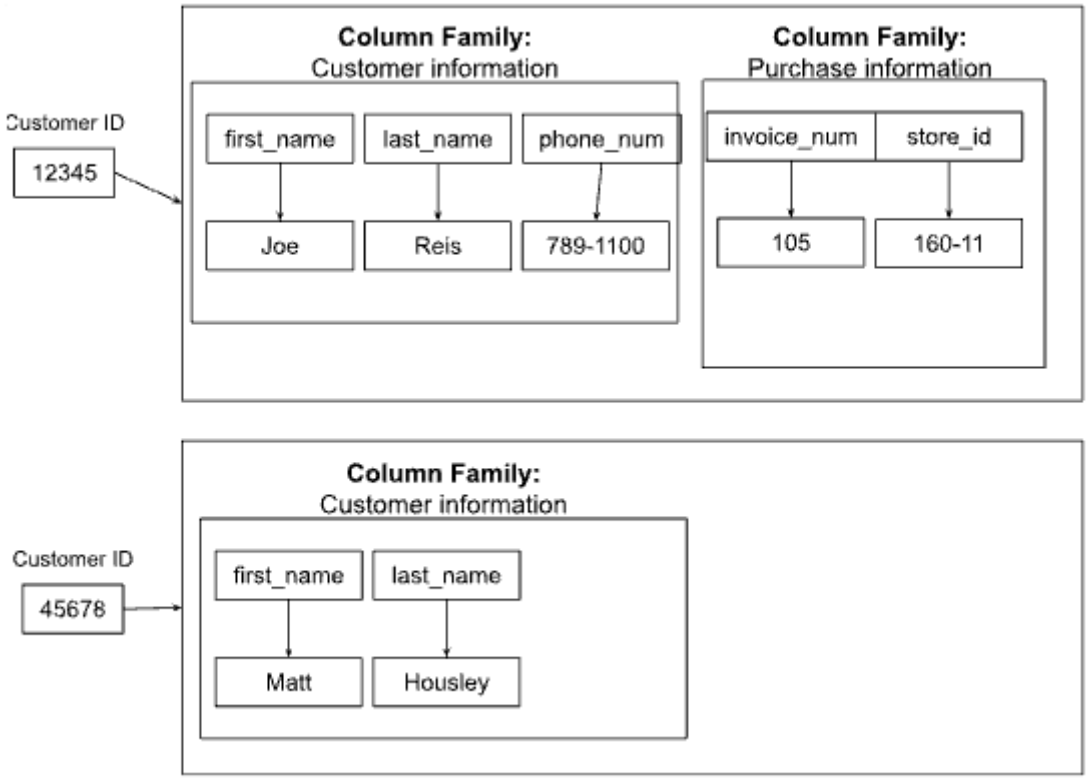


Por otro lado, una base de datos de columnas anchas almacena los datos en tablas que son como mapas clave-valor bidimensionales:



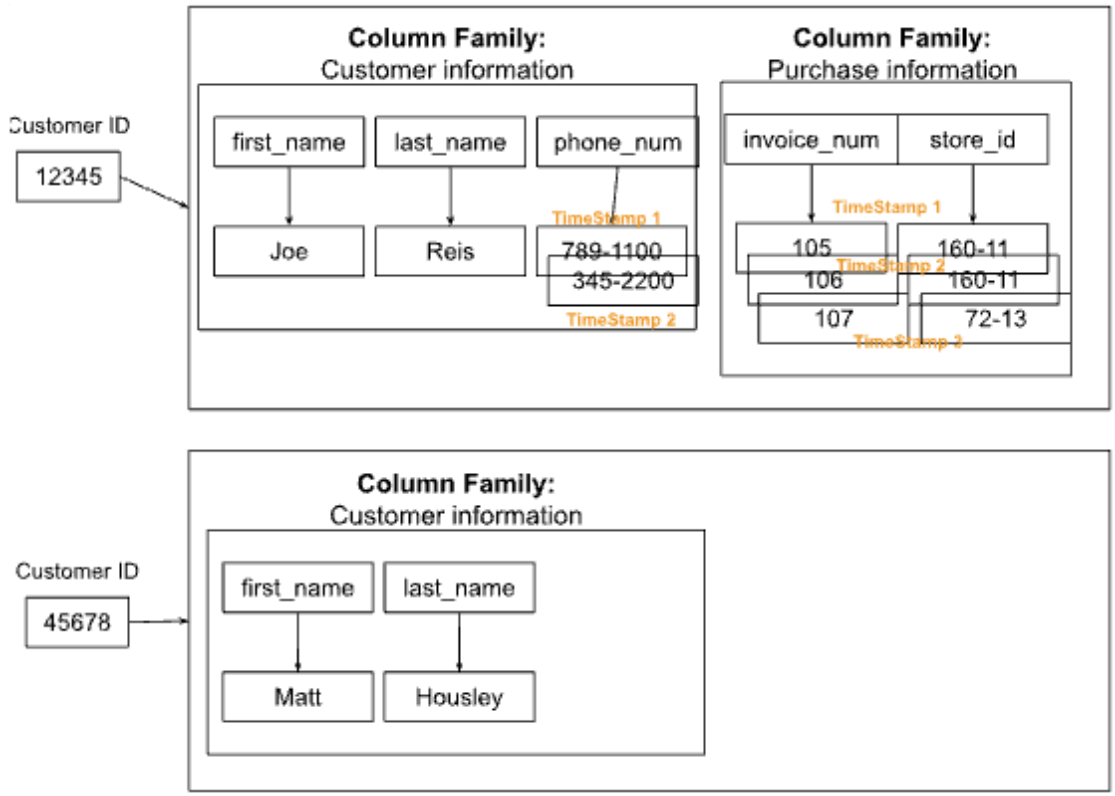
Cada fila suele describir una única entidad y se identifica por su clave de fila. Dentro de una fila, los datos relacionados se modelan en familias de columnas que contienen columnas con nombres de columna únicos. Los datos reales se almacenan en celdas, que se identifican unívocamente por la combinación de clave de fila, familia de columnas y nombre de columna (por ejemplo, clave de fila 1, familia de columnas 4, clave de columna 3).

Por ejemplo, supongamos que desea almacenar información sobre clientes y compras en una base de datos de columnas anchas como ésta:



Podría utilizar customer_id 12345 como clave de fila que apunte a dos familias de columnas: información de cliente con 3 columnas (nombre, apellido y número de teléfono) e información de compra con 2 columnas (número de factura e ID de tienda).

Cada valor de celda está versionado e identificado unívocamente por su número de versión, normalmente su marca de tiempo. Digamos que este cliente cambió su número de teléfono o hizo más de una compra, entonces estas celdas contendrán múltiples valores, cada uno identificado por su marca de tiempo.



Supongamos que tiene otro cliente, con customer_id 45678, del que no conoce el número de teléfono y que no ha realizado ninguna compra. Entonces su clave de fila apuntará a la familia de columnas de información del cliente que contiene sólo las columnas nombre y apellido del cliente sin la columna número_teléfono. Como este cliente no ha realizado ninguna compra, no es necesario que apunte a una familia de columnas de información de compras. Cuando esta información esté disponible, podrá añadirla a la base de datos de columnas anchas. Dado que la base de datos de columna ancha no impone un esquema de tabla estricto, la adición de columnas es muy flexible y sólo se escribe una columna si hay datos para ella.

Una base de datos de columna ancha suele almacenar las familias de columnas por separado en el disco. Así, almacenará la información del cliente separada de la información de la compra. Luego, los datos dentro de una familia de columnas se almacenan de forma orientada a filas. Así, para la información de compra, almacenaría todos los datos de una clave de fila específica uno al lado del otro en el disco, almacenando todos los valores de invoice_num juntos, luego los valores de store_id juntos, antes de pasar a la siguiente clave de fila, y así sucesivamente.

Algunos ejemplos de bases de datos de columnas anchas son HBase

, BigTable, Apache Cassandra y Amazon keyspaces

(para Apache Cassandra)

Recursos (lecturas adicionales opcionales)

- Bases de datos de columnas anchas

 https://www.scylladb.com/glossary/wide-column-database/

 Cassandra: La guía definitiva

 Introducción al diseño de esquemas de HBase

 Apache Cassandra VS Apache HBase

-

[Opcional] Formato Parquet

Estado: Traducido automáticamente del Inglés

Traducido automáticamente del Inglés

Información:

Este elemento incluye contenido que aún no se tradujo a tu idioma preferido.

Visión general

El almacenamiento orientado a columnas es adecuado para cargas de trabajo analíticas en las que se desea aplicar operaciones de agregación sobre columnas. Pero no es adecuado para leer o escribir/actualizar filas. Por otro lado, el almacenamiento orientado a filas es adecuado para cargas de trabajo transaccionales que requieren que la lectura y la escritura se realicen con baja latencia. Pero no es adecuado para cargas de trabajo analíticas eficientes.

Parquet y ORC (optimized row columnar) son formatos de archivo que combinan ambos enfoques siguiendo un planteamiento híbrido que trata de obtener lo mejor de ambos mundos. El enfoque híbrido se basa en la partición de filas en grupos donde cada grupo de filas se almacena en un formato por columnas.

Product SKU	Price	Quantity	Customer ID
045865	40	1	67t
902348	23	4	56t
125893	45	2	87q
456829	50	3	98q

Row group 1

Row group 2

Hybrid approach

045865	902348	40	23	1	4	67t	56t
125893	456829	45	50	2	3	87q	98q

Aunque son similares, ORC es generalmente menos popular que Parquet (ORC fue muy popular para su uso con Apache Hive - un almacén de datos), y goza de algo menos de apoyo en las herramientas modernas del ecosistema de la nube. Así que vamos a centrarnos en Parquet.

Un poco más de detalle sobre Parquet

Con un archivo Parquet, los datos se dividen horizontalmente en grupos de filas, donde cada grupo de filas tiene un tamaño predeterminado de 128 megabytes. Un grupo de filas consta de un fragmento de columna para cada columna del conjunto de datos. Cada fragmento de columna se divide en páginas, cada una de las cuales contiene los valores codificados para ese fragmento de columna, metadatos como el mínimo, el máximo y el recuento de los valores, junto con otros datos (niveles de repetición y definición) utilizados para reconstruir la estructura anidada de los datos. Parquet puede utilizarse para almacenar datos tabulares, así como datos anidados (es decir, datos semiestructurados como JSON).

Otra ventaja de Parquet es su portabilidad. Por lo tanto, obtendrá un mejor rendimiento con Parquet al interoperar con herramientas externas, a diferencia de los formatos columnares de almacén de datos en nube patentados, que requieren deserialización y re-serialización para ser compatibles.

Para saber más sobre el formato Parquet, echa un vistazo a este vídeo

de Databricks.

Recursos (lecturas adicionales opcionales)

- Parquet, ORC y Avro

 Documentación de Parquet

-