INTRODUCCIÓN A LA BASE DE DATOS APACHE CASSANDRA

1° Desarrollo Aplicaciones Multiplataforma IES PLAYAMAR

Juan Manuel Carmona Ruiz

CURSO: 2022 – 2023

ÍNDICE

1.INTRODUCCIÓN

Las bases de datos NoSQL (acrónimo de Not Only SQL) son un tipo de bases de datos diseñadas para manejar grandes volúmenes de datos y ofrecer una escalabilidad horizontal más fácil que las bases de datos relacionales tradicionales lo que significa que la capacidad de la base de datos para manejar grandes volúmenes de datos se puede aumentar al agregar más servidores o nodos al sistema, en lugar de aumentar la capacidad de los servidores existentes.

A diferencia de las bases de datos relacionales, que utilizan una estructura de tabla rígida y un lenguaje de consulta SQL, las bases de datos NoSQL permiten la flexibilidad en el esquema y utilizan modelos de datos no tabulares como documentos, gráficos y clave-valor.

Entre las bases de datos NoSql podemos encontrar diferentes tipos clasificándolos entre:

- Bases de datos de columnas
- Bases de datos de grafos
- Bases de datos orientadas a documentos
- Bases de datos clave-valor

2.CARACTERÍSTICAS PRINCIPALES

2.1. Definición

Apache Cassandra se encuentra entre las bases de datos clave-valor que se caracterizan por ser similares a un diccionario, ya que cada una de las claves (palabra del diccionario) tiene un valor o conjunto de valores para esa clave (equivalente a el significado/s que tendría una palabra en dicho diccionario).

Esta base de datos fue lanzada en 2008 y su creación se debe inicialmente por Facebook ya que debían desarrollar una herramienta para poder cubrir sus necesidades de almacenamiento de datos a gran escala. Tras su éxito la conocida red social decidió traspasarla a la “Fundación Apache”, convirtiéndola en una herramienta de código abierto que a día de hoy se sigue manteniendo.

2.2. Teorema CAP

Para poder comprender dónde debemos situar la base de datos Apache Cassandra entre los sistemas de bases de datos es necesario conocer el teorema CAP:

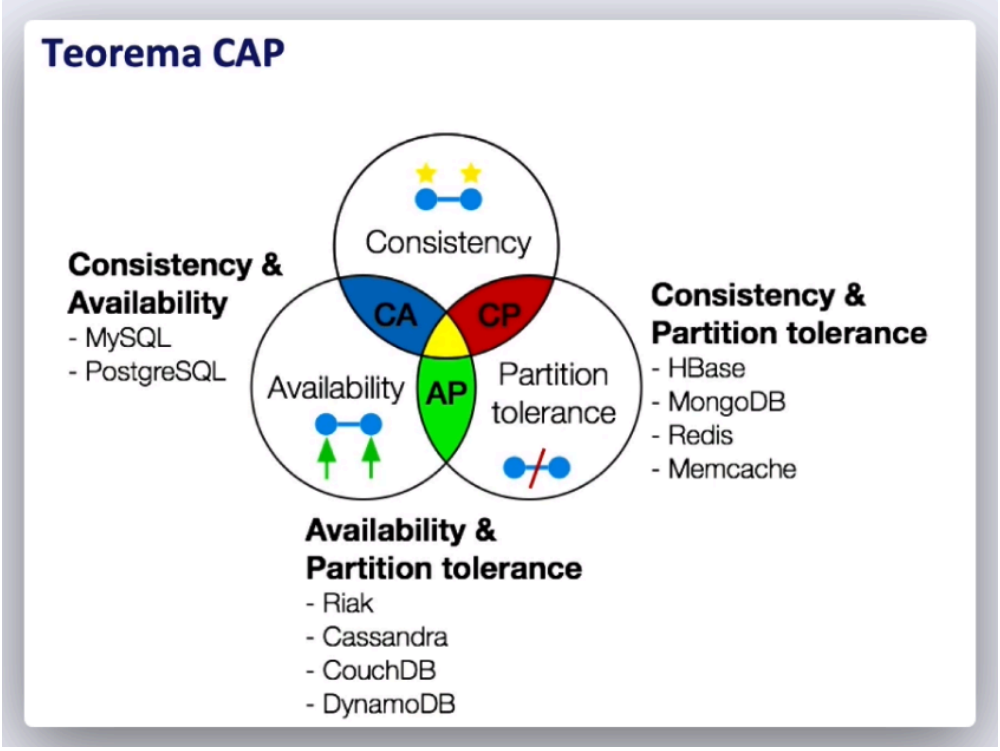

Figura 1. Teorema CAP

Este teorema se basa en que un sistema distribuido (aquel en el que sus datos no se encuentran físicamente en un solo servidor, sino que se encuentran repartidos entre distintas máquinas) no puede conseguir las 3 siglas que los describen sino solo 2 de las siguientes:

Consistency (Consistencia)	Se refiere a la necesidad de que los datos en todos los nodos de una base de datos distribuida sean iguales y estén actualizados en todo momento. Esto significa que, si un nodo actualiza los datos, todos los demás nodos deben tener esos mismos datos actualizados. Una base de datos consistente garantiza que todas las transacciones se ejecuten en el mismo orden en todos los nodos.

Availability (Disponibilidad)	Se refiere a la capacidad de los usuarios para acceder a los datos en cualquier momento, incluso si un nodo de la base de datos falla. Una base de datos disponible significa que los nodos de la base de datos deben estar en línea y accesibles para los usuarios en todo momento, lo que garantiza que las solicitudes de datos se puedan manejar y procesar en tiempo real.
Partition tolerance (Tolerancia a particiones)	Se refiere a la capacidad de una base de datos distribuida para funcionar incluso si los nodos en la red están separados o no pueden comunicarse entre sí. Una base de datos tolerante a particiones significa que la base de datos seguirá funcionando incluso si algunos nodos de la red fallan o no pueden comunicarse, lo que garantiza que la base de datos siempre esté disponible para los usuarios.

Tabla 1. Descripción CAP

Como se puede observar en la figura 1, Apache Cassandra se encuentra situada en la intersección entre la disponibilidad y la tolerancia a particiones “sacrificando” el apartado de consistencia.

2.3. Ventajas y desventajas.

Apache Cassandra usa el lenguaje CQL, similar a SQL y se caracteriza porque escala linealmente lo que quiere decir que en función de la cantidad de nodos que se añadan, la cantidad de operaciones por segundo que va a realizar es equivalente a la cantidad de nodos añadida.

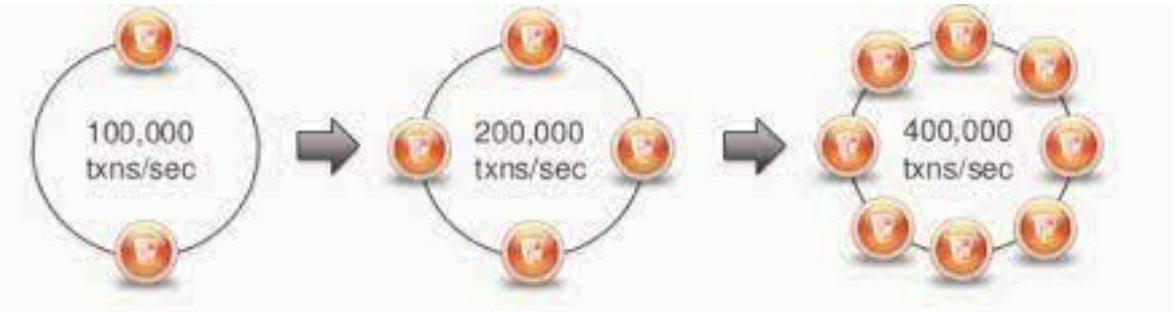

Figura 2. Escala lineal de nodos en Apache Cassandra

Como se puede observar en la figura 2, al duplicar la cantidad de nodos involucrados en la base de datos, la cantidad de operaciones por segundo se ven también duplicadas.

El patrón que sigue es peer-to-peer (P2P) lo que quiere decir que todos los dispositivos tienen el mismo nivel de importancia, no como en el modelo maestro-esclavo, por lo que no es necesario depender de un servidor central para compartir y recibir información, sino que los dispositivos se conectan entre sí mejorando la velocidad y la eficiencia de la transferencia de datos.

La principal ventaja que presentan este tipo de bases de datos es que trabajan en memoria. Esto significa que los datos se acceden y se procesan más rápidamente, ya que el acceso a la memoria es mucho más rápido que el acceso a los discos duros. También puede mejorar el rendimiento de la base de datos al reducir los tiempos de latencia.

Sin embargo, trabajar en memoria tiene algunas desventajas. Primero, la cantidad de datos que se pueden almacenar en la memoria principal es limitada y suele ser más costosa que el almacenamiento en disco. Además, si se produce un fallo del sistema o se pierde la energía, los datos en memoria se pierden, ya que la memoria no es persistente.

Además, la creación de nuevos nodos es un proceso complejo que conlleva tiempo al tener que sincronizar las diferentes máquinas implicadas.

3.PROCESO DE INSTALACIÓN

El proceso de instalación que se va a describir a continuación es para aquellos ordenadores que utilicen un sistema operativo Windows 10 basado en x64. La descarga de las versiones de las aplicaciones de Java y Python son importantes ya que sino no

funcionará. La documentación gráfica de este proceso se encuentra en el Anexo 1 del presente documento.

Tras la instalación de estos programas necesitaremos ver la configuración avanzada del sistema que podemos encontrar en “Propiedades de sistema”, una vez nos encontremos sobre esta ventana tendremos que direccionarnos a las “Variables de entorno”.

Clicamos sobre la variable Path y posteriormente en el botón “Nueva…”.Nos aparecerá una ventana con el título “Editar variable de entorno”, clicamos sobre el boton “Nuevo” de la parte superior derecha y después en examinar. Tendremos que buscar la carpeta donde hemos instalado Python y la carpeta donde se encuentra Apache Cassandra. Deberán aparecernos las rutas absolutas de los archivos, le damos a aceptar y cerramos la ventana.

Ahora tendremos que añadir una nueva variable del sistema, para ello clicaremos sobre “Nueva…” de la ventana “Variables de entorno” que se encuentra en la parte inferior a la izquierda de “Editar…”. Como nombre de la variable introduciremos “JAVA_HOME” y para el valor de la variable examinaremos los directorios con “Examinar directorio…” y seleccionamos la carpeta en la que se encuentra el jdk1.8.0_u251. Aceptamos en todas las ventanas y nos dirigimos a la carpeta donde tenemos instalado apache-cassandra.

Nos introducimos hasta la carpeta bin y en la parte superior donde aparece la ruta absoluta, escribimos “CMD” lo que nos abrirá la consola de comandos del sistema estando situados en la carpeta bin. Introducimos el comando “Cassandra” y cuando aparezca “Startup complete” habremos establecido conexión con la base de datos.

Una vez llegados a este punto podremos trabajar con Apache-Cassandra desde el cqlsh donde introduciremos en una consola los comandos que queramos ejecutar. Sin embargo, también es posible utilizar la interfaz gráfica devCenter.

Una vez instalada, nos dirigimos a ”File” situado en la parte superior izquierda “New” -> “Connection” en “Connection name” introducimos el nombre de nuestra conexión, el host que vamos a utilizar en nuestro caso “Localhost” y el puerto que viene por defecto que es el 9042, una vez realizados estos pasos, podremos probar la conexión y al configurarse con éxito aparece en el apartado de conexiones el nombre de nuestra conexión seguridad de [1/1 Connected].

Al reiniciar nuestro equipo es necesario saber que la conexión se apagará y que tendremos que volver a realizar la conexión desde la carpeta bin y la consola de comandos, donde deberemos introducir el comando “Cassandra -f”.

3.1.Instalación de Java 8_u251

En primer lugar, es necesaria la instalación del jdk1.8.0_ 251 que se puede encontrar en :

https://www.oracle.com/java/technologies/javase/8u251-relnotes.html

3.2. Instalación de Python 2.7

Para la instalación de Python vamos a necesitar la versión 2.7.18 que se puede encontrar en la página de Python:

https://www.python.org/downloads/release/python-2718/

3.3. Instalación de Apache Cassandra 3.11.10

Para la instalación de Apache Cassandra vamos a necesitar la versión 3.11.10 que podemos encontrar en:

https://archive.apache.org/dist/cassandra/3.11.10/

3.4. Interfaz gráfica devCenter

Para la instalación de la interfaz gráfica nos dirigiremos a:

https://www.datastax.com/tools/devcenter

4.DOCUMENTACIÓN DE APACHE CASSANDRA

La documentación de Apache Cassandra se puede consultar desde el siguiente enlace:

https://cassandra.apache.org/doc/latest/cassandra/cql/dml.html