Herramientas Big Data para almacenar datos

Hace un par de décadas un terabyte era una cantidad de información casi inimaginable. Hoy por hoy, sin embargo, muchos centros de datos se miden en petabytes, incluso, en zetabytes. Para almacenar una cantidad tan abrumadora de datos hacen falta herramientas con una capacidad enorme. En este contexto juegan un papel clave las bases de datos.

Las bases de datos son un compendio de datos relacionados con un mismo contexto y almacenados masivamente para su posterior uso. La mayoría de bases de datos ya están en formato digital, lo que permite tratarlas informáticamente y acceder a ellas en menos tiempo. Pueden albergar tanto información estructurada como no estructurada. En informática, por su forma de estructurar la información y el lenguaje que utilicen, se clasifican a grandes rasgos en bases de datos SQL y NoSQL.

Bases de datos SQL

Las bases de datos SQL (Structured Query Language o lenguaje de consulta estructurada) usan un lenguaje declarativo de acceso a bases de datos relacionales que permite efectuar consultas para almacenar, modificar y extraer información de forma sencilla.

La principal característica es que las bases de datos SQL siguen un estándar, tanto en la forma en la que se diseñan como en el modo en el que almacenan la información y en el que deben consultarse.

Todas las bases de datos SQL cumplen con las propiedades ACID (Atomicidad de operaciones, Consistencia de datos, Aislamiento de operaciones concurrentes, y Durabilidad de los datos). Algunos ejemplos: DB2, Oracle, SQLite….

7 Herramientas Big Data para tu empresa

A continuación, te mostramos una selección de herramientas open source que ofrecen soluciones para la explotación de software de Big Data en todos sus procesos: almacenamiento, procesamiento y análisis, que seguramente te serán útiles en tus proyectos.

1. Hadoop

No se puede hablar de Big Data sin hablar de la veterana Apache Hadoop. Esta herramienta Big Data open source se considera el framework estándar para el almacenamiento de grandes volúmenes de datos; se usa también para analizar y procesar, y es utilizado por empresas como Facebook y Yahoo!.

La biblioteca Hadoop utiliza modelos de programación simples para el almacenamiento y procesamiento distribuido de grandes conjuntos de datos en clusters, dando redundancia para no perder nada y, al mismo tiempo, aprovechando muchos procesos a la vez.

Dispone de un sistema de archivos distribuido en cada nodo del cluster: el HDFS (Hadoop Distributed File System), y se basa en el proceso de MapReduce de dos fases.

Soporta diferentes sistemas operativos y también se usa frecuentemente sobre cualquiera de las principales plataformas en la nube, como Amazon EC2/S3 o Google Cloud.

2. MongoDB

Dentro de las bases de datos NoSQL, probablemente una de las más famosas sea MongoDB. Con un concepto muy diferente al de las bases de datos relacionales, se está convirtiendo en una interesante alternativa para almacenar los datos de nuestras aplicaciones.

MongoDB es una base de datos orientada a documentos (guarda los datos en documentos, no en registros). Estos documentos son almacenados en BSON, que es una representación binaria de JSON.

A pesar de que las bases de datos NoSQL no tienen una extensa variedad de uso, MongoDB tiene un ámbito de aplicación más amplio en diferentes tipos de proyectos: es especialmente útil en entornos que requieran escalabilidad. Con sus opciones de replicación y sharding, podemos conseguir un sistema que escale horizontalmente sin demasiados problemas.

3. Elasticsearch

Elasticsearch es una potente herramienta para la búsqueda entre grandes cantidades de datos, especialmente cuando los datos son de tipo complejo.

Nos permite indexar y analizar en tiempo real un gran volumen de datos y hacer consultas sobre ellos. Un ejemplo de uso son las consultas de texto completo; al estar los datos indexados, los resultados se obtienen de forma muy rápida. En el IIC utilizamos esta herramienta para indexar datos dentro de nuestras soluciones de entorno digital.

A diferencia de otros sistemas parecidos, no necesita declarar un esquema de la información que añadimos, no sabemos exactamente qué forma van a tener los datos.

Con Elasticsearch podemos hacer búsquedas de texto complicadas, visualizar el estado de nuestros nodos y escalar sin demasiadas necesidades, si se diera el caso de que necesitáramos más potencia.

4. Apache Spark

Apache Spark es un motor de procesamiento de datos de código abierto realmente rápido.

Creado por Matei Zaharia en la Universidad de Berkeley, se considera el primer software open source que hace la programación distribuida (muy en esencia, consiste en distribuir el trabajo entre un grupo de ordenadores, “cluster”, que trabajan como uno) realmente accesible a los científicos de datos.

Se pueden programar aplicaciones usando diferentes lenguajes como Java, Scala, Python o R. pudiendo ser, según el programa, hasta 100 veces más rápido en memoria o 10 veces más en disco que Hadoop MapReduce.

5. Apache Storm

Apache Storm es un sistema de computación distribuida en tiempo real orientado a procesar flujos constantes de datos, por ejemplo, datos de sensores que se emiten con una alta frecuencia o datos que provengan de las redes sociales, donde a veces es importante saber qué se está compartiendo en este momento.

Aunque Hadoop sea un gran sistema para el procesado de un gran volumen de datos, no está pensado para hacerlo en tiempo real, ya que tiene una alta latencia. Apache Storm está siendo una revolución para procesar grandes cantidades de información en tiempo real, ya que es capaz de procesar millones de mensajes por segundo. En el IIC utilizamos Apache Storm para nuestra herramienta Lynguo, que requiere esta tecnología Big Data para procesar en tiempo real los comentarios de las redes sociales para su monitorización y análisis.

Apache Storm puede ser utilizado para procesar los logs de nuestras aplicaciones para ver el uso que se hace de los distintos servicios y gestión de errores; para extraer información de redes sociales a través de sus APIs y analizar un fenómeno en tiempo real; recoger y procesar datos de sensores; buscadores verticales, web analytics, etc.

6. Lenguaje R

R es un lenguaje de programación y entorno de software para cálculo estadístico y gráficos. El lenguaje R es de los más usados por los estadistas y otros profesionales interesados en la minería de datos, la investigación bioinformática y las matemáticas financieras.

R se parece más al lenguaje de las matemáticas que a otros lenguajes de programación, lo que puede ser un inconveniente para los programadores a la hora de elegir programar en R para temas de Big Data. Lo que está claro es que si eliges usar R podrás disponer de una gran cantidad de librerías creadas por la comunidad de R y otras tantas herramientas de altísima calidad (por ejemplo, RStudio).

7. Python

Python es un lenguaje avanzado de programación con la ventaja de ser relativamente fácil de usar para usuarios que no estén familiarizados con la informática de manera profesional, pero que necesitan trabajar con análisis de datos (estadistas, biólogos, físicos, lingüistas…).

Es una herramienta para Big Data muy eficiente, en parte debido a la gran comunidad existente, por lo que Python dispone de muchas librerías ya hechas por otros usuarios.

**Cómo seleccionar las mejores herramientas big data para análisis**

Para seleccionar las mejores herramientas big data para análisis es fundamental entender los requisitos de procesamiento de datos tanto transaccionales como analíticos de nuestros sistemas y tras esto, elegir en consecuencia.

Coherencia: la clave de la mejor elección de herramientas big data

La gestión de datos operativos o transaccionales busca garantizar la baja latencia en los tiempos de respuesta y en la gestión de una variedad de solicitudes simultáneas. Aunque, en ocasiones, la analítica en tiempo real puede estar involucrada, generalmente las consultas analíticas suelen quedar limitadas a un pequeño conjunto de variables relevantes para los procesos de toma de decisiones más inmediatos.

Es habitual que dicha información se muestre en un tablero simple que permita a los usuarios de negocio ejecutar informes, estándar o personalizados, en función de sus propias necesidades y nivel de experiencia.

Sin embargo, independientemente del tipo de motivación que inicie el proceso de reporting, una de las características más importantes de una transacción de datos es la confiabilidad. Es necesario mantener la coherencia transaccional y ello depende de elegir la solución correcta para los datos de la organización.

Las herramientas de análisis de Big Data han surgido para cargas de trabajo interactivas en tiempo real y análisis complejos y retrospectivos de conjuntos de datos más grandes. Las diferencias entre los dos podrían sintetizarse de la siguiente forma:

Los sistemas NoSQL, como las bases de datos de documentos y los almacenes, son soluciones comunes para bases de datos operacionales rápidas y escalables. Con una base de datos NoSQL apropiada, las transacciones se pueden procesar rápidamente, y el sistema puede gestionar muchas transacciones pequeñas al mismo tiempo durante los períodos de máxima actividad. Las transacciones por segundo se consideran un indicador de rendimiento más relevante que el tiempo de respuesta. Los sistemas transaccionales están optimizados para operaciones y transacciones cortas, atómicas, repetitivas, orientadas a la selección. Pueden ajustarse con gran precisión para operaciones de uso frecuente, aunque son bastante dependientes del almacenamiento en caché, una gran cantidad de recursos compartidos y rutas de código prescritas, algo que hay que tener en cuenta al seleccionar herramientas big data.

Las bases de datos de procesamiento masivo paralelo y MapReduce, donde se incluiría Hadoop, son soluciones clave en el espacio analítico. Los sistemas analíticos proporcionan riqueza funcional, velocidad de procesamiento y facilidad de uso. Por lo general, tienen mucha capacidad y permiten, además, mover datos rápidamente cuando sea necesario, aunque hay que tener en cuenta que su diseño está pensado para reducir el movimiento de datos en general. Se caracterizan por depender de pocas estructuras compartidas y por hacer posible que todas sus funciones puedan integrarse en el servidor y ampliarse para cumplir los requisitos cambiantes del usuario final.

Confiar en un solo sistema de base de datos para manejar ambos tipos de actividad requiere mucha mano de obra para TI, ya que los sistemas de bases de datos convencionales demuestran una gran variabilidad en el rendimiento cuando se les pide manejar cargas de trabajo analíticas y transaccionales.

Por supuesto, no todas las herramientas big data de análisis satisfacen todas las necesidades posibles, lo que significa que, a nivel empresarial, la mayoría de las organizaciones terminan usando sistemas complementarios para satisfacer todas sus necesidades de carga de datos.

**Tratamiento de datos de big data en en el Internet de las Cosas**

Con el Internet de las cosas, objetos que normalmente no estarían conectados a Internet pero que tenían potencial para obtener y procesar datos, están ahora equipados con sensores y chips que facilitan la recolección de datos. Y esa información de los dispositivos IoT se acumula en el Big Data.

El tratamiento de datos de ese big data de forma eficiente permite:

Tomar decisiones más inteligentes.

Desarrollar mejores productos.

Obtener información más profunda.

Tener un mayor conocimiento.

El marco arquitectónico Big Data para el tratamiento de datos

Una arquitectura estructurada en capas es la mejor forma de solucionar la complejidad que implica Big Data en cuanto al tratamiento de datos. Cada una de estas capas realiza una función particular, por lo que los datos se van canalizando en función de los requisitos del sistema de procesamiento por lotes o del sistema de procesamiento de flujo.

Para garantizar un flujo seguro de datos, la arquitectura Big Data, debe, al menos constar de las siguientes capas:

Capa de ingestión de datos: esta capa es el primer paso para que los datos procedentes de una variedad de fuentes comiencen su recorrido. Aquí los datos se clasifican los datos en función de su prioridad, lo que hace que el flujo de datos sea uniforme en capas adicionales.

Capa de recopilación de datos: en esta capa, se presta más atención al transporte de datos desde la capa de ingestión al resto de la canalización de datos. Para apoyar el desarrollo de las capacidades analíticas, aquí los componentes están desacoplados.

Capa de procesamiento de datos: al llegar a este nivel, el foco se pone en procesar los datos que han sido recopilados en la capa anterior. Es el primer punto donde puede realizarse la analítica.

Capa de almacenamiento de datos: a este nivel comienzan los grandes retos que big data plantea al tratamiento de datos ya que, el tamaño de los datos con los que se trata aumenta, por lo que es necesario encontrar una solución de almacenamiento eficiente.

Capa de consulta de datos: en esta es la capa se lleva a cabo un procesamiento analítico sólido. El objetivo principal en este punto es reunir el valor de los datos para que sean más útiles para la siguiente capa.

Capa de visualización de datos: éste se considera el nivel más importante de la arquitectura, ya que es en esta capa donde los usuarios de los canales de datos pueden extraer el valor de los datos. Aquí el dato ya es conocimiento y el usuario que accede a él tiene la posibilidad de hacer hallazgos que le conduzcan a la innovación, la toma de una decisión acertada o le den una pista nueva para seguir profundizando en algún concepto en su investigación.