Lección 5.6 Otras arquitecturas y herramientas:

1. Introducción

- **Hadoop/MapReduce** está optimizado para procesamiento y análisis secuencial (Batch Processing, recordemos el tema 3) de grandes lotes de datos Big Data (5v's)
- Aunque soporta la mayoría de aplicaciones de BI para Big Data, tiene algunas limitaciones hacen que esta arquitectura no sea la más adecuada en algunos escenarios.
    - No soporta el procesamiento y análisis de Big Data en tiempo real: Los procesos Map/Reduce se ejecutan en minutos u horas.
    - No está optimizada para la búsqueda y actualización de unos pocos registros de forma aleatoria: no ofrece soporte para la gestión de transacciones
- Para afrontar las limitaciones de Hadoop/MapReduce disponemos de otras arquitecturas y herramientas
    - Arquitecturas para el procesamiento y análisis de Big Data en tiempo real
    - SGBDR extendidos con tecnología MPP
    - Bases de datos NoSQL

A continuación vamos a estudiar las características de estas arquitecturas, la adecuación de las mismas a cada posible escenario y algunos ejemplos de herramientas que las implementan.

2. Big Data en tiempo real

Como ya vimos en el módulo 3 cuando estudiamos las fuentes de datos Big Data y su naturaleza la característica de Velocidad tiene ciertas implicaciones importantes.

- La característica de Velocidad en Big Data implica
    - La necesidad de procesar datos que se generan y distribuyen en tiempo real o "Streaming" (además de procesar datos bien empaquetados en archivos, datos en lotes)
    - Algunos datos pueden perder valor a medida que pasa el tiempo (ej. Datos Financieros, datos de fraude en tarjetas de crédito,...), por lo que es necesario procesarlos y analizarlos en el menor tiempo posible.
- Recientemente han surgido arquitecturas para dar soporte a estos requerimientos

- Algunos ejemplos son Apache Spark y Apache Storm, herramientas que forman parte del ecosistema Hadoop pero que, a diferencia de la mayoría, no aplican el modelo MapReduce o hacen un uso distinto de los recursos del clúster HDFS.

3 y 4. Apache Spark y Apache Storm

- **Apache Spark:** Plataforma que implementa una arquitectura "**In-Memory**" para el procesamiento y análisis de Big Data en **tiempo real**.
  - "In-Memory": Hace uso de la memoria RAM de los equipos que forman parte del clúster HDFS
  - No aplica el modelo de computación Map/Reduce
  - **Hasta 100 veces más rápido que HDFS/MapReduce**
- **Apache Storm:** Sistema para el procesamiento distribuido, tolerante a fallos y en **tiempo real** de datos en **Streaming**
  - Soporta el **procesamiento de más de un millón de filas por segundo y nodo**
  - No aplica el modelo de computación Map/Reduce, sino que se basa en el uso de topologías
    - Grafo que describe como se ha de llevar a cabo la computación formado por nodos **Spouts** (representa una fuente de datos streaming) y nodos **Bolt** (representan un procesamiento sobre los datos)

5. SGBDR extendidos con MPP

Por otro lado, cuando requerimos el procesamiento y análisis de Big Data en tiempo real puede darse un escenario distinto a los anteriores si...

- Necesitamos procesar y analizar grandes volúmenes de datos Big Data en el que el nivel de estructura predominante es estructurado
  - Ej. Información estructurada en tablas en las que alguna comuna almacena documentos XML o JSON
- Para este escenario, es adecuado el uso de **Sistemas de Gestión de Bases de Datos Relacionales con características extendidas para dar soporte a Big Data**

(Ya vimos algunas de las características de los SGBDR extendidos al inicio del tema, las cuales pueden resumirse en) (Los sub-puntos siguientes no los muestro en las transparencias a excepción de los ejemplos)

- Campos que permiten almacenar información semi o nada estructurada
- Funciones de análisis definidas por el usuario (UDF) que se ejecutan lo más cerca posibles de los datos.

    - MPP (Massively Parallel Processing): Procesamiento distribuido muy eficiente,
    - Coste elevado:
        - Licencia
        - Infraestructura
    - **Ejemplos**: Vertica de HP o Greemplun de EMC
- Adecuados para el análisis OLAP tras dotar de estructura a los datos con herramientas como las del entorno Hadoop.

6. Bases de datos NoSQL

En el último de los posibles escenarios es donde entran en juego las bases de datos conocidas como NoSQL.

- Nuestro objetivo (en este curso) es el desarrollo de aplicaciones de Inteligencia de Negocio que hagan uso del Big Data
    - Para ello necesitamos características similares a los sistemas OLAP tradicionales pero evolucionados para dar soporte a Big Data
- Sin embargo, en otros escenarios, podemos necesitar características OLTP (Sistemas transaccionales) sobre Big Data
    - Ej. Gestión de datos de tipo semi estructurados y no estructurados generados y usados en una aplicación web. Un ejemplo muy evidente es una red social como Facebook o Twitter.
- En otros casos, también podemos necesitar realizar un análisis de elementos interrelacionados que pueden ser representados de forma óptima en forma de grafo
    - Ej. Análisis de las relaciones existentes entre los usuarios de una Red Social como Facebook o aplicaciones para el análisis de trayectorias a partir de datos de tags RFID.

7 y 8. Bases de datos NoSQL

- Para casos como los anteriores, la mejor opción es el **uso de Bases de Datos NoSQL**
    - A diferencia de Hadoop, están optimizadas para la lectura y actualización aleatoria de pequeños subconjuntos de datos dentro del conjunto de datos Big Data almacenado.
    - Permiten la inserción y actualización a nivel de fila (Hive no lo permitía)
    - Relajan las restricciones ACID (necesarias para OLTP) para dar soporte a Big Data.
- Existen diversos tipos de bases de datos NoSQL y numerosas implementaciones.
    - Clave/Valor, documentales, columnares y de grafos.

9. Conclusiones

Como hemos podido ver, existen diversas alternativas a la combinación Hadoop HDFS/MapReduce para dar soporte al Big Data

- Sin embargo, cada arquitectura o herramienta da soporte a determinados requisitos del procesamiento y análisis del Big Data, pero ninguna es capaz de ofrecer soluciones a todos los problemas.
- Por ello, es necesario hacer un uso combinado de aquellas que necesitemos para nuestros objetivos de análisis, conociendo cuales son las más adecuadas en cada posible escenario.