Hadoop

Historia

◦ Hadoop está inspirado en los documentos de Google para MapReduce y GFS (Google File System)
◦ Actualmente es un proyecto de alto nivel de Apache soportado por la comunidad global de contribuyentes.
◦ Fue creado por Dough Cutting, que lo nombró así por el elefante de juguete de su hijo
◦ Fue inicialmente desarrollado para apoyar la distribución del proyecto de un motor de búsquedas llamado Nutch
◦ Principales distribuciones:
    ◦ Cloudera
    ◦ Hortonworks
    ◦ MapR

Justificación

◦ Cada vez se producen más datos en el mundo: más dispositivos conectados
    ◦ Sensores, Logs, IoT, Transacciones, RRSS, Clickstream, etc…
◦ Y cada vez más rápidamente
    ◦ Conectividad constante
    ◦ Incremento de la automatización de las industrias
    ◦ Seguimiento más granularizado del usuario

Valor de los datos

◦ Predicción
◦ Mejora de procesos (+completos, +potentes, +sencillos..)
◦ Segmentación -> conocimiento 360º
◦ Ahorro de costes de almacenamiento
◦ Disminución del tiempo de procesado
◦ Ejecución de procesos antes imposibles

Lo más importante en Hadoop es:
◦ Llevar el procesamiento a los datos
También muy importante
◦ Distribuir los datos según llegan al Cluster
◦ Aplicaciones escritas en alto nivel
◦ Los nodos se comunican entre sí lo menos posible
◦ Los datos se replican para obtener
    ◦ Disponibilidad
    ◦ Fiabilidad
    ◦ Tolerancia a fallos

Hadoop es
◦ Escalable horizontalmente y con coste reducido
  ◦ + Datos
  ◦ + Potencia
◦ Tolerante a fallos
  ◦ Que un servidor falle es inevitable
  ◦ Pero los sistemas deben seguir funcionando
  ◦ De manera transparente al usuario
  ◦ Sin perder datos
  ◦ Debe recuperarse sin intervención del usuario
◦ Los jobs han de ejecutarse de manera independiente
◦ La salida de un job no debe condicionar la salida de otro job, salvo que sea necesario
◦ El rendimiento de un job se puede ver alterado por la ejecución de otro job dado que la potencia del Cluster es la que es, pero es algo que ya se tiene en cuenta.

Paradigma de Hadoop. Ha de ser:
◦ Sencillo de aprender
◦ Fácil de manejar desde un punto de vista conceptual
◦ Sencillo de sacar partido: APIs para programadores novatos o gente de negocio
oHadoop está programado en Java
oOriginalmente se interactúa con él, programando en Java
oAunque existe la posibilidad de hacerlo con otros lenguajes de programación o herramientas
oEl programador solo se preocupa de codificar el código que resuelve el caso de uso, no se encarga de gestionar la coordinación, sincronización o los posibles fallos del sistema

Tipos de datos en Hadoop:
◦ Estructurados
◦ Semi-estructurados
◦ No estructurados
Tiempo de procesado:
◦ Originalmente procesado por lotes
◦ Near Real Time
◦ Real Time (Hbase-> db noSQL, Storm-> no tiene una arquitectura de origen y final, se procesa y se analiza continuamente)
Tipo de procesado:
◦ Ejecución en paralelo
◦ Sobre datos distribuidos (HDFS)

Hadoop se compone de
◦ HDFS
  ◦ Almacena datos en el cluster
◦ MapReduce
  ◦ Procesa datos en el cluster
◦ Ecosistema de Herramientas
  ◦ Conjunto de herramientas que hacen más fácil trabajar con Hadoop
Cluster
◦ Un cluster se compone de un conjunto de servidores (nodos) que trabajan juntos para conseguir un objetivo común
◦ Hay dos grandes tipos de nodos que componen un cluster:
  ◦ Maestros (gobernar el cluster)
  ◦ Esclavos (procesar/almacenar la inf)
◦ Cada nodo tiene sus Demonios corriendo, dependiendo del tipo que sea