Introducción al Ecosistema Hadoop

- Hadoop por sí solo no sería suficiente como entorno de trabajo en producción
- Las empresas necesitan agilidad a la hora de integrar nuevos sistemas
- Sus profesionales no siempre están acostumbrados a trabajar en entornos tecnológicos puros.
- Hay analistas de datos tradicionales que no saben que es /root
- Hay un montón de orígenes de datos válidos para su estudio
- Hay muchas herramientas de exploración y descubrimiento de datos
- Hoy en día casi todos los proveedores de SW del mercado proveen conectores de sus herramientas con Hadoop
- A continuación presentamos alguna de las herramientas principales empleadas en la mayoría de soluciones Big Data

Introducción al Ecosistema Hadoop

- Hive
  - Es un SW para interrogar/consultar y manipular datos escritos en HDFS
  - Fue creado por Facebook para suplir la carencia de conocimiento de java de sus programadores
  - A través de metadatos es posible trabajar con datasets con un lenguaje similar al SQL, llamado HQL
  - Su característica fundamental es que traduce Queries HQL a MapReduce, aprovechando toda la potencia del cluster
  - Conforme las versiones aumentan, se va añadiendo más funcionalidad
  - Un ejemplo:

```
SELECT sample_07.description, sample_07.salary
FROM
  sample_07
WHERE
( sample_07.salary > 100000)
SORT BY sample_07.salary DESC
```

Introducción al Ecosistema Hadoop

- Pig
  - Es una plataforma para analizar grandes datasets almacenados en HDFS
  - Desarrollado por Yahoo! Por la misma razón que FB creó Hive
  - Consta de un lenguaje propio llamado Pig Latin semejante a HQL pero menos que Hive
  - Su principal característica es que transforma consultas en PigLatin en MapReduce
  - Ejemplo

```
A = LOAD 'file1' AS (x, y, z);
B = LOAD 'file2' AS (t, u, v);
C = FILTER A by y > 0;
D = JOIN C BY x, B BY u;
E = GROUP D BY z;
F = FOREACH E GENERATE
    group, COUNT(D);
STORE F INTO 'output';
```

Introducción al Ecosistema Hadoop

- Sqoop
  - Su función principal es la de traer datos desde BBDD relacionales a HDFS
  - Su sintaxis consta de una parte de configuración más otra parte de SQL
  - Originalmente fue creado por Cloudera
  - Actualmente existen conectores de casi todas las bases de datos relacionales con esta herramienta
  - Ejemplo

```
--target-dir /biginsights/hive/warehouse/obsidiana.db/big_CLDOMICILIO
$SQOOP_HOME/bin/sqoop import --options-file
/gneis/datos/comunes/conexion/sqoop.cfg/sqoop-options-SRVepifis.txt --fields-terminated-
by "\t" --m 1 --null-non-string "\\\\N" --hive-import  --hive-overwrite --hive-table
obsidiana.HST_EXT_CONTACTOS_BKTEL    --query  "select FECHA,TRANSACION,PERSONA
from epistage.HST_EXT_CONTACTOS_BKTEL where \$CONDITIONS"
```

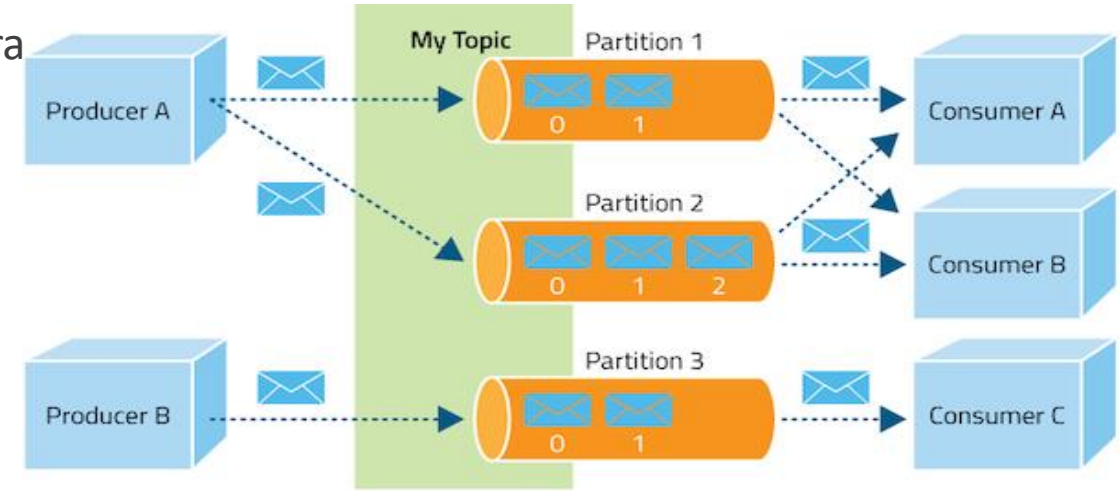
Introducción al Ecosistema Hadoop

- Flume
  - Herramienta diseñada para importar datos en un cluster en tiempo real
  - Pensada para orígenes de datos diferentes a BBDD relacionales (servidores web, servidores de correo, logs de dispositivos, etc.)
  - Fue inicialmente creado por Cloudera
  - Consta de tres partes que hay que configurar dependiendo de lo que queramos obtener o necesitemos
    - Source
    - Channel
    - Sink
  - Ejemplo

```
a1.channels = c1
a1.sources = r1
a1.sinks = k1

a1.channels.c1.type = memory

a1.sources.r1.channels = c1
a1.sources.r1.type = avro
# For using a thrift source set the following instead of the above line.
# a1.source.r1.type = thrift
a1.sources.r1.bind = 0.0.0.0
a1.sources.r1.port = 41414

a1.sinks.k1.channel = c1
a1.sinks.k1.type = logger
```

Introducción al Ecosistema Hadoop

- Kafka
  - Proyecto cuyo objetivo es proporcionar una plataforma unificada, de alto rendimiento y de baja latencia para la manipulación en tiempo real de fuentes de datos
  - Puede verse como una cola de mensajes bajo el **patrón publicación-subscripción**, **masivamente escalable** y concebida como un registro de transacciones distribuidas
  - Desarrollado por Linkedin
  - Arquitectura

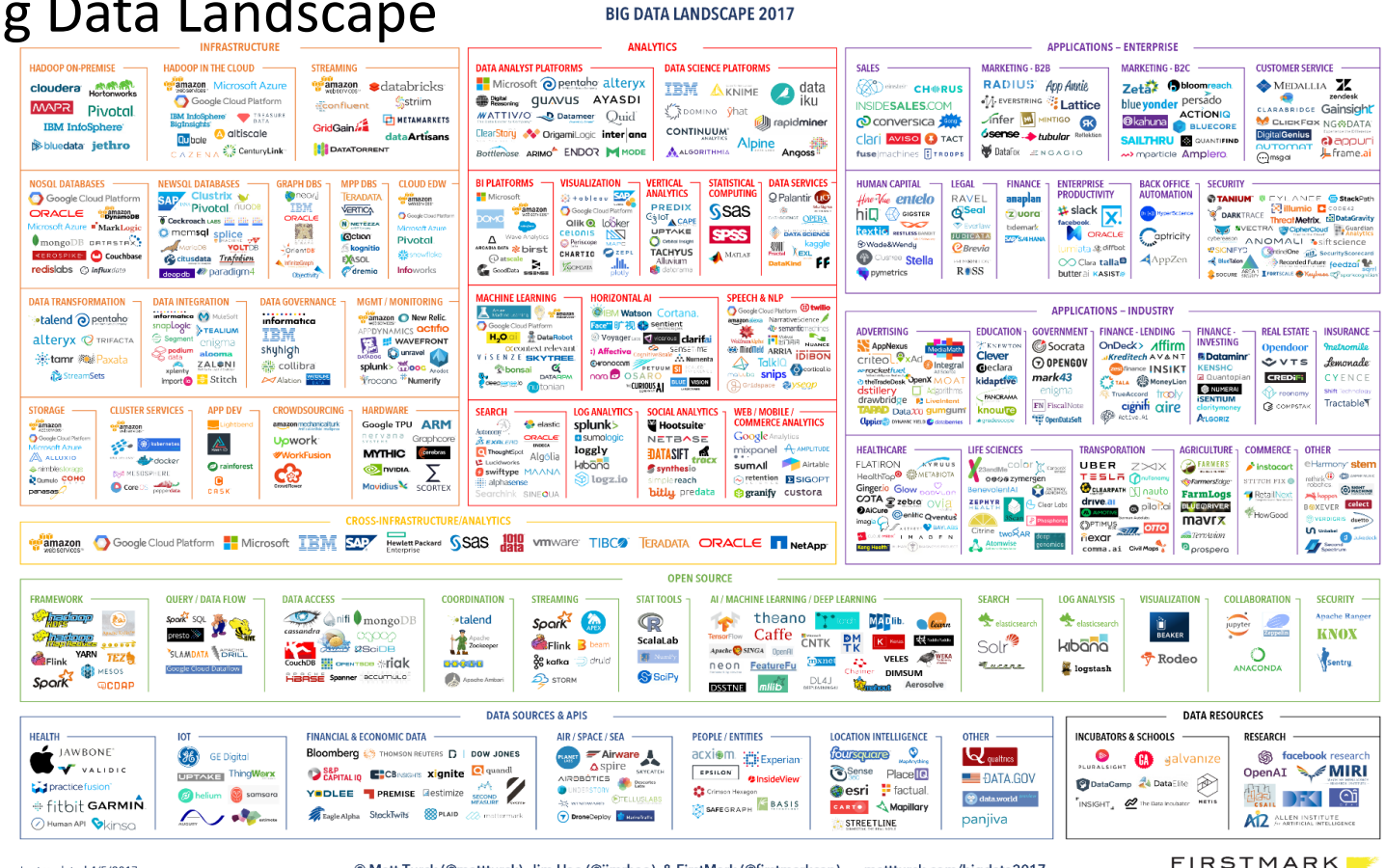


Introducción al Ecosistema Hadoop

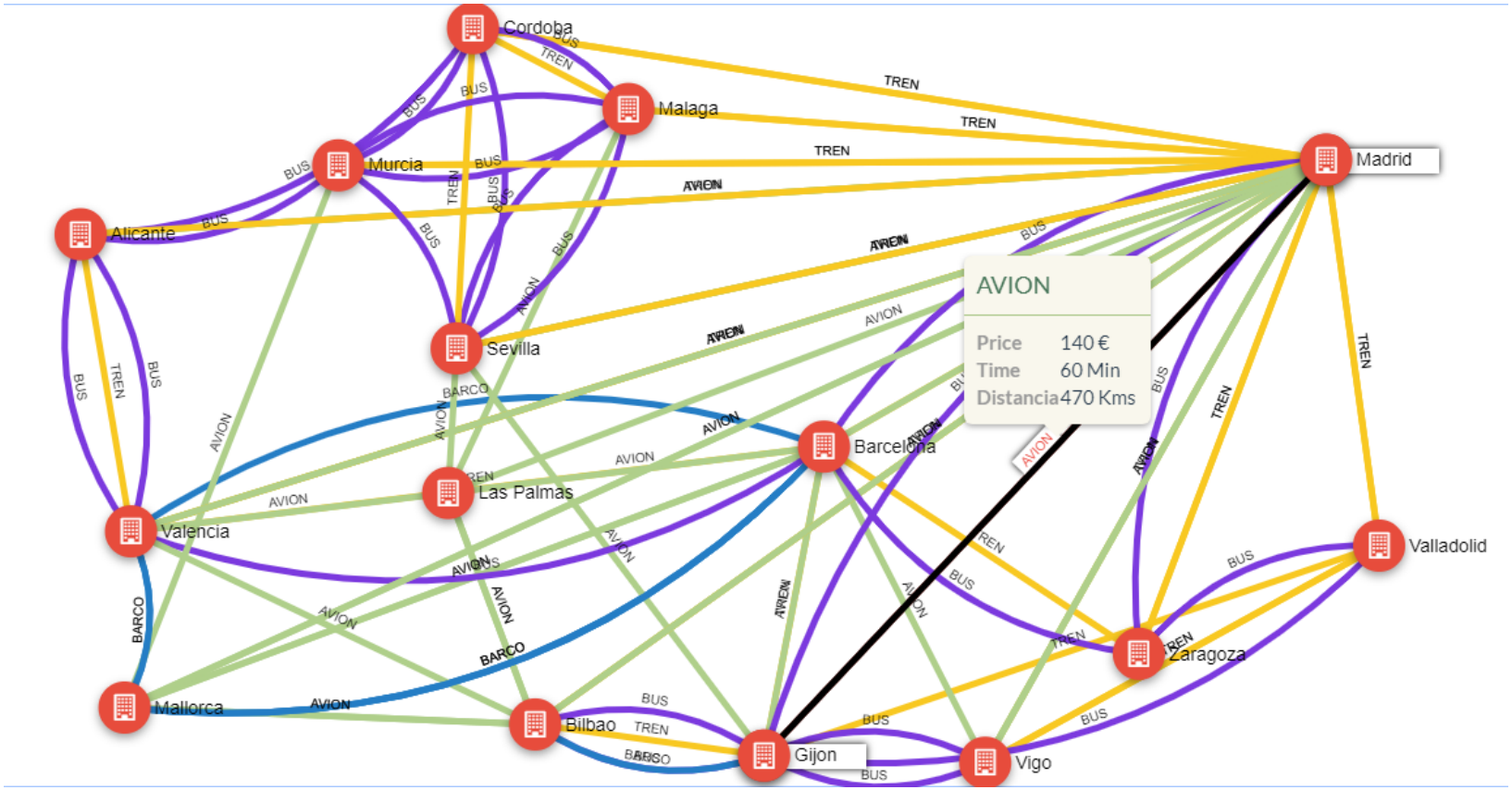
- Bases de datos No-SQL
  - No-SQL significa “Not Only SQL”, es decir, que no usan por defecto el lenguaje SQL para hacer consultas
  - Pensadas para trabajar con datos multiestructurados
  - Permiten distribución de datos y ejecución en paralelo
  - No garantizan completamente ACID (atomicidad, consistencia, aislamiento y durabilidad)
  - Escalan muy bien horizontalmente
  - Pensadas para millones de columnas por billones de filas
  - Hay varios tipos de ellas
    - De clave valor como Cassandra
    - Orientadas a documentos como MongoDB
    - Orientadas a grafos como Neo4j
    - Orientada a columnas del tipo clave valor como Hbase
    - Otras

Introducción al Ecosistema Hadoop

- Big Data Landscape



Neo4j


Cordoba
Malaga
Murcia
Alicante
Sevilla
Madrid
Barcelona
Las Palmas
Valencia
Mallorca
Bilbao
Gijon
Vigo
Zaragoza
Valladolid
AVION
Price 140 €
Time 60 Min
Distancia 470 Kms
TREN
BUS
AVION
BARCO

Planificación cluster Hadoop

- Planificar las capacidades de un cluster Hadoop no es complicado siempre y cuando sepamos lo que vamos a hacer con él
- Hay muchos factores que importan
  - Volumen de datos actual
  - Previsión de crecimiento anual
  - Casos de uso que vamos a llevar a cabo en él
  - Tipo de datos que vamos a incorporar
  - Herramientas que vamos a instalar
- Ejemplo
  - Supongamos que queremos crear un cluster que permita incorporar 6 TB de datos al mes
  - Tenemos un factor de replicación de 3
  - Por lo que necesitamos 3x6 TB = 18TB de almacenamiento nuevo cada mes
  - A lo que hay que añadir un 25% de espacio para uso del propio cluster
  - Necesitaremos alrededor de 24TB de almacenamiento nuevo al mes
  - Actualmente se están montando nodos con estas características
    - dual Xeon octocore, 256 GB RAM, 8 x 3TB SATA III, dual 10 Gbit de red
  - Por lo que necesitaremos un nodo nuevo cada mes

Planificación cluster Hadoop

Como sabemos, hay dos tipos de nodos

- Maestros
- Esclavos

- Los maestros están pensados para almacenar metadatos y gobernar el cluster (SPOF)
- Los esclavos para procesar/almacenar la información
- Originalmente, cuando Hadoop fue creado, se pensó para procesamiento por lotes, por lo que era justificable, en base al uso que se le daba a cada nodo, que los maestros tuvieran más RAM y menos disco y los esclavos más disco y menos RAM
- Con la llegada de nuevas herramientas y el enorme éxito que está teniendo Big Data, cada vez se pide más que todo el procesamiento sea en memoria para que se realice más rápidamente, Near Real Time o Real Time (Spark, Storm, Flink …)
- Por esta razón los nodos de los cluster actuales tiene características similares entre ellos
- No es necesario un gasto extra en CPU dado que habitualmente los cuellos de botella de los clusters se encuentran en el Disco/Memoria y en el tráfico de red
- Si nuestro cluster va a ser utilizado para trabajar con algoritmos de ML, sí es recomendable gastar dinero en un procesador potente, dado que nos ahorraremos bastantes horas de trabajo esperando a que terminen los procesos
- Algunos clusters incorporan discos SSD (discos de estado sólido) para agilizar algunos procesos (Solr) aunque resultan caros todavía en comparación a los discos ópticos

Big Data: Negocio

1. Cuándo implantar una solución Big Data
2. Como impacta Big Data en las empresas
3. Beneficios del Big Data
   - Democratización del Dato: transversalidad
   - Unificación de conocimiento departamental
   - Decisiones basadas en el Dato
   - Mejora de procesos
   - Generación de nuevos procesos
   - Cambio de mentalidad directiva

Preguntas/Debate