Repaso examen 21/05

Que tiene el master node? Los nombres de los ficheros.

1- Introducción

1.1 – Las 5 V del Big Data

¿Qué es big data? 1. Introd al Big Data pág 5

Volum → Moltes dades
Varietat → De molts tipus i fonts diferents
Velocidad → Analitzades ràpidament
Veracidad → Les dades son verdaderes
Valor → Aporten un valor (informació)

Exemple Volumetria:
- Telescopis
- Sequenciador genètic
- Netflix: Analisi de recomanacions a través de visualitzacions
- Google: Anàlisi de cerques, de mapa (trànsit, recorreguts, estades...)
- Amazon: Compres
- Facebook:
- Ebay: Anàlisi de comprres
- Spotify:

Exemple Volum + Velocitat:
- Manteniment predictiu: Predir el manteniment d'algun component.
- Gestió d'alarmes: Analitzar el procés d'algun component industrial i fer saltar l'alarma si falla. Per exemple si es filtren pèrdua de seguretat de arxius.
- Publicitat dirigida: Després de que una persona estigui comentant o buscant un cert producte, posar-li de seguida la publicitat encaminada.
- Detecció de frau: Analitzar quan un assegurador surt rentable o no. Analitzar gent que gasta diners a on no toca perquè li han robat el compte o perquè està fent un comportament il·legal.

1.2- Estructurado, Desestructurado, Semiestructurado

Introd al Big Data pág 12 a 15

Variabilidad de datos
Estrucutrado. Procedentes de bases de datos relacionales. Estructurados en tablas.
Desestructurado: Imágenes, Vídeos, Texto
Semi estructurado: Logs de un servidor. Formato JSON

1.3- Que es Big Data (escalable)

Introd al Big Data pág 17 a 21

BIG DATA: Búsqueda de soluciones para almacenar y analizar datos (estructurados y no estructurados) de una manera económica, escalable y tolerante a fallos con el objetivo de descubrir información nueva y valiosa.

SIRVE PARA:

• **Recopilar:** Big data facilita el paso para recopilar datos variados sin procesar (como registros, dispositivos móbiles…) en tiempo real o por lotes.

• **Almacenar:** Big data proporciona un repositorio seguro, escalable y duradero donde almacenar datos antes y después de procesarlos.

• **Procesar y Analizar:** Procesar (classificar, acumular, agregar…) datos para el consumo.

• **Consumir y Visualizar:** Consumir información visualizandola o inteligencia de negocio.

1.3dos – Qué tipos de transformaciones hay?

Ejemplos de tipos de procesos:
• Transformación técnica: Cambio formato, adaptación de los datos a una tecnología de uso.
• Transformación funcional. Aplicar una función a los datos: Filtrar, Agregar, Enriquezer

1.3tres – Qué tipos análisi hay?

Ejemplos de tipos de análisis:
• Descriptivo. *¿Qué ha pasado y por qué?* → Cuadro de mando
• Predictivo. *¿Probabilidad de que ocurra un evento?* → Alertas, Fraude, Mantenimento preventivo
• Prescriptivo. *¿Qué debería hacer si pasa x?* → Sistemas de recomendación.

1.3cuatro – Qué esclados hay?

Introd al Big Data pág 25 y 26

Vertical. Limitación técnica. Capacidad ordenador.

Horizontal.

Solucionamos poniendo más ordenadores y tener un sistema distribuido. No hay límite teórico de escalabilidad horitzontal en distribuida

1.3cinco – Tipos de modelos: Tradicional y Distribuido

Introd al Big Data pág 33

1.4- Hadoop (ecosistema). ¿Qué es cluster? ¿Qué es nodo?

Introd al Big Data pág 35

Hadoop: solución para el almacenamiento de datos distribuidos que proporciona una plataforma para implementar potentes frameworks de procesamiento distribuido

Permite almacenar cualquier tipo de dato tal como se genera en su origen, sin aplicar ninguna restricción de procesamiento.
• Datos distribuidos en su almacenado/ingesta.
• Procesos de computación en el lugar de su almacenaje.
• Hadoop se despliega sobre un conjunto de ordenadores (**clúster)** que trabajan de manera conjunta.
Se despliega en servidores profesionales montados en un rack, ordenadores domésticos, raspberrys...
• Cada una de las máquinas que forman el clúster se conoce como **nodo**.
• Hadoop proporciona fiabilidad, disponibilidad, escalabilidad y tolerancia a fallos (si falla un nodo, el sistema sigue funcionando. Se reasignan las tareas a otros clústeres).

Tiene sus propios frameworks de procesamiento: Frameworks principales:
• Map reduce (para batch)
• Apache Spark (batch y real time)

1.4dos- ¿Por qué tenemos replicas?

Introd al Big Data pág 42

- Por si hay fallos y se cae, tener copias. Tolerancia a fallos.

- Para tener más disponibilidad.

1.5- Batch vs Streaming

Introd al Big Data pág 51 a 54

Características Batch:
• La información se vuelca periódicamente (cada min, hora, semana, mes...) en un clúster.
• El procesamiento no se realiza en el momento en el que el dato se genera en la fuente.
• La respuesta al procesamiento puede demorarse un tiempo.
Aplicaciones: Generación de reportes y cuadros de mandos, análisis de datos históricos...

Características Real time:
• La información se vuelca en un clúster en el tiempo en el que se genera en la fuente.
• El dato es procesado en el momento en que llega al sistema.
• La respuesta al procesamiento es practicante inmediata.
Aplicaciones: Sistemas de alertas, publicidad dirigida, IoT...

1.5dos- ¿Cuales son los dos elementos para el procesamiento? Y cual tiene un modulo para real time?

Map Reduce

Spark – Tiene para real time

1.6- RDBMS (SQL, transaccional, schema)

Introd al Big Data pág 56

1.7- ACID

Introd al Big Data pág 56

Saber que estan los ACID.
Bases relacionales VS no relacionales.

1.8-

Introd al Big Data pág 62

Normalización

No normalización

2- Apache Hadoop (HSDF, yarn, mapreduce)

2.0 – ¿Por qué apareció Hadoop?

Dos señores querían indexar páginas.

2.0 dos – ¿Qué es un diamond?

Es un proceso. Depende de ese el nodo va a tomar una cosa y otra.

Tenemos Diamonds que se runnean en Slaves

Tenemos Diamonds que se runnean en Master

2.0 tres – ¿Definición master node? ¿Definición de slave node?

Master node: Envía las ordenes. Organización y distribución tareas.

Slave node: Procesan y almacenan

2.0 cuatro – ¿Para que sirve Sqoop? ¿Hive? ¿HBase?

Sqoop: Para ingestar datos

Hive: Para el procesado

HBase: Almacenamiento

Flume: Ingesta

2.0 cinco – ¿Diferencia entre Resource manager y aplication manager?

Aplication se encarga de dar containers en los sitios que dice el Resource manager (worker)

2.0 seis – ¿Como se llaman diamonds de HSDF? ¿Que almacena Name Node? ¿Que es un job, tarea?

Apache pág 28

Name Node

Data Node

Job es el trabajo que todo el cluster debe hacer.

Tarea son las ejecuciones de los mapers i reducers en cada nodo. Para un job se pueden tener 2 tareas de mapper i 3 de reducer.

2.0 siete – ¿Tipos de tareas en map reduce? Comandos sobre como hacer algunas cosas.

Map, Reducer, Shuffle and Sort, Combiner, Partitioner.

Shuffle and sort se hace después del maper y antes del reducer.

El Suffle and sort asegura que las mismas keys vayan al mismo reducer. Que no haya dos reducers que tengan la misma key.

2.1 – Yarn, gestión de recursos. Diferencia entre FIFO, Capacity Scheduler y Fair Scheluder

MIN 10:26

2.1dos – Mínimo número de nodos recomendado en un cluster

Apache pág 47

Dos para master node

Tres para los workers que tienen tres réplicas

2.1tres – Número de replicas para HDSF

Tres réplicas

2.1cuatro – Cual es el concepto de añadir nodos en bloque

Generación de nodos

2.1cinco – Por qué no añadimos los nodos 1 a 1

Porque los nodos tengan todas las mismas características de tecnología y antigüedad.

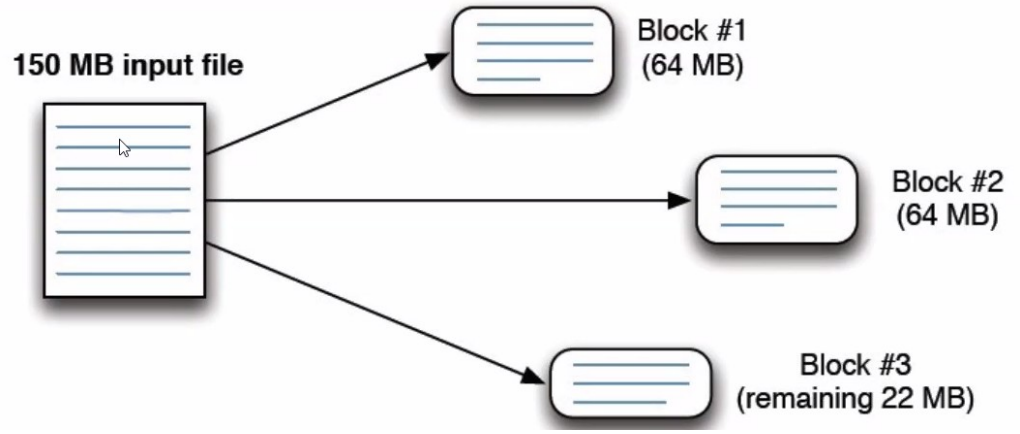
No sirve de nada poner nodos muy potentes de repente ya que la idea de hadoop es distribuir por igual el peso de las tareas.

2.1seis – Diferencia entre réplica y bloque

Replica es un nodo completo. Replica es toda la data. Bloque es una parte de la replica.

De cada bloque tiene 3 replicas. Una réplica es un bloque, un bloque no es una replica.

Un archivo nuevo se recibe, se consulta lo que tiene y se hace las replicas de los bloques que tiene ese archivo.



2.1siete – Rack vs replicas

Los nodos no están en el mismo rack.

En un rack se ponen réplicas y en otro rack se ponen otras réplicas.

2.1nueve – Qué es el Rack awareness?

Los nodos no están en el mismo rack.

2.2 – Yarn Ejercicio pg 29

2.3 – Pg 29. Client resource mangager application master mapear (datos intermedios en HDFS) Reducer

2.4 – Vamos a tener tantos maps como bloques (Examen!)

2.5 – Reducer empezara cuando todos los maps hayan terminado.

2.6 – El aplication master es reservado por resource manager. Despuest todo lo demás lo hace el aplicaction master.

2.7 – Yarn gestión concurrencia – scheduler

2.8 – Para matar tienes que hacer un kill, no vale con un control + C.

2.9 – Pg 42, cluster escala en horizonta, comodity hardware

2.10 – Pg 52. Roles

2.11 – Pg 57, distribuciones hadoop. Es un paqueta comercializado.

2.12 – Pg 68, back ups.

2.13 – Pg 80. Write once (es el tema de la replicas, si se modificase uno, se deberían modificar todas). HDSS vs Google

HDFS es basado en GFS (Google File System) los archivos son de “Write once” no se pueden editar.

2.14 – HDFS v2 el tamaño de bloque es 128. Pg 83. Esta mal. 3 réplicas por defecto ¿Cual es el tamaño por defecto de los bloques?

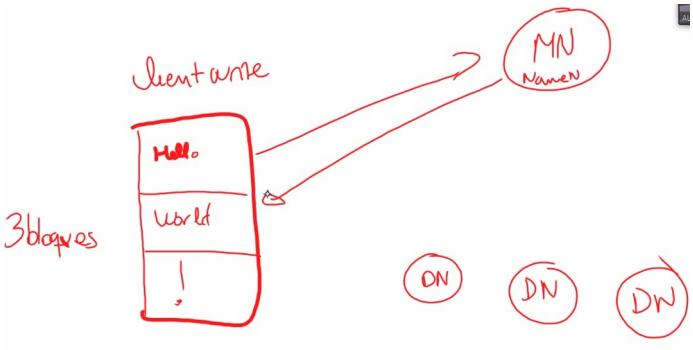
El tamaño por defecto de bloque es 128

2.15 – Pg 86 example. Importante NameNode tiene la metada de saberlo un poco todo.

Para escribir, vamos a hacer el diagrama write.

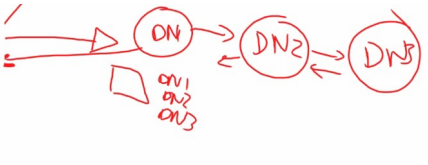
Cuantas IP devuelve? Una por cada replica.

Quien escribe? Data Node



DN3-DN2-DN se van diciendo lo que van escribiendo.

El masterNode nunca tiene los datos.



Para leer:

Los datos no viajan por el Name Node ni para escritura ni lectura. Hablamos con el Master Node pero nunca ve los datos.

Al escribir hacemos réplicas.

Los nodos hablan entre ellos, saben donde tienen que hacer las replicas.

El namenode tienen un hardware mejor.

2.15dos – Por que en HDFS interesa tener menos archivos y mas grandes

Apache pág 91

NameNode IP

Cuantos bloques hay y si hay replicados.

Al tener archivos más grandes, en el namenode lo que va a cambiar es el tamaño de su diccionario que se va a reducir. Vamos a tener menos datos.

- Datos almacenados como 1 archivo x 1GB
  - Nombre: 1 ítem
  - Bloques: 8 ítems
  - Ítems totales: 9

- Datos almacenados como 1000 archivos x 1MB
  - Nombres: 1000 ítems
  - Bloques: 1000 ítems
  - Ítems totales: 2000

2.15tres – ¿Cuando una tarea de map reduce es mas lenta, hay una manera de solucionarlo, cual es? ¿Que dos cosas hay que hacer con las clases antes de mandar el job? ¿Para que se crea el jar?

El driver tiene la configuración.

Para crear las clases hay que compilar y crear el job antes de mandarla.

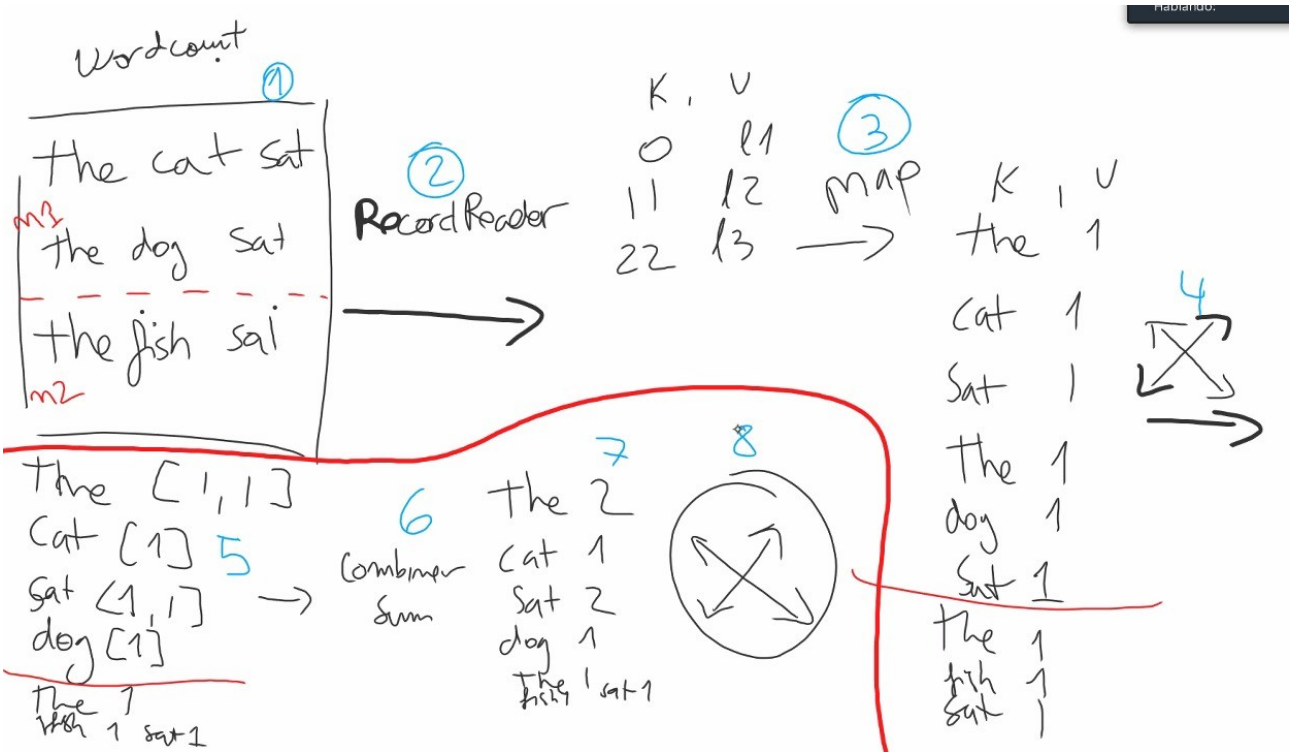
El jar se crea para ejecutar todo. Encapsulación. Tiene todas las dependencias que necesita.

2.16 – Map reduce , 109.

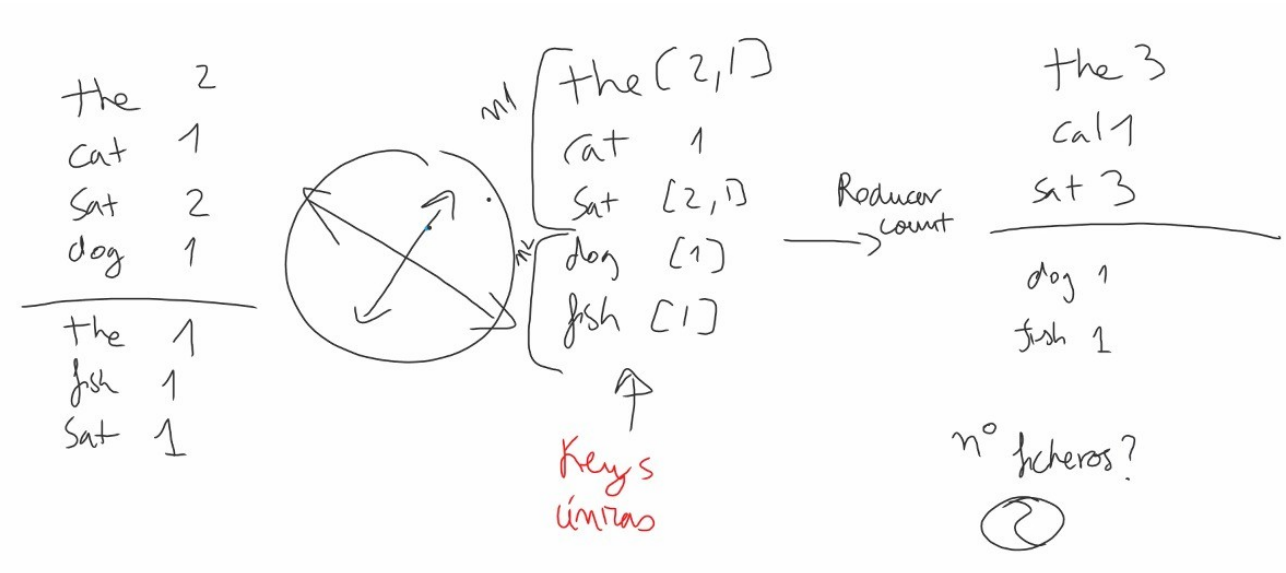
2.17 – Pg 114. Detalls.

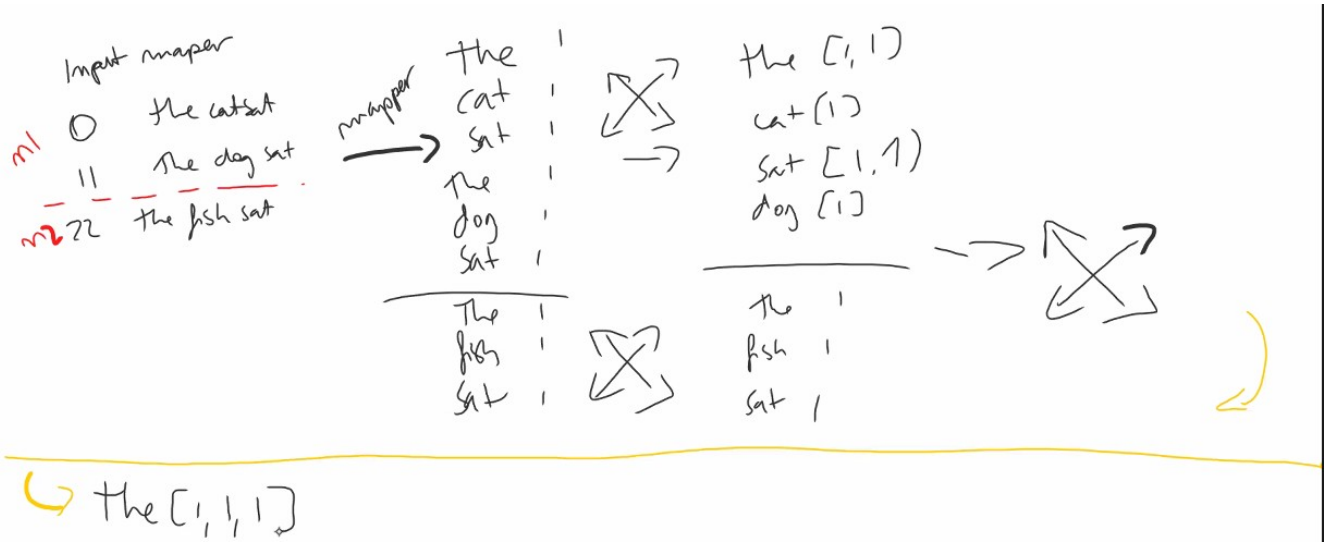
2.17dos – Pg 117 Maper+Combiner+Reducer // Maper+Reducer

• Maper + Suffle & Sort + Combiner + Suffle & Sort + Reducer



• Maper + Suffle & Sort + Reducer




Input maper
0 the cat sat
1 the dog sat
2 the fish sat
mapper
the 1
cat 1
sat 1
the 1
dog 1
sat 1
the 1
fish 1
sat 1
the [1, 1]
cat [1]
sat [1, 1]
dog [1]
the 1
fish 1
sat 1

↳ the [1, 1, 1]

2.18 – Pg 119. Tantos outputs como reducers, y tanto bloques como mapers.

2.19 – Ejemplo 120. Interesant. Interesante ver el partitioner i el shuffle&sort. El nombre, la r es reducer y la m es maper.

2.20 – 143. Com funciona en overview el map reduce.

2.21 – 22144 segona informacio.

2.22 – 145 speculative execution.

146

5- Spark

1- ¿Cuales son los cuatro modulos de Spark?

SQL Spark, SparkML, Streaming Spark, GraphX

2- Se pueden distribuir todos los modulos?

Hay algunos que no.

SQL para realizar consultas a dataframes de forma distribuida.

Streming para real time.

Graph para procesar grafos.

3- Al ejecutar un job de Spark tenemos dos grupos de procesos distintos. Cuales son?

Los drivers
Los executors que es lo que se ejecuta en los workers nodes.

4- Que es un DAG y para que sirve?

Directed Asisted Graph (Gráfico Acíclico Dirigido). Sirve para que no haya dependencias reversas. Solo van hacia adelante.

5- ¿Que es lo que no podemos hacer en Spark? ¿Por qué tenemos este problema?

No podemos hacer ciclicas.
No se puede asignar a una variable el mismo valor que tenia antes.

No podemos porque tenemos las Layzy executions.

6- ¿Que es el Spark sesion / Spark context?

Conecta con el clúster manager.

7- Tres elementos principales de Spark.

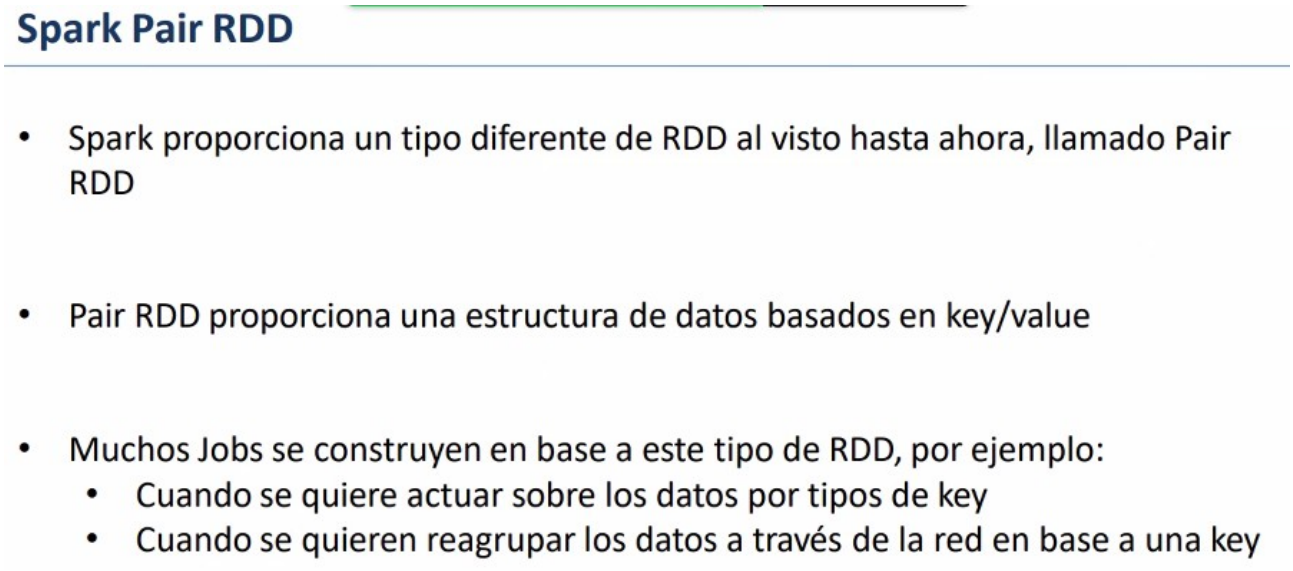
RDD, Transformación, Acción

8- Qué es un RDD?

RDD (Resilient Distributed Dataset). Colección de objetos distribuidos. Es immutable.

Las trasnformaciones filtran el dataset. Y hasta que no se ejectutan las acciones no se devuelve nada.

9- En PairRDD se pueden usar las mismas transformaciones que en RDD?



No. Puedes usar count pero no hay otras que no se pueden usar

10- En spark streaming. ¿Cual es el tipo de datos que se usa? ¿Que se puede hacer?

Se llama DStreams.

Se puede hacer transformaciones y output operations.

11- En spark streaming. ¿Qué dos tipos de transformaciones hay?

Uno usa windows. Independiente del estado anterior.

Uno no usa windows. Depende del estado anterior.

12- En spark streaming. Dos parámetros de las windows

Tamaño (Frame Size) = duración del muestreo

Salto (Hop) = Frecuencia del muestreo

13- En spark ML.

RF, Regresiones logísticas...

• PDF2 Apache Hadoop core2 part 2 estas importantes pero sale más a cuenta estudiarlas todas: 5 a 6 + 8 a 11 + 14 + 19 + 24 + 30 + 33 + 40 + 42 + 58 + 63 a 66 …

• PDF3 Apache Spark de la parte2. NO ENTRAN Ninguna de 1 a 12

• PDF4 → 1 a 14 y lo demás leer un poco y tener claro lo que hacen sin estudiar sintaxis ni comparativas que hay. De 15,16,18 NO

• PDF5 → entran las diapos 1 a 31 + 33 + 38 a 45 + 60 + 62 a 73 + 87 a 95 (entran 62 de 97; 64%)

• PDF6 → entran las diapos 1 a 14 + 18 a 21 + 30 a 31 + 46 a 51 (entran 26 de 52; 50%)

• PDF7 → entran las diapos de 1 a 10. (entran 10 de 87; 11%)

TOTAL: 98 de 236 → 42% de diapositivas útiles de estudio