Programa de formación MLDS

UNIVERSIDAD NACIONAL DE COLOMBIA

Módulo BIG DATA

MapReduce

Por
Ing. Jorge E. Camargo, PhD

Agenda

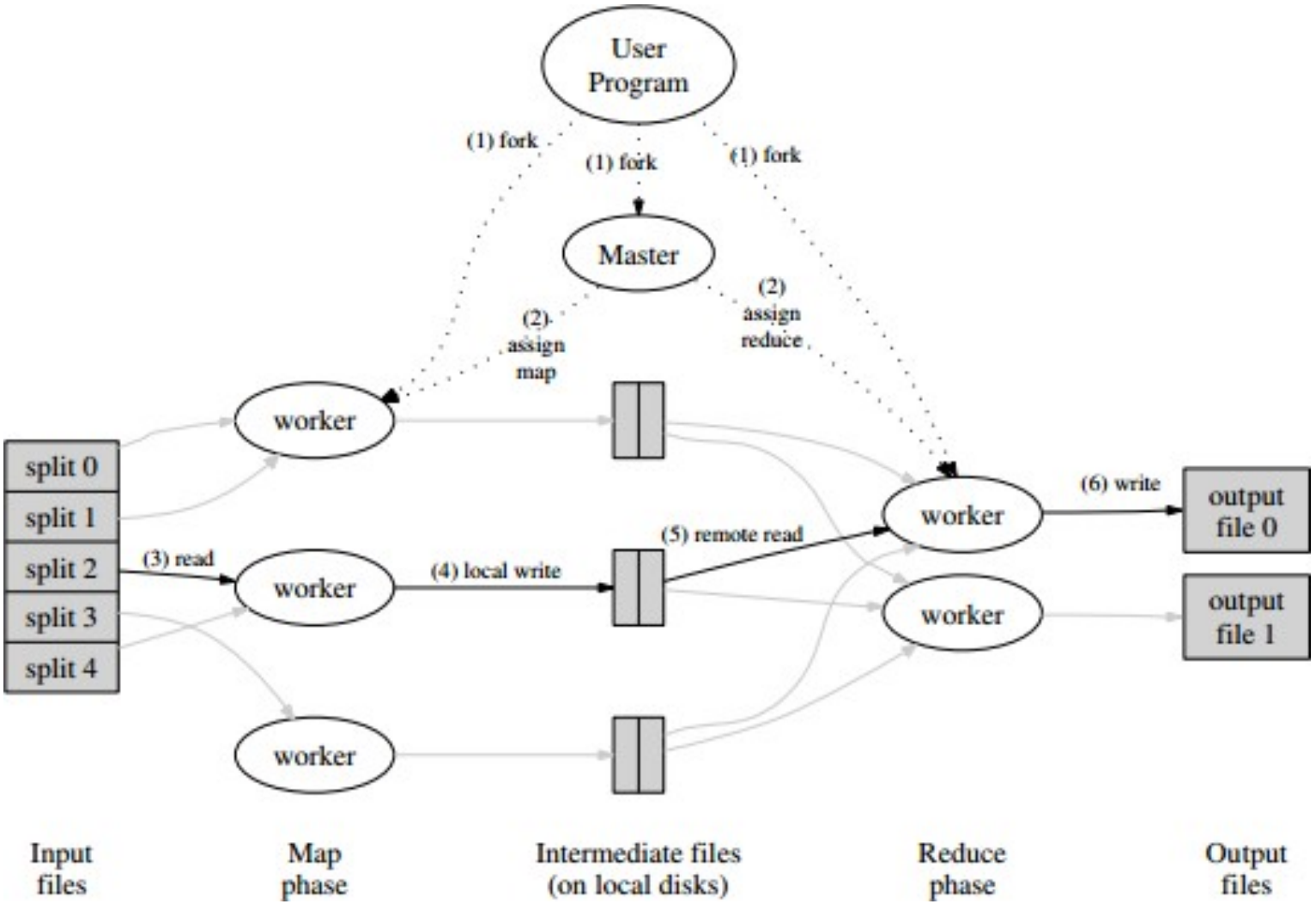
- Definición
- Funciones map y reduce
- Ejemplos
  - MapReduce en MongoDB
- Taller

Definición

- Es un “framework” para soporte de computación paralela en grandes conjuntos de datos (petabytes) de ciertos problemas específicos.
- Modelo de programación propuesto y usado por Google para mejorar rendimiento.
- Se ejecuta en sistemas de archivos distribuidos (HDFS)

Definición

- Los datos de entrada se dividen en “chunks” que son procesados por el “map” de manera paralela. El “framework” ordena las salidas del mapeo y son la entrada para el “reduce”. El framework se encarga de admininstrar las tareas, monitoreo y re-ejecutar en caso de falla.
- Dos funciones funcionales: Map y Reduce
- Definidas en tuplas del tipo (clave,valor).


User Program
(1) fork
(1) fork
(1) fork
Master
(2) assign map
(2) assign reduce
worker
split 0
split 1
split 2
split 3
split 4
(3) read
worker
(4) local write
worker
(5) remote read
worker
(6) write
output file 0
worker
output file 1
Input files
Map phase
Intermediate files (on local disks)
Reduce phase
Output files

Función Map()

- Recibe una serie de pares (clave/valor) en un dominio específico, procesa cada una, y genera cero o mas salidas de (clave/valor) en otro dominio.
- Se encarga del “mapeo” y es ejecutada en paralelo para cada par recibido.

Función Reduce()

- Por cada clave única, la función de reduce itera a través de los valores produciendo una salida.
- Se ejecuta en paralelo

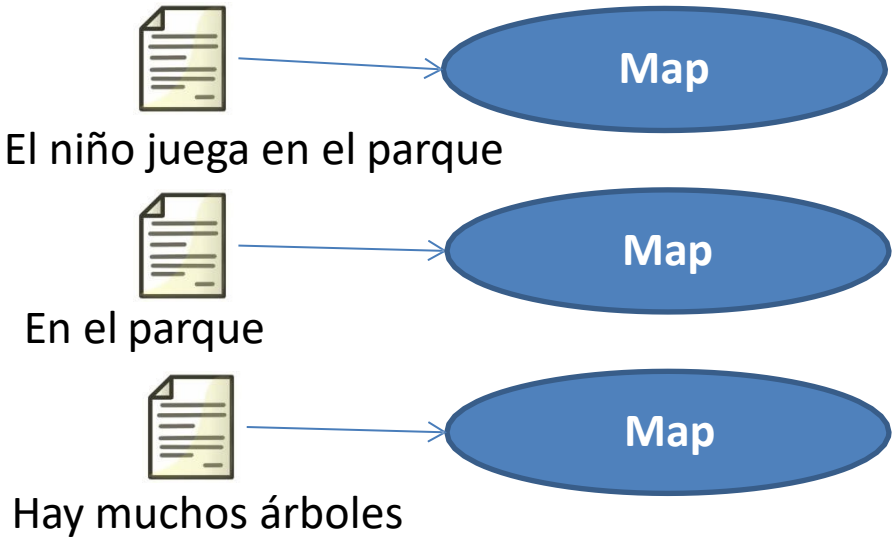
Ejemplo

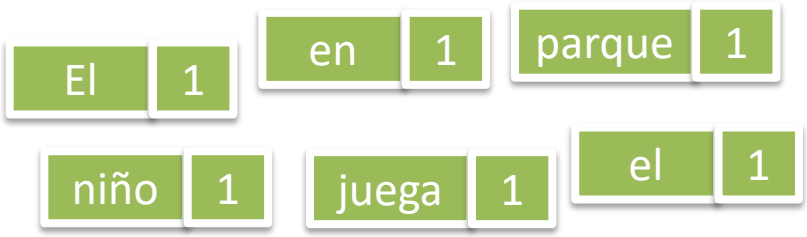
- Contar palabras en un documento (frecuencia de las palabras)

Doc (“El niño juega en el parque, en el parque hay muchos árboles”)

Map

Tareas de tokenization, lemmatization, steeming, etc





Reduce

Ciclo sobre las claves:

list : lista de resultados

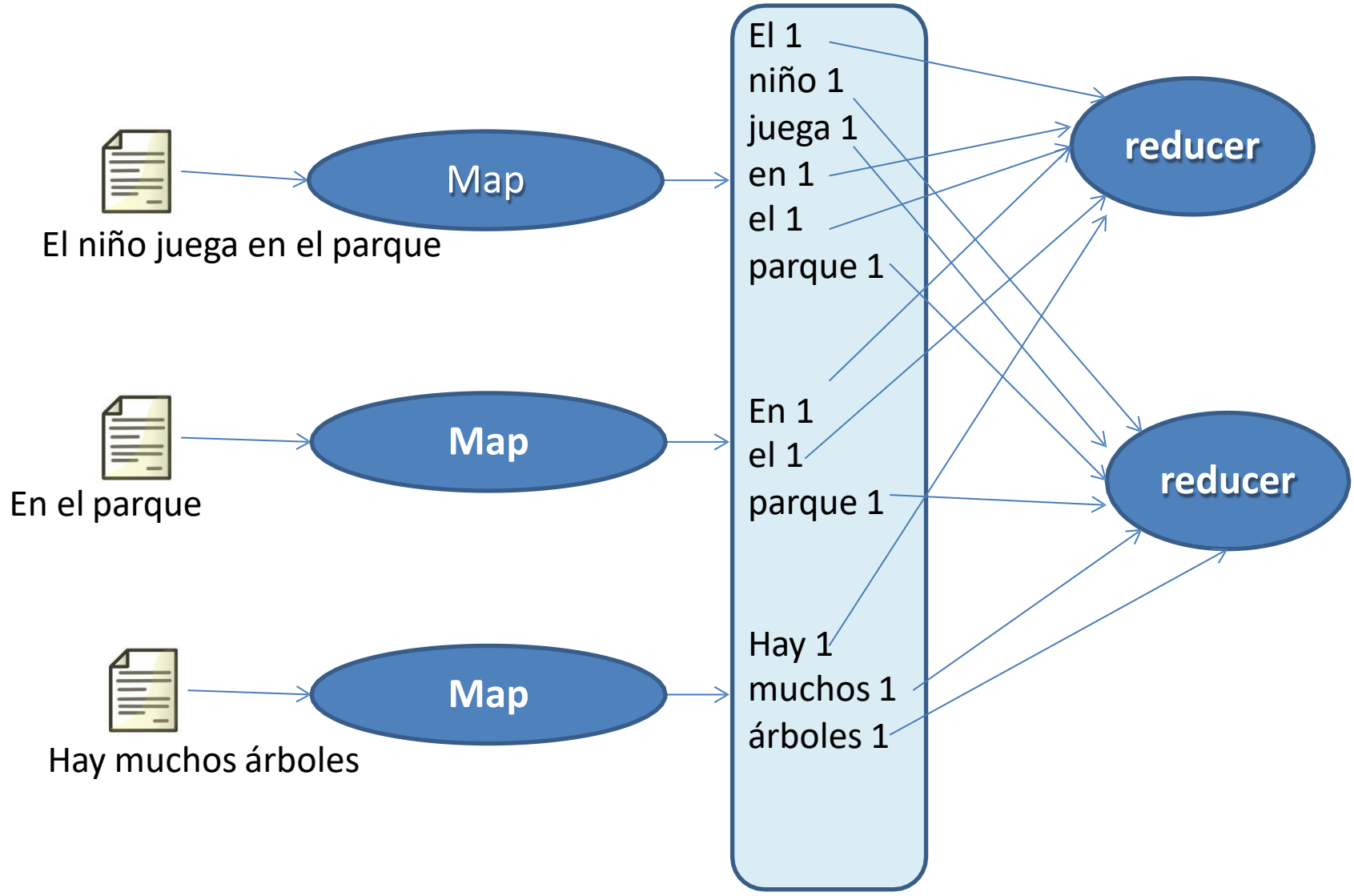
palabra <- get (ste palabra)

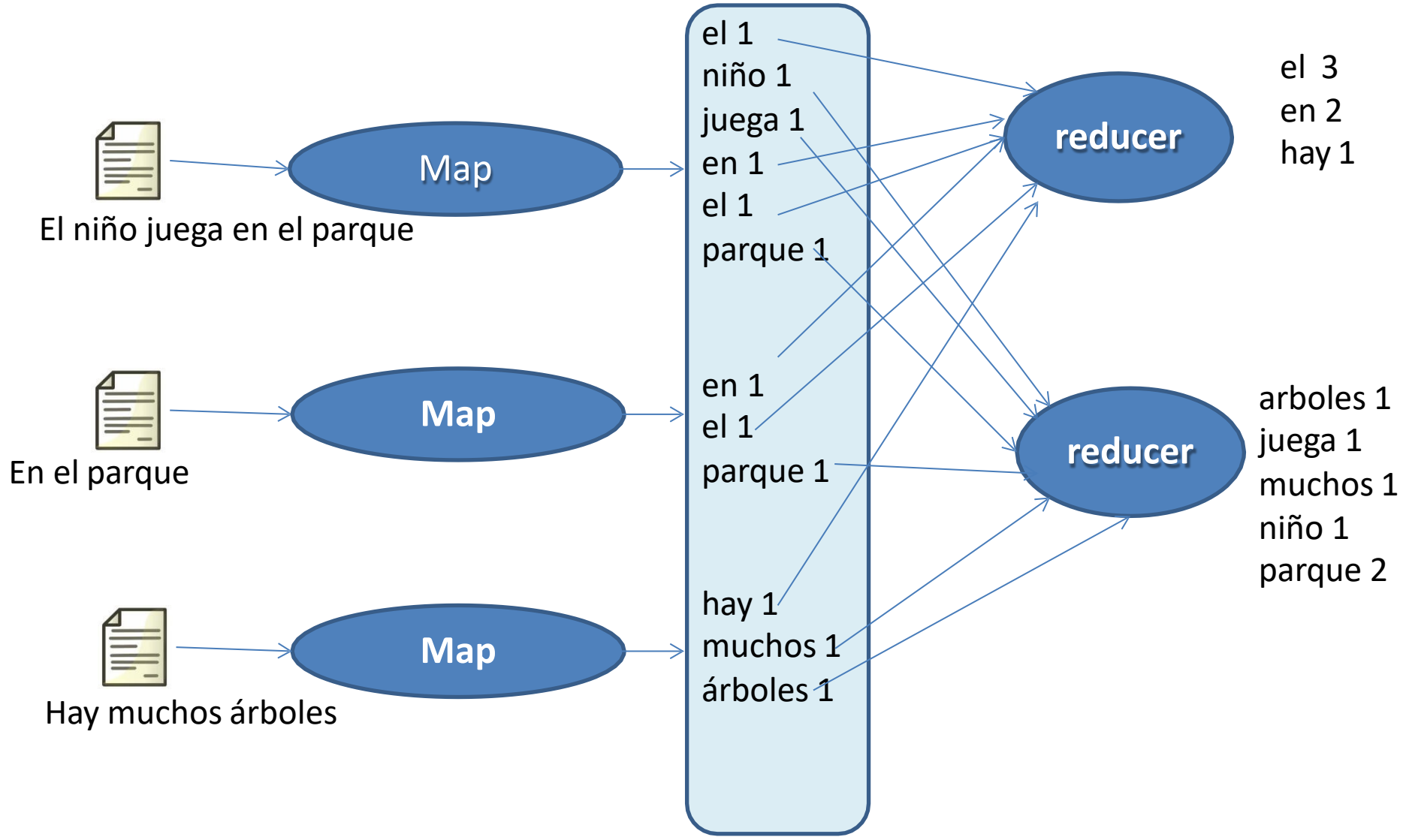
Si palabra es igual a palabra en list

add 1 to count in list

else

add <palabra, 0> to list


El niño juega en el parque
Map
En el parque
Map
Hay muchos árboles
Map
El 1
niño 1
juega 1
en 1
el 1
parque 1
En 1
el 1
parque 1
Hay 1
muchos 1
árboles 1
reducer
reducer


El niño juega en el parque
Map
En el parque
Map
Hay muchos árboles
Map
el 1
niño 1
juega 1
en 1
el 1
parque 1
en 1
el 1
parque 1
hay 1
muchos 1
árboles 1
reducer
reducer
el 3
en 2
hay 1
arboles 1
juega 1
muchos 1
niño 1
parque 2

Ejemplo: Construcción de un índice

- Contar palabras en documentos (frecuencia de las palabras) para construir un índice invertido.

Doc1 (“el niño juega en el parque, en el parque hay muchos árboles”)
Doc2 (“el fenómeno del niño acabará pronto ”)
Doc3 (“en el parque niños elevan cometa ”)
Doc4 (“el niño eleva cometa”)

	Doc 1	Doc 2	Doc 3	Doc 4
acabará	0	1	0	0
arboles	1	0	0	0
cometa	0	0	1	1
del	0	1	0	0
eleva	0	0	1	1
el	3	1	1	1
en	2	0	1	0
fenómeno	0	1	0	0
hay	1	0	0	0
juega	1	0	0	0
muchos	1	0	0	0
niño	1	1	1	1
parque	2	0	1	0
pronto	0	1	0	0

Ejemplo: Construcción de un índice

- Map(palabra, IdDocumento)
- Reduce (palabra, lista de documentos con su frecuencia c/u)

acabará	doc2,1
arboles	doc1,1
cometa	doc3,1 doc4,1
del	doc2,1
eleva	doc3,1 doc4,1
el	doc1,3 doc2,1 doc3,1 doc4 1
en	doc1,2 doc2,1
fenómeno	doc2,1
hay	doc1,1
juega	doc1,1
muchos	doc1,1
niño	doc1,1 doc2,1 doc3,1 doc4,1
parque	doc1,2 doc3,1
pronto	doc2,1

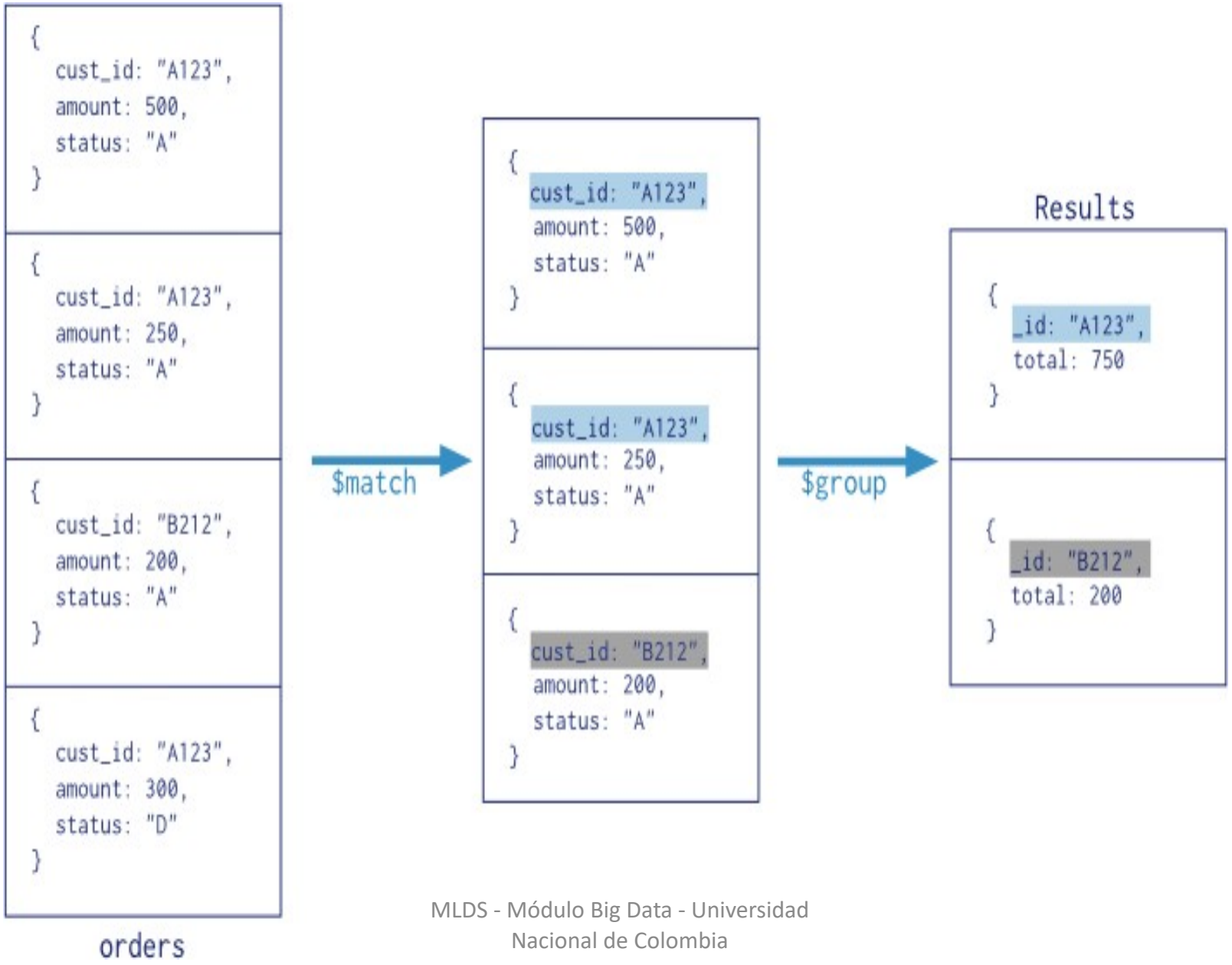
Ejemplo: Conteo de consultas

- Contar acceso a páginas web de un usuario:
  - la función MAP procesa los archivos weblogs y da la urls consultadas.
  - la función REDUCE sumariza el total de consultas por cada página.

MapReduce en MongoDB

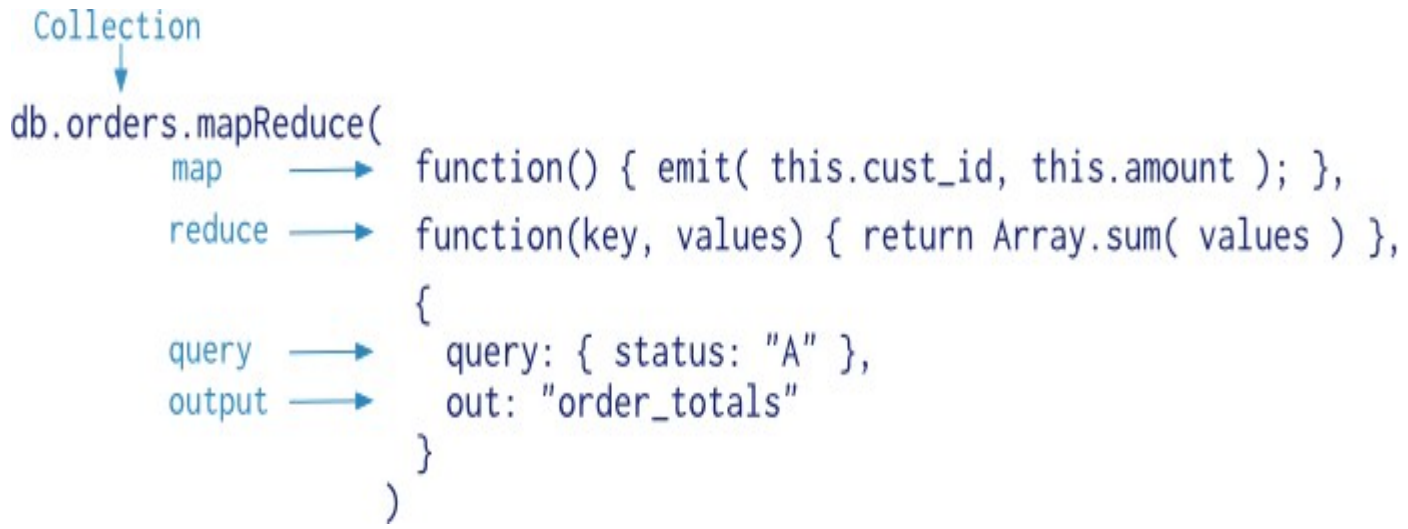
- Las funciones MapReduce deben estar escritas en JavaScript.
- Se puede acceder fácilmente a través de db.collection.mapReduce()
- Las vistas de MongoDB no soportan MapReduce.

Ejemplo: Agregación de documentos

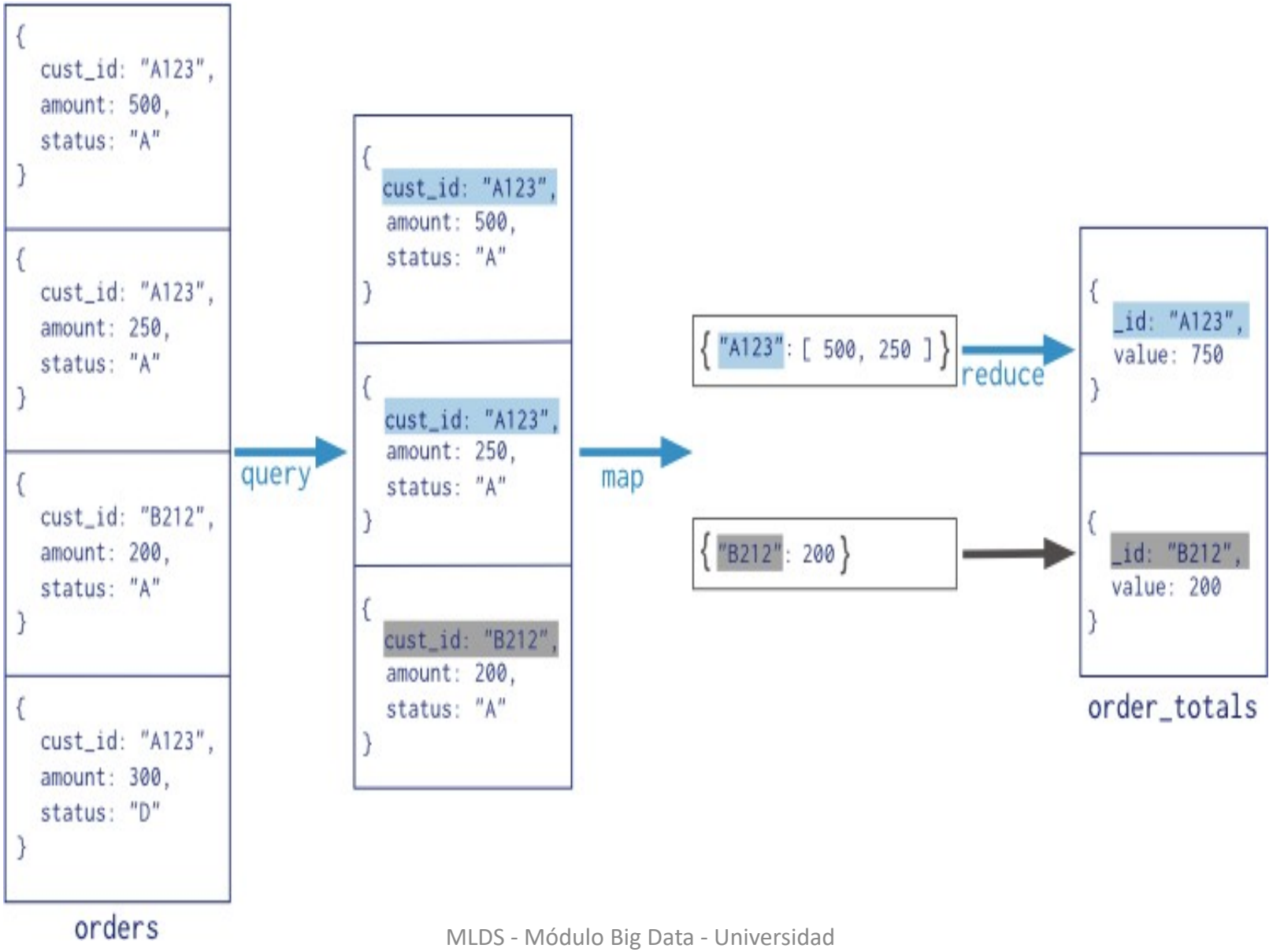


Ejemplo: Agregación de documentos

- Código en JavaScript:



Aplicación detallada de cada operación



Referencias

- Apache https://hadoop.apache.org/docs/r1.2.1
- MapReduce: Simplified Data Processing on Large Clusters, Jeffrey Dean and Sanjay Ghemawat, Google, Inc.