

NoSQL

Gerardo Rossel



2017

Map-Reduce

Introducción

- Vistas Materializadas
- Computación de grandes volúmenes de información
- ¿Dónde realizar el cómputo?
- Ranking de páginas WEB por importancia.
- Búsquedas en “amigos” en redes sociales que involucra grafos con cientos de millones de nodos.

Introducción

Jeffrey Dean y Sanjay Ghemawat

"MapReduce: Simplified Data Processing on Large Clusters"

In OSDI'04: Proceedings of the 6th conference on Symposium on Operating Systems Design & Implementation. 2004

Visión general

- Leer secuencialmente una enorme cantidad de datos
- **Map:** Extraer algo que nos interesa.
- Agrupar por clave. Ordenar y Mezclar
- **Reduce:** Agregar, sumarizar, filtrar o transformar
- Devolver el resultado

Adecuación

El esquema general siempre es el mismo. **Map** y **Reduce** cambian para adecuarse al problema

- 1 Entrada: un conjunto de pares clave-valor. De esta manera se logra permitir la composición de procesos
- 2 El desarrollador especifica dos funciones:
 - $Map(k, v) \rightarrow \langle k', v' \rangle^*$
 - Toma un par *key-value* y devuelve un conjunto de pares *key-value*
 - Hay un llamada a Map por cada par (k,v)
 - $Reduce(k', \langle v' \rangle^*) \rightarrow \langle k'', v'' \rangle^*$
 - Todos los valores de v' para la misma k' se reducen y procesan en conjunto

Esquema general

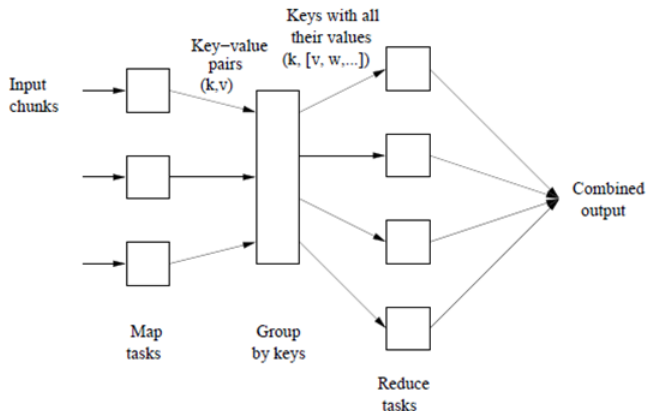


Figura de J. Leskovec, A. Rajaraman, J. Ullman: Mining of Massive Datasets, <http://www.mmds.org>

Ejemplo - Contar Palabras

```
map(String input_key, String input_value):  
  // input_key: document name  
  // input_value: document contents  
  for each word w in input_value:  
    EmitIntermediate(w, 1);
```

Map

```
reduce(String output_key, Iterator intermediate_values):  
  // output_key: word  
  // output_values: ????  
  int result = 0;  
  for each v in intermediate_values:  
    result += v;  
  Emit(result);
```

Reduce

Ejemplo

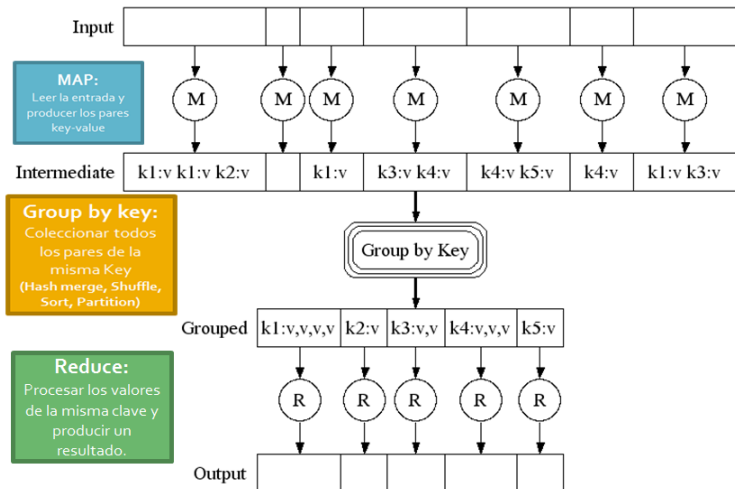
- Al finalizar las tareas Map los pares *key-value* son agrupados por key y los valores asociados con cada key son colocados en una lista de valores $(k, [v_1, v_2, \dots, v_n])$. Este agrupamiento es desarrollado por el sistema independientemente de lo que hagan Map y Reduce.

Paralelismo y arquitectura

El entorno Map-Reduce se encarga de:

- Particionar los datos de entrada.
- Planificar la ejecución de los programas en un conjunto de computadoras.
- Desarrollar el agrupamiento por clave.
- Manejar las fallas en las máquinas.
- Administrar la comunicación.

Esquema general



Paralelismo

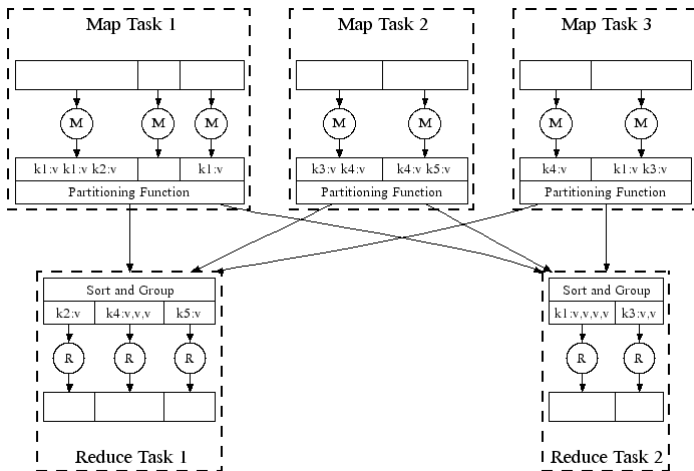
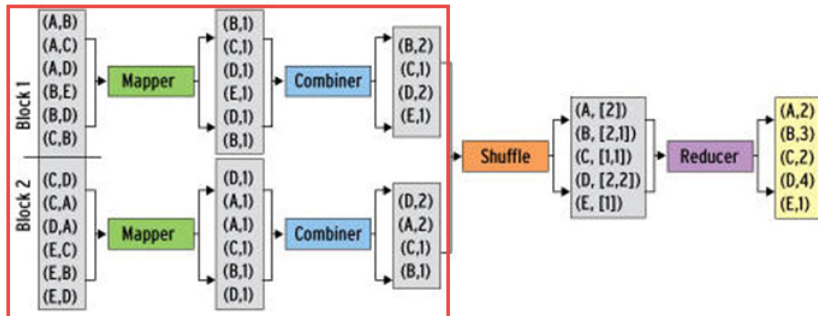


Figura de J. Leskovec, A. Rajaraman, J. Ullman: Mining of Massive Datasets, <http://www.mmms.org>

Paralelismo

● Combinable Reducer

- Es un *Reduce* que se ejecuta en el mismo nodo que el *Map*
- No todas las funciones *Reduce* son combinables
- La entrada debe coincidir con la salida. La función *Reduce* debe ser asociativa y conmutativa



Ejemplo - Map

Calcular el ingreso total por producto.

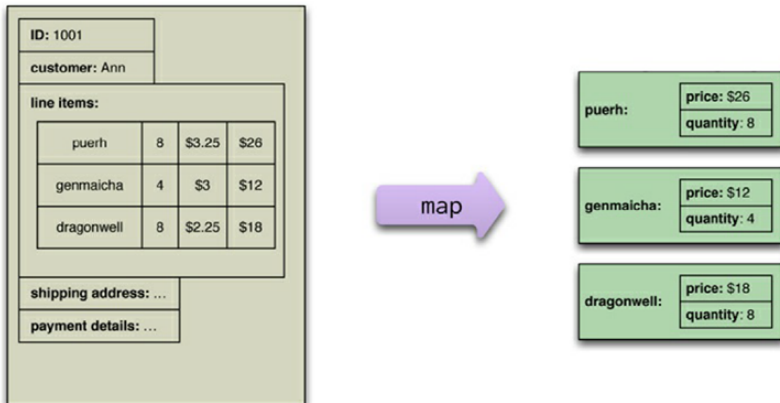


Figura de Sadalage y Fowler / NoSQL distilled : a brief guide to the emerging world of polyglot persistence

Ejemplo - Reduce

Calcular el ingreso total por producto.

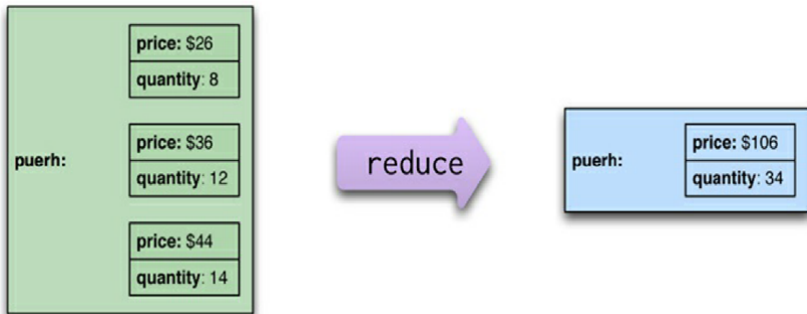


Figura de Sadalage y Fowler / NoSQL distilled : a brief guide to the emerging world of polyglot persistence

Ejemplo - Paralelismo

Calcular el ingreso total por producto.

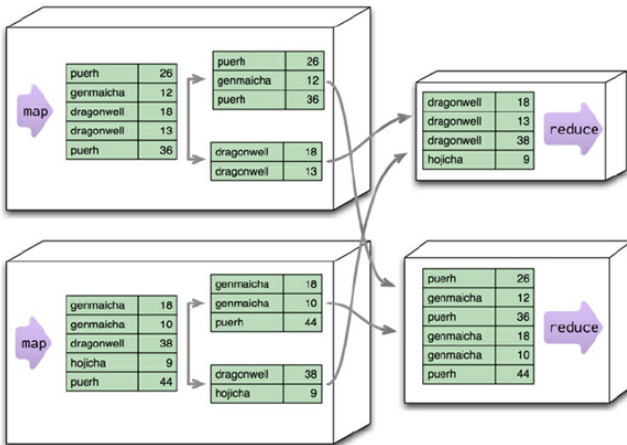


Figura de Sadalage y Fowler / NoSQL distilled : a brief guide to the emerging world of polyglot persistence

Ejemplo - Combinable Reduce

Calcular el ingreso total por producto. ¿Se puede optimizar más?

Ejemplo - Combinable Reduce

Calcular el ingreso total por producto.

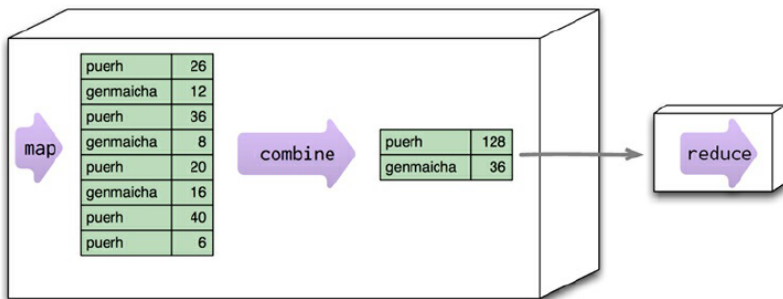


Figura de Sadalage y Fowler / NoSQL distilled : a brief guide to the emerging world of polyglot persistence

Implementar la *Junta* del Algebra Relacional

Bar	Cerveza	Precio
Moe	Duff	25
Cito	Quilmes	35
Joe's	Miller	27.5
Joe's	Bud	25

Vende

Bar	Dirección
Moe	Maple St
Cito	River Rd.
Joe's	Balcarce 50

Bar

Bar	Cerveza	Precio	Dirección
Moe	Duff	25	Maple St
Cito	Quilmes	35	River Rd.
Joe's	Miller	27.5	Balcarce 50
Joe's	Bud	25	Balcarce 50

$Vende \bowtie Bar$

Implementar la *Junta* del Algebra Relacional -Map

Bar	Cerveza	Precio
Moe	Duff	25
Cito	Quilmes	35
Joe's	Miller	27.5
Joe's	Bud	25

Vende

(Bar, Moe, Maple St)

(Bar, Cito, River Rd.)

(Bar, Joe's, Balcarce 50)

(Vende, Moe, Duff, 25)

(Vende, Cito, Quilmes, 35)

(Vende, Joe's, Miller, 27.5)

(Vende, Joe's, Bud, 25)

Bar	Dirección
Moe	Maple St
Cito	River Rd.
Joe's	Balcarce 50

Bar

Implementar la *Junta* del Algebra Relacional -Map

Bar	Cerveza	Precio
Moe	Duff	25
Cito	Quilmes	35
Joe's	Miller	27.5
Joe's	Bud	25

Vende

Bar	Dirección
Moe	Maple St
Cito	River Rd.
Joe's	Balcarce 50

Bar

(Bar, Moe, Maple St)

(Bar, Cito, River Rd.)

(Bar, Joe's, Balcarce 50)

(Vende, Moe, Duff, 25)

(Vende, Cito, Quilmes, 35)

(Vende, Joe's, Miller, 27.5)

(Vende, Joe's, Bud, 25)

Implementar la *Junta* del Algebra Relacional -Map

Bar	Cerveza	Precio
Moe	Duff	25
Cito	Quilmes	35
Joe's	Miller	27.5
Joe's	Bud	25

Vende

(Bar, Moe, Maple St)

(Bar, Cito, River Rd.)

(Bar, Joe's, Balcarce 50)

(Vende, Moe, Duff, 25)

(Vende, Cito, Quilmes, 35)

(Vende, Joe's, Miller, 27.5)

(Vende, Joe's, Bud, 25)

Bar	Dirección
Moe	Maple St
Cito	River Rd.
Joe's	Balcarce 50

Bar

Implementar la *Junta* del Algebra Relacional -Map

Bar	Cerveza	Precio
Moe	Duff	25
Cito	Quilmes	35
Joe's	Miller	27.5
Joe's	Bud	25

Vende

(Bar, Moe, Maple St)

(Bar, Cito, River Rd.)

(Bar, Joe's, Balcarce 50)

(Vende, Moe, Duff, 25)

(Vende, Cito, Quilmes, 35)

(Vende, Joe's, Miller, 27.5)

(Vende, Joe's, Bud, 25)

Bar	Dirección
Moe	Maple St
Cito	River Rd.
Joe's	Balcarce 50

Bar

Implementar la *Junta* del Algebra Relacional -Map

Bar	Cerveza	Precio
Moe	Duff	25
Cito	Quilmes	35
Joe's	Miller	27.5
Joe's	Bud	25

Vende

(Bar, Moe, Maple St)

(Bar, Cito, River Rd.)

(Bar, Joe's, Balcarce 50)

(Vende, Moe, Duff, 25)

(Vende, Cito, Quilmes, 35)

(Vende, Joe's, Miller, 27.5)

(Vende, Joe's, Bud, 25)

Bar	Dirección
Moe	Maple St
Cito	River Rd.
Joe's	Balcarce 50

Bar

Implementar la *Junta* del Algebra Relacional -Map

Bar	Cerveza	Precio
Moe	Duff	25
Cito	Quilmes	35
Joe's	Miller	27.5
Joe's	Bud	25

Vende

(Bar, Moe, Maple St)

(Bar, Cito, River Rd.)

(Bar, Joe's, Balcarce 50)

(Vende, Moe, Duff, 25)

(Vende, Cito, Quilmes, 35)

(Vende, Joe's, Miller, 27.5)

(Vende, Joe's, Bud, 25)

Bar	Dirección
Moe	Maple St
Cito	River Rd.
Joe's	Balcarce 50

Bar

Implementar la *Junta* del Algebra Relacional -Map

Bar	Cerveza	Precio
Moe	Duff	25
Cito	Quilmes	35
Joe's	Miller	27.5
Joe's	Bud	25

Vende

(Bar, Moe, Maple St)

(Bar, Cito, River Rd.)

(Bar, Joe's, Balcarce 50)

(Vende, Moe, Duff, 25)

(Vende, Cito, Quilmes, 35)

(Vende, Joe's, Miller, 27.5)

(Vende, Joe's, Bud, 25)

Bar	Dirección
Moe	Maple St
Cito	River Rd.
Joe's	Balcarce 50

Bar

Implementar la *Junta* del Algebra Relacional - Map

(Bar, Moe, Maple St)
(Bar, Cito, River Rd.)
(Bar, Joe's, Balcarce 50)
(Vende, Moe, Duff, 25)
(Vende, Cito, Quilmes, 35)
(Vende, Joe's, Miller, 27.5)
(Vende, Joe's, Bud, 25)

key= Moe **values**= (Bar, Moe, Maple St)
key= Cito **values**= (Bar, Cito, River Rd.)
key= Joe's **values**= (Bar, Joe's, Balcarce 50)
key= Moe **values**= (Vende, Moe, Duff, 25)
key= Cito **values**= (Vende, Cito, Quilmes, 35)
key= Joe's **values**= (Vende, Joe's, Miller, 27.5)
key= Joe's **values**= (Vende, Joe's, Bud, 25)

Junta. ¿Que hace el Reduce?

key= Cito **values**=[(Bar, Cito, River Rd.), (Vende, Cito, Quilmes, 35)]

key= Joe's **values**=[(Bar, Joe's, Balcarce 50), (Vende, Joe's, Miller, 27.5), (Vende, Joe's, Bud, 25)]

Junta. ¿Que hace el Reduce?

key= Cito **values**=[(Bar, Cito, River Rd.), (Vende, Cito, Quilmes, 35)]




(Cito, Quilmes, 35, River Rd)

key= Joe's **values**=[(Bar, Joe's, Balcarce 50), (Vende, Joe's, Miller, 27.5), (Vende, Joe's, Bud, 25)]


Junta. ¿Que hace el Reduce?

key= Cito **values**=[(Bar, Cito, River Rd.), (Vende, Cito, Quilmes, 35)]



(Cito, Quilmes, 35, River Rd)

key= Joe's **values**=[(Bar, Joe's,Balcarce 50),(Vende, Joe's, Miller, 27.5),(Vende, Joe's, Bud, 25)]



(Joe's, Miller, 27.5, Balcarce 50)
(Joe's, Bud, 25, Balcarce 50)

Junta Resumen

Sean las relaciones $R(a, b)$ y $S(b, c)$ computar $R \bowtie S$

Función Map

Para cada tupla (a, b) de R producir el par clave-valor $\langle b, (R, a) \rangle$
Para cada tupla (b, c) de S producir el par clave-valor $\langle b, (S, c) \rangle$

Junta Resumen

Sean las relaciones $R(a, b)$ y $S(b, c)$ computar $R \bowtie S$

Función Map

Para cada tupla (a, b) de R producir el par clave-valor $\langle b, (R, a) \rangle$
Para cada tupla (b, c) de S producir el par clave-valor $\langle b, (S, c) \rangle$

Función Reduce

Para cada valor de b Recibe una lista de pares $\langle (R, a), (S, c) \rangle$
Produce para cada entrada un valor de la forma (a, b, c) , La clave es irrelevante.