

(Big) Data Engineering In Depth

From Beginner to Professional

Moustafa Alaa

Senior Data Engineer

The Definitive Guide to Big Data Engineering Tasks

Previous video recap!

Map Reduce Components

The Map-Reduce consists of three "main" parts

- The Driver.

Map Reduce Components

The Map-Reduce consists of three "main" parts

- The Driver.
- The Mapper.

Map Reduce Components

The Map-Reduce consists of three "main" parts

- The Driver.
- The Mapper.
- The Reducer.

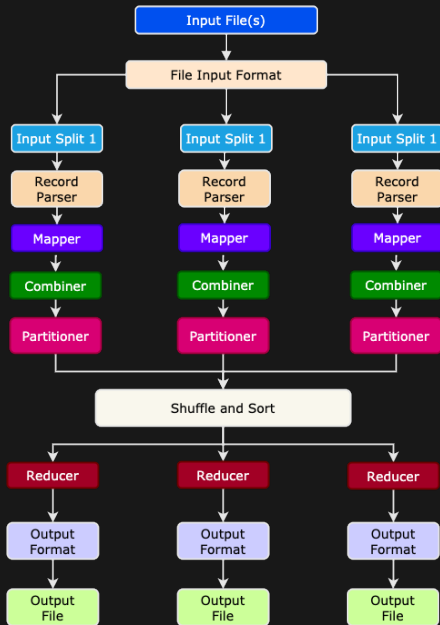


Figure: Map Reduce Stages

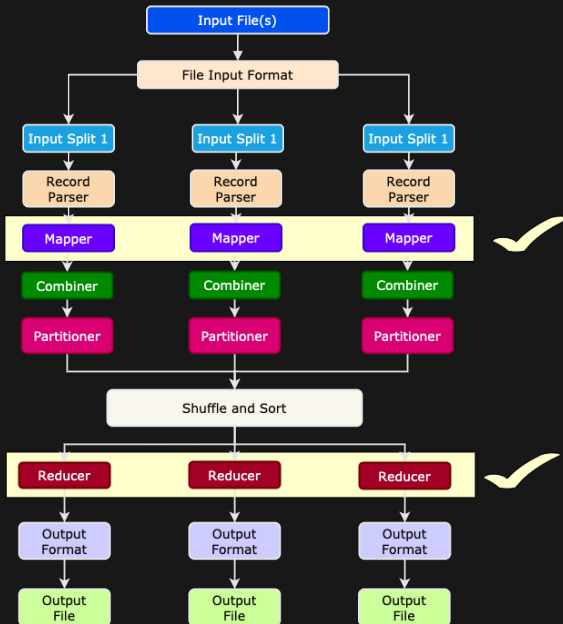


Figure: Map Reduce Stages

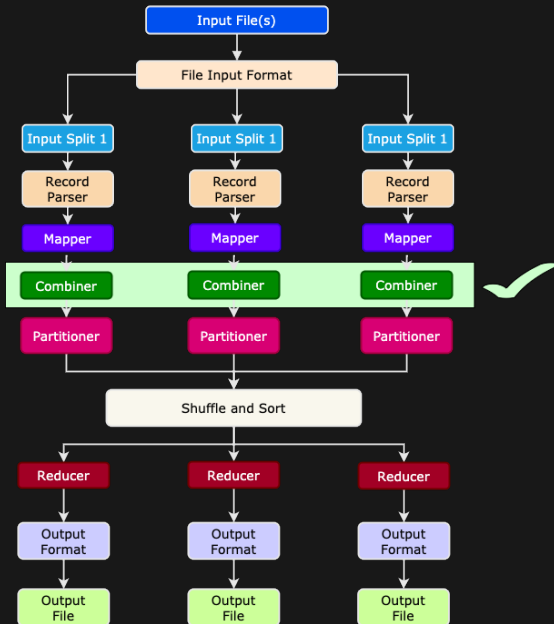


Figure: Map Reduce Stages

The Combiners

Increase The Map-Reduce Processing Using
The Combiners

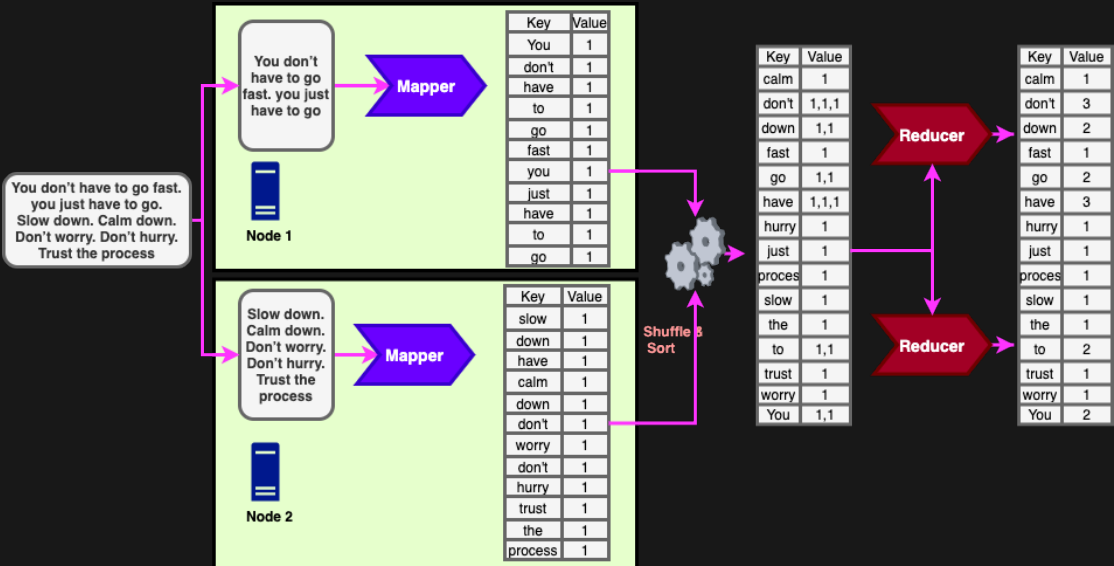


Figure: Map Reduce Without Combiner

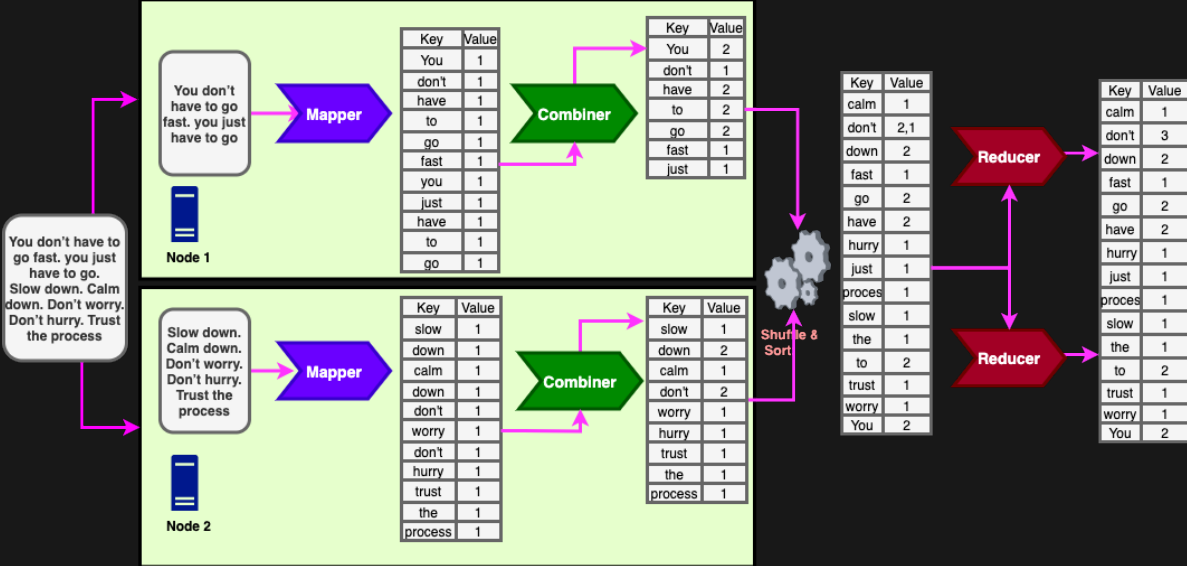


Figure: Map Reduce Without Combiner

Combiners Motivation

- Moving data between nodes is one of the bottlenecks in distributed systems.

Combiners Motivation

- Moving data between nodes is one of the bottlenecks in distributed systems.
- We need to find always solutions to reduce the amount of data movement in the network.

Combiners Motivation

- Moving data between nodes is one of the bottlenecks in distributed systems.
- We need to find always solutions to reduce the amount of data movement in the network.
- In most cases, mappers produce large amounts of intermediate data passed on to the reducers for further processing. This leads to enormous network congestion.

Combiners Motivation

- Moving data between nodes is one of the bottlenecks in distributed systems.
- We need to find always solutions to reduce the amount of data movement in the network.
- In most cases, mappers produce large amounts of intermediate data passed on to the reducers for further processing. This leads to enormous network congestion.
- One of the solutions is to reduce the mapper output using combiners. using **combiners "mini-reducer"**.

How Combiner Works?

- The combiner must implement the **Reducer** interface's **reduce()** method.

How Combiner Works?

- The combiner must implement the **Reducer** interface's **reduce()** method.
- The combiner process on each map output key & value (same as the Reducer).

Combiners and Reducers

Combiner and Reducer code are often identical.

- Combiner & Reducer must have identical input and output data types.

Combiners and Reducers

Combiner and Reducer code are often identical.

- Combiner & Reducer must have identical input and output data types.
- The operation must be commutative and associative.

Combiners and Reducers

- ***Combiner runs on the Map-Side.***

Combiners and Reducers

- ***Combiner runs on the Map-Side.***
- The output of the Combiner passed to the Reducer.

Associative and Commutative

- In math, the associative and commutative properties are laws applied to addition and multiplication that always exist.

¹This example taken from <https://sciencing.com/associative-commutative-property-of-addition-multiplication-with-examples-13712459.html>

Associative and Commutative

- In math, the associative and commutative properties are laws applied to addition and multiplication that always exist.

¹This example taken from <https://sciencing.com/associative-commutative-property-of-addition-multiplication-with-examples-13712459.html>

Associative and Commutative

- The associative property states that you can re-group numbers and you will get the same answer.

$$a + (b + c) = (a + b) + c$$
$$1 + (2 + 3) = (1 + 2) + 3$$

$$a \times (b \times c) = (a \times b) \times c$$
$$1 \times (2 \times 3) = (1 \times 2) \times 3$$

¹This example taken from <https://sciencing.com/associative-commutative-property-of-addition-multiplication-with-examples-13712459.html>

Associative and Commutative

- The commutative property states that you can move numbers around and still arrive at the same answer.

$$a + b = b + a$$

$$1 + 2 = 2 + 1$$

$$a \times b = b \times a$$

$$1 \times 2 = 2 \times 1$$

¹This example taken from <https://sciencing.com/associative-commutative-property-of-addition-multiplication-with-examples-13712459.html>

Combiners and Reducers

- Reducers may be used as Combiners if the operation is associative and commutative.

Combiners and Reducers



Mapper 1

(The,1)
(The,1)
(The,1)



Sum (The,1,1,1)



Mapper 2

(The,1)
(The,1)



Sum (The,1,1)

Sum(The,3,2)



(The,5)

=

Sum (The,1,1,1,1,1)

(The,5)

Figure: Associative and Commutative Example

Combiners and Reducers



Mapper 1

(The,0.3)
(The,0.4)
(The,0.7)



AVG (The,0.3,0.4,0.7)



Mapper 2

(The,0.5)
(The,0.6)



AVG (The,0.5,0.6)

AVG (The,0.467,0.55)



AVG (The,0.5085)

≠

AVG (The,0.3,0.4,0.5,0.6,0.7)

AVG (The,0.5)

Figure: Associative and Commutative Example

Thank you for watching!

See you in the next video 😊