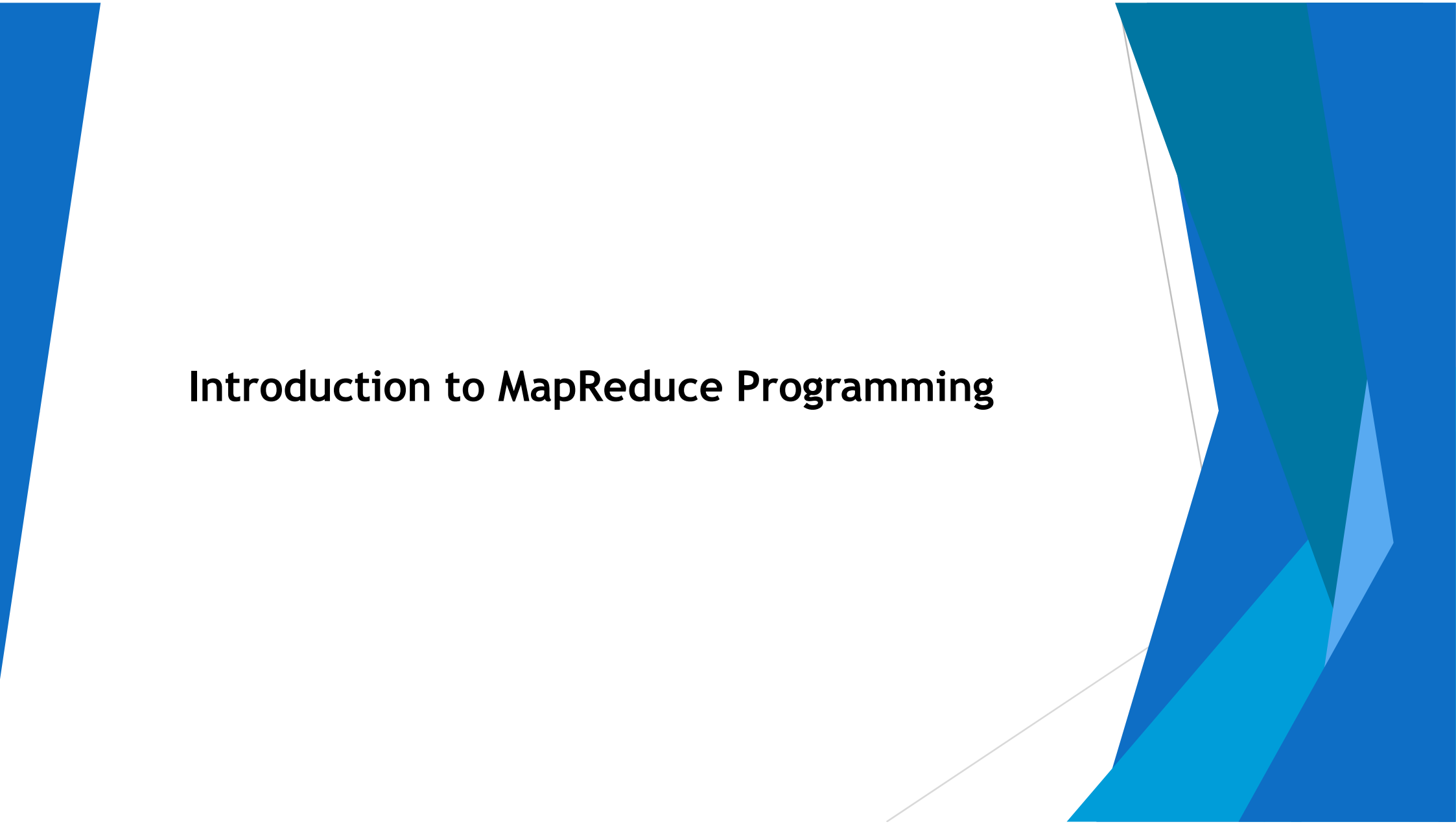
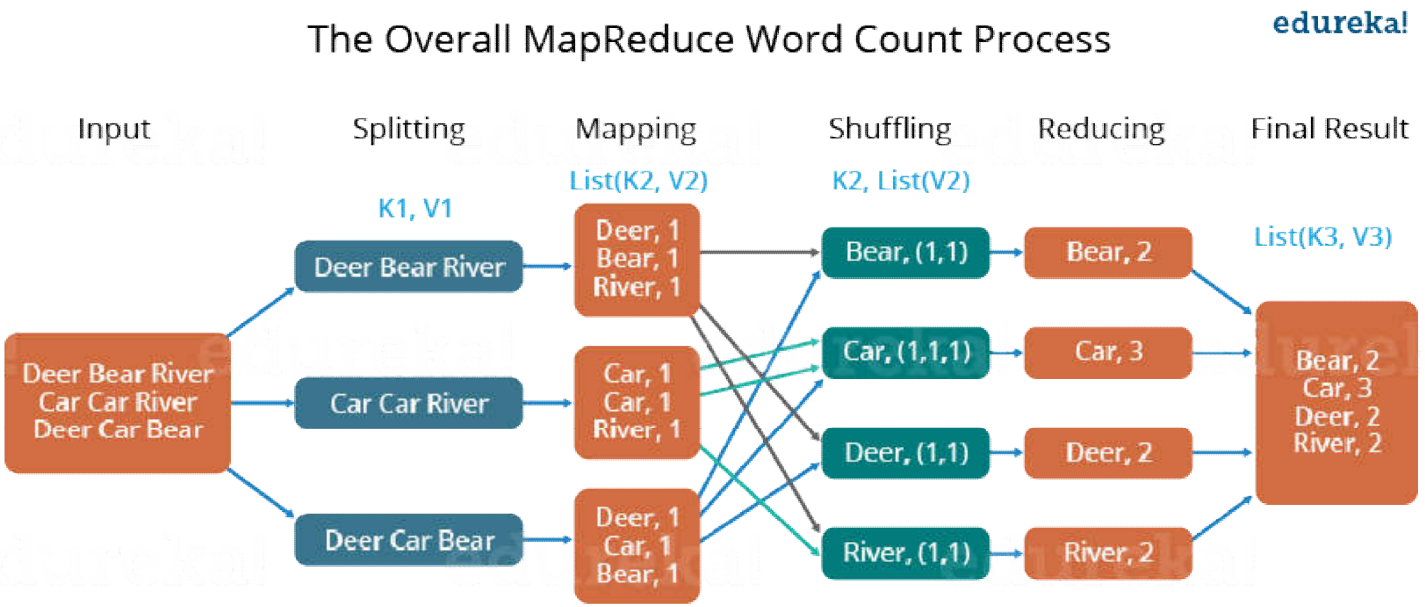
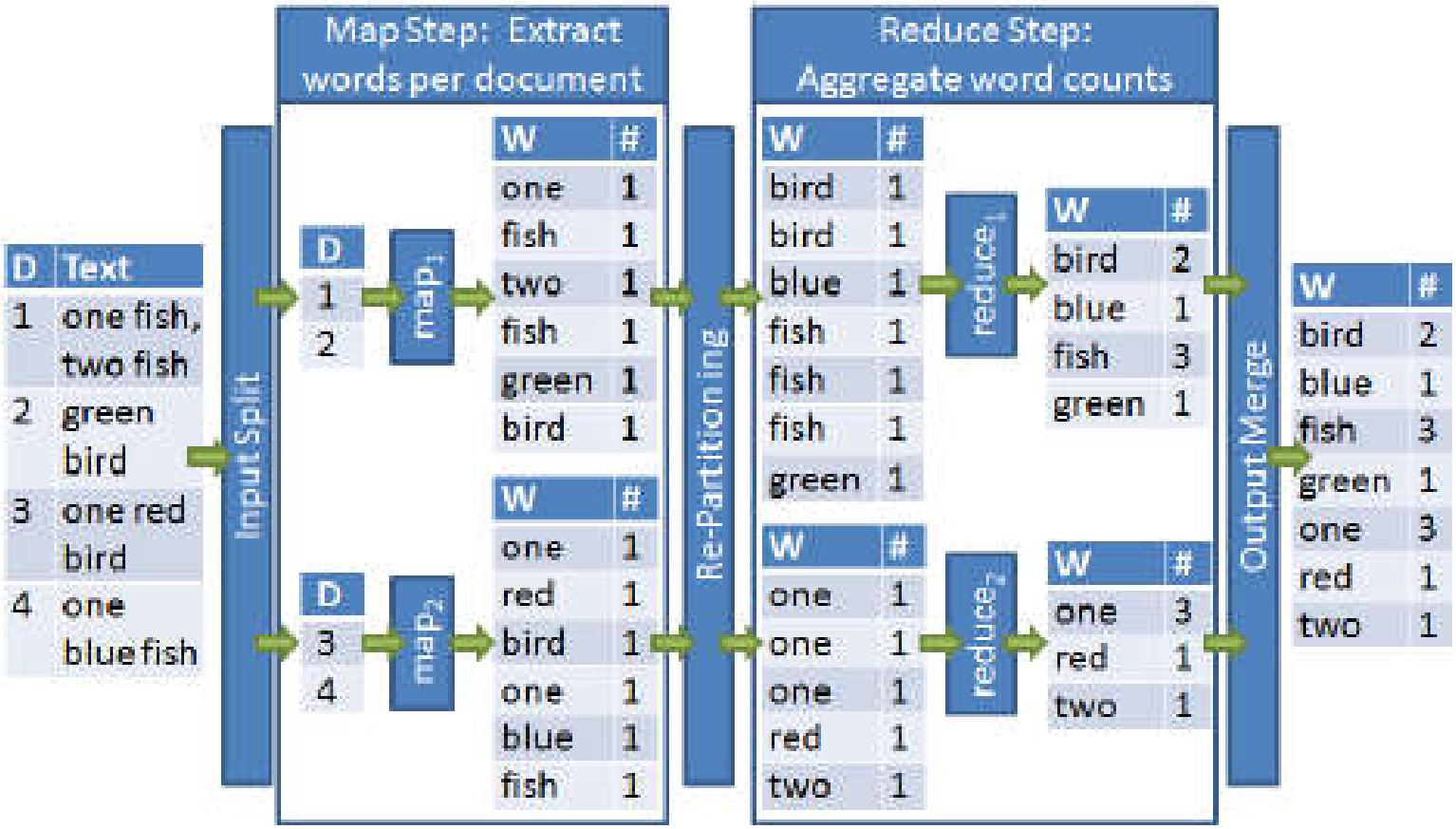
Introduction to MapReduce Programming




The Overall MapReduce Word Count Process
edureka!
Input
Splitting
Mapping
Shuffling
Reducing
Final Result
K1, V1
List(K2, V2)
K2, List(V2)
List(K3, V3)
Deer Bear River
Car Car River
Deer Car Bear
Deer Bear River
Car Car River
Deer Car Bear
Deer, 1
Bear, 1
River, 1
Car, 1
Car, 1
River, 1
Deer, 1
Car, 1
Bear, 1
Bear, (1,1)
Car, (1,1,1)
Deer, (1,1)
River, (1,1)
Bear, 2
Car, 3
Deer, 2
River, 2
Bear, 2
Car, 3
Deer, 2
River, 2


Map Step: Extract words per document
Reduce Step: Aggregate word counts
Input Split
Re-Partitioning
Output Merge
map1
map2
reduce1
reduce2
D
Text
1
one fish, two fish
2
green bird
3
one red bird
4
one blue fish
W
#
one 1
fish 1
two 1
fish 1
green 1
bird 1
one 1
red 1
bird 1
one 1
blue 1
fish 1
bird 2
blue 1
fish 3
green 1
one 3
red 1
two 1

Introduction

In MapReduce Programming, Jobs (Applications) are split into a set of map tasks and reduce tasks. Then these tasks are executed in a distributed fashion on Hadoop cluster.

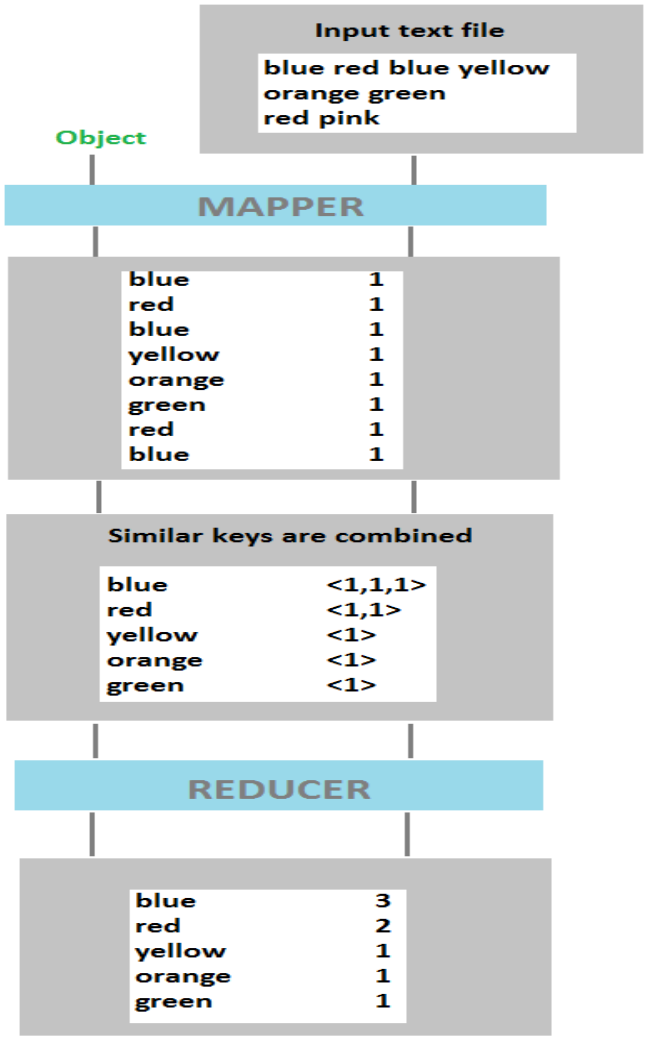
Each task processes small subset of data that has been assigned to it. This way, Hadoop distributes the load across the cluster.

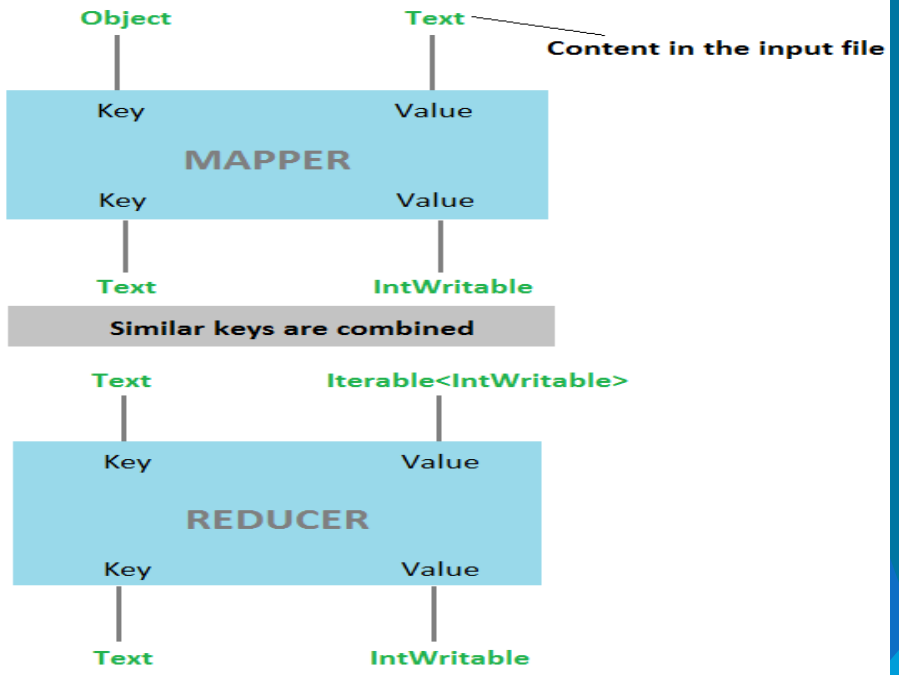
MapReduce job takes a set of files that is stored in HDFS (Hadoop Distributed File System) as input.

Map task takes care of loading, transforming, parsing and filtering

Reduce task is responsible for grouping and aggregation.

Word Count Program





Mapper

Mapper

A mapper maps the input key–value pairs into a set of intermediate key-value pairs. Maps are individual tasks that have the responsibility of transforming input records into intermediate key-value pairs.

Mapper Consists of following phases:

- **RecordReader**
- **Map**
- **Combiner**
- **Partitioner**

Reducer

Reducer

The primary chore of the Reducer is to reduce a set of intermediate values (the ones that share a common key) to a smaller set of values.

The Reducer has three primary phases:

- Shuffle
- Sort
- Reduce
- Output Format.

- Hadoop assigns map tasks to Data Node where the actual data to be processed.
- Hadoop ensures data locality.(moving code not data).

Mapper

Record Reader:

1.It coverts a byte- oriented view of the input into record –oriented view and presents to mapper tasks
2.It presents task with key- value pair
3. Key: positional information
4. Value is chunk of data that constitutes the record.

Map:

1. works on the key- value pair
2.generates zero or more intermediate key- value pairs

Combiner:

1. It is an optional function , but it provides high performance.(bandwidth and disk space)
2. It takes intermediate key- value pair and applies user specified aggregate function only on that mapper
3. It is also known as local reducer.

Mapper

Partitioner:

1.It takes the intermediate key-value pair and splits them into shard, sends the shard to the particular reducer as per the user specific code.

2. Key with same value goes to the same reducer.

3. Partitioner controls the partitioning of the keys of the intermediate map-outputs.

4. The key (or a subset of the key) is used to derive the partition, typically by a hash function.

5. The total number of partitions is the same as the number of reduce tasks for the job.

6. Controls which of the m reduce tasks the intermediate key (and hence the record) is sent for reduction.

Reducer

Shuffle and Sort:

1. This phase takes the output of all Partitioner and downloads them into the local machine where the reducer is running.
2. These individual data pipes are sorted by key.
3. Main purpose of sort is grouping similar words so that their values can be easily iterated by reduce task.

Reduce:

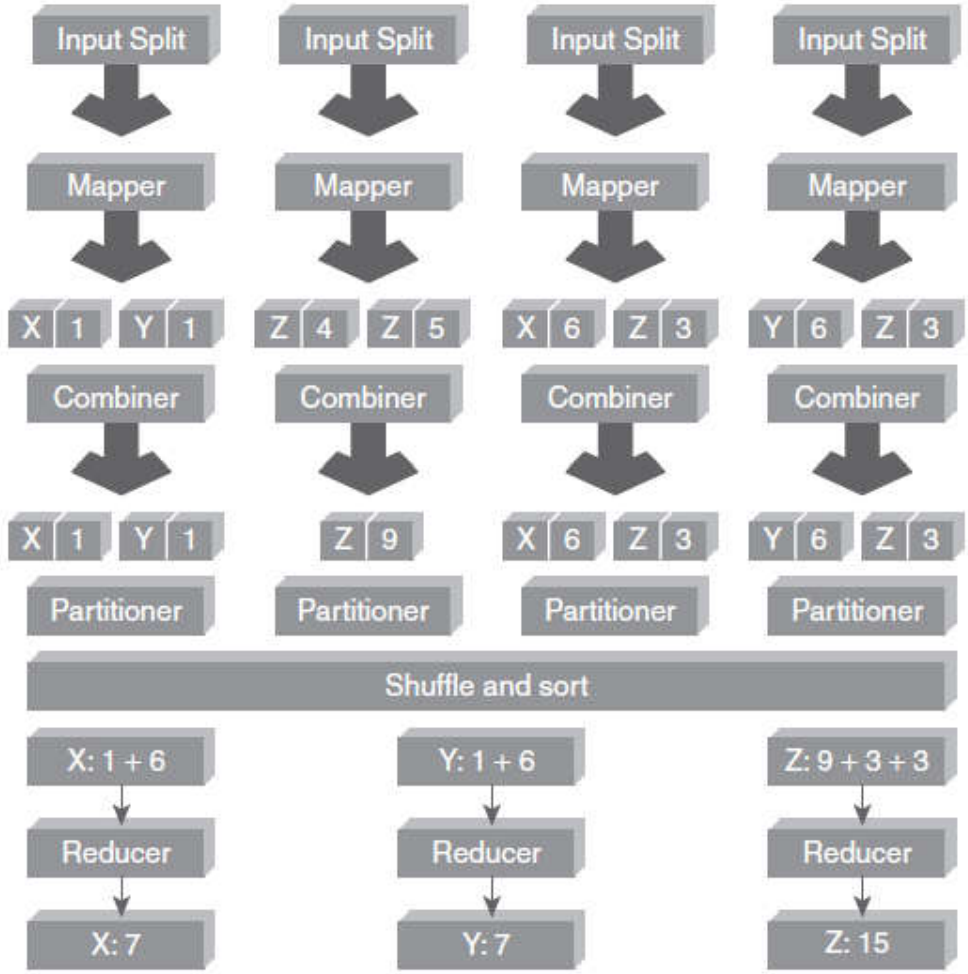
1. It takes the grouped data f, applies the reduce function, and processes one group at a time .
2. The reduce function iterates all the values associated with that key.
3. It provides various operations like aggregation, filtering and combining data.
4. Once it is done is sent to output format.

Output Format:

1. It separates key- value pair with tab and writes it out to a file using record write.

The chores of Mapper, Combiner, Partitioner, and Reducer

The chores of Mapper, Combiner, Partitioner, and Reducer



Combiner

Combiner

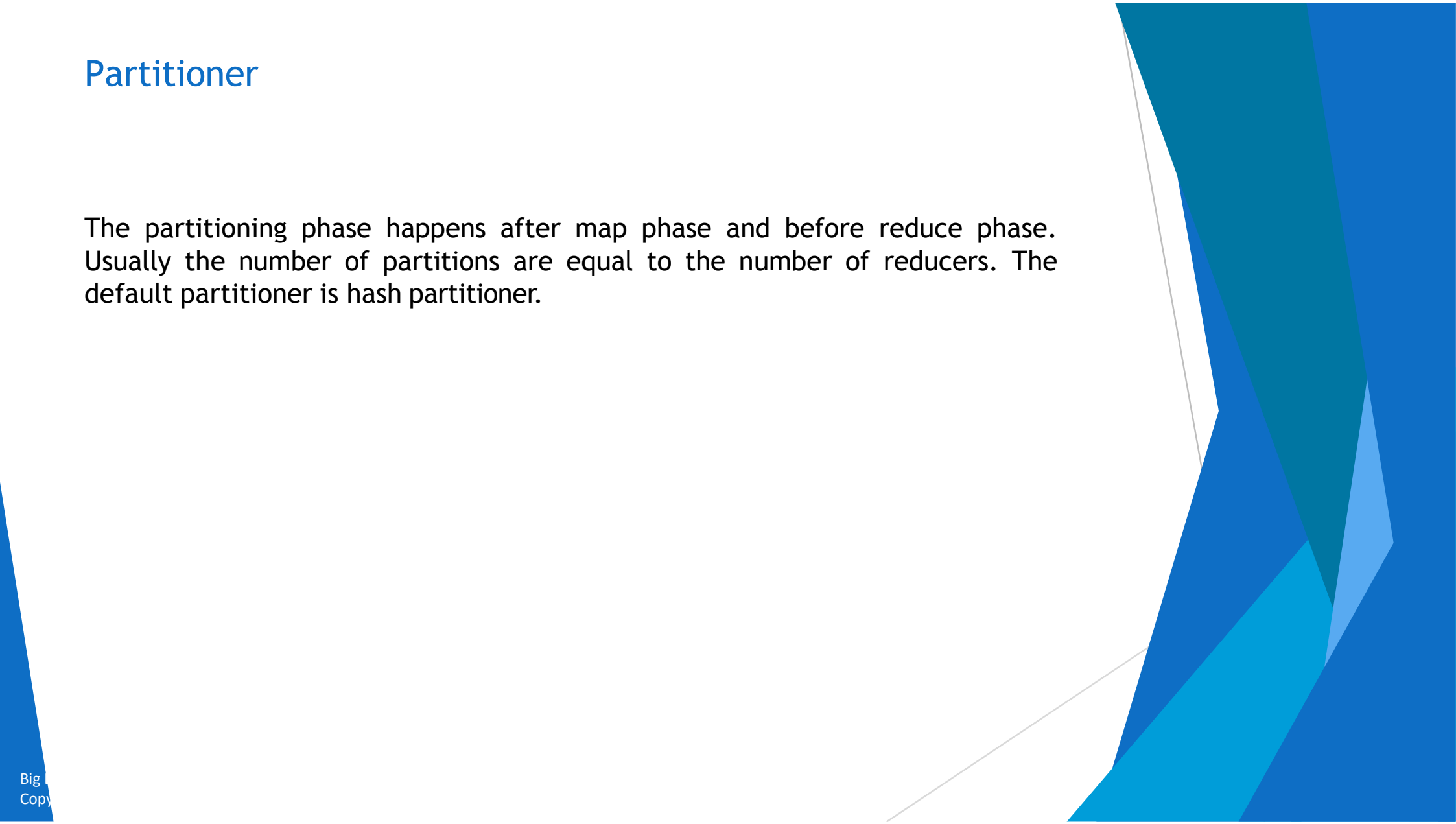
It is an optimization technique for MapReduce Job. Generally, the reducer class is set to be the combiner class. The difference between combiner class and reducer class is as follows:

- Output generated by combiner is intermediate data and it is passed to the reducer.
- Output of the reducer is passed to the output file on disk.

Partitioner

Partitioner

The partitioning phase happens after map phase and before reduce phase. Usually the number of partitions are equal to the number of reducers. The default partitioner is hash partitioner.



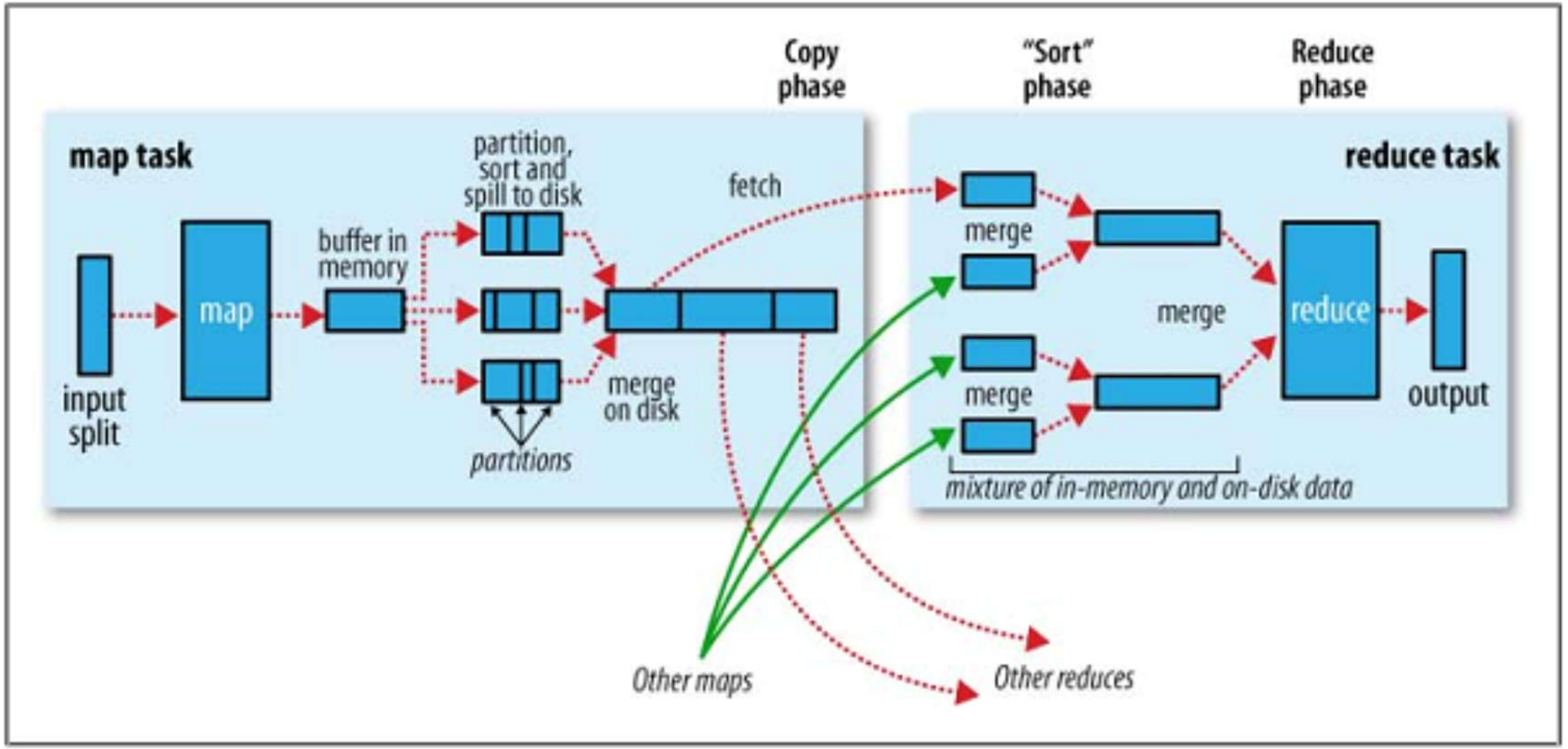


Figure 6-4. Shuffle and sort in MapReduce

Thank You