Hadoop

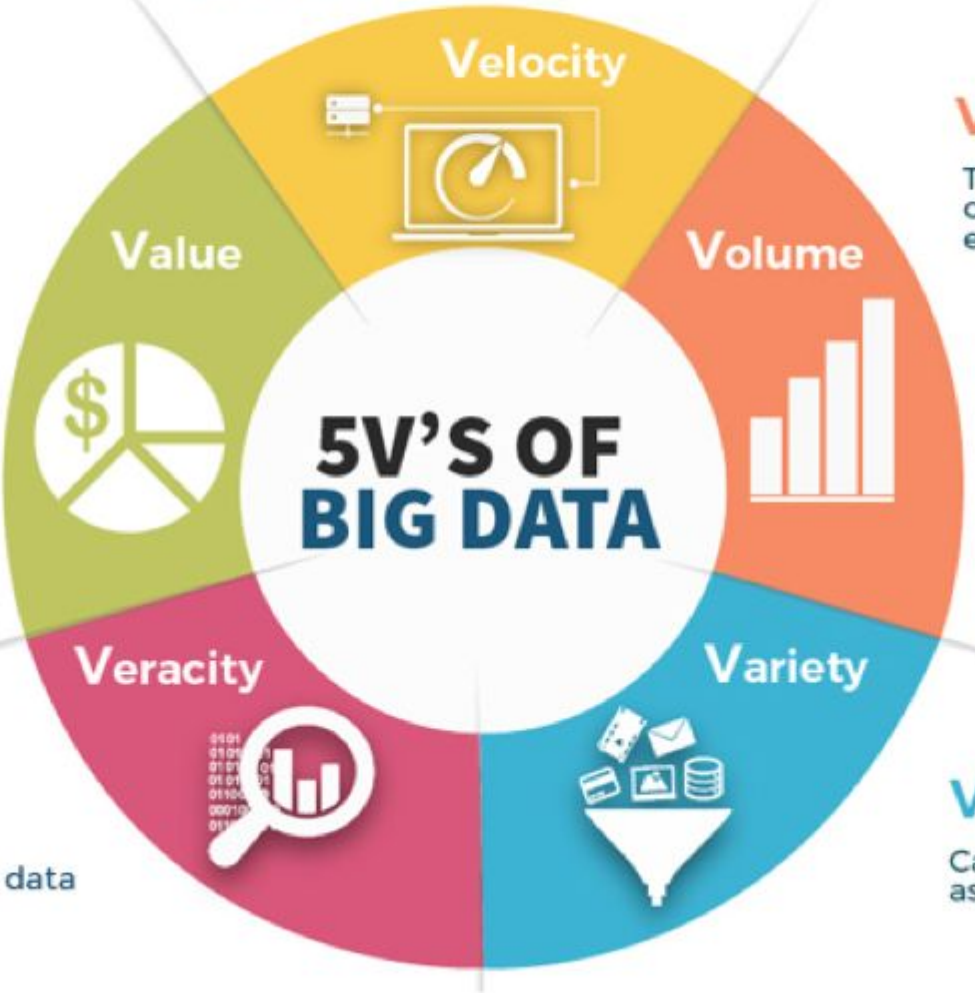
Types of Data

The following three types of data can be identified:

Structured data:
Data which is represented in a tabular format
E.g.: Databases

Semi-structured data:
Data which does not have a formal data model
E.g.: XML files

Unstructured data:
Data which does not have a pre-defined data model
E.g.: Text files



Velocity

Speed at which data is emanating and changes are occurring between the diverse data sets

Volume

This refers to the sheer volume of data being generated every second.

Value

Having access to big data is all well and good but that's only useful if we can turn it into a value.

Veracity

Data reliability and trust.
Verifying and validating the data

Variety

Can use structured as well as unstructured data.

Big Data Storage & Computation ?

Storing Big Data was a Problem

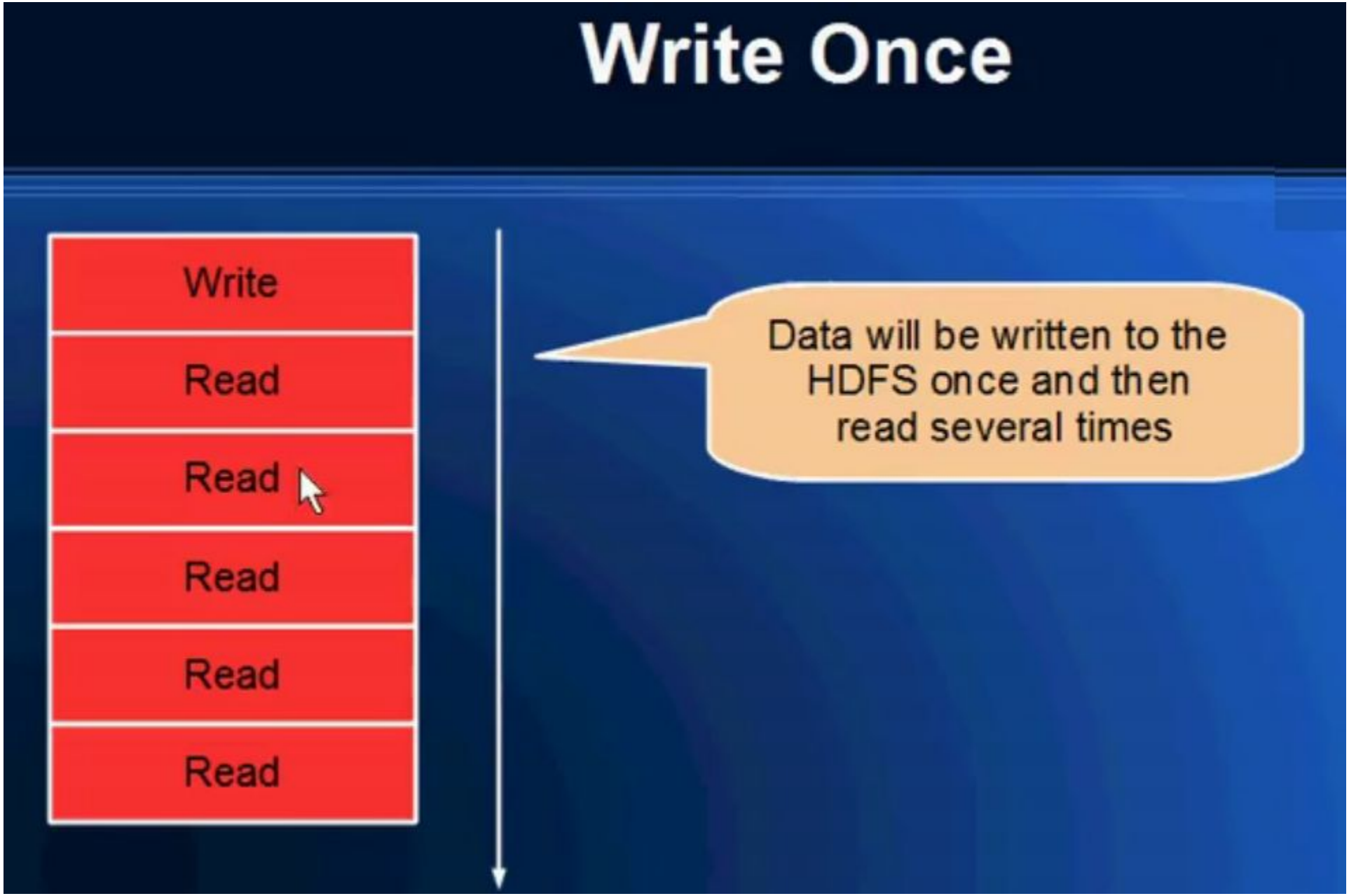
Even if a part of Big Data is Stored-
Processing it, took Years

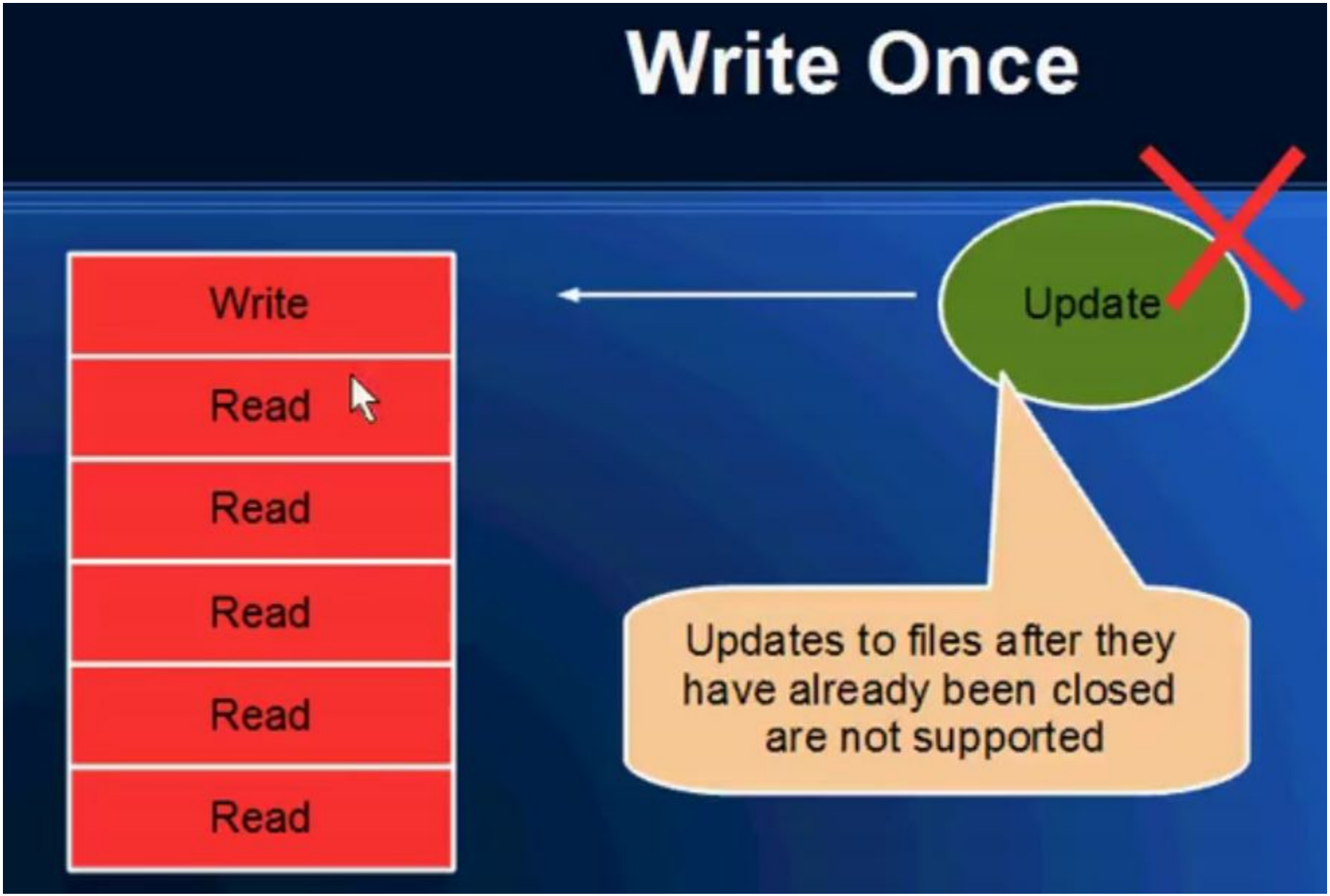
Hadoop Solves Big Data Problems

Storing Big Data was no more a Problem

And Processing did not take Years

Write Once


Write
Read
Read
Read
Read
Read
Read
Data will be written to the HDFS once and then read several times


Write Once
Write
Read
Read
Read
Read
Read
Read
Update
Updates to files after they have already been closed are not supported

HDFS Architecture

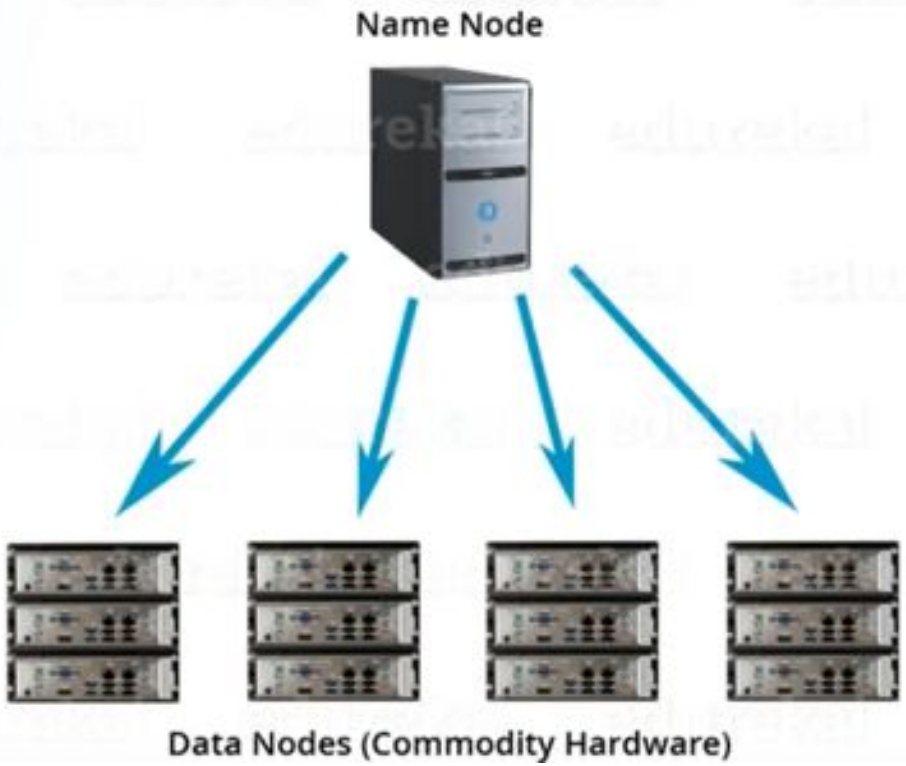
NameNode and DataNode

NameNode

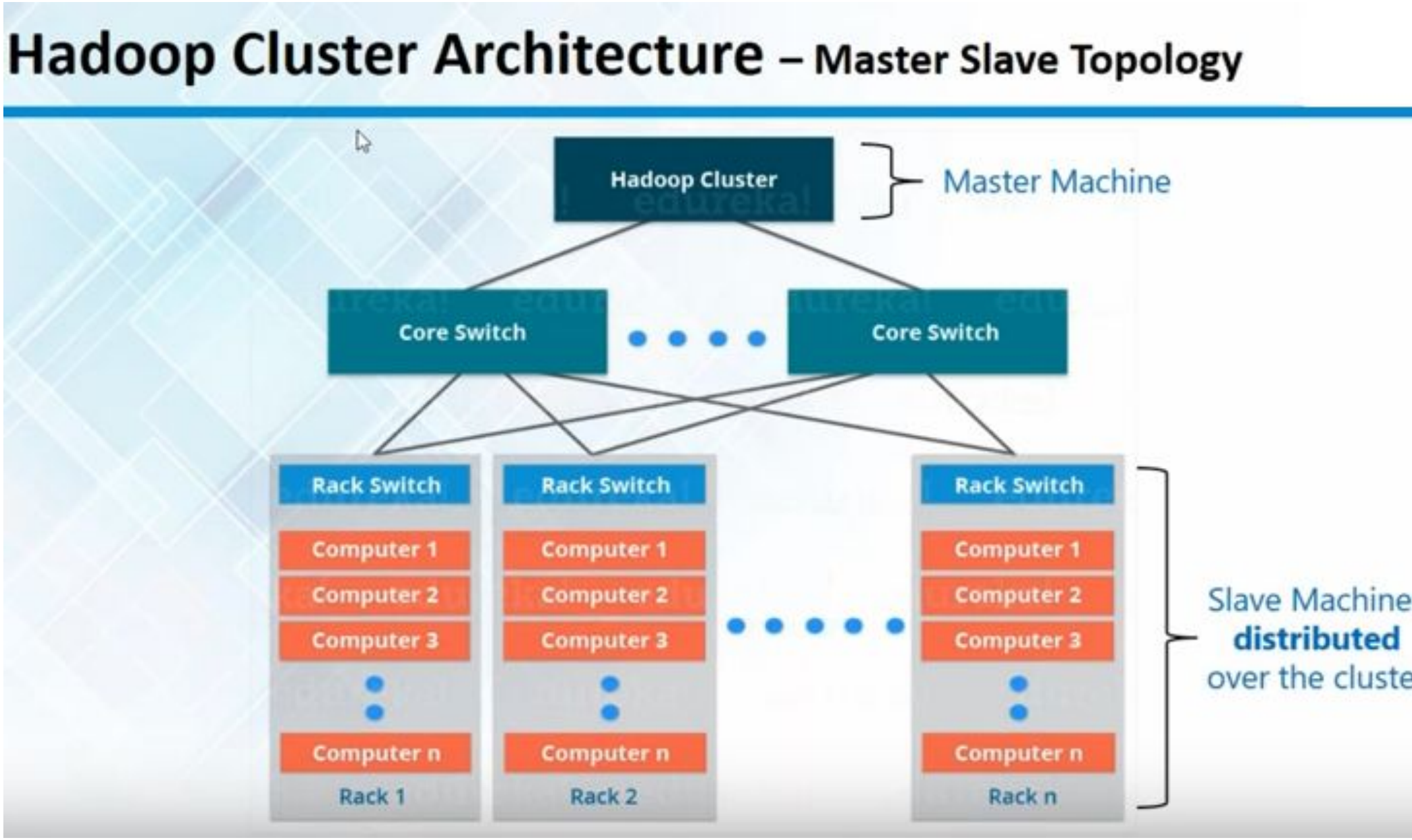
- Master daemon
- Maintains and Manages DataNodes
- Records metadata e.g. location of blocks stored, the size of the files, permissions, hierarchy, etc.
- Receives heartbeat and block report from all the DataNodes

DataNode

- Slave daemons
- Stores actual data
- Serves read and write requests from the clients



Hadoop Cluster Architecture – Master Slave Topology



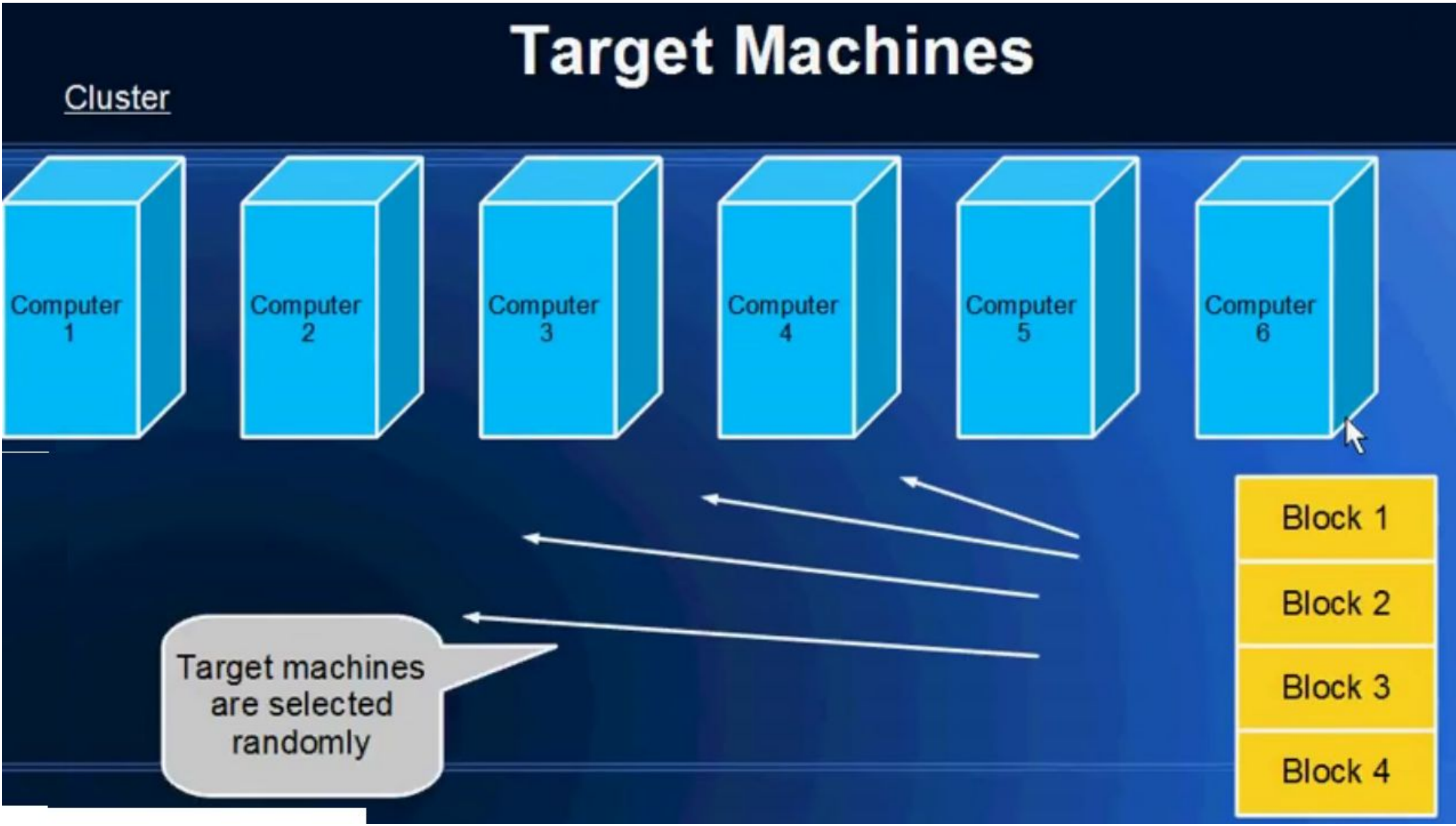
Hadoop cluster

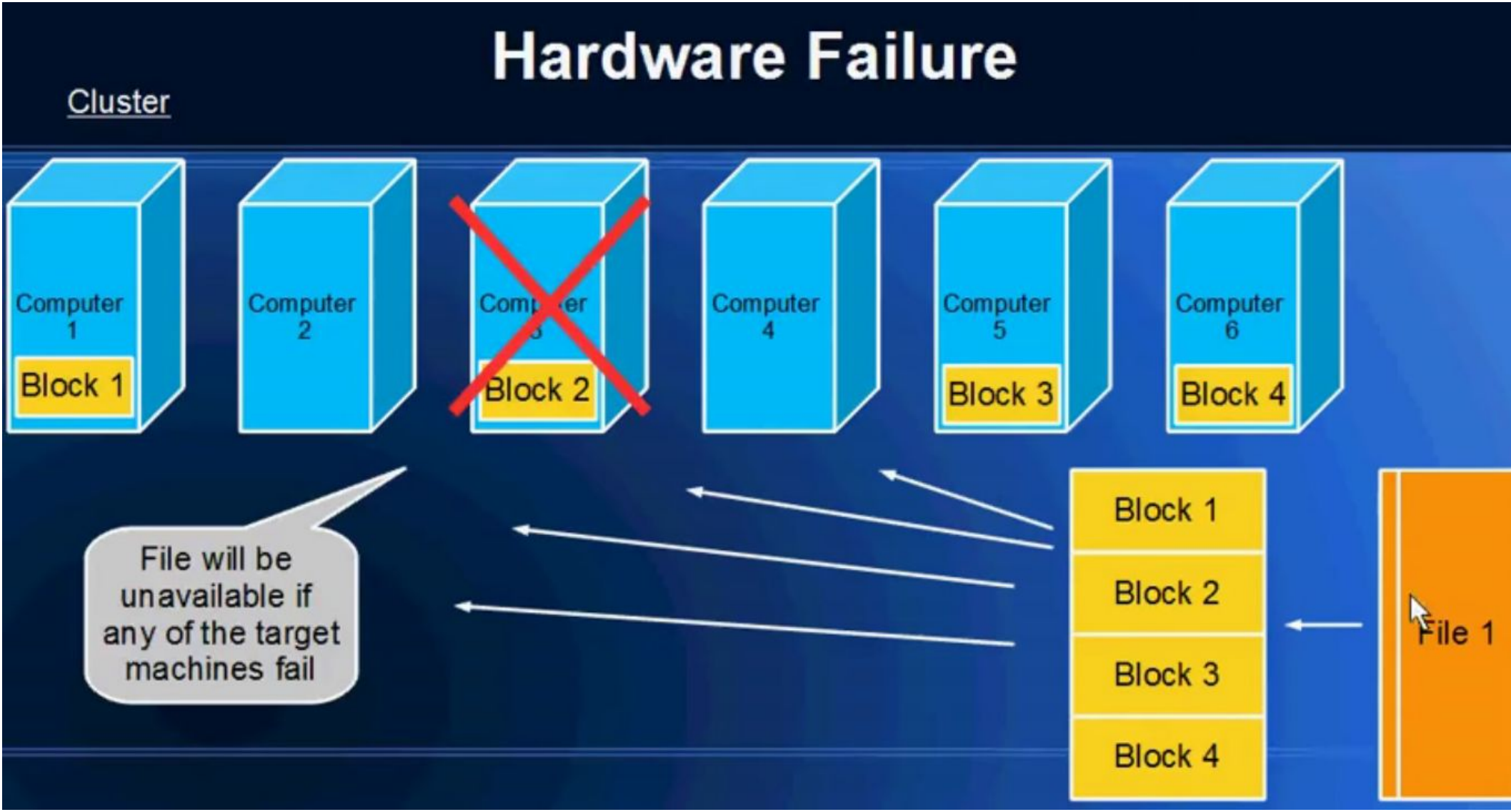
Cluster of machine running Hadoop at Yahoo!

HDFS block

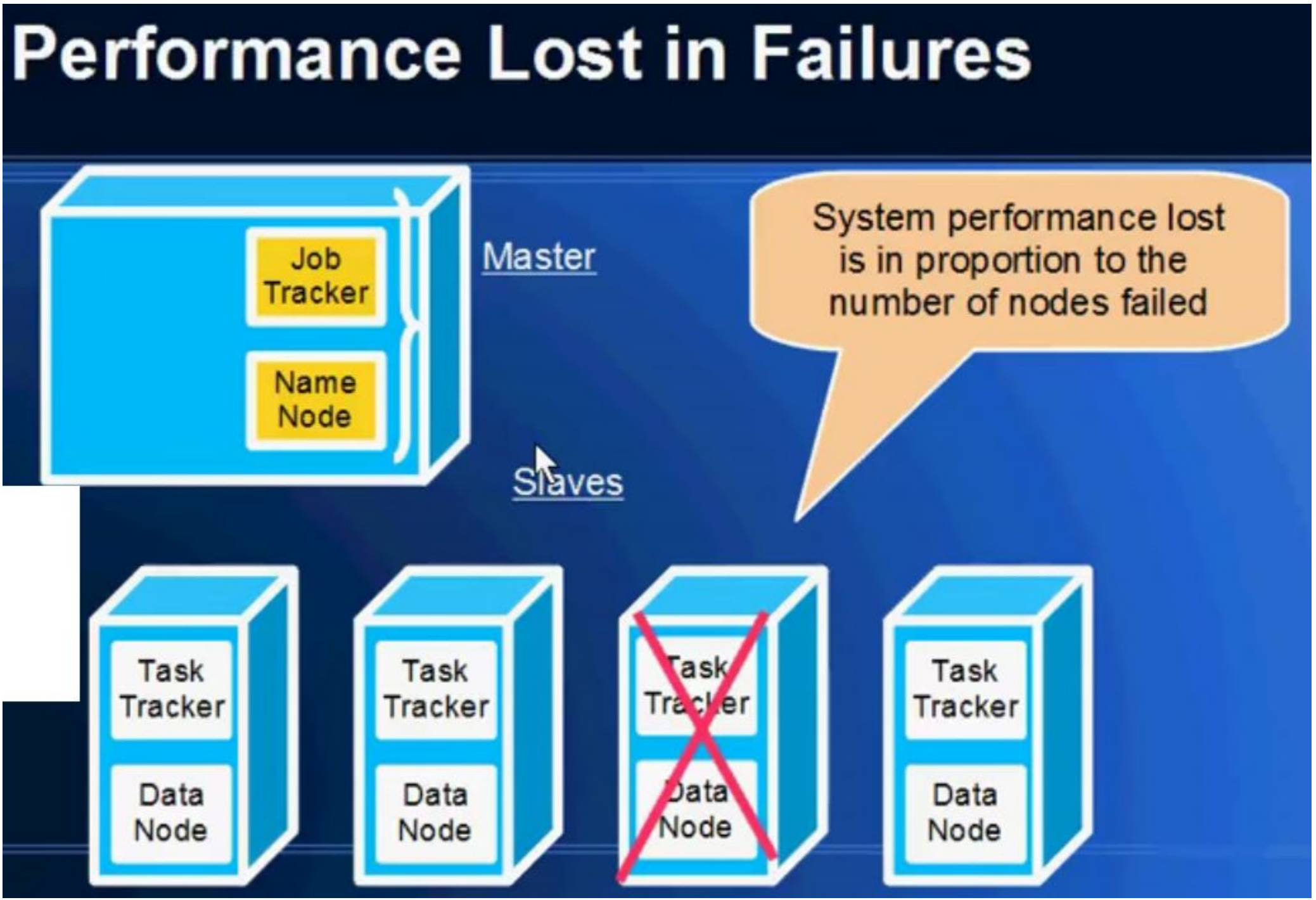
- Each file is stored on HDFS as blocks
- The default size of each block is **128 MB** in Apache Hadoop 2.x (64 MB in Apache Hadoop 1.x)
- Let us say, I have a file example.txt of size 248 MB. Below is the representation of how it will be stored on HDFS

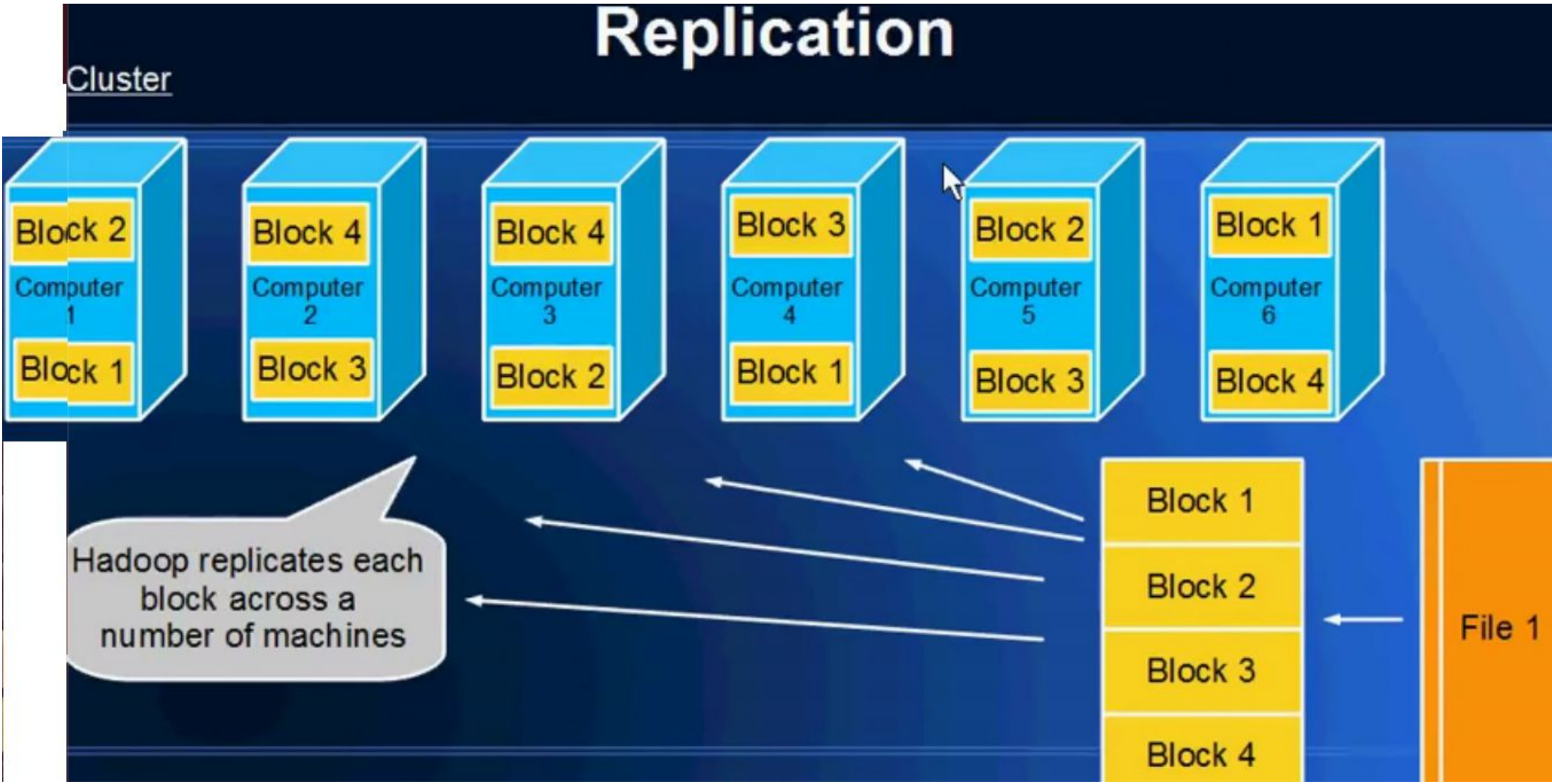
How many blocks will be created if a file of size 514 MB is copied to HDFS ?

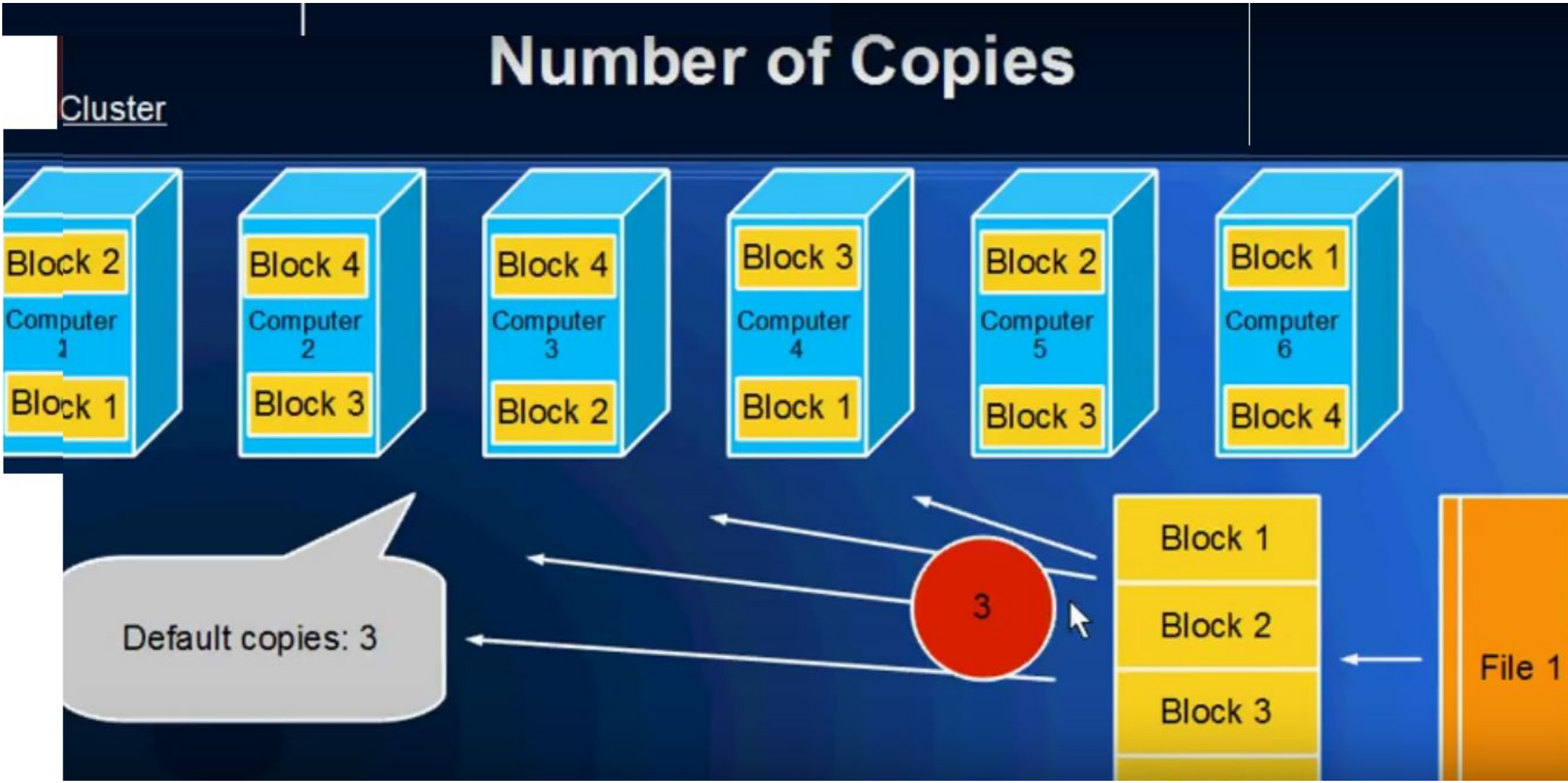

Target Machines
Cluster
Computer 1
Computer 2
Computer 3
Computer 4
Computer 5
Computer 6
Block 1
Block 2
Block 3
Block 4
Target machines are selected randomly


Hardware Failure
Cluster
Computer 1
Block 1
Computer 2
Computer 3
Block 2
Computer 4
Computer 5
Block 3
Computer 6
Block 4
File will be unavailable if any of the target machines fail
Block 1
Block 2
Block 3
Block 4
File 1

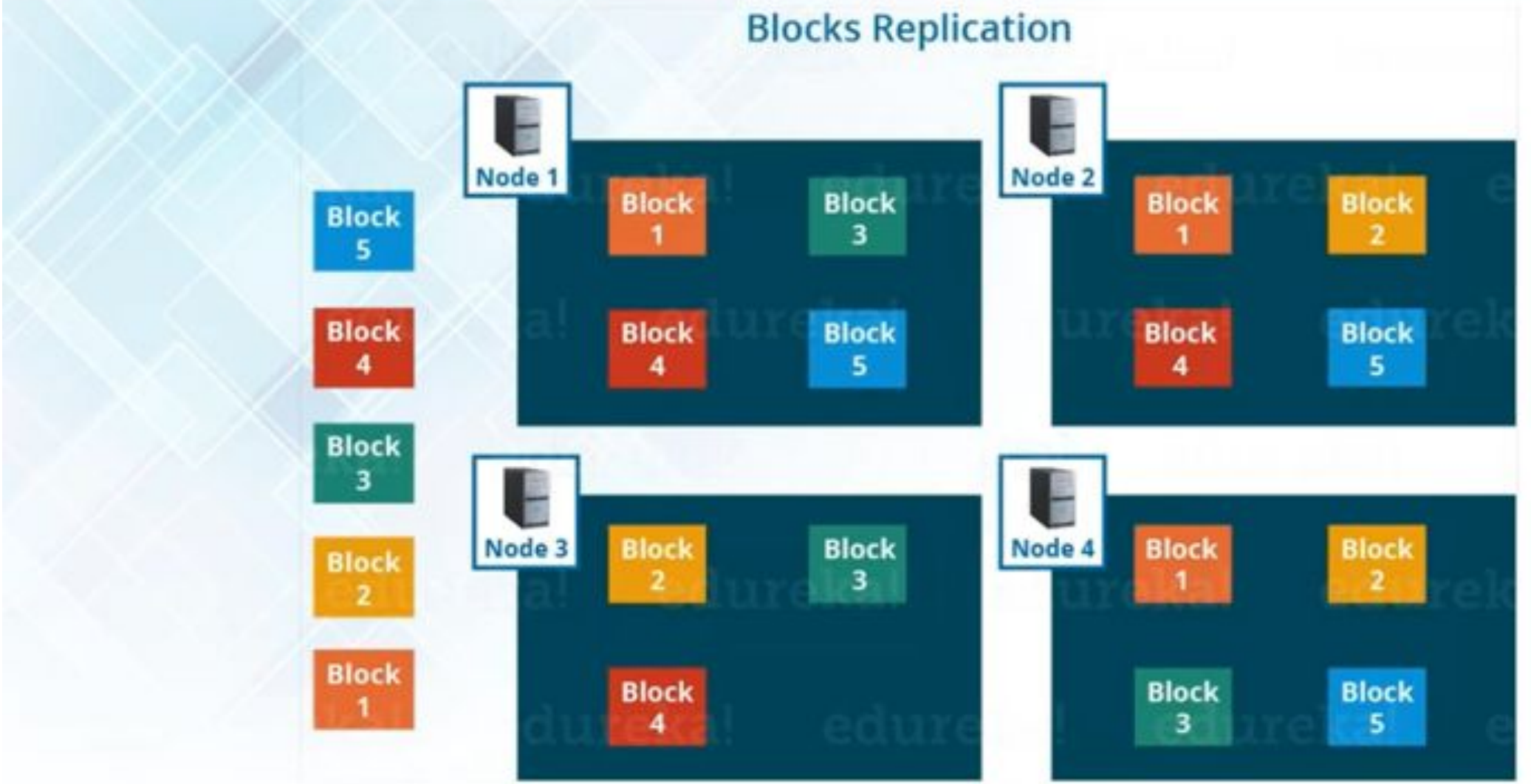
Performance Lost in Failures



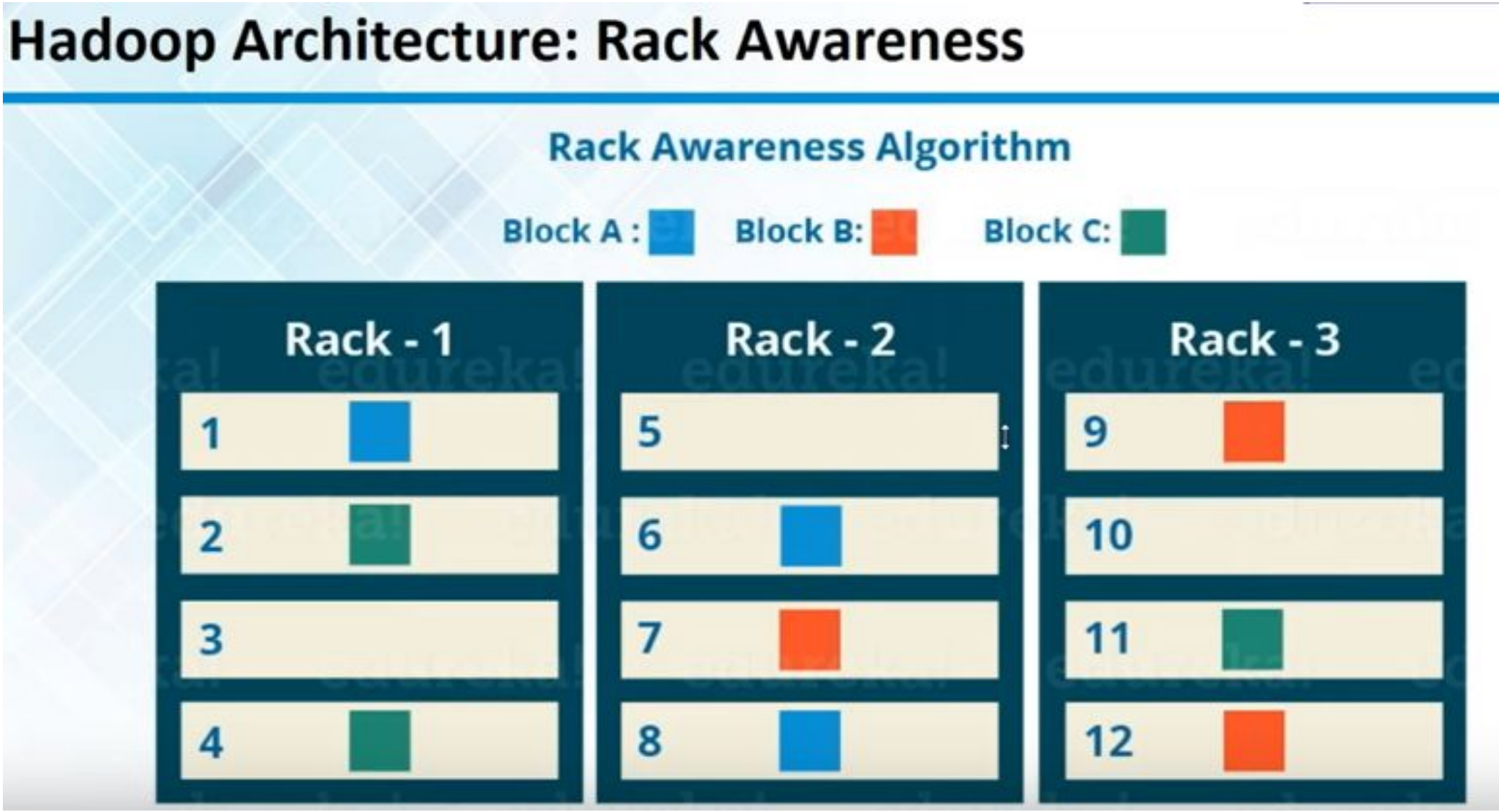

Replication
Cluster
Block 2
Computer 1
Block 1
Block 4
Computer 2
Block 3
Block 4
Computer 3
Block 2
Block 3
Computer 4
Block 1
Block 2
Computer 5
Block 3
Block 1
Computer 6
Block 4
Hadoop replicates each block across a number of machines
Block 1
Block 2
Block 3
Block 4
File 1


Number of Copies
Cluster
Block 2
Computer 1
Block 1
Block 4
Computer 2
Block 3
Block 4
Computer 3
Block 2
Block 3
Computer 4
Block 1
Block 2
Computer 5
Block 3
Block 1
Computer 6
Block 4
Default copies: 3
3
Block 1
Block 2
Block 3
File 1

Hadoop Architecture – Block Replication

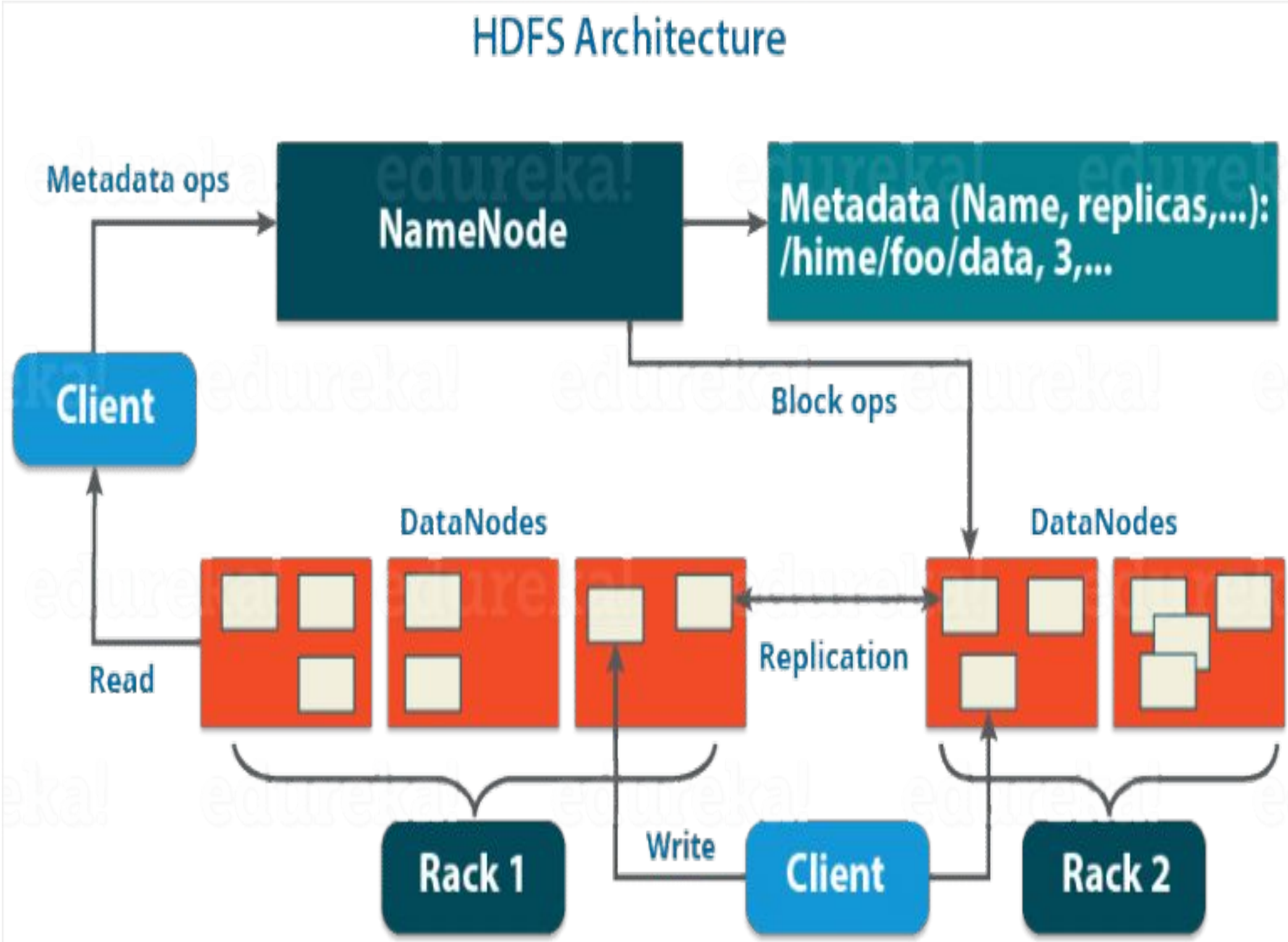


Hadoop Architecture: Rack Awareness



Suggested replication topology

- 1st replica placed on same node as client;
- 2nd replica placed on different rack from 1st rack; and
- 3rd replica placed on same rack as 2nd rack, but on a different node.


HDFS Architecture
Metadata ops
NameNode
Metadata (Name, replicas,...):
/hime/foo/data, 3,...
Client
Block ops
DataNodes
DataNodes
Read
Replication
Write
Rack 1
Client
Rack 2

HDFS Operation Principle

The HDFS components comprise different servers like NameNode, DataNode, and Secondary NameNode.

NameNode Server (single instance)

- Maintains the file system name space
- Manages the files and directories in the file system tree
- Stores information in the namespace image and the edit log
- NameNode knows the data nodes on which all the blocks for a given file exist
- NameNode is a critical one point failure node

DataNode Server (multiple instances)

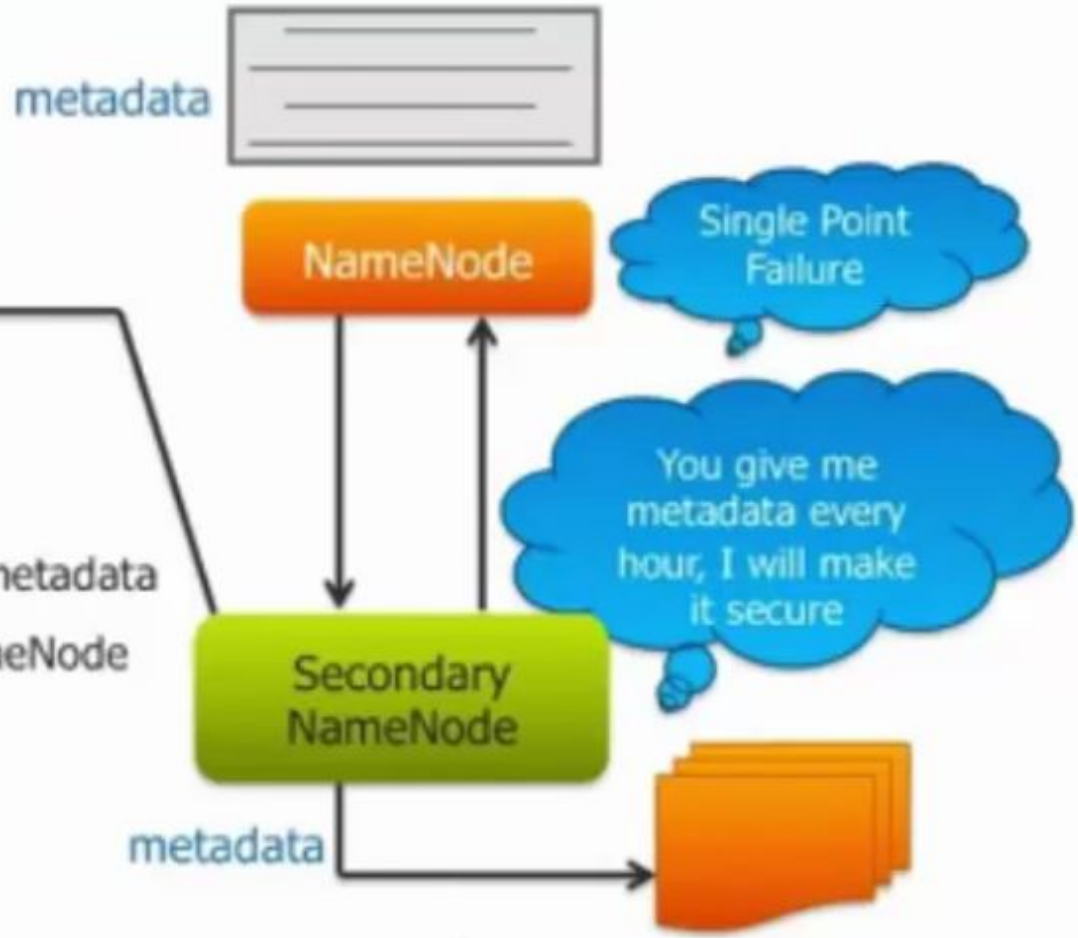
- Associated with data storage places in the file system
- Reports to NameNode periodically with lists of blocks they store
- Stores and retrieves blocks when referred by clients or NameNode
- Servers read, write requests, performs block creation, deletion, and replication upon instruction from NameNode

Secondary NameNode Server (single instance)

- Not exactly a hot backup of the actual NameNode server
- Used for recovery of NameNode in case of NameNode failure
- Keeps namespace image through edit log periodically
- Namespace image lags behind, so total recovery is impossible

Secondary Name Node

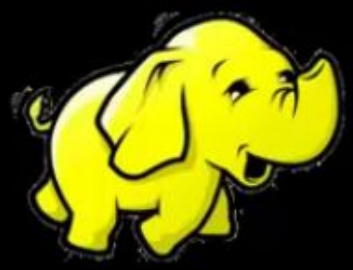
- ✓ Secondary NameNode:
  - ✓ Not a hot standby for the NameNode
  - ✓ Connects to NameNode every hour*
  - ✓ Housekeeping, backup of NemeNode metadata
  - ✓ Saved metadata can build a failed NameNode



Hadoop processing Unit
Map Reduce

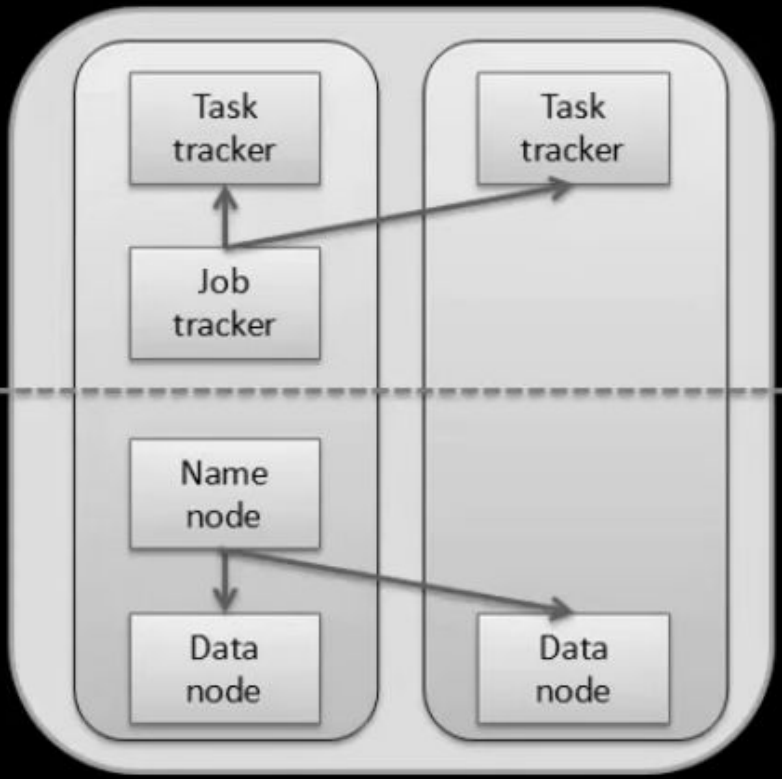
How MapReduce Works

Hadoop Processing Architecture



Map Reduce Layer

HDFS Layer

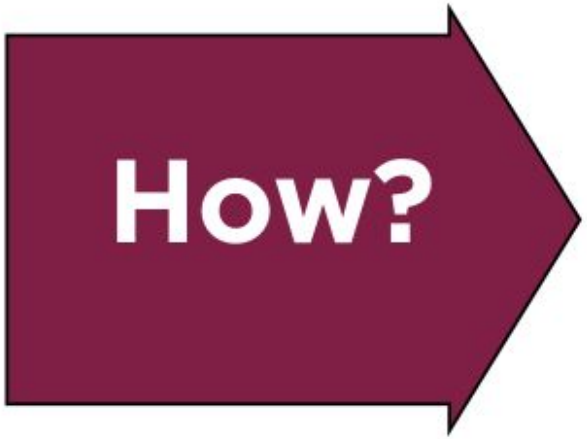


- The client submits the job to Job Tracker
- Job Tracker asks Name node the location of data
- As per the reply from name node, the Job Tracker ask respective task trackers to execute the task on their data
- All the results are stored on some Data Node and the Name Node is informed about the same
- The task Trackers inform the job completion and progress to Job Tracker
- The Job Tracker inform the completion to client
- Client contacts the Name Node and retrieve the results

Counting Word Frequencies

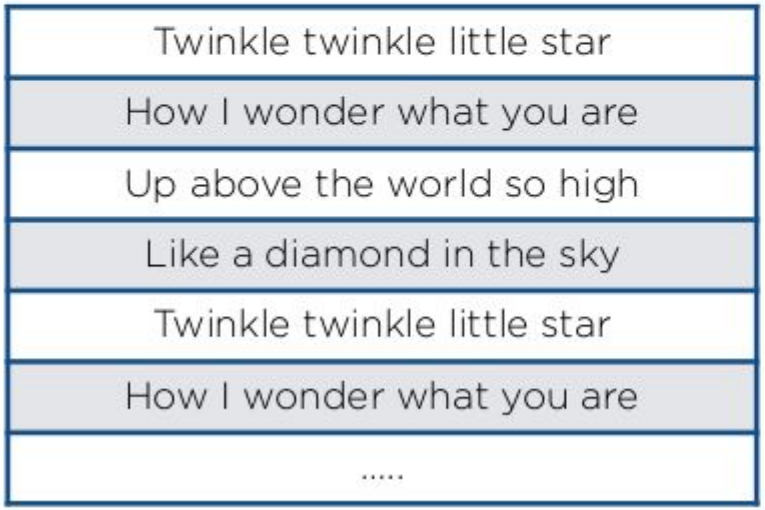
Consider a large text file

Twinkle twinkle little star
How I wonder what you are
Up above the world so high
Like a diamond in the sky
Twinkle twinkle little star
How I wonder what you are
.....



Word	Frequency
above	14
are	20
how	21
star	22
twinkle	32
...	..

MapReduce Flow

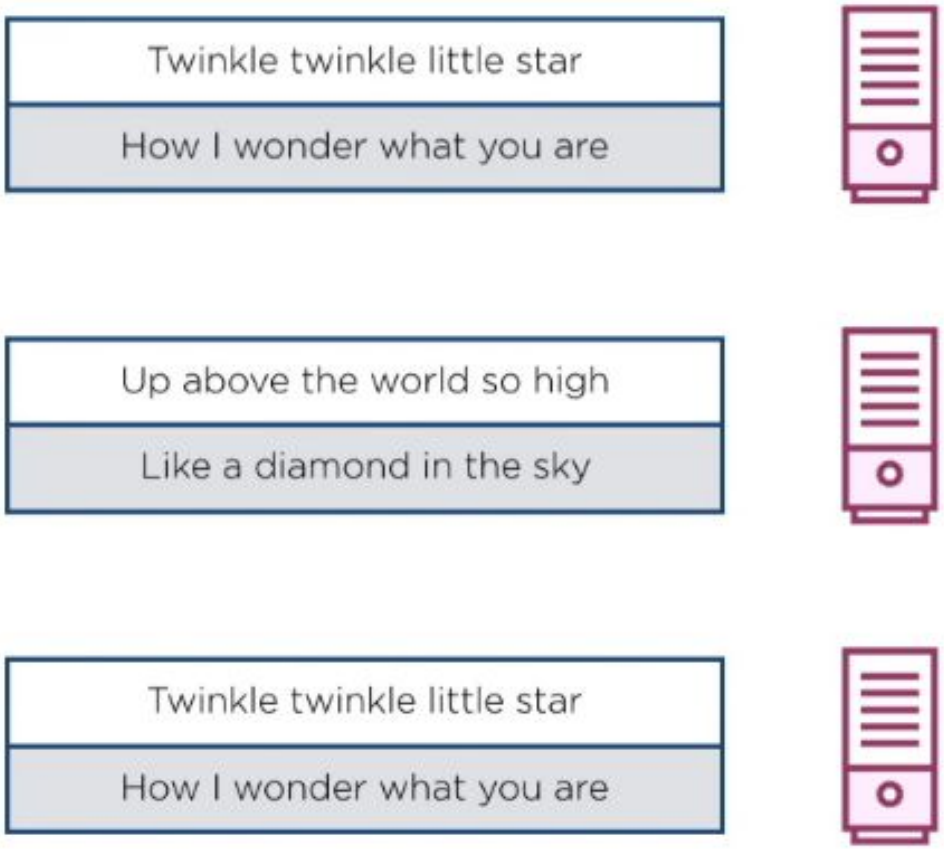


The raw data is really large (potentially in PetaBytes)

It's distributed across many machines in a cluster

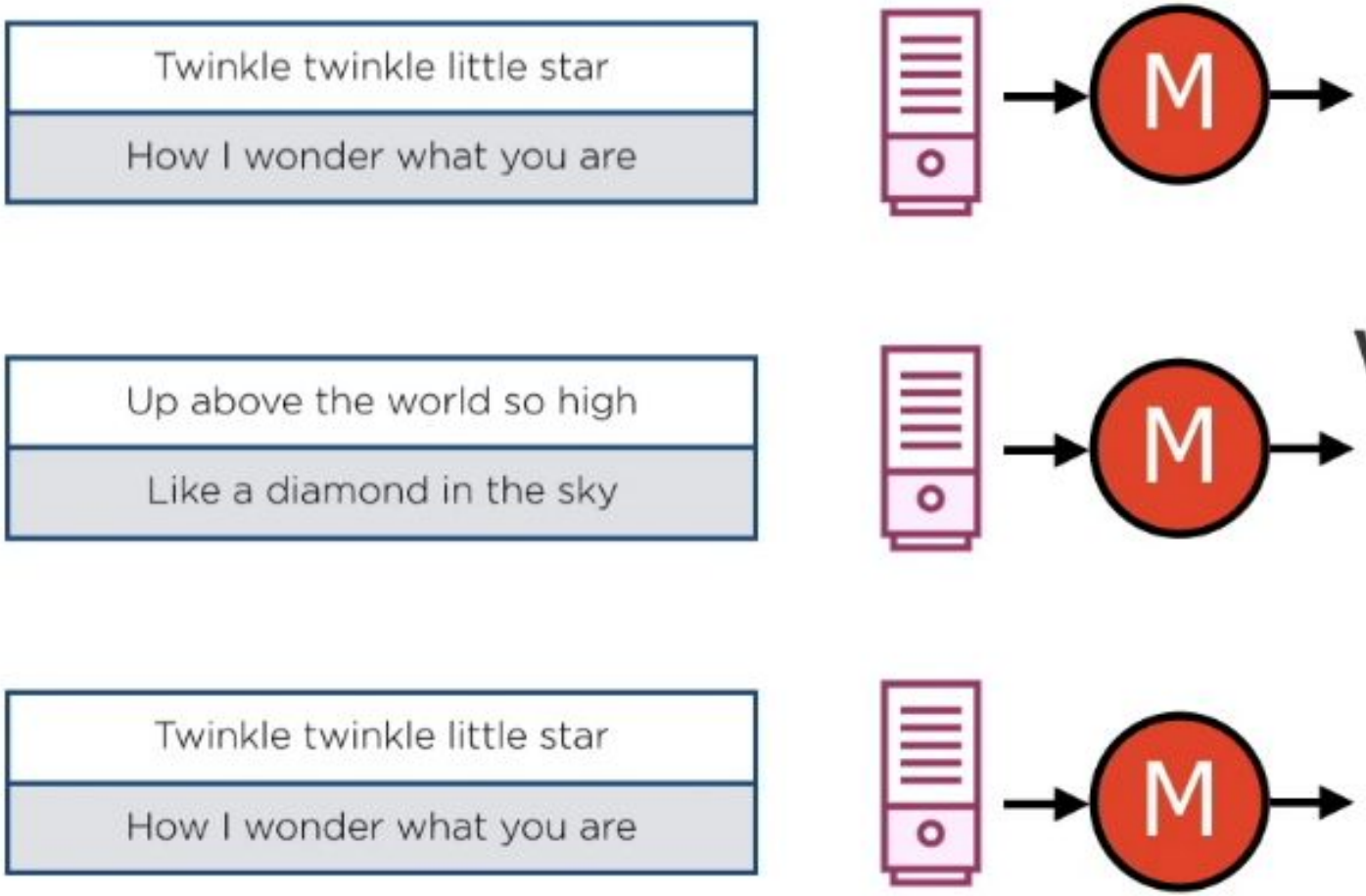
Each machine holds a partition of data

MapReduce Flow



Each partition is given to a different process i.e. to mappers

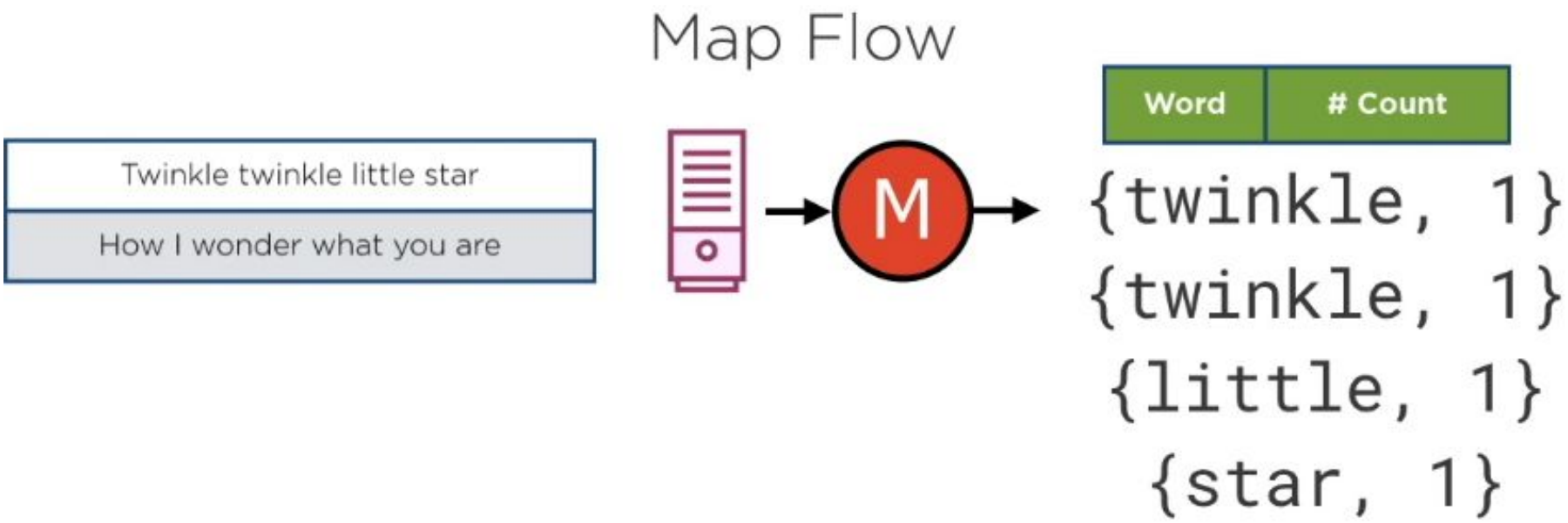
MapReduce Flow



Each mapper works in parallel

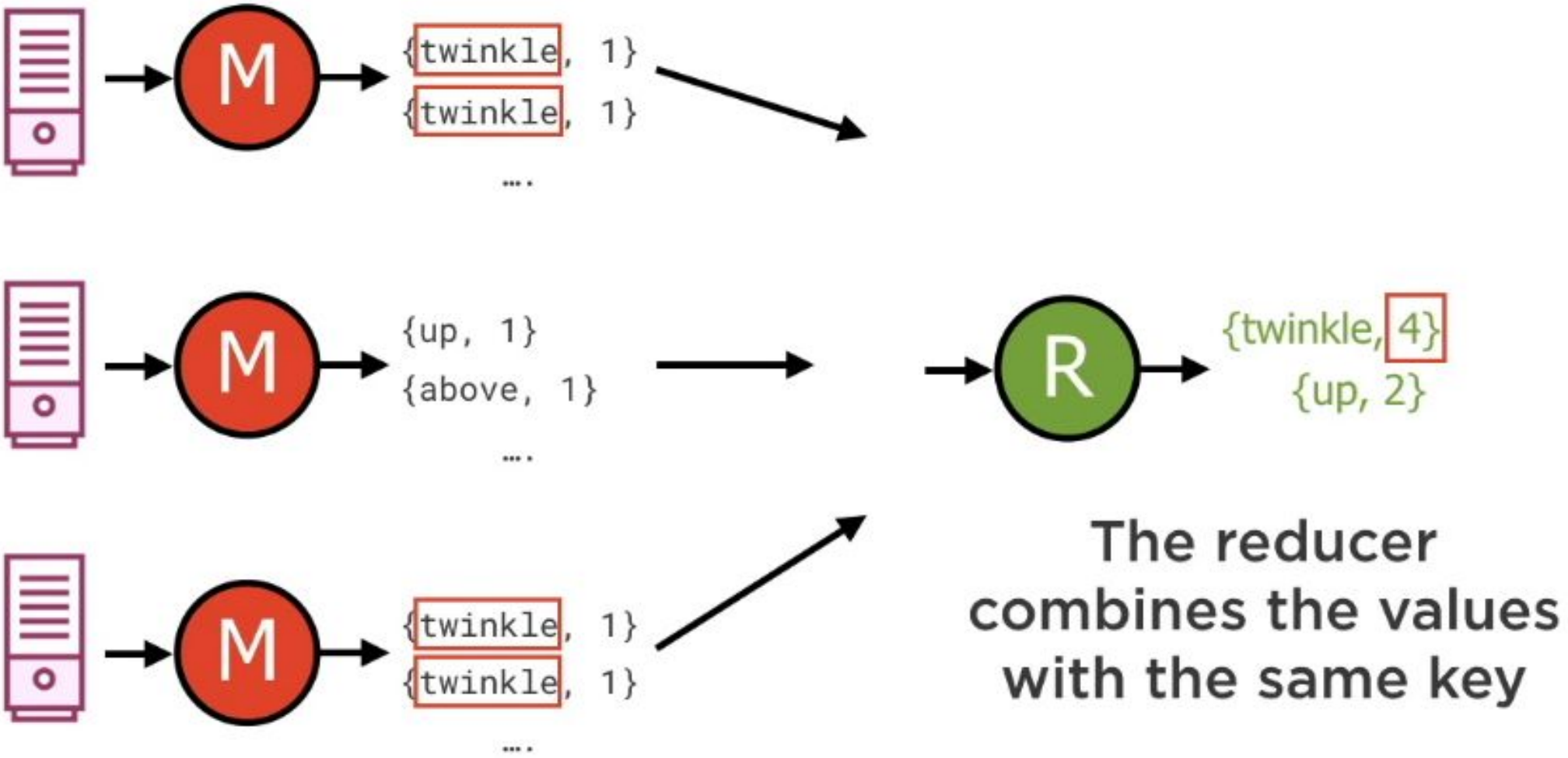
Map Flow

Within each mapper, the rows are processed serially



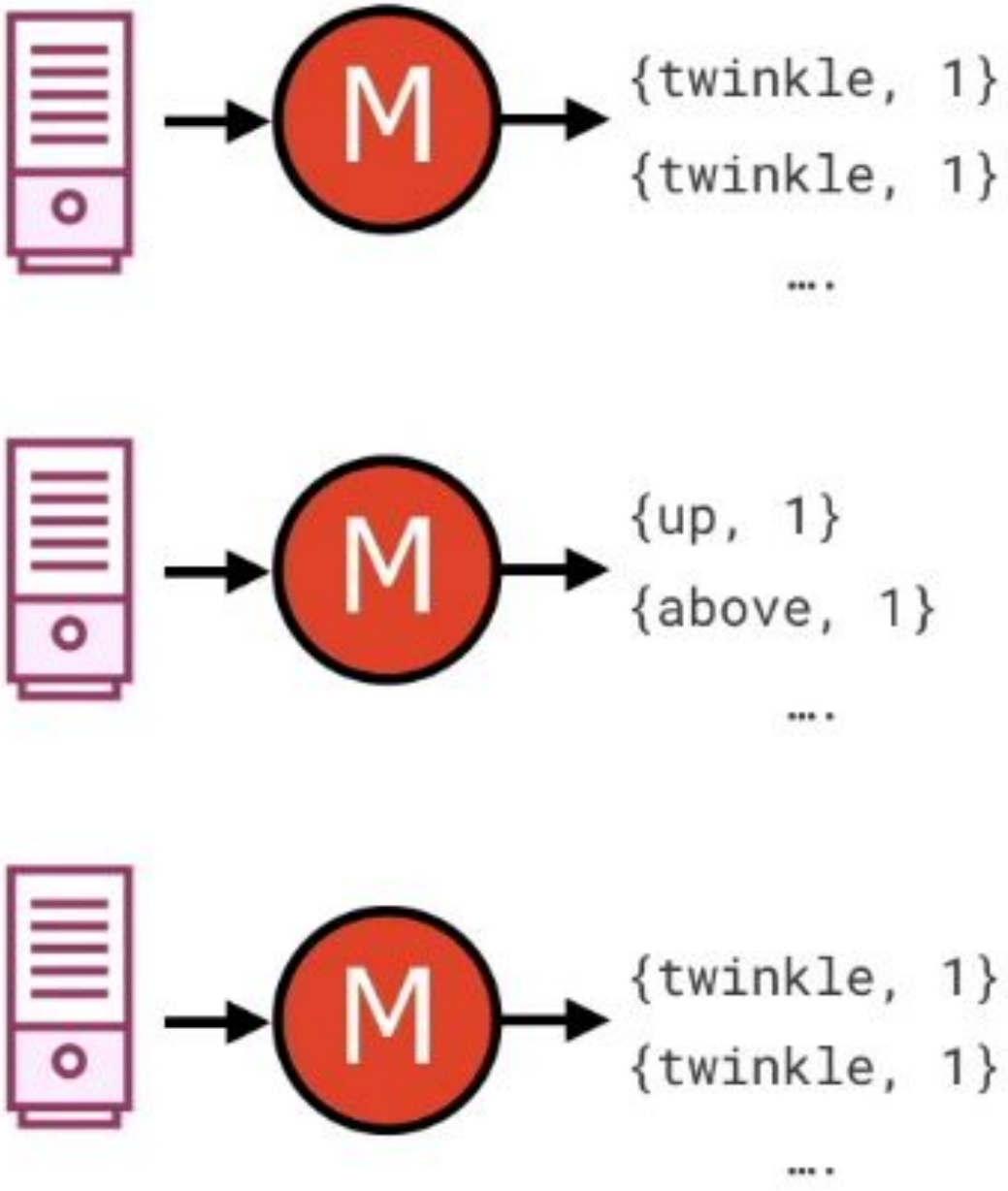
Each row emits {key, value} pairs

Reduce Flow



The reducer combines the values with the same key

Reduce Flow



Key Insight Behind MapReduce

Many data processing tasks can be expressed in this form

Counting Word Frequencies

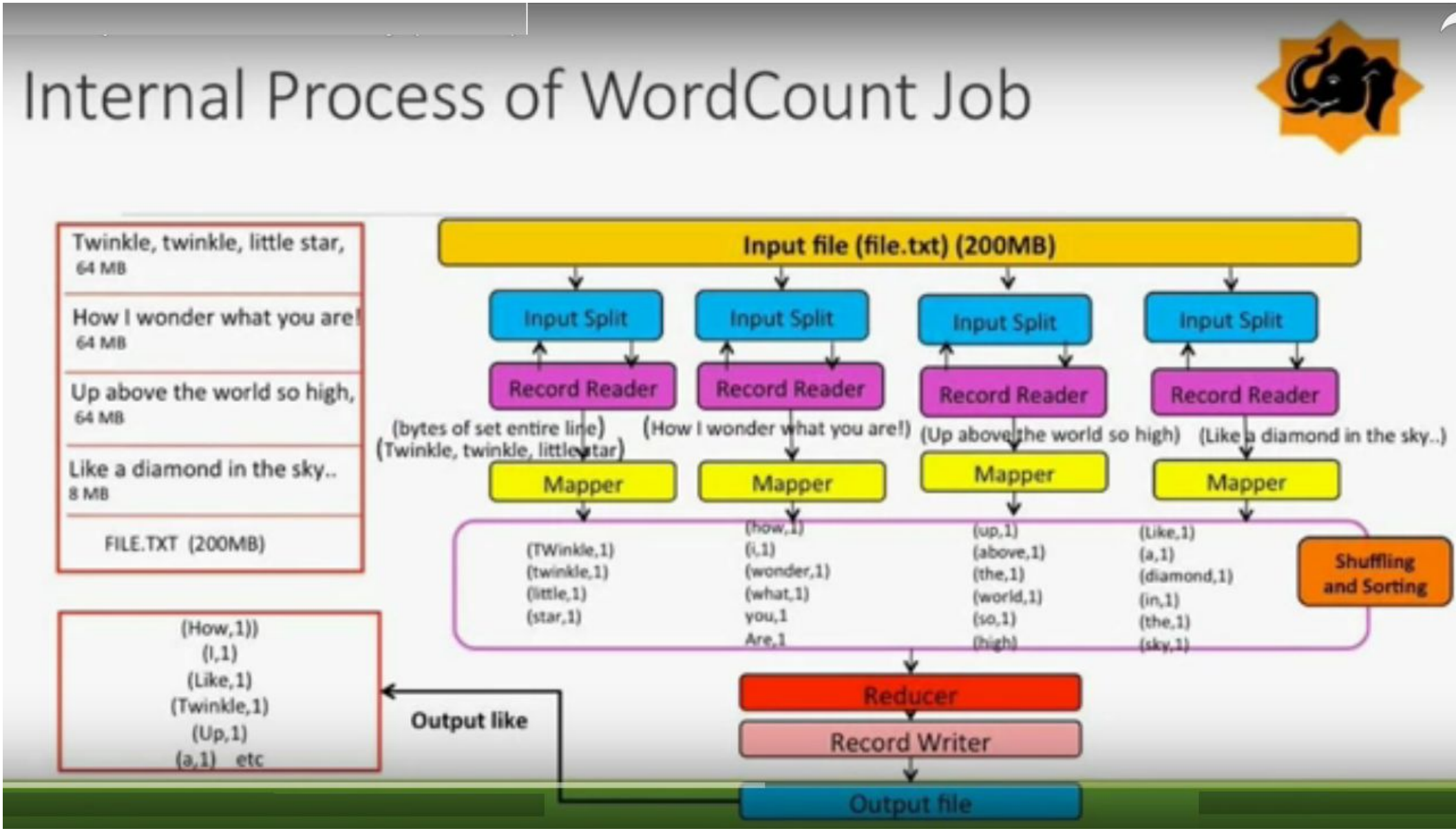
Twinkle twinkle little star
How I wonder what you are
Up above the world so high
Like a diamond in the sky

Word	Count
twinkle	2
little	1
...	...
...	...
...	...
...	...

For each word in each line

```
{twinkle, 1}
{twinkle, 1}
{little, 1}
{star, 1}
..
```

Internal Process of WordCount Job



Hadoop Framework Keys & Values

JAVA	HADOOP
int	IntWritable
long	LongWritable
float	FloatWritable
double	DoubleWritable
boolean	BooleanWritable
null	NullWritable
String	Text