HADOOP ARCHITECTURE

Agenda for today's Session

1. Hadoop Components
2. DFS – Distributed File System
3. Hadoop Services
4. Blocks in Hadoop
5. Block Replication
6. Rack Awareness
7. HDFS Architecture
8. HDFS Read/Write Mechanism

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

Hadoop has a Distributed File System.

But Why?

DFS – Distributed File System

Read 1TB Data

1 Machine
4 I/O Channels
Each channel - 100MB/s

43 Minutes

DFS – Distributed File System

Read 1TB Data

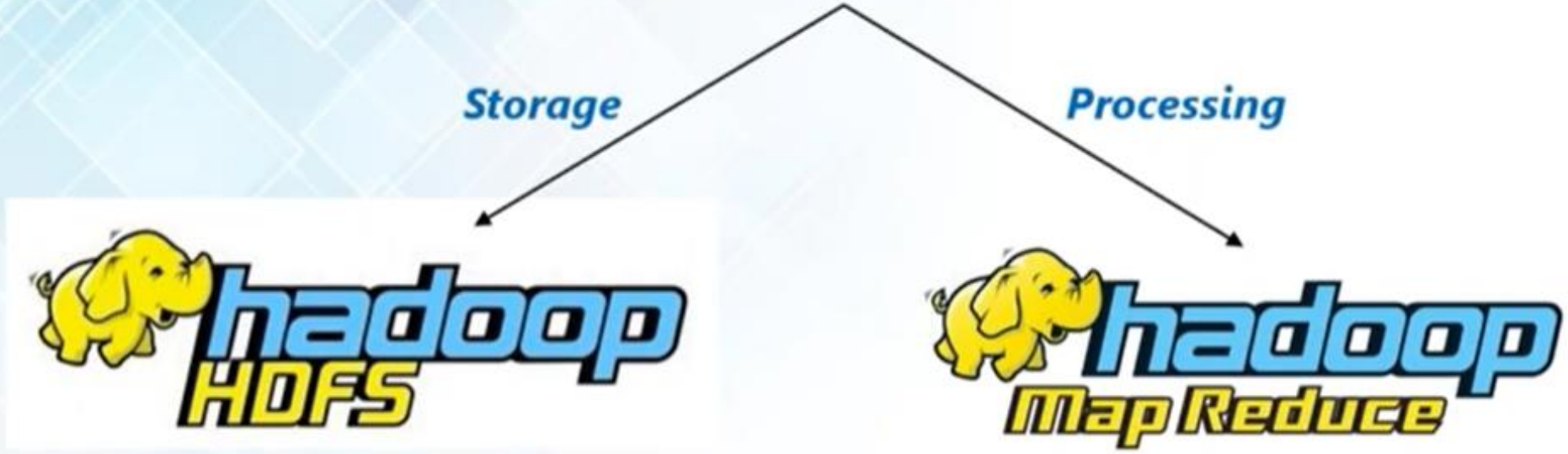
1 Machine
4 I/O Channels
Each channel - 100MB/s

43 Minutes

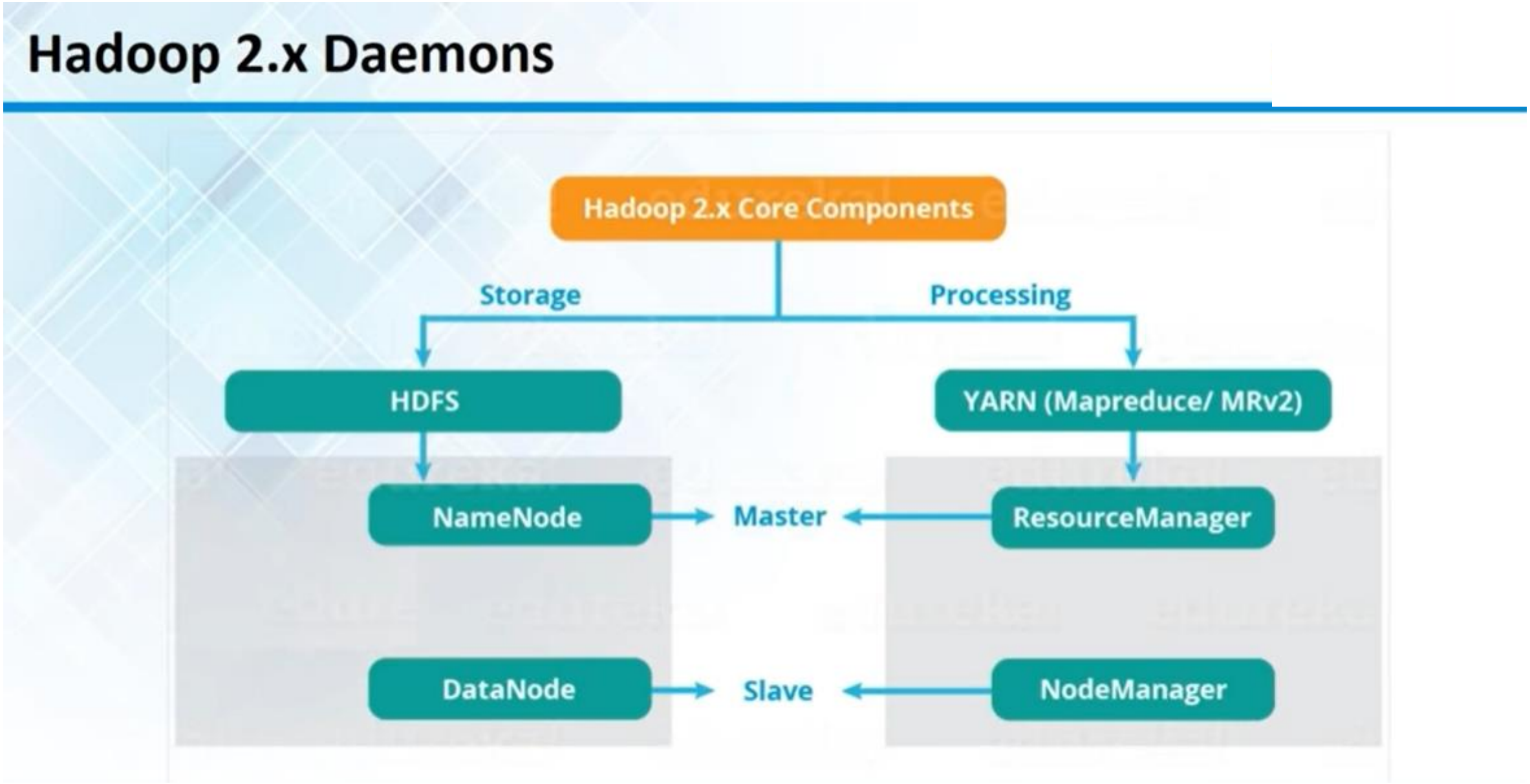
10 Machines
4 I/O Channels
Each channel - 100MB/s

Hadoop Components

2 Main Hadoop Components



Hadoop 2.x Daemons



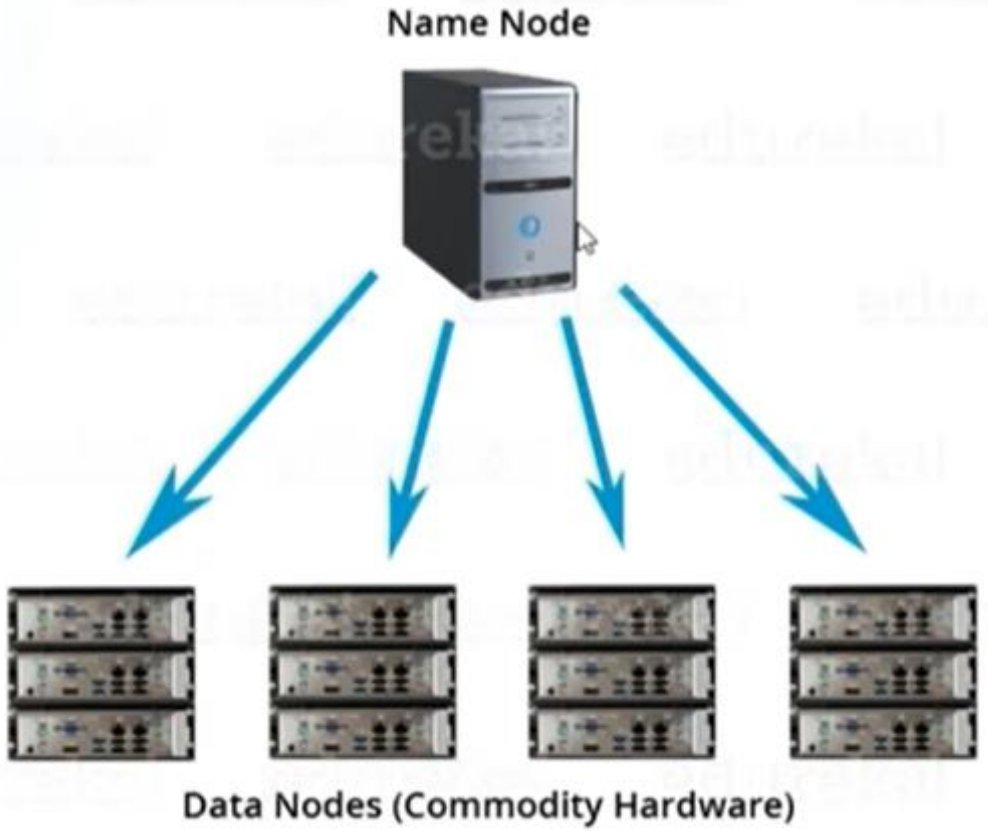
NameNode and DataNode

NameNode

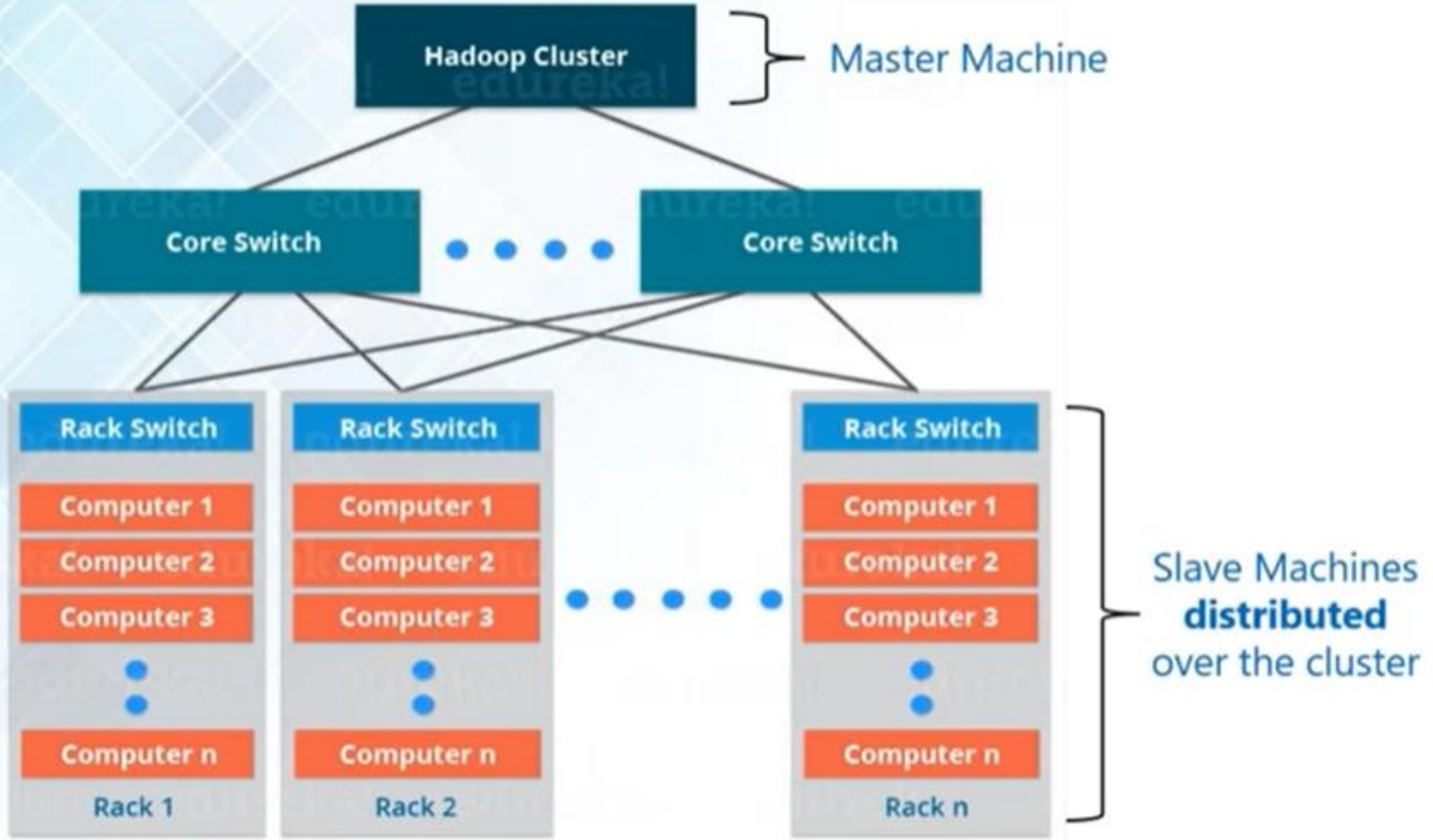
- Master daemon
- Maintains and Manages DataNodes
- Records metadata e.g. location of blocks stored, the size of the files, permissions, hierarchy, etc.
- Receives heartbeat and block report from all the DataNodes

DataNode

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



Hadoop Cluster Architecture – Master Slave Topology



Let us talk about,

how data is stored in HDFS?

HDFS Blocks

- 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

HDFS Blocks

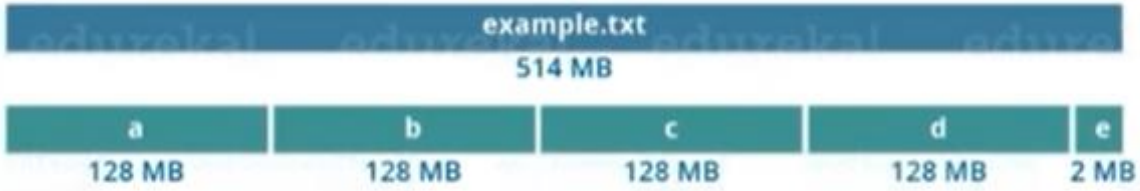
- Each file is stored on HDFS as blocks
- The default size of each block is **128 MB** in Apache Hadoop 2.x (64 MB in Apache Hadoop 1.x)
- Let us say, I have a file example.txt of size 248 MB. Below is the representation of how it will be stored on HDFS

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

HDFS Blocks

- 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

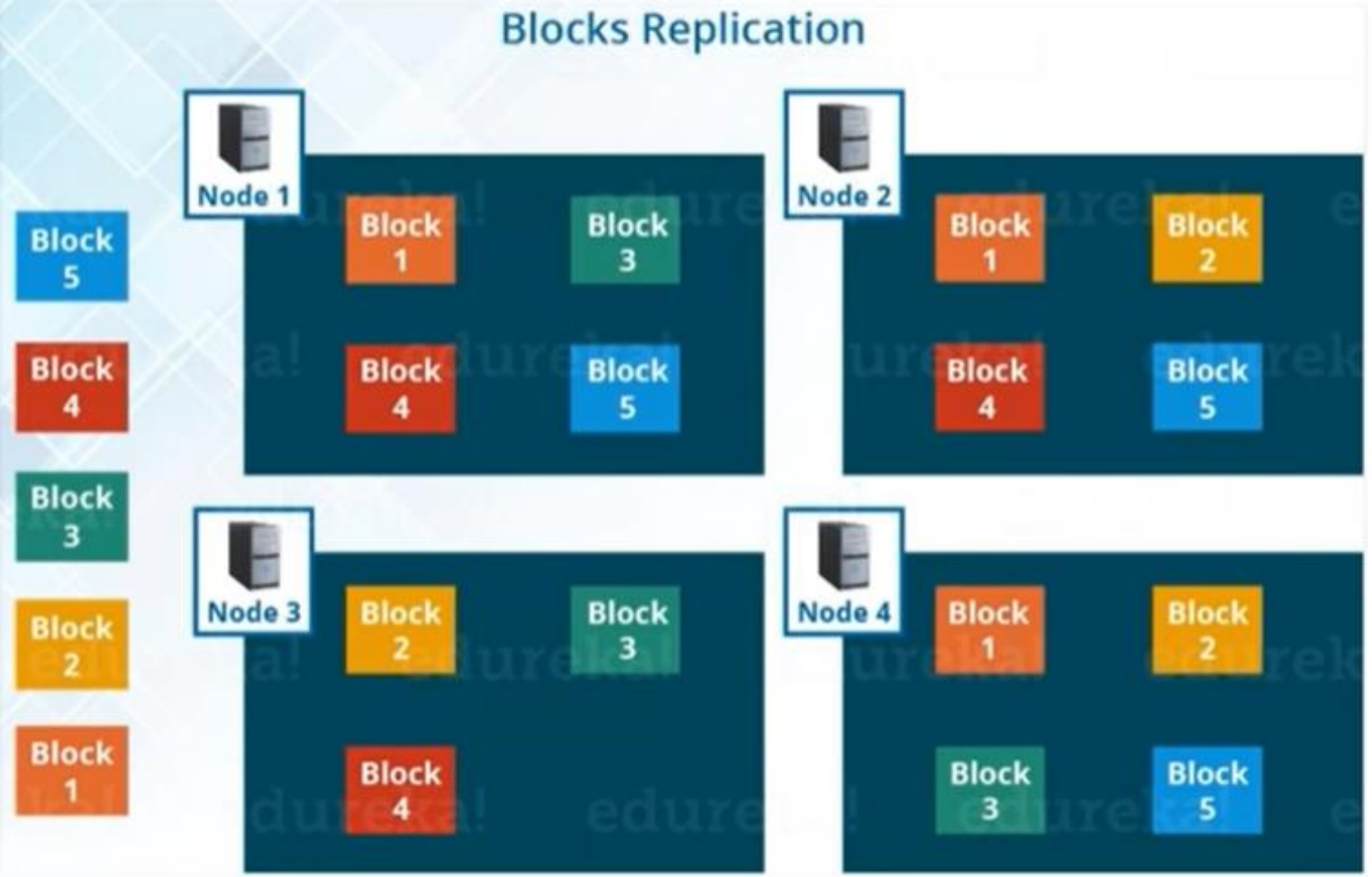
How many blocks will be created if a file of size 514 MB is copied to HDFS ?



Is it safe to have just 1 copy of each block?

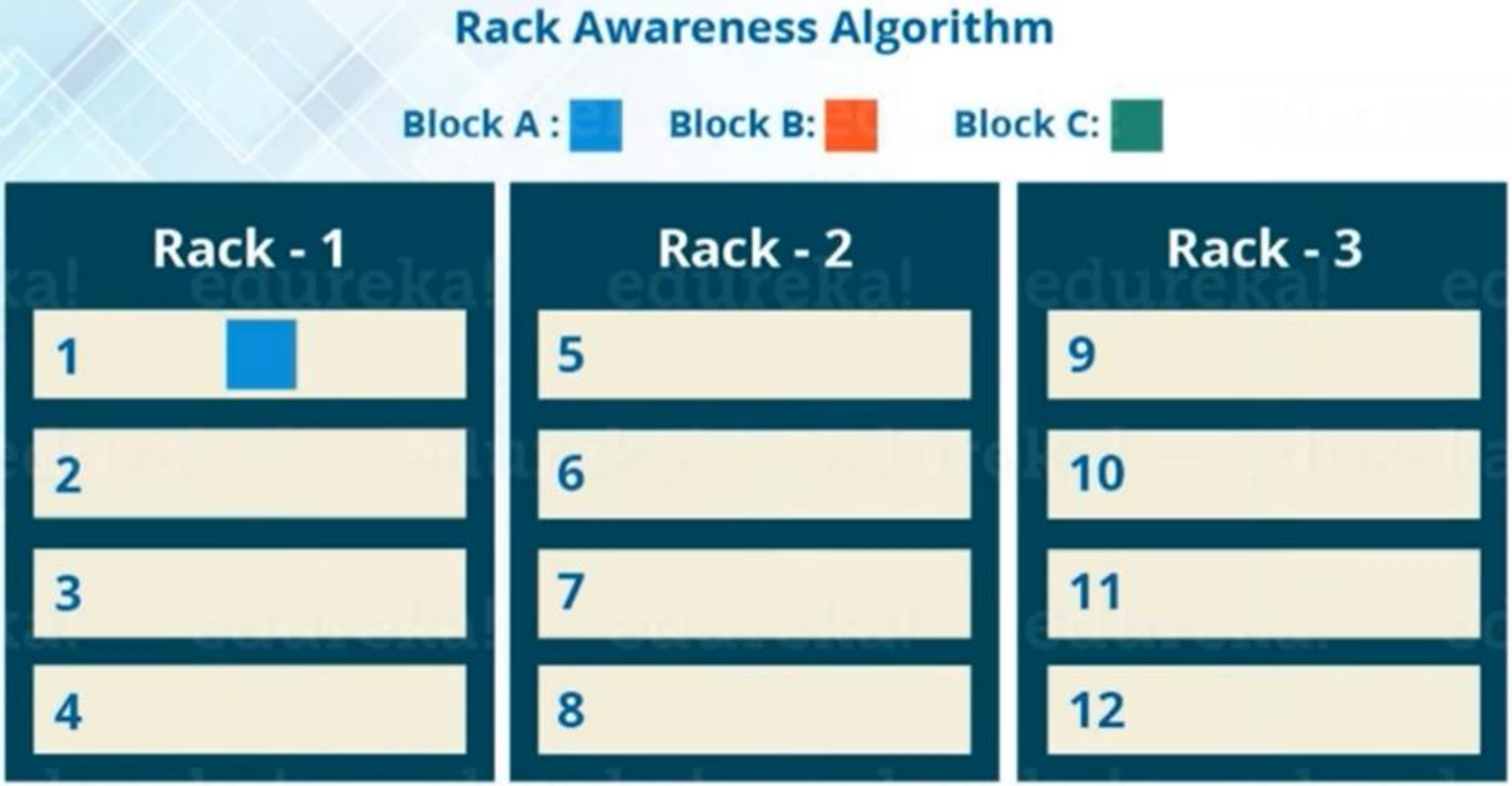
What do you think?

Hadoop Architecture – Block Replication

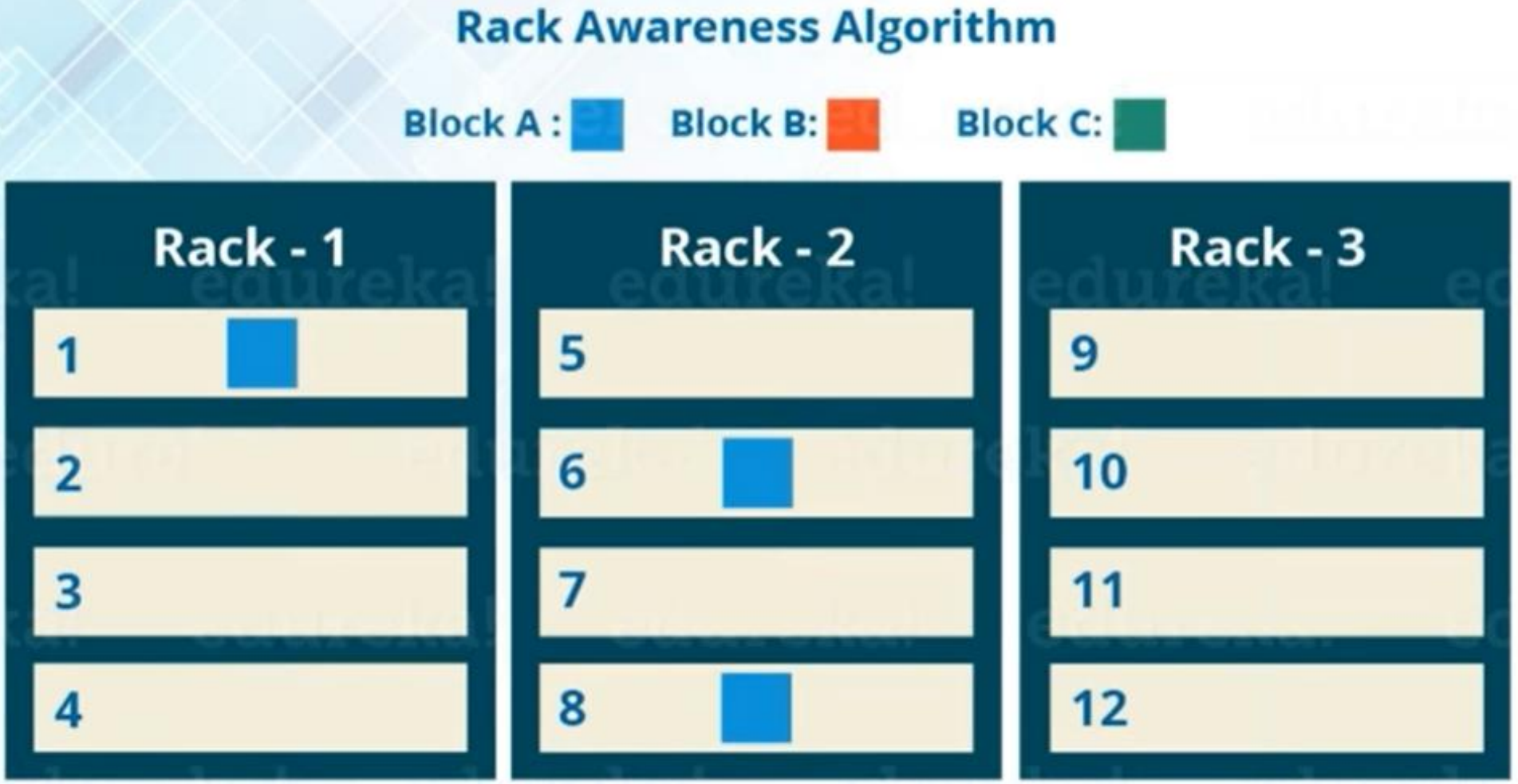


Hadoop Architecture: Rack Awareness

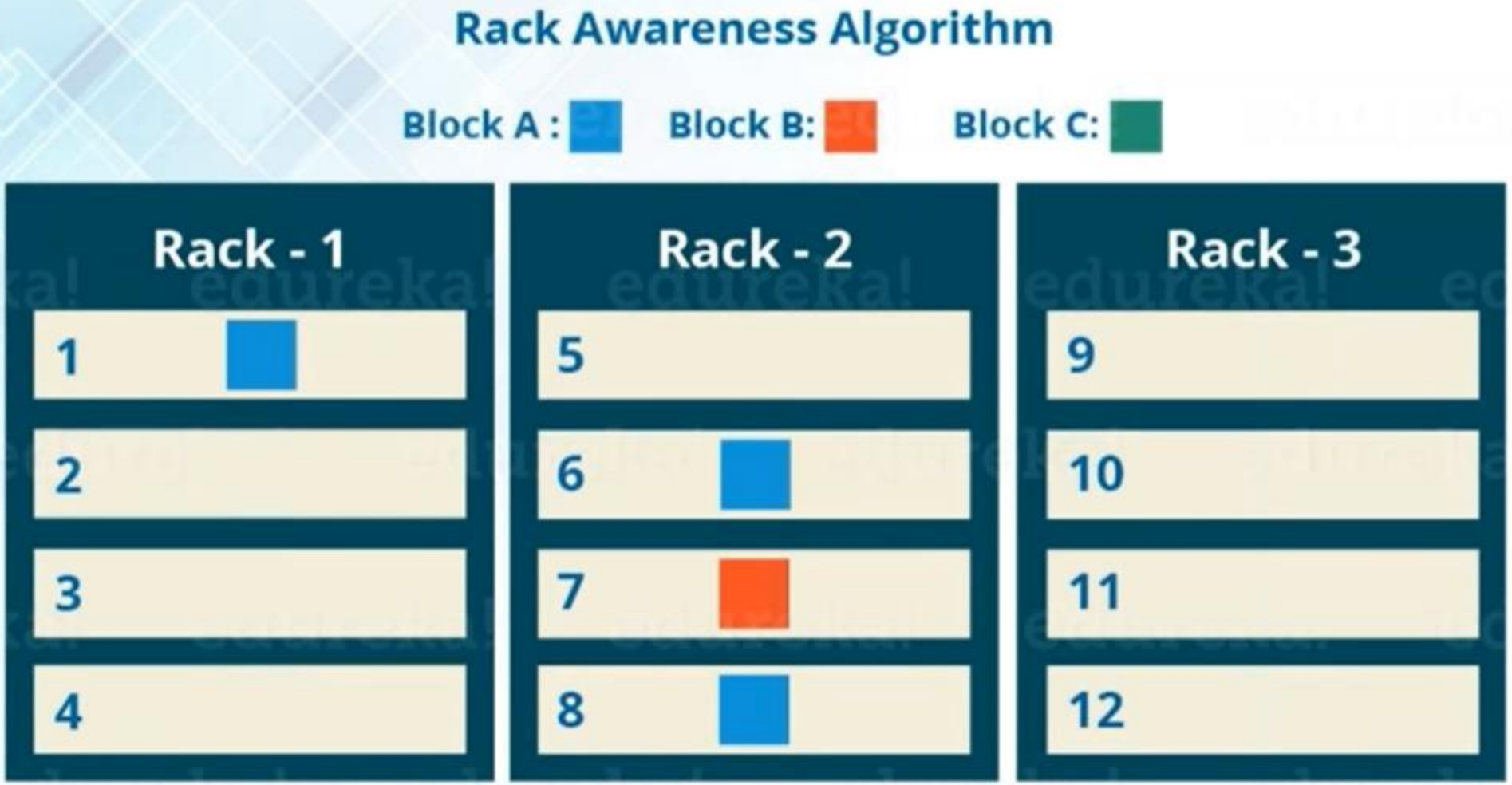
Hadoop Architecture: Rack Awareness



Hadoop Architecture: Rack Awareness



Hadoop Architecture: Rack Awareness

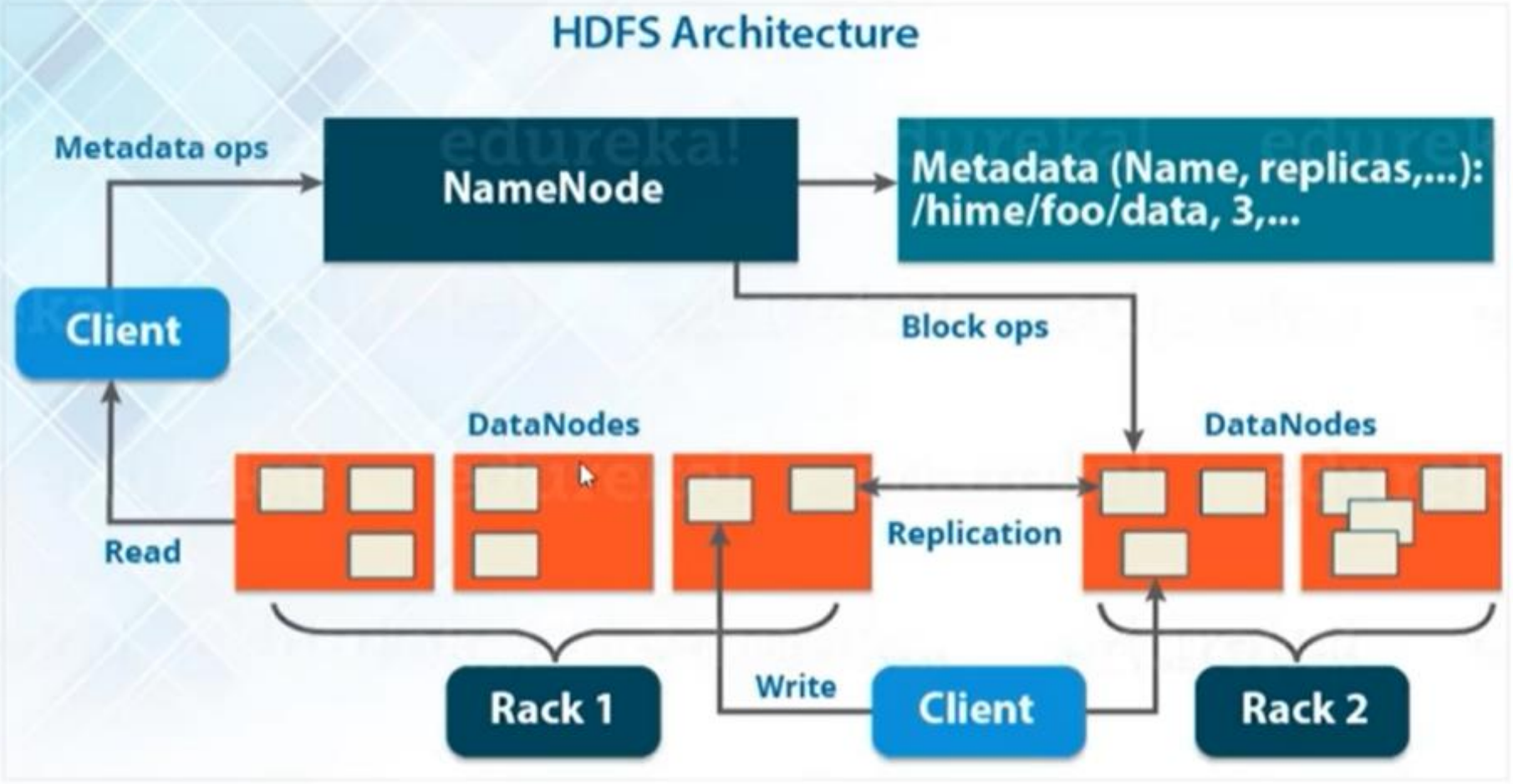


Hadoop Architecture: Rack Awareness

Hadoop Architecture: Rack Awareness

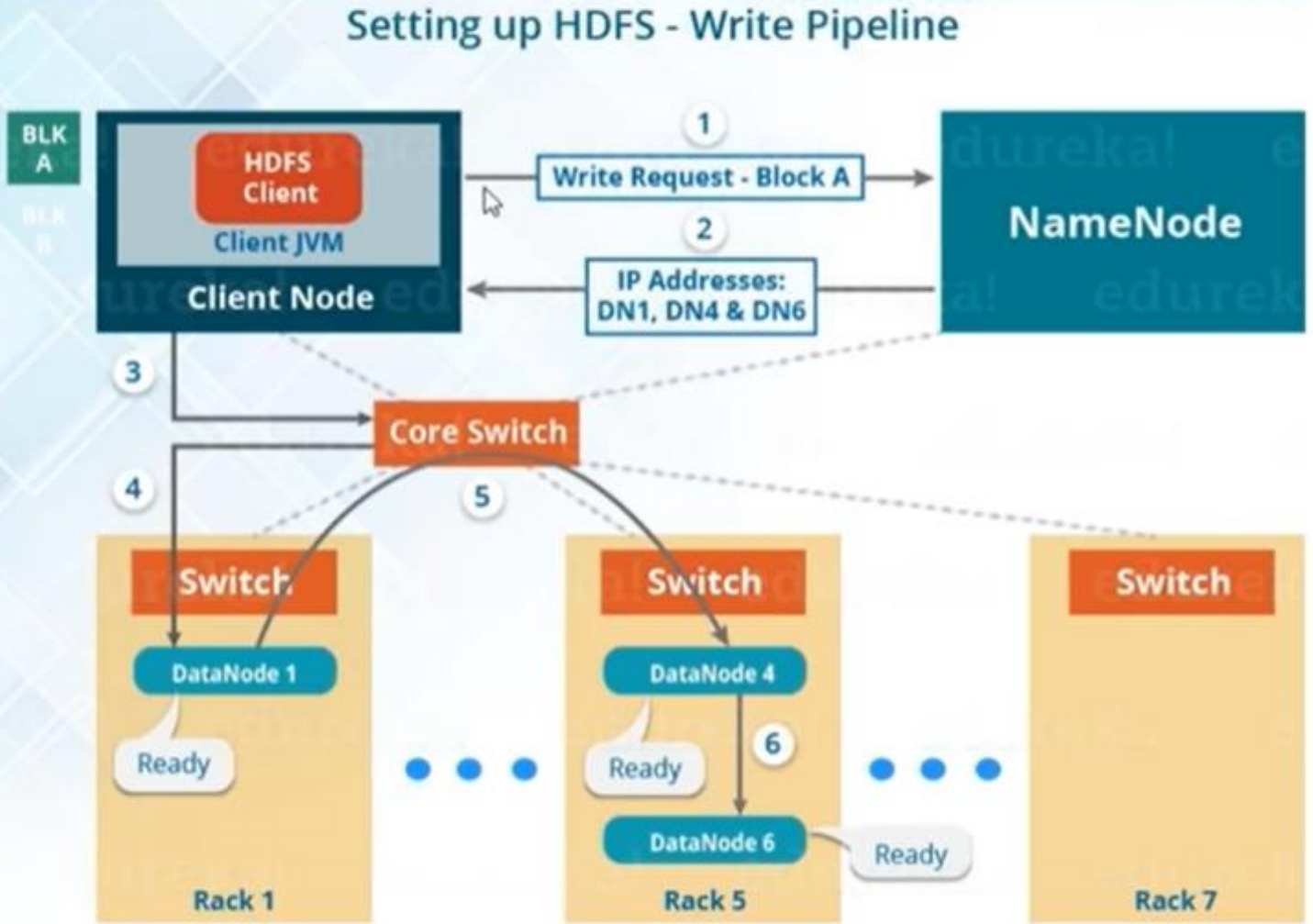
Architecture of HDFS

HDFS Architecture

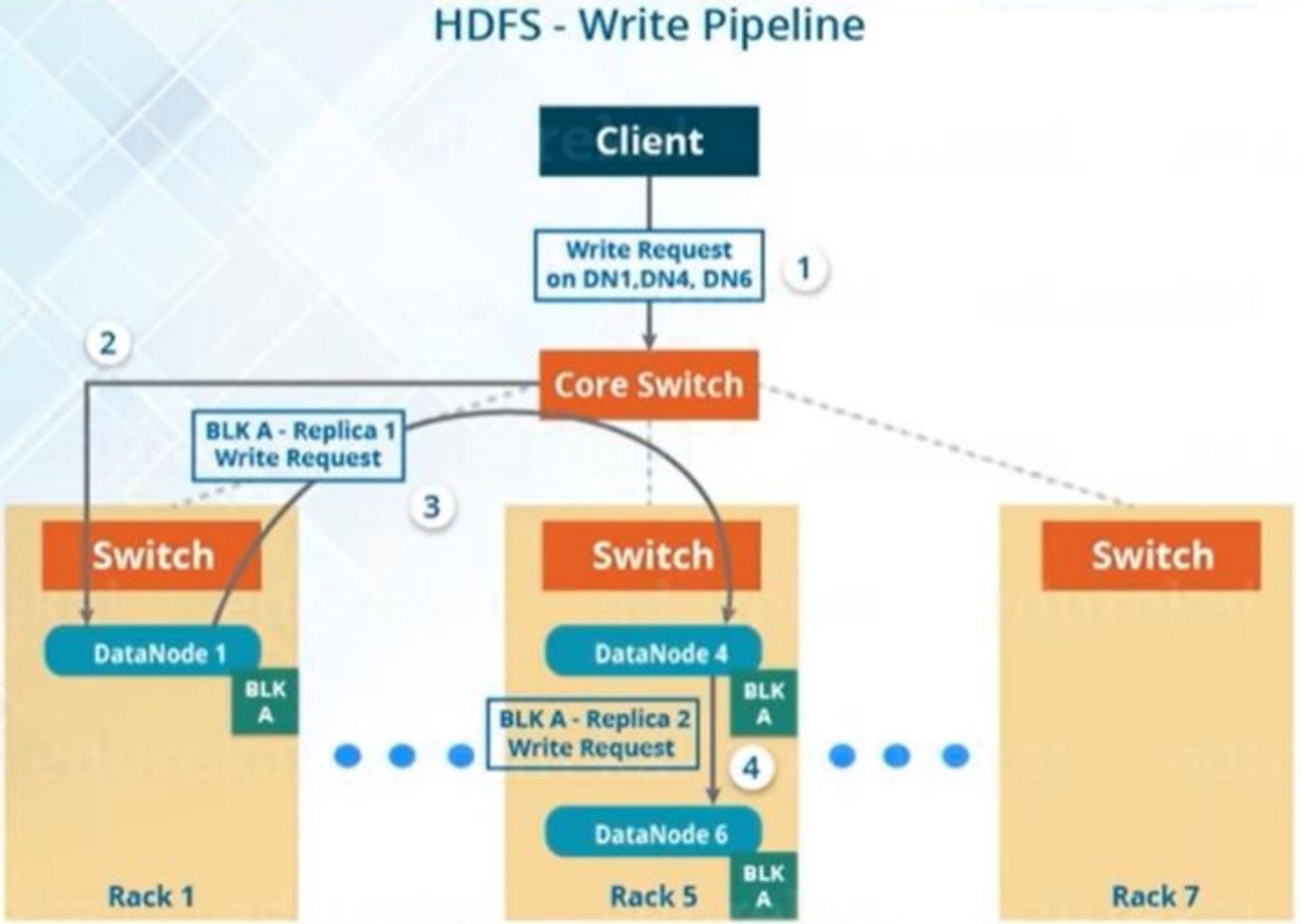


HDFS Read/Write Mechanism

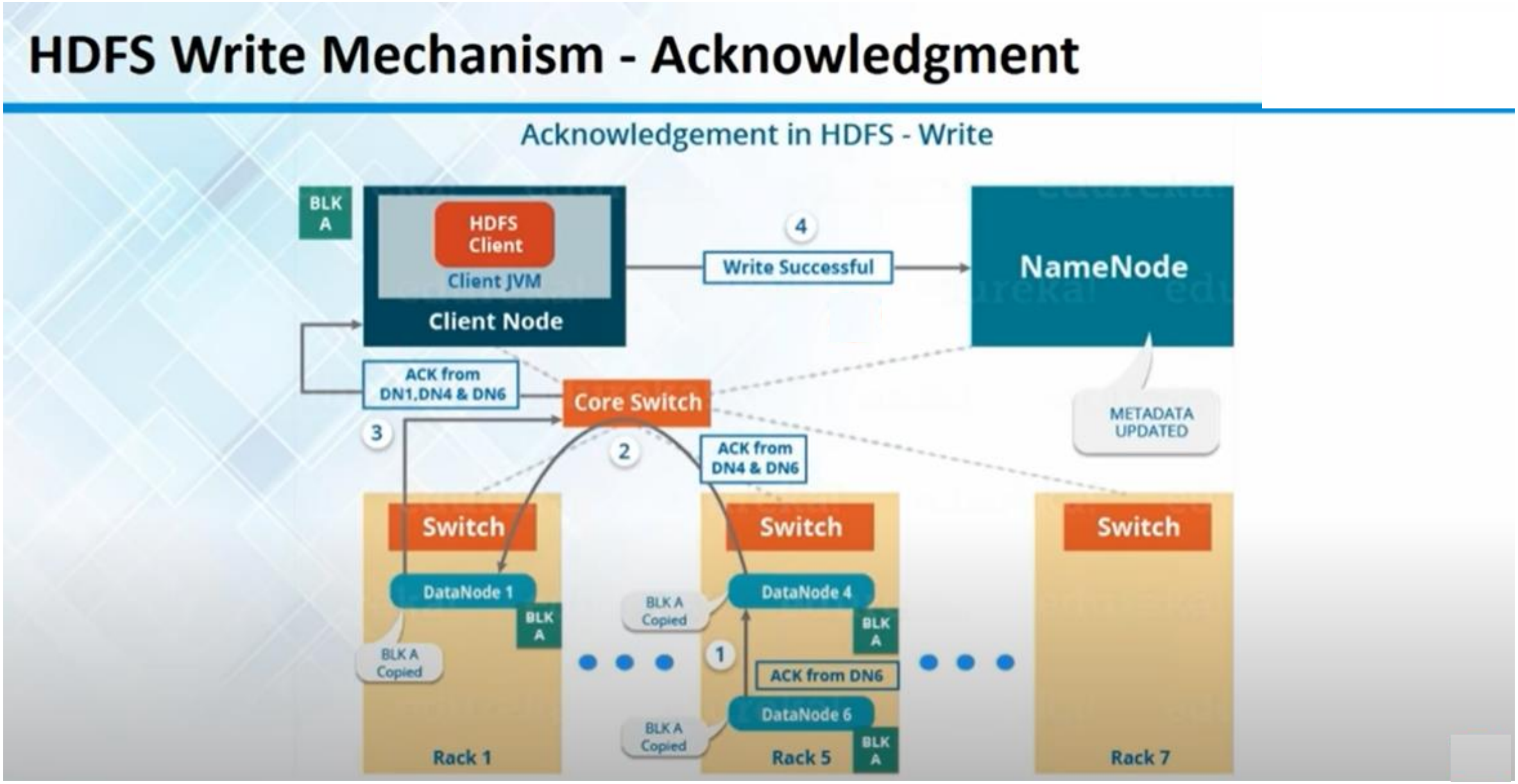
HDFS Write Mechanism – Pipeline Setup



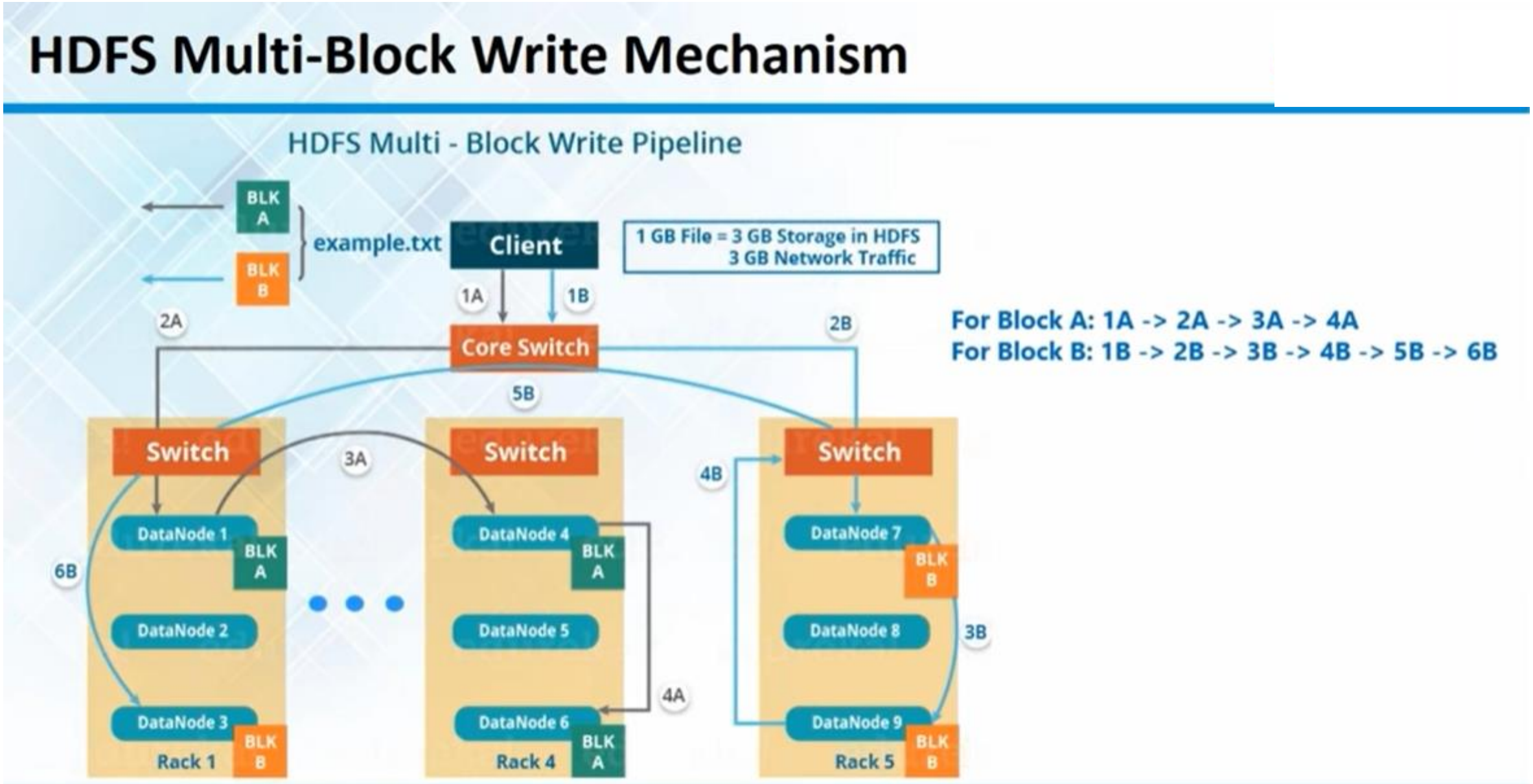
HDFS Write Mechanism – Writing a Block



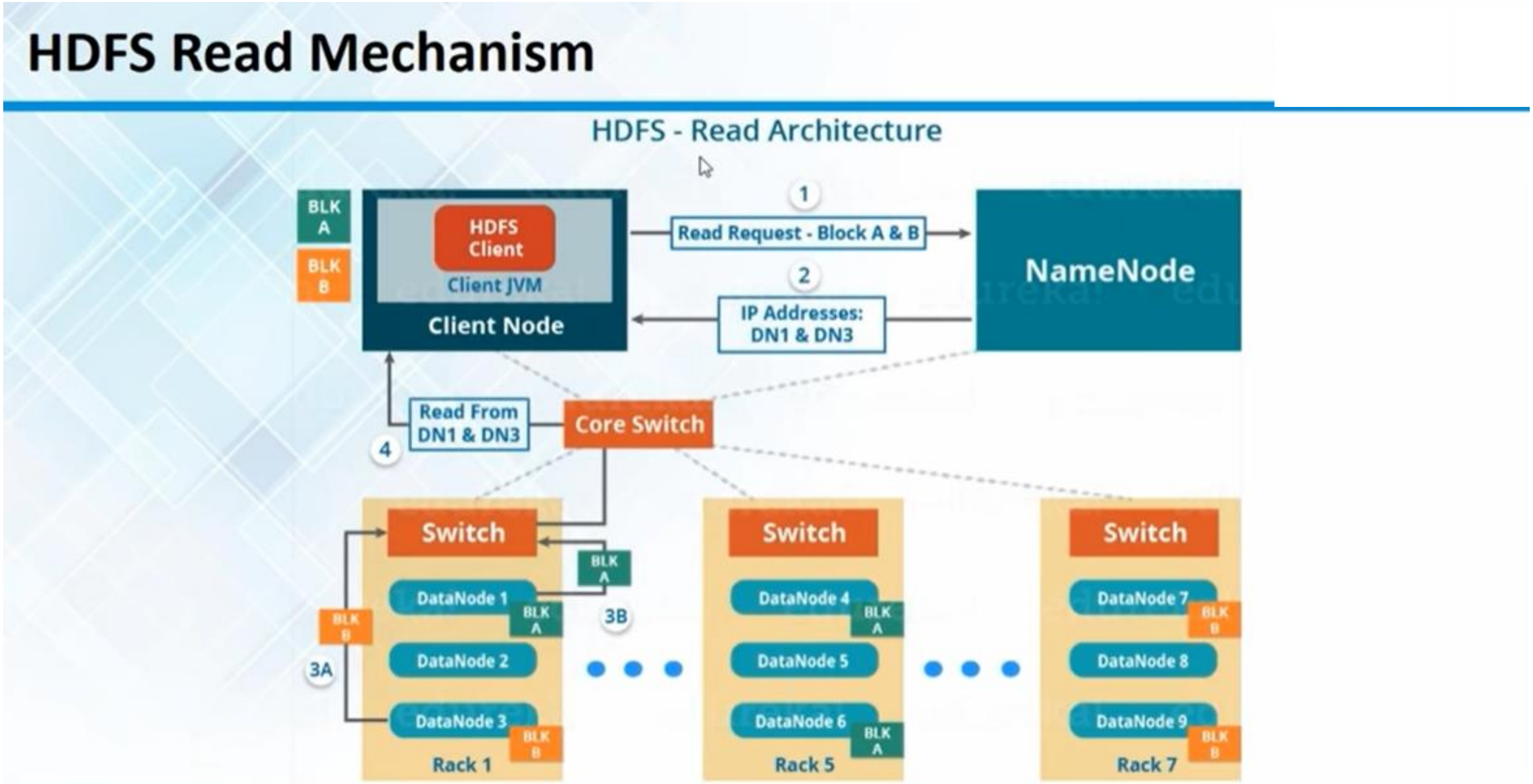
HDFS Write Mechanism - Acknowledgment



HDFS Multi-Block Write Mechanism



HDFS Read Mechanism



Let us see few Hadoop/HDFS Commands

HDFS commands

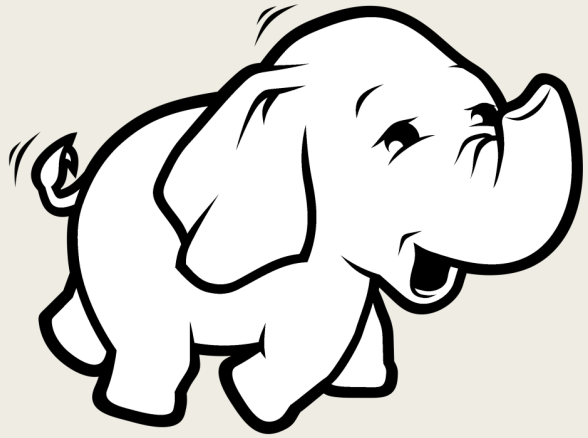
Administrator: Command Prompt

```
C:\WINDOWS\system32>hdfs fsck /
Connecting to namenode via http://localhost:9870/fsck?ugi=MarizzaMil&path=%2F
FSCK started by MarizzaMil (auth:SIMPLE) from /127.0.0.1 for path / at Thu Aug 26 17:56:47 IDT 2021


Status: HEALTHY
 Number of data-nodes:  1
 Number of racks:               1
 Total dirs:                    3
 Total symlinks:                0

Replicated Blocks:
 Total size:    69 B
 Total files:   6
 Total blocks (validated):      2 (avg. block size 34 B)
 Minimally replicated blocks:   2 (100.0 %)
 Over-replicated blocks:        0 (0.0 %)
 Under-replicated blocks:       0 (0.0 %)
 Mis-replicated blocks:         0 (0.0 %)
 Default replication factor:    1
 Average block replication:     1.0
 Missing blocks:                0
 Corrupt blocks:                0
 Missing replicas:              0 (0.0 %)
 Blocks queued for replication: 0

Erasure Coded Block Groups:
 Total size:    0 B
 Total files:   0
```

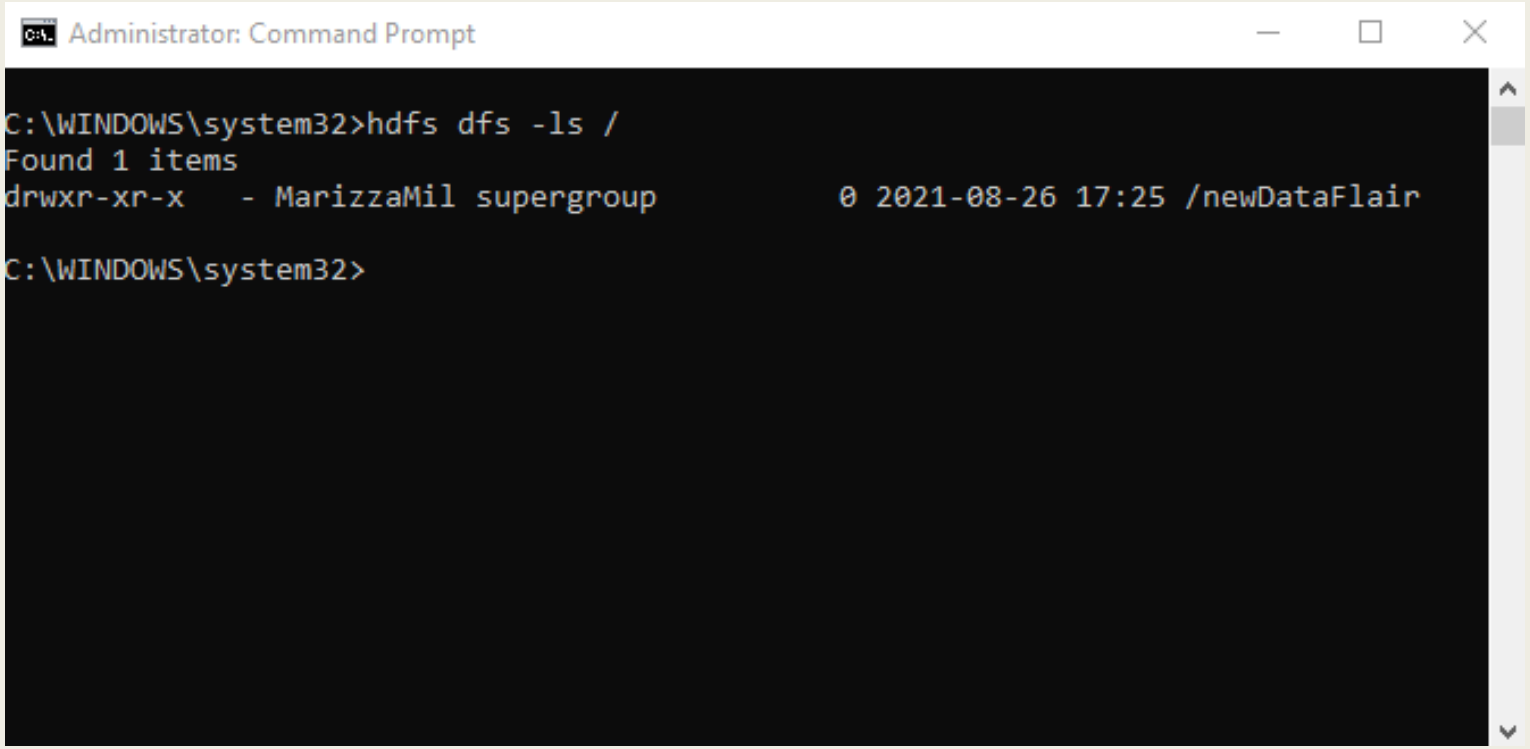
- **fsck**

HDFS Command to check the health of the Hadoop file system.

Command: **hdfs fsck /**



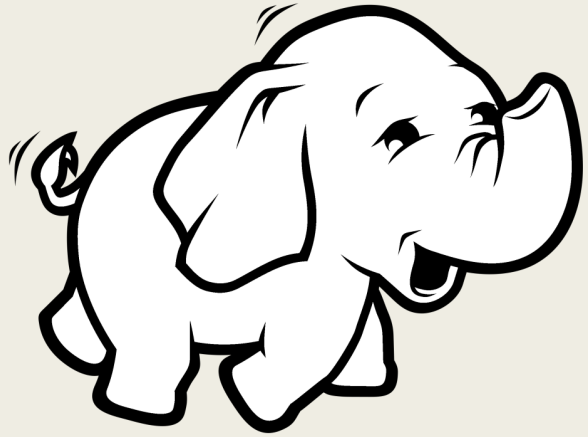
HDFS commands



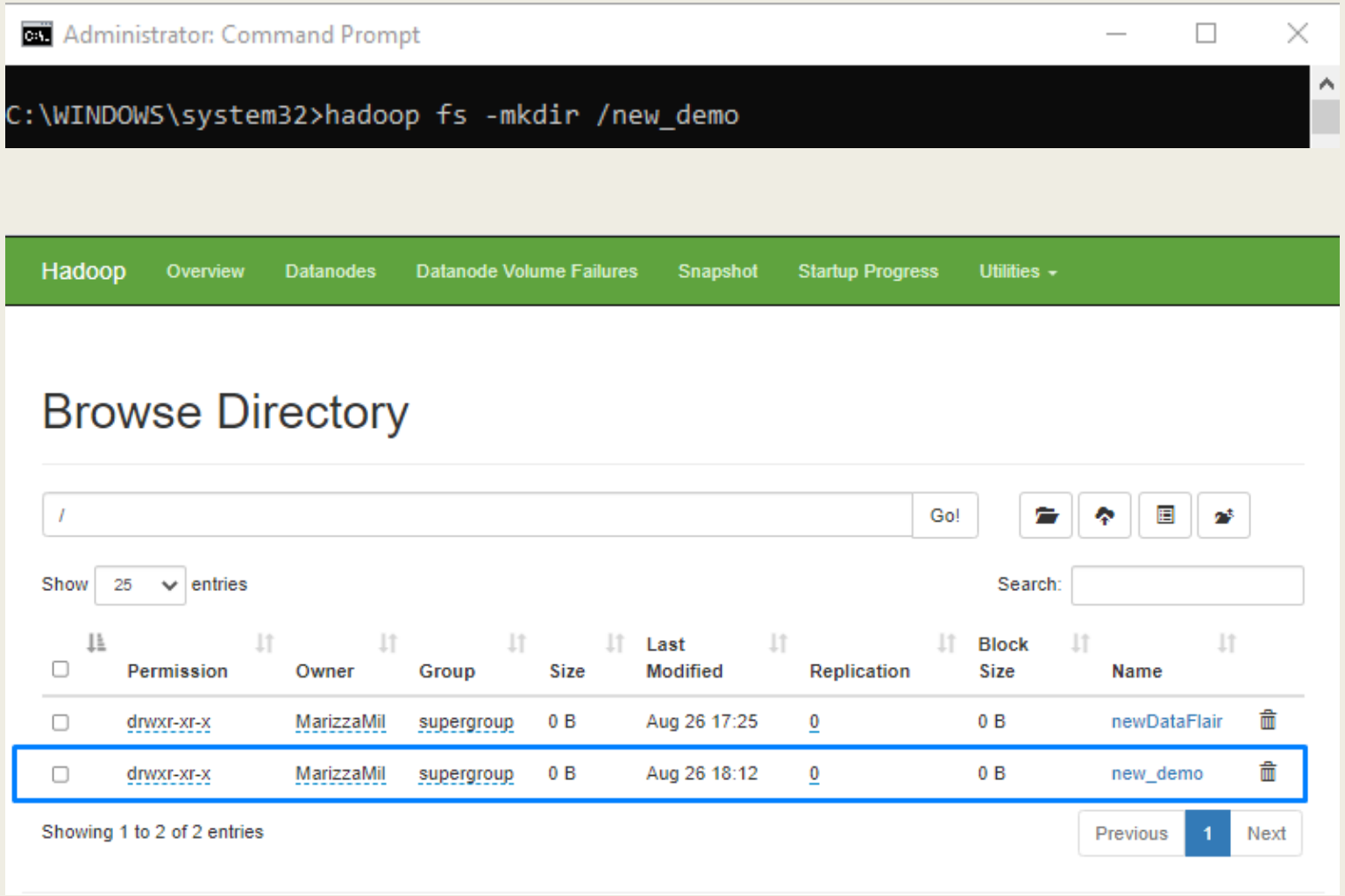
- **ls**

HDFS Command to display the list of Files and Directories in HDFS.

Command: hdfs dfs -ls /



HDFS commands

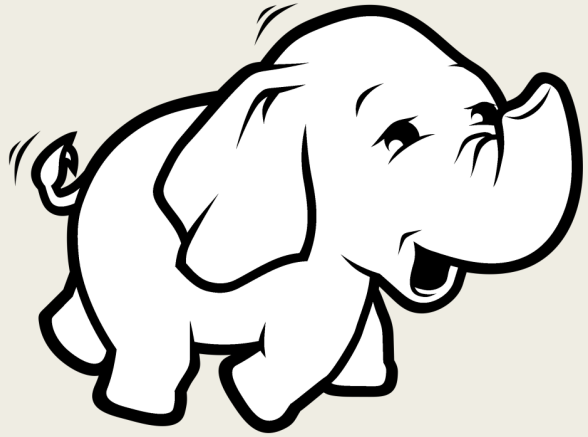


- mkdir

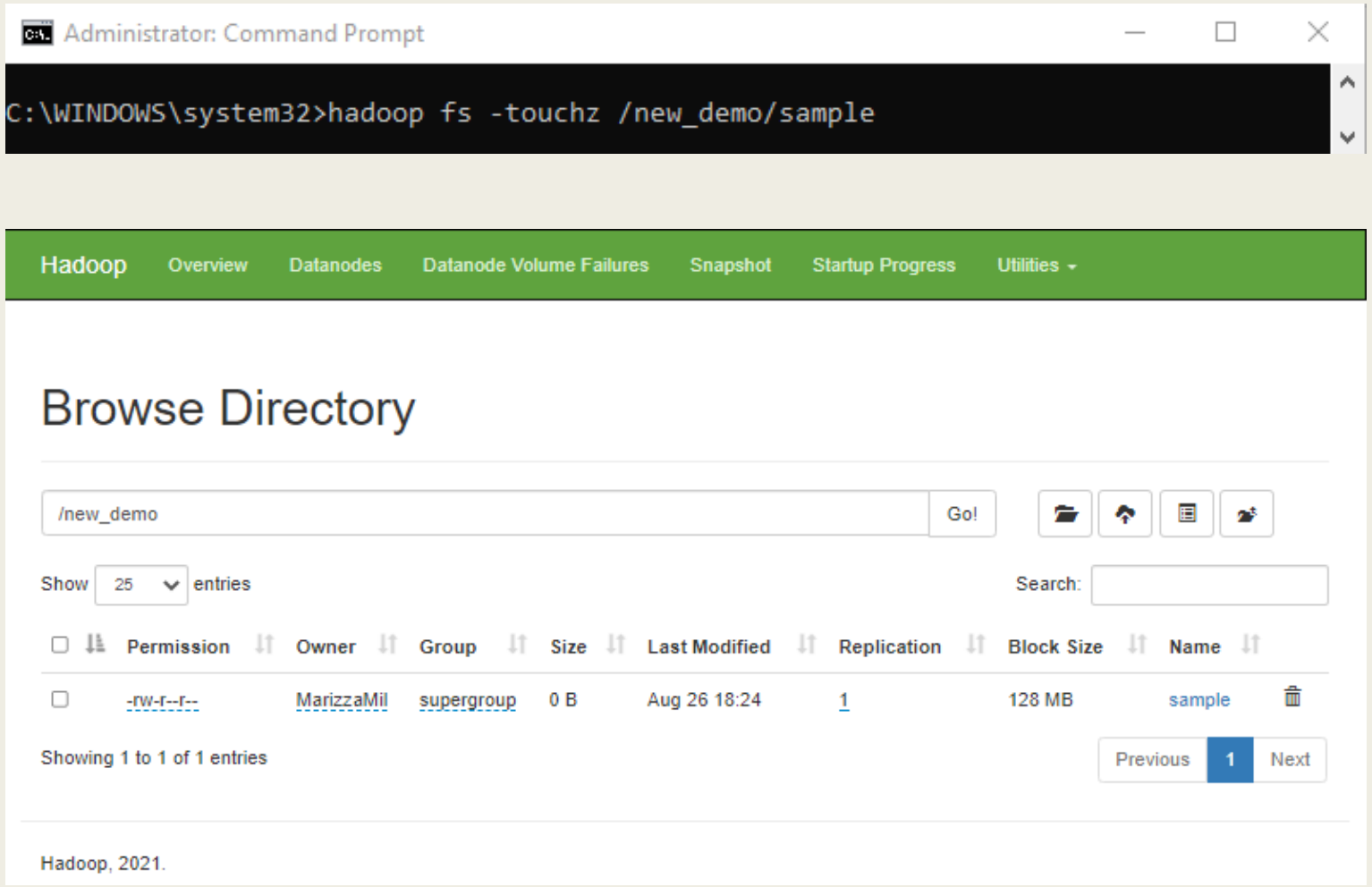
HDFS Command to create the directory in HDFS.

Usage: hadoop fs –mkdir /directory_name

Command: hadoop fs –mkdir /new_demo



HDFS commands

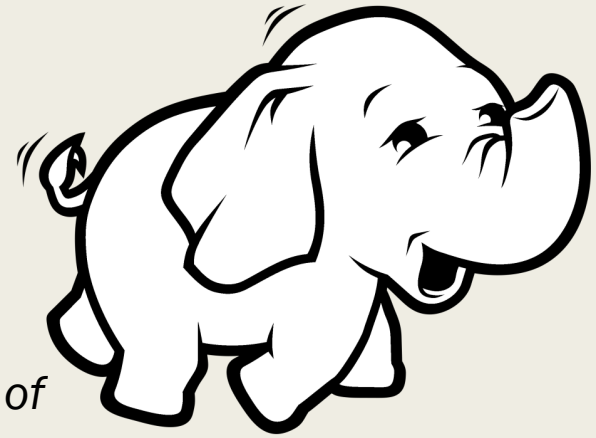


- touchz

HDFS Command to create a file in HDFS with file size 0 bytes.

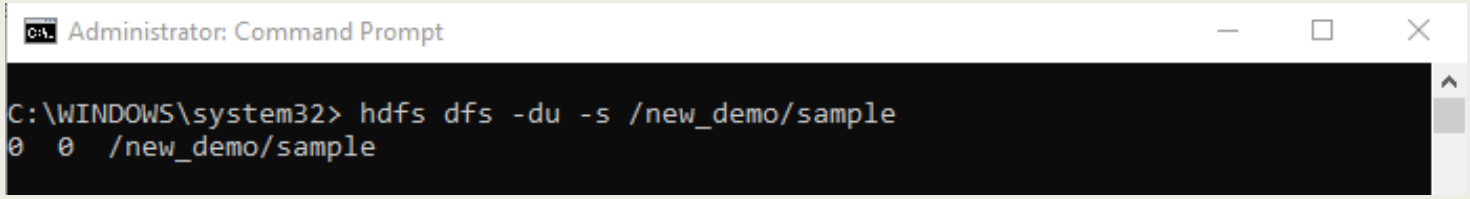
Usage:
hadoop fs –touchz /directory/filename

Command:
hadoop fs –touchz /new_demo/sample



Note: *Here we are trying to create a file named "sample" in the directory "new_demo" of hdfs with file size 0 bytes.*

HDFS commands

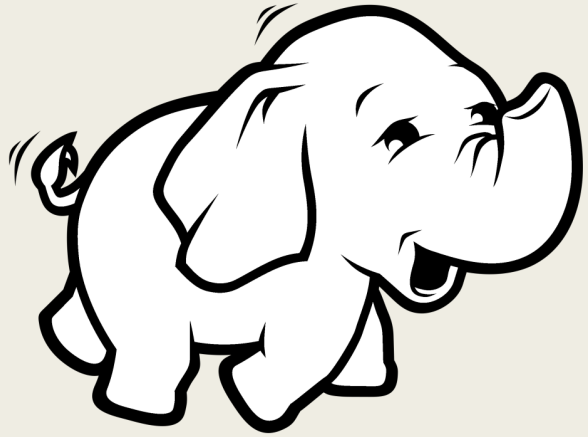


- du

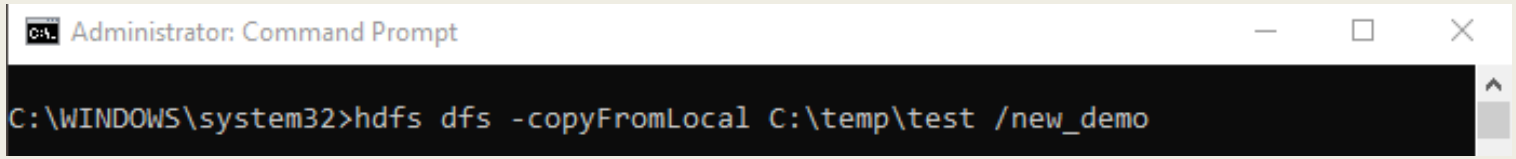
HDFS Command to check the file size.

Usage:
hdfs dfs –du –s /directory/filename

Command:
hdfs dfs –du –s /new_demo/sample



HDFS commands

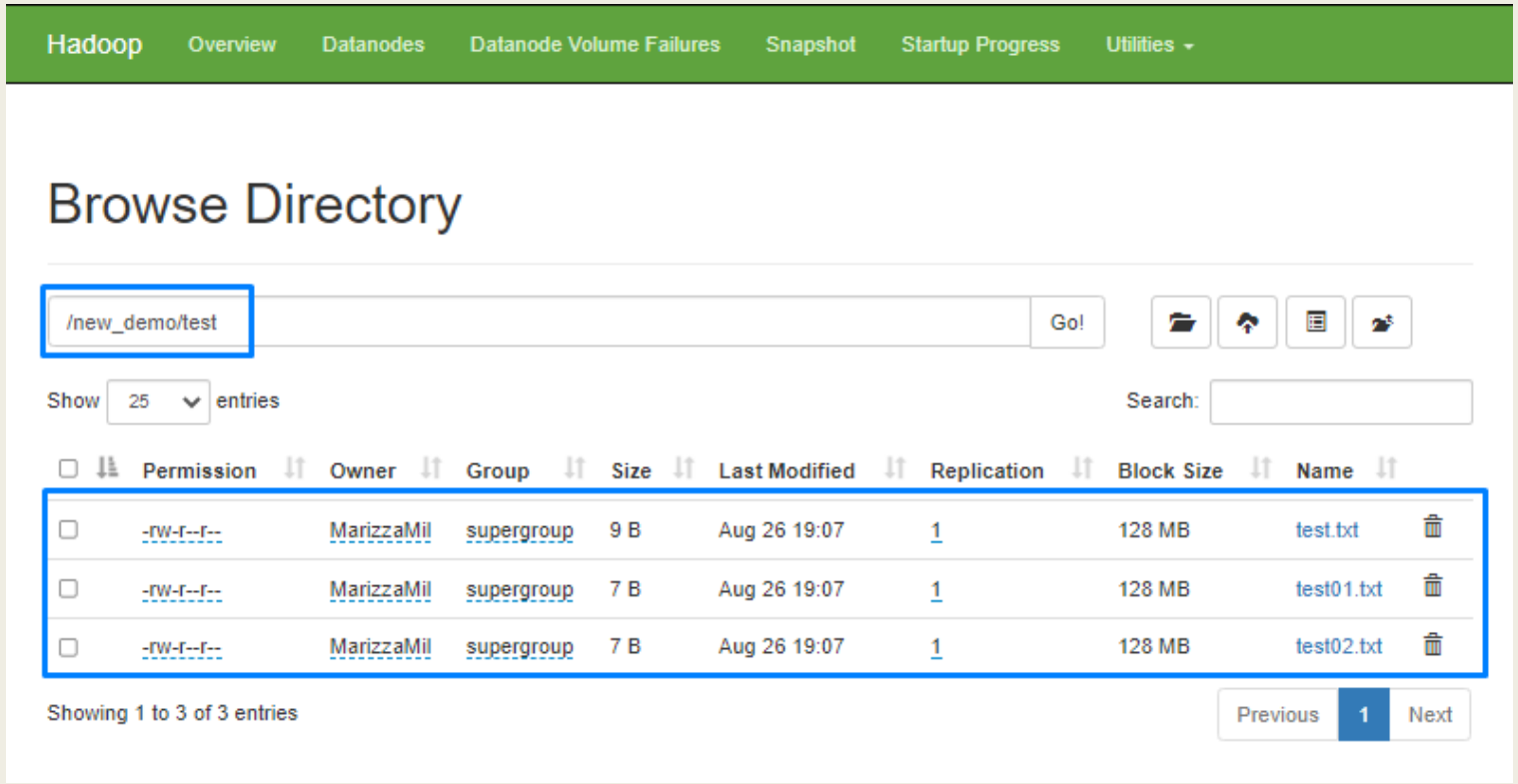


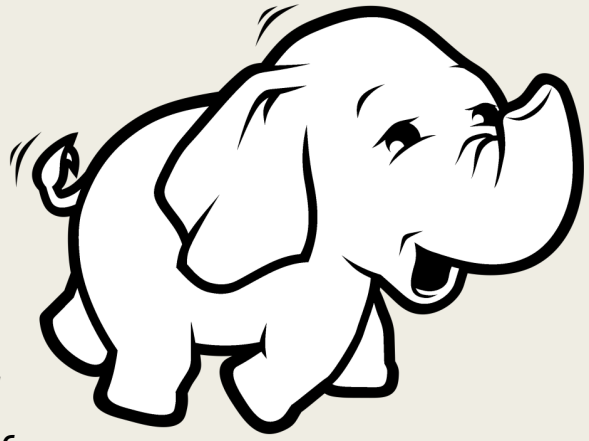
- copyFromLocal

HDFS Command to copy the file from a Local file system to HDFS.

Usage:
hdfs dfs -copyFromLocal <localsrc> <hdfs destination>

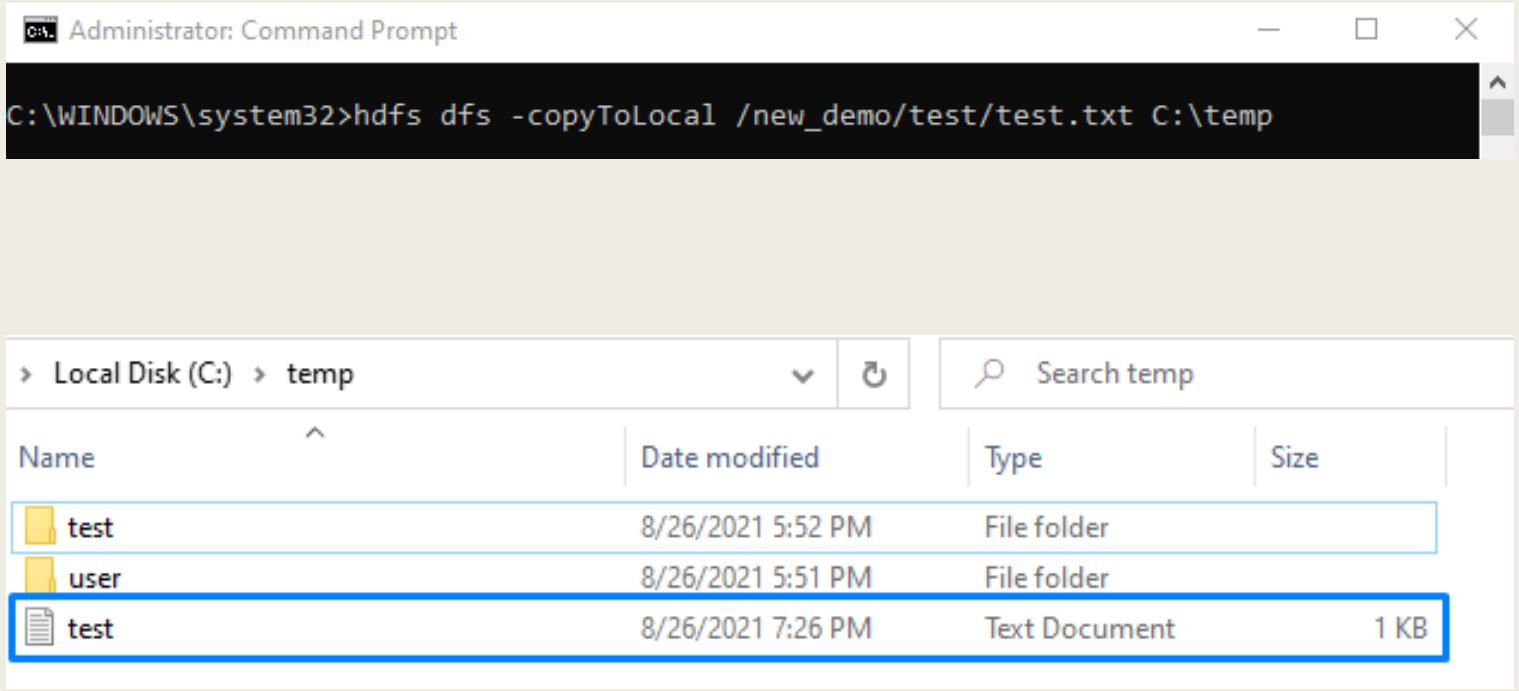
Command:
hdfs dfs -copyFromLocal C:\temp\test /new_demo





Note: *Here the test is the directory with files present in the local directory C:\temp and after the command gets executed the test file will be copied in /new_demo directory of*

HDFS commands



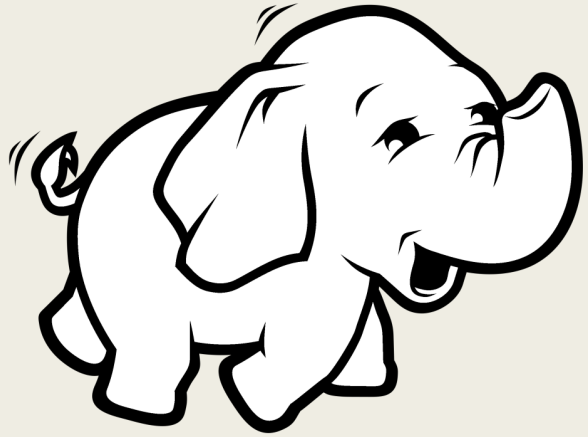
- copyToLocal

HDFS Command to copy the file from HDFS to Local File System

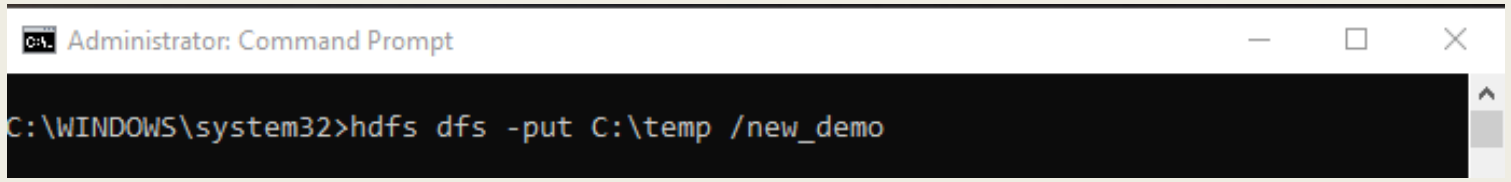
Usage:
hdfs dfs -copyToLocal <hdfs source> <localdst>

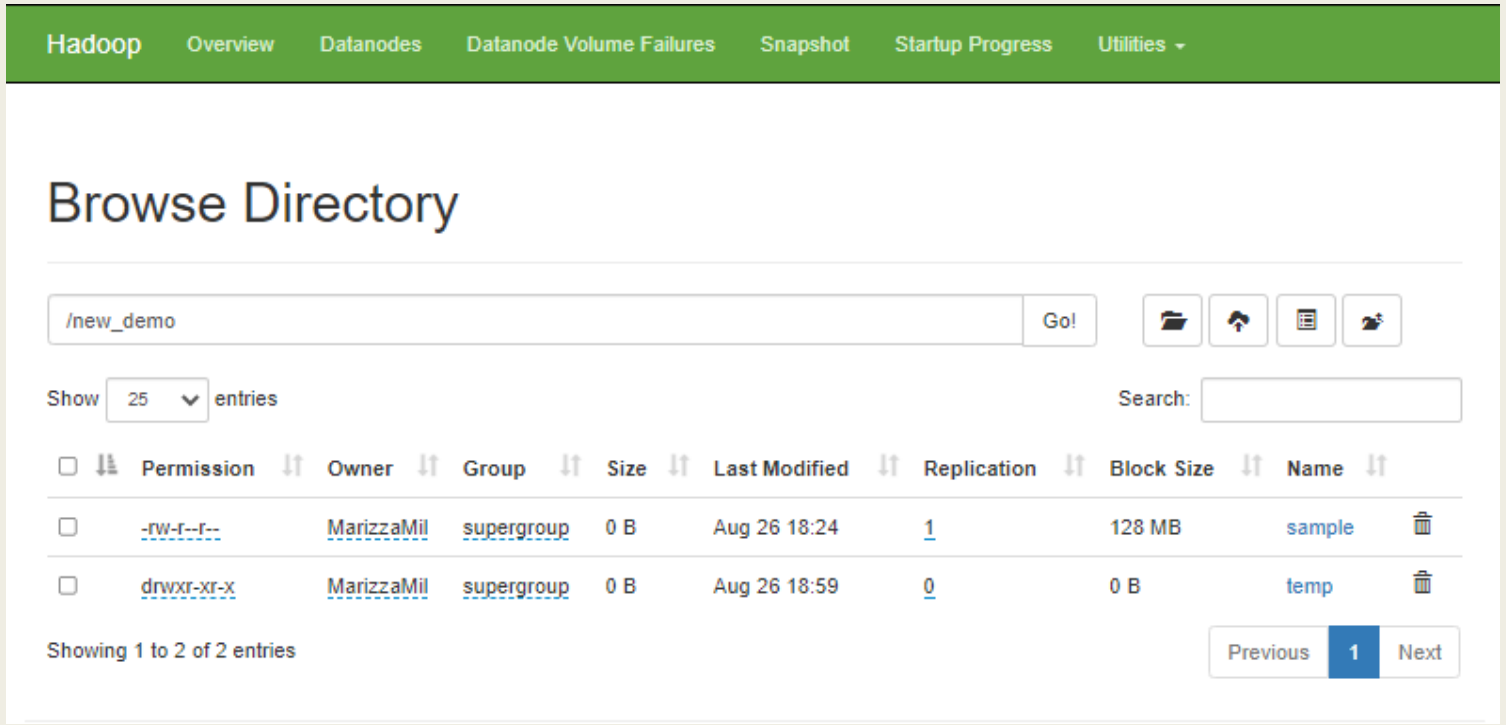
Command:
hdfs dfs -copyToLocal /new_demo/test/test.txt C:\temp

Note: *Here test.txt is a file present in the new_demo/test directory of HDFS and after the command gets executed the test file will be copied to local directory C:\temp*



HDFS commands



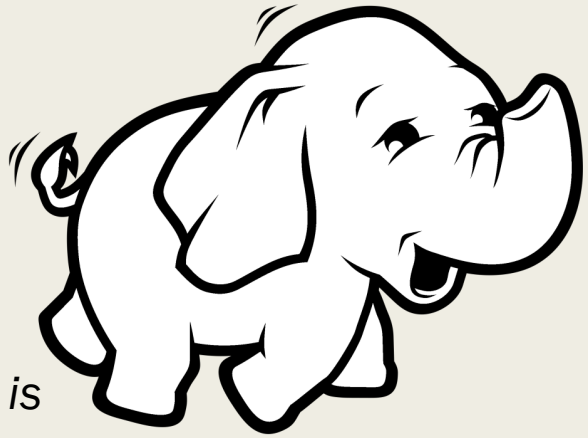


- **put**

HDFS Command to copy single source or multiple sources from local file system to the destination file system.

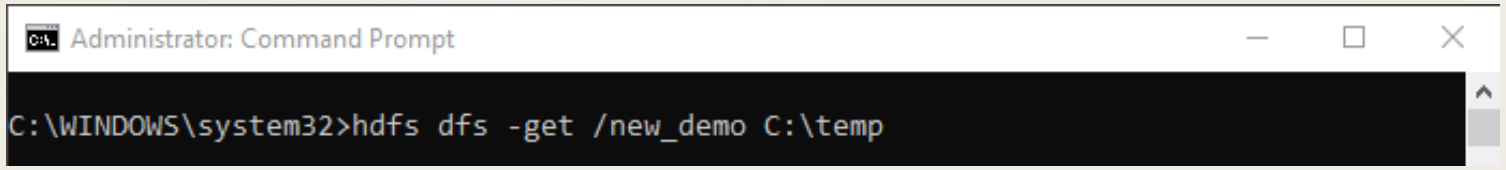
Usage:
hdfs dfs -put <localsrc> <destination>

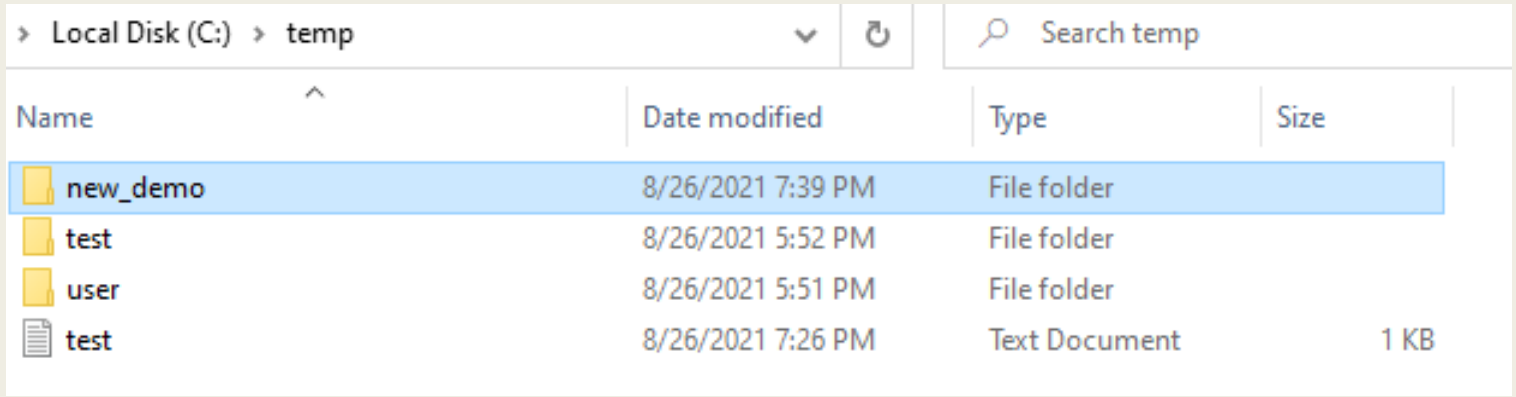
Command:
hdfs dfs -put C:\temp /new_demo



Note: *The command copyFromLocal is similar to put command, except that the source is restricted to a local file reference.*

HDFS commands



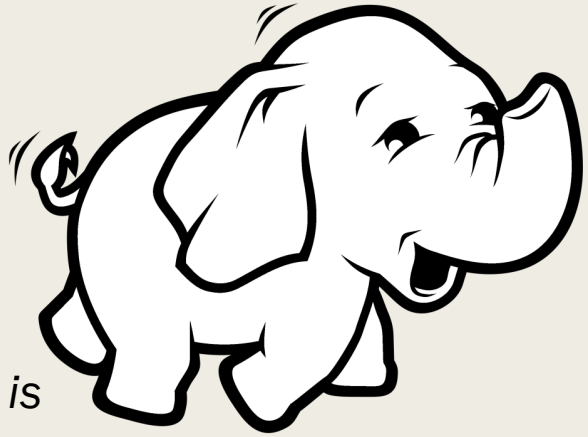


- get

HDFS Command to copy files from hdfs to the local file system.

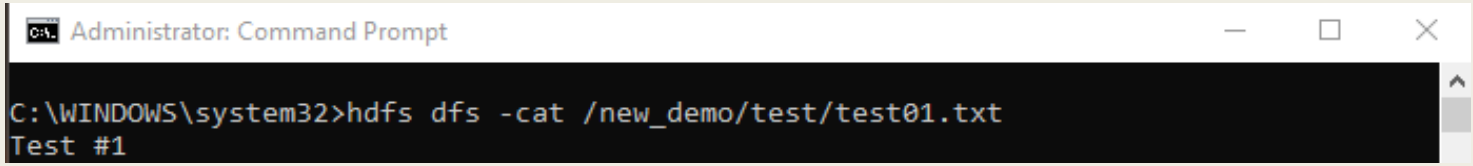
Usage:
hdfs dfs -get <src> <localdst>

Command:
hdfs dfs –get /new_demo C:\temp



Note: *The command copyFromLocal is similar to put command, except that the source is restricted to a local file reference.*

HDFS commands

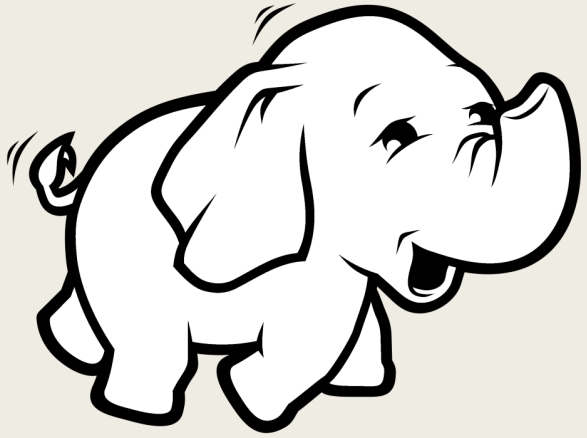


- cat

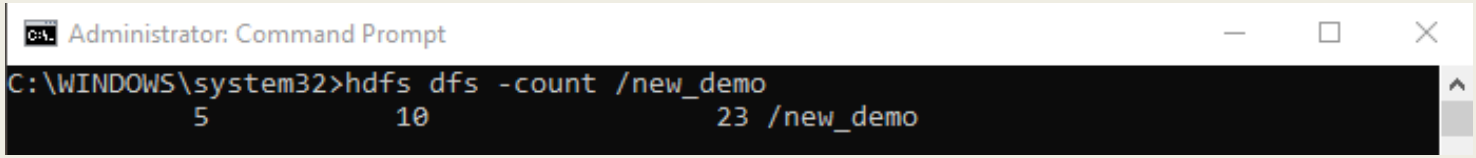
HDFS Command that reads a file on HDFS and prints the content of that file to the standard output.

Usage:
hdfs dfs –cat /path/to/file_in_hdfs

Command:
hdfs dfs –cat /new_demo/test/test01.txt



HDFS commands

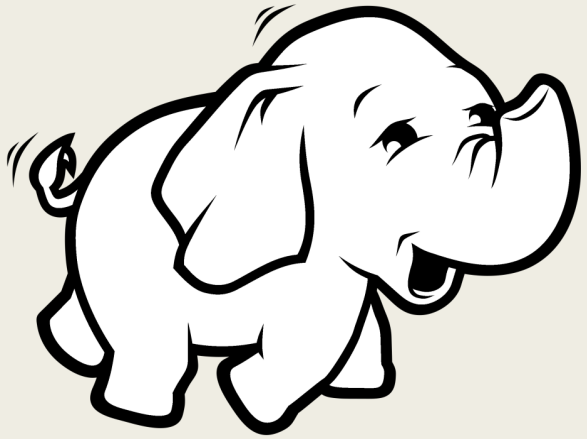


- count

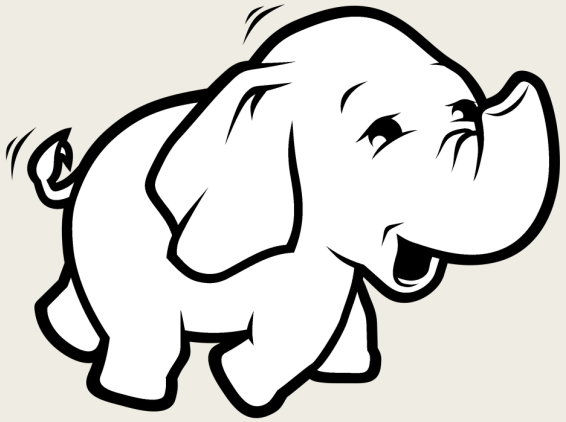
HDFS Command to count the number of directories, files, and bytes under the paths that match the specified file pattern.

Usage:
hdfs dfs -count <path>

Command:
hdfs dfs –count /new_demo



HDFS commands



- rm

HDFS Command to remove the file from HDFS.

Usage:

hdfs dfs –rm <path>

Command:

hdfs dfs –rm /new_demo/test

- rm -r

HDFS Command to remove the entire directory and all of its content from HDFS.

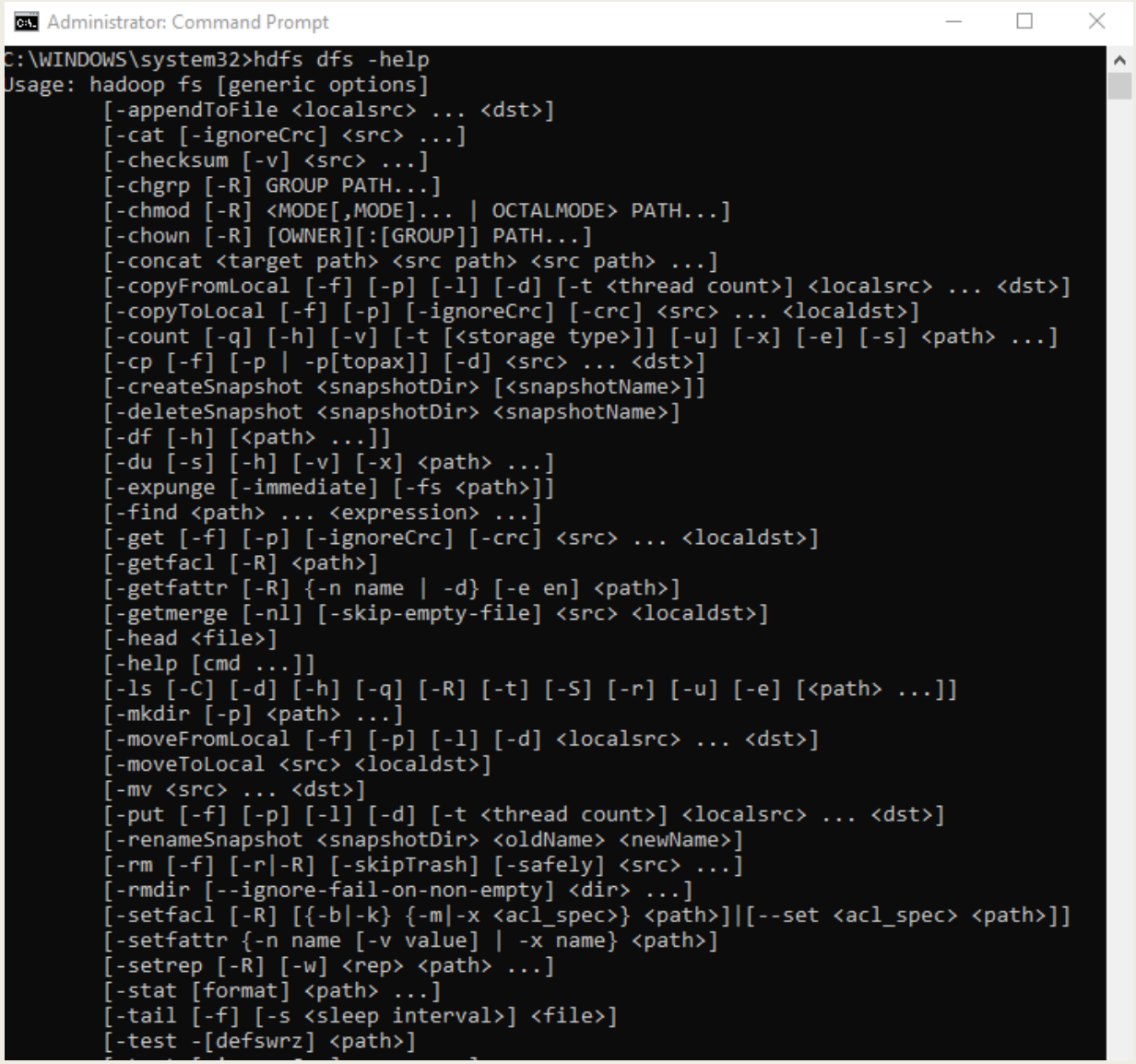
Usage:

hdfs dfs -rm -r <path>

Command:

hdfs dfs -rm -r /new_demo

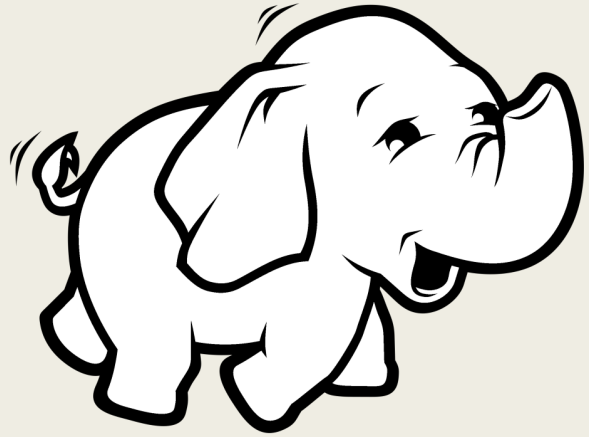
HDFS commands



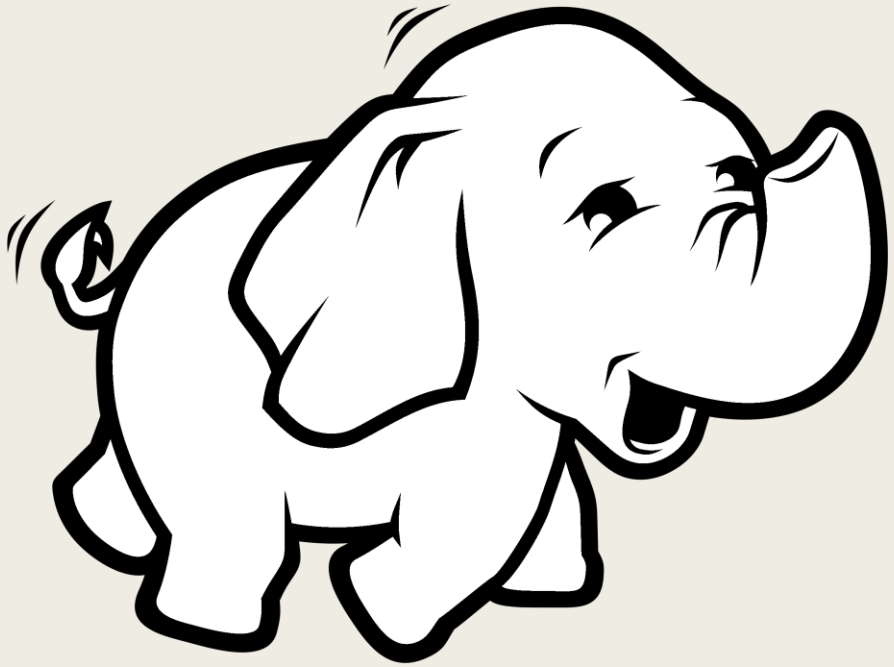
- help

HDFS Command that displays help for given command or all commands if none is specified.

Command: hdfs dfs -help



HDFS COMMANDS



For more HDFS Commands, you may refer Apache Hadoop documentation **here.**