HDFS Blocks

What are HDFC blocks?

def : the smallest unit of storage in HDFS

functionality: when files are stored in HDFS are divided into blocks & distributed across multiple data nodes

default size: 128 MB (much larger than traditional file systems which use KB or MB)

File (600mB) ———

A	B	C	D	E
128mB	128mB	128mB	128mB	8t mD

2GD 90 mD

metadata

1.56L × 5

**Benefits : (larger Block Size)**

Reduced Metadata Management : name node has less metadata to track

Efficient for large files: Optimized for handling big data workloads

Distributed Storage & computation : enables parallel processing across multiple nodes

Block size Configuration :

→ Decreasing Block Size (eg. 1MB)

5 mD cruied

5×3 = 15 blocks

600 mB   ⟶   1mD

B1	B2	B3	B4	B5	B6 ...		B 600

1mD
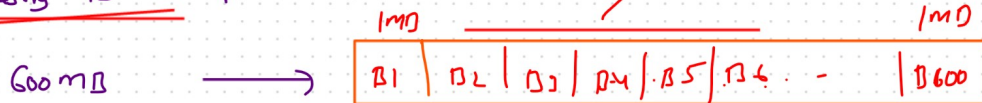
pros
- Reduces wasted space for small files
- Increased parallelism [i.e processing on multiple data blocks]

cons
- Significantly increases metadata load on the name Node

2   5 mB

(Increasing Block Size (eg 512 mB)

- reduced metadata overhead
- reduced parallelism
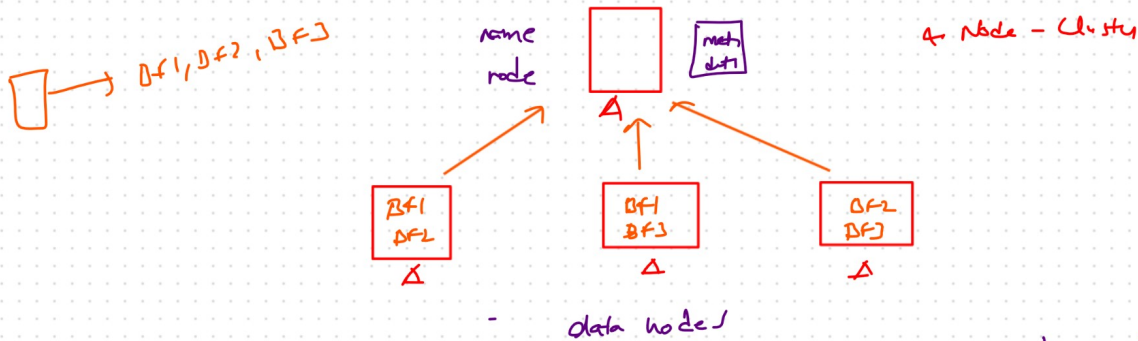- may increase processing time for large files

Why RSMB is default ?

* determined through extensive research & testing

* Balances :

    * metadata overload
    * parallelism needs
    * processing efficiency   } factor
    * storage efficiency

HDFS fault Tolerance & Replication

Hadoop (HDFS & mapReduce)

→ DF1, DF2, DF3

name node   [meta data]   → Node - Cluster

data nodes

BF1 DF2 · BF1 BF3 · DF2 DF3

* data nodes are built on Commodity hardware, makes them susceptible to failures

* system must be fault-tolerant to prevent data loss

Replication in HDFS

* purpose: process of making multiple copies of data blocks
    └→ ensures fault tolerance by storing copies among different data nodes

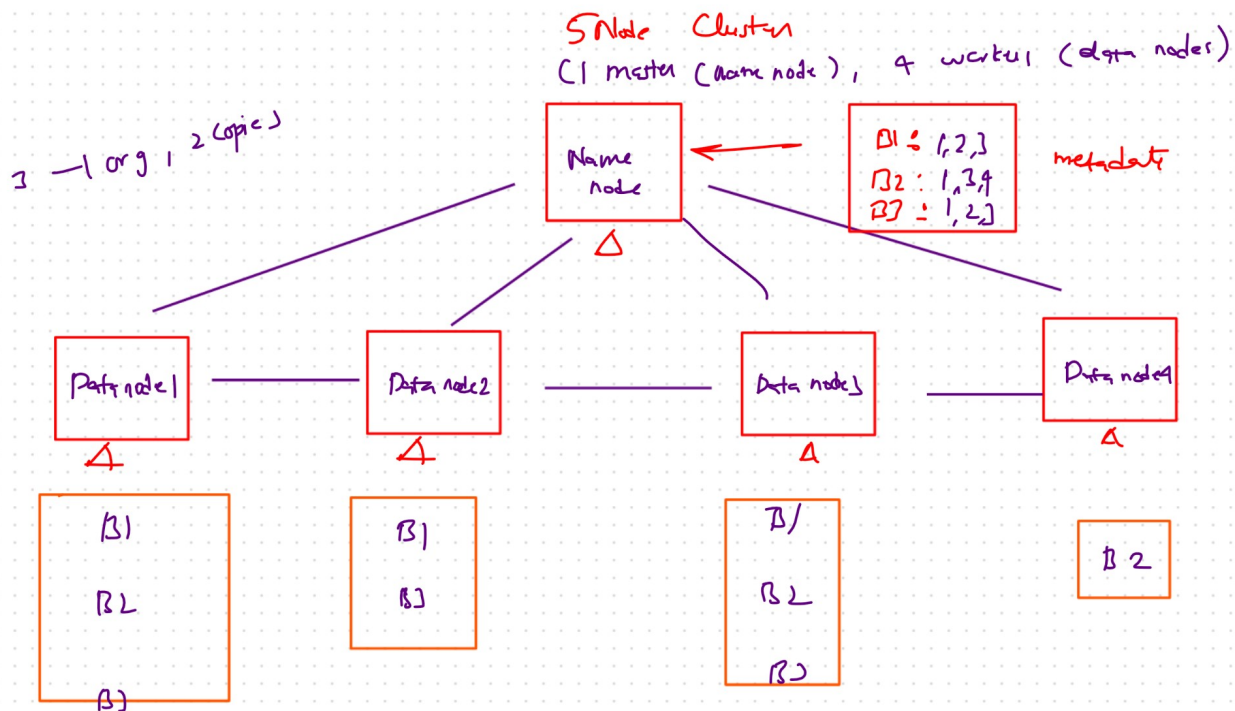default replication factor: 3 (1 original + 2 copies)

Storage implications :   1GB $\left[1GB + [1GB, 1GB] \right]$ ⇒ 3GB

1GB of data requires 3GB of storage space

Why replication is need
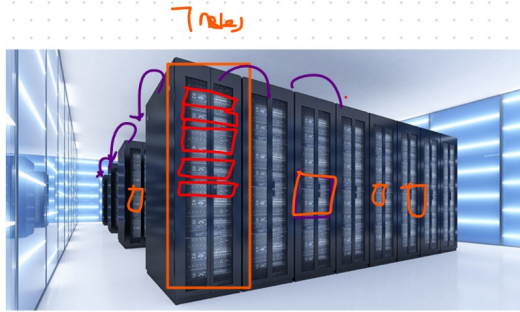
* if one data node fails, the system can retrieve data from nodes with replica
* ensure data availability
* prevents data loss
* Allows the cluster to continue working without interruption

ex: ## Replication Distribution

300MB [file] —— blocks ——> B1, B2, B3

5 Node Cluster
(1 master (name node), 4 workers (data nodes))

3 → 1 org 1 2 copies

Name node	B1 : 1,2,3
	B2 : 1,3,4
	B3 : 1,2,3

metadata

Data node1 — Data node2 — Data node3 — Data node4

Data node1:
B1
B2
B3

Data node2:
B1
B3

Data node3:
B1
B2
B3

Data node4:
B2

Rack Awareness in Hadoop

What is Rack ?

→ A rack is a collection of data nodes that are physically close together

→ data nodes within the same rack are connected via wires

→ different racks are connected to each other via network switches

Why Rack Awareness ?

① Fault Tolerance:

② Optimized Network Traffic

* Since data distributed across racks, it helps balancing network usage

* reduces network latency & congestion

* prevents all data flowing from a single n/w path

DataCenter