

06 – Hadoop Distributed File System

Abdel Dadouche
DJZ Consulting

adadouche@hotmail.com
@adadouche

What is HDFS?

Hadoop Distributed File System

- Files and directories are organized in a tree structure just like your local file system
- Allows you to see these files and folders stored on many (hundreds or thousands) as a single file system

This is totally transparent to the developer or the end-user!

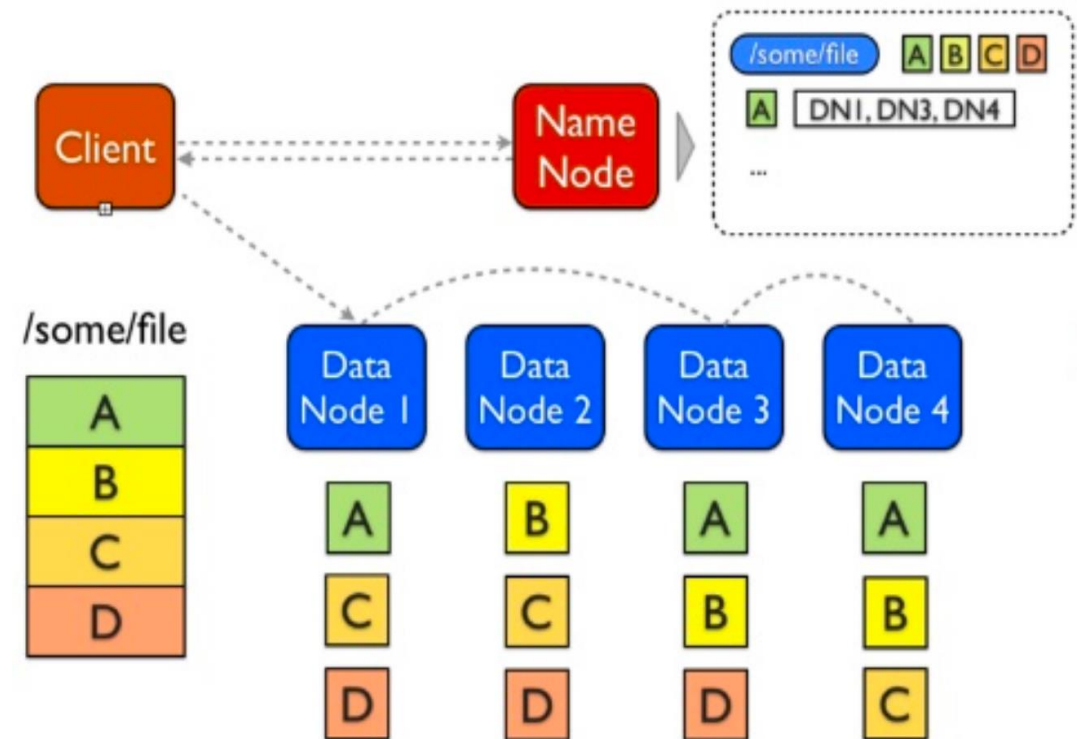
Hadoop Distributed File System

- Stored as chunks which are replicated across on multiple computers
- Therefore if a disk fails, the complete file will remain available as the potential chunks stored on the crash instance will be available somewhere else
- HDFS is also able to optimize and limit data transfer between node for both the replica and processing (helth check)

!

A Master and Slaves Cluster Architecture

- In a cluster architecture each machine / compute represents a node which has a role
- NameNode (master):
 - There is usually one master
 - Keeps a registry with the file names/path and the chunk count and location
 - Considered the Single Point Of Failure (SPOF)
- DataNodes:
 - There is usually multiple slaves (one per node)
 - Stores the file chunks
- There is a constant communication between the DataNode and the NameNode
 - Share current health, problems...

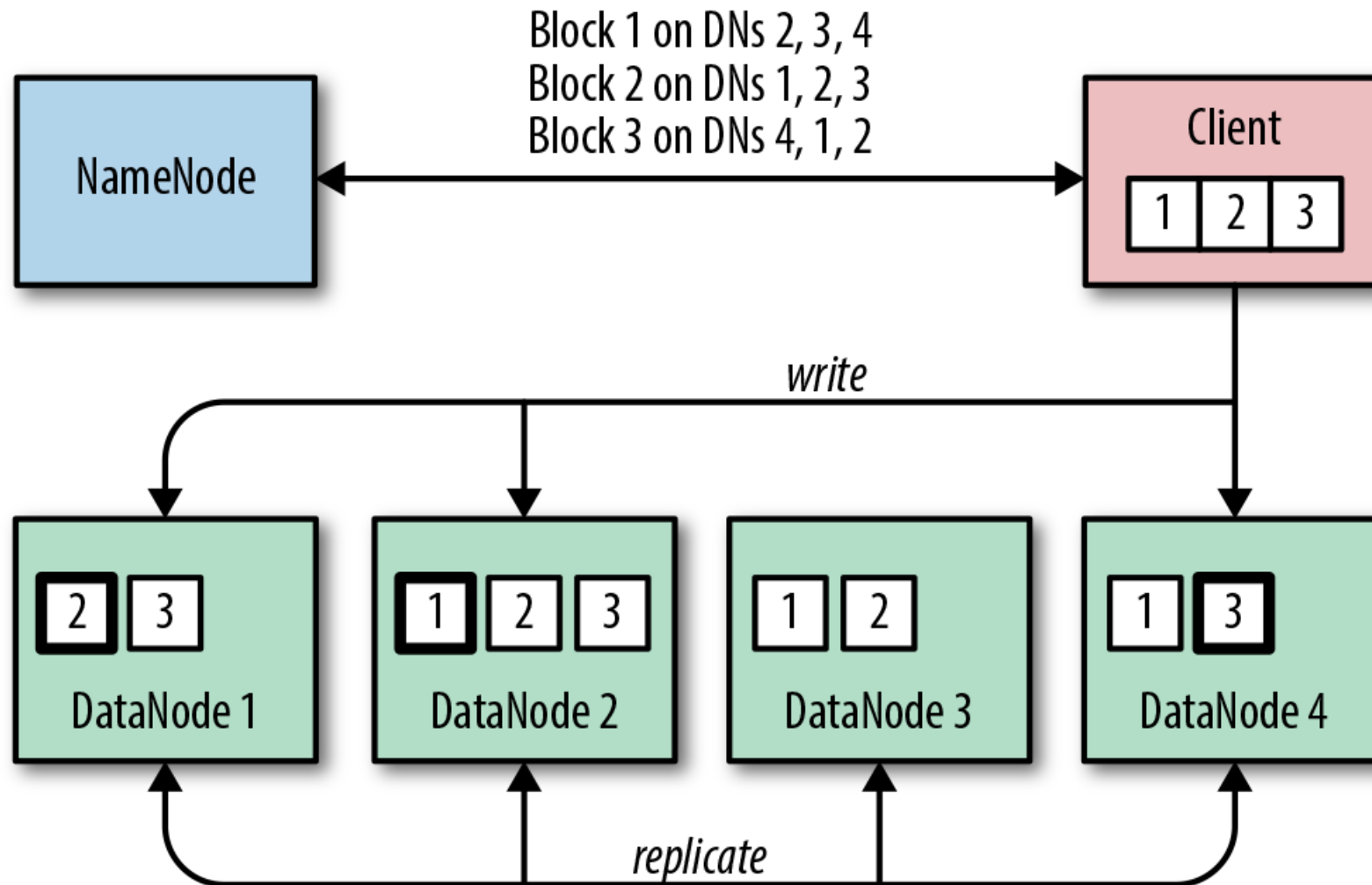


Important Configuration/Parameters

- BlockSize:
 - This the chunk size and will determine how many blocks will be created out of a file
- Replication Factor
 - This the number of copies a single block that must be distributed across the cluster
- There is no good or bad number for these settings
 - Depends on the file content (CSV, images)
 - Depends on the files size (smalll, medium, large, xtra large)
 - Depends on the type of processes that will utilize the file
 - Depends on the cluster size and resilience

HDFS Write Operation Explained

- To write a file, you can use the command:
`hadoop fs [generic options] -put [-f] [-p] <localsrc> ... <dst>`
- Based on the file size and the block size, the client will request the NameNode to provide the locations where each block will go, so the client can execute it.
- The replica copy will be handled by the DataNodes not the clients.
- The NameNode never see the file itself!



HDFS Read Operation Explained

- To read a file you can use the command:
`hadoop fs [generic options] -get [-p] [-ignoreCrc] [-crc] <src> ... <localdst>`
- The client will request the NameNode to provide the locations for each block (including the replica), so the client can get them it.
- In case, a block on a node is not available, then the next block replica will be retrieved. The replica list is usually sorted based on the data node health
- The NameNode never see the file itself!

Important HDFS commands

`hdfs dfs -ls /-cat / -tail / -mkdir`

Basic file operation

`hdfs dfs -stat`

Get the file statistics like the number of replica etc.

`hdfs fsck`

Check the health of the file system

`hdfs dfsadmin -report`

Get the cluster status

`hdfs dfsadmin -printTopology`

Get the cluster topology

Summary

- The client transfer the file only once, and the DataNodes replicate across the cluster
- If a DataNode crash there is a copy somewhere else!
- The NameNode never see the file content
- Everything is transparent to the developer/user
- The NameNode is the “Single Point Of Failure” (SPOF)