The Hadoop Distributed File System (HDFS):

A Storage System for Big Data

After this video you will be able to..

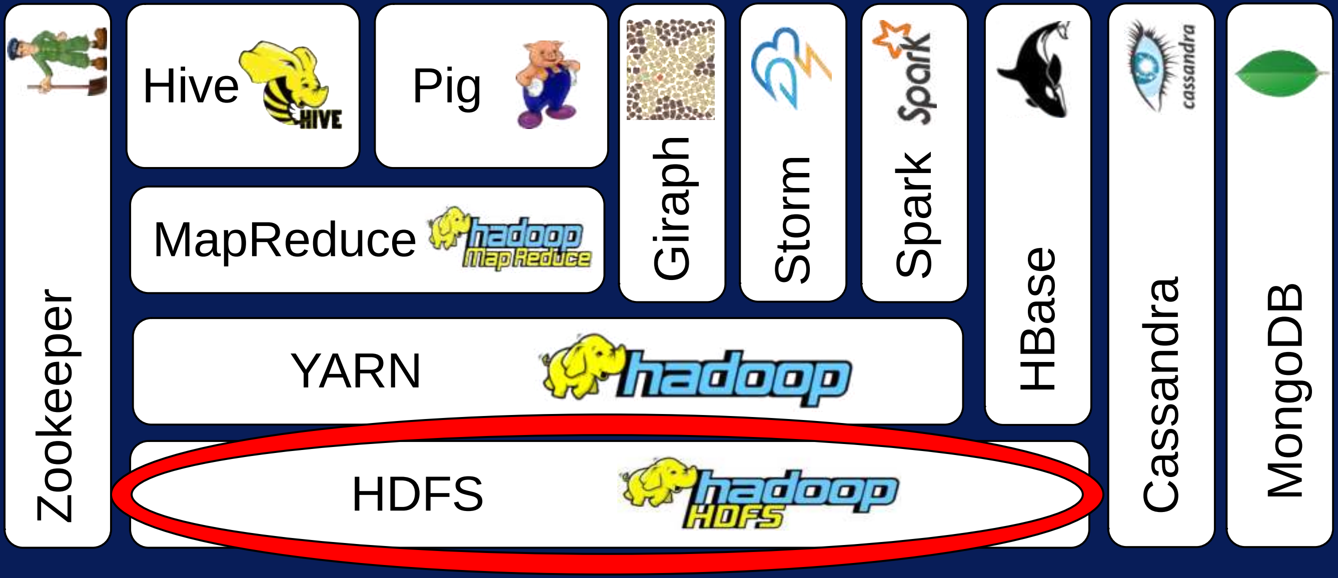
- Describe how HDFS provides scalable and reliable storage

- Differentiate two key HDFS components: the NameNode and the DataNode

HDFS = foundation for Hadoop ecosystem

Scalability

Reliability



Hive

Pig

Giraph

Storm

Spark

HBase

Cassandra

MongoDB

Zookeeper

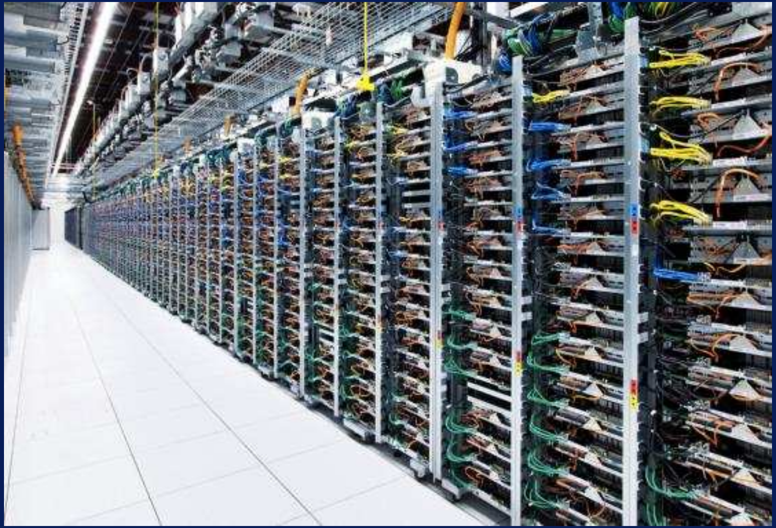
MapReduce

YARN

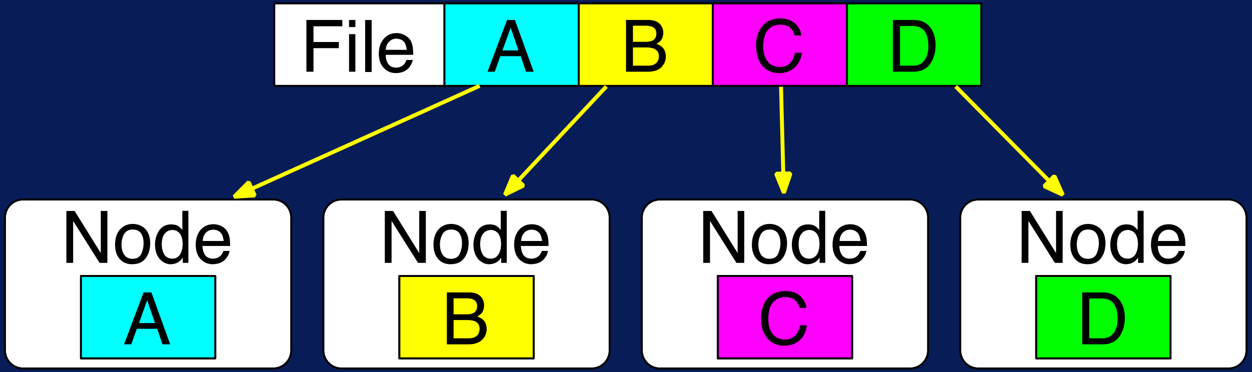
HDFS

Store massively large data sets

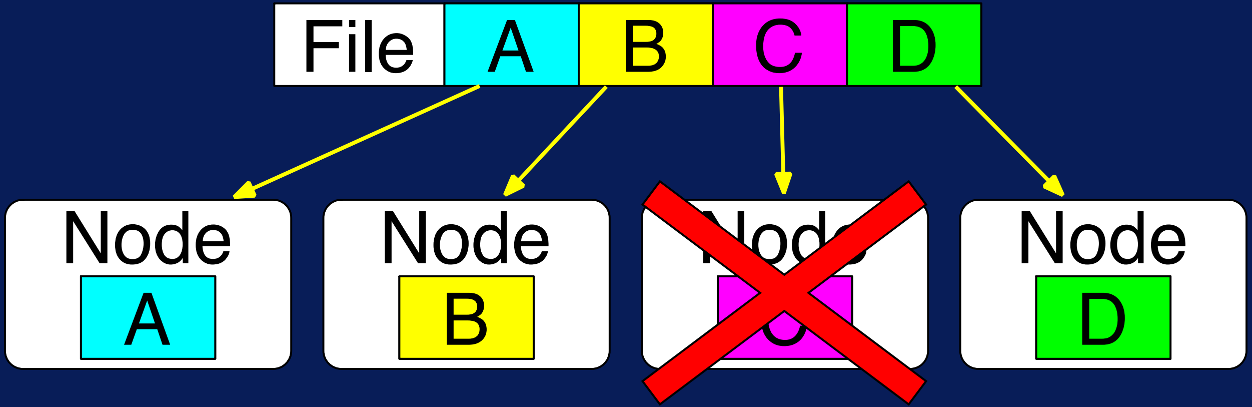
up to 200 Petabytes,
4500 servers,
1 billion files and blocks!



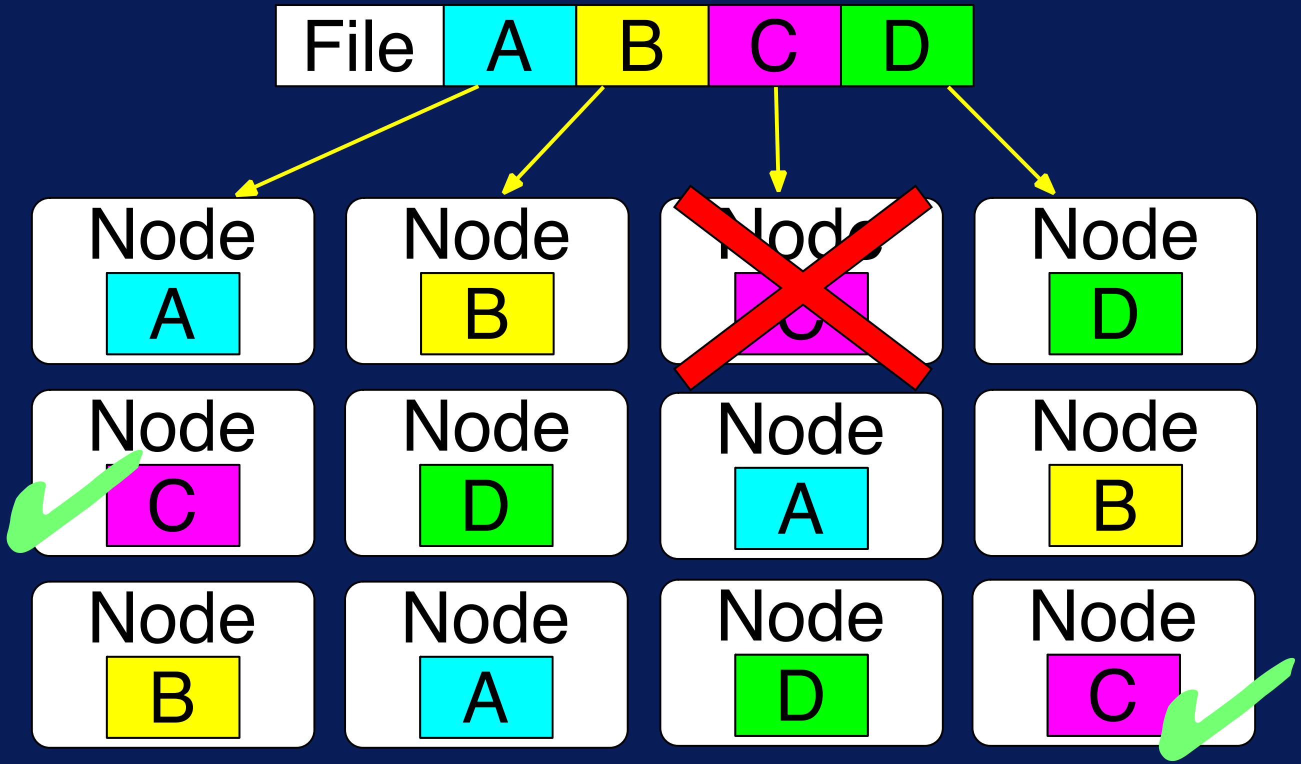
HDFS splits files across nodes for parallel access



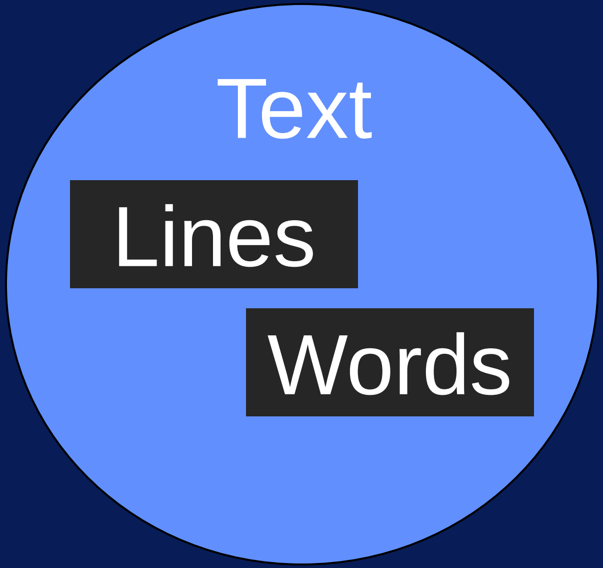
What happens if node fails?



Replication for fault tolerance



Customized reading to handle *variety* of file types

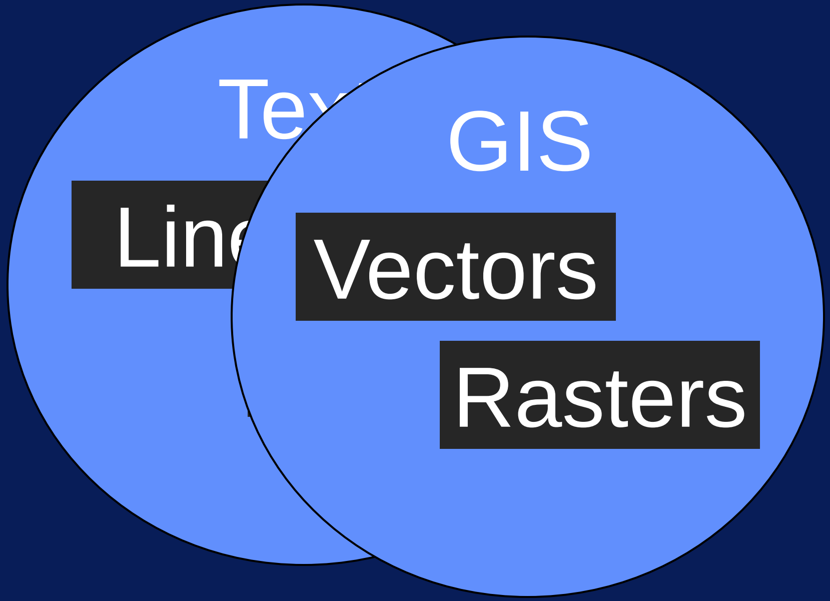


Text

Lines

Words

Customized reading to handle *variety* of file types



Tex

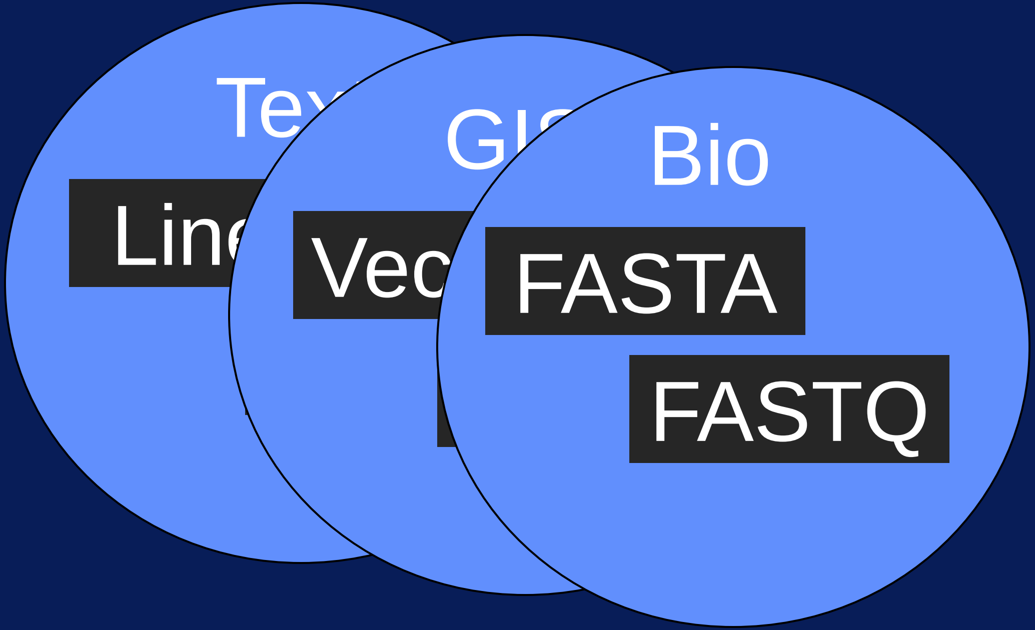
Line

GIS

Vectors

Rasters

Customized reading to handle *variety* of file types



Tex

Line

GI

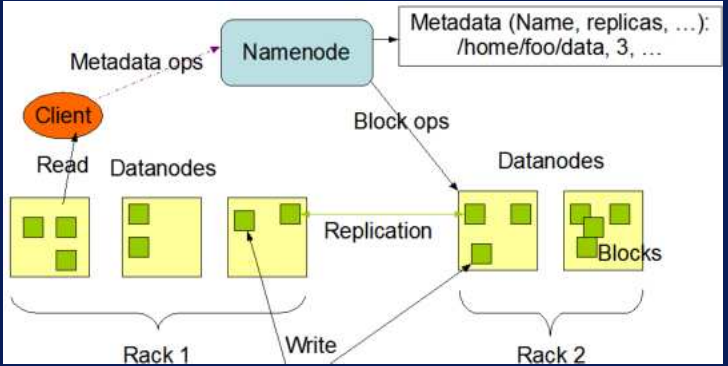
Vec

Bio

FASTA

FASTQ

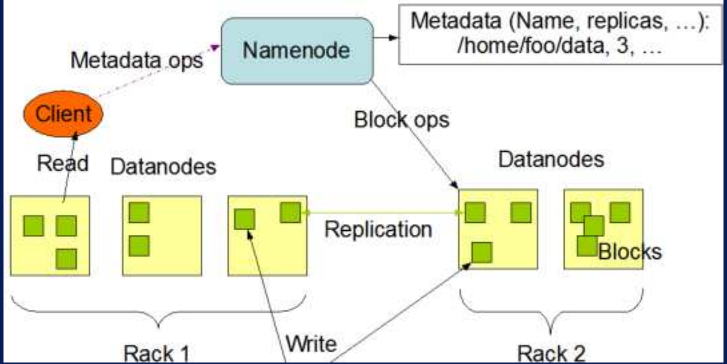
Two key components of HDFS



1. NameNode for metadata

2. DataNode for block storage

Two key components of HDFS



1. NameNode for metadata

   Usually one per cluster

2. DataNode for block storage

   Usually one per machine

The NameNode coordinates operations

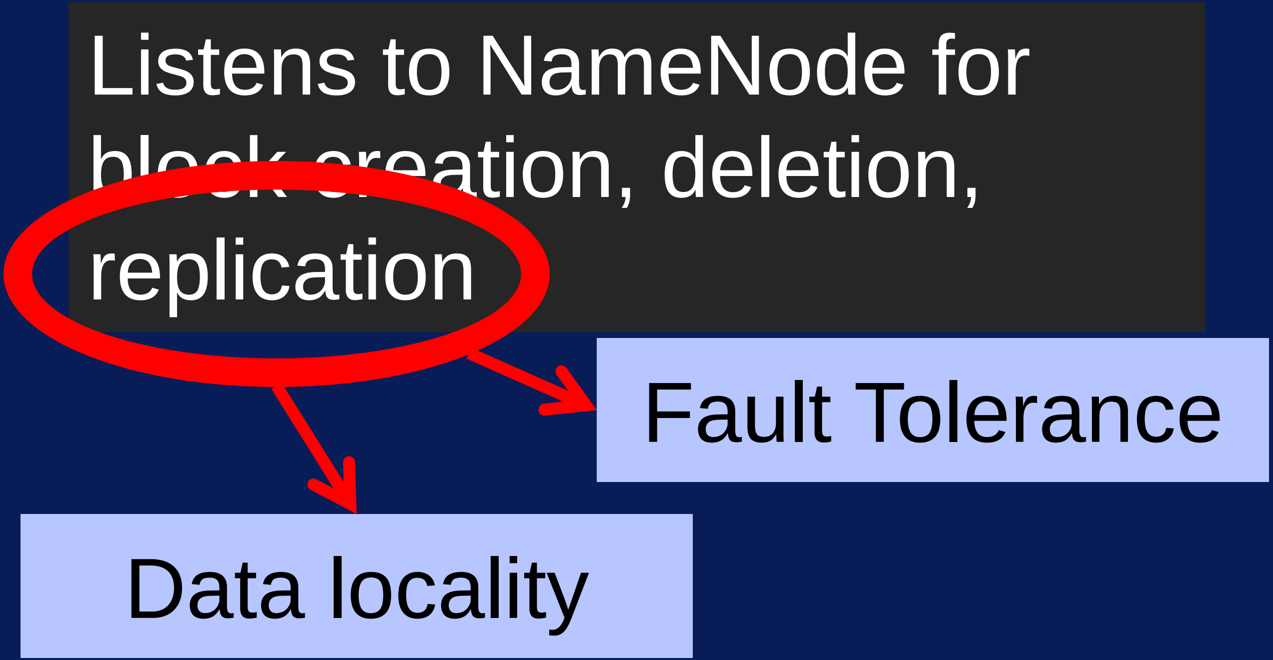
Keeps track of file name, location in directory, etc.

Mapping of contents on DataNode.

DataNode stores file blocks

Listens to NameNode for block creation, deletion, replication

DataNode stores file blocks



Listens to NameNode for block creation, deletion, replication

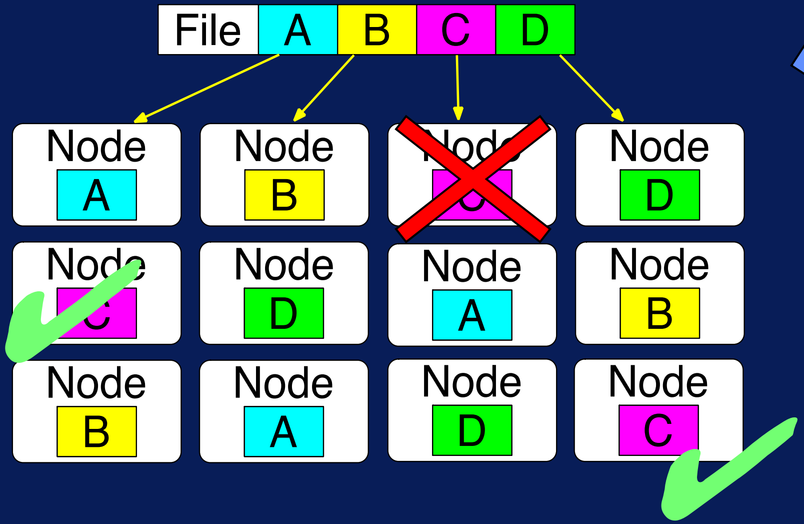
Fault Tolerance

Data locality

Data partitioning → Scalability

Data replication → Fault tolerance

→ Data locality


File
A
B
C
D
Node A
Node B
Node C
Node D
Node C
Node D
Node A
Node B
Node B
Node A
Node D
Node C