Introduction to Big Data
IN411 – M1S2 – 3 Credits

Course Contents

- Chapter 1: Overview of Big Data
- **Chapter 2: Hadoop Ecosystem**
- Chapter 3: Apache Spark Basics
- Chapter 4: Introduction to NoSQL Databases
- Chapter 5: Big Data Storage & Data Lake Concepts

Chapter 2:

Hadoop Ecosystem

Content

- Introduction to the Hadoop platform (HDFS, YARN, MapReduce)
- Hadoop distributed file system (architecture, reading/writing data)
- MapReduce paradigm & programming exercises
- Hadoop ecosystem tools introduction (Hive, Pig, HBase): short overview

Hadoop Platform Introduction

Introduction

- **Small** datasets are datasets that fit comfortably in RAM, leaving memory to spare for manipulation and transformations.
- They are usually no more than 2–4 GB in size, and complex operations like sorting and aggregating can be done without paging.
- Such tools have upper limits to their feasibility when working with datasets beyond a certain size.

Introduction

- **Medium** datasets are datasets that cannot be held entirely in RAM but can fit comfortably in a single compute's persistent storage.
- **Large** datasets are datasets that can neither fit in RAM nor fit in a single computer's persistent storage.

Introduction

- Two possible (and easy solutions) to not run into memory problems when processing data, are to:
- **Optimize Storage**
  - Store data in efficient memory formats (compression).
  - Store data in a database to allow for sql queries.
  - Use chunking to trim and thin out your data so it does fit in memory.

- **Scale** your problem:
  - Replace your existing equipment with faster, more efficient equipment (scale-up).
  - Hire more workers to work in parallel (scale-out).

Introduction

- The exponential growth of data (Big Data) exceeded the capabilities of traditional relational databases and centralized processing systems.
- Challenges included:
  - High cost of enterprise storage systems
  - Storage of unstructured and semi-structured data
  - Processing of unstructured and semi-structured data efficiently
  - Scalability limits
  - Fault tolerance limits
  - Throughput limits

Introduction

- **Apache Hadoop** emerged as an open-source framework enabling:
  - A platform for big data
  - Running applications on large clusters of commodity hardware
  - Distributed storage
  - Distributed processing
  - Fault tolerance

Core Design Principles

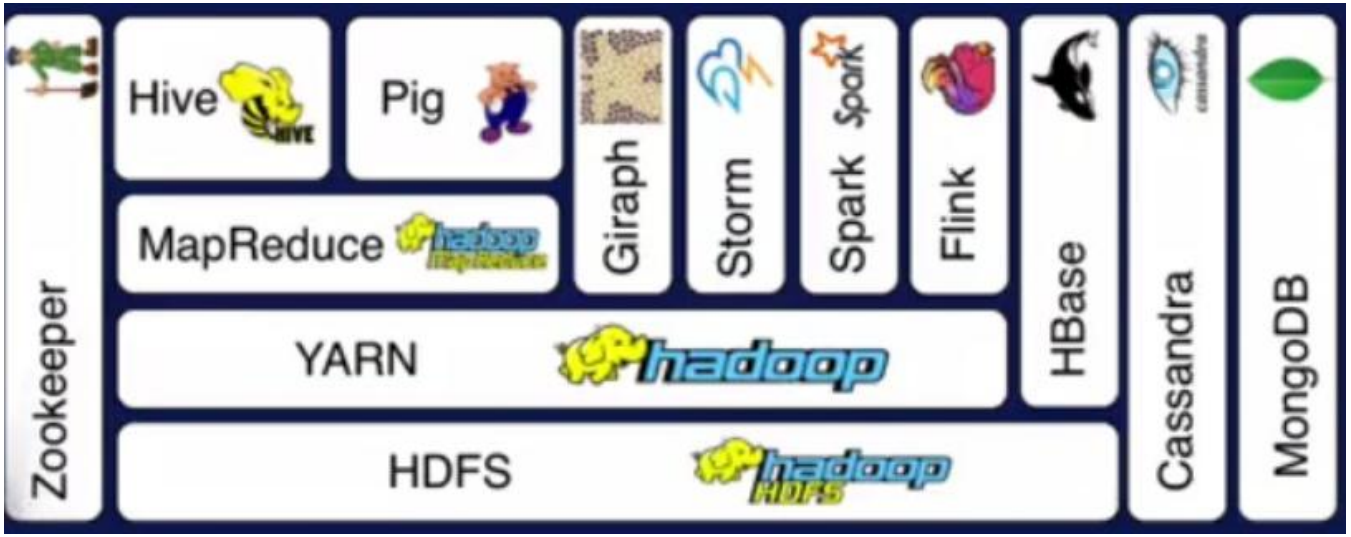
- Hadoop is based on the following principles:
  - **Data locality**: Move computation to where data resides, because it is more efficient than moving large data.
  - **Horizontal scalability**: Scale by adding nodes.
  - **Fault tolerance**: Expect hardware failures.
  - **Batch-oriented processing**: jobs are collected, placed in a queue for processing.
  - **Write-once, read-many** data access pattern.
  - **Resource sharing**: which is the motivation behind distributed Systems.
  - **File System**: which is responsible for the organization, storage, retrieval, naming, sharing, and protection of files.

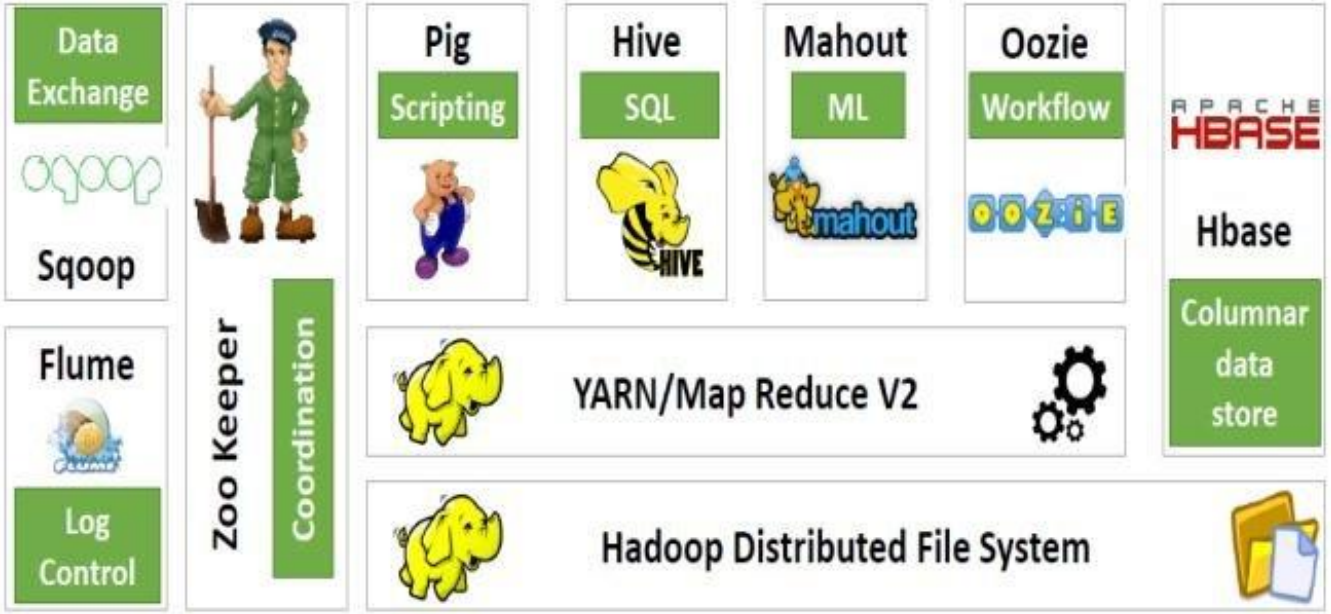
Hadoop Logo

Hadoop Architecture Overview

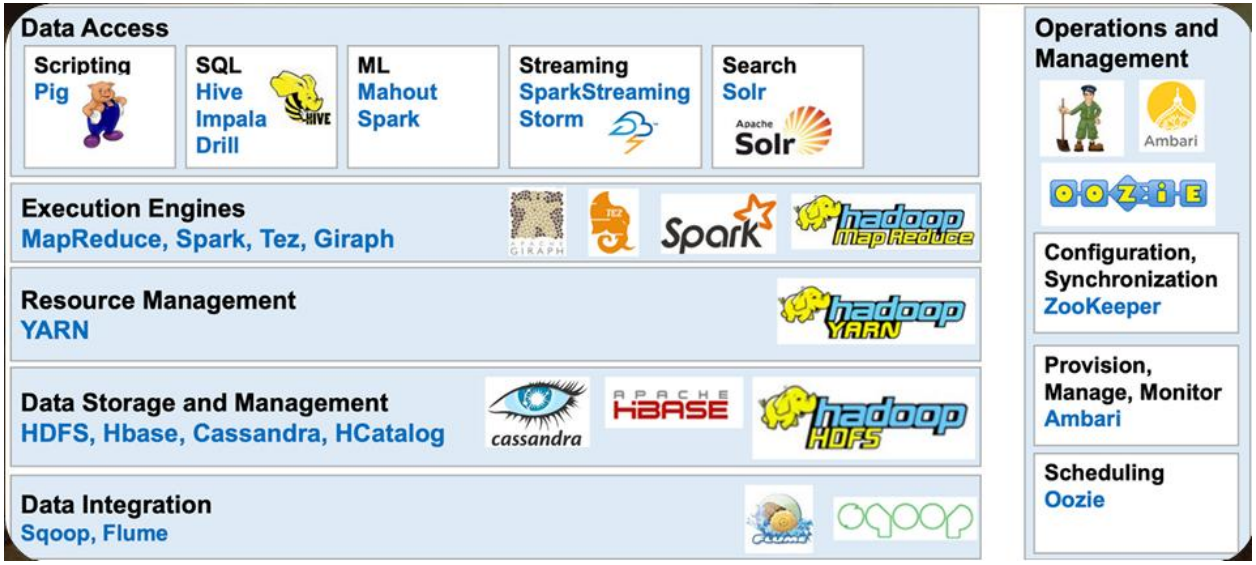
- Apache Hadoop includes :
  - **HDFS**: a distributed filesystem for Big Data **storage.**
  - **YARN**: for Resource Management.
  - **Map/Reduce**: a computing engine implementing programming model for Big Data **processing.**
  - **Data Access and Analytics Tools**: Spark, Pig, Mahout, Tez, Giraph, Flink, DataMeer, Alpine Miner, …
  - **NoSQL and Storage Extensions**: Hive, Hbase, Cassandra, MongoDB, Impala, drill, Hcatalog, …
  - **Data Ingestion and Integration Tools**: Sqoop, Flume, Kafka, Storm, Solr…
  - **Workflow, Coordination, and Management**: Oozie, ZooKeeper, Ambari…
  - **Hue**: The Hadoop desktop GUI, allows launching Data Browsers and Query Tools.
  - **Security**: Ranger, Knox, …

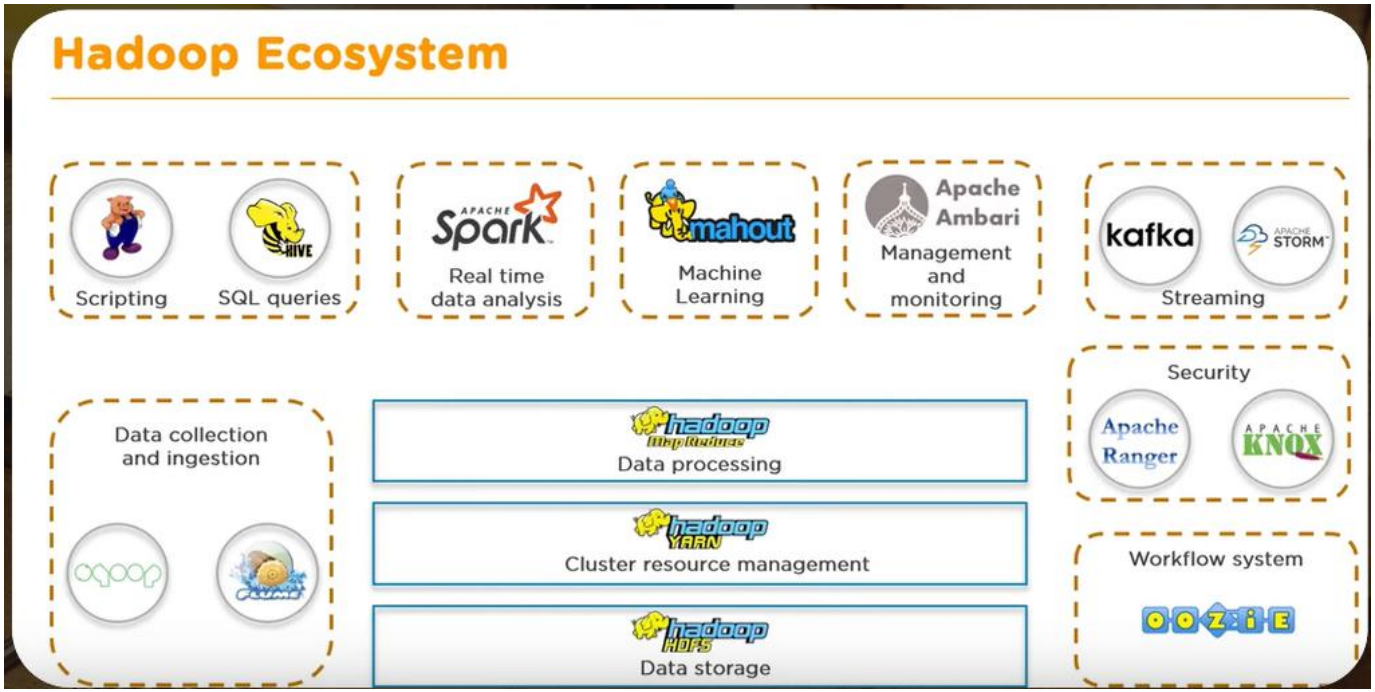
Hadoop Architecture Overview





Hadoop Architecture Overview





Advantages

- Supports use of inexpensive, commodity hardware
- No RAID needed. Also, the servers need not be the latest and greatest hardware.
- Provides for simple, massive parallelism
- Provides resilience by replicating data and eliminating tape backups
- Provides locality of execution, as it knows where the data is
- Software free
- High quality support available at modest cost
- High quality training available at modest cost
- Certification available
- Easy to support when using GUI such as Cloudera Manager or Ambari
- Add-on tools available at relatively low cost, or in some cases no cost
- Evolving technology with a high degree of interest around the world

Use cases

- Yahoo and Facebook are two well known companies that use Hadoop.
- Yahoo has over 42,000 nodes.
- Largest amount of disk storage in use: Facebook over 100 Petabytes, Growing at over ½ Petabyte per day.

Security in Hadoop Ecosystem

- **Authentication**
  - Kerberos-based authentication
  - Secure cluster access
- **Authorization**
  - Apache Ranger / Apache Sentry
  - Fine-grained access control
- **Data Protection**
  - HDFS encryption at rest
  - TLS for data in transit
  - Audit logging

Hadoop Deployment Models

- **On-Premises Clusters**
- **Cloud-based Hadoop** (Amazon EMR, Azure HDInsight)
- **Hybrid architectures**

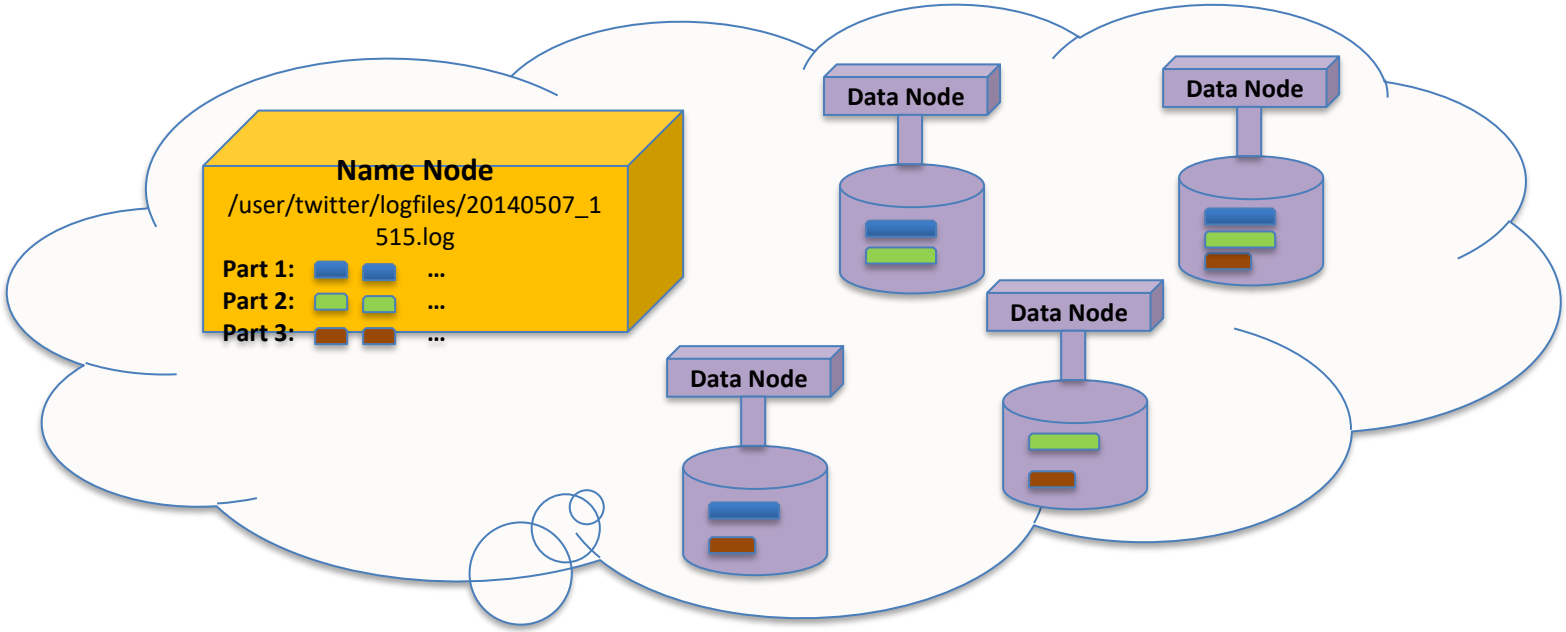
Hadoop distributed file system (HDFS)

Introduction

- The **Hadoop Distributed File System (HDFS)** is a scalable, fault-tolerant distributed storage system designed to store **very large datasets** (terabytes to petabytes) across clusters of commodity hardware.
- HDFS is optimized for **high-throughput batch access** rather than low-latency data access.
- HDFS is a core component of the Hadoop ecosystem and provides the storage foundation for higher-level processing frameworks such as **MapReduce**, **Spark**, **Hive**, and **HBase**.

HDFS

- The key concept is to split the data and store it across collection of machines known as a cluster.
- Then the data is processed where it is actually stored.

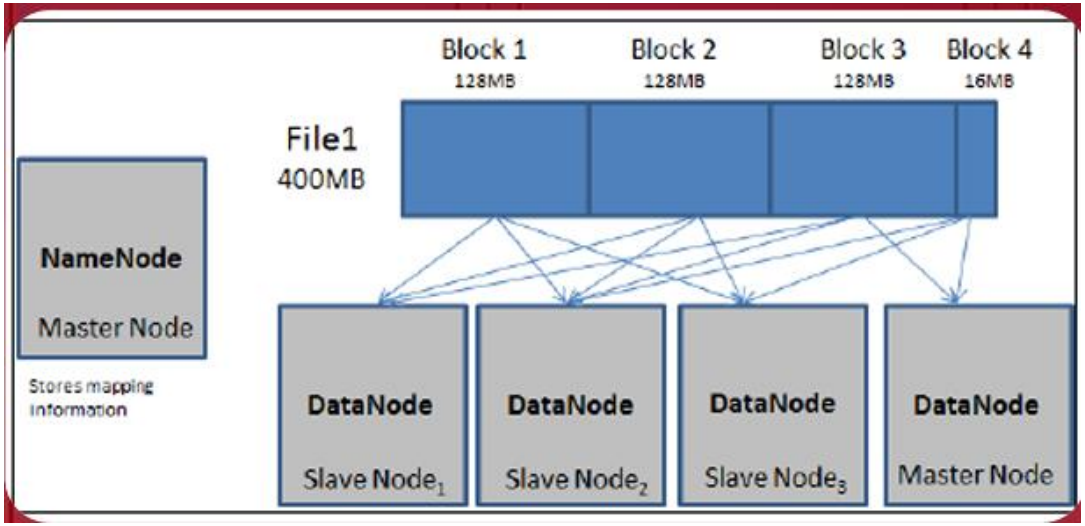


HDFS

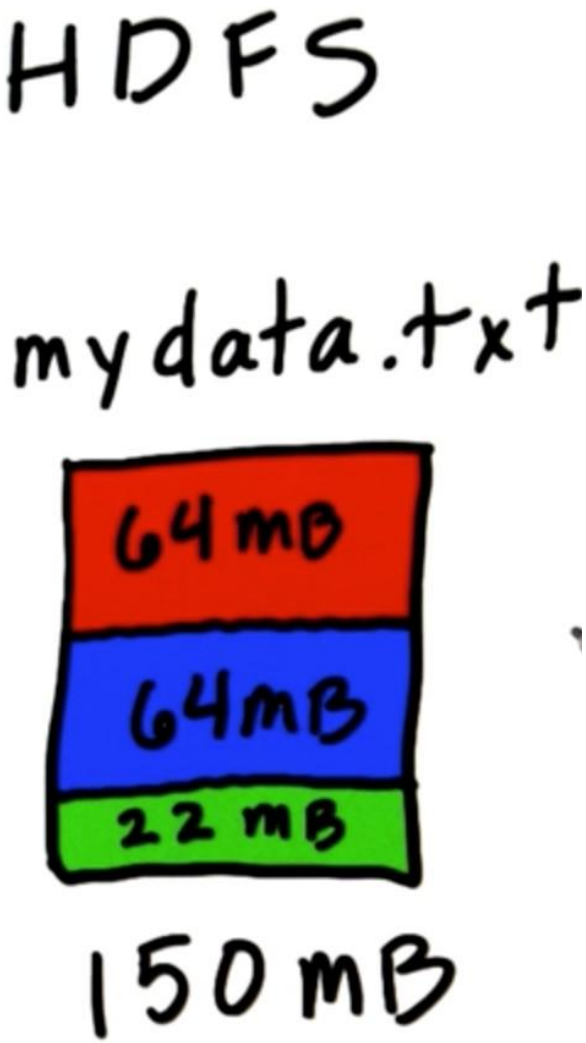
- To store data, Hadoop utilizes its own distributed filesystem, HDFS, which makes data available to multiple computing **nodes**.
- A distributed file system is a file system that can store large files spread across the nodes of a cluster.
- HDFS works by breaking large files into smaller pieces called **blocks**.
- When data is stored in Hadoop, the **NameNode** file automatically stores and replicates the data (3 times) in multiple blocks (64 MB or 128 MB by default) across the various **DataNode**.
- This is done to ensure fault tolerance and high availability.

HDFS

- The master node (aka **NameNode**) stores all the metadata, access rights, mapping and location of files and blocks, and so on.
- The slaves (aka **DataNode**) are nodes where the actual data is stored.
- All the requests go to the master and then are handled by the appropriate slave node.

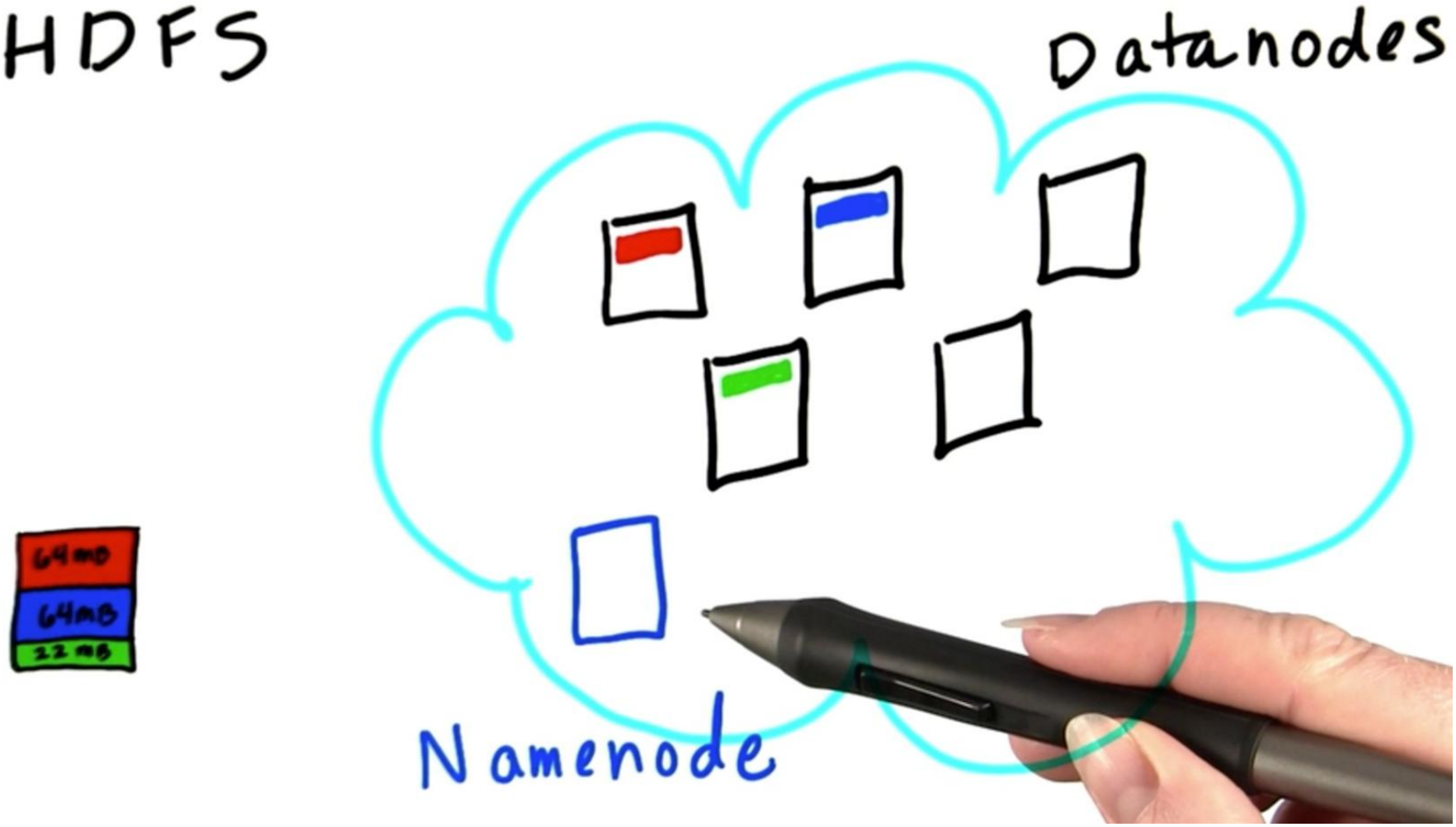


HDFS



HDFS

HDFS

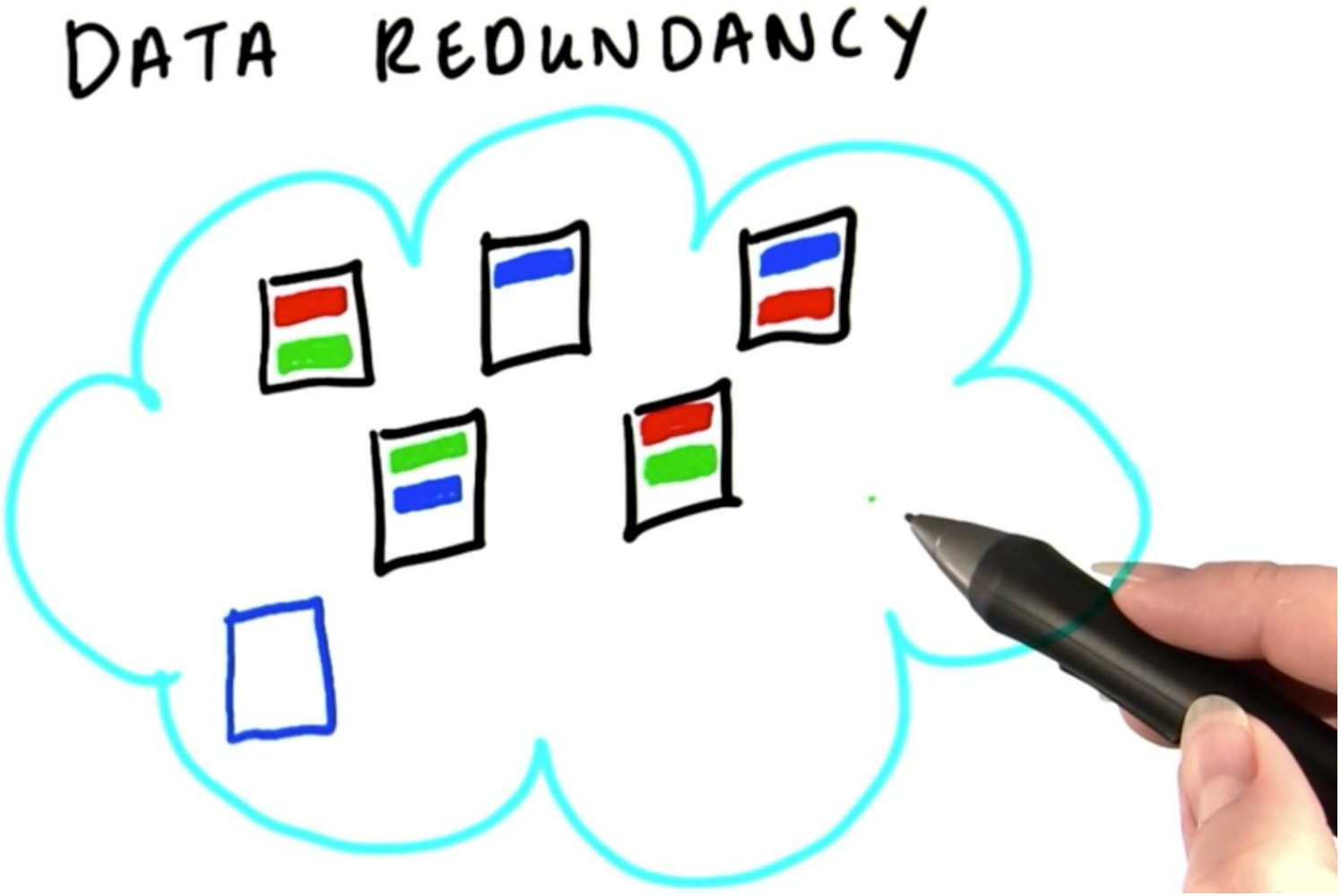


HDFS

IS THERE A PROBLEM?

- [x] NETWORK FAILURE
- [x] DISK FAILURE ON DN
- [ ] NOT ALL DN USED
- [ ] BLOCK SIZES DIFFER
- [x] DISK FAILURE ON NN

HDFS



HDFS

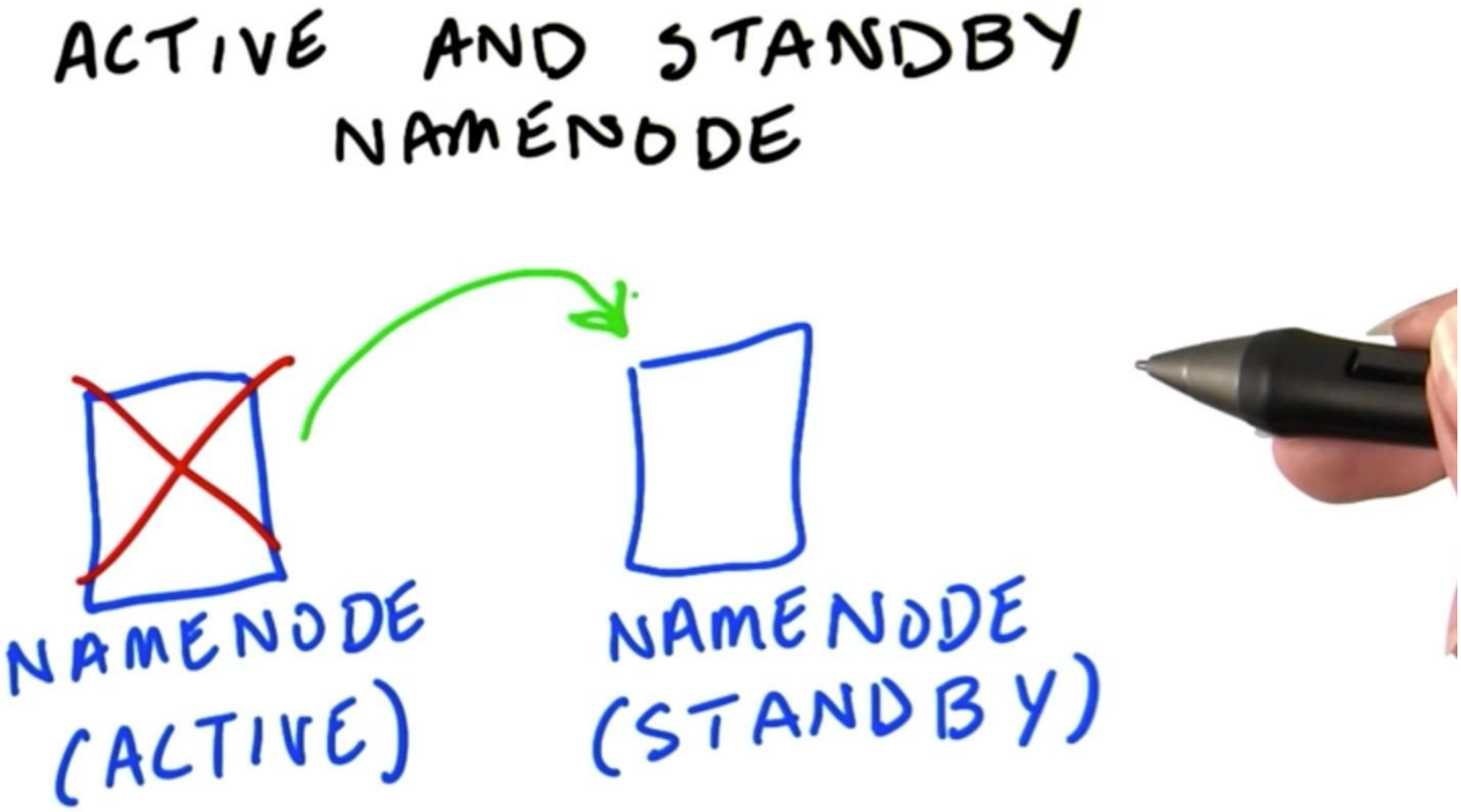
ANY PROBLEMS NOW?

- [x] DATA INACCESSIBLE
- [x] DATA LOST FOREVER
- [ ] NO PROBLEM

HDFS

NAMENODE HIGH AVAILABILITY

HDFS


ACTIVE AND STANDBY
NAMENODE
NAMENODE
(ACTIVE)
NAMENODE
(STANDBY)

NameNode

- HDFS metadata is stored in the **NameNode**:
  - When the file was created, accessed, modified, deleted, and so on.
  - Where the blocks of the file are stored in the cluster.
  - Who has the rights to view or modify the file.
  - How many files are stored on the cluster.
  - How many data nodes exist in the cluster.
  - The location of the transaction log for the cluster.

Data nodes

- **Data nodes** are servers that contain the blocks for a given set of files.
- It is reasonable to think of data nodes as "block servers".
  - Stores (and retrieves) the data blocks in the local file system of the server.
  - HDFS is available on many different operating systems and behaves the same whether on Windows, Mac OS, or Linux.
  - Stores the metadata of a block in the local file system based on the metadata template in the NameNode.
  - Sends regular reports to the NameNode about what blocks are available for file operations.

Characteristics

- **Scalable**: New nodes can be added as needed, without changing data formats, or how data is loaded, how jobs are written, or the applications on top.
- **Cost effective**: Hadoop brings massively parallel computing to commodity servers. The result is a sizeable decrease in the cost per terabyte of storage, which in turn makes it affordable to model all your data.
- **Flexible**: Hadoop is schema-less, and can absorb any type of data, structured or not, from any number of sources. Data from multiple sources can be joined and aggregated in arbitrary ways enabling deeper analyses than any one system can provide.
- **Fault tolerant**: When you lose a node, the system redirects work to another location of the data and continues processing without missing a beat

Example

```
File  Edit  View  Search  Terminal  Help
[training@localhost data]$ hadoop fs -mv purchases.txt newname.txt
[training@localhost data]$ hadoop fs -ls
Found 1 items
-rw-r--r--   1 training supergroup  211312924 2013-09-12 21:16 newname.txt
[training@localhost data]$ hadoop fs -rm newname.txt
Deleted newname.txt
[training@localhost data]$ hadoop fs -mkdir myinput
[training@localhost data]$ hadoop fs -put purchases.txt myinput
[training@localhost data]$ hadoop fs -ls
Found 1 items
drwxr-xr-x   - training supergroup          0 2013-09-12 21:16 myinput
[training@localhost data]$ hadoop fs -ls myinput
Found 1 items
-rw-r--r--   1 training supergroup  211312924 2013-09-12 21:16 myinput/purchases.txt
[training@localhost data]$
```

Example

```
File  Edit  View  Search  Terminal  Help
[training@localhost data]$ ls
access_log.gz  purchases.txt
[training@localhost data]$ hadoop fs -ls
[training@localhost data]$ hadoop fs -put purchases.txt
[training@localhost data]$ hadoop fs -ls
Found 1 items
-rw-r--r--   1 training supergroup  211312924 2013-09-12 21:16 purchases.txt
[training@localhost data]$ hadoop fs -tail purchases.txt
31      17:59   Norfolk Toys    164.34  MasterCard
2012-12-31      17:59   Chula Vista     Music   380.67  Visa
2012-12-31      17:59   Hialeah Toys    115.21  MasterCard
2012-12-31      17:59   Indianapolis    Men's Clothing  158.28  MasterCard
2012-12-31      17:59   Norfolk Garden  414.09  MasterCard
2012-12-31      17:59   Baltimore       DVDs    467.3   Visa
2012-12-31      17:59   Santa Ana       Video Games     144.73  Visa
2012-12-31      17:59   Gilbert Consumer Electronics    354.66  Discover
2012-12-31      17:59   Memphis Sporting Goods  124.79  Amex
2012-12-31      17:59   Chicago Men's Clothing  386.54  MasterCard
2012-12-31      17:59   Birmingham      CDs     118.04  Cash
2012-12-31      17:59   Las Vegas       Health and Beauty       420.46  Amex
2012-12-31      17:59   Wichita Toys    383.9   Cash
2012-12-31      17:59   Tucson  Pet Supplies    268.39  MasterCard
2012-12-31      17:59   Glendale        Women's Clothing        68.05   Amex
2012-12-31      17:59   Albuquerque     Toys    345.7   MasterCard
2012-12-31      17:59   Rochester       DVDs    399.57  Amex
2012-12-31      17:59   Greensboro      Baby    277.27  Discover
2012-12-31      17:59   Arlington       Women's Clothing        134.95  MasterCard
2012-12-31      17:59   Corpus Christi  DVDs    441.61  Discover
[training@localhost data]$
```

MapReduce

Hadoop ecosystem tools

Thank you