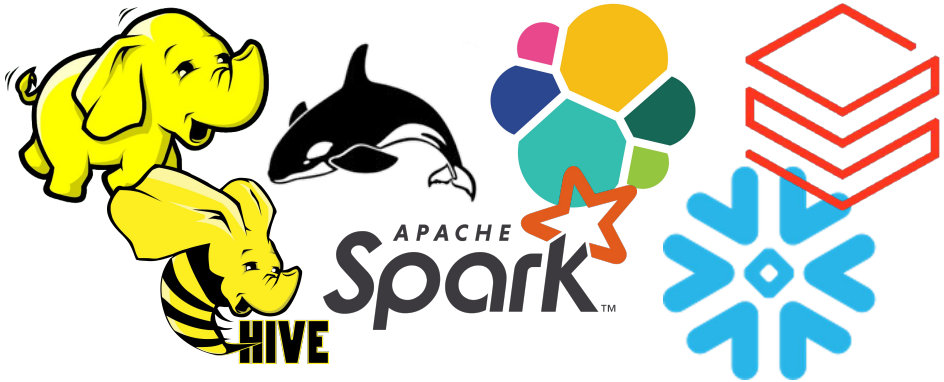
ADALTAS

Big Data Ecosystem



2. Hadoop core: HDFS + YARN

Reminder: the Hadoop Ecosystem

- Open Source
- Java -> JVM
- Linux

Reminder: the Hadoop Ecosystem

ADALTAS

- Distributed Filesystem: HDFS
- Cluster Manager: YARN
- Execution Engines: MapReduce, Tez, Spark
- Warehouse /SQL: Hive
- NoSQL DB: HBase
- And other stuff

Apache Hadoop core

ADALTAS

- **HDFS**: Hadoop Distributed File System
- **YARN**: Yet Another Resource Negotiator
- **MapReduce**: Big Data applications framework

What is HDFS?

ADALTAS

The Hadoop Distributed File System:

- Data stored on hundred/thousands of nodes
- Data is **replicated**: avoid data loss + optimize access
- Support **huge files**: typical file size from GB to TB
- Focus on **high throughput** vs low latency
- **Unix-like** file system (tree + rwx permissions)

Master / Slave architecture

ADALTAS

- Master component coordinates workers
- Slave components do the jobs

→ As opposed to ?

HDFS: File storage

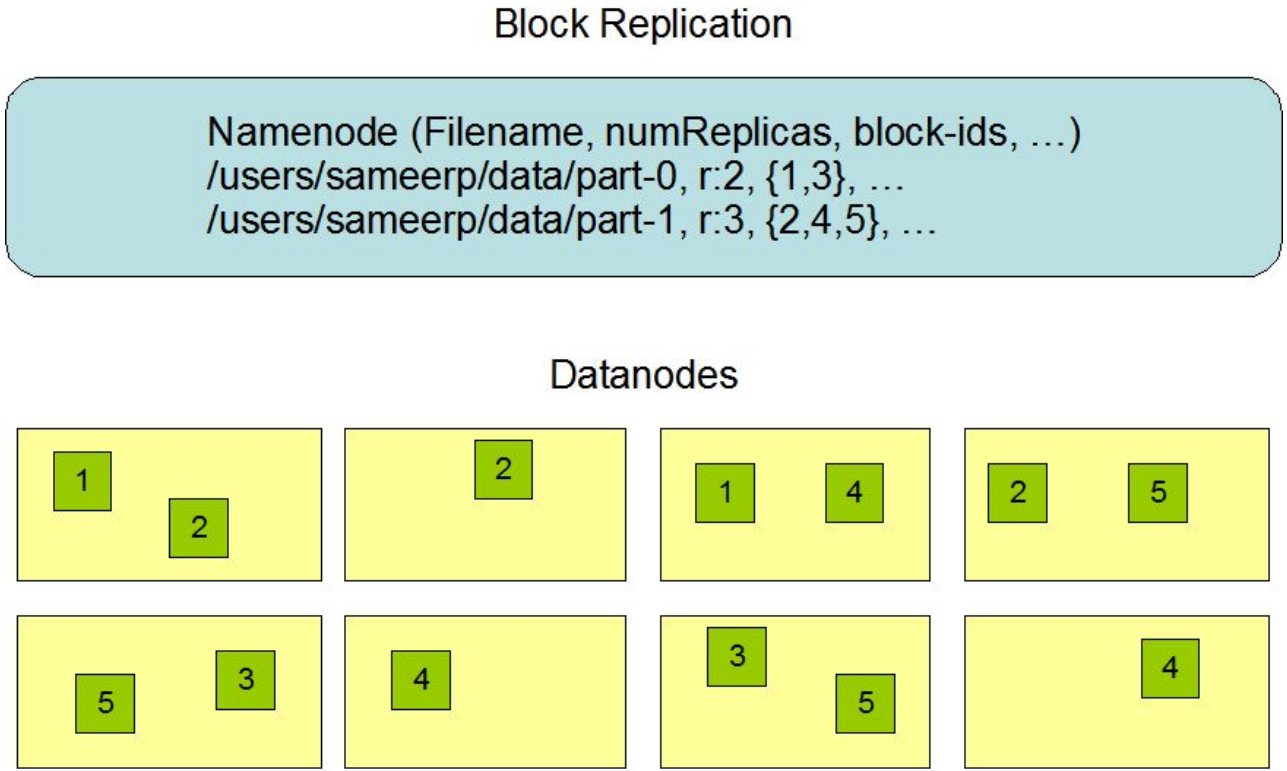
- One file is divided into **blocks**
  - 1 block = **128 MB** (max)
  - Each block is **replicated** (x3 by default)
- E.g. 1 file of 400 MB:
  - 3 blocks of 128 MB + 1 block of 16 MB
  - The file actually takes 1.2 GB of disk in the cluster

HDFS: Architecture

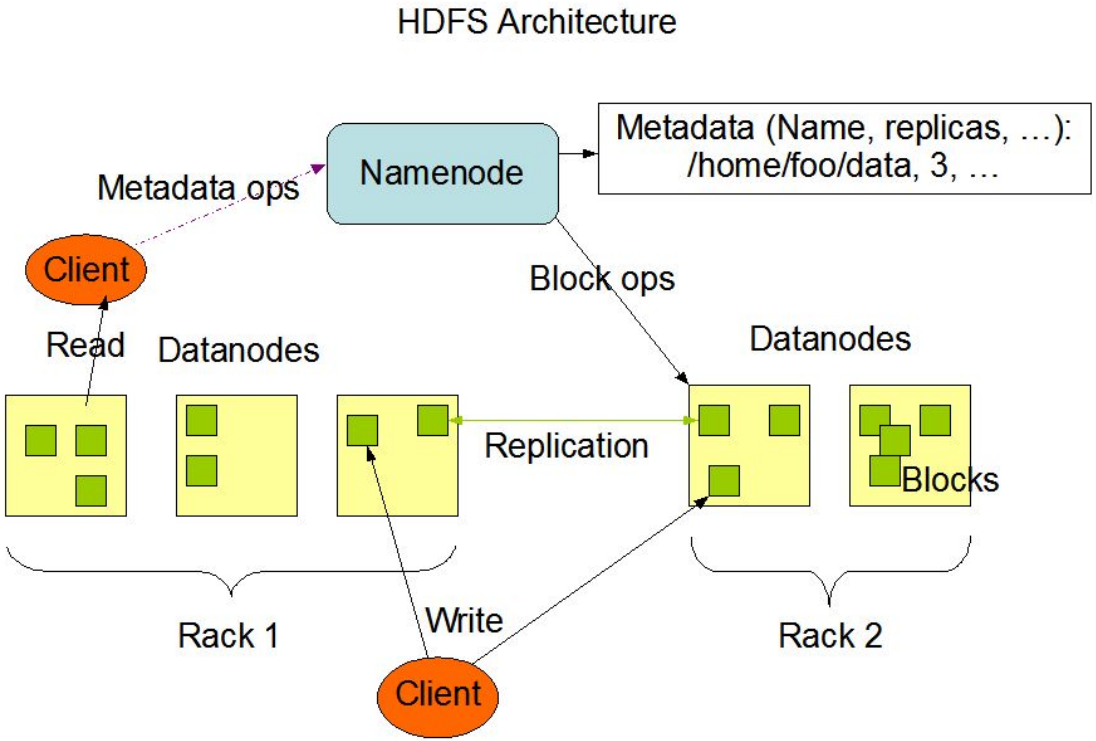
ADALTAS

- **NameNode** (= master): For each file, it knows
  - The blocks making the file
  - The position of each block (on which DN)
- **DataNodes** (= workers):
  - Store the blocks on hardware
  - Read/write operations

HDFS: Data replication example



HDFS: Client interactions



HDFS: Important properties

ADALTAS

- **WORM** = Write Once Read Many: no update on files
- Rack awareness
- NameNode stores everything in RAM: **small files problem**
- **Secondary NameNode**:
  - Builds NameNode checkpoints (= FSImage)
  - To go deeper article

HDFS: Sum up

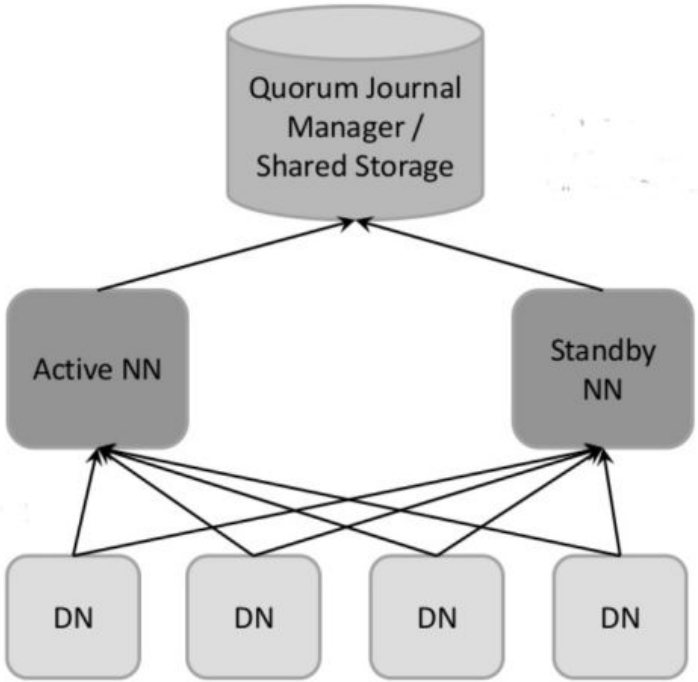
ADALTAS

- Components:
  - **NameNode**: tracks blocks/files
  - **DataNode**: stores blocks + read/write operations
  - **Secondary NameNode**: builds checkpoints based on edit logs
- What is wrong? **SPOF**

High availability in distributed systems

ADALTAS

HDFS: High Availability mode



HDFS HA: Sum up

ADALTAS

- Components:
  - **NameNodes**: 1 active NN + 1 standby NN
  - **DataNodes**: store blocks + read/write operations
  - **JournalNodes**: keep track of every action

What is YARN?

ADALTAS

Yet Another Resource Negotiator:

- Cluster resource manager:
  - Handles the **RAM** and the **CPU** of workers
  - Allocates resources to applications
- Job monitoring

YARN: Architecture

ADALTAS

- **ResourceManager** (= master):
  - Schedule applications based on available resources
  - Gather information about running applications
- **NodeManager** (= worker): Handle resources on one worker

YARN: Architecture schema

ADALTAS

YARN: Applications

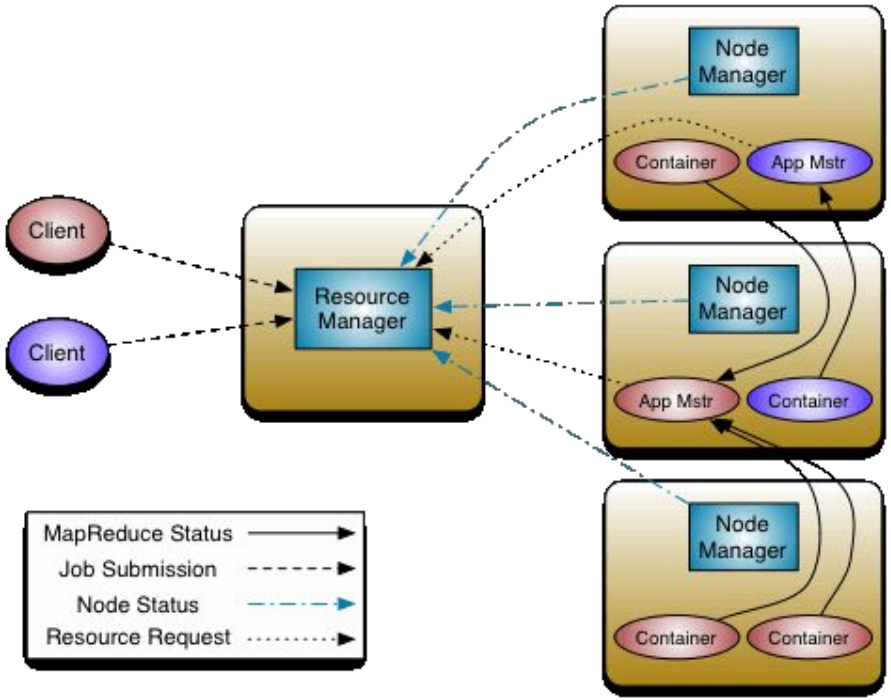
ADALTAS

Application = single job or DAG of jobs

Components: JVMs

- **ApplicationMaster**: First container to be allocated, it requests resources for other containers and monitor
- **Containers**: They do the computing

YARN: Application lifecycle



YARN: Resource sharing

ADALTAS

- Queues
  - ECE - 70% 100%
    - APP - 50% 100%
    - INI - 50% 100%
  - Adaltas - 30% 50%
- Fair scheduling
- The peak problem resolved