

# TỔNG QUAN VỀ HADOOP VÀ HADOOP ECOSYSTEM

# Nội dung chính

1. Tại sao sử dụng Hadoop?
2. Tổng quan các thành phần Hadoop
3. Các thành phần trong hệ sinh thái Hadoop
4. Tổng quan về HDFS

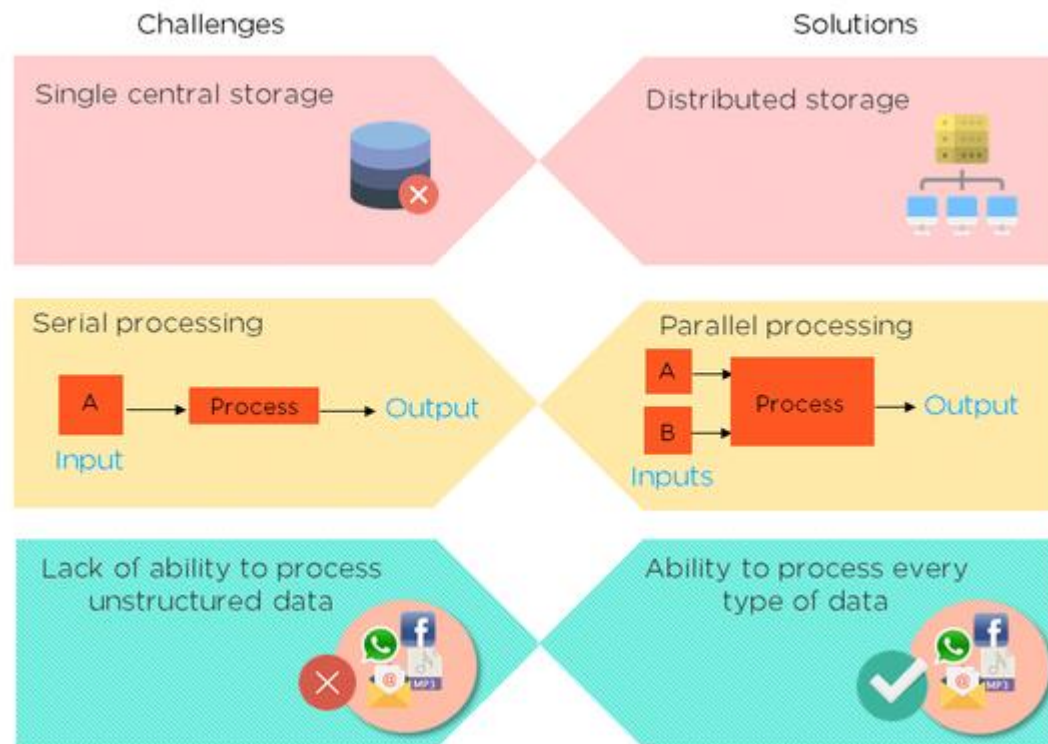
# 1. Tại sao sử dụng Hadoop?

- Sự phát triển của Big Data ngày càng lớn yêu cầu giải pháp lưu trữ, xử lý mới.
- Dữ liệu lớn là tập dữ liệu quá lớn và phức tạp để xử lý bởi cơ sở dữ liệu thông thường.
- Các nguồn sinh dữ liệu như: IOT, Mạng xã hội, dịch vụ kinh doanh, giáo dục,...



# Giải pháp Hadoop cung cấp

- Hadoop lưu trữ dữ liệu phân tán
- Hadoop cung cấp Map-Reduce hỗ trợ xử lý song song
- Hệ sinh thái Hadoop xử lý các loại dữ liệu đa dạng



## 2. Tổng quan các thành phần Hadoop

Hadoop là framework lưu trữ và xử lý phân tán cho dữ liệu lớn

- Map Reduce: mô hình xử lý dữ liệu lớn phân tán
- HDFS: tầng lưu dữ liệu phân tán
- YARN: tầng quản lý tài nguyên tính toán
- Common Utilities: các gói thư viện tiện ích phục vụ Hadoop



Map Reduce

HDFS

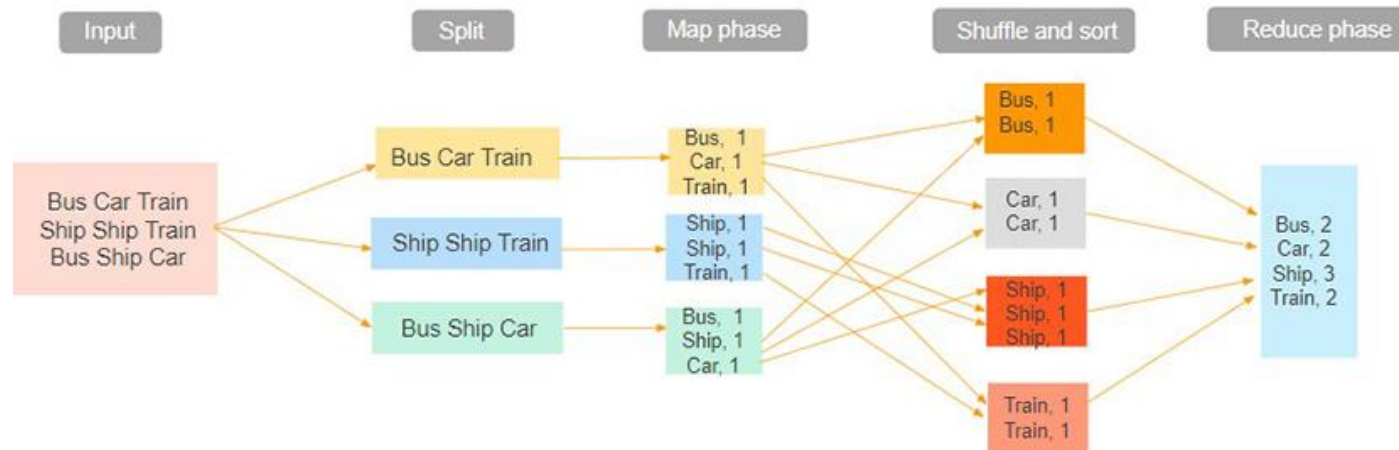
YARN

Common Utilities

## Map Reduce

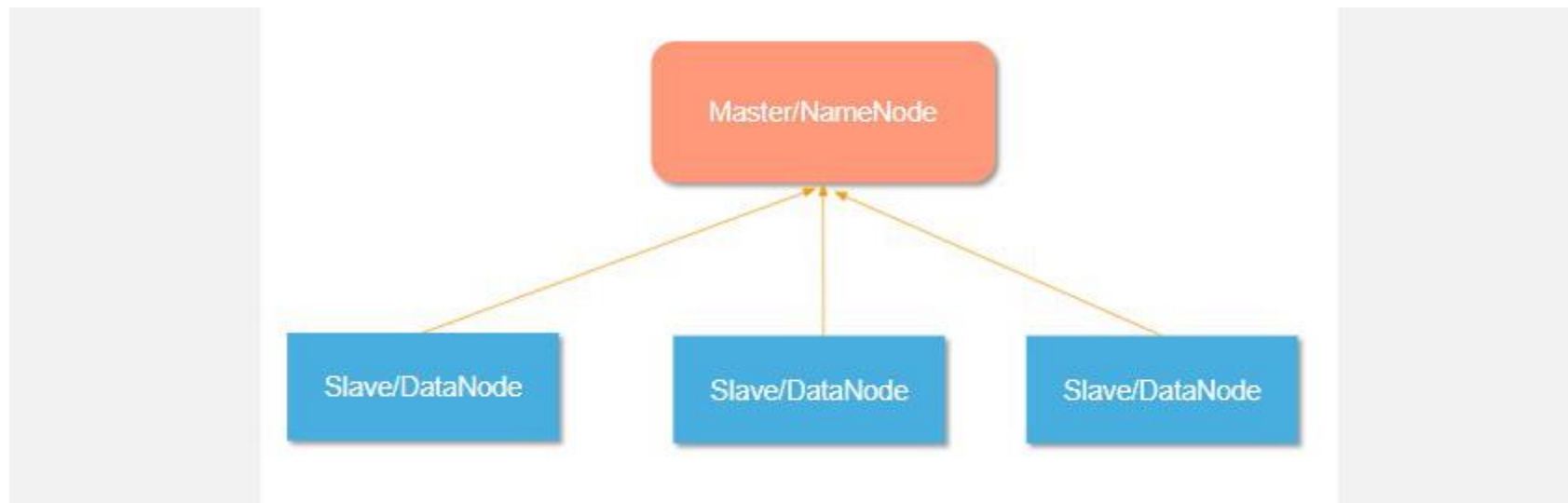
Mô hình Map Reduce chia làm 2 giai đoạn:

- Map: Biến đổi dữ liệu thành giá trị mới theo dạng key-value
- Reduce: Bước tổng hợp dữ liệu theo key



## Hadoop Distributed File System (HDFS)

HDFS là tầng lưu trữ dữ liệu phân tán, hoạt động theo kiến trúc master, slave  
HDFS cung cấp khả năng mở rộng, tính linh hoạt và khả năng chịu lỗi

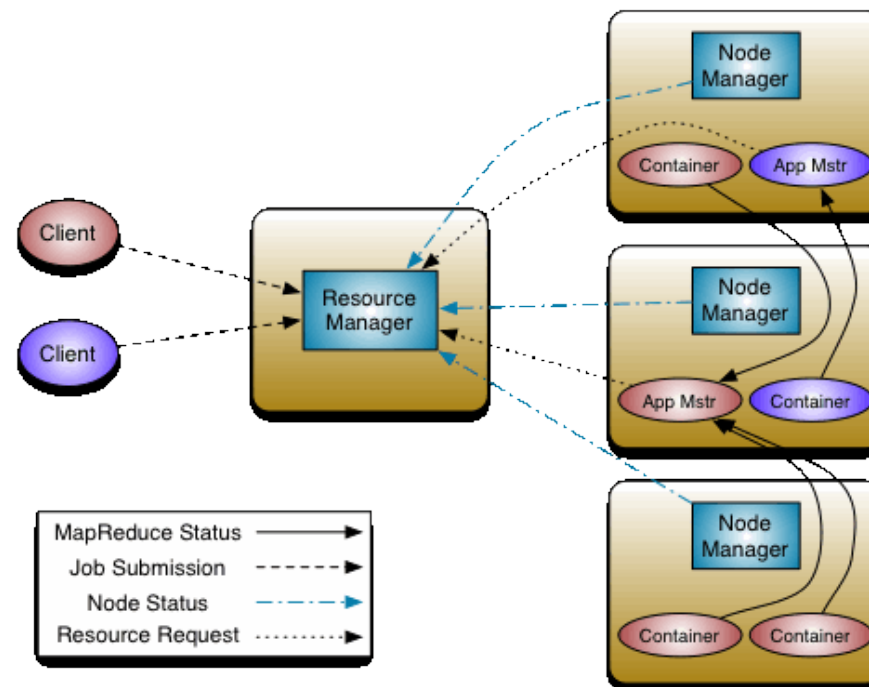


## Yet Another Resource Negotiator (YARN)

YARN xuất hiện ở Hadoop version 2

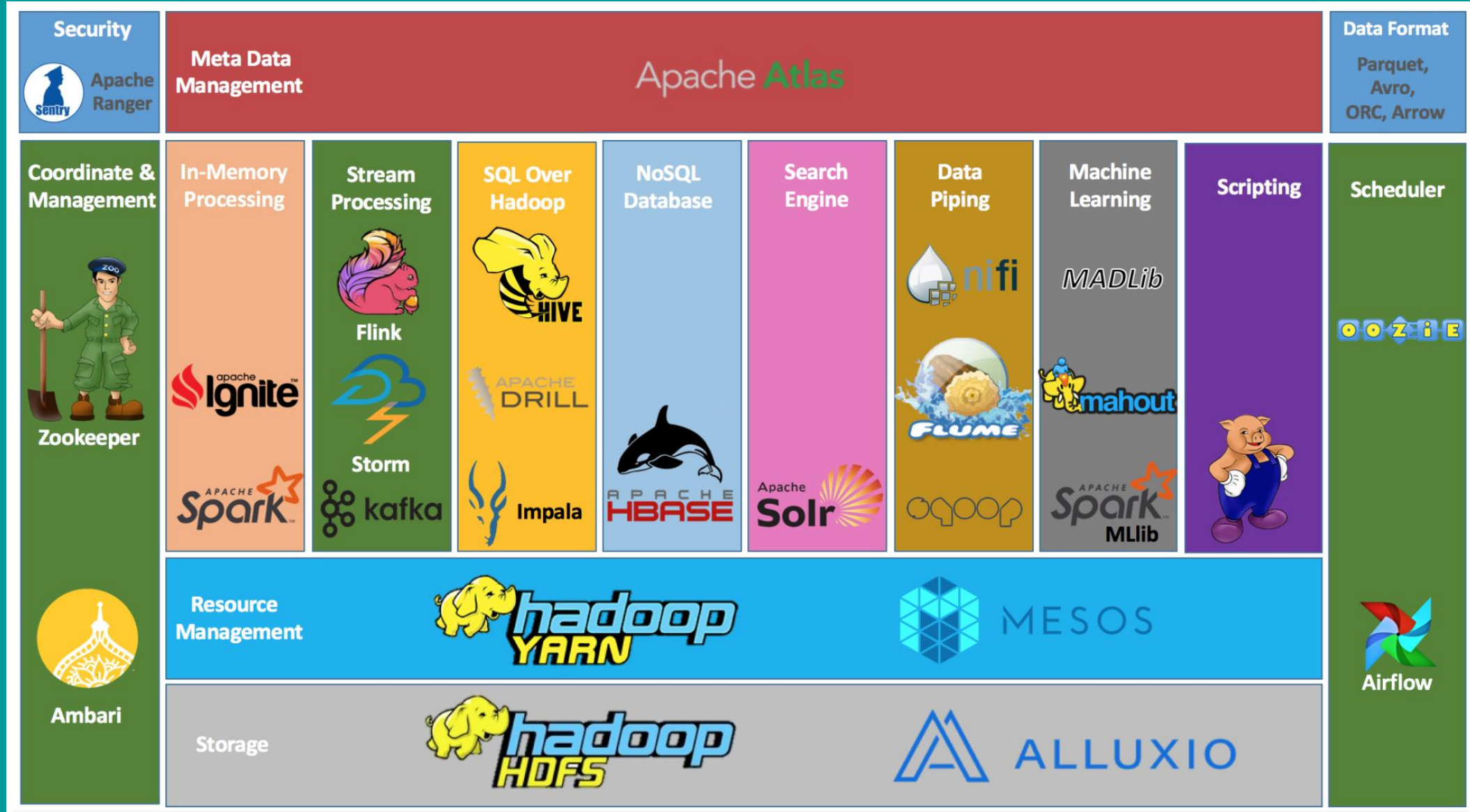
Thực hiện quản lý tài nguyên, điều phối công việc sử dụng tài nguyên

Các thành phần chính: Resource Manager, Node Manager, App Manager và Container

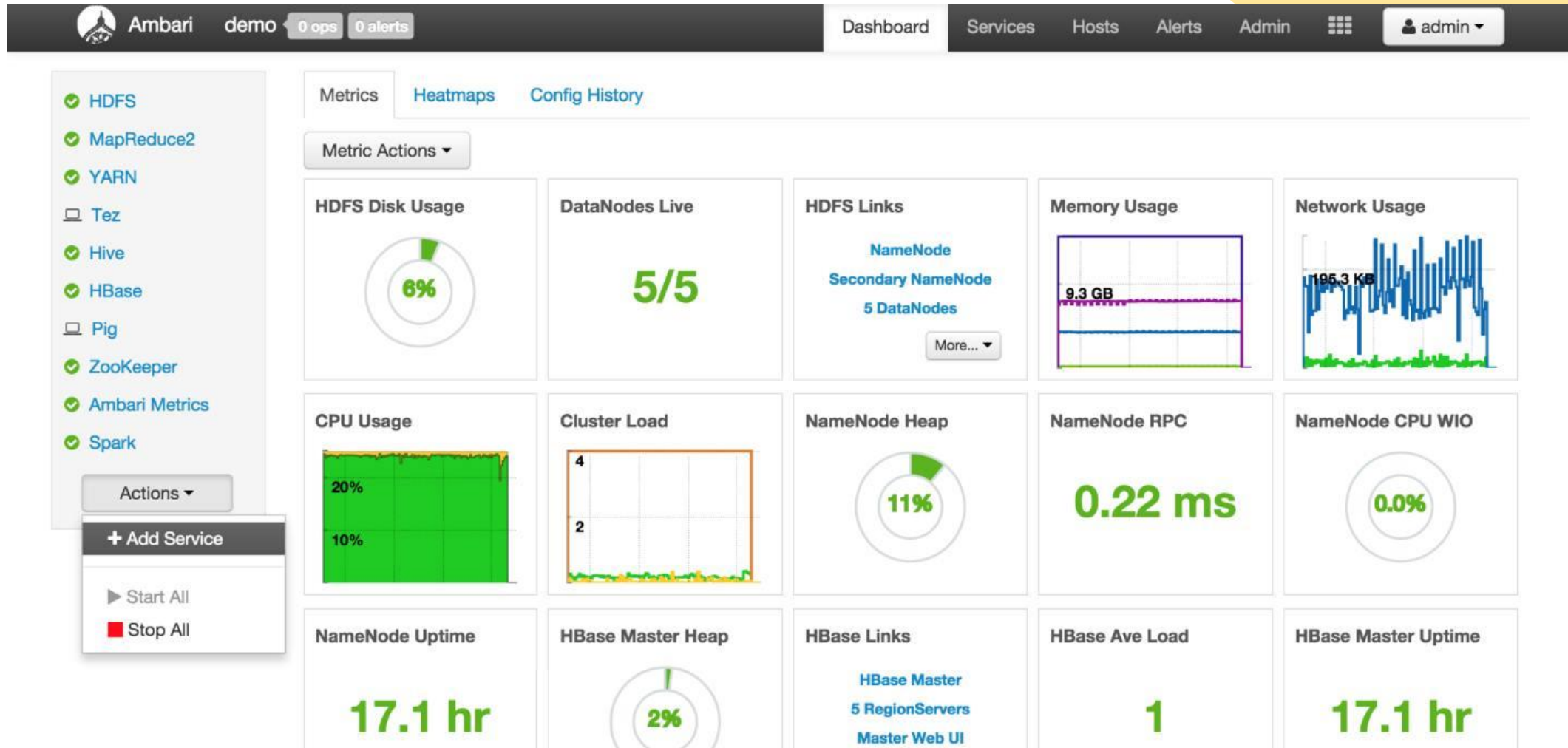




# 3. Các thành phần trong hệ sinh thái Hadoop



# Ambari quản lý cấu hình



# 4. Tổng quan về HDFS

Lịch sử hình thành:

2002: Được bắt đầu với dự án Apache Nutch

2007: Yahoo sử dụng hadoop với 1,000 nodes cluster

2008: Yahoo ra mắt Apache Hadoop như là mã nguồn mở

2011: Apache released Hadoop version 1.0

2012: Apache released Hadoop version 2.0, có chứa Yarn

2017: Apache released Hadoop version 3.0

2020: Apache released Hadoop version 3.2.1

Mike Cafarella and Doug Cutting



# Các đặc điểm của HDFS

## Cost

HDFS sử dụng phần cứng thông dụng rẻ, dễ mở rộng

## Variety and volume

Lưu trữ lượng lớn dữ liệu, có cấu trúc và không cấu trúc.

## Reliability and Fault Tolerance

Dữ liệu được replicas và phân tán trên toàn cụm.

## Data Integrity

HDFS kiểm tra tính toàn vẹn của dữ liệu

## High Throughput

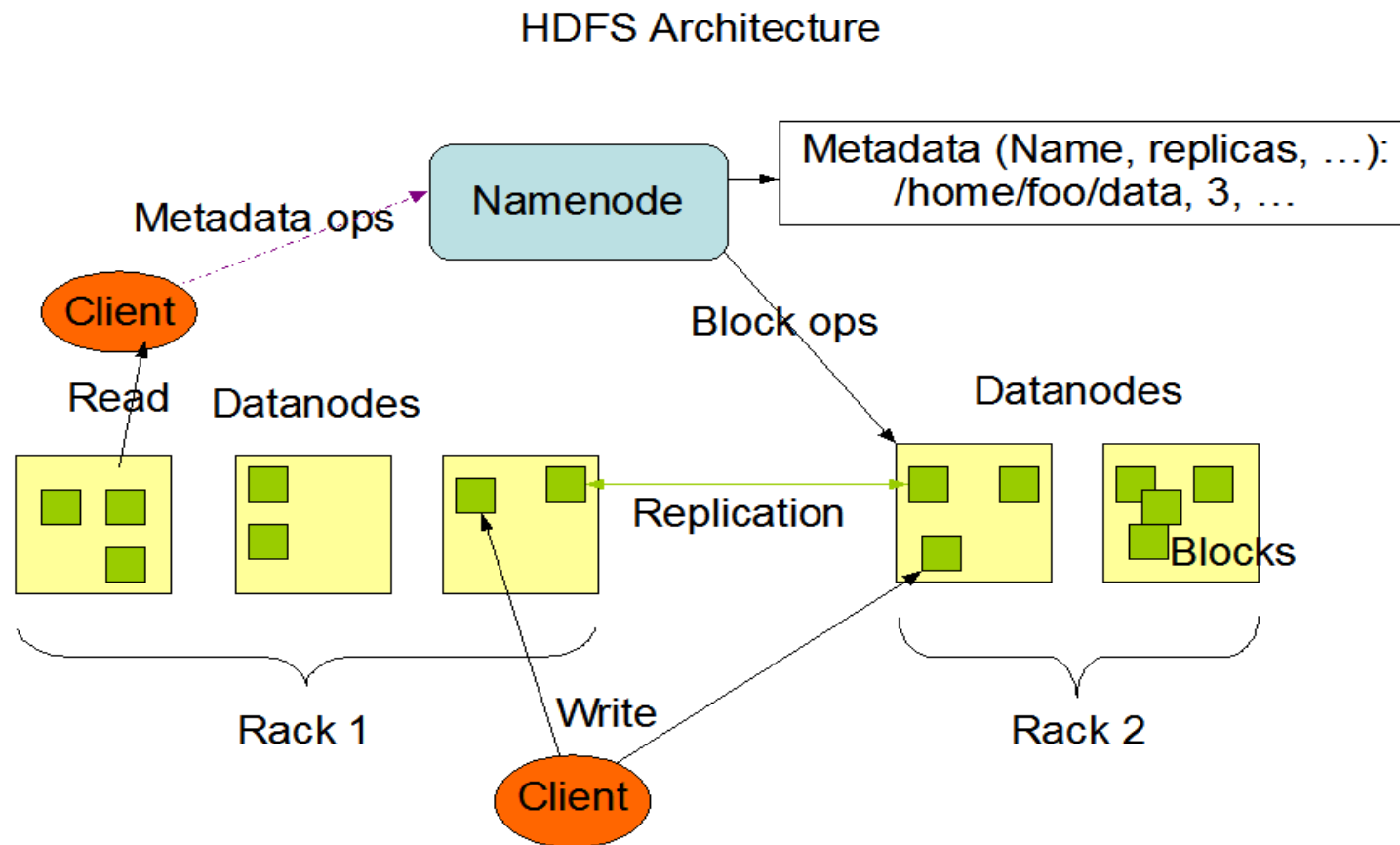
Sử dụng nhiều máy tính xử lý dữ liệu song song nên tốc độ xử lý nhanh

## Data Locality

Mang code xử lý đến chạy trên node chứa dữ liệu.

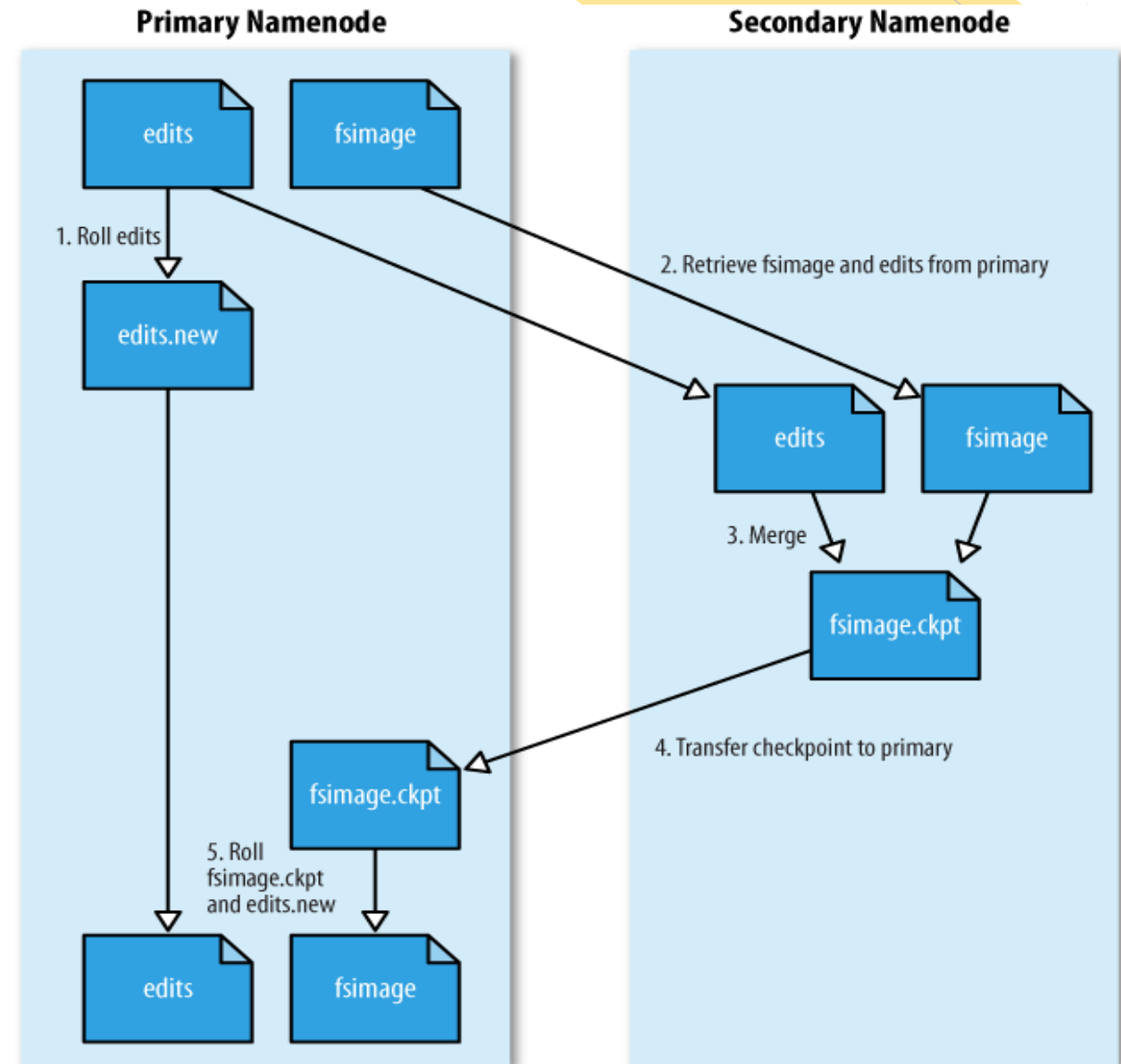
# Kiến trúc HDFS

Kiến trúc sử dụng dạng master/slave



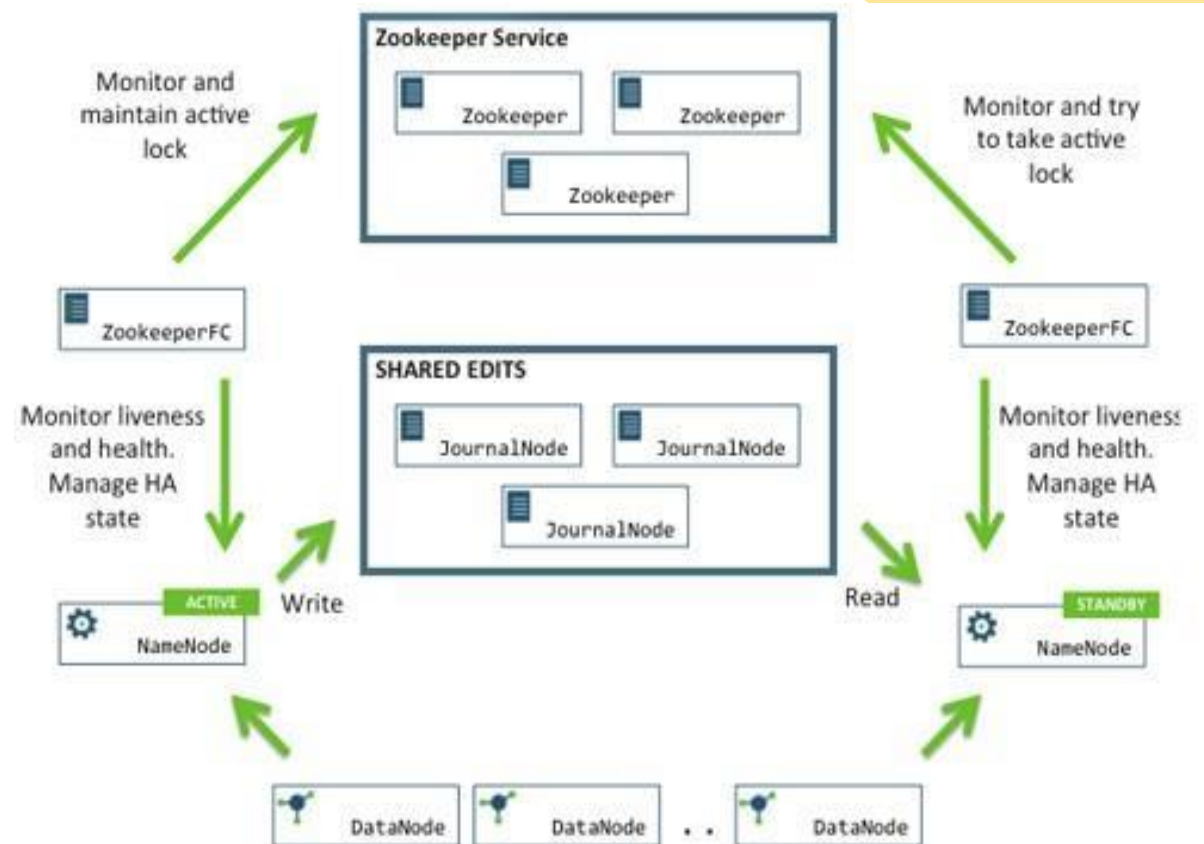
## NameNode

- Secondary NameNode là tiến trình chạy song song với NameNode, không thay thế NameNode.
- Thực hiện checkpointing theo interval để backup dữ liệu, giúp phục hồi NameNode nhanh hơn.
- FSImage: Lưu metadata từ lúc start NameNode.
- EditLog: Lưu thay đổi gần nhất.



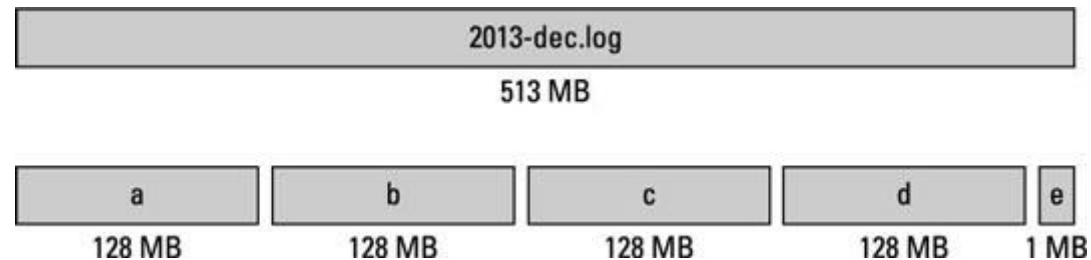
## NameNode High Availability

- Gồm 2 NameNode: Active và Standby.
- QJM: Qorum Journal Manager, chia sẻ file editLog.
- DataNode report tới cả 2 NameNode.
- ZKFailover giúp StandBy thành Active khi Active down.



## Blocks

- HDFS lưu file thành các blocks nhỏ, default = 128MB.
- Có thể tùy chỉnh lưu block size = 256MB.
- Tránh việc lưu file với block nhỏ, ảnh hưởng đến NameNode.





## Replication and Rack Awareness




- HDFS replicate các block trên nhiều DataNode, default = 3.
- 1block = 128MB với replication = 3 chiếm 384MB vật lý.
- RackAwareness lựa chọn node gần nhất dựa trên thông tin Rack.
- Rack là tập hợp các DN cùng kết nối mạng (network switch)
- Tăng hiệu năng network trong 1 Rack.
- Tránh việc 1 Rack down ảnh hưởng đến dữ liệu.

### Rack Awareness Algorithm

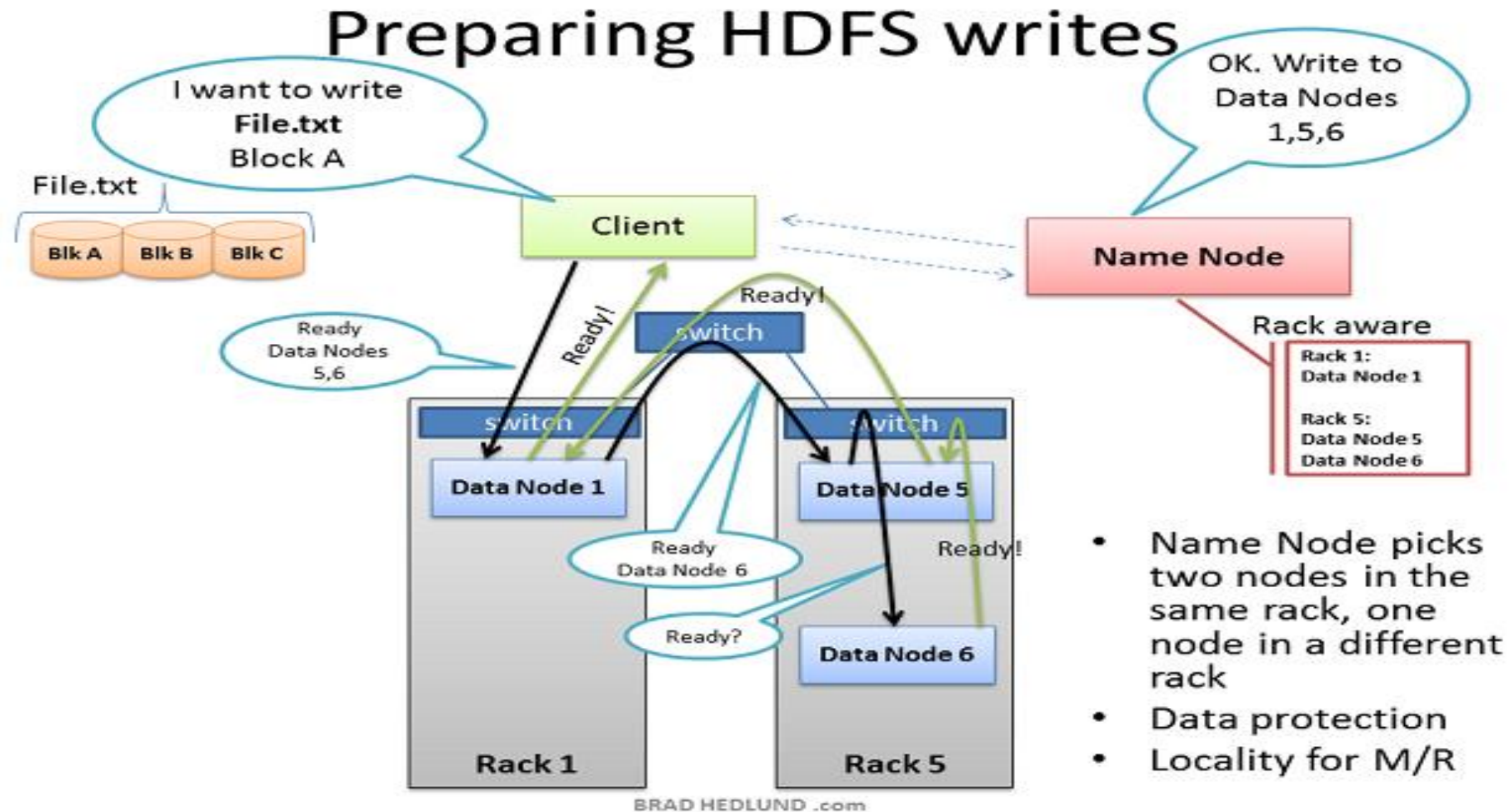
Block A: 

Block B: 

Block C: 

Rack - 1	Rack - 2	Rack - 3
1 	5	9 
2 	6 	10
3	7 	11 
4 	8 	12 

# HDFS Write Architecture





## HDFS Command line

- `Hdfs dfs -ls <path>` : liệt kê danh sách thư mục, file trên hdfs
- `Hdfs dfs -cp <src> <dest>` : copy file từ nguồn sang đích
- `Hdfs dfs -mv <src> <dest>` : di chuyển file từ nguồn sang đích
- `Hdfs dfs -rm -r <path>`: xóa file
- `Hdfs dfsadmin <command>` : các lệnh quản trị cụm

Command	Description
-help	Hiển thị hướng dẫn cách dùng của lệnh
-report	Hiển thị số liệu thống kê và thông tin sức khỏe của datanodes
-metasave	Ghi thông tin về blocks đang được replicated hoặc deleted và danh sách các datanodes đang kết nối ra file trên hadoop.
-safemode	Thay đổi hoặc tìm kiếm thông tin về safemode
-saveNameSpace	Lưu thông tin fsImage trong bộ nhớ hiện tại ra fsImage mới và reset edits file
-fetchImage	Lấy về file fsImage cuối từ NameNode và lưu ra local
-refreshNodes	Cập nhật thông tin các datanodes trong cụm
-upgradeProgress	Lấy thông tin về tiến độ upgrade HDFS.
-finalizeUpgrade	Loại bỏ phiên bản cũ của thư mục lưu trữ NameNode và DataNode. Dùng sau khi upgrade thành công.
-setQuota	Set Quota cho thư mục hdfs, chi tiết xem tại đây.
-clrQuota	Xóa cấu hình quota cho thư mục hiện tại
-setSpaceQuota	Set dung lượng quota cho thư mục, chi tiết tại đây.
-clrSpaceQuota	Xóa bỏ cấu hình space quota cho thư mục,
-refreshServiceAcl	Làm mới chính sách ủy quyền mức dịch vụ của NameNode
-allowSnapshot	Cho phép tạo snapshot trên thư mục
-disallowSnapshot	Không cho phép tạo snapshot cho thư mục

**Thanks for listening**  
**Q&A**