

25 YEARS ANNIVERSARY  
SCHOOL OF INFORMATION AND COMMUNICATION TECHNOLOGY

HA NOI UNIVERSITY OF SCIENCE AND TECHNOLOGY  
SCHOOL OF INFORMATION AND COMMUNICATION TECHNOLOGY



HA NOI UNIVERSITY OF SCIENCE AND TECHNOLOGY  
SCHOOL OF INFORMATION AND COMMUNICATION TECHNOLOGY

# Chương 3

# Hadoop HDFS

# Tổng quan về HDFS

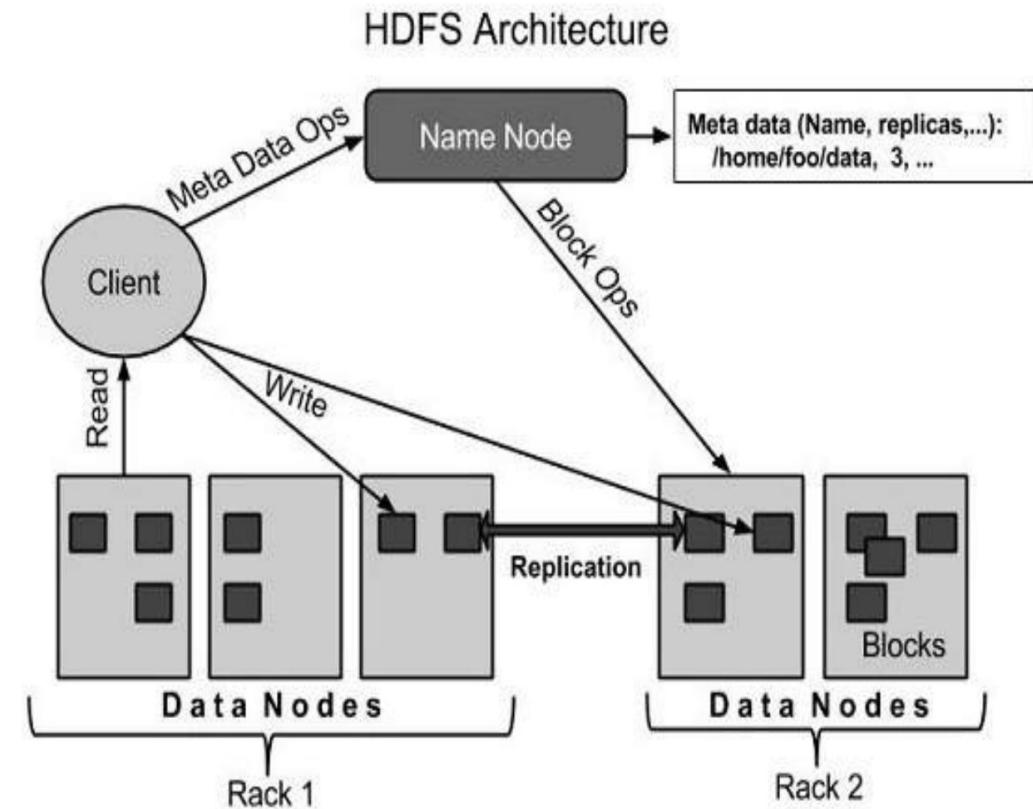
- Cung cấp khả năng lưu trữ đáng tin cậy và giá rẻ cho lượng dữ liệu lớn
- Được thiết kế cho
  - Các tệp lớn (kích thước tệp từ 100 MB đến vài TB)
  - Viết một lần, đọc nhiều lần (Chỉ thêm vào)
  - Chạy trên phần cứng hàng hóa
- Hệ thống tập tin theo phong cách UNIX phân cấp
  - (ví dụ, /hust/soict/hello.txt)
  - Quyền sở hữu và quyền truy cập tệp theo phong cách UNIX

# Nguyên tắc thiết kế chính của HDFS

- Mẫu I/O • Chỉ thêm vào để giảm đồng bộ hóa
- Phân phối dữ liệu
  - Tệp được chia thành nhiều phần lớn (64 MB) à giảm kích thước siêu dữ liệu à giảm giao tiếp mạng
- Sao chép dữ liệu • Mỗi khối thường được sao chép trong 3 nút khác nhau
- Khả năng chịu lỗi
  - Nút dữ liệu: sao chép lại • Nút tên
    - Namenode thứ cấp • Namenode dự phòng, hoạt động

# Kiến trúc HDFS

- Kiến trúc chủ/tớ
- Chủ HDFS: Namenode
  - Quản lý không gian tên và siêu dữ liệu
  - Giám sát Datanode
- HDFS slaves: Datanodes
  - Xử lý đọc/ghi dữ liệu thực tế {chunks}
  - Các khối là các tập tin cục bộ trong hệ thống tập tin cục bộ



# Chức năng của Namenode

- Quản lý Không gian tên Hệ thống tệp • Ánh xạ tên tệp vào một tập hợp các khối
  - Ánh xạ khối vào các Datanode nơi nó lưu trú
- Quản lý cấu hình cụm • Công cụ sao chép cho các khối

# Siêu dữ liệu Namenode

- Siêu dữ liệu trong bộ nhớ •

Toàn bộ siêu dữ liệu nằm trong bộ nhớ chính •

Không phân trang theo yêu cầu của siêu

- dữ liệu • Các loại siêu dữ liệu

- Danh sách các

- tệp • Danh sách các khối cho mỗi

- tệp • Danh sách các nút dữ liệu cho mỗi

- khối • Thuộc tính tệp, ví dụ thời gian tạo, hệ số sao chép • Nhật

- ký giao dịch • Ghi lại việc

- tạo tệp, xóa tệp, v.v.

# Nút dữ liệu

- Máy chủ khôi

- Lưu trữ dữ liệu trong hệ thống tệp cục bộ (ví dụ ext3) •

- Lưu trữ siêu dữ liệu của một khôi (ví dụ CRC) •

- Phục vụ dữ liệu và siêu dữ liệu cho Máy khách

- Báo cáo khôi •

- Định kỳ gửi báo cáo về tất cả các khôi hiện có tới  
Nút tên

- Tạo điều kiện thuận lợi cho việc truyền dữ

- liệu • Chuyển tiếp dữ liệu đến các Datanode được chỉ định khác

- Nhịp tim

- Datanode gửi nhịp tim đến Namenode

- Cứ 3 giây một lần

- Namenode sử dụng nhịp tim để phát hiện lỗi Datanode

# Sao chép dữ liệu

- Vị trí khôi phục • Chiến lược
  - hiện tại • Một bản sao trên nút cục bộ • Bản sao thứ hai trên giá đỡ từ xa • Bản sao thứ ba trên cùng giá đỡ từ xa • Các bản sao bổ sung được đặt ngẫu nhiên
  - Khách hàng đọc từ bản sao gần nhất
- Namenode phát hiện lỗi Datanode
  - Chọn Datanode mới cho các bản sao mới • Cân bằng việc sử dụng đĩa • Cân bằng lưu lượng truyền thông đến Datanode

# Cân bằng lại dữ liệu

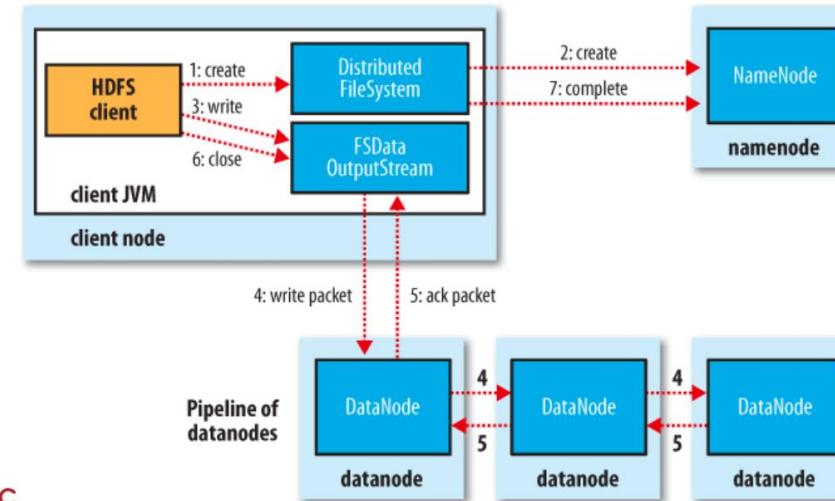
- Mục tiêu: % đĩa đầy trên Datanode phải tương tự nhau • Thường chạy khi Datanode mới được thêm vào • Cụm trực tuyến khi Rebalancer hoạt động • Rebalancer bị hạn chế để tránh tắc nghẽn mạng
- Công cụ dòng lệnh

# Độ chính xác của dữ liệu

- Sử dụng Checksum để xác thực dữ liệu
  - Sử dụng CRC32
- Tạo tệp • Máy khách
  - tính toán tổng kiểm tra cho mỗi 512 byte
  - Datanode lưu trữ tổng kiểm tra
- Truy cập tập tin
  - Khách hàng lấy dữ liệu và tổng kiểm tra từ Datanode
  - Nếu Xác thực không thành công, Khách hàng sẽ thử các bản sao khác

# Đường ống dữ liệu

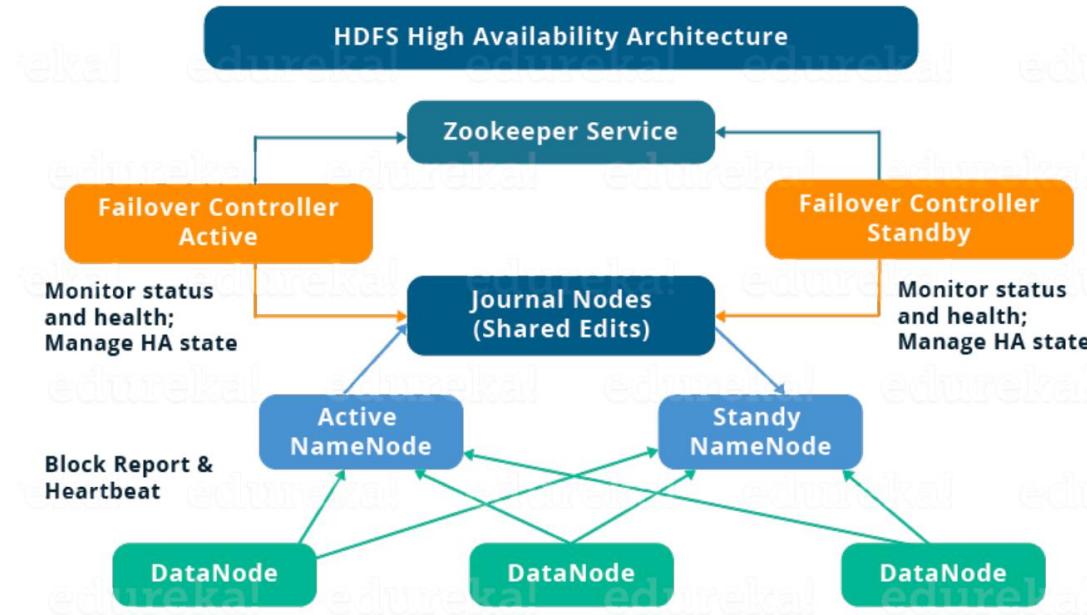
- Khách hàng lấy danh sách các Datanode để đặt bản sao của một khối
- Khách hàng ghi khối vào Datanode đầu tiên
- Datanode đầu tiên chuyển tiếp dữ liệu đến nút tiếp theo trong Đường ống
- Khi tất cả các bản sao được ghi, Client sẽ chuyển sang ghi khối tiếp theo vào tệp



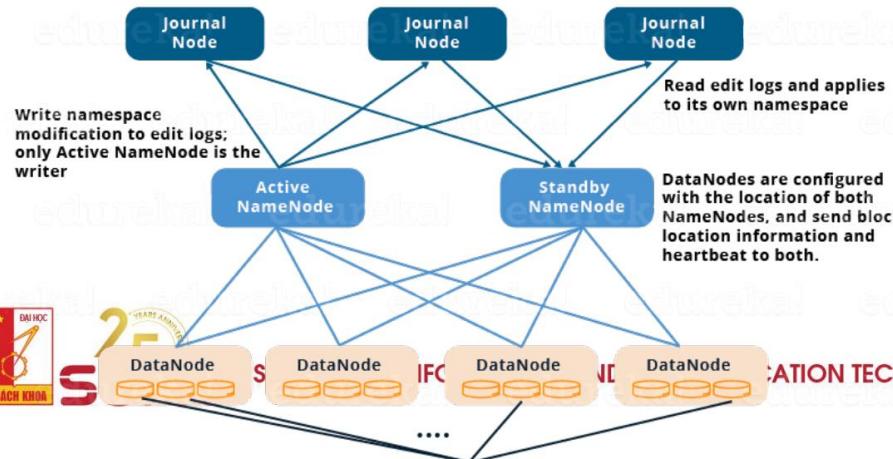
# Nút Tên phụ

- Namenode là điểm lỗi duy nhất
- Namenode thứ cấp
  - Kiểm tra điểm sao chép mới nhất của FsImage và Giao dịch Tập nhật ký.
  - Sao chép FsImage và Nhật ký giao dịch từ Namenode vào một thư mục tạm thời
- Khi Namenode khởi động lại
  - Hợp nhất FsImage và Nhật ký giao dịch thành một FsImage mới trong thư mục tạm thời
  - Tải FsImage mới lên Namenode
    - Nhật ký giao dịch trên Namenode bị xóa

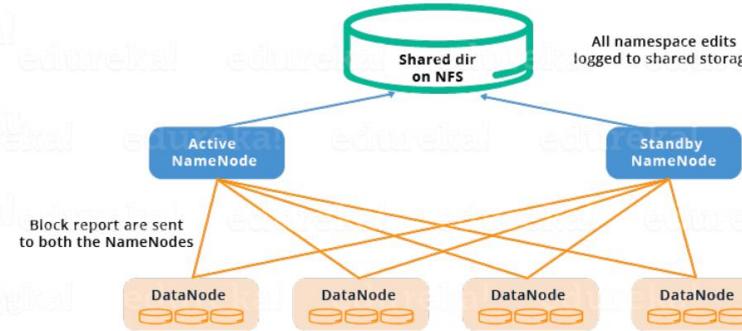
# Namenode có tính khả dụng cao (HA)



Nút Nhật ký Quorum



Lưu trữ chia sẻ



# Giao diện dòng lệnh HDFS

## List Files

<code>hdfs dfs -ls /</code>	List all the files/directories for the given hdfs destination path.
<code>hdfs dfs -ls -d /hadoop</code>	Directories are listed as plain files. In this case, this command will list the details of hadoop folder.
<code>hdfs dfs -ls -h /data</code>	Format file sizes in a human-readable fashion (eg 64.0m instead of 67108864).
<code>hdfs dfs -ls -R /hadoop</code>	Recursively list all files in hadoop directory and all subdirectories in hadoop directory.
<code>hdfs dfs -ls /hadoop/dat*</code>	List all the files matching the pattern. In this case, it will list all the files inside hadoop directory which starts with 'dat'.

## Read/Write Files

<code>hdfs dfs -text /hadoop/derby.log</code>	HDFS Command that takes a source file and outputs the file in text format on the terminal. The allowed formats are zip and TextRecordInputStream.
<code>hdfs dfs -cat /hadoop/test</code>	This command will display the content of the HDFS file test on your stdout .
<code>hdfs dfs -appendToFile /home/ubuntu/test1 /hadoop/text2</code>	Appends the content of a local file test1 to a hdfs file test2.

# Tải lên, tải xuống các tập tin

Upload/Download Files	
hdfs dfs -put /home/ubuntu/sample /hadoop	Copies the file from local file system to HDFS.
hdfs dfs -put -f /home/ubuntu/sample /hadoop	Copies the file from local file system to HDFS, and in case the local already exists in the given destination path, using -f option with put command will overwrite it.
hdfs dfs -put -l /home/ubuntu/sample /hadoop	Copies the file from local file system to HDFS. Allow DataNode to lazily persist the file to disk. Forces replication factor of 1.
hdfs dfs -put -p /home/ubuntu/sample /hadoop	Copies the file from local file system to HDFS. Passing -p preserves access and modification times, ownership and the mode.
hdfs dfs -get /newfile /home/ubuntu/	Copies the file from HDFS to local file system.
hdfs dfs -get -p /newfile /home/ubuntu/	Copies the file from HDFS to local file system. Passing -p preserves access and modification times, ownership and the mode.
hdfs dfs -get /hadoop/*.txt /home/ubuntu/	Copies all the files matching the pattern from local file system to HDFS.
hdfs dfs -copyFromLocal /home/ubuntu/sample /hadoop	Works similarly to the put command, except that the source is restricted to a local file reference.
hdfs dfs -copyToLocal /newfile /home/ubuntu/	Works similarly to the put command, except that the destination is restricted to a local file reference.
hdfs dfs -moveFromLocal /home/ubuntu/sample /hadoop	Works similarly to the put command, except that the source is deleted after it's copied.

# Quản lý tập tin

## File Management

<code>hdfs dfs -cp /hadoop/file1 /hadoop1</code>	Copies file from source to destination on HDFS. In this case, copying file1 from hadoop directory to hadoop1 directory.
<code>hdfs dfs -cp -p /hadoop/file1 /hadoop1</code>	Copies file from source to destination on HDFS. Passing -p preserves access and modification times, ownership and the mode.
<code>hdfs dfs -cp -f /hadoop/file1 /hadoop1</code>	Copies file from source to destination on HDFS. Passing -f overwrites the destination if it already exists.
<code>hdfs dfs -mv /hadoop/file1 /hadoop1</code>	Move files that match the specified file pattern <src> to a destination <dst>. When moving multiple files, the destination must be a directory.
<code>hdfs dfs -rm /hadoop/file1</code>	Deletes the file (sends it to the trash).
<code>hdfs dfs -rm -r /hadoop</code> <code>hdfs dfs -rm -R /hadoop</code> <code>hdfs dfs -rmdir /hadoop</code>	Deletes the directory and any content under it recursively.
<code>hdfs dfs -rm -skipTrash /hadoop</code>	The -skipTrash option will bypass trash, if enabled, and delete the specified file(s) immediately.
<code>hdfs dfs -rm -f /hadoop</code>	If the file does not exist, do not display a diagnostic message or modify the exit status to reflect an error.
<code>hdfs dfs -rmdir /hadoop1</code>	Delete a directory.
<code>hdfs dfs -mkdir /hadoop2</code>	Create a directory in specified HDFS location.
<code>hdfs dfs -mkdir -f /hadoop2</code>	Create a directory in specified HDFS location. This command does not fail even if the directory already exists.
<code>hdfs dfs -touchz /hadoop3</code>	Creates a file of zero length at <path> with current time as the timestamp of that <path>.



# Quyền sở hữu và xác thực

Ownership and Validation	
hdfs dfs -checksum /hadoop/file1	Dump checksum information for files that match the file pattern <src> to stdout.
hdfs dfs -chmod 755 /hadoop/file1	Changes permissions of the file.
hdfs dfs -chmod -R 755 /hadoop	Changes permissions of the files recursively.
hdfs dfs -chown ubuntu:ubuntu /hadoop	Changes owner of the file. 1st ubuntu in the command is owner and 2nd one is group.
hdfs dfs -chown -R ubuntu:ubuntu /hadoop	Changes owner of the files recursively.
hdfs dfs -chgrp ubuntu /hadoop	Changes group association of the file.
hdfs dfs -chgrp -R ubuntu /hadoop	Changes group association of the files recursively.
Filesystem	
hdfs dfs -df /hadoop	Shows the capacity, free and used space of the filesystem.
hdfs dfs -df -h /hadoop	Shows the capacity, free and used space of the filesystem. -h parameter Formats the sizes of files in a human-readable fashion.
hdfs dfs -du /hadoop/file	Show the amount of space, in bytes, used by the files that match the specified file pattern.
hdfs dfs -du -s /hadoop/file	Rather than showing the size of each individual file that matches the pattern, shows the total (summary) size.
hdfs dfs -du -h /hadoop/file	Show the amount of space, in bytes, used by the files that match the specified file pattern. Formats the sizes of files in a human-readable fashion.

# Sự quản lý

Administration	
hdfs balancer -threshold 30	Runs a cluster balancing utility. Percentage of disk capacity. This overwrites the default threshold.
hadoop version	To check the version of Hadoop.
hdfs fsck /	It checks the health of the Hadoop file system.
hdfs dfsadmin -safemode leave	The command to turn off the safemode of NameNode.
hdfs dfsadmin -refreshNodes	Re-read the hosts and exclude files to update the set of Datanodes that are allowed to connect to the Namenode and those that should be decommissioned or recommissioned.
hdfs namenode -format	Formats the NameNode.

# Giao diện nút tên HDFS



## Overview 'hd01:8020' (active)

Started:	Thu Mar 14 11:01:37 +0700 2019
Version:	3.1.1.3.1.0.0-78, re4f82af51faec922b4804d0232a637422ec29e64
Compiled:	Thu Dec 06 20:34:00 +0700 2018 by jenkins from (HEAD detached at e4f82af)
Cluster ID:	CID-a1dea38d-b6cf-44e4-b5e8-3bde87ffad35
Block Pool ID:	BP-1412866890-10.10.137.41-1544787807355

## Summary

Security is off.

Safemode is off.

165,739 files and directories, 99,858 blocks (99,858 replicated blocks, 0 erasure coded block groups) = 265,597 total filesystem object(s).

Heap Memory used 250.76 MB of 1011.25 MB Heap Memory. Max Heap Memory is 1011.25 MB.



SCHOOL OF INFORMATION AND COMMUNICATION TECHNOLOGY

# Giao diện nút tên HDFS (2)

In operation

Show 25 entries		Search:									
Node	Http Address	Last contact	Last Block Report	Capacity	Blocks	Block pool used	Version				
✓ hd01:50010 (10.10.137.41:50010)	<a href="http://hd01:50075">http://hd01:50075</a>	1s	251m	296.83 GB	<div style="width: 80%; background-color: orange;"></div>	99858	166.56 GB (56.11%)	3.1.1.3.1.0.0-78			
✓ hd02:50010 (10.10.137.42:50010)	<a href="http://hd02:50075">http://hd02:50075</a>	0s	57m	296.83 GB	<div style="width: 80%; background-color: green;"></div>	99858	166.57 GB (56.12%)	3.1.1.3.1.0.0-78			
✓ hd03:50010 (10.10.137.43:50010)	<a href="http://hd03:50075">http://hd03:50075</a>	2s	197m	296.83 GB	<div style="width: 80%; background-color: green;"></div>	99858	166.57 GB (56.12%)	3.1.1.3.1.0.0-78			

Showing 1 to 3 of 3 entries

Previous 1 Next

Hadoop Overview Datanodes Datanode Volume Failures Snapshot Startup Progress Utilities ▾

## Browse Directory

/	Go!			
Show 25 entries	Search:			
<input type="checkbox"/>	Permission	Owner	Group	Size
<input type="checkbox"/>	drwxrwxrwt	yarn	hadoop	0 B
<input type="checkbox"/>	drwxr-xr-x	hdfs	hdfs	0 B
<input type="checkbox"/>	drwxr-xr-x	yarn	hadoop	0 B
<input type="checkbox"/>	drwxr-xr-x	hdfs	hdfs	0 B
	Last Modified	Replication	Block Size	Name
	Jun 01 17:52	0	0 B	app-logs
	Dec 18 2018	0	0 B	apps
	Dec 14 2018	0	0 B	ats
	Dec 14 2018	0	0 B	atsv2



# Các giao diện HDFS khác

- Giao diện lập trình Java
- API tiết kiệm
- Cầu chì
- WebDAV

# Định dạng dữ liệu HDFS

Chữ

Tập tin trình tự

Avro

Sàn gỗ

Hàng cột được tối ưu hóa (ORC)



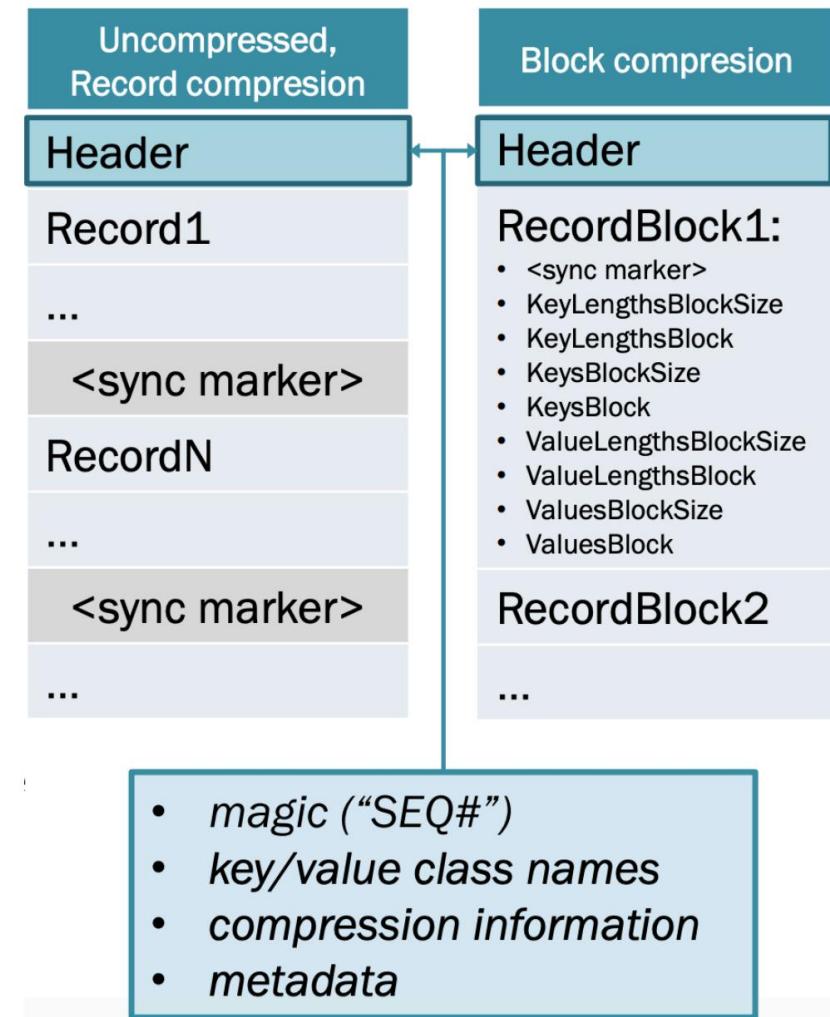
SCHOOL OF INFORMATION AND COMMUNICATION TECHNOLOGY

## Tập tin văn bản

- Bản ghi CSV, TSV, Json
- Định dạng thuận tiện để sử dụng để trao đổi giữa các ứng dụng hoặc tập lệnh
- Có thể đọc và phân tích cú pháp bằng con người • Không hỗ trợ nén khôi • Không hiệu quả khi truy vấn • Tốt cho lúc đầu, nhưng không đủ tốt cho thực tế mạng sống.

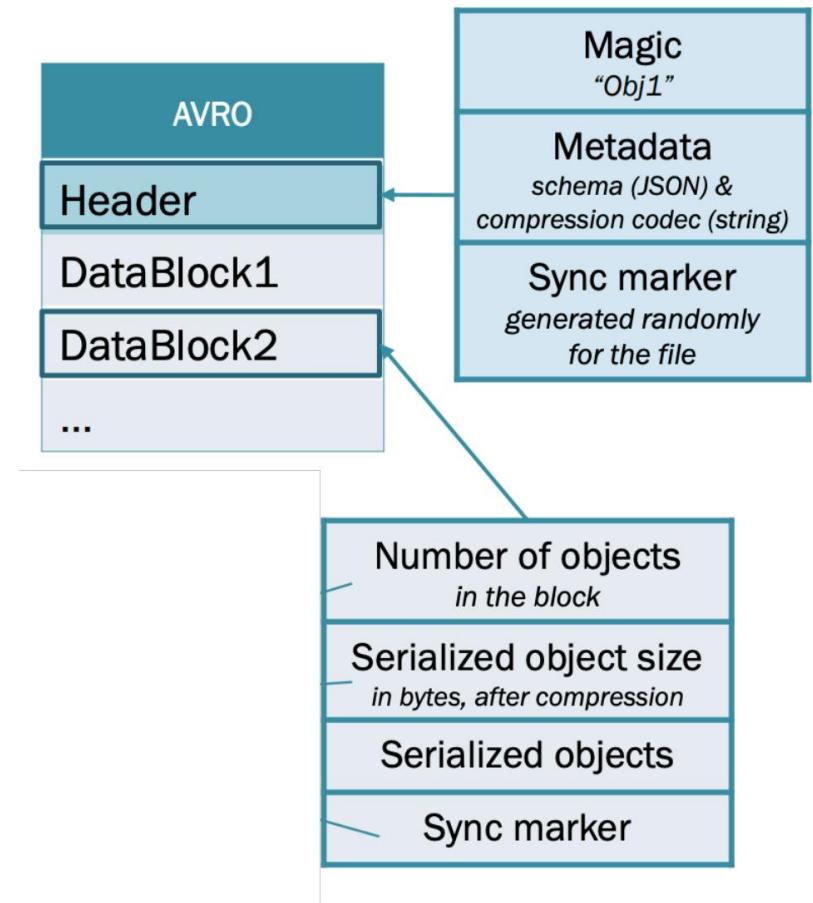
# Tập tin trình tự

- Cung cấp dữ liệu liên tục cấu trúc cho cặp khóa-giá trị nhị phân
- Thường được sử dụng để chuyển dữ liệu giữa các tác vụ Map Reduce
- Có thể được sử dụng như một kho lưu trữ để đóng gói các tệp nhỏ trong Hadoop
- Dựa trên hàng
- Nén
- Có thể chia tách • Hỗ trợ chia tách ngay cả khi dữ liệu được nén



# Avro

- Dựa trên hàng
- Hỗ trợ nén và chia tách (đối tượng)
- Sơ đồ dữ liệu linh hoạt
  - Sơ đồ (JSON) được bao gồm trong tệp
- Kiểu dữ liệu
  - kiểu nguyên thủy: null, boolean, int, long, ...
  - phức tạp: bản ghi, mảng, bản đồ, ...
- Tuần tự hóa dữ liệu nhị phân và JSON
- Phát hiện dữ liệu bị hỏng

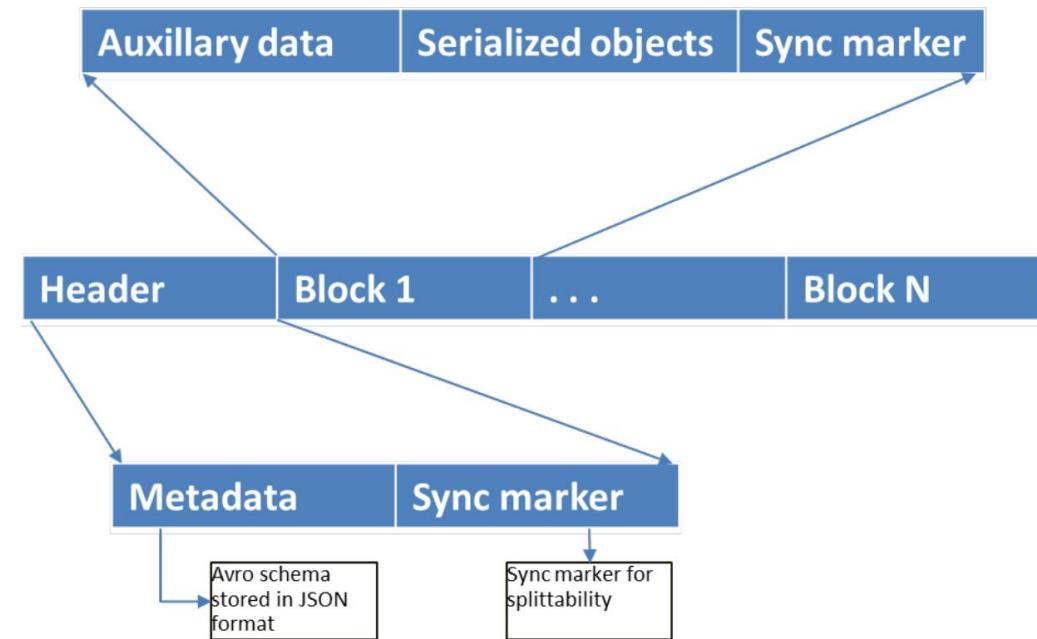


# Avro - Cấu trúc tập tin và ví dụ

Mẫu lược đồ AVRO ở định dạng JSON { "type" :

```
"record", "name" :  
  "tweets", "fields" :  
    [ {  
      "name" : "username",  
      "type" :  
        "string", }, { "name" :  
          "tweet", "type" :  
            "string", }, { "name" :  
              "timestamp",  
              "type" : "long", } ], "doc:" : "schema để lưu  
trữ tweet" }
```

Cấu trúc tập tin Avro



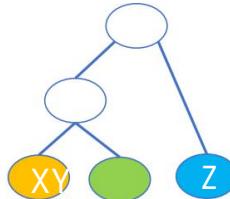
# Sàn gỗ

- Định dạng tệp nhị phân theo cột

Hiệu quả về mặt I/O đĩa khi cần truy vấn các cột cụ thể • Hỗ trợ nén và chia tách (trang)

- Hỗ trợ các cột lồng nhau (mã hóa Dremel)

Sơ đồ lồng nhau



Biểu diễn bảng

x	có	z
x1	y1	z1
x2	y2	z2
x3	y3	z3
x4	y4	z4
x5	y5	z5

Định dạng hàng

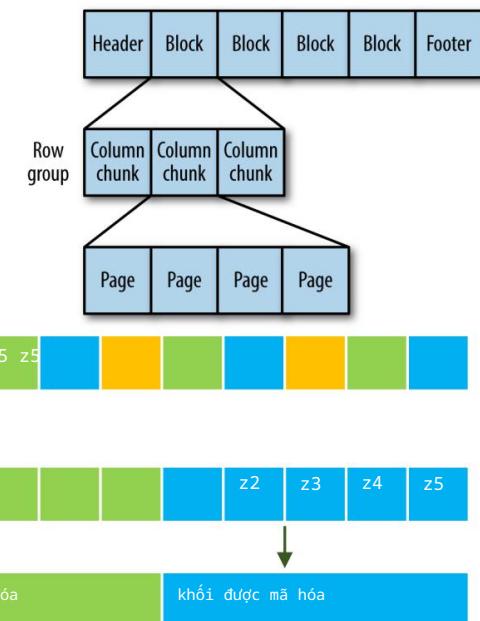
x1	y1	z1	x2	y2	z2	x3	y3	z3	x4	y4	z4	x5	y5	z5				
x1	y1	z1	x2	y2	z2	x3	y3	z3	x4	y4	z4	x5	y5	z5				
x1	y1	z1	x2	y2	z2	x3	y3	z3	x4	y4	z4	x5	y5	z5				
x1	y1	z1	x2	y2	z2	x3	y3	z3	x4	y4	z4	x5	y5	z5				
x1	y1	z1	x2	y2	z2	x3	y3	z3	x4	y4	z4	x5	y5	z5				

Định dạng cột

x1	x2	x3	x4	x5	y1	y2	y3	y4	y5	z1								
x1	x2	x3	x4	x5	y1	y2	y3	y4	y5	z1								
x1	x2	x3	x4	x5	y1	y2	y3	y4	y5	z1								
x1	x2	x3	x4	x5	y1	y2	y3	y4	y5	z1								
x1	x2	x3	x4	x5	y1	y2	y3	y4	y5	z1								

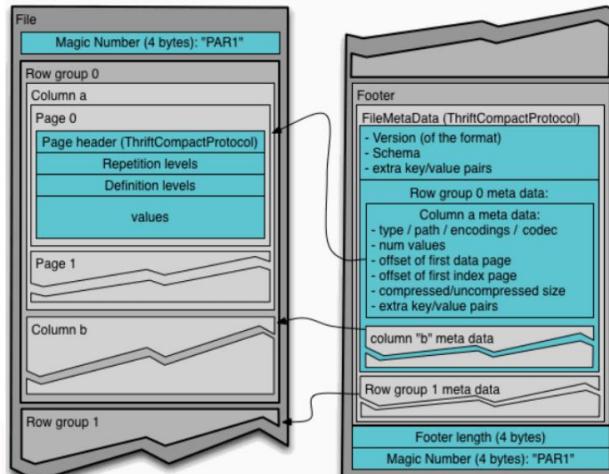
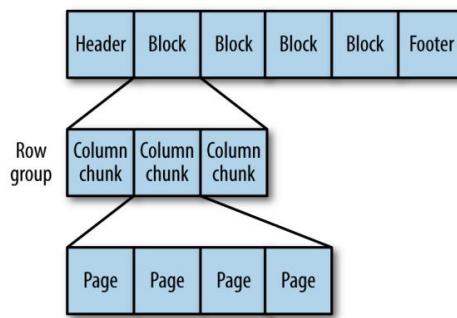
khối được mã hóa

Cấu trúc bên trong của tệp sàn gỗ



# Cấu trúc và cấu hình tập tin Parquet

Cấu trúc bên trong của tệp sàn gỗ



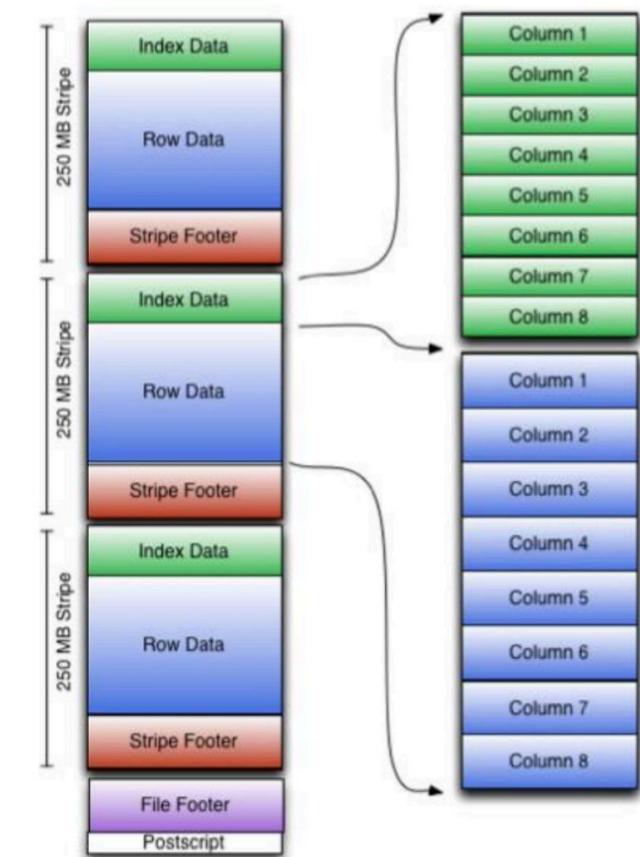
Các thông số sàn gỗ có thể cấu hình

Tên tài sản	Giá trị mặc định Mô tả	
sàn gỗ.khối.kích thước	128MB	Kích thước tính bằng byte của một khối (nhóm hàng).
sàn gỗ.trang.kích thước	1MB	Kích thước tính bằng byte của một trang.
parquet.dictionary.page.size 1MB		Kích thước tối đa được phép tính theo byte của một từ điển trước khi quay lại mã hóa thông thường cho một trang.
parquet.enable.dictionary	ĐÚNG VẬY	Có nên sử dụng mã hóa từ điển không.
sàn gỗ.nén	KHÔNG NÉN	Loại nén: KHÔNG NÉN, NHANH CHÓNG, GZIP & LZO

Parquet được tối ưu hóa cho khả năng nén cao và hiệu quả quét cao

# Hàng cột được tối ưu hóa (ORC)

- Hồ sơ RC
  - Mỗi cột được nén riêng lẻ trong nhóm hàng
- Tệp ORC
  - Nén ché độ khôi
  - Hồ trợ kiểu dữ liệu
  - Lưu trữ dữ liệu theo thứ tự (trong một dải)
- Lưu trữ các bộ sưu tập hàng và trong bộ sưu tập dữ liệu hàng được lưu trữ theo định dạng cột
- Giới thiệu một chỉ mục nhẹ cho phép bỏ qua các khói hàng không liên quan
- Có thể chia tách: cho phép xử lý song song các tập hợp hàng
- Chỉ mục có giá trị tổng hợp theo cấp độ cột (min, max, tổng và đếm)



## Tài liệu tham khảo

- Borthakur, Dhruba. "Hướng dẫn kiến trúc HDFS." Dự án Apache Hadoop 53.1-13 (2008): 2.
- Ghemawat, Sanjay, Howard Gobio và Shun-Tak Leung. "Hệ thống tập tin Google." Biên bản hội thảo ACM lần thứ mười chín về Nguyên tắc hệ điều hành. 2003.
- Vohra, Deepak. "Apache parquet." Hệ sinh thái Hadoop thực tế. Apress, Berkeley, CA, 2016. 325-335.



25  
YEARS ANNIVERSARY  
**SOICT**

**VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG**  
SCHOOL OF INFORMATION AND COMMUNICATION TECHNOLOGY

Cảm ơn sự  
chú ý  
của bạn!!!

