

# CHƯƠNG 6

---

HADOOP NỀN TẢNG ĐỂ XỬ LÝ PHÂN TÁN

# NỘI DUNG BÀI HỌC

- TỔNG QUAN VỀ HADOOP
- KIẾN TRÚC VÀ CƠ CHẾ CỦA HDFS
- CƠ CHẾ XỬ LÝ PHÂN TÁN (MAP-REDUCE)

# TỔNG QUAN VỀ HADOOP

## ❖ Tiến trình phát triển:

- Là một dự án được khởi xướng từ Dough Cutting.
- Ban đầu Hadoop được bắt nguồn từ Nutch một hệ mã nguồn mở hỗ trợ xây dựng máy tìm kiếm.
- Vào năm 2003, Google giới thiệu kiến trúc về hệ lưu trữ file phân tán có tên là Google File System (GFS).
- Lấy ý tưởng từ GFS, Nutch đã phát triển riêng một nền tảng lưu trữ khác gọi là Nutch Distributed File System (NDFS)



# TỔNG QUAN VỀ HADOOP

## ❖ Các thành phần của Hadoop:

- **Hadoop-core:**

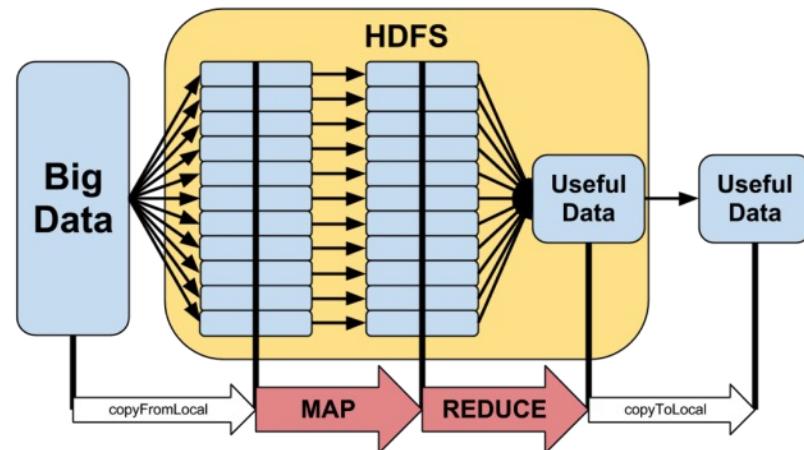
Đây là phần lõi của Hadoop, đóng vai trò nền tảng cung cấp các cổng/phương thức giao tiếp nhập/xuất (IO) của Hadoop. Hadoop-core liên kết 2 thành phần còn lại là HDFS và Map-Reduce.

- **HDFS (Hadoop Distributed File System):**

Đóng vai trò bộ phận lưu trữ và hệ thống kiến trúc tập tin phân tán.

Kiến trúc của HDFS để đáp ứng cho việc lưu trữ các khối lượng dữ liệu lớn cũng như quản lý tối ưu hóa việc lưu trữ, sắp xếp dữ liệu lưu trữ một cách hiệu quả dưới hình thức phát tán dữ liệu ra các máy con (node) với số lượng node có thể lên đến hàng chục hoặc hàng triệu.

HDFS có khả năng chịu lỗi cao cũng như đảm bảo cho việc kiểm soát quá trình truyền tải dữ liệu giữa các node được thông suốt



# TỔNG QUAN VỀ HADOOP

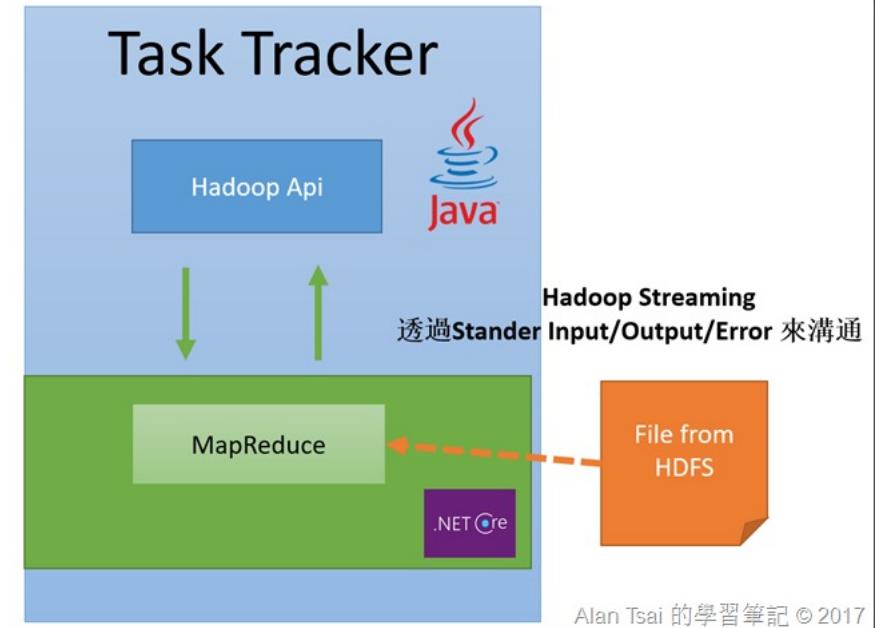
## ❖ Các thành phần của Hadoop:

- Map – reduce framework:

Đóng vai trò cơ chế xử lý song song của hadoop.

Map – reduce có thể coi là một dạng framework trong lập trình, cung cấp thư viện để phát triển các ứng dụng cũng như giải thuật lập trình các ứng dụng xử lý theo mô hình phân tán của Hadoop.

Điểm nổi bật của Map-reduce là dễ tiếp cận cũng như dễ lập trình



# TỔNG QUAN VỀ HADOOP

## ❖ Kiến trúc của Hadoop-Cluster:

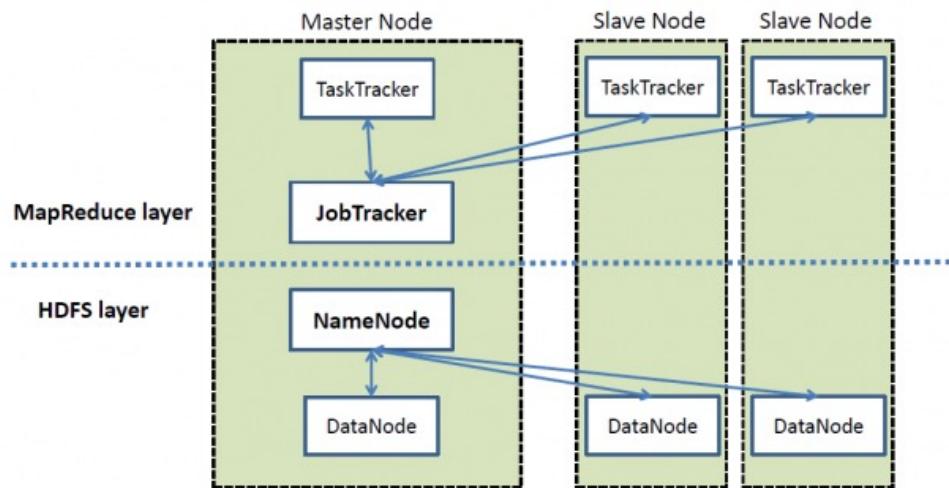
Một kiến trúc Hadoop-Cluster bao gồm 2 thành phần chính là Namenode và datanode, đây sẽ là cụm máy chủ vật lý hoạt động độc lập với nhau và mỗi máy sẽ chứa các thành phần khác nhau, bao gồm:

- **NameNode (Master):**

Đóng vai trò chủ đạo trọng toàn bộ kiến trúc Hadoop-Cluster, chứa toàn bộ cơ sở dữ liệu metadata của toàn bộ hệ thống HDFS.

HDFS phân tán các tập tin thành nhiều block và đánh ID cho từng block, các block này sẽ được gửi xuống các datanode để lưu trữ.

Vai trò của Namenode sẽ quản lý các ID của các block và quá trình giám sát tình trạng cũng như việc lưu trữ của các Data node trong toàn hệ thống.



# TỔNG QUAN VỀ HADOOP

## ❖ Kiến trúc của Hadoop-Cluster:

- **Jobtracker (Master):**

- ✓ Là phần lõi trung tâm của cơ chế Map-reduce đóng vai trò làm bộ phận tiếp nhận toàn bộ các yêu cầu xử lý (Job).
- ✓ Sau đó nó chia nhỏ các yêu cầu này ra thành nhiều phần nhỏ (task) và gửi đến các máy con trong mạng cluster để thực thi.
- ✓ Jobtracker giám sát các task này thông qua hồi đáp từ Tasktracker tại các máy con.

- **Data node (Slave):**

- ✓ Làm nhiệm vụ chứa và quản lý dữ liệu trong mạng cluster. Đây là các dữ liệu được phân chia từ Namenode
- ✓ Chịu trách nhiệm xử lý các yêu cầu read/write/delete/update được gửi từ Namenode thông qua việc tương tác trực tiếp lên các block dữ liệu đang lưu trữ.

# TỔNG QUAN VỀ HADOOP

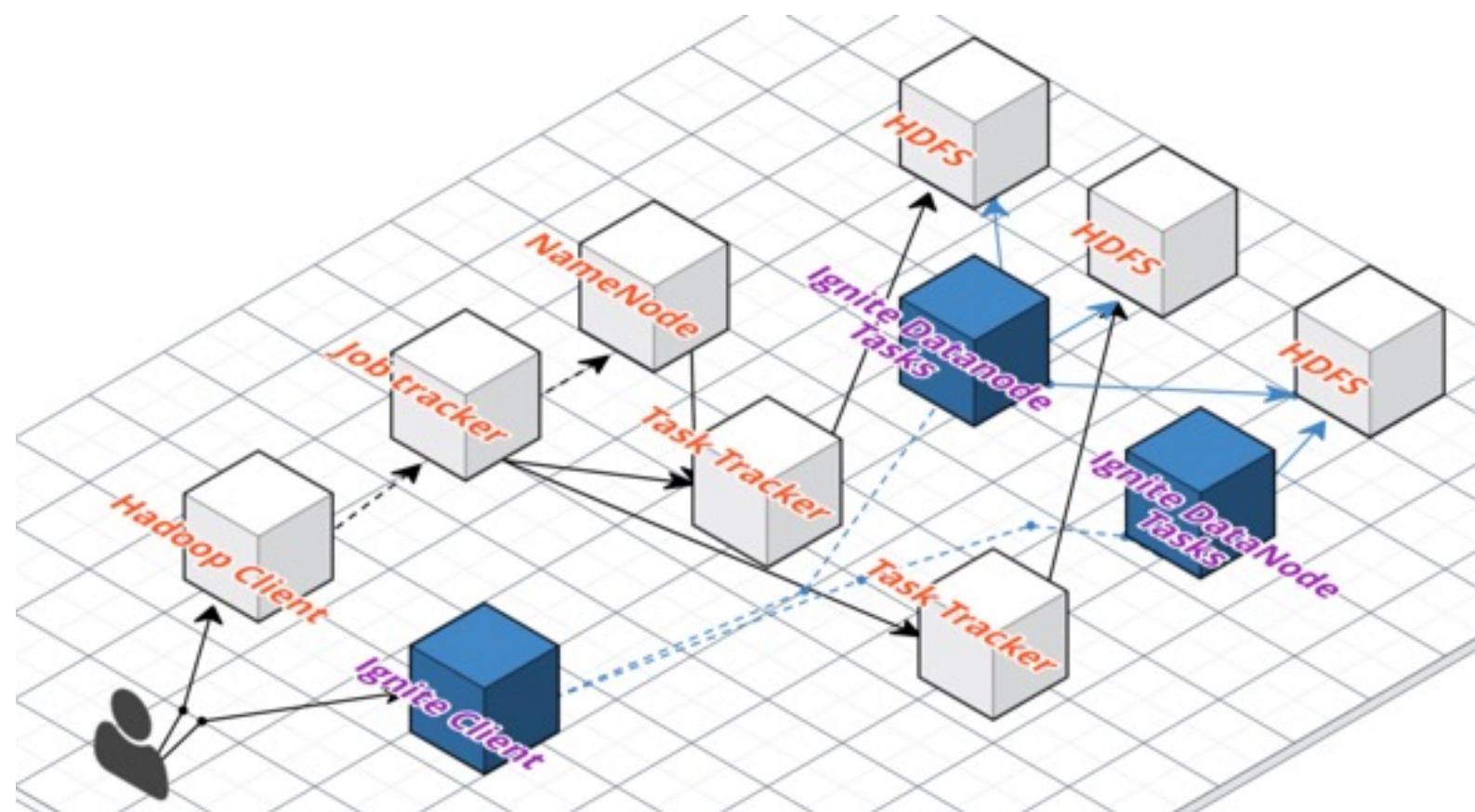
## ❖ Kiến trúc của Hadoop-Cluster:

- **Tasktracker (Slave):**

Nằm ở Datanode trong cụm máy (cluster).

Chịu trách nhiệm nhận các tác vụ (task) được gửi đến bởi JobTracker, sau đó thực thi chúng tại node con đó.

Sau khi hoàn thành tác vụ Tasktracker sẽ gửi thông báo đến Jobtracker.



# KIẾN TRÚC VÀ CƠ CHẾ CỦA HDFS

## ❖ Lưu trữ dữ liệu:

- **Khả năng chịu lỗi (fault tolerance):**

- ✓ Trong hệ thống phân tán, có khi sẽ có từ hàng chục đến hàng ngàn các máy con (node), việc không đồng bộ về chất lượng và số lượng của các node là chắc chắn xảy ra.
- ✓ Mô hình HDFS đảm bảo cho việc phục hồi và giám sát các node khi xảy ra tình trạng mất dữ liệu cục bộ.

- **Phân chi dữ liệu hợp lý:**

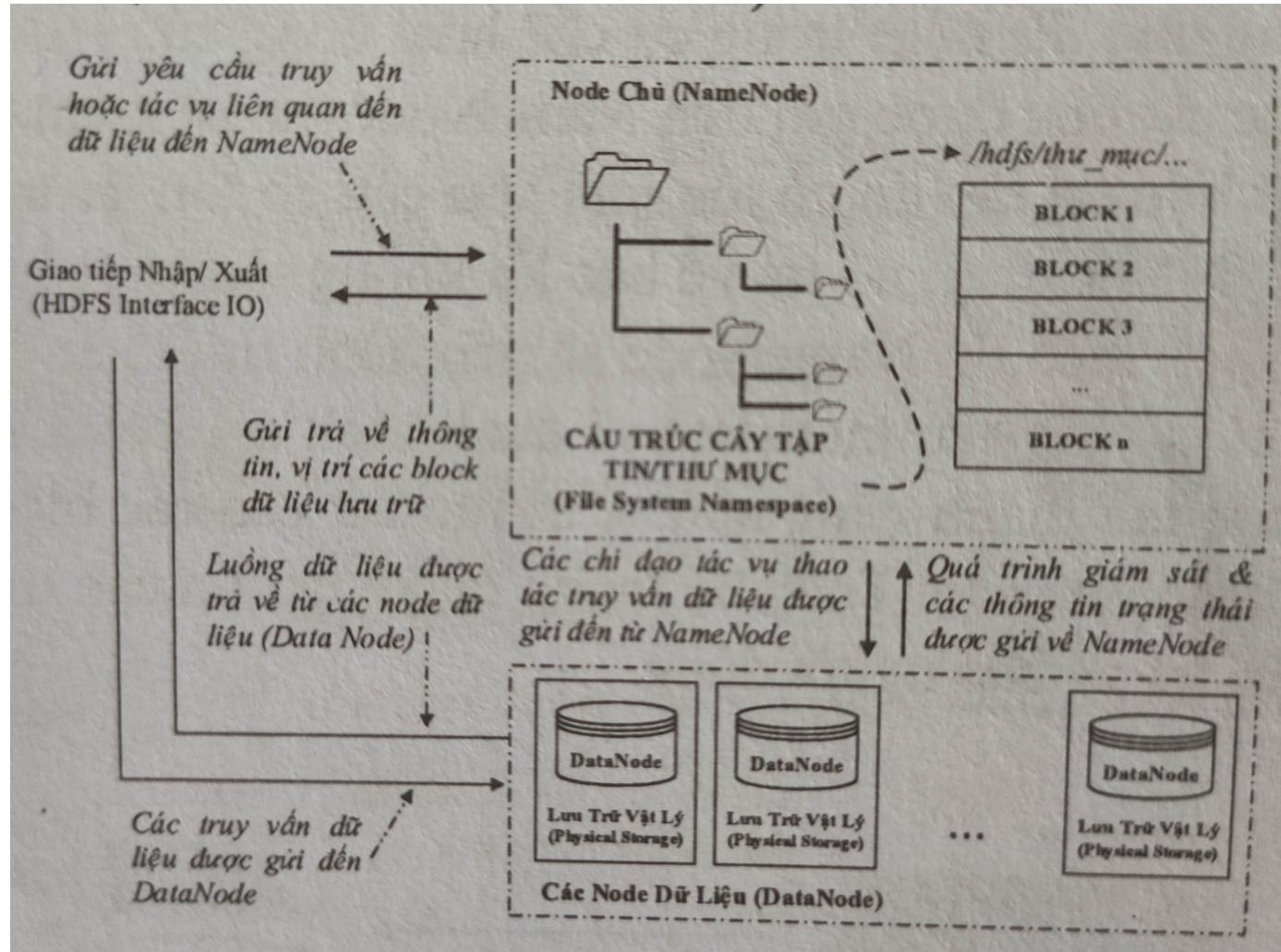
- ✓ Ưu tiên các tập tin với khối lượng lớn hơn là nhiều tập tin có khối lượng nhỏ.
- ✓ Việc tối ưu dung lượng lớn cho các tập tin giúp việc xử lý tác vụ hiệu quả hơn trong mô hình.
- ✓ Tuy nhiên, việc phân tán dữ liệu xuống các block còn phụ thuộc vào nhu cầu của người sử dụng.

- **Truy xuất dữ liệu song song:**

- ✓ Việc phân tán dữ liệu ra các node để xử lý giúp cho công đoạn xử lý dữ liệu và xử lý các task được diễn ra nhanh hơn, do công việc được thực thi cùng lúc tại các node làm tăng hiệu suất của toàn hệ thống.

# KIẾN TRÚC VÀ CƠ CHẾ CỦA HDFS

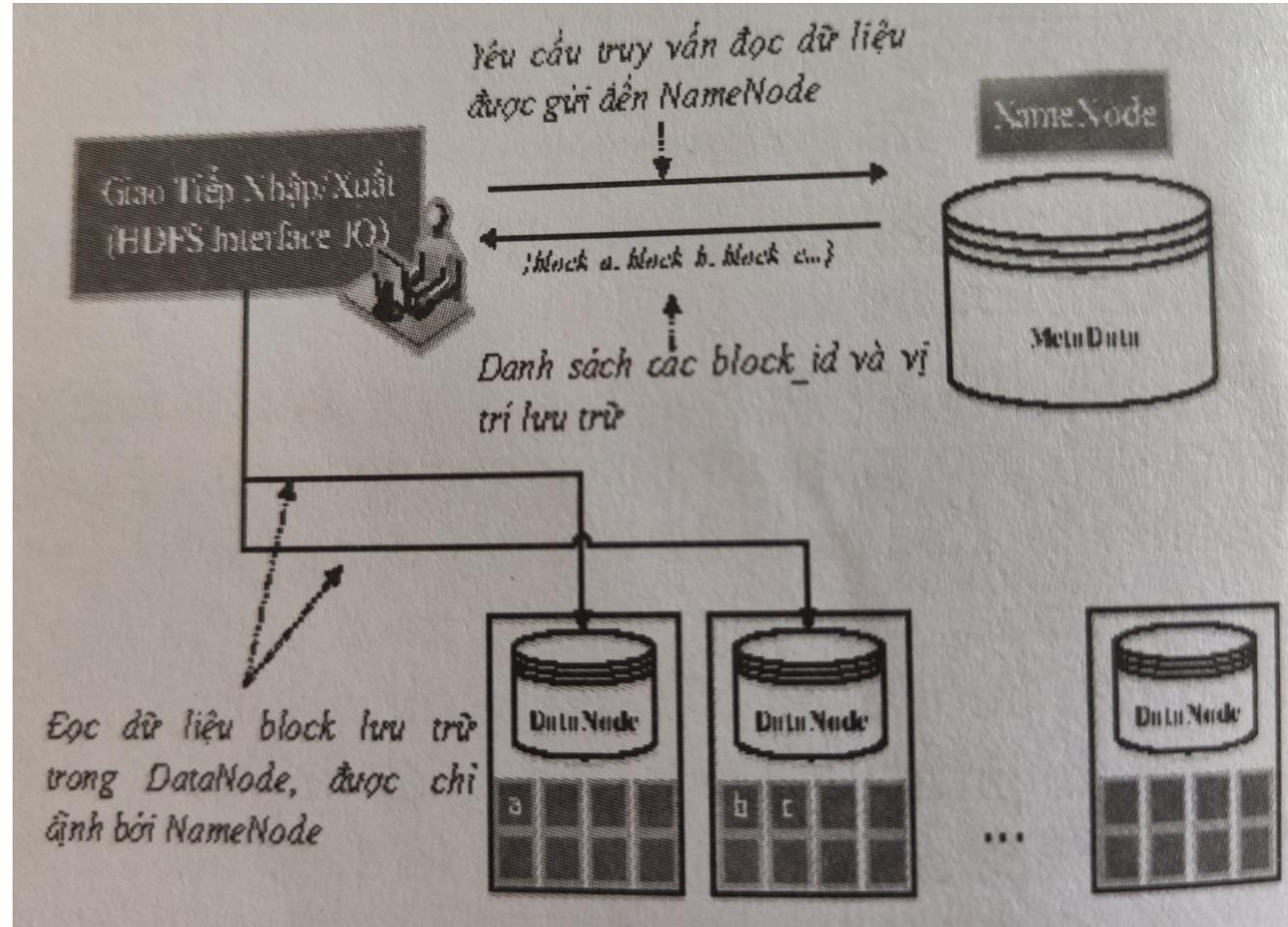
## ❖ Kiến trúc của HDFS:



# KIẾN TRÚC VÀ CƠ CHẾ CỦA HDFS

## ❖ Các cơ chế thao tác dữ liệu của HDFS:

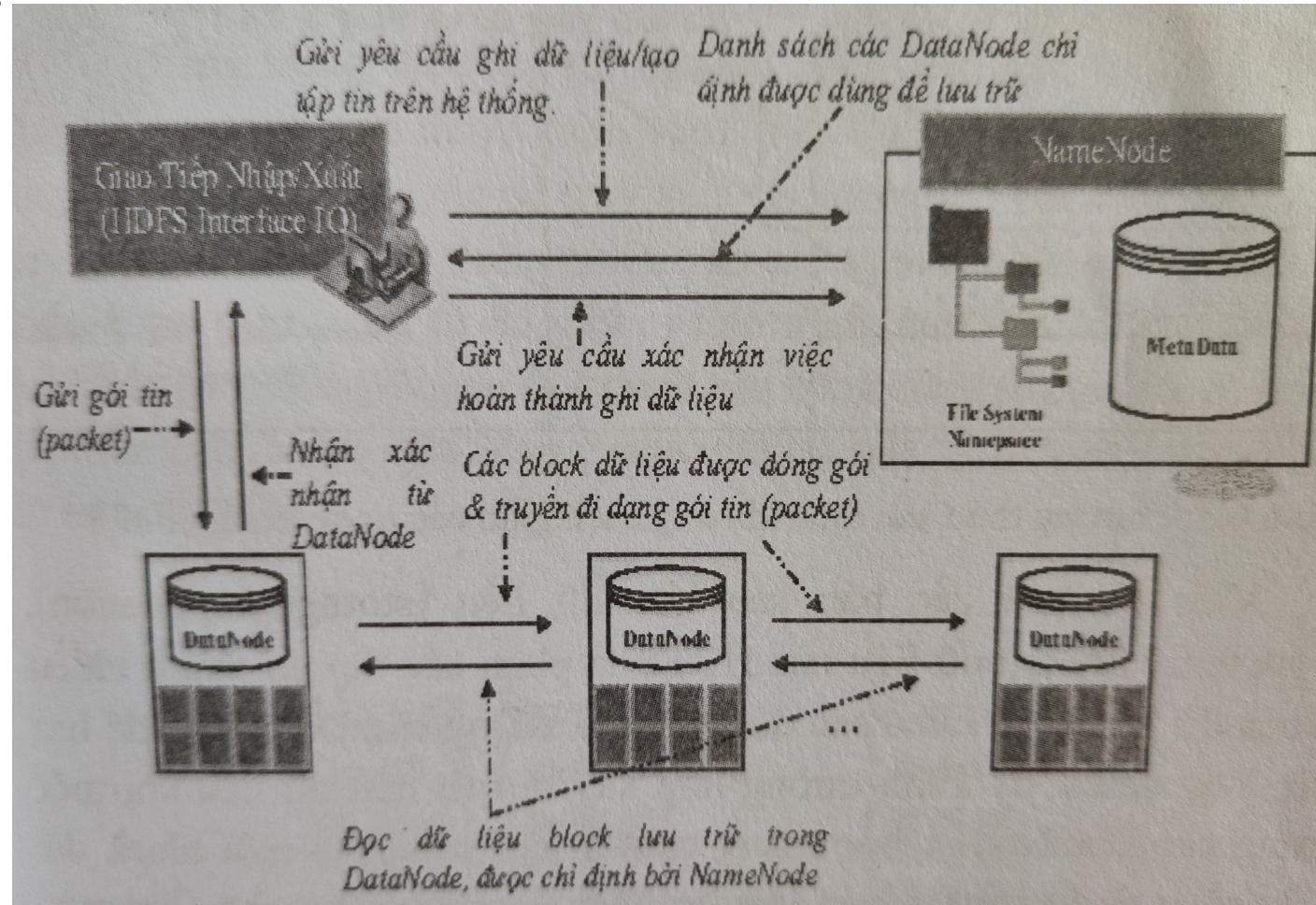
- Cơ chế đọc dữ liệu:



# KIẾN TRÚC VÀ CƠ CHẾ CỦA HDFS

## ❖ Các cơ chế thao tác dữ liệu của HDFS:

- Cơ chế ghi dữ liệu:



# KIẾN TRÚC VÀ CƠ CHẾ CỦA HDFS

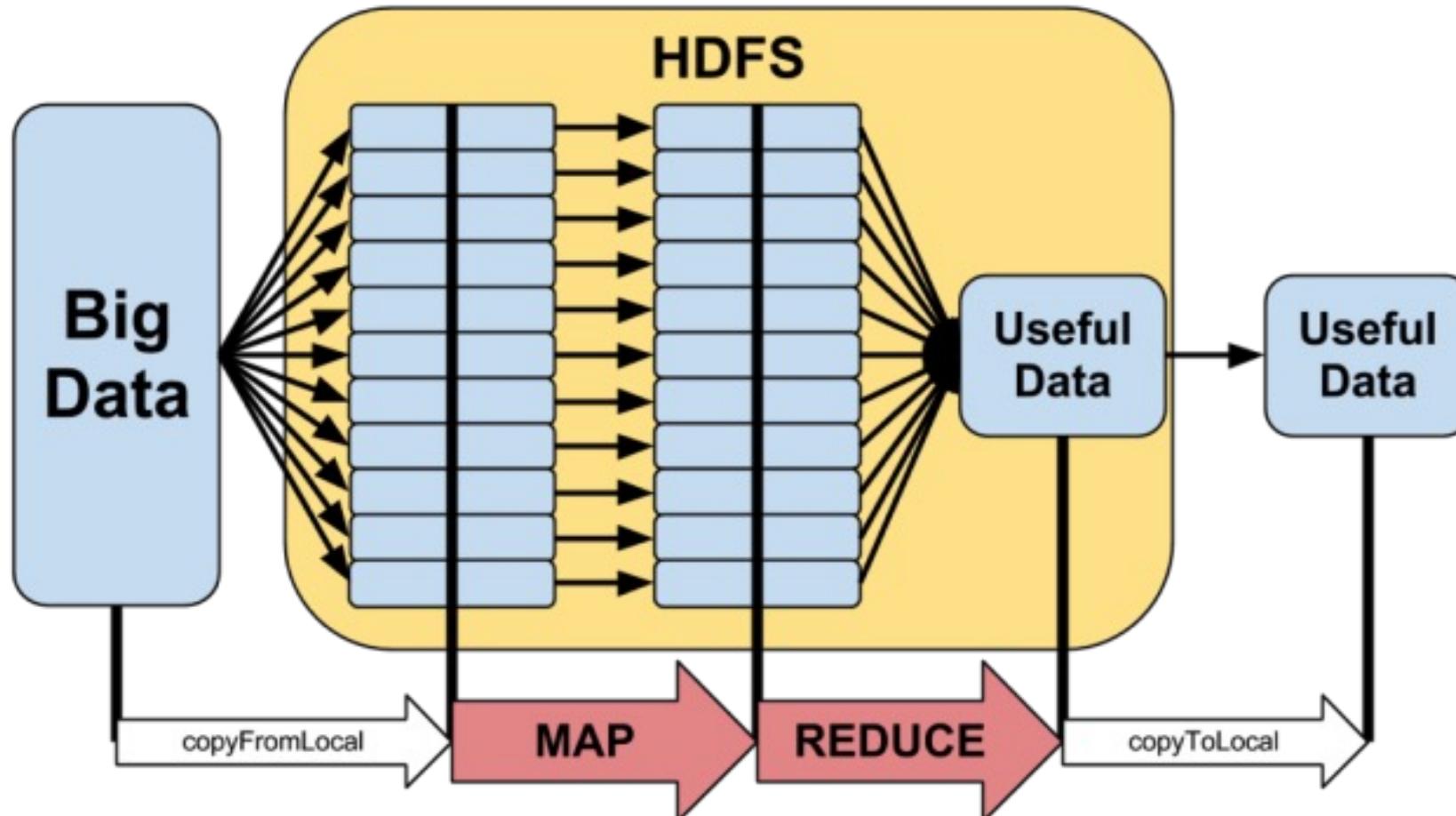
## ❖ Các cơ chế thao tác dữ liệu của HDFS:

- **Cơ chế điều phối cân bằng:**

- ✓ Hadoop sẽ xác định khoảng cách giữa các node bằng cách thông qua địa chỉ IP và các nhánh mạng gần nhất.
- ✓ Sau đó các block sẽ được gửi xuống lưu trữ tại các datanode.
- ✓ Để tránh tình trạng bị mất dữ liệu, các node trong cùng 1 cluster sẽ chứa bản sao lưu nhân bản của các block khác.
- ✓ Mức độ sao lưu các nhân bản càng cao thì độ an toàn của hệ thống càng cao.
- ✓ HDFS -balancer là chức năng giúp cân bằng việc di chuyển các block sao cho tất cả các node trong hệ thống có mức độ lưu trữ cân bằng với nhau.

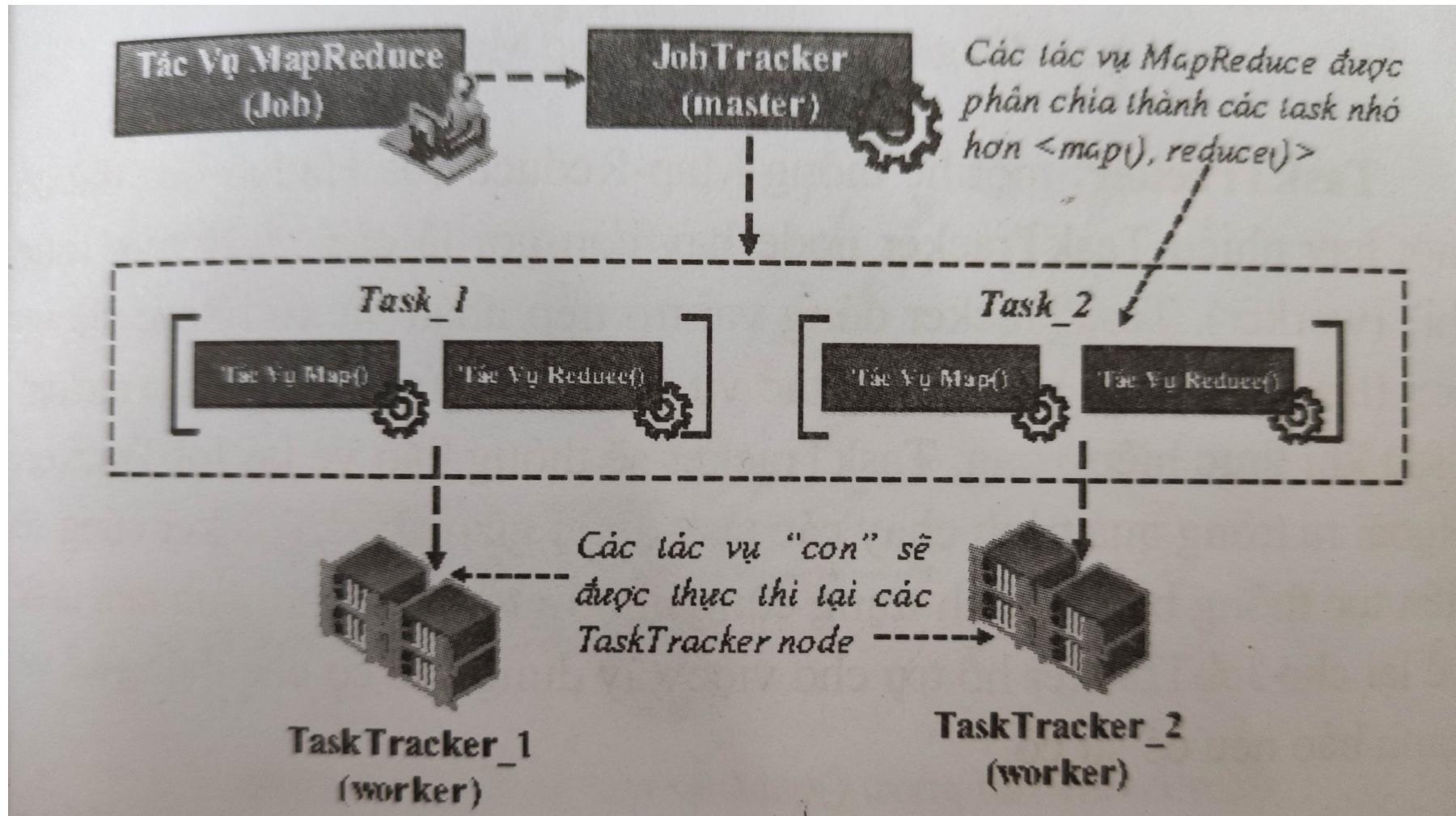
# CƠ CHẾ XỬ LÝ PHÂN TÁN (MAP-REDUCE)

❖ Các thành phần của hadoop mapreduce:



# CƠ CHẾ XỬ LÝ PHÂN TÁN (MAP-REDUCE)

## ❖ Các thành phần của hadoop mapreduce:



# CƠ CHẾ XỬ LÝ PHÂN TÁN (MAP-REDUCE)

## ❖ Các thành phần của hadoop mapreduce:

- **Jobtracker:**

- ✓ Là máy trung tâm của hệ thống Hadoop.
- ✓ Chịu trách nhiệm tiếp nhận toàn bộ các yêu cầu thực thi của công việc theo kiểu map-reduce, sau đó phân chia các job này thành nhiều tác vụ con (task) và phân tán xuống các máy vật lý con khác được kiểm soát bởi Tasktracker.
- ✓ Jobtracker đóng vai trò như 1 namenode.
- ✓ Giám sát toàn bộ tình trạng của các tasktraker thông qua cơ chế gửi/nhận thông báo.

# CƠ CHẾ XỬ LÝ PHÂN TÁN (MAP-REDUCE)

## ❖ Các thành phần của hadoop mapreduce:

- **Tasktracker:**

- ✓ Trong một hệ thống Hadoop-Mapreduce có thể có 1 hoặc nhiều Tasktracker (hay còn gọi là máy thợ Worker).
- ✓ Đóng vai trò tiếp nhận và xử lý các tác vụ con (task).
- ✓ Sau khi thực hiện xong tasktracker sẽ thông báo lại cho Jobtracker.
- ✓ Ngoài ra, trong quá trình chạy tasktracker cũng liên tục thông báo cho Jobtracker về tình trạng của task, để phục vụ cho việc thống kê và báo cáo nếu có sự cố.

# NỘI DUNG BÀI HỌC

- TỔNG QUAN VỀ HADOOP
- KIẾN TRÚC VÀ CƠ CHẾ CỦA HDFS
- CƠ CHẾ XỬ LÝ PHÂN TÁN (MAP-REDUCE)