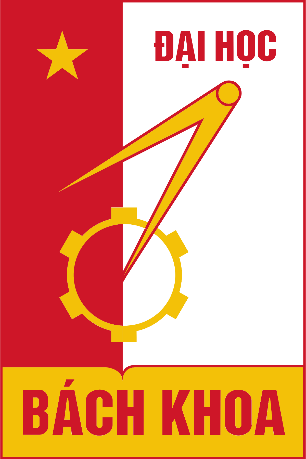
**ĐẠI HỌC BÁCH KHOA HÀ NỘI**

**TRƯỜNG CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG**

**KỸ THUẬT MÁY TÍNH**



**BÁO CÁO BÀI TÂP LỚN**

**Đề tài: Lưu trữ và xử lý dữ liệu về chứng khoán**

Lớp: 144943

Học phần: Lưu trữ và xử lý dữ liệu lớn

Mã học phần: IT4931

Giảng viên hướng dẫn: TS. Trần Việt Trung

Danh sách thành viên nhóm:

|  |  |
| --- | --- |
| Họ và tên | Mã số sinh viên |
| Thái Bảo Trung | 20204858 |
| Nguyễn Khánh Trường Lộc | 20204763 |
| Nguyễn Công Tú | 20204800 |
| Mai Minh Nhật | 20194346 |
| Nguyễn Vũ Đình Tuấn | 20204802 |

Hà Nội, tháng 12 năm 2023

**MỤC LỤC**

[CHƯƠNG 1: TỔNG QUAN ĐỀ TÀI 3](#_Toc154685882)

[1.1 Giới thiệu đề tài 3](#_Toc154685883)

[1.2 Mục tiêu đề tài 3](#_Toc154685884)

[1.3 Các bước thực hiện đề tài 3](#_Toc154685885)

[CHƯƠNG 2: TỔNG QUAN HỆ THỐNG 4](#_Toc154685886)

[2.1 Sơ đồ hệ thống 4](#_Toc154685887)

[2.2 Chi thiết thành phần hệ thống 5](#_Toc154685888)

[2.2.1 Hadoop cluster 5](#_Toc154685889)

[2.2.2 Spark cluster 6](#_Toc154685890)

[2.2.3 Kafka cluster 6](#_Toc154685891)

[CHƯƠNG 3. NHỮNG TRẢI NGHIỆM KHI XÂY DỰNG HỆ THỐNG 8](#_Toc154685892)

[3.1 Lưu trữ dữ liệu vào HDFS 8](#_Toc154685893)

[Trải nghiệm 1: Lưu trữ dữ liệu phân tán 8](#_Toc154685894)

[Trải nghiệm 2: Nhân bản dữ liệu để tăng tính chịu lỗi 8](#_Toc154685895)

[Trải nghiệm 3: Thử loại bỏ các node để chứng minh khả năng chịu lỗi của hệ thống 9](#_Toc154685896)

[3.2 Xử lý dữ liệu bằng spark 11](#_Toc154685897)

[Trải nghiệm 4: Dùng spark để xử lý dữ liệu 11](#_Toc154685898)

[Trải nghiệm 5: Thử hoạt động của spark sau khi tắt một workernode 11](#_Toc154685899)

[3.3 Truy xuất dữ liệu 12](#_Toc154685900)

[Trải nghiệm 6: Truy vấn dữ liệu bằng Pyspark 12](#_Toc154685901)

[Trải nghiệm 7: Lấy dữ liệu và hiển thị 13](#_Toc154685902)

[CHƯƠNG 4 KẾT LUẬN 15](#_Toc154685903)

# CHƯƠNG 1: TỔNG QUAN ĐỀ TÀI

## Giới thiệu đề tài

Trong thời đại công nghệ hiện nay, thị trường chứng khoán trở nên ngày càng phức tạp với lượng dữ liệu lớn được sinh ra liên tục từ các giao dịch, tin tức tài chính, và các yếu tố khác liên quan. Để hiểu rõ hơn về xu hướng và dự đoán biến động của thị trường, việc lưu trữ và xử lý dữ liệu lớn đó trở thành một thách thức quan trọng. Đề tài này hướng tới việc vận dụng các kiến thức học được từ học phần từ đó xây dựng một hệ thống để lưu trữ và xử lý dữ liệu chứng khoán để phục vụ các mục đích phân tích và đánh giá.

## Mục tiêu đề tài

* Vận dụng được các kiến thức cơ bản về lưu trữ và xử lý dữ liệu lớn.
* Phân tích và triển khai các thuật toán xử lý dữ liệu chứng khoán để đưa ra các thông tin phù hợp.
* Xây dựng hệ thống lưu trữ và xử lý dữ liệu chứng khoán dễ dàng quản lý và truy xuất.

## Các bước thực hiện đề tài

* **Thu Thập Dữ Liệu**: Tập trung vào việc thu thập và làm sạch dữ liệu chứng khoán từ các nguồn
* **Lưu Trữ Dữ Liệu**: Phát triển mô hình lưu trữ dữ liệu từ những kiến thức đã học.
* **Xử Lý Dữ Liệu**: Áp dụng các phương pháp xử lý dữ liệu để trích xuất thông tin.
* **Phân Tích và Đánh Giá**: Sử dụng các kỹ thuật phân tích dữ liệu để hiểu rõ hơn về xu hướng thị trường và đánh giá kết quả.

# CHƯƠNG 2: TỔNG QUAN HỆ THỐNG

## 2.1 Sơ đồ hệ thống

A diagram of a computer

Description automatically generated

*Hình 1. Kiến trúc hệ thống*

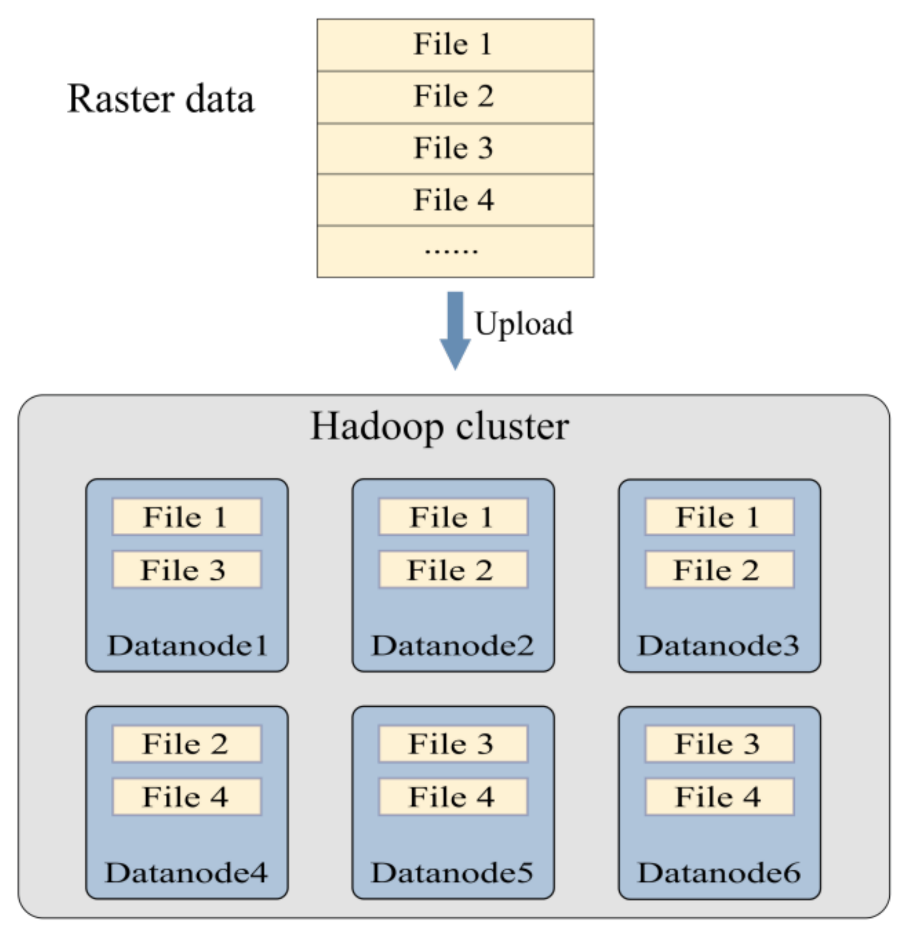
- Hệ thống sử dụng 1 masternode và 2 datanodes, sau khi datanode thu thập dữ liệu và gửi vào kafka, masternode subcribe vào topic chứa dữ liệu được thu thập, sử dụng Spark để xử lý và làm sạch dữ liệu. Dữ liệu sau đó được lưu vào Hadoop để có thể lưu trữ dữ liệu phân tán.

- Dữ liệu về chứng khoán sau khi lưu vào Hadoop có thể được truy vấn để phân tích, đánh giá và biểu diễn ra thông tin cần thiết.

## 2.2 Chi thiết thành phần hệ thống

### 2.2.1 Hadoop cluster

Hadoop Cluster bao gồm nhiều máy tính chạy Hadoop Distributed File System (HDFS), một hệ thống lưu trữ dữ liệu phân tán, và Hadoop MapReduce, một mô hình lập trình và xử lý dữ liệu phân tán. Hadoop Cluster cho phép phân chia công việc xử lý dữ liệu thành các tác vụ nhỏ và phân phối chúng trên nhiều máy tính, giúp tăng hiệu suất xử lý và khả năng mở rộng.



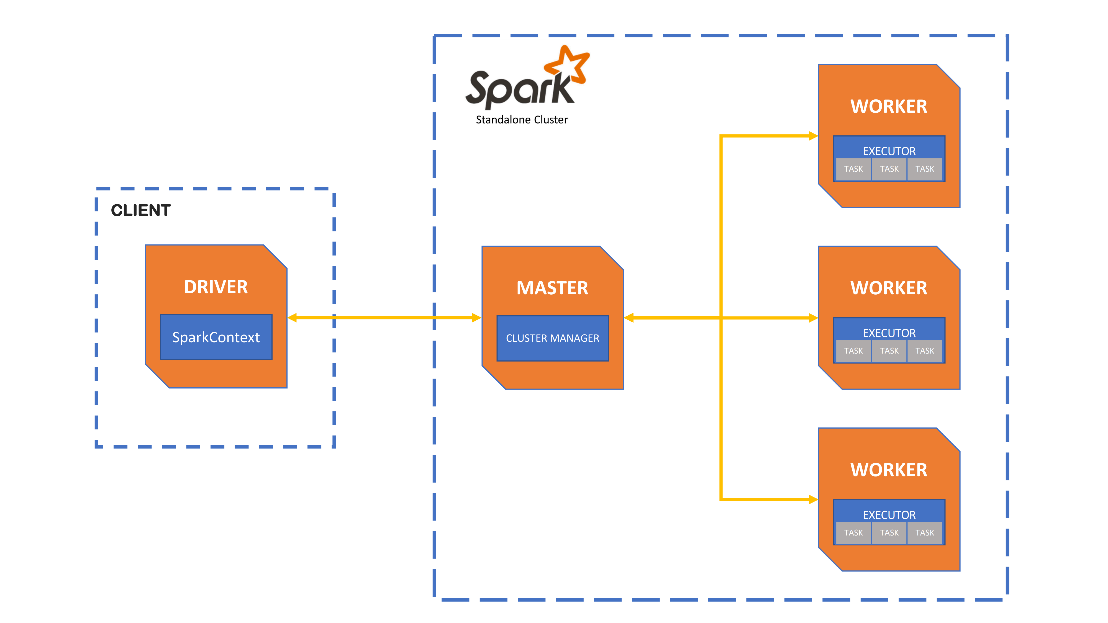
*Hình 2. Hadoop cluster*

HDFS là nơi lưu dữ liệu của Hadoop, HDFS chia chia nhỏ dữ liệu thành các đơn vị dữ liệu nhỏ hơn gọi là các blocks và lưu trữ chúng phân tán trong các node của cụm Hadoop. HDFS sử dụng kiến trúc master/slave, trong đó master gồm một masternode để quản lý hệ thống file metadata và một hay nhiều slave datanodes để lưu trữ dữ liệu.

Dữ liệu về chứng khoán được thu thập và xử lý sẽ được lưu trong cụm Hadoop gồm một masternode và 2 datanodes. Khi lượng dữ liệu tăng lên, chúng ta có thể bổ sung thêm datanode để tăng cường dung lượng lưu trữ của hệ thống.

### 2.2.2 Spark cluster

Apache Spark là một hệ thống xử lý dữ liệu phân tán và mã nguồn mở, được thiết kế để xử lý các ứng dụng xử lý dữ liệu lớn và phức tạp. Một Spark Cluster là một tập hợp các máy tính kết nối với nhau để tạo thành một môi trường xử lý phân tán cho việc thực thi các công việc. Spark Cluster chia các tác vụ xử lý dữ liệu thành nhiều phần nhỏ và thực hiện chúng đồng thời trên nhiều máy tính.

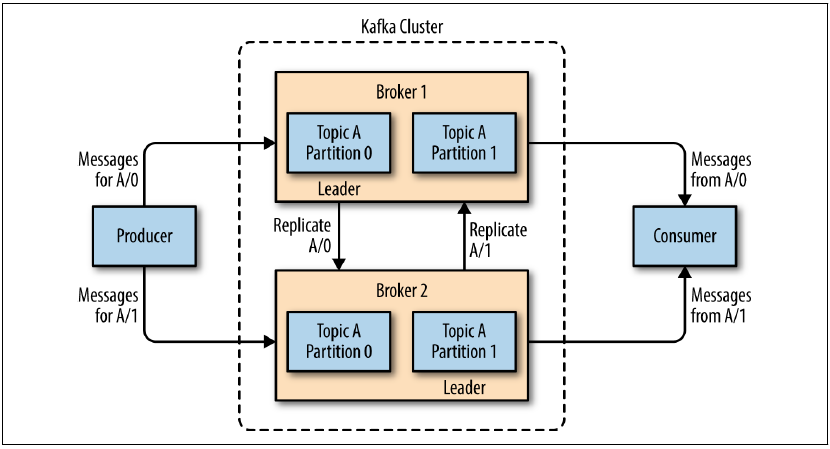


*Hình 3. Spark cluster*

Kiến trúc của Spark bao gồm hai thành phần chính: trình điều khiển(driver) và trình thực thi (executors). Trình điều khiển dùng để chuyển đổi mã của người dùng thành nhiều tác vụ (tasks) có thể được phân phối trên các nút xử lý (worker nodes). Khi thực thi, trình điều khiển Driver tạo ra 1 SparkContext, sau đó giao tiếp với Cluster Manager để tính toán tài nguyên và phân chia các tác vụ đến cho các worker nodes.

### 2.2.3 Kafka cluster

Kafka là một hệ thống mã nguồn mở được thiết kế để xử lý, lưu trữ và truyền tải dữ liệu theo kiểu tin nhắn (message) giữa các ứng dụng. Được tạo ra bởi Apache Software Foundation, Apache Kafka đã trở thành một phần quan trọng trong các hệ thống phân tán và xử lý dữ liệu lớn.



*Hình 4. Kafka cluster*

Dữ liệu chứng khoán được thu thập và publish vào hệ thống Kafka. Namenodes subcribe vào topic chứa dữ liệu và lấy dữ liệu. Sau đó sử dụng Spark để xử lý dữ liệu một cách hiệu quả trước khi đẩy vào Hadoop.

# CHƯƠNG 3. NHỮNG TRẢI NGHIỆM KHI XÂY DỰNG HỆ THỐNG

## 3.1 Lưu trữ dữ liệu vào HDFS

### Trải nghiệm 1: Lưu trữ dữ liệu phân tán

Hệ thống có 2 Live Nodes và 0 Dead Nodes, chứng tỏ dữ liệu đã được lưu phân tán trên 2 máy là datanode1 và datanode2.

A screenshot of a computer

Description automatically generated

*Hình 5. Lưu trữ dữ liệu phân tán vào 2 máy (datanode1 và datanode2)*

### Trải nghiệm 2: Nhân bản dữ liệu để tăng tính chịu lỗi

Trong hệ thống nhận thấy Replication là 2, chứng tỏ dữ liệu đã được sao lưu trên 2 máy datanode1 và datanode2. Mỗi máy sẽ lưu toàn bộ dữ liệu để tăng tính chịu lỗi trong hệ thống.

A screenshot of a computer

Description automatically generated

*Hình 6. Nhân bản dữ liệu*

### Trải nghiệm 3: Thử loại bỏ các node để chứng minh khả năng chịu lỗi của hệ thống

Khi tắt một Node nhận thấy trong ảnh dưới LiveNodes là 1, hệ thống vẫn hoạt động được vì dữ liệu được sao lưu ở cả hai node, khi tắt một node dữ liệu vẫn được lưu ở node còn lại.

A screenshot of a computer

Description automatically generated

*Hình 7. Hệ thống sau khi shutdown một datanode*

Khi tắt cả hai datanode nhận thấy LiveNodes là 0, chúng ta sẽ không thể truy xuất được dữ liệu nữa

A screenshot of a computer

Description automatically generated

*Hình 7. Hệ thống sau khi shutdown cả hai datanode*

Khi muốn truy xuất dữ liệu, không thể truy cập vì đã shutdown cả 2 datanodes

*A screenshot of a computer screen

Description automatically generated*

*Hình 8. Không thể xem dữ liệu khi shutdown cả hai datanode*

## 3.2 Xử lý dữ liệu bằng spark

### Trải nghiệm 4: Dùng spark để xử lý dữ liệu

Dùng Spark để xử lý dữ liệu, việc xử lý dữ liệu được chia cho 2 worker là datanode1 và datanode2, việc điều khiển sẽ do masternode đảm nhiệm.

A screenshot of a computer

Description automatically generated

*Hình 9. Spark được khởi chạy*

### Trải nghiệm 5: Thử hoạt động của spark sau khi tắt một workernode

MasterNode sẽ chia các job cho 2 WorkerNodes cùng xử lý, khi tắt đi một WorkerNode thì WorkerNode sẽ phải thực hiện tất cả các công việc, dẫn điến tốc độ xử lý sẽ chậm hơn, thời gian hoàn thành công việc sẽ lâu hơn.

A screenshot of a computer

Description automatically generated

*Hình 10. Sau khi tắt một WorkerNode*

## 3.3 Truy xuất dữ liệu

### Trải nghiệm 6: Truy vấn dữ liệu bằng Pyspark

Sử dụng thư viện PySpark trong ngôn ngữ lập trình Python để thực hiện các câu truy vấn và xử lý dữ liệu trên môi trường Apache Spark

A computer screen shot of a computer

Description automatically generated

*Hình 10. Truy vấn dữ liệu bằng Pyspark*

### Trải nghiệm 7: Lấy dữ liệu và hiển thị

- Những công ty có mã cổ phiếu tăng mạnh nhất

A black screen with white text

Description automatically generated

*Hình 12. Truy xuất những công ty có mã cổ phiếu tăng mạnh nhất*

- Những công ty có mã cổ phiếu có giá cao nhất

A black and white screen with white text

Description automatically generated

*Hình 13. Truy xuất những công ty có mã cổ phiếu có giá cao nhất*

# CHƯƠNG 4 KẾT LUẬN

Nhóm đã vận dụng những kiến thức cơ bản đã được học để xây dựng được một hệ thống lưu trữ và xử lý dữ liệu lớn về chứng khoán. Từ hệ thống đó cho thấy được những ưu điểm mà một hệ thống BigData mang lại như khả năng lưu trữ, biểu diễn lượng dữ liệu lớn mà một số hệ thống khác không có. Việc xây dựng hệ thống giúp các thành viên trong nhóm hiểu rõ hơn về các kiến thức đã được học.