

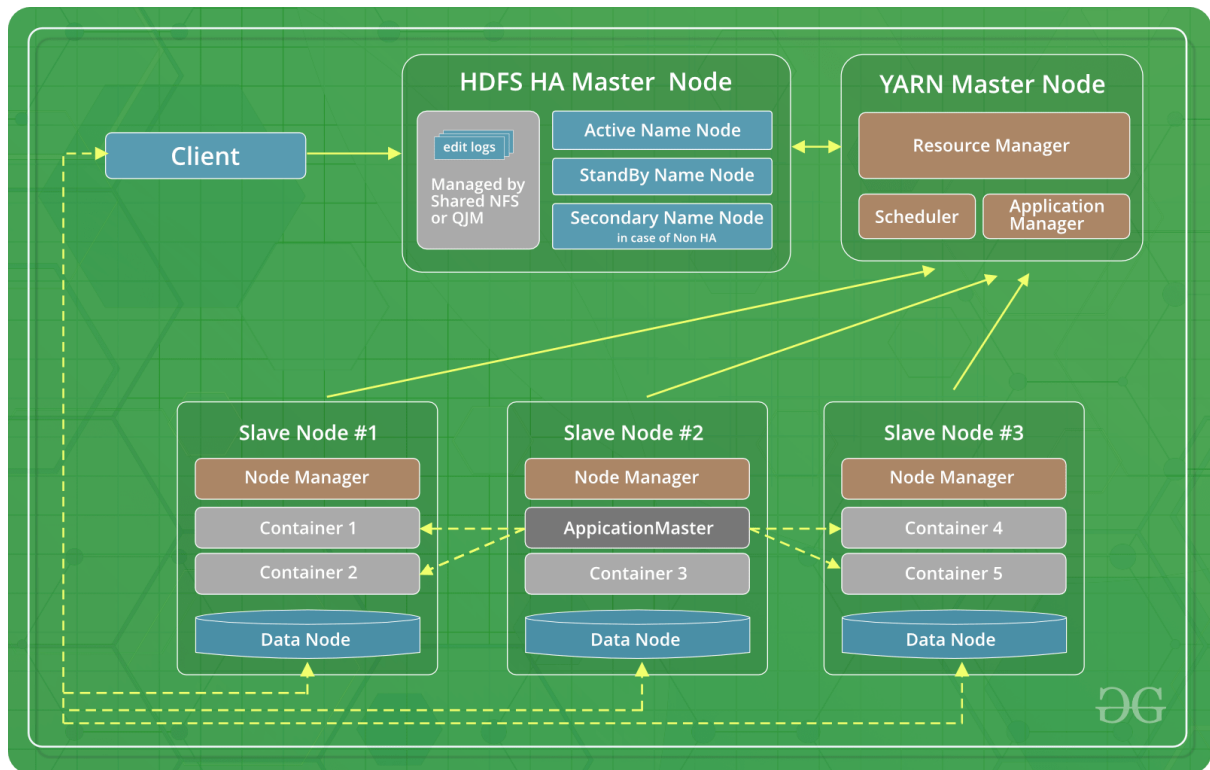


Kiến trúc Hadoop

Các vấn đề Hadoop giải quyết

Các chức năng dưới đây sẽ là xương sống để hiểu hướng xây dựng các kiến trúc thành phần của Hadoop

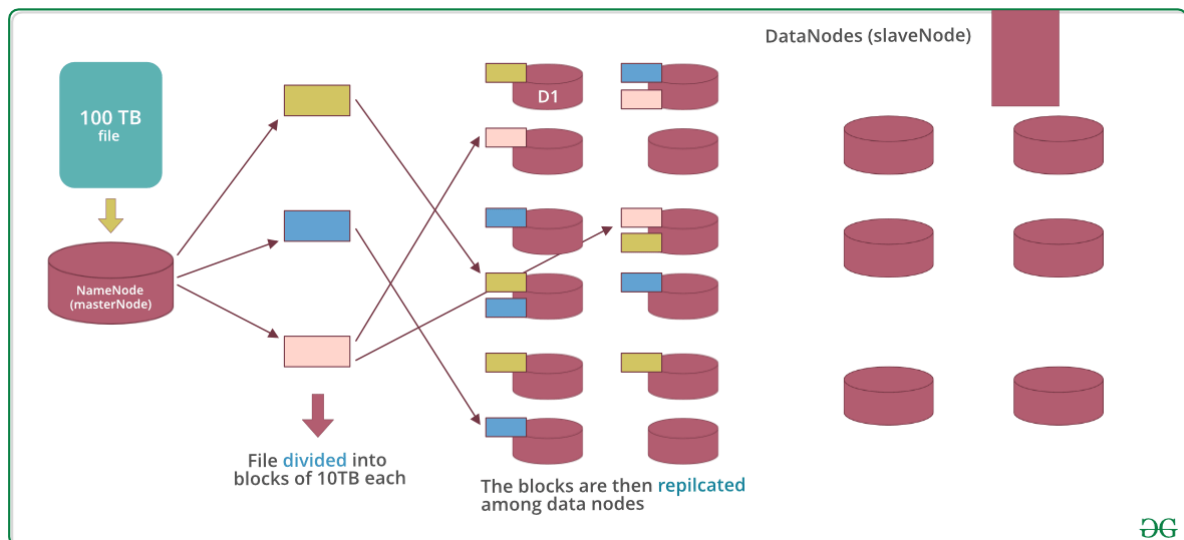
1. Xử lý lượng dữ liệu khổng lồ (Tùy định nghĩa có thể là Terabyte, Petabyte,...)
2. Xử lý môi trường phân tán, dữ liệu ở nhiều phần cứng khác nhau, yêu cầu xử lý đồng bộ
3. Các lỗi xuất hiện thường xuyên
4. Bảng thông giữa các phần cứng vật lý chứa dữ liệu phân tán có giới hạn



Thành phần

HDFS

Định nghĩa: Hệ thống tệp phân tán Hadoop là một hệ thống tệp chuyên dụng để lưu trữ dữ liệu lớn với một nhóm phần cứng hàng hóa hoặc phần cứng rẻ hơn với mô hình truy cập trực tuyến. Nó cho phép dữ liệu được lưu trữ tại nhiều nút trong cụm, đảm bảo bảo mật dữ liệu và khả năng chịu lỗi.



HDFS sử dụng kiến trúc master/slave, trong đó master gồm một NameNode để quản lý hệ thống file metadata và một hay nhiều slave DataNodes để lưu trữ dữ liệu thực tại.

1. NameNode(MasterNode):

- Quản lý các Slave và phân công công việc cho chúng
- Thực hiện các hoạt động như đóng, mở, đổi tên tệp và thư mục
- Phải được triển khai trên phần cứng đáng tin cậy
- Lưu trữ metadata như đường dẫn, số lượng blocks, block id,...
- Yêu cầu dung lượng RAM lớn
- Lưu trong RAM để truy xuất nhanh, có một bản sao lưu trên ổ đĩa(Secondary NameNode).

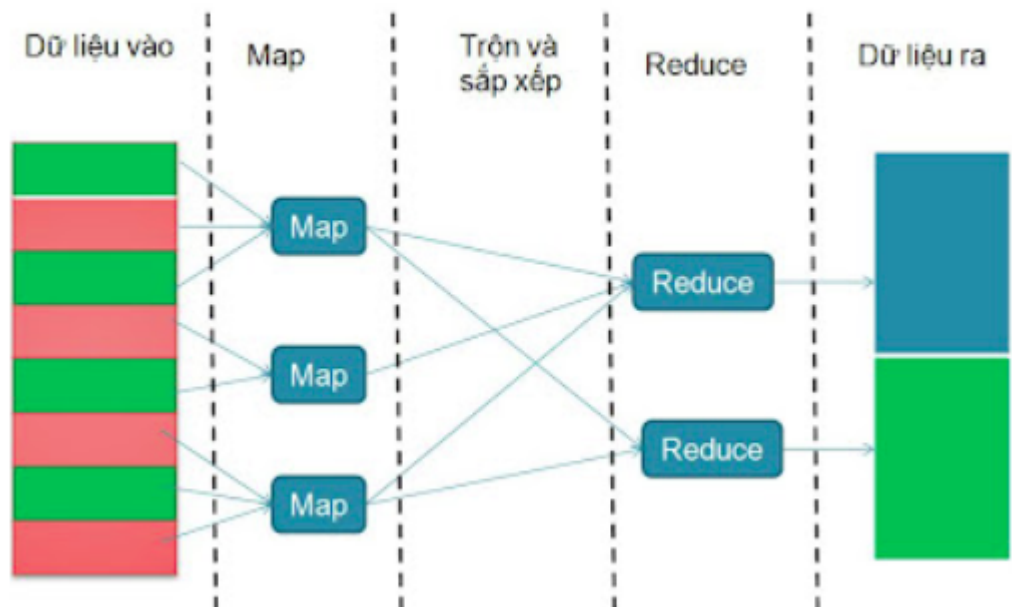
2. DataNode(SlaveNode):

- Các node làm việc, làm các công việc thực tế như đọc, viết, xử lý,...
- Chúng thực hiện tạo, xóa và sao chép theo hướng dẫn từ Master.
- Có thể thực hiện trên các phần cứng thông thường.
- Cần bộ nhớ đĩa lớn vì dữ liệu thực được lưu trữ ở đây

MapReduce

Định nghĩa: Khi người dùng cần thao tác với dữ liệu đã được lưu trữ phân tán, cần có cơ chế xác định hay ánh xạ (Map) và tổng hợp (Reduce) tất chúng lại rồi trả về cho người dùng.

Các giai đoạn



1. Dữ liệu đầu vào

Hadoop chấp nhận dữ liệu ở nhiều định dạng khác nhau và lưu trữ nó trong HDFS. Dữ liệu đầu vào này được thực hiện bởi nhiều Map tasks.

2. Map

Map đọc dữ liệu, xử lý và tạo các cặp khóa-giá trị. Số lượng Map tasks phụ thuộc vào tệp đầu vào và định dạng của nó.

Thông thường, một tệp trong cụm Hadoop được chia thành các blocks, mỗi blocks có kích thước mặc định là 128 MB. Tùy thuộc vào kích thước, tệp đầu vào được chia thành nhiều phần. Một nhiệm vụ bản đồ sau đó sẽ chạy cho từng đoạn. Lớp ánh xạ có các hàm ánh xạ quyết định thao tác nào sẽ được thực hiện trên mỗi đoạn.

3. Trộn và sắp xếp

- Trộn: Giai đoạn này kết hợp tất cả các giá trị được liên kết với một khóa giống hệt nhau.
- Sắp xếp: Sau khi xáo trộn được thực hiện, đầu ra được gửi đến giai đoạn sắp xếp nơi tất cả các cặp (khóa, giá trị) được sắp xếp tự động.

4. Reduce

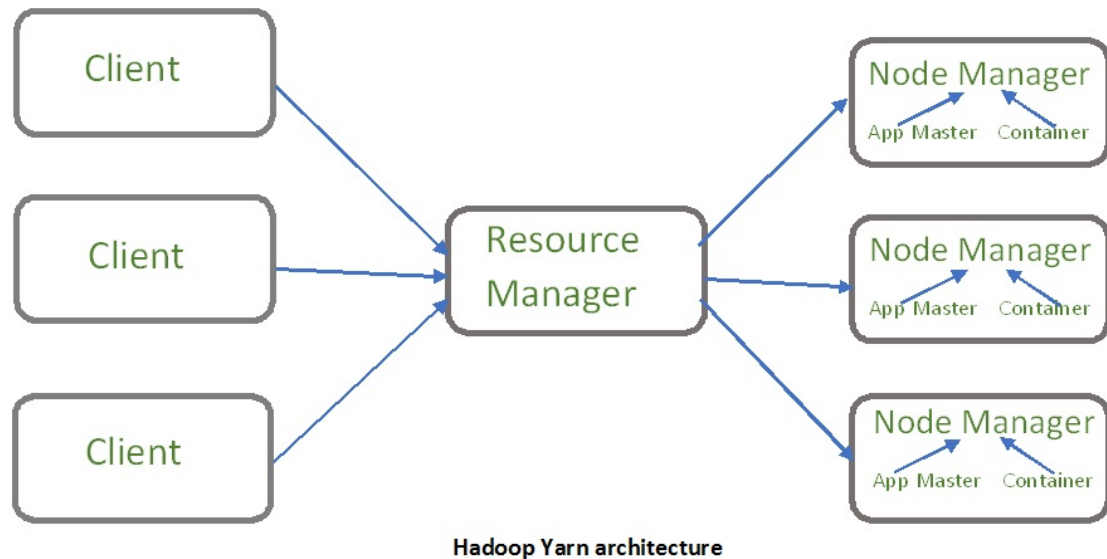
Trong giai đoạn Reduce, một lớp giảm thiểu thực hiện các hoạt động trên dữ liệu được tạo ra từ các Map tasks thông qua một chức năng Reduce. Nó xáo trộn, sắp xếp và tổng hợp các cặp khóa-giá trị trung gian (bộ giá trị) thành một tập hợp các bộ giá trị nhỏ hơn.

5. Dữ liệu ra

Tập hợp các bộ giá trị nhỏ hơn là đầu ra cuối cùng và được lưu trữ trong HDFS.

YARN

là một framework hỗ trợ phát triển ứng dụng phân tán. YARN cung cấp daemons và APIs cần thiết cho việc phát triển ứng dụng phân tán, đồng thời xử lý và lập lịch sử dụng tài nguyên tính toán (CPU hay memory) cũng như giám sát quá trình thực thi các ứng dụng đó. YARN tổng quát hơn MapReduce thế hệ đầu tiên (gồm mô hình JobTracker / TaskTracker).



- **Client:**

Nó gửi các công việc Map-Reduce.

- **Trình quản lý tài nguyên:**

Nó là daemon chính của YARN và chịu trách nhiệm phân công và quản lý tài nguyên giữa tất cả các ứng dụng. Bất cứ khi nào nó nhận được một yêu cầu xử lý, nó sẽ chuyển tiếp nó đến người quản lý nút tương ứng và phân bổ tài nguyên để hoàn thành yêu cầu đó. Nó có hai thành phần chính:

- **Bộ lập lịch:**

Nó thực hiện việc lập lịch dựa trên ứng dụng được cấp phát và các tài nguyên có sẵn. Nó là một bộ lập lịch thuần túy, có nghĩa là nó không thực hiện các tác vụ khác như giám sát hoặc theo dõi và không đảm bảo khởi động lại nếu một tác vụ không thành công. Bộ lập lịch YARN hỗ trợ các plugin như Bộ lập lịch công suất và Bộ lập lịch công bằng để phân vùng tài nguyên cụm.

- **Người quản lý ứng dụng:**

Có trách nhiệm chấp nhận ứng dụng và thương lượng vùng chứa đầu tiên từ người quản lý tài nguyên. Nó cũng khởi động lại vùng chứa Application Master nếu một tác vụ không thành công.

- **Trình quản lý nút:**

Nó quản lý từng nút riêng lẻ trên cụm Hadoop và quản lý ứng dụng và quy trình làm việc cũng như nút cụ thể đó. Công việc chính của nó là theo kịp với Người quản lý tài nguyên. Nó đăng ký với Trình quản lý tài nguyên và gửi nhịp tim với tình trạng sức khỏe của nút. Nó giám sát việc sử dụng tài nguyên, thực hiện quản lý nhật ký và cũng giết một vùng chứa dựa trên chỉ dẫn từ trình quản lý tài nguyên. Nó cũng chịu trách nhiệm tạo tiến trình vùng chứa và bắt đầu nó theo yêu cầu của Application master.

- **Ứng dụng Master:**

Đơn xin việc là một công việc đơn lẻ được nộp vào một khuôn khổ. Ứng dụng chủ chịu trách nhiệm thương lượng tài nguyên với người quản lý tài nguyên, theo dõi trạng thái và giám sát tiến độ của một ứng dụng duy nhất. Ứng dụng chủ yêu cầu vùng chứa từ trình quản lý nút bằng cách gửi Bối cảnh khởi chạy vùng chứa (CLC) bao gồm mọi thứ mà ứng dụng cần để chạy. Khi ứng dụng được khởi động, nó sẽ gửi báo cáo tình trạng đến người quản lý tài nguyên theo thời gian.

- **Vùng chứa:**

Nó là một tập hợp các tài nguyên vật lý như RAM, lõi CPU và đĩa trên một nút duy nhất. Các vùng chứa được gọi bởi Ngủ cảnh khởi chạy vùng chứa (CLC) là một bản ghi có chứa thông tin như biến môi trường, mã thông báo bảo mật, phụ thuộc, v.v.