생산자

thesoul214
토픽 8 / 봤어요 6

하둡(Hadoop) 소개 및 기본 구성요소 설명

2017-07-31 21:16:09

하둡(Hadoop) 소개 및 기본 구성요소 설명, HDFS, MapReduce

하둡이란?

분산 환경에서 빅 데이터를 저장하고 처리할 수 있는 자바 기반의 오픈 소스 프레임 워크.

구성요소

1. 하둡 분산형 파일시스템(Hadoop Distributed FileSystem, HDFS)

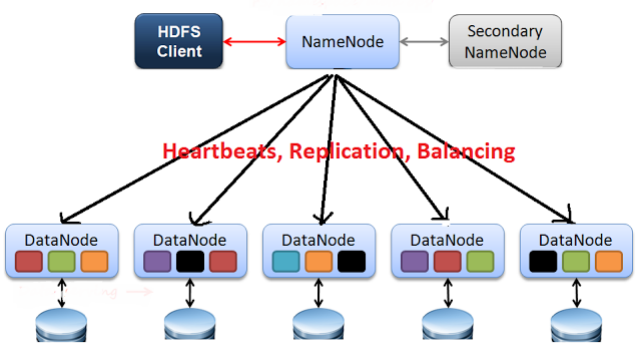
하둡 네트워크에 연결된 기기에 데이터를 저장하는 분산형 파일시스템

특징 :

1. HDFS는 데이터를 저장하면, 다수의 노드에 복제 데이터도 함께 저장해서 데이터 유실을 방지
2. HDFS에 파일을 저장하거나, 저장된 파일을 조회하려면 스트리밍 방식으로 데이터에 접근해야 함.
3. 한번 저장한 데이터는 수정할 수 없고, 읽기만 가능하게 해서 데이터 무결성을 유지.

   (2.0 알파버전부터는 저장된 파일에 append가 가능하게 됨)
4. 데이터 수정은 불가능 하지만 파일이동, 삭제, 복사할 수 있는 인터페이스를 제공함.

아키텍처 :

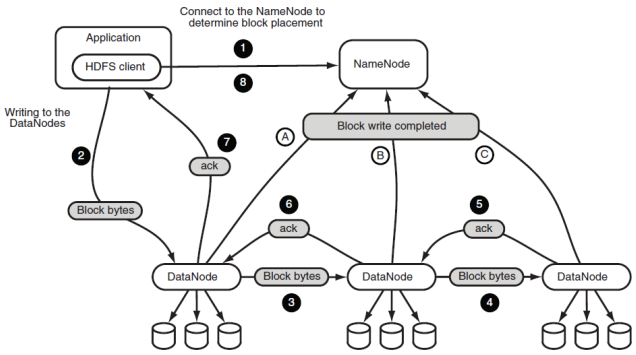


1. 블록 구조의 파일 시스템으로, 저장하는 파일은 특정 사이즈의 블록으로 나눠져 분산된 서

2. 하나의 블록은 3개(수정 가능)로 복제되며, 각각 다른 HDFS의 노드에 분산저장됨

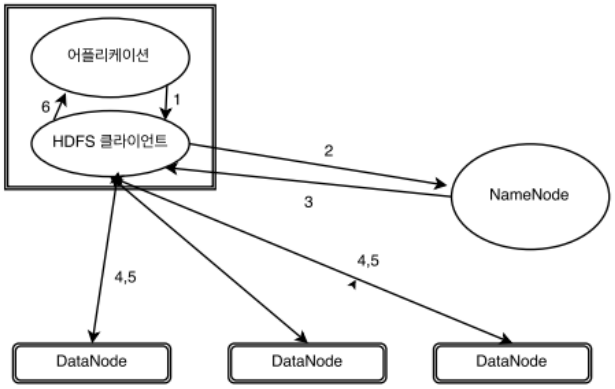
3. HDFS에는 마스터 역할을 하는 네임노드 서버 한 대와, 슬레이브 역할을 하는 데이터노드 서버가 여러 대로 구성된다.
4. 네임 노드는 HDFS의 모든 메타데이터(블록들이 저장되는 디렉토리의 이름, 파일명등..)를 관리하고, 클라이언트가 이를 이용하여 HDFS에 저장된 파일에 접근할 수 있음.
5. 하둡 어플리케이션은 HDFS에 파일을 저장하거나, 저장된 파일을 읽기 위해 HDFS 클라이언트를 사용하며, 클라이언트는 API형태로 사용자에게 제공됨.
6. 데이터 노드는 주기적으로 네임노드에서 블록 리포트(노드에 저장되어 있는 블록의 정보)를 전송하고 이를 통해 네임노드는 데이터 노드가 정상 동작하는지 확인.
7. 클라이언트는 네임노드에 접속해서 원하는 파일이 저장된 블록의 위치를 확인하고, 해당 블록이 저장된 데이터 노드에서 직접 데이터를 조회함.

파일 저장 플로우



1. 어플리케이션이 HDFS 클라이언트에게 파일 저장을 요청하면,
   HDFS 클라이언트는 네임노드에게 파일 블록들이 저장될 경로 생성을 요청.
   네임노드는 해당 파일 경로가 존재하지 않으면 경로를 생성한 후,
   다른 클라이언트가 해당 경로를 수정하지 못하도록 락을 검.
   그 후, 네임노드는 클라이언트에게 해당 파일 블록들을 저장할 데이터노드의 목록을 반환
2. 클라이언트는 첫 번째 데이터 노드에게 데이터를 전송
3. 첫 번째 데이터 노드는 데이터를 로컬에 저장한 후, 데이터를 두 번째 데이터 노드로 전송
4. 두 번째 데이터 노드는 데이터를 로컬에 저장한 후, 데이터를 세 번째 데이터 노드로 전송
5, 6. 로컬에 데이터를 저장하였으면 자기에게 데이터를 넘겨준 데이터 노드에게,
   데이터의 로컬 저장이 완료 되었음을 응답
7. 첫 번째 데이터 노드는 클라이언트에게 파일 저장이 완료 되었음을 응답.

파일 읽기 플로우



1. 어플리케이션이 클라이언트에게 파일 읽기를 요청
2. 클라이언트는 네임노드에게 요청된 파일이 어떤 블록에 저장되어 있는지 정보를 요청
3. 메타데이터를 통해 파일이 저장된 블록 리스트를 반환
4. 클라이언트는 데이터 노드에 접근하여 블록 조회 요청
5. 데이터 노드는 클라이언트에게 요청된 블록을 전송
6. 클라이언트를 어플리케이션에 데이터를 전달

2. 맵리듀스(MapReduce)

대용량의 데이터 처리를 위한 분산 프로그래밍 모델, 소프트웨어 프레임워크

맵리듀스 프레임워크를 이용하면 대규모 분산 컴퓨팅 환경에서, 대량의 데이터를 병렬로 분석 가능

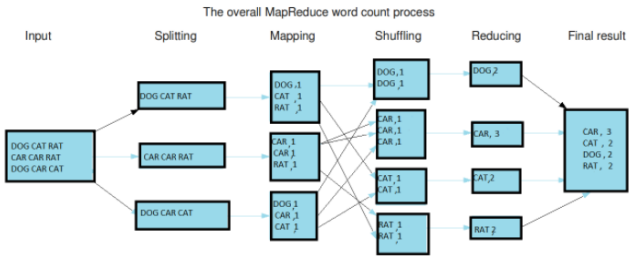
프로그래머가 직접 작성하는 맵과 리듀스 라는 두 개의 메소드로 구성

맵(Map)

흩어져 있는 데이터를 연관성 있는 데이터들로 분류하는 작업(key, value의 형태)

리듀스(Reduce)

Map에서 출력된 데이터를 중복 데이터를 제거하고 원하는 데이터를 추출하는 작업.



위의 프로세스는, 문자열 데이터에 포함된 단어의 빈도수를 출력해주는 과정

1. Splitting : 문자열 데이터를 라인별로 나눈다.
2. Mapping : 라인별로 문자열을 입력받아, <key, value> 형태로 출력.
3. Shuffling : 같은 key를 가지는 데이터끼리 분류.
4. Reducing : 각 key 별로 빈도수를 합산해서 출력.
5. Final Result : 리듀스 메소드의 출력 데이터를 합쳐서 하둡 파일 시스템에 저장.

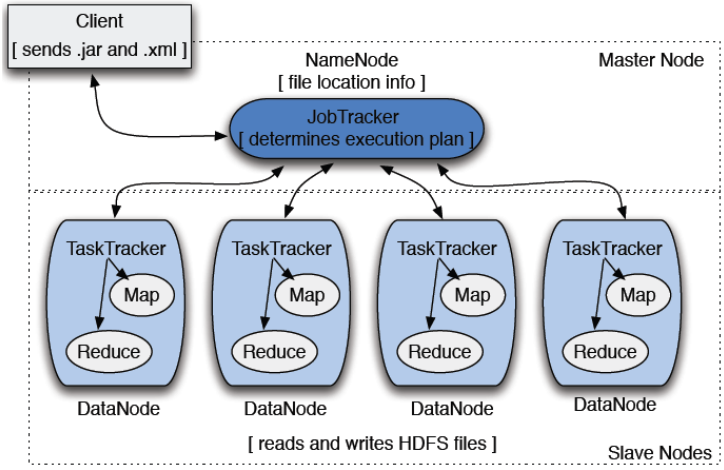
맵 리듀스의 잡(Job)

Client가 수행하려는 작업단위(입력데이터, 맵리듀스 프로그램, 설정 정보로 구성)

맵 리듀스 시스템 구성

맵 리듀스 시스템은 Client, JobTracker, TaskTracker 로 구성된다.

JobTracker는 NameNode에, TaskTracker는 DataNode에 위치한다.



Client : 분석하고자 하는 데이터를 잡의 형태로 JobTracker에게 전달

JobTracker : 하둡 클러스터에 등록된 전체 job을 스케줄링하고 모니터링

TaskTracker : DataNode에서 실행되는 데몬이고, 사용자가 설정한 맵리듀스 프로그램을 실행하며,

JobTracker로부터 작업을 요청받고 요청받은 맵과 리듀스 개수만큼

맵 태스크와 리듀스 태스크를 생성