1강

목표

1. 병렬분산시스템을 구현하자
2. 맵 리듀스 프레임워크를 이해하자
3. 하둡을 사용해보자

병렬분산시스템이란?

Scale-out VS Scale-up

- Scale-out : 싸고 많은 서버를 사용하는 것
- Scale-up: 비싸지만 적은 서버를 이용하는 것
- 데이터 중심 분야에서는 Scale- out을 선호한다
    - 왜냐하면, 비싼 서버 하나 쓰는 것보다 싼 서버 여러개 쓰는 것이 대체로가격이 싸기 때문이다.

MapReduce?

- 한대의 컴퓨터로 데이터처리가 어렵기에 수십, 수백대를 합쳐서 데이터 처리를 한다.
- MapReduce는 이런 빅데이터 묶음 처리를 효율적으로 하게하는 프로그래밍 모델이다.
- 사용하게 된다면, 데티어 관리가 쉽고, 효율적이며, 컴퓨터 증설시 자동 확장처리가 됩니다.
- scalable한 특성(사용자나 데이터가 급증해도 프로그램의 성능이 멈추거나 떨어지지 않음)을 가집니다.
- 대표적인 오픈소스 중 하둡이 존재합니다

Function Programming Model

- Main 함수, Map 함수, Reduce 함수 로 구성됨
    - Map 함수와 Reduce 함수는 인풋 아웃풋이 (key,value)이다.
- 각 레코드나 튜플은 (KEY,VALUE) 쌍으로 표현됨
- 메인 함수를 한 개의 마스터 머신으로 수행
    - 맵 함수의 수행 전 전처리 또는 리듀스 함수의 결과 후처리로 사용할 수 있음
- 메인 함수에서 맵리듀스 페이즈를 한번 또는 여러번 수행이 가능하다
- 맵리듀스페이즈를 진행하다가 맵함수-컴바인함수-리듀스함수를 수행시킬수도 있다.

Map reduce phase

MAP phase

- 제일 먼저 수행되며, 여러 파티션에 병렬 분산 호출되어 수행됨
- 각 머신의 Mapper가 입력 데이터 한 줄마다 맵 함수 호출
- 컴퓨터만 보면 한 줄씩 순차적으로 실행되지만, 전체적으로보면 수백대의 컴퓨터가 동시처리하면서 병렬적처리가 된다.
- 각 map 함수는 (key,value)로 이루어진 결과를 여러 머신으로 분산해서 보내진다

Shuffling phase

- 모든 머신에서 맵페이즈가 끝나면 실행된다.
- key를 이용하여 정렬(sorting)하고 같은 키를 가진 값끼리는 (key,[value-list])로 바꿔서 보내줍니다

Reduce phase

- 받은 (key,[value-list])쌍에 reduce 처리를 해서 원하는 (key,[value-list])쌍을 추출한다

Hadoop

Hadoop 이 하는일

- 빅데이터 파일을 조각내서 여러 컴퓨터에 저장
- fault tolerance(:시스템 구성 부품 일부가 파손 또는 고장되어 정상기능을 못하는 것)를 막기 위해서 같은 조각 파일도 여러 머신의 저장함
- 빅데이터를 수천 대의 컴퓨터에 병렬처리시킴

hadoop 구성요소

- MapReduce - 소프트웨어의 수행을 분산
- HDFS(Hadoop Distributed File System) - 데이터를 분산
- Namenode : 파일시스템을 관리하고 클라이언트가 파일에 접근할 수 있게함, 하나
- Datanode: 컴퓨터에 들어있는 데이터를 접근할 수 있게 함, 여러개

map-reduce 함수

- map함수
  - 라인단위로 실행되며, Mapper 클래스를 상속받아서 수정하여 사용
  - key: 입력 텍스트에 맨 앞문자를 기준으로 해당 라인의 첫문자까지 오프셋
  - value: 해당 라인 텍스트 전체
- reduce함수
  - Reducer 클래스를 상속받아서 수정하여 사용
  - 원하는 값으로 가공
- combine 함수
  - map 함수에서 처리된 데이터를 셔플링페이즈 전에 미리 데이터사이즈를 줄이는 역활(셔플링에서 하기때문에 선택사항)

- setup 함수
    - 첫 map 함수나 reduceㅎ마수가 호출되기전 가장 먼저 수행하여 자료구조 초기화나 피라미터 정보를 main에서 받아올때 사용
- cleanup 함수
    - 마지막 map이나 reduce 함수 이후 자료구조의 결과를 출력