Chapter01. 하둡 살펴보기

1.2 하둡이란?

- 대용량 데이터를 분산 처리할 수 있는 자바 기반의 오픈소스 프레임워크
- 하둡은 분산 파일 시스템인 HDFS(Hadoop Distributed File System)에 데이터를 저장하고, 분산 처리 시스템인 맵리듀스를 이용해 데이터를 처리

1.2.1 왜 하둡인가?

- 하둡의 저렴한 구축 비용과 비용 대비 빠른 데이터 처리, 그리고 장애를 대비한 특성
    - 웹 로그 같은 비정형 데이터를 RDBMS에 저장하기에는 데이터 크기가 너무 큼.
    - 하둡은 오픈소스 프로젝트이기 때문에 라이센스 비용에 대한 부담이 없음
    - 데이터의 복제본을 저장하기 때문에 데이터의 유실이나 장애가 발생했을 때도 복구가 가능
    - 분산 컴퓨팅을 통해 데이터 처리 속도가 굉장히 빠름

1.3 하둡 에코시스템

- 하둡은 비즈니스에 효율적으로 적용할 수 있게 다양한 서브 프로젝트를 제공. 이러한 서브 프로젝트가 상용화되면서 하둡 에코시스템이 구성
    - 하둡 코어 프로젝트 : HDFS(분산 데이터를 저장), 맵리듀스(분산 데이터를 처리)
    - 하둡 서브 프로젝트
        - YARN : 데이터 처리 작업을 실행하기 위한 클러스터 자원(CPU, 메모리, 디스크 등)과 스케줄링을 위한 프레임워크. 기존 하둡의 데이터 처리 프레임워크인 맵리듀스의 단점을 극복하기 위해 시작된 프로젝트. 하둡 2.0부터 사용 가능

        - Spark : 인메모리 기반의 범용 데이터 처리 플랫폼. 배치 처리, 머신러닝, SQL 질의 처리, 스트리밍 데이터 처리, 그래프 라이브러리 처리와 같은 다양한 작업을 수용할 수 있도록 설계

1.4 하둡에 대한 진실

- RDBMS와 상호보완적인 특성을 띄고 있음
    - BI(Business Intelligence)나 OLAP(On-line Analytical Processing) 시스템을 사용하는 기업은 분석을 위한 데이터를 처리하기 위해 ETL(Extraction, Transformation, Loading) 과정을 거치게 된다. ETL 과정을 거친 데이터는 DW(Data Warehouse)나 DM(Data Mart)에 전송과 적재하는데, 하둡은 이러한 ETL과정에 도움을 줄 수 있다.
    - 하둡은 무결성을 보하지 않고, 실시간 처리가 힘들다. 하둡은 배치성으로 데이터를 저장하거나 처리하는 데 적합한 시스템으로 구성되어 있기 때문이다. 따라서 온라인 쇼핑몰에서 제품을 구매할 때 생성되는 데이터와 인터넷 뱅킹에서 자금을 이체할 때 생성되는 데이터와 같은 데이터 처리에 적합하지 않다.
- 하둡은 NoSQL이 아니다. 하둡이 RDBMS에 속하는 것은 아니지만 NoSQL의 핵심 기능인 데이터 베이스 시스템의 역할을 수행하는 것은 아니다.
    - NoSQL이란?
        - NoSQL은 RDBMS가 분산환경에 적합하지 않기 때문에 고안된 데이터베이스 시스템. NoSQL의 데이터베이스는 단순히 키와 쌍으로만 이뤄져 있고, 인덱스와 데이터가 분리되어 별도로 운영된다. 또한 조인이 없고, RDBMS에서는 여러 행으로 존재하던 데이터를 하나의 집합된 형태로 저장한다. 또한 샤딩(Sharding)이라는 기능이 있어서 데이터를 분할해 다른 서버에 나누어 저장한다.
        - 완벽한 데이터 무결성과 정합성을 제공하지 않기 때문에, 핵심 데이터는 RDBMS를 이용하고, 데이터를 보관하고 처리해야 하는 경우 NoSQL을 이용하면 된다.

1.5 하둡의 문제점

- 고가용성 지원 문제
    - 가용성(Availability) : 시스템 장애 발생 후 정상으로 돌아오는 상태를 분석하는 척도
    - 고가용성 : 99.999%상태의 가용. 이는 일년 중 30분 정도를 제외하고 서비스가 가능한 수치

    - 하둡의 HDFS 네임노드와 데이터노드로 구성되는데, 네임노드가 HDFS에 저장하는 모든 데이터의 메타정보를 관리. 만약 네임노드에 장애가 발생하면 데이터를 더는 HDFS에 저장할 수 없고, 네임노드의 데이터가 유실될 경우 기존에 저장한 파일도 조회할 수 없음
    - 네임노드에 대한 고가용성이 지원되지 않았지만, 하둡2 부터는 고가용성을 지원
- 파일 네임스페이스 제한
    - 네임노드가 관리하는 메타 정보는 메모리로 관리되기 때문에 메모리의 용량에 따라 HDFS에 저장하는 파일과 디렉토리 개수가 제한을 받음
- 데이터 수정 불가
    - 한 번 저장한 파일은 더는 수정할 수 없음. HDFS에 파일을 저장하면 파일의 이동이나 이름변경과 같은 작업은 가능하지만 저장된 파일의 내용은 수정할 수 없음. 그래서 파일 읽기나 배치 작업만이 하둡에 적당
    - 하둡 0.21 버전에서는 기존에 저장된 파잉ㄹ에 내용을 붙일 수 있는 append 기능이 제공
- POSIX 명령어 미지원
    - 기존 파일 시스템에서 사용하던 rm, mv 같은 POSIX 형식의 파일 명령어를 이용할 수 없음.
    - 하둡에서 제공하는 별도의 셀 명령어와 API를 이용해 파일을 제어해야 함.
- 전문 업체 부족
    - 오라클이나 MS-SQL 같은 DBMS는 벤더나 다양한 유지보수 업체가 있지만 아직까지 국내에는 하둡과 관련된 다양한 업체가 부족.

1.6 하둡 배포판

- Cloudera
- Hortonworks
- MapR Technologies