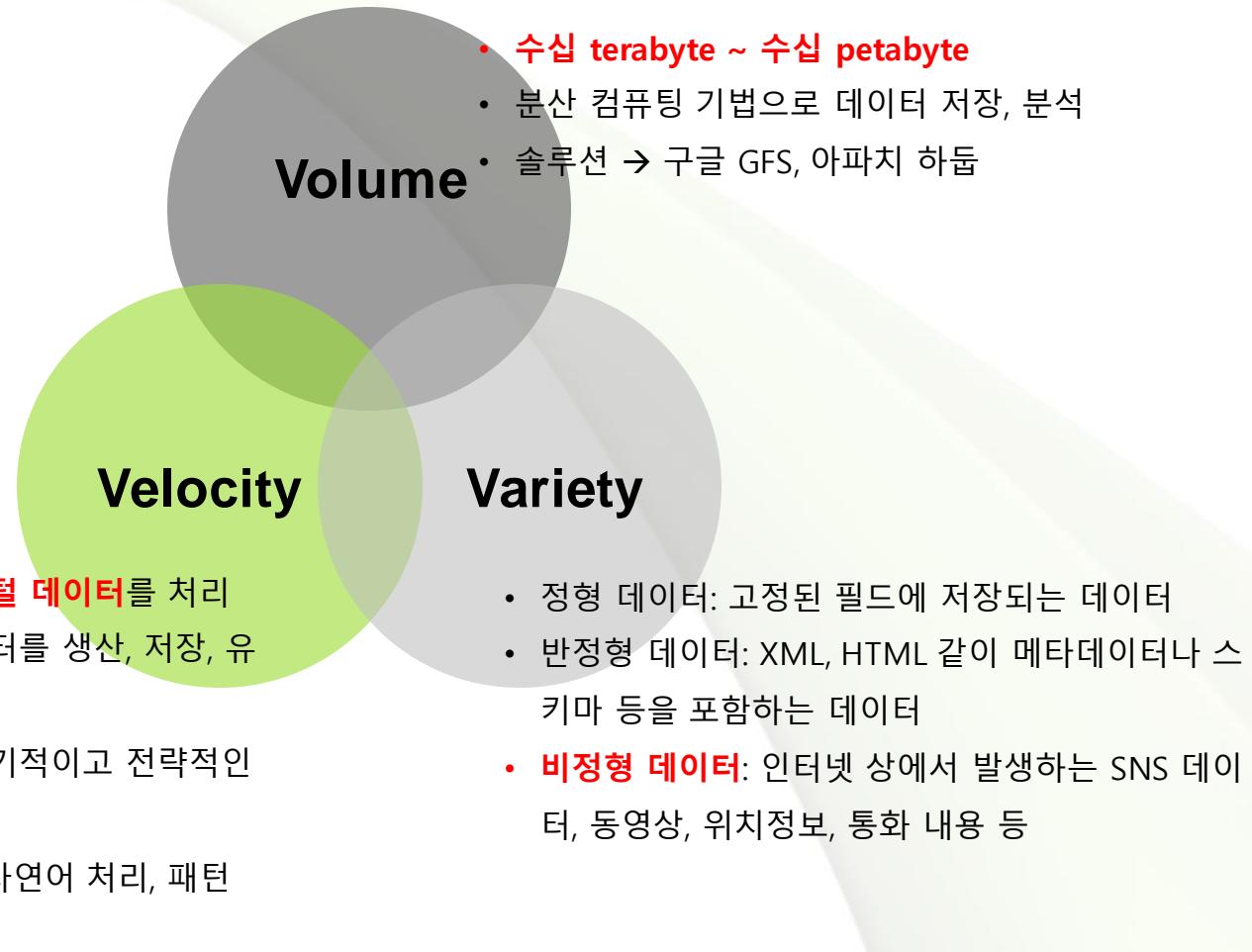


# **Introduction to Hadoop**

# Big Data

- 3대 요소: 크기(Volume), 속도(Velocity), 다양성(Variety)



# Big Data

- 저장 용량 발전과 읽기 및 전송 속도 발전의 불균형
  - » 저장 용량의 발전 속도에 비해 읽기 및 전송 속도의 발전이 느림
  - » 전체 작업에서 읽기 및 쓰기 작업이 차지하는 부하 비중이 높음
- 읽기 및 쓰기 시간을 줄이는 확실한 방법 → 여러 개의 디스크에서 동시에 병렬로 데이터 처리
- 병렬 처리의 문제
  - » 하드웨어 장애 : 하드웨어 수가 많아지면 장애 확률도 증가 → 데이터 손실을 막기 위한 방법 필요
  - » 분할된 데이터 결합 : 병합 과정에서 데이터 정합성 유지 어려움
- 하둡은
  - » 병렬 처리의 문제를 해결하고 안정적이고 확장성 높은 저장 및 분석 플랫폼 제공
  - » 범용 하드웨어에서 실행되는 오픈 소스로 낮은 비용 발생

# Hadoop

- 대용량 데이터를 분산 처리할 수 있는 자바 기반 오픈소스 프레임워크
  - » 분산 파일 시스템인 HDFS(Hadoop Distributed Files System)에 데이터 저장
  - » 분산 처리 시스템인 맵리듀스를 이용해 데이터 처리
- 2005년에 더그 커팅(Doug Cutting)이 구글이 논문으로 발표한 GFS(Google File System)와 MapReduce를 구현한 결과물
- 일괄 질의 처리기
  - » 전체 데이터 셋을 대상으로 비정형 쿼리를 수행하고 합리적인 시간 내에 그 결과를 도출하는 능력 탁월
  - » 상대적으로 대화형 분석에는 적합하지 않음 → 다양한 에코 시스템 구성 기술을 통해 보완
- 데이터의 복제본을 저장하기 때문에 데이터의 유실이나 장애가 발생해도 데이터 복구 가능

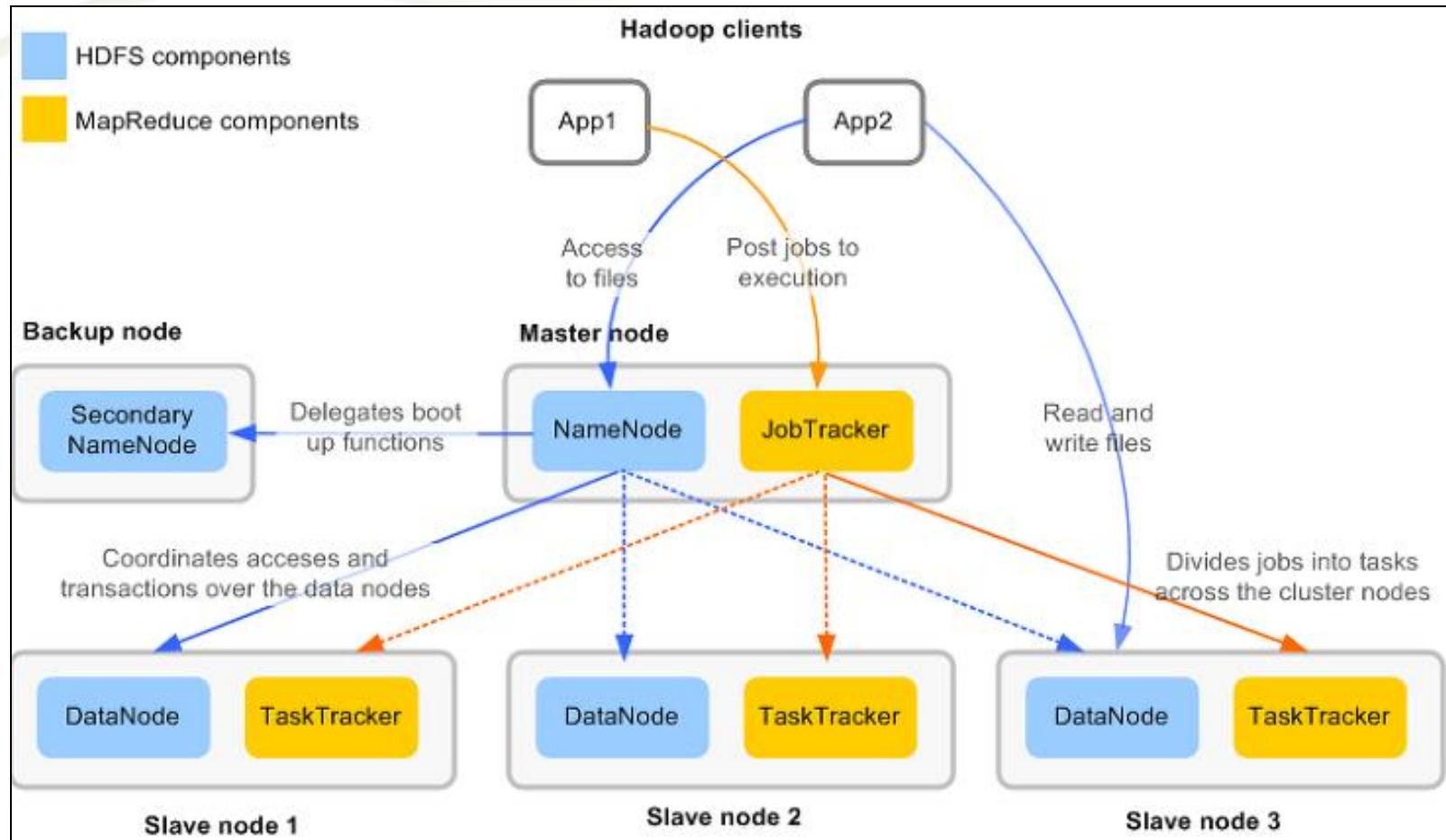
# 하둡과 RDBMS

- 하둡은 기존 RDBMS를 대체하지 않으며 RDBMS와 상호 보완적인 특성을 갖음
- 하둡과 RDBMS 비교

RDBMS	Hadoop
데이터 크기	기가바이트
접근 방식	대화형처리 / 일괄처리
변경	여러 번 읽고 쓰기
트랜잭션	ACID
구조	쓰기 기준 스키마
무결성	높음
확장성	비선형

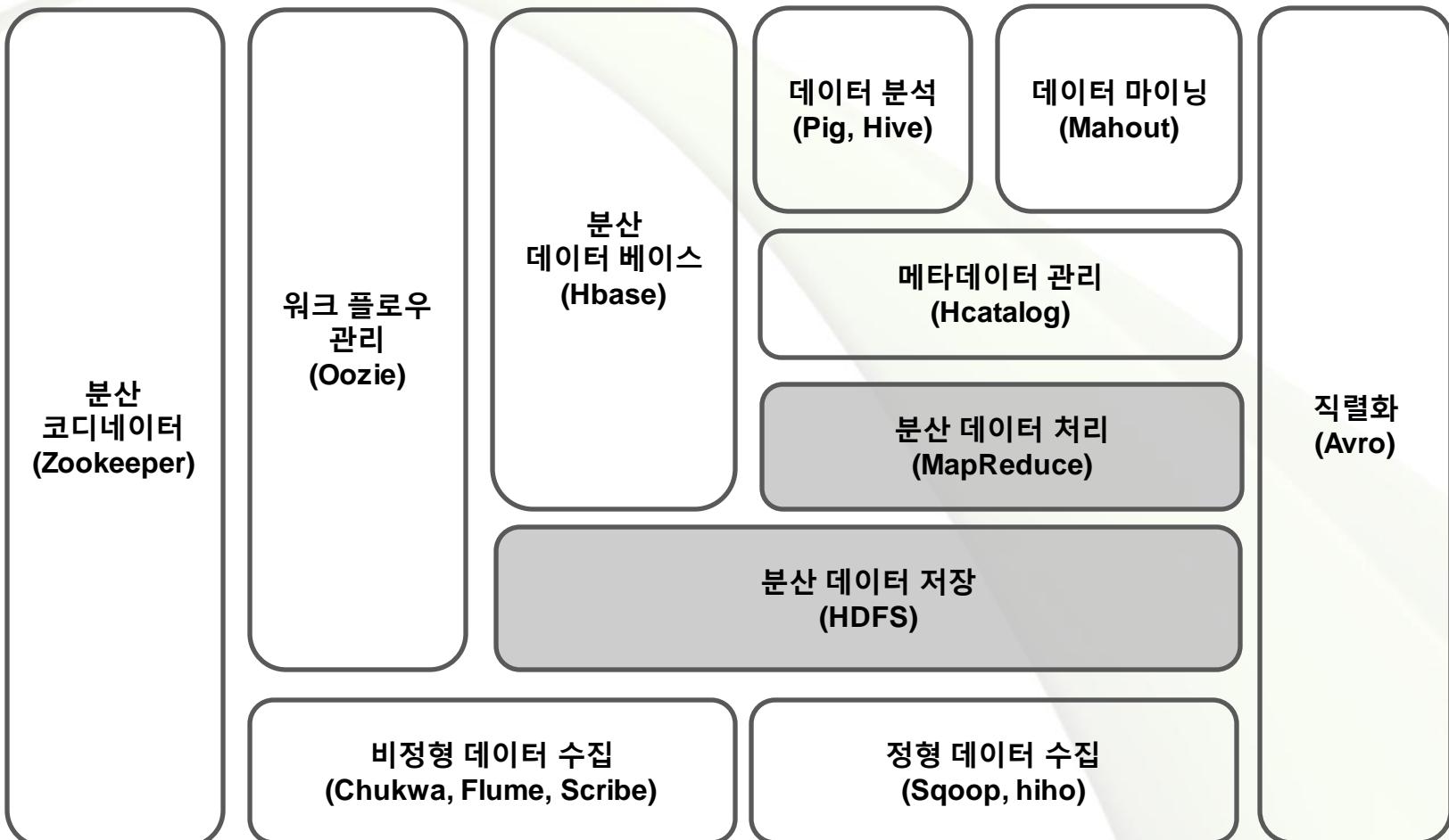
- » 하둡은 비정형 분석과 같이 일괄 처리 방식으로 전체 데이터 처리에 적합
- » RDBMS는 특정 쿼리와 데이터 변경에 적합
- » RDBMS는 정형 데이터 처리에 적합 / 하둡은 비정형, 반정형 처리에 강점

# 하둡의 동작 구조



# Hadoop Ecosystem

- 비즈니스에 효율적으로 적용하기 위한 다양한 서브 프로젝트



# Hadoop Ecosystem

## 분산 코디네이터 (Zookeeper)

### Zookeeper

분산 환경에서 서버 간의 상호 조정이 필요한 다양한 서비스를 제공하는 시스템

1. 하나의 서버에만 서비스가 집중되지 않게 서비스를 알맞게 분산해 동시에 처리
2. 하나의 서버에서 처리한 결과를 다른 서버와도 동기화해서 데이터의 안정성을 보장
3. 운영(active) 서버에 문제가 발생해서 서비스를 제공할 수 없을 경우, 다른 대기 중인 서버를 운영 서버로 바꿔서 서비스를 중지 없이 운영
4. 분산 환경을 구성하는 서버들의 환경 설정을 통합적으로 관리

# Hadoop Ecosystem



## Hbase

- HDFS 기반의 칼럼 기반 데이터베이스
- 실시간 랜덤 조회 및 업데이트 가능
- 각 프로세스는 개인의 데이터를 비동기적으로 업데이트
- 단, 맵리듀스는 일괄 처리 방식으로 수행됨
- 트위터, 야후!, 어도비, 국내 NHN(모바일 메신저 Line)

## Oozie

- 하둡 작업을 관리하는 워크플로우 및 코디네이터 시스템
- 자바 서블릿 컨테이너에서 실행되는 자바 웹 애플리케이션 서버
- 맵리듀스 작업이나 피그(데이터 분석) 작업 같은 특화된 액션으로 구성된 워크 플로우를 제어

# Hadoop Ecosystem

## Pig

- 야후!에서 개발, 아파치 프로젝트로 전환
- 복잡한 맵리듀스 프로그래밍을 대체할 Pig Latin이라는 자체 언어 제공
- 맵리듀스 API를 크게 단순화
- SQL과 유사한 형태. 단, SQL 활용이 어려운 편

데이터 분석  
(Pig, Hive)

데이터 마이닝  
(Mahout)

## Hive

- 데이터 웨어하우징 솔루션
- 페이스북에서 개발, 아파치 프로젝트로 전환
- SQL과 매우 유사한 HiveQL 쿼리 제공(내부적으로 맵리듀스 잡으로 변환되어 실행됨)
- 자바를 모르는 데이터 분석가들도 쉽게 하둡 데이터를 분석할 수 있도록 지원
- 짧은 임시쿼리보다는 일괄적인 MapReduce 처리에 이상적

## Mahout

- 하둡 기반 데이터 마이닝 알고리즘 구현
  - › 분류(Classification)
  - › 클러스터링(Clustering)
  - › 추천 및 협업 필터링(Recommenders/Collaborative filtering)
  - › 패턴 마이닝(Pattern Mining)
  - › 회귀 분석(Regression)
  - › 차원 축소(Dimension reduction)
  - › 진화 알고리즘(Evolutionary Algorithms)

# Hadoop Ecosystem

## Hcatalog

- 하둡으로 생성한 데이터를 위한 테이블 및 스토리지 관리 서비스
- 하둡 에코시스템 간의 상호운용성 향상에 큰 영향
- Hcatalog의 이용으로 Hive에서 생성한 테이블이나 데이터 모델을 Pig나 맵리듀스에서 손쉽게 이용 가능

메타데이터 관리  
(Hcatalog)

직렬화  
(Avro)

## Avro

- RPC(Remote Procedure Call)과 데이터 직렬화를 지원
- JSON을 이용해 데이터 형식과 프로토콜을 정의
- 작고 빠른 바이너리 포맷으로 데이터를 직렬화

# Hadoop Ecosystem

## Chukwa

- 분산 환경에서 생성되는 데이터를 HDFS에 안정적으로 저장하는 플랫폼(야후!에서 개발)
- 분산된 각 서버에서 에이전트(agent)를 실행하고, 콜렉터(collector)가 에이전트로부터 데이터를 받아 HDFS에 저장
- 콜렉터는 100개의 에이전트당 하나씩 구동
- 데이터 중복 제거 등 작업은 맵리듀스로 처리

## Flume

- Chukwa처럼 분산된 서버에 에이전트가 설치되고, 에이전트로부터 데이터를 전달받는 콜렉터로 구성
- 전체 데이터의 흐름을 관리하는 마스터 서버 사용
- 데이터 수집 장소, 전송 방식, 저장소 등을 동적으로 변경 (클라우데라 개발)

비정형 데이터 수집  
(Chukwa, Flume, Scribe)

정형 데이터 수집  
(Sqoop, hiho)

## Sqoop

- 대용량 데이터 전송 솔루션
- HDFS, RDBMS, DW, NoSQL 등 다양한 저장소에 대용량 데이터를 신속하게 전송할 수 있는 방법 제공
- 상용 RDBMS도 지원하고, MySQL, PostgreSQL 오픈소스 RDBMS도 지원함

# Hadoop Ecosystem

## Scribe

- 데이터 수집 플랫폼(페이스북에서 개발)
- 데이터를 중앙 집중 서버로 전송
- 최종 데이터는 HDFS 외 다양한 저장소 활용
- 설치와 구성이 쉽도록 다양한 프로그래밍 언어 지원
- HDFS에 데이터 저장을 위해 JNI(Java Native Interface) 사용

## Hiho

- 대용량 데이터 전송 솔루션
- 하둡에서 데이터를 가져오기 위한 SQL 지정 가능
- JDBC 인터페이스 지원
- 오라클과 MySQL의 데이터 전송만 지원

비정형 데이터 수집  
(Chukwa, Flume, Scribe)

정형 데이터 수집  
(Sqoop, hoho)