1. 하둡이란?

하둡은 대용량 데이터를 분산 처리할 수 있는 자바 기반의 오픈 소스 프레임워크이다. 하둡은 구글이 논문으로 발표한 GFS(Google File System)과 맵리듀스(MapReduce)를 2005년 더그커팅이 구현한 결과물이다. 하둡은 분산시스템인 HDFS(Hadoop Distributed File System)에 데이터를 저장하고, 맵리듀스를 이용해 데이터를 처리한다.

하둡은 여러 대의 서버에 데이터를 저장하고, 저장된 각 서버에서 동시에 데이터를 처리하는 방식이다. 하둡은 기존의 RDBMS(Oracle,MS-SQL, MySQL등)을 대치하는 것이 아니다. 즉 트랜젝션이나 무결성을 보장해야하는 데이터처리에는 적합하지 않다. 하둡은 배치성으로 데이터를 저장하고 처리하는데 적합한 시스템이다.

쇼핑물에서 회원가입이나, 결제진행등은 모두 트랜젝션이나 무결성을 보장해야한다. 이런 것들을 하둡으로 처리하는 것이 아니라, 회원이 관심있게 보는 물품들이나, 이동경로, 머무르는 시간등 배치성으로 저장되는 데이터에 적합하다. 이런 것들을 매번 비용이 비싼 RDBMS에 저장하면 낭비요소이다. 그러므로 하둡은 RDBMS와 경쟁하는 것이 아닌 RDBMS와 협력하는 것이라 볼 수 있다.

2. HDFS

HDFS는 Hadoop Distributed File System의 약자이다.  
수십 테라바이트 또는 페타바이트 이상의 대용량 파일을 분산된 서버에 저장하고, 그 저장된 데이터를 빠르게 처리할 수 있게 하는 파일시스템이다. 또한 저사양의 서버를 이용해서 스토리지를 구성할 수 있어 기존의 대용량파일스시스템(NAS, DAS, SAN등)에 비해 장점을 가진다. HDFS는 블록 구조의 파일 시스템이다. 파일을 특정크기의 블록으로 나누어 분산된 서버에 저장된다. 블록크기는 64MB에서 하둡2.0부터는 128M로 증가되었다.

ZooKeeper란 무엇인가?

분산 시스템을 설계 하다보면, 가장 문제점 중의 하나가 분산된 시스템간의 정보를 어떻게 공유할것이고, 클러스터에 있는 서버들의 상태를 체크할 필요가 있으며 또한, 분산된 서버들간에 동기화를 위한 락(lock)을 처리하는 것들이 문제로 부딪힌다.

이러한 문제를 해결하는 시스템을 코디네이션 서비스 시스템 (coordination service)라고 하는데, Apache Zookeeper가 대표적이다. 이 코디네이션 서비스는 분산 시스템 내에서 중요한 상태 정보나 설정 정보등을 유지하기 때문에, 코디네이션 서비스의 장애는 전체 시스템의 장애를 유발하기 때문에, 이중화등을 통하여 고가용성을 제공해야 한다. ZooKeeper는 이러한 특성을 잘 제공하고 있는데, 그런 이유로 이미 유명한 분산 솔루션에 많이 사용되고 있다. NoSQL의 한종류인 Apache HBase, 대용량 분산 큐 시스템인 Kafka등이 그 대표적인 사례이다.

분산 시스템을 코디네이션 하는 용도로 디자인이 되었기 때문에, 데이타 억세스가 빨라야 하며, 자체적으로 장애에 대한 대응성을 가져야 한다. 그래서 Zookeeper는 자체적으로 클러스터링을 제공하며, 장애에도 데이타 유실 없이 fail over/fail back이 가능하다.

Apache Zookeeper의 기능은 사실상 별로 없다. 디렉토리 구조기반으로 znode라는 데이타 저장 객체를 제공하고, (key-value식). 이 객체에 데이타를 넣고 빼는 기능만을 제공한다. 일단 디렉토리 형식을 사용하기 때문에 데이타를 계층화된 구조로 저장하기 용이하다.