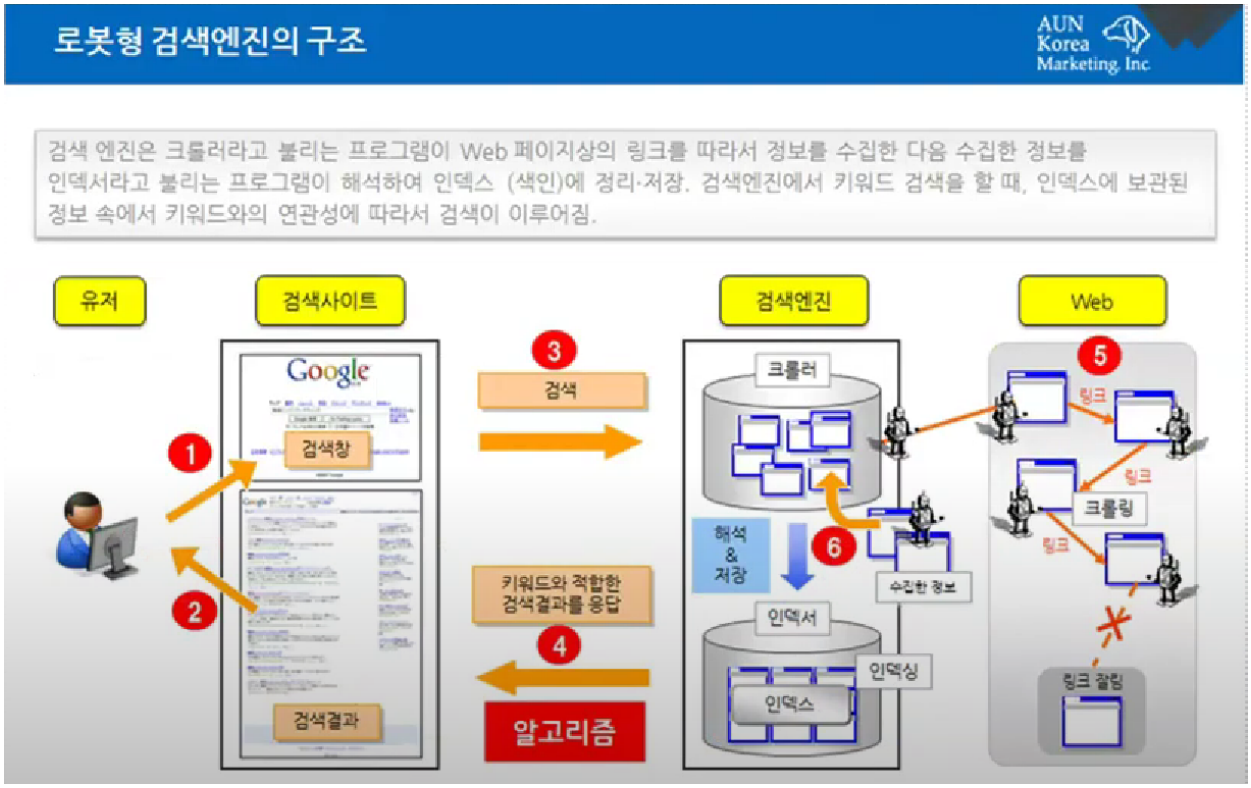
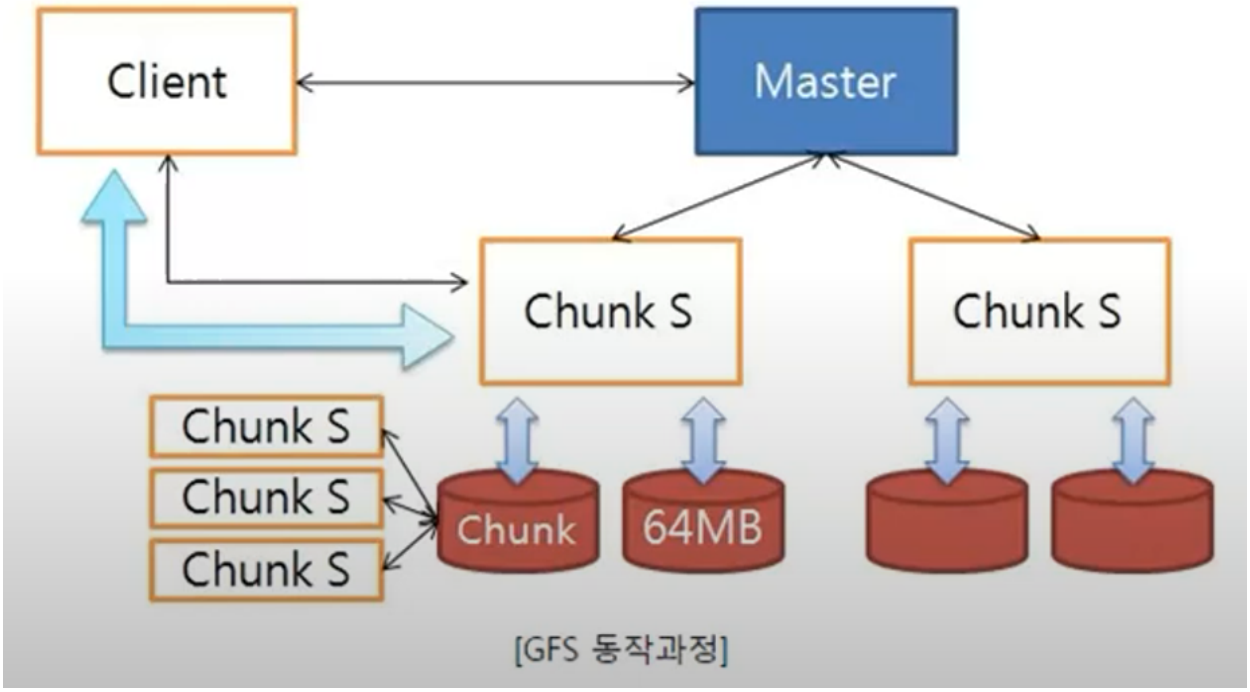
5. 구글_파일 시스템(GFS, Google File System)

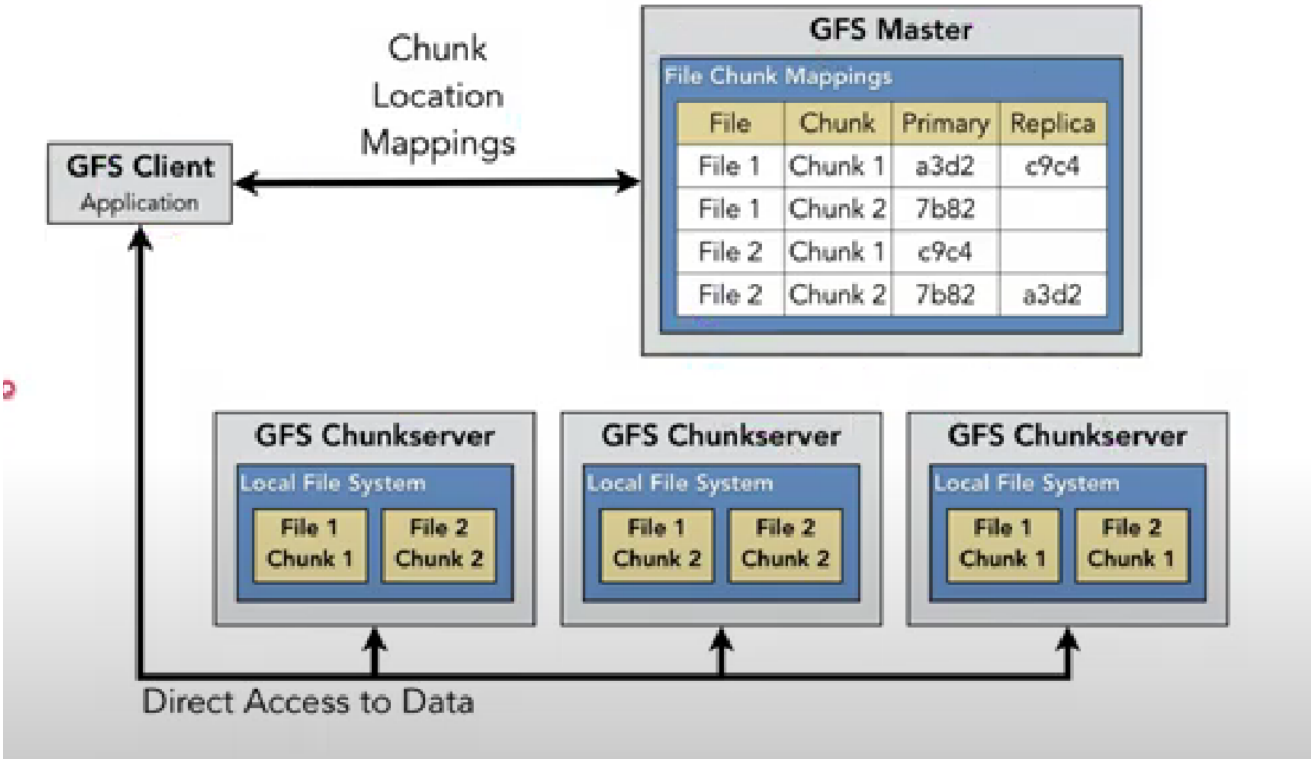
0. 배경



- 구글의 GFS
- 관련된 색인이나 정보들을 정형화된 데이터로 저장하기가 힘들어짐
  --〉 새로운 구조의 DB를 만들 필요성 생김 -〉 처음 시도한 회사가 바로 구글
  --〉 구글파일시스템(GFS)
  - 2003년에 구글 제품에 활용 중이던 GFS(Google File System)라는 구글 분산 파일 시스템의 아키텍처가 논문으로 출판(뒷장그림)
  - 분산 파일 시스템 (DFS: distributed file system) : 클라이언트 측에서 서버에 저장된 데이터에 접근하여, 마치 자신에게 저장되어 있는 데이터인 것처럼, 처리할 수 있는 클라이언트/서버 기반의 애플리케이션
  - GFS(분산파일시스템) : 웹 크롤러 색인처리 과정에서 생성되는 매우 큰 파일에 대한 스토리지 요구 사항을 해결
- 구글이 학회지에 'GFS'와 'MapReduce' 논문을 실음 -〉 '더그 커팅'이 갖고와서 '아파치'에 실었고 -〉 아파치가 '야후'에 넘어가면서 -〉 발전 -〉 현재의 하둡
  =〉 결국, 구글에서 다 만들어 놓은 것이라고 봐도 됨

1. GFS





- 구성요소 (3) - Cilent, Master, Chunkserver
  - Cilent - 파일을 읽고 쓰는 동작을 요청하는 어플리케이션
    (고객은 마스터와 소통함)
  - Master - GFS 전체의 상태를 관리하고 통제하는 중앙 서버 역할
    (네임노드, 파일에 대한 정보를 엑셀처럼 text로 저장, 파일 위치/분할된 chunk의 위치/P.K/복제된 키 =〉 맵핑 (어느 하나 파일을 찍으면 어느 위치에 있는지 매칭이 된다는 것), 메타정보 저장)

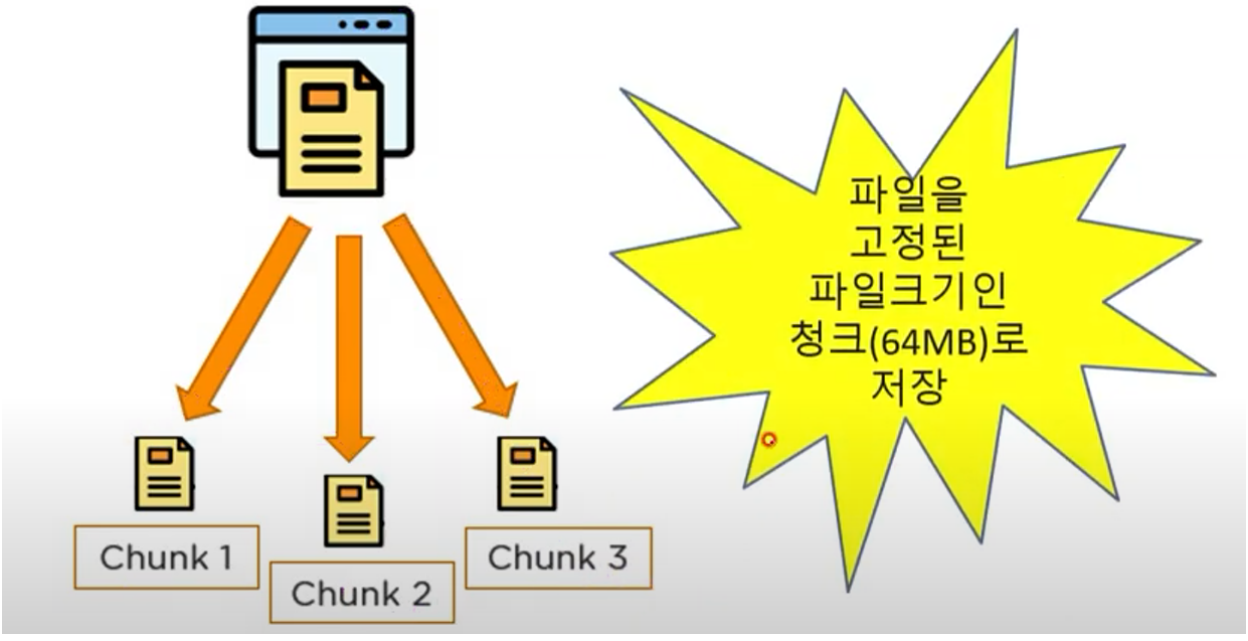
- Chunkserver – 물리적인 하드디스크에 실제 입출력 처리
  (실제 데이터 서버에 저장)

2. GFS 개발 배경
   : 구글 클라우드 시스템을 위한 저비용, 고효율 분산 저장 시스템 니즈 필요. 구글에서 개발

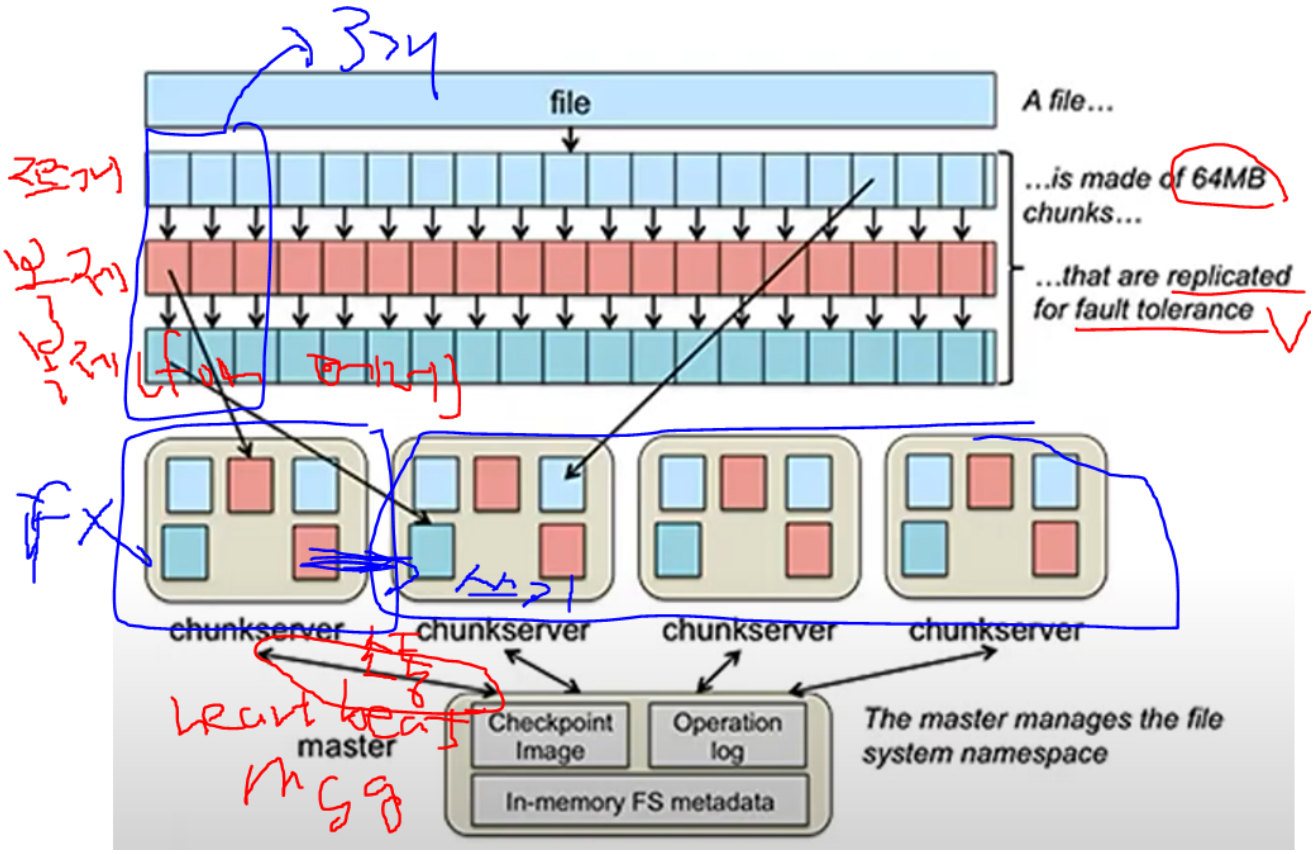
- Fault Tolerance : 네트워크, 하드, CPU 등 장애조치가 원만해야함.
- Scalablity : 다수 사용자, 수많은 서버 등등. 확장성
- Auto Computing : 관리, 모니터가 힘듦 -〉 자동 할당.

3. GFS 원리, 키워드

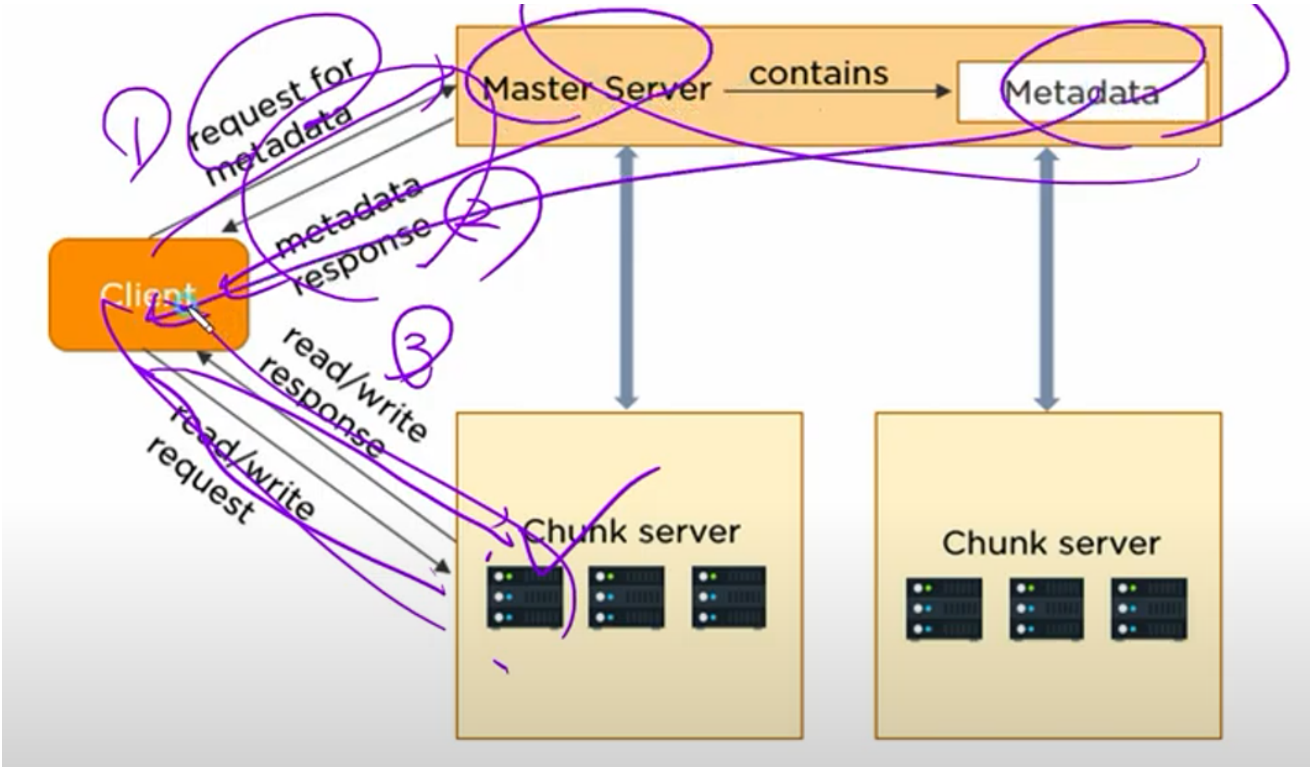
- 청크(Chunk)
  : GFS에서 파일을 나누는 조각 한 단위. 1개의 청크는 64MB 고정된 크기로 분할 저장
- 마스터 – 단일마스터, 파일 메타정보 저장, 하트비트msg(상태체크msg)
  - cf) 서버에 고장이 나면 관리하는 마스터가 존재한다는 것
- 청크서버 – 청크 저장, 청크 파일처리, 하트비트 마스터 전달
- 클라이언트 – 마스터에 청크인데스 요청. 청크 읽기/쓰기
  - 클라이언트(서버를 사용하는 사용자 pg)
  - 파일을 직접적으로 읽는 것이 아니라, 마스터를 통해서 위치/정상 여부를 통보받은 후 파일을 읽기/쓰기를 하게 됨

4. 

- 파일을 항상 64MB의 청크로 잘라서 보관



- 청크서버는 원본만 갖고 있는 것이 아니라, 복제본도 같이 갖고 있음
  --〉 해당 청크서버 자체를 복사해서 갖고 있음 (서버 고장 대비)



- Client가 파일의 위치를 물어봄 -〉 Master가 ChunkIndex를 주면서 위치를 알려줌 -〉 특정 파일이 있는 ChunkServer에 접근 -〉 읽기/쓰기

- 앞선 과정들 확인하고, 아래 2개 도식을 다시보면 이해 됨