

유튜브 설계

1단계 문제 이해 및 설계 범위 확정

Q: 어떤 기능이 가장 중요한가요?

W: 비디오를 올리는 기능과 시청하는 기능

Q: 어떤 클라이언트 지원

W: 모바일 앱, 웹, 웹 브라우저, 그리고 스마트 TV

Q: 일간 능동 사용자 수는 몇 명?

W: 5백만

Q: 제품에 평균적으로 소비하는 시간은?

W: 30분

Q: 다국어 지원

W: 지원 해야 합니다.

요구사항을 정리하면

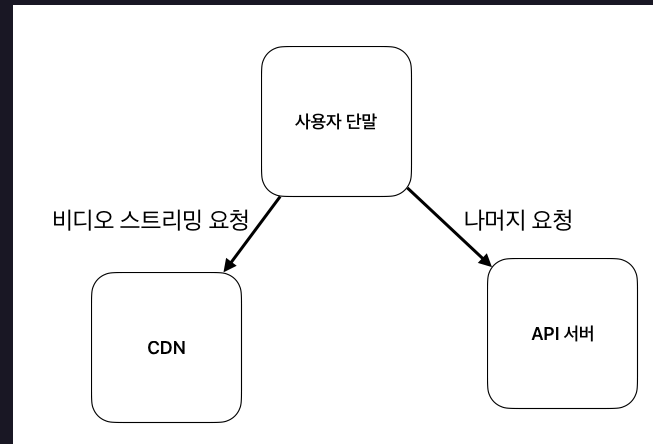
- 빠른 비디오 업로드
- 원활한 비디오 재생
- 재생 품질 선택 기능
- 낮은 인프라 비용
- 높은 가용성과 규모 확장성, 그리고 안정성
- 지원 클라이언트: 모바일 앱, 웹브라우저, 그리고 스마트 TV

규모 추정하면

- 일간 능동 사용자(DAU)는 5qoraks
- 평균적으로 한 사용자는 5개 비디오 시청
- 10% 사용자가 하루에 1비디오 업로드
- 비디오 평균 크기는 300MB
- 비디오 저장을 위해 매일 새로 요구되는 저장용량 = 5백만 * 10% * 300MB = 150TB
- CDN 비용

2단계 개략적 설계안 제시 및 동의 구하기

이 시스템은 세 개 컴포넌트로 구성

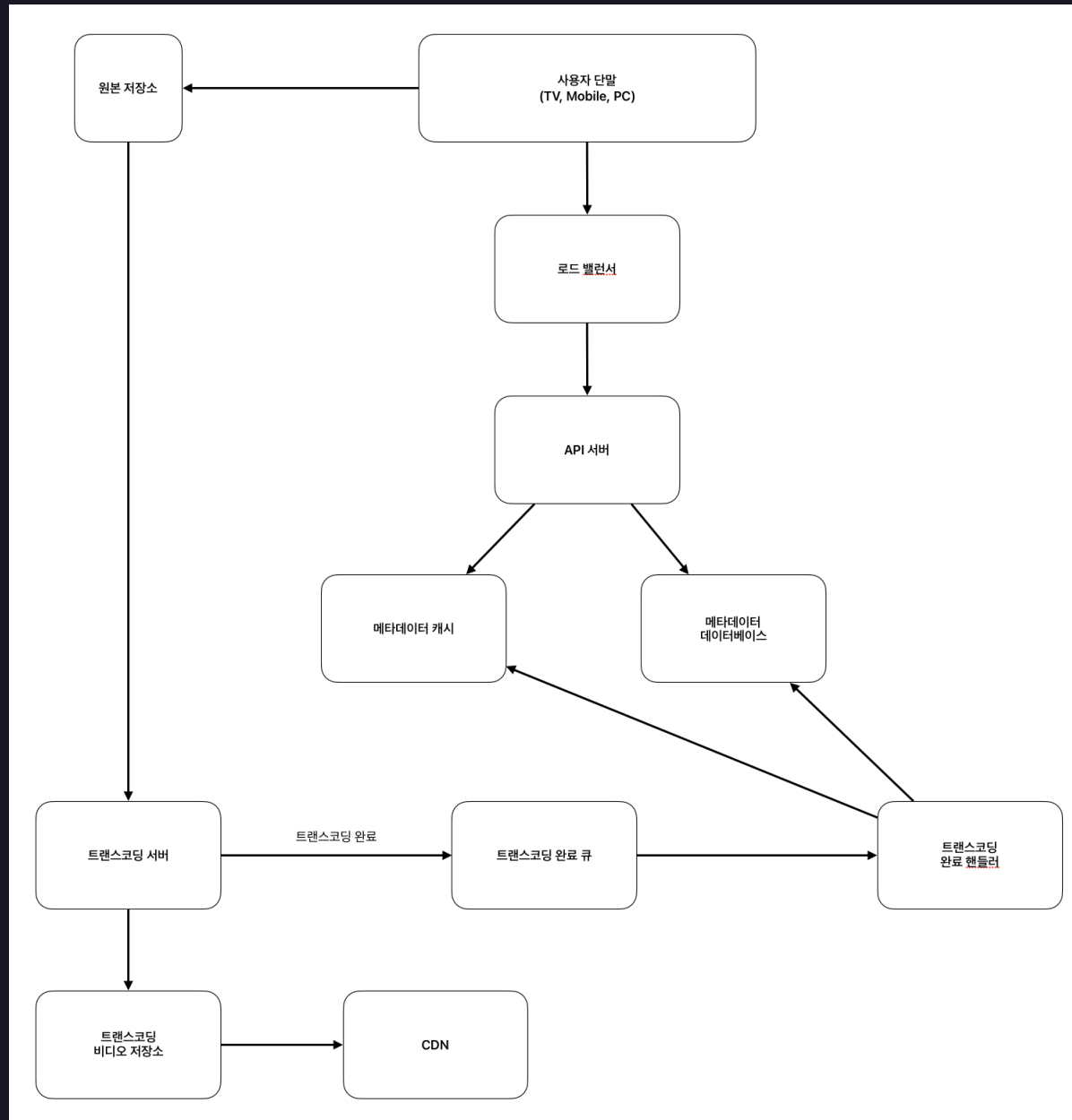


단말(client)

CDN

API 서버

스트리밍을 제외한 모든 요청은 API 서버가 처리. 피드 추천, 비디오 업로드 URL 생성, 메타데이터 데이터베이스와 캐시 갱신, 사용자 가입



메타데이터 데이터베이스

- 비디오의 메타데이터를 보관. 샤딩과 다중화를 적용하여 성능 및 가용성 요구사항을 충족

메타데이터 캐시

- 성능을 높이기 위해 비디오 메타데이터와 사용자 객체는 캐시

트랜스코딩 서버

- 비디오 트랜스코딩은 비디오 인코딩이라 부르기도 하는 절차로, 비디오의 포맷을 변환하는 절차

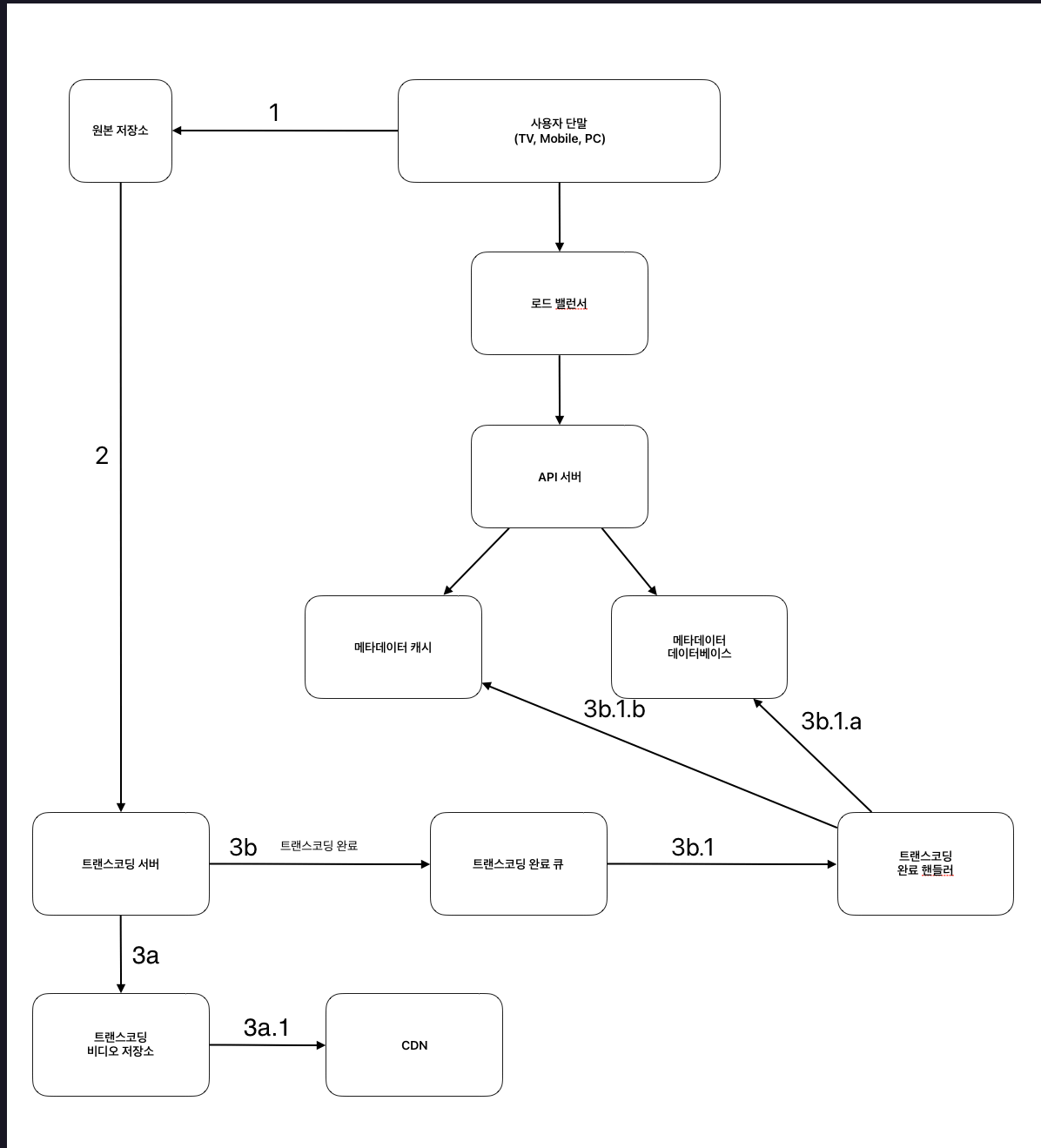
트랜스코딩 비디오 저장소

- 트랜스코딩이 완료된 비디오를 저장하는 BLOB 저장소

트랜스코딩 완료 핸들러

- 트랜스코딩 완료 큐에서 이벤트 데이터를 꺼내어 메타데이터 캐시와 데이터베이스를 갱신할 작업 서버들

a. 비디오 업로드시



3단계 상세 설계

이야기 하고 싶은 컴포넌트

- 비디오 트랜스코딩
- 유향 비순환 그래프 모델
- 비디오 트랜스코딩 아키텍처
- DAG 스케줄러
- 자원 관리자
- 작업 서버
- 임시 저장소
- 인코딩된 비디오
- 시스템 최적화
 - 속도 최적화: 비디오 병렬 업로드
 - 속도 최적화: 업로드 센터를 사용자 근거리에 지정
- 오류 처리

비디오 트랜스코딩

원본을 그대로 저장하면 저장공간을 많이 차지 한다. 그러므로, 특정 비트레이트로 가공하기도하고, 호환되는 포맷으로 변경되어야 할 수도 있다.

유형 비순환 그래프(DAG)모델

각기 다른 유형의 비디오 프로세싱 파이프라인을 지원하는 한편 처리 과정의 병렬성을 높이기 위해서는 적절한 수준의 추상화를 도입하여 클라이언트 프로그래머로 하여금 실행할 작업(task)을 손수 정의할 수 있도록 한다. ex. 썸네일 추출, 워터마크