

CH007.

## 분산 시스템을 위한 유일 ID 생성기 설계

## Q) AI는 정말 최선일까?

## A) 아니

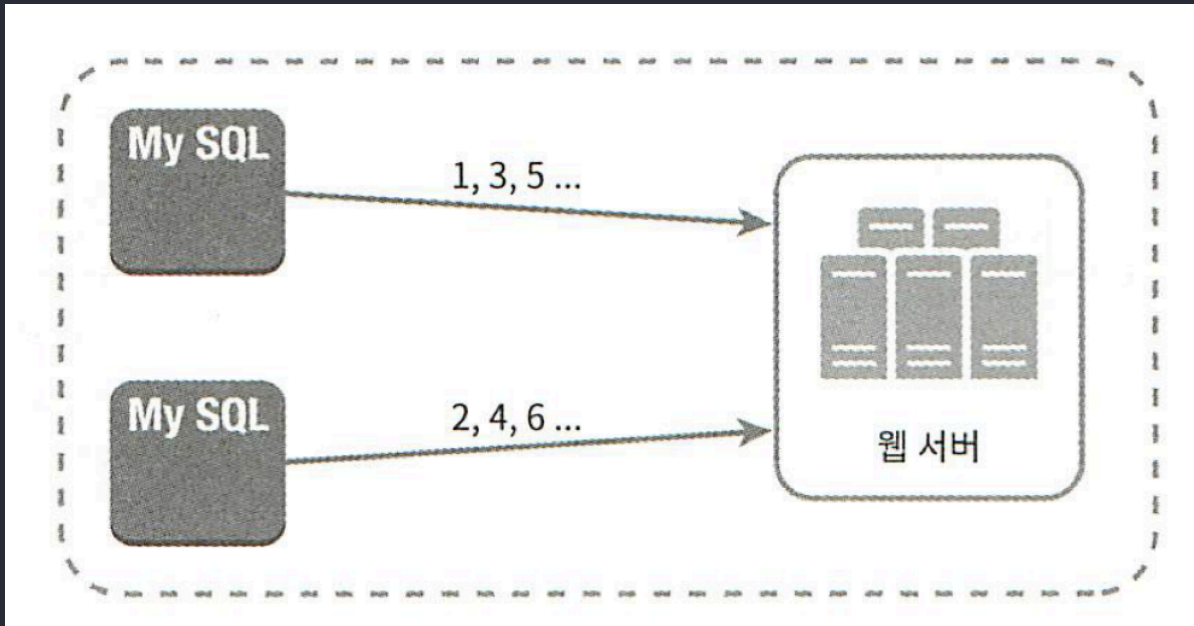
- 애초에 분산 환경에서 통할까?
- 분산 환경이라는 것은 대규모 데이터를 생성해 내기에 성능상 이점을 내려고 분산한 것
- 이는 서버 한 대로 무리
- 만약 여러 RDB 쓰는 경우, **AI 값을 동기화 하기위한 delay가 발생함**

**Q) 분산 시스템에서 유일성이 보장되는 ID 만는 방법?**

## A)

- 다중 마스터 복제
- UUID
- 티켓 서버
- 트위터 스노우플레이크

## Q) 다중 마스터 복제란?



- AI 기능을 그대로 활용하되, 복제된 서버 개수( $k$ )만큼 증가시킴
- 근데  $k$ 가 확정되면 바꾸기 어려움

## Q) UUID란?

- 128비트 짜리 유니크 식별자 ➡ 이게 최적화 필요함
- 완전히 독립된 서버에서도 생성 가능
- 시간순으로 정렬이 안됨 ➡ 최근에 나온 `UUIDv7` 은 시간 순서 보장

9b1deb4d-3b7d-41

## Q) 티켓 서버란?

- 유일성이 보장되는 AI ID를 만드는 중앙 집중식 서버
- 구현하기 너무나도 쉬움
- 근데 이제 SPOF임 ➡ 티켓 서버를 여러대 준비해도 동기화 문제 발생



## Q) 트위터 스노우플레이크란?

### snowflake 식 접근 방식

1비트	41비트	5비트	5비트	12비트
sign	타임스탬프	데이터 센터 ID	서버 ID	일련번호

## Q) 트위터 스노우플레이크란?

- 일단 데이터센터 ID랑 서버 ID는 시스템 시작시 결정되고, 이후 변경 불가함
- 41비트의 타임스탬프 덕분에 ➡ 시간 순서 보장  
🤔 표현할 수 있는 값은 **69년**인데, epoch을 현재에 가깝게 맞추면 오버플로 방지할 수 있음
- 12비트의 일련번호는  $2^{12} = 4096$ 개 값을 갖는다는 의미임  
➡ 0.001초당 최대 4096개의 서로 다른 ID를 생성함

끗.

나머지는 공부해와