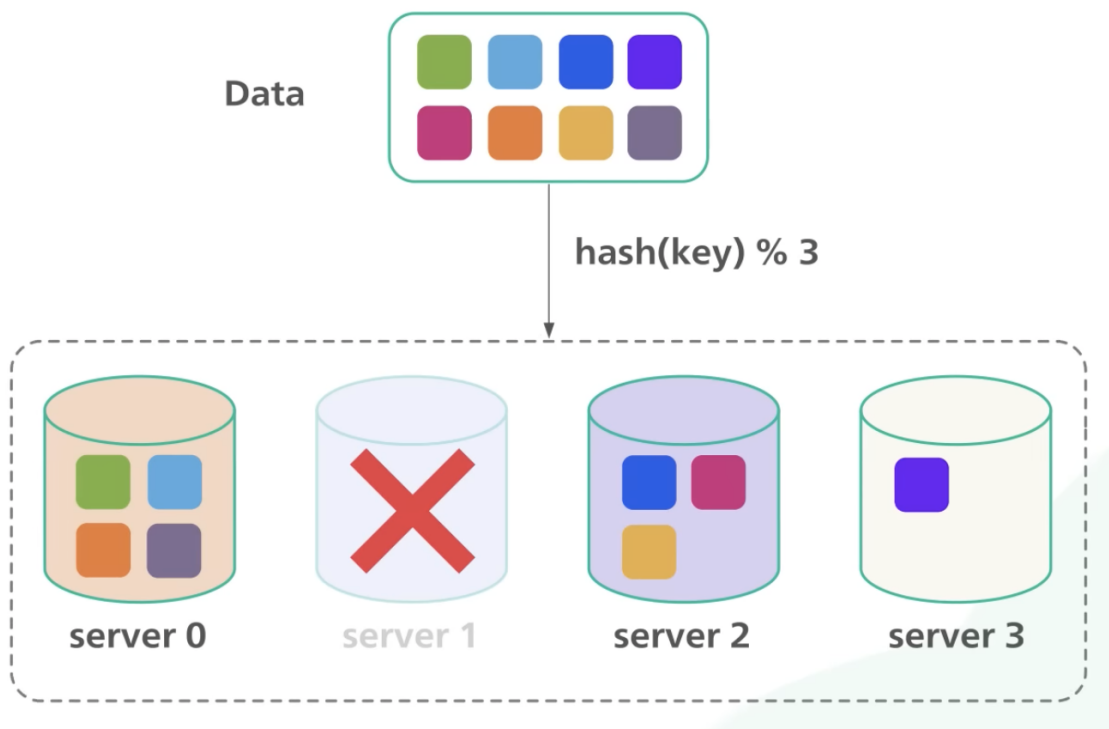


안정해시설계

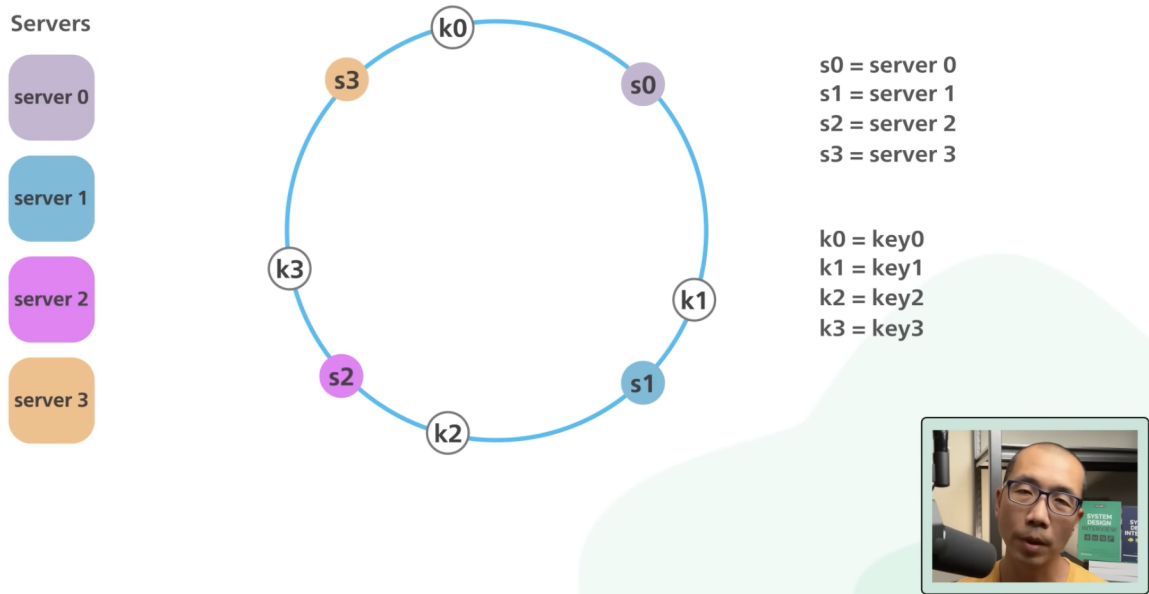
해시 키 재배치 문제

- $\text{serverIndex} = \text{hash}(\text{key}) \% N(\text{server count})$
 - 서버의 풀의 크기가 고정되어 있을 때, 데이터 분포가 균등할 때 잘 동작한다.
 - 이 상황에서 서버의 풀 크기의 변화가 발생하면 거의 대부분의 키가 재분배된다.
 - 이는 대규모 cache miss를 이끈다.



안정 해시란?

안정 해시는 해시 테이블 크기가 조정될 때 평균적으로 k/n 개의 키만 재배치하는 해시 기술



해시 공간과 해시 링

- 해시 공간은 해시 함수를 통해 나올 수 있는 값의 범위로 보면 된다.
- $x_0 \sim x^{n-1}$ 를 범위로 볼때 양 극단의 값을 마주보게 하여 링을 만들 수 있다. 이를 hash ring이라고 한다.

해시 서버 : 서버를 해시 링 위에 배치시킨다.

해시 키 : 해시 공간안에 해시 키는 배치될 수 있다.

서버 조회(해시 조회) : 해당 키의 위치로부터 시계 방향으로 링을 탐색해서 만나게 되는 첫 번째 서버다.

이 때 추가하는 경우나 제거하는 경우에 서버의 위치를 이동시키지 않고 임의의 위치에 추가 하기 때문에 재배포 되는 키의 량이 현격히 줄어든데. 그래서 보통 k/n 개의 키만 재 배치된다고 볼 수 있다.

문제

1. 서버가 추가되거나 삭제되는 상황에서 파티션의 크기를 균등하게 유지하는 것이 불가능하다.
2. 키의 균등 분포를 달성하기 매우 어렵다.

가상 노드 기법

1. 실제 노드보다 더 많은 수의 임의의 가상 노드들을 링 위에 배치함으로써 키의 균등 분포 문제를 해소하는 방식이다.
2. 예를 들어 A, B 서버에 대해서 A1, A2, A3, B1, B2, B3 가상노드를 링위에 교차하여 배치하고 키를 시계 방향의 첫 번째 가상 노드로 배치시키는 것이다.
3. 가상 노드의 개수를 늘리면 키의 분포를 점점 더 균등하게 만들 수 있다.

키를 재배포하는 경우

1. 추가되는 경우 : 추가할 노드의 반시계 방향의 모든 키들을 추가된 노드로 배치한다.
2. 삭제되는 경우 : 삭제될 노드의 이전 키들을 모두 다음 노드의 키로 재 배치한다.

마치며

1. 서버 추가 제거에 따른 재배포 키 수를 최소화할 수 있다.
2. 균등 분포가 가능하므로 수평적 규모 확장에 유리하다
3. hotspot 문제를 줄일 수 있다. 특정한 샤드에 대한 접근이나 지나치게 빈번한 서버 과부하 문제를 줄일 수 있다.
4. 관련 컴포넌트 : 다이نامो 데이터베이스, 카산드라, 디스코드, 아카마이 CDN, 매그래프