

Managing Redis - Scaling

🕒 생성일	@2022년 12월 28일 오후 8:14
🏷 태그	

Scaling with Redis Cluster

| 레디스 클러스터를 이용한 수평적 스케일링

Redis를 수평으로 확장하려면 Redis Cluster라는 배포 토폴로지를 사용해야한다.

Redis Cluster 101

Redis cluster는 파티션중에도 어느 정도의 가용성을 제공한다. 여기서 가용성은 일부 노드가 통신할 수 없거나 실패할 때도 작업을 계속할 수 있는 기능을 말한다. 하지만 모든 마스터를 사용할 수 없는 결정적인 장애인 경우에는 클러스터를 사용할 수 없다.

클러스터 사용시 이점

- 여러 노드 간에 데이터 세트를 자동으로 분할한다. (오토 파티셔닝)
- 노드의 하위 집합에 장애가 발생하거나 나머지 클러스터와 통신할 수 없는 경우 작업을 계속한다.(가용성)

Redis Cluster TCP ports

모든 클러스터 노드에는 두 개의 개방형 TCP 연결이 필요하다. 보통은 6379와 16370이다.

클러스터 버스는 대역폭과 처리 시간이 적어 노드간 정보 교환에 더 적합한 바이너리 프로토콜을 사용하는 노드 간 통신 채널이다. 장애 감지, 구성 업데이트, 장애 조치 권한 부여 등에 사용된다. 클라이언트는 버스포트를 사용하면 안되고 Redis 명령 포트를 사용해야한다. 그리고 방화벽에서 두 포트 모두 열어야한다.

Redis 클러스터가 작동하기 위해 각 노드에 필요한 것.

- 클라이언트 통신 포트는 모든 클라이언트와 모든 클러스터 노드에서 접근할 수 있어야 한다.
- 클러스터 버스 포트는 모든 클러스터 노드에서 접근할 수 있어야 한다.

두 TCP 포트를 열지 않으면 클러스터가 예상대로 작동하지 않는다.

Redis Cluster and Docker

Docker를 이용해 Redis Cluster와 호환되도록 하려면 Docker의 호스트 네트워킹 모드를 사용해야 한다.

Redis Cluster data sharding

Redis Cluster는 일관된 해싱을 사용하지 않고 모든 키가 개념적으로 해시 슬롯 이라고 하는 것의 일부인 다른 형태의 샤딩을 사용한다. 총 16384개의 해시 슬롯이 있고, 각 노드에서 이 해시슬롯을 분할해서 가지고 있을 수 있다. 이를 통해 클러스터 노드를 쉽게 추가하고 제거할 수 있다. 한 노드에서 다른 노드로 해시 슬롯을 이동하는 데 작업을 중지할 필요가 없다. 그러므로 노드를 추가, 제거하거나 노드가 가진 해시슬롯의 비율을 변경할 때도 가동 중지 시간이 필요하지 않는다.

Redis Cluster는 단일 명령 실행(트랜잭션과 Lua 스크립트 포함)에 관련된 모든 키가 동일한 해시 슬롯에 속하는 한 여러키 작업을 지원한다. 해시태그라는 기능을 이용해 여러 키를 동일한 해시 슬롯의 일부로 만들 수 있다.

쉽게 이야기해서 어떤 데이터를 같은 해시슬롯에 넣어둘 수 있고, 다중 키 조작으로 이 두 키를 조작할 수 있다.

Redis Cluster master-replica model

마스터 노드의 하위 집합이 실패하거나 대부분의 노드와 통신할 수 없을 때 사용가능한 상태를 유지하기 위해 Redis 클러스터는 모든 해시슬롯에 대해 1~N개의 복제본을 가질 수 있다.

Redis Cluster consistency guarantees

Redis 클러스터는 강력한 일관성을 보장하지 않는다. 쓰기 읽기를 누락할 수 있다. 이유는 비동기 복제에 있다.

기본적으로 성능과 일관성 사이에는 트레이드 오프가 있다. WAIT명령어를 사용하면 동기식으로 쓰기 작업을 할 수 있다. 손실 가능성이 훨씬 줄어들지만 성능이 저하될 수 있고, 100% 일관성을 보장하지 않는다.

노드 타임아웃은 파티션의 다수측의 복제본을 마스터로 승격하기에 충분한 시간이 경과한 이후 소수측의 모든 마스터 노드는 쓰기 수락을 중지하는 것을 말한다. 노드 제한 시간이 경과한 후 마스터 노드는 실패한 것으로 간주되어 해당 복제본 중 하나로 교체될 수 있다. 이 경우 모든 쓰기 수락이 중지된다.

Redis Cluster configuration parameters

redis.conf 파일에 들어가는 구성요소 설명

- 클러스터 여부 설정

- 클러스터 구성에 대한 옵션들이 자동으로 유지되는 파일
- 타임아웃 시간 설정
- 항상 마스터를 믿을 것인지, 아니면 장애 시 복제본을 사용할 것인지 여부와 기간 설정
- 복제본 마이그레이션할 수 있는 마스터와 연결된 최소 복제본 수

Create and use a Redis Cluster

Requirements to create a Redis Cluster

클러스터를 생성하려면 먼저 클러스터에서 실행중인 세개 이상의 몇개의 빈 Redis 인스턴스가 있어야한다. 배포를 위해 3개의 마스터와 3개의 복제본이 있는 6노드 클러스터를 강력히 권장한다. 클러스터를 실행하면 노드ID가 생성되고, 이 노드ID는 클러스터내에서 고유하고 변하지 않는다. 포트와 IP는 변경 가능하다.

Create a Redis Cluster

명령어를 사용하거나, 수동으로 클러스터 생성 가능하다.

Interact with the cluster

클러스터와 연결하려면 클러스터 인식 Redis 클라이언트가 필요하다. redis-cli를 이용해 테스트 가능하다.

Reshard the cluster

리샤딩은 노드 집합에서 다른 노드 집합으로 해시 슬롯을 이동하는 것을 의미한다. redis-cli 유틸리티를 사용하여 수행한다. 단일 노드를 지정하면 redis-cli가 다른 노드를 자동으로 찾고, 현재 redis-cli는 관리자 지원으로만 리샤딩 가능하다.

리샤딩을 할때 많은 파라미터를 사용해서 한번에 진행할 수 있다. 이 경우 어느정도 자동화가 가능하지만 필요에 따라 지능적으로 슬롯을 이동하는 클러스터를 자동으로 재조정할 수 있는 방법이 없다.

Test the failover

장애를 발생 시키기 위해서는 단일 프로세스를 충돌시켜야 한다. 충돌시키는 명령어는 따로 있다.

Add a new node

새 노드를 추가하는 방법

1. 빈 노드를 추가한다.

2. 용도에 따른 설정을 한다.

- a. 새 마스터인 경우 일부 데이터를 그 안으로 이동
- b. 새 복제본인 경우 알려진 노드의 복제본으로 설정

Remove a node

노드를 삭제하려면 노드ID를 알고 있어야 한다. 마스터 노드를 제거할 수 있지만, 마스터 노드를 삭제하려면 비어있어야 한다.

Replica migration

시스템 관리자의 도움 없이 복제본의 자동 재구성을 복제본 마이그레이션이라고 한다. 이를 통해 Redis Cluster의 안정성을 향상시킬 수 있다. 클러스터 복제본을 한 마스터에서 다른 마스터로 이동시키는 이유는 일반적으로 Redis 클러스터가 주어진 마스터에 연결된 복제본 수만큼 장애에 강하기 때문이다.

Upgrade nodes in a Redis Cluster

노드를 중지하고 업데이트된 Redis 버전으로 다시 시작하기만 하면 되기 때문에 복제본 노드 업그레이드가 쉽다.

Migrate to Redis Cluster

단일 마스터나 다른 샤딩 옵션을 채택한 구조에서 Redis Cluster로 마이그레이션 할 경우, 가장 중요한 세부사항은 애플리케이션에서 다중 키 작업을 사용하는지 여부와 방법이다.