

Redis Partitioning

Partitioning là gì

Partitioning là quá trình phân chia dữ liệu thành nhiều phân vùng riêng biệt (instance) trên Redis, do đó mọi phân vùng sẽ chỉ chứa một tập hợp con các key của bạn.

Lợi ích của việc Partitioning

- Nó cho phép Redis database lưu trữ lớn hơn nhiều bằng cách sử dụng tổng bộ nhớ của nhiều máy tính. Nếu không partitioning, bạn sẽ bị giới hạn số lượng bộ nhớ mà một máy tính có thể hỗ trợ.
- Nó cho phép mở rộng sức mạnh tính toán đa luồng (multiple cores), đa máy tính (multiple computers), và băng thông mạng cho nhiều máy tính và network.

Nhược điểm của Partitioning

- Các hoạt động liên quan đến nhiều khóa (multiple keys) thường không được hỗ trợ. Chẳng hạn, bạn không thể thực hiện thi trên hai tập (sets) nếu chúng được lưu trữ trong các khóa được ánh xạ tới các Redis instance khác nhau.
- Giao dịch Redis liên quan đến nhiều khóa (multiple keys) không thể được sử dụng.
- Partitioning đặc biệt là khóa, vì vậy không thể loại bỏ một tập dữ liệu với một khóa lớn như một tập hợp được sắp xếp rất lớn.
- Khi Partitioning được sử dụng, xử lý dữ liệu phức tạp hơn. Chẳng hạn, bạn phải xử lý nhiều tệp RDB / AOF và để có bản sao lưu dữ liệu của mình, bạn cần tổng hợp các tệp lưu trữ từ nhiều phiên bản và máy chủ lưu trữ.
- Thêm và loại bỏ capacity có thể phức tạp. Chẳng hạn, Redis Cluster hỗ trợ cân bằng lại dữ liệu trong suốt với khả năng thêm và xóa các node khi chạy. Tuy nhiên, các hệ thống khác như Partitioning phía máy khách và proxy không hỗ trợ tính năng này. Một kỹ thuật được gọi là `Presharding` giúp về vấn đề này.

Các loại Partitioning

Có hai loại Partitioning có sẵn trong Redis. Giả sử chúng ta có bốn phiên bản Redis, R0, R1, R2, R3 và nhiều `key` đại diện cho người dùng như: user:1, user:2....

Range Partitioning

Range Partitioning được thực hiện bằng cách ánh xạ phạm vi của các đối tượng vào các Redis instance cụ thể. Giả sử trong ví dụ của chúng tôi, người dùng có từ ID.0 đến ID.10000 sẽ đi vào R0 instance, trong khi người dùng từ ID.10001 đến ID.20000 sẽ vào R1 instance...

Hash Partitioning

Trong kiểu phân vùng này, một hàm băm - hash (ví dụ: hàm mô đun) được sử dụng để chuyển đổi key thành số và sau đó dữ liệu được lưu trữ trong các Redis instance khác nhau.

Revision #1

Created 13 October 2019 06:05:16 by Laptrinh.vn

Updated 13 October 2019 06:23:19 by Laptrinh.vn