In-Memory: bộ nhớ trong

Key-Value Store: là một hệ thống lưu trữ dưới cặp khóa và giá trị

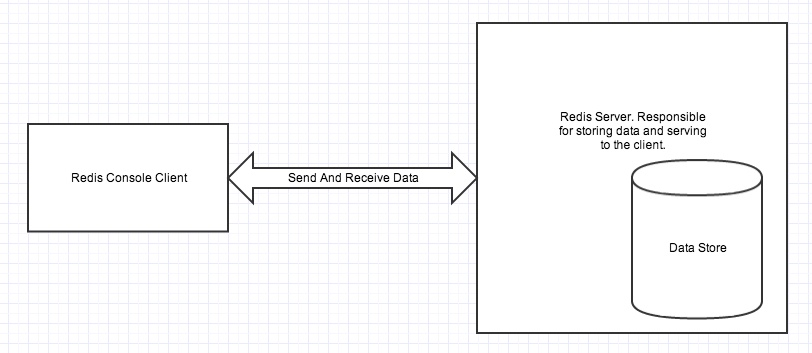
Trong Redis khóa phải là một chuỗi, giá trị có thể là string, list, set, sorted set hoặc hash.

Ưu điểm: . Redis lưu trữ dữ liệu trong bộ nhớ chính (RAM) nên tốc độ đọc ghi dữ liệu rất nhanh.

Nhược điểm: Vì bộ nhớ RAM hạn chế, nên Redis không thể lưu được các dữ liệu lớn mà chỉ lưu những văn bản nhỏ mà cần truy cập, sữa đổi và chèn vào tới một tốc độ rất nhanh.

Kiến trúc của Redis: nó chưa 2 quy trình chính

* Redis client: điều khiển truy vấn dữ liệu từ giao diện hoặc bất kỳ ngôn ngữ lập trình khác thông qua ngôn ngữ API của Redis
* Redis server: chịu trách nhiệm cho việc sử lý và lưu trưc dữ liệu trong bộ nhớ. Nó xử lý tất cả các loại hình quản lý và tạo thành phần chính của kiến trúc.
* 2 máy này có thể nằm trên cùng một hoặc 2 máy tính khác nhau



Nguyên lý: Bộ nhớ chính là không ổn định và sẽ mất tất cả các dữ liệu được lưu trữ khi máy chủ gặp sự cố. Vì vậy chúng ta cần một cách để duy trì. Gồm có 3 cơ chế;

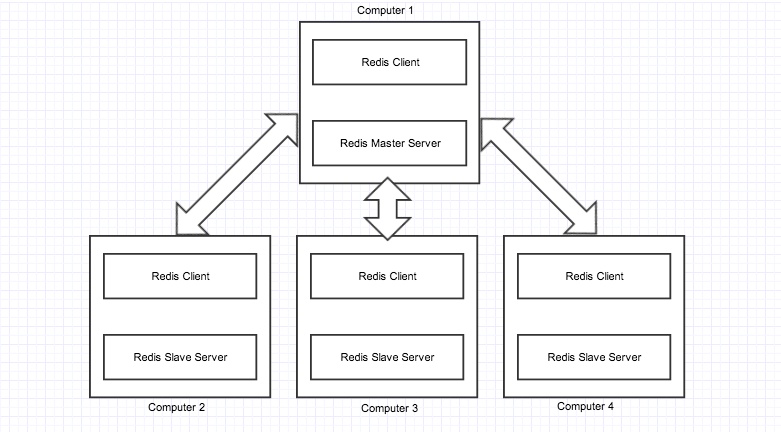
https://redis.io/topics/persistence

* RDB Mechanism: tạo 1 bản sao toàn bộ dữ liệu từ ram và cất sang bộ nhớ đĩa. Điều này được thực hiện định kỳ, do đó sẽ mất dữ liệu kể từ bản sao cuối. Phù hợp cho việc phục hồi từ thảm họa, dể dàng cất dữ trên trung tâm dữ liệu. Khởi động nhanh.
* AOF: ghi log tất cả các hoạt động ghi mà được nhận bởi server, mọi thứ đều bền bỉ nhưng file có kích thước lớn hơn so với kiểu RDB.
* SAVE Command: Người dùng ra lệnh cho máy chủ thực hiện RDB bất cứ lúc nào.

Sao lưu và phục hồi: Redis không cung cấp bất kỳ cơ chế nào cho việc sao lưu và khôi phục kho dữ liệu. Do đó, nếu có bất kỳ tai nạn đĩa cứng hoặc bất kỳ loại thiên tai khác thì tất cả dữ liệu sẽ bị mất. Bạn cần phải sử dụng một số phần mềm sao lưu và khôi phục máy chủ của bên thứ ba để khắc phục sự cố.

Redis Replication(mô phỏng, nhân rộng): là một kỹ thuật liên quan đến nhiều máy tính để cho phép chia sẻ khả năng truy cập dữ liệu. Trong một môi trường nhân rộng, nhiều máy tính chia sẻ cùng một dữ liệu với nhau để ngay cả khi một vài máy tính bị hỏng, tất cả dữ liệu sẽ có sẵn.

https://redis.io/topics/replication



Tất cả các nô lệ có chứa chính xác cùng một dữ liệu như tổng thể. Có thể có nhiều như nô lệ cho mỗi máy chủ chính. Khi một nô lệ mới được đưa vào môi trường, tổng thể sẽ tự động đồng bộ hóa tất cả dữ liệu với nô lệ.

Các Slave có cùng dữ liệu với nhau, khi một Slave được đưa vào môi trường thì hệ thống sẽ tự đồng bộ hóa dữ liệu với Slave mới. Mọi truy vấn được đưa đến master, master sẽ thực hiện các hoạt động. Khi có thao tác ghi master sẽ phân phối dữ liệu đó đến Slave

Master chết thì đưa Slave củ lên làm master, công việc được tiếp tục.

Gom nhóm trong Redis: