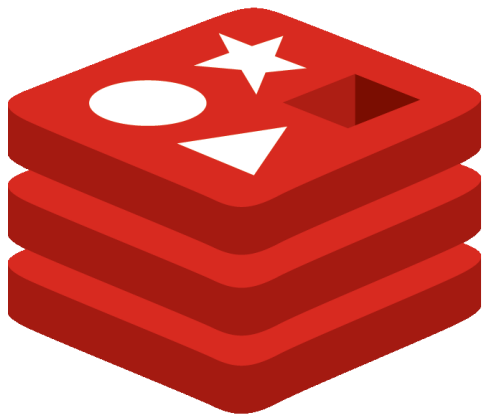


I used redis

and I liked it!



redis

What is redis?

- not a (traditional) disk data store
- shared memory
- key/value store
- single-threaded
- drivers for all major languages (and most minor ones too)

Features

- pubsub
- stored Lua functions
- transactions
- master/slave replication
- high-availability (failover) via redis-sentinel

Features

- snapshot persistence
- append-only log persistence
- clustering (sharding) [beta]

Why would I use it?

- memory is fast
- no read/write IO latency
- caching
- pubsub

Keys

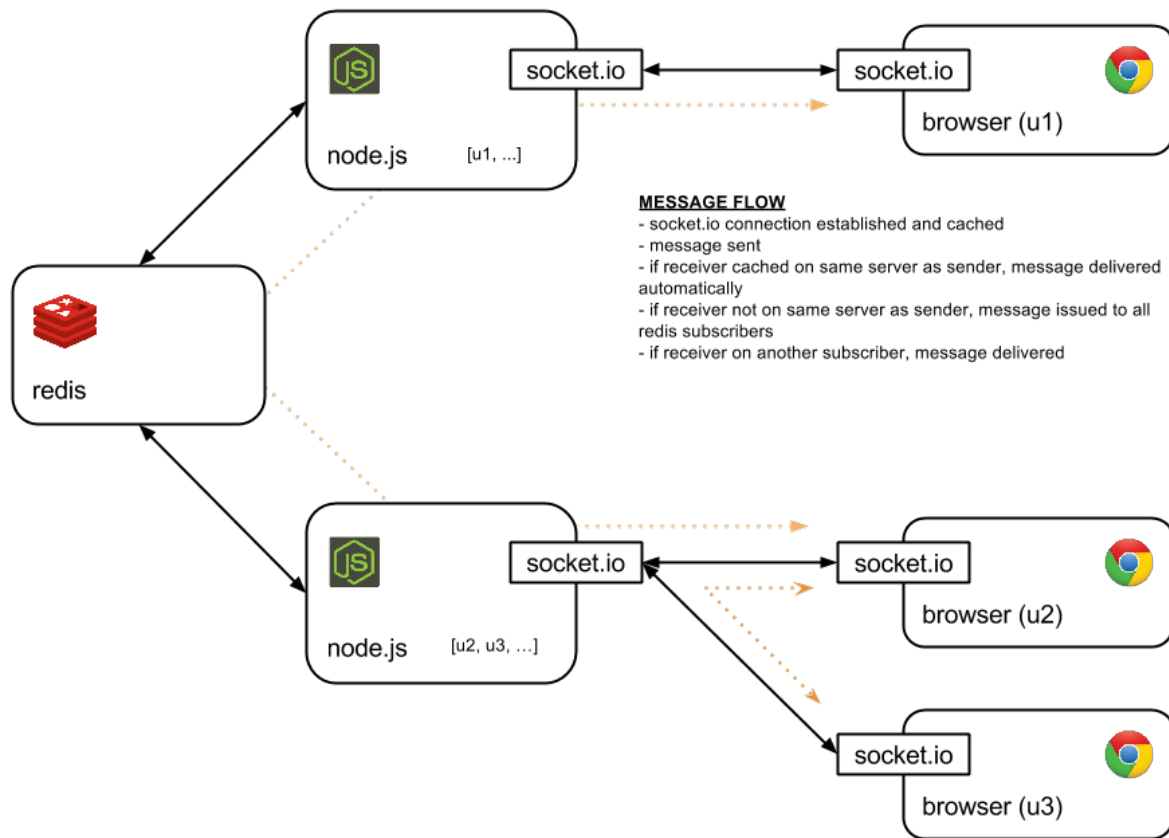
- keys are KEY!
- keys are conceptually similar to RESTful URLs
- `keys` command

Values (data types)

- ***string*** (`"`)
- ***hash*** (key/value pairs)
- ***list*** (array of values)
- ***set*** (array of unique values)
- ***sorted set*** (array of sorted unique values)

My redis use case

- chat system with redundant node+socketio servers
- chat servers need to communicate with each other
- cloud provider doesn't support AMQP messaging (e.g. rabbitMQ), but does support redis



Resources

- *The Little Redis Book* - Karl Seguin
- redis.io
- [youtube](https://www.youtube.com/watch?v=U0133333333)