Redis Replication

How Redis supports high availability and failover with replication

leader follower (master-replica)

This system works using three main mechanisms:

1. When a master and a replica instances are well-connected, the master keeps the replica updated by sending a stream of commands to the replica to replicate the effects on the dataset happening in the master side due to: client writes, keys expired or evicted, any other action changing the master dataset.
2. When the link between the master and the replica breaks, for network issues or because a timeout is sensed in the master or the replica, the replica reconnects and attempts to proceed with a partial resynchronization: it means that it will try to just obtain the part of the stream of commands it missed during the disconnection.
3. When a partial resynchronization is not possible, the replica will ask for a full resynchronization. This will involve a more complex process in which the master needs to create a snapshot of all its data, send it to the replica, and then continue sending the stream of commands as the dataset changes.

Important facts about Redis replication

- Redis uses asynchronous replication, with asynchronous replica-to-master acknowledges of the amount of data processed.
- A master can have multiple replicas.
- Redis replication is non-blocking on the master side. This means that the master will continue to handle queries when one or more replicas perform the initial synchronization or a partial resynchronization.
