

Redis, distributed or not?

Asked 11 years, 4 months ago Modified 5 years, 2 months ago

Viewed 28k times



42



Question 1: I know Redis load all the data into memory, thus improving the read/write speed. So, does it mean that if my memory size is 2G, the maximum dataset size should not be larger than 2G? Now my database has a 100G+ data, the memory of my server cannot be larger than 32G, so, Redis is no longer suitable for me?

Question 2: Is Redis a distributed system or not? When I use google to search for redis's CAP property, it says Redis is not a distributed system, so, it has nothing to do with [CAP](#) . But from [Wikipedia](#), it says it has a master-slave architecture, one master with many slaves. How confusing.

nosql

redis

distributed

Share Follow

edited Aug 22, 2017 at 0:52



Bob Fang

7,361 ● 10 ● 43 ● 77

asked Aug 22, 2013 at 9:33



wuchang

3,069 ● 9 ● 44 ● 68

2 Answers

Sorted by:

Highest score (default)



72



Regarding question 1, Redis is an in-memory store with some persistency capabilities. All your dataset should fit in memory. A single instance is therefore limited by the maximum memory of your server. Now, you can also shard the data to several Redis instances, running on multiple servers. Provided you have the budget for it, it is perfectly possible to store 100GB - 1TB of data on a set of Redis boxes. Please note sharding is not automatic: it has to be implemented by the client or the application. It also puts some constraints on the operations you can do on your data (for instance it would not be possible on server-side to calculate the intersections of two sets hosted by different Redis instances).

Regarding question 2, a single Redis instance is not a distributed system. It is a remote centralized store. Now by using several Redis instances, you can build a distributed system. Because it is a do-it-yourself approach, you can decide to make it a CP or AP system.

A single instance can replicate its activity to slave instances (which are therefore eventually consistent with the master). The application can choose to always connect to the master for read and write. In that case, you may get a CP system. It can also write on the master, and read from all instances (including slaves), so you may get an AP system. I said "may", because it requires some significant work to build such systems on top of Redis.

You can mix sharding and master/slave replication to build the distributed system you need. However, Redis only provides basic bricks to do this. Especially, it does offer very little to deal with resiliency and HA (and address the P in the CAP theorem). IMO, [Redis sentinel](#) alone is not enough to support an HA Redis configuration, since it only covers role management. You need to complement it with a resource manager, and put a lot of logic into the client/application.

There is an on-going project called [Redis Cluster](#), whose purpose is to provide a minimalistic ready-to-use distributed system, but it still lacks of lot of things, and is not usable yet for production purpose.

If you need an off-the-shelf distributed store, Redis is probably not a good option. You will be better served by Cassandra, Riak, MongoDB, Couchbase, Aerospike, MySQL Cluster, Oracle NoSQL, etc ... However, if you want to build your own specialized system, Redis is an excellent component to build upon.

Share Follow

edited Aug 22, 2013 at 18:35

answered Aug 22, 2013 at 10:10



Didier Spezia

73.1k ● 12 ● 193 ● 157

Well ,your information is really important and on the point for me.From your description , redis is really not suitable for me since we have on interest on building a specialized

system, just a off-the-shelf NoSQL database storing 100G+ data. I am now pay more attention to MongoDB and CouchDB. – [wuchang](#) Aug 22, 2013 at 10:43

- 1 Also, I think Redis is a good solution for **caching** and not storing the whole data. So, it's possible to store a fragmentation of data for frequent use and access. – [Babak](#) Dec 8, 2014 at 9:27
-

- 2 Since the answer is a bit old, it's good to mention that now Redis supports an automated sharding solution across the cluster: redis.io/topics/cluster-tutorial – [Erfun](#) Jun 5, 2020 at 20:30
-

Why is reading from master ONLY CP? It's no longer even distributed if we ignore the slaves. – [Kun](#) May 21, 2021 at 8:23



3



Here is a useful link, **Redis Cluster Tutorial**:

<https://redis.io/topics/cluster-tutorial>

You might also benefit from looking at the **Facebook solution with memcache**:

<https://engineering.fb.com/web/introducing-mcrouter-a-memcached-protocol-router-for-scaling-memcached-deployments/>

Share Follow

answered Sep 26, 2019 at 9:23



kintsukuroi

1,479 ● 18 ● 16