



Caching

Let's learn how Chubby implements its cache.

We'll cover the following ^

- Chubby cache
- Cache invalidation

Chubby cache#

In Chubby, caching plays an important role, as read requests greatly outnumber write requests. To reduce read traffic, Chubby clients cache file contents, node metadata, and information on open handles in a consistent, write-through cache in the client's memory. Because of this caching, Chubby must maintain consistency between a file and a cache as well as between the different replicas of the file. Chubby clients maintain their cache by a lease mechanism and flush the cache when the lease expires.

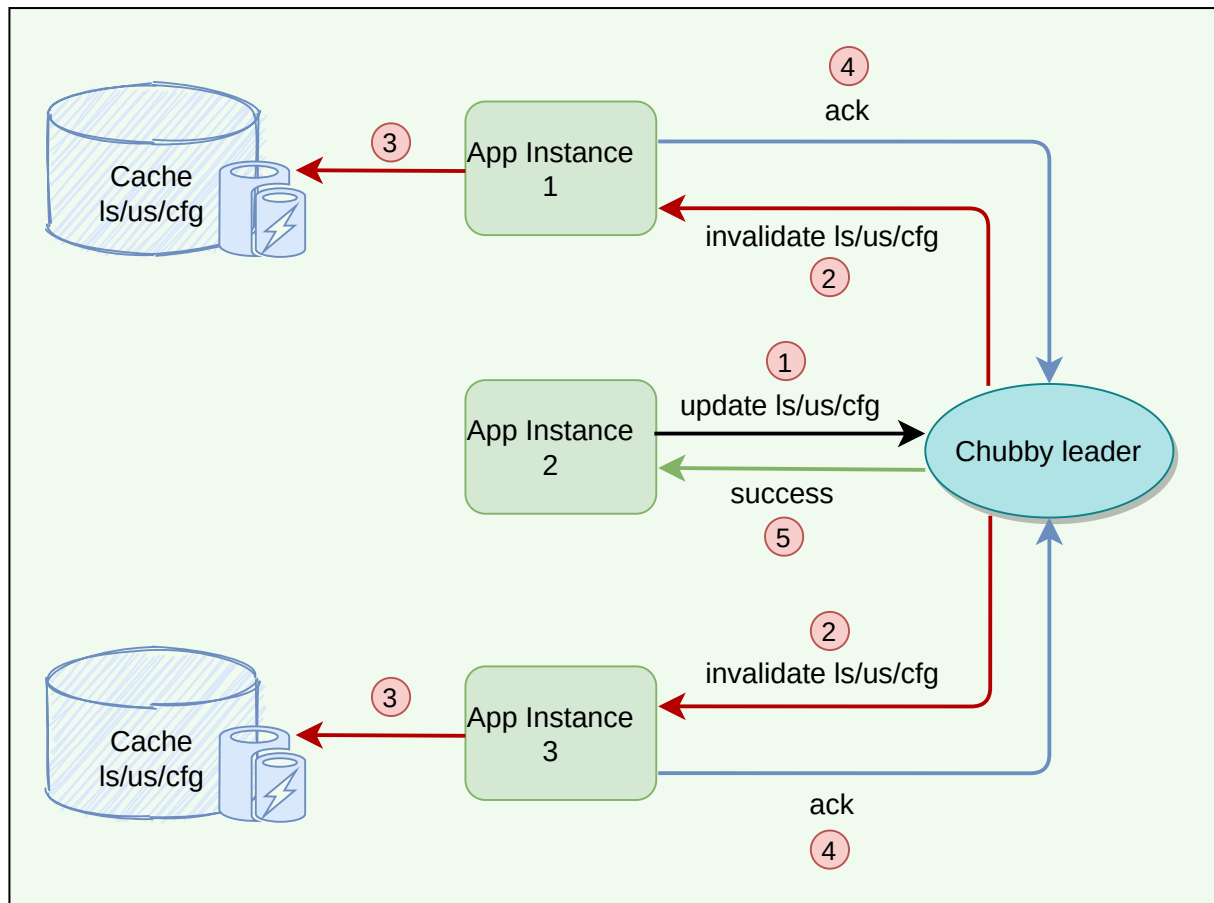
Cache invalidation#

Below is the protocol for invalidating the cache when file data or metadata is changed:

- Master receives a request to change file contents or node metadata.
- Master blocks modification and sends cache invalidations to all clients who have cached it. For this, the master must maintain a list

of each client's cache contents.

- For efficiency, the invalidation requests are piggybacked onto KeepAlive replies from the master.
- Clients receive the invalidation signal, flushes the cache, and sends an acknowledgment to the master with its next KeepAlive call.
- Once acknowledgments are received from each active client, the master proceeds with the modification. The master updates its local database and sends an update request to the replicas.
- After receiving acknowledgments from the majority of replicas in the cell, the master sends an acknowledgment to the client who initiated the write.



Cache invalidation

Question: While the master is waiting for acknowledgments, are other clients allowed to read the file?

Answer: During the time the master is waiting for the acknowledgments from clients, the file is treated as ‘uncachable.’ This means that the clients can still read the file but will not cache it. This approach ensures that reads always get processed without any delay. This is useful because reads outnumber writes.

Question: Are clients allowed to cache locks? If yes, how is it used?

Answer: Chubby allows its clients to cache locks, which means the client can hold locks longer than necessary, hoping that they can be used again by the same client.

Question: Are clients allowed to cache open handles?

Answer: Chubby allows its clients to cache open handles. This way, if a client tries to open a file it has opened previously, only the first `open()` call goes to the master.

[← Back](#)[Master Election and Chubby Events](#)[Next →](#)[Database](#)[Mark as Completed](#)[Report an Issue](#)