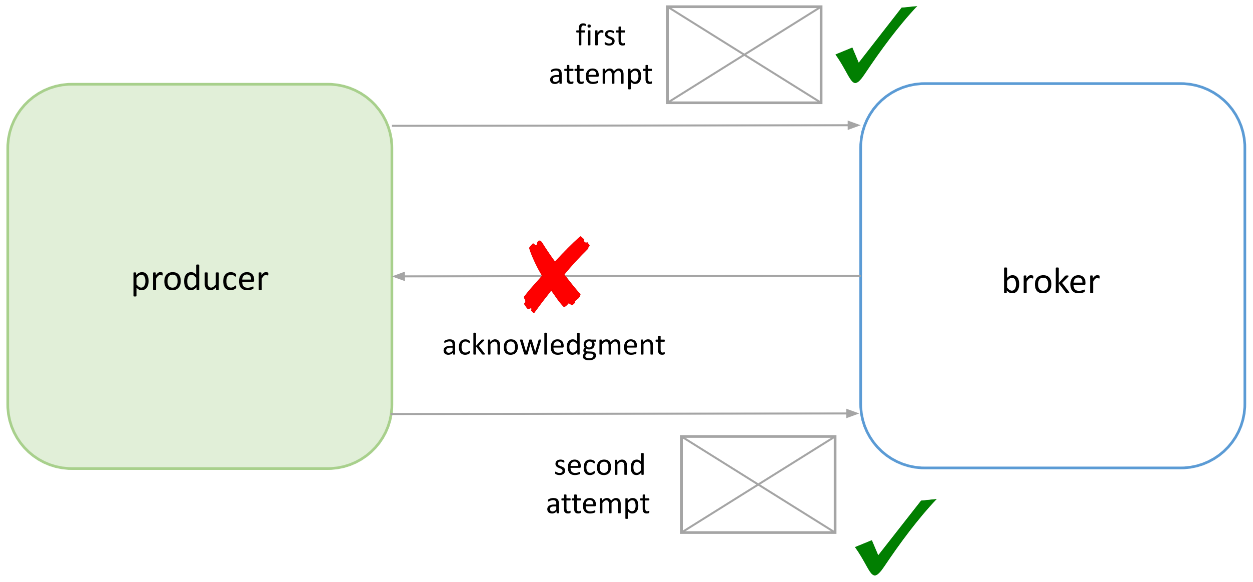
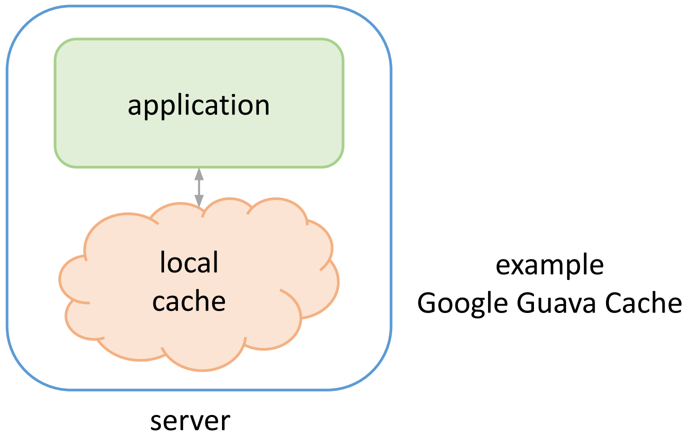
Deduplication cache



Deduplication cache

local cache

(private)

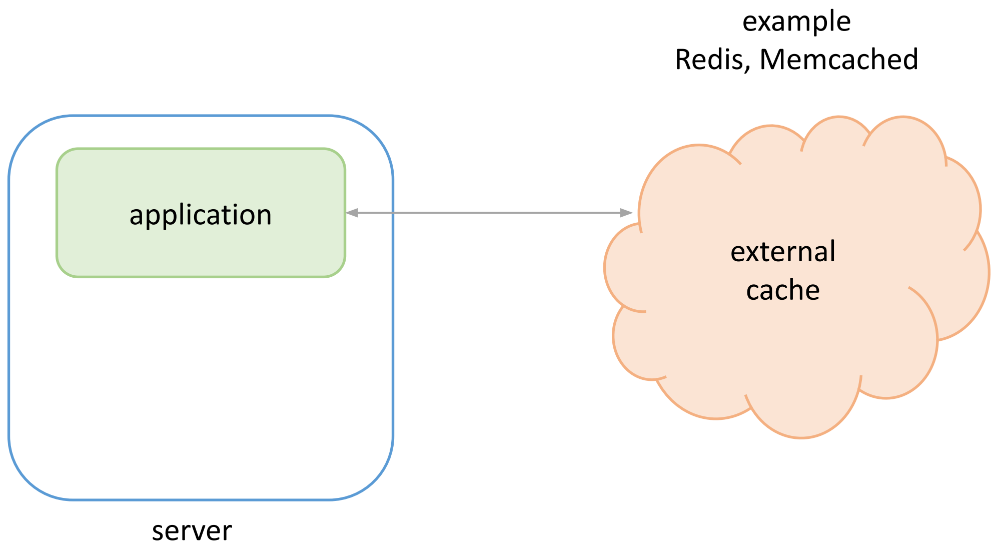


- Simple and extremely fast.

- Not scalable, as memory size is limited.
- Zero fault-tolerance and durability.

external cache

(shared, remote)



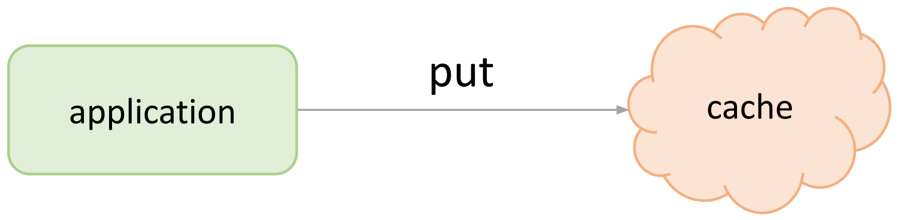
- Scalable (total capacity is the sum of each server capacity).
- Fault-tolerant and durable (if supports replication).

- Requires maintenance.

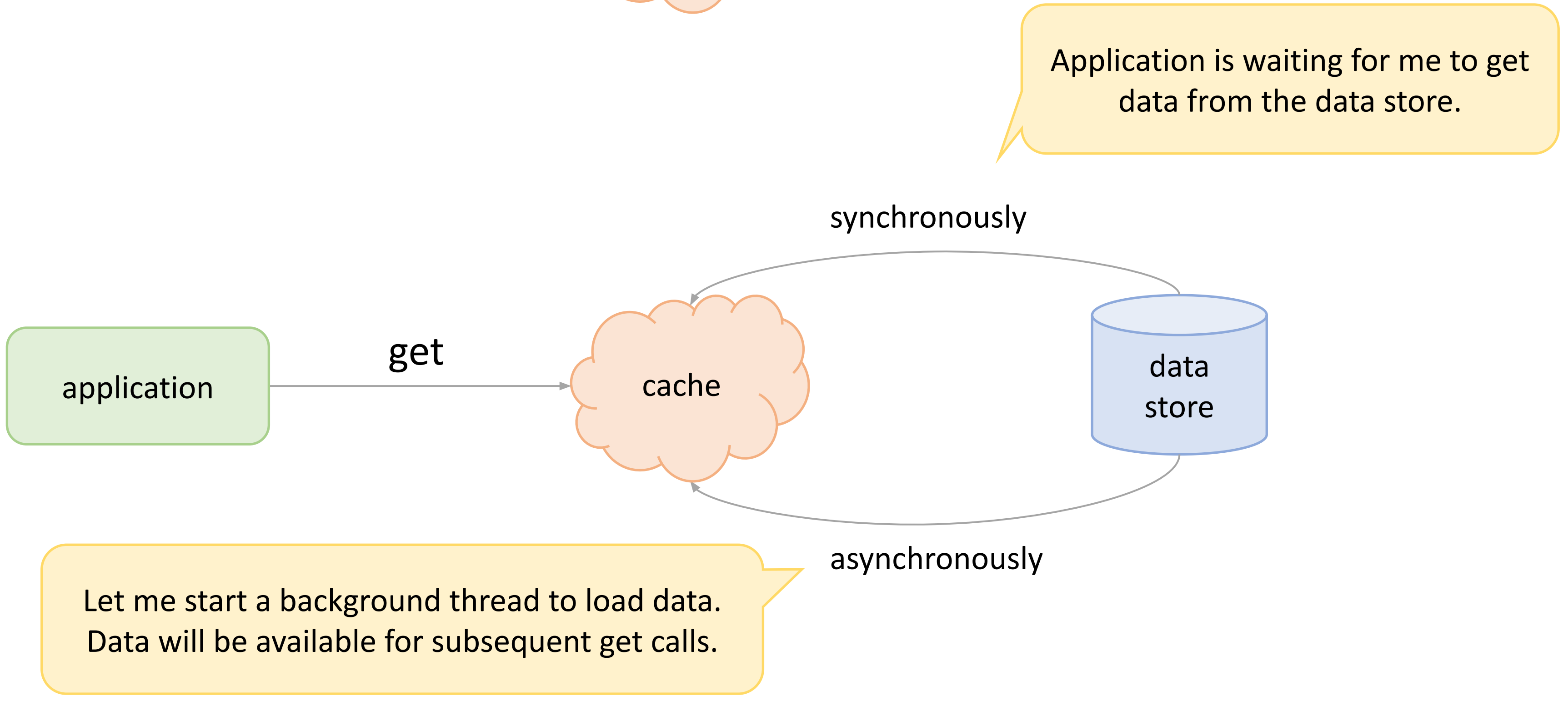
Deduplication cache

How do we add data to cache?

explicitly



implicitly


Application is waiting for me to get data from the data store.
synchronously
application
get
cache
data store
asynchronously
Let me start a background thread to load data. Data will be available for subsequent get calls.

Deduplication cache

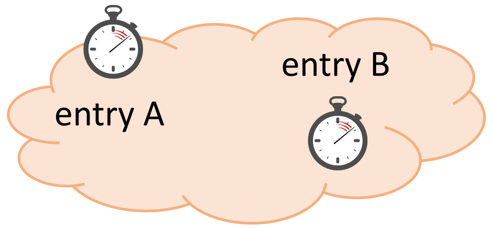
How is data evicted from cache?

size-based eviction

eviction (replacement) policies

- evict entries that haven't been used recently (**LRU**, least recently used)
- evict entries that were used least often (**LFU**, least frequently used)

time-based eviction



passive expiration

when entry is accessed

active expiration

background thread that runs at regular intervals

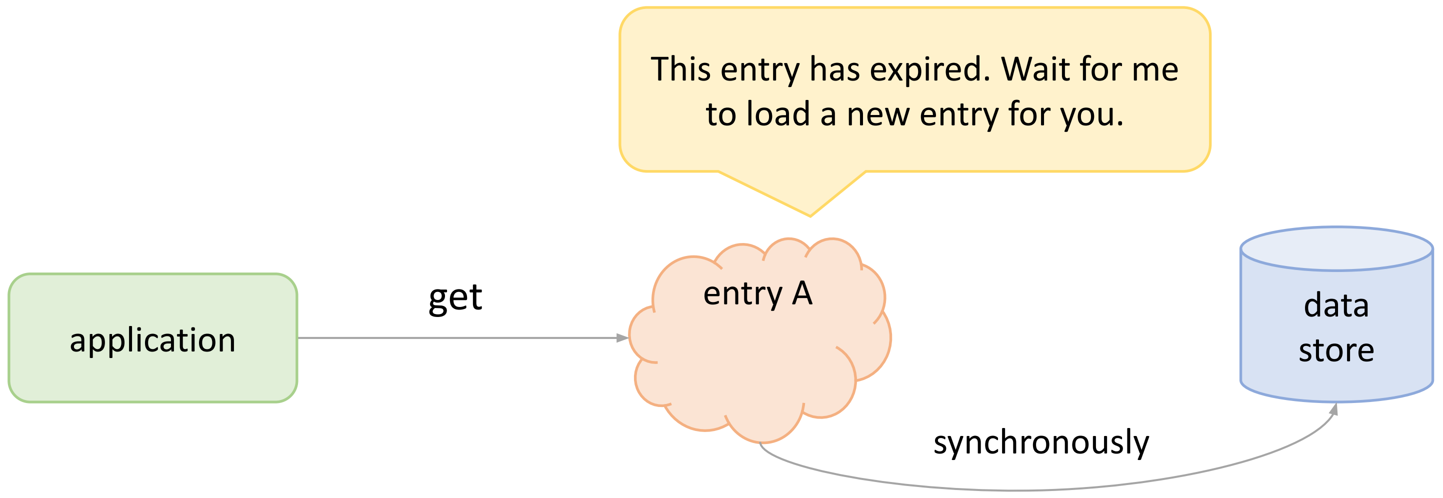
explicit removal

```
cache.invalidate(key)
```

Deduplication cache

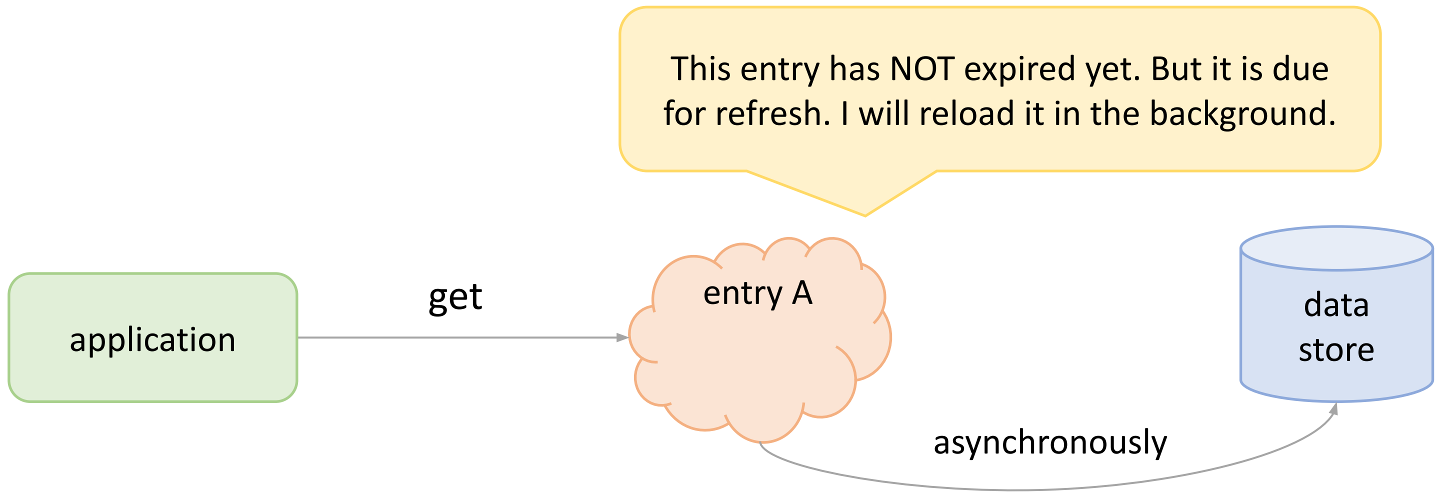
expiration

(time-based eviction)


This entry has expired. Wait for me to load a new entry for you.
application
get
entry A
data store
synchronously

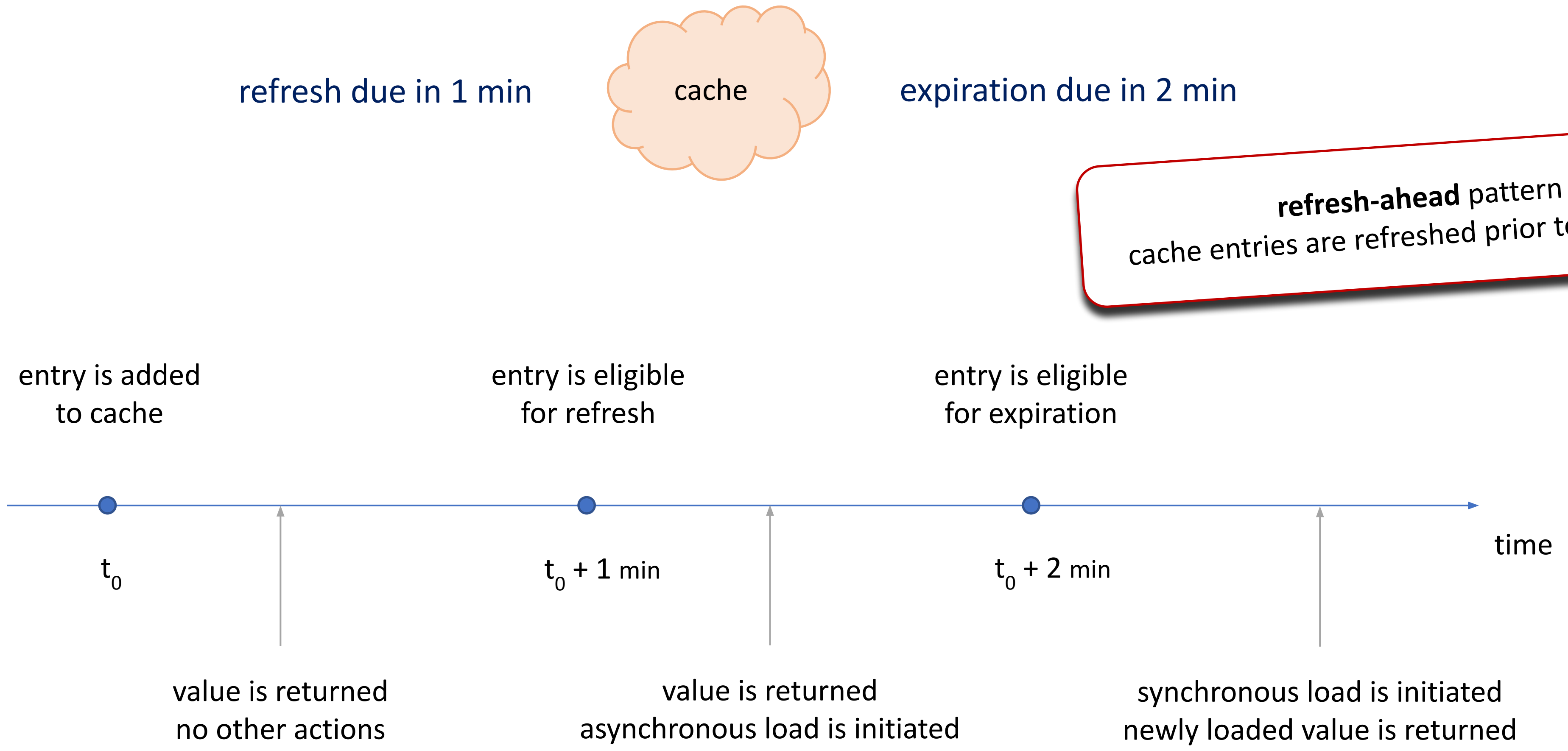
refresh technique is especially useful for frequently accessed keys ("hot" keys)

refresh

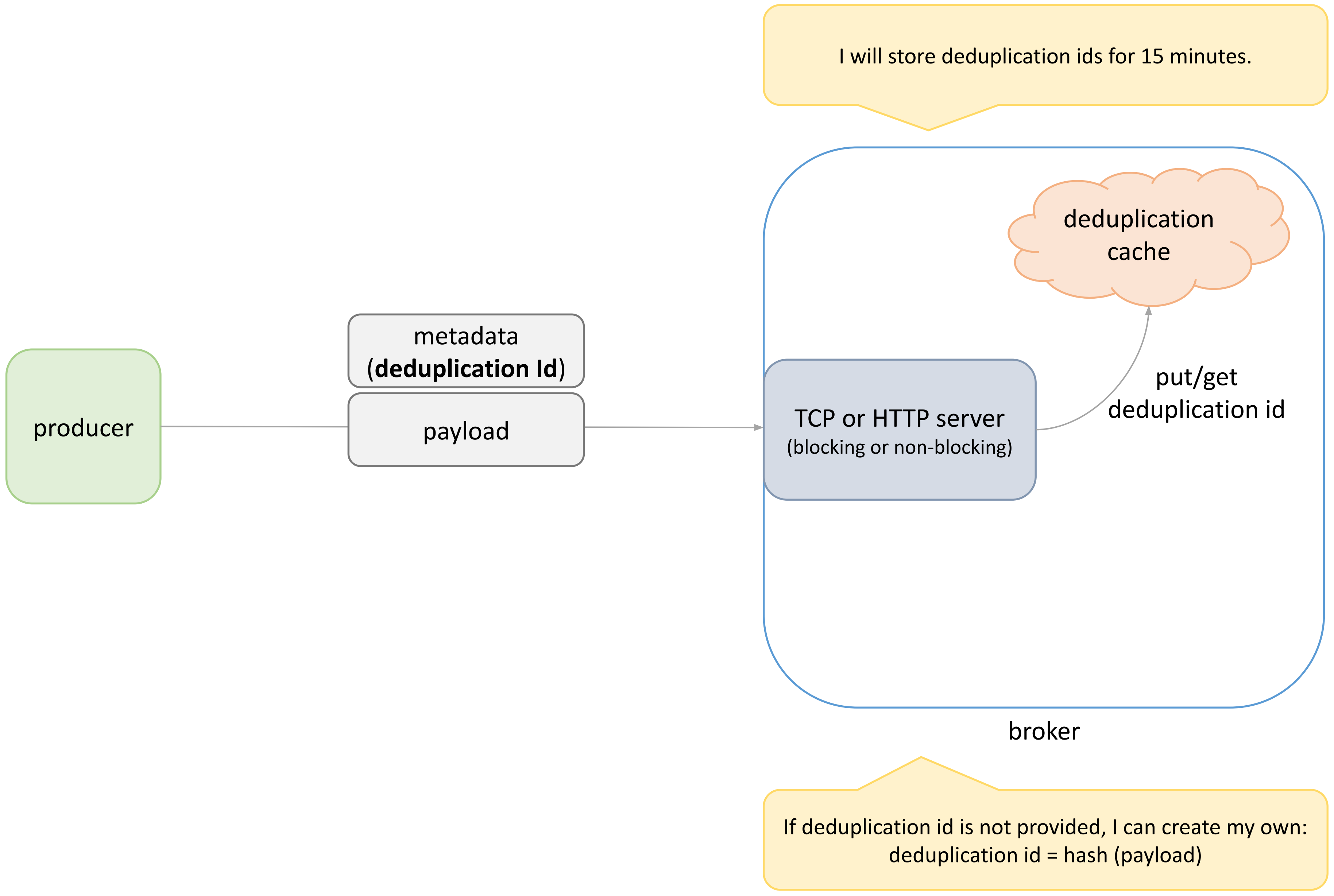

This entry has NOT expired yet. But it is due for refresh. I will reload it in the background.
application
get
entry A
data store
asynchronously

Deduplication cache

refresh-ahead pattern
cache entries are refreshed prior to expiration



Deduplication cache


I will store deduplication ids for 15 minutes.
producer
metadata (**deduplication Id**)
payload
TCP or HTTP server (blocking or non-blocking)
deduplication cache
put/get deduplication id
broker
If deduplication id is not provided, I can create my own: deduplication id = hash (payload)