Making your API calls wicked fast with Redis

- Lindsey Brockman

Outline:

- ★ What is Redis?
- ★ What can I use instead?
- ★ When should I use caching?
- ★ Pros/Cons
- ★ Demo: Redis + python-redis + Flask

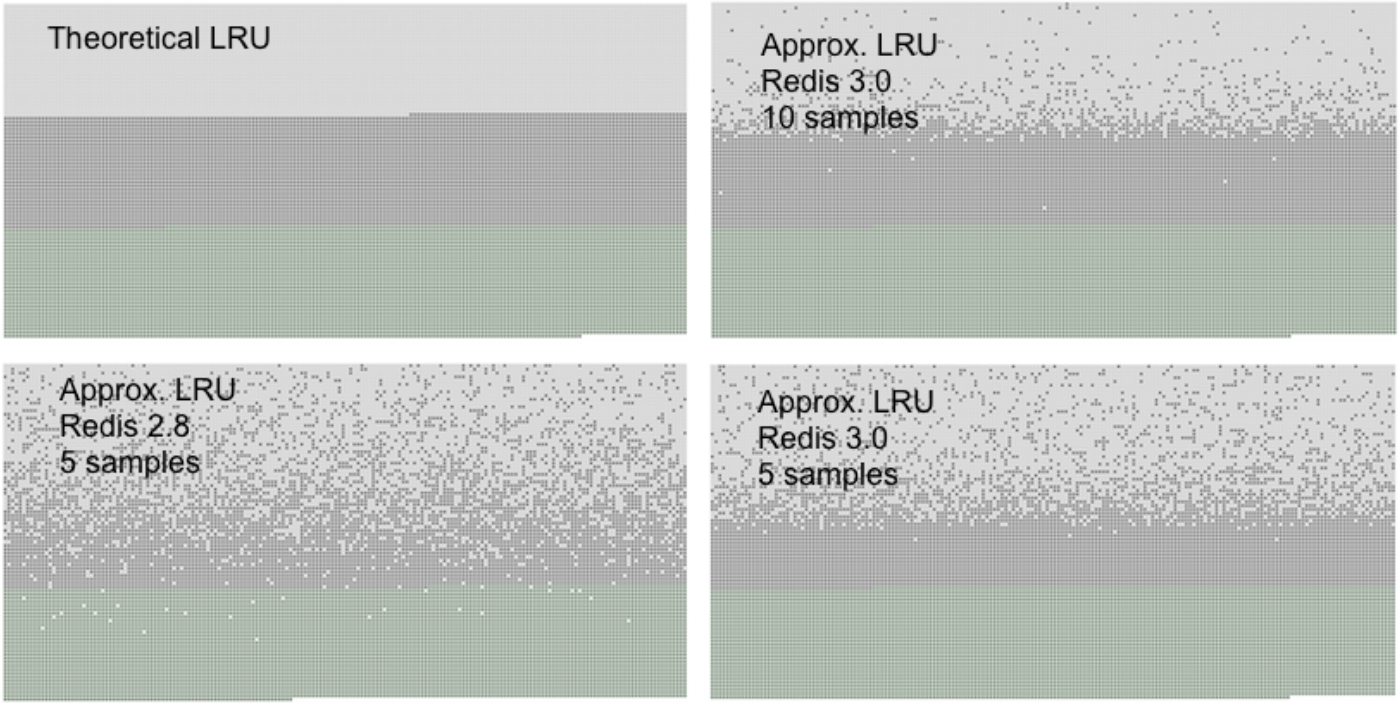
Whoa backup... what is Redis?

- ★ Key value store
- ★ Free & Open Source
- ★ Data Structure Values:
  - ○ lists
  - ○ sets
  - ○ sorted sets
  - ○ hashes
  - ○ bit arrays

Redis as an LRU cache

- ★ LRU = Least Recently Used
- ★ Redis uses an approximation of the LRU algorithm
- ★ Configurable precision with `maxmemory-samples` configuration directive

Redis LRU visualization



Redis data persistence

- ★ Snapshots (.rdb)
  - Good for rollbacks
  - Possibility of losing data
- ★ Append Only File (AOF)
  - Bad for rollbacks
  - Great for persisting current dataset as is

Redis Alternatives

memcached

- ★ Free and open source
- ★ Every command is fast: O(1)
- ★ Serialized data only
  (no data structures)
- ★ No data persistence

@functools.lru_cache()

- ★ Python 3.2+
- ★ clear_cache()
- ★ cache_info()

```
@lru_cache(maxsize=None)
def fib(n):
    if n < 2:
        return n
    return fib(n-1) + fib(n-2)

>>> [fib(n) for n in range(16)]
[0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55,
89, 144, 233, 377, 610]

>>> fib.cache_info()
CacheInfo(hits=28, misses=16,
maxsize=None, currsize=16)


#https://docs.python.org/3.
4/library/functools.html#functools.
lru_cache
```

To cache?

- ★ External API calls with slow response times
- ★ Avoid API rate limiting
- ★ ...but only if you’re allowed to!
- ★ Alleviate database load locally
- ★ Same end-points hit frequently

Or not to cache?

- ★ Your application is already fast
- ★ You don’t want added complexity
- ★ You have concerns about persistence
- ★ You can just optimize your db instead:
  - ○ check slow query logs
  - ○ use `EXPLAIN` or `EXPLAIN ANALYZE`
  - ○ add an index

Pros:

- Faster access to cached data
- Lessened DB load
- Faster external API calls
- Ability to avoid rate limiting

Cons:

- Added code complexity
- Cache invalidation blues
- Debugging complications

Demo !

Questions?

Helpful Links!

- ★ redis.io
- ★ redis.io/topics/lru-cache
- ★ cheatography.com/tasjaevan/cheat-sheets/redis/
- ★ memcached.org/
- ★ docs.python.org/3.4/library/functools.html#functools.lru_cache