Deploying Large Scale Web Applications

Thierry Sans

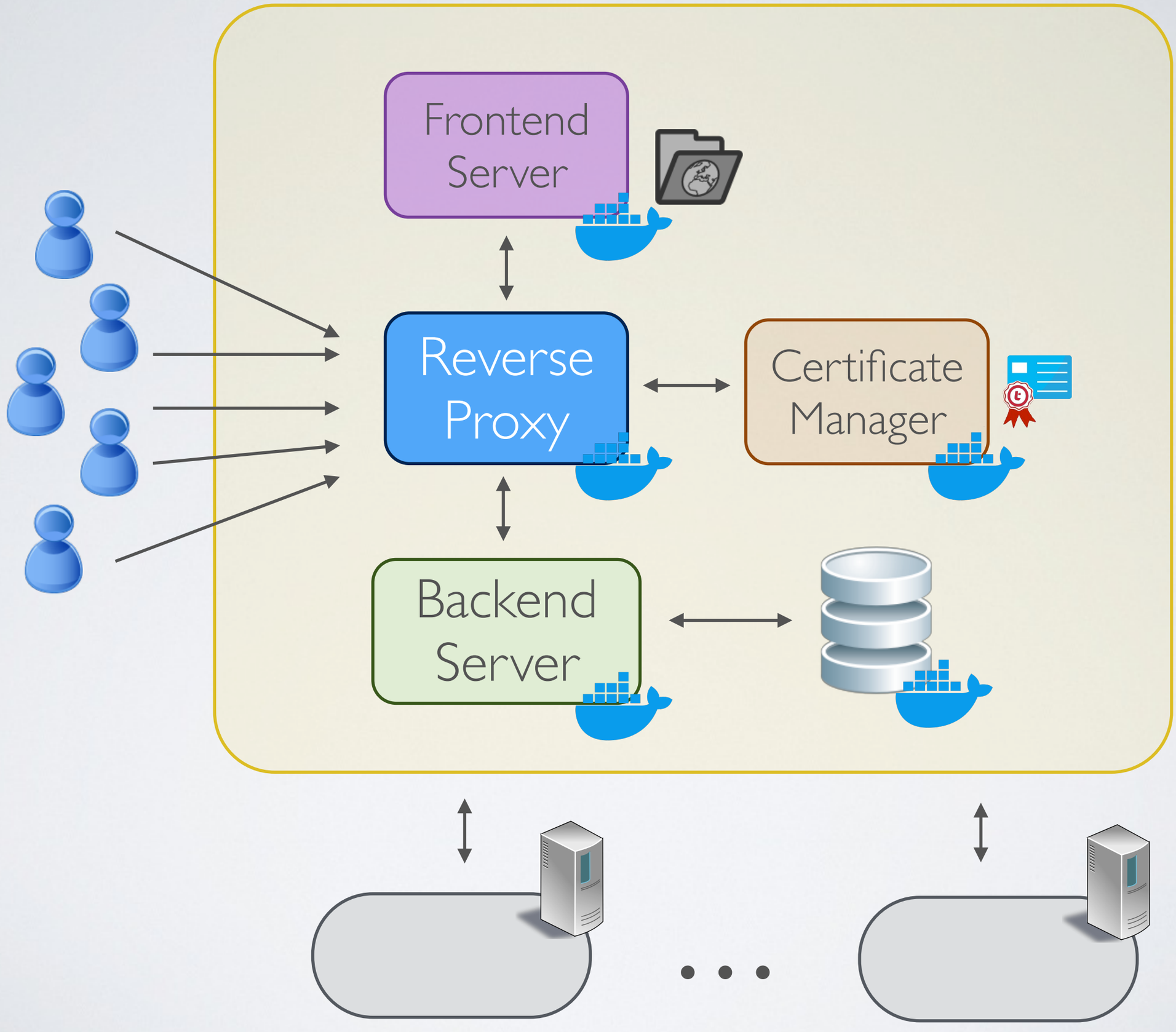
Users respond to speed

"Amazon found every 100ms of latency cost them 1% in sales"

"Google found an extra .5 seconds in search page generation time dropped traffic by 20%"

http://blog.gigaspaces.com/amazon-found-every-100ms-of-latency-cost-them-1-in-sales/

Our microservice deployment (so far)

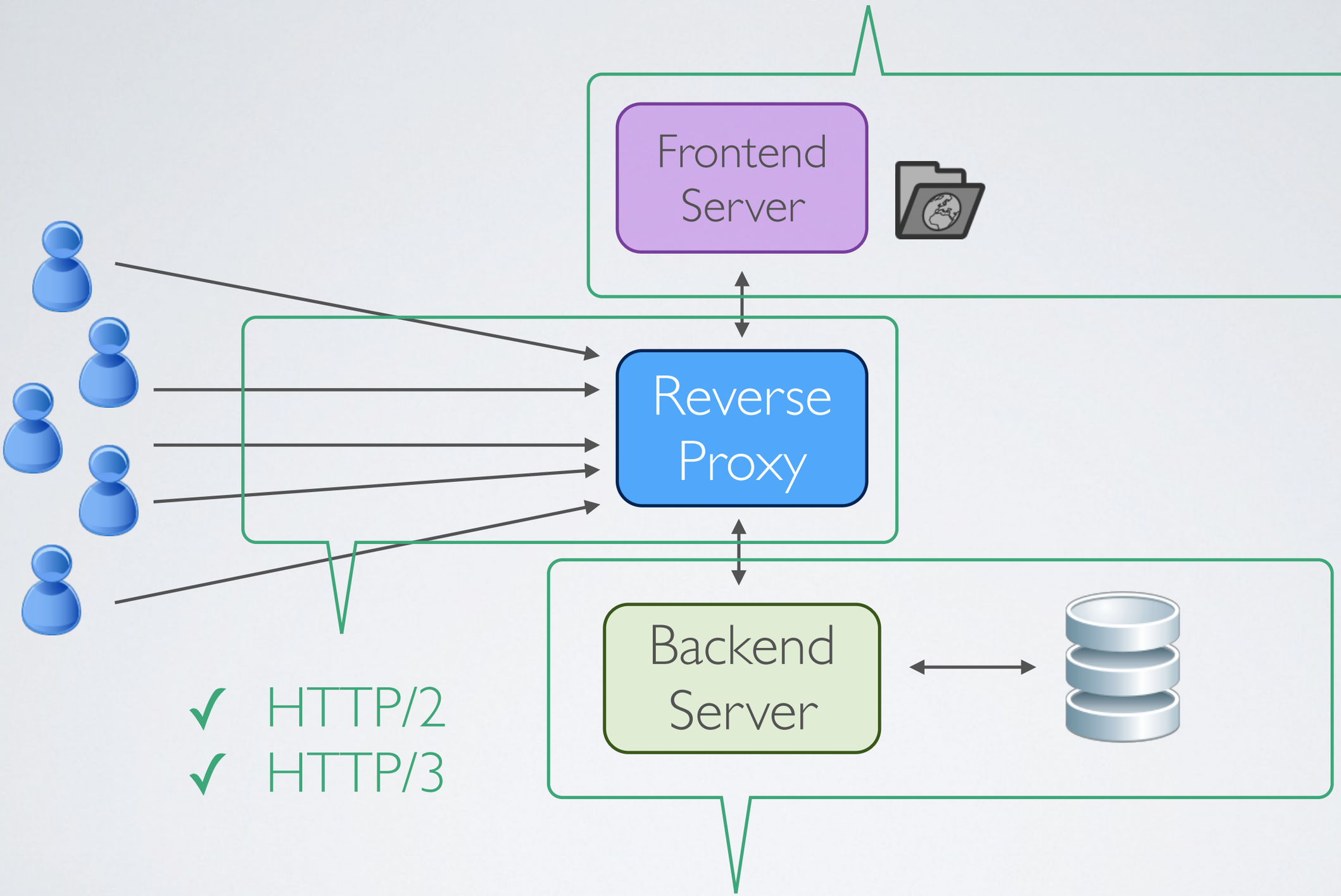


Problems

- How to increase the throughput?
- How to scale to serve millions of users?

Solutions

- ✓ Web Packing
- ✓ Progressive Web Applications (PWA)



- ✓ HTTP/2
- ✓ HTTP/3

- ✓ Faster web caching
- ✓ Better scalability with load balancer and CDN

Backend Web Caching

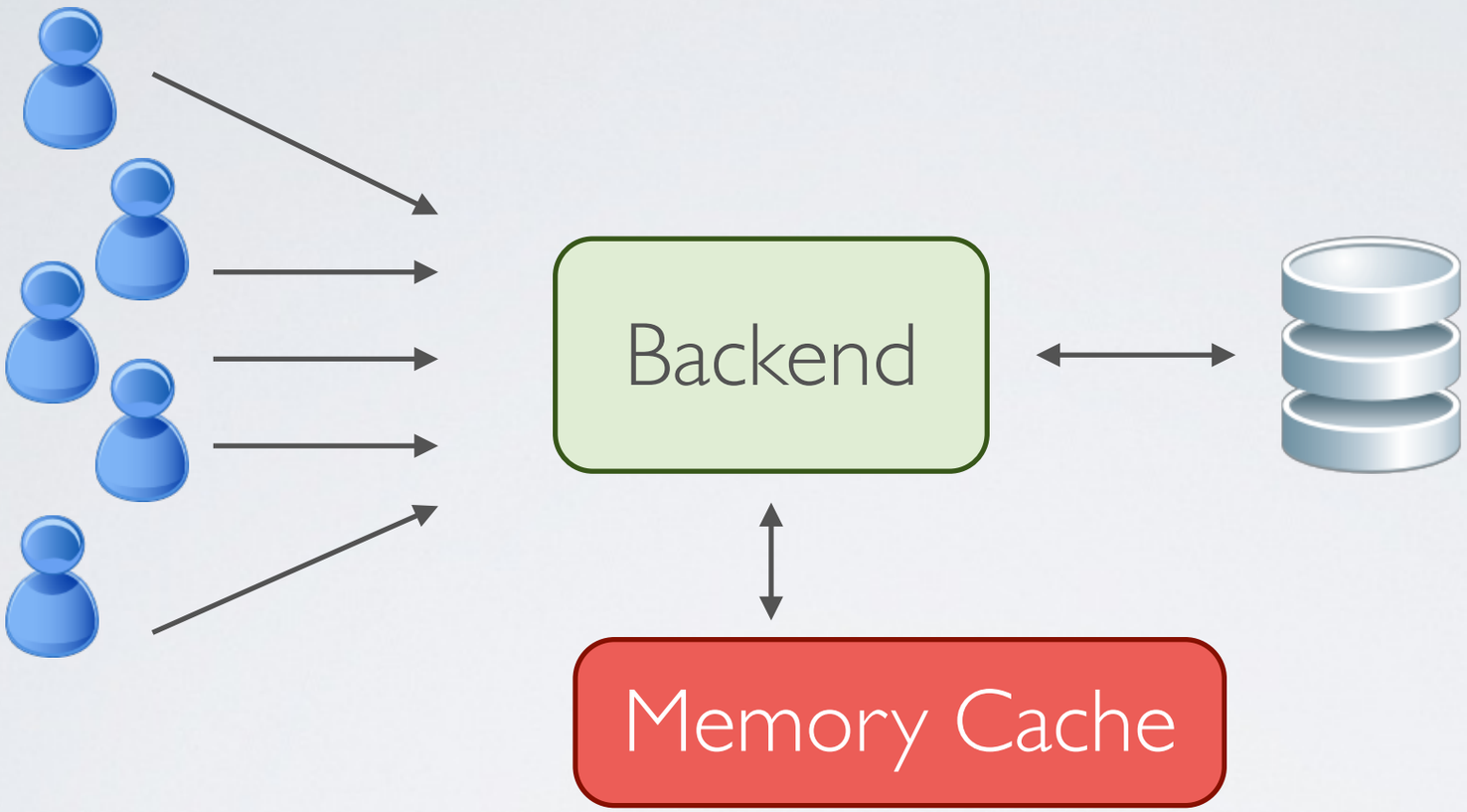
How to improve response time?

Processing the request means:

1. Parse the HTTP request
2. Map the URL to the handler
3. Query the database or third-party API
4. Compute the HTTP response

DB and API accesses are expensive (time and money when your host charges you each access)

Fine-grained caching with the web application



Cache controlled by the program

- ◉ Specific for each app
- ✓ Good for caching database requests and storing sessions
- ➡ Popular memory cache : *Memcached*

Distributed Shared Cache : Memcached

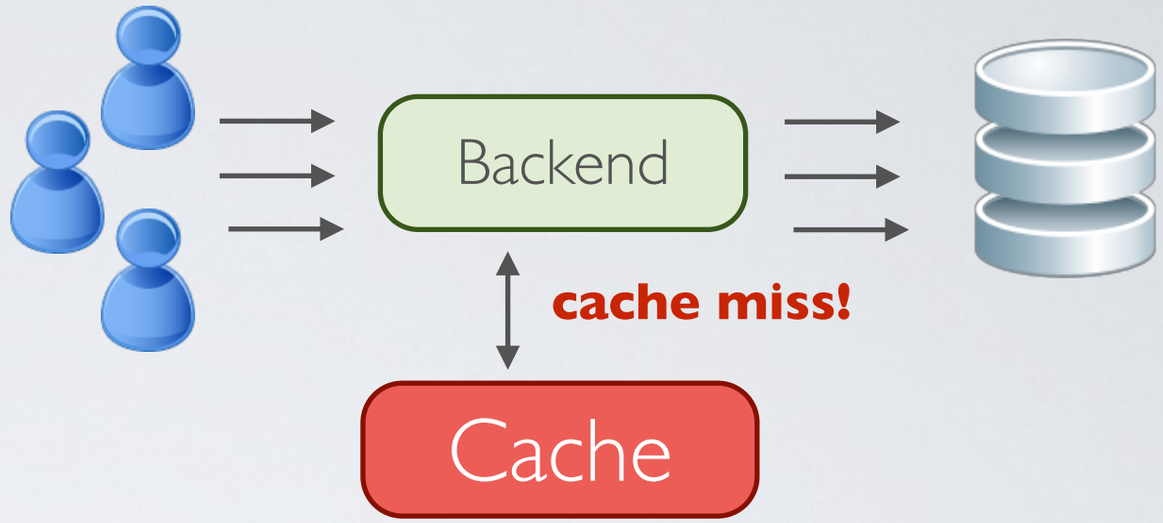
http://memcached.org/

- Store key/value pairs in memory
- Throw away data that is the least recently used

A typical cache algorithm

```
retrieve from cache
if data not in cache:
   # cache miss
   query the database or API
   update the cache
return result
```

Cache Stampede (a.k.a dog piling)



Problem:

Multiple concurrent requests doing the same request because cache was cleared

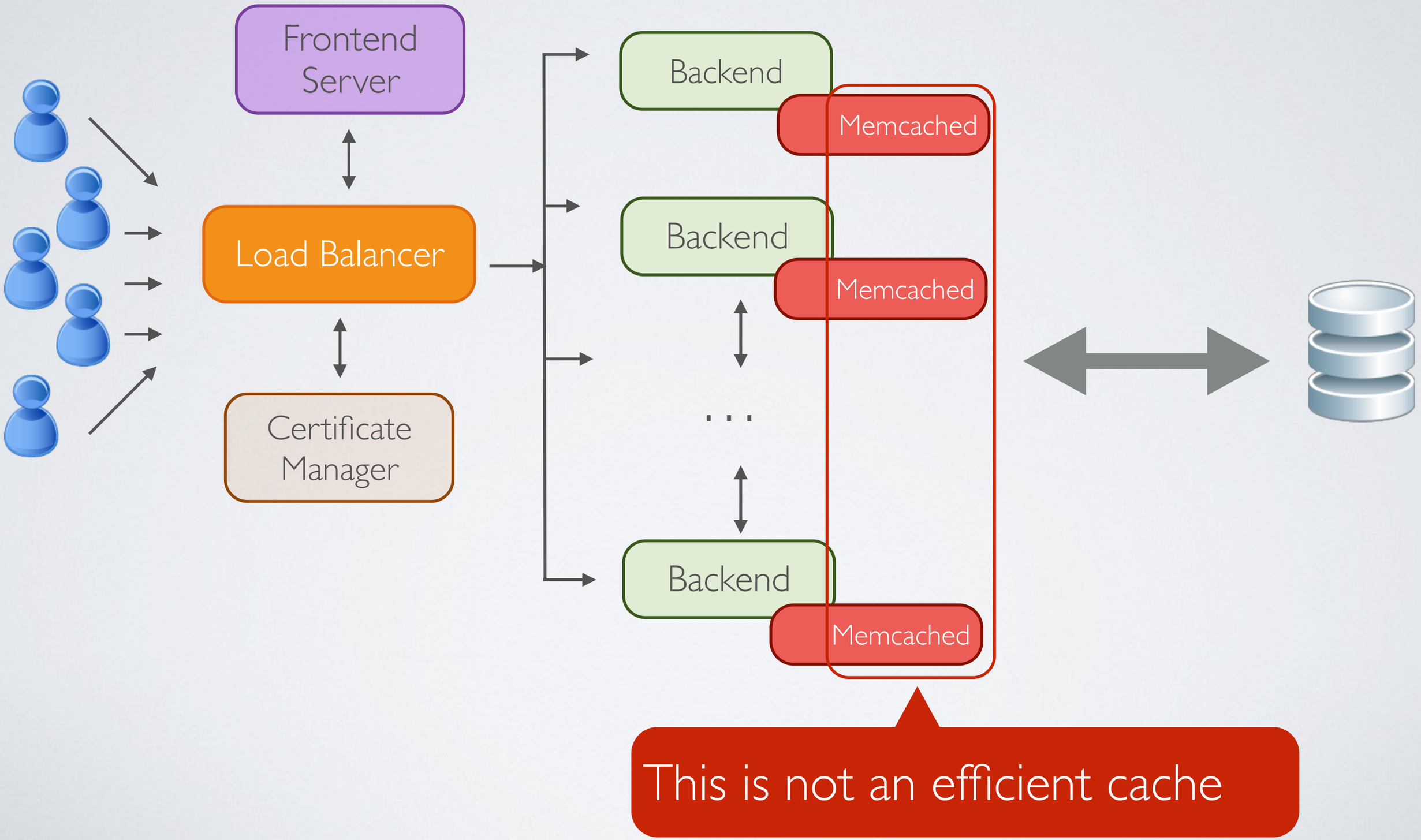
Solution:

- update the cache instead of clearing it after an insert
- a page view will never query the database

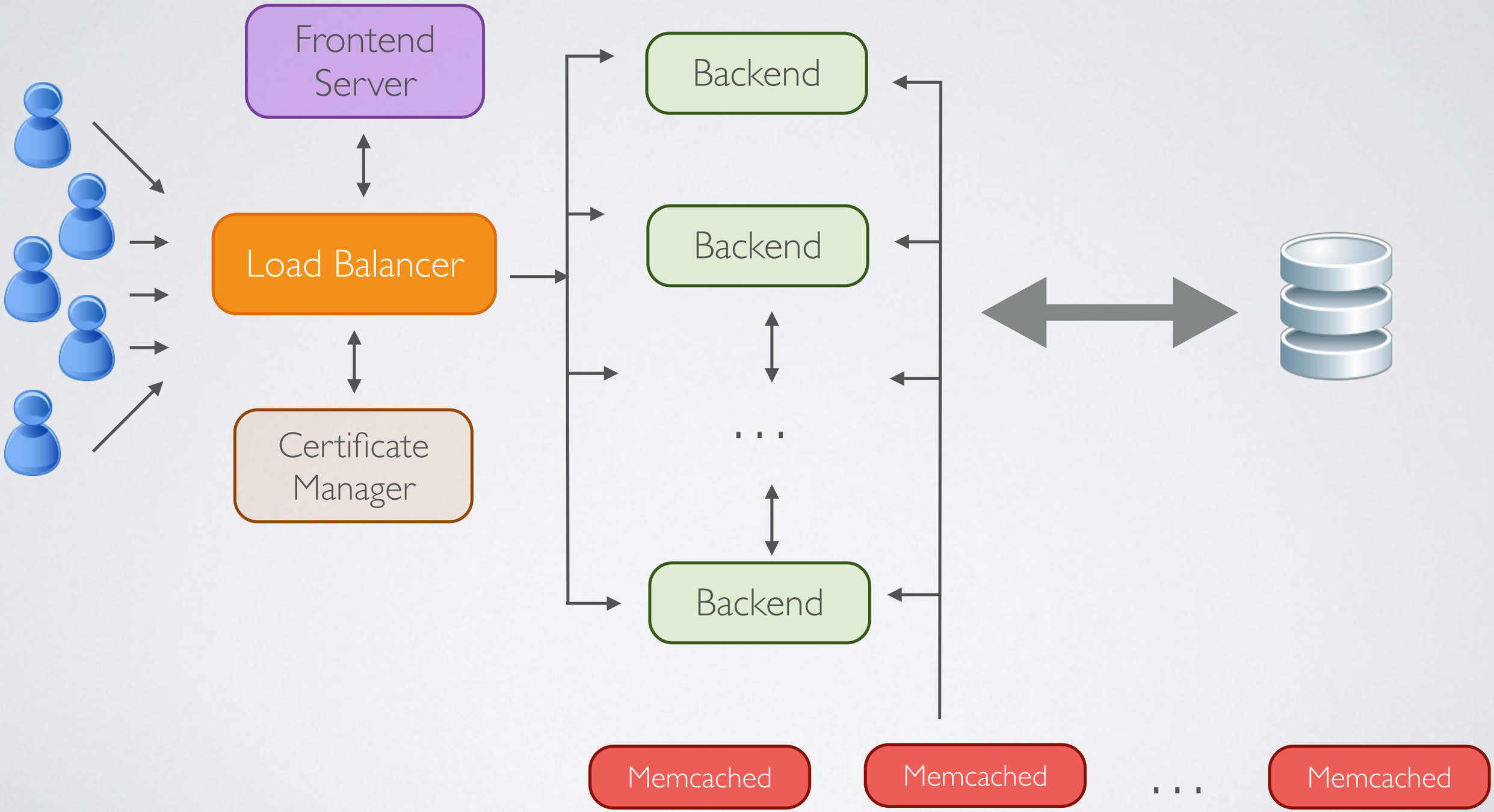
➡ Requires cache warming

Scaling The Backend

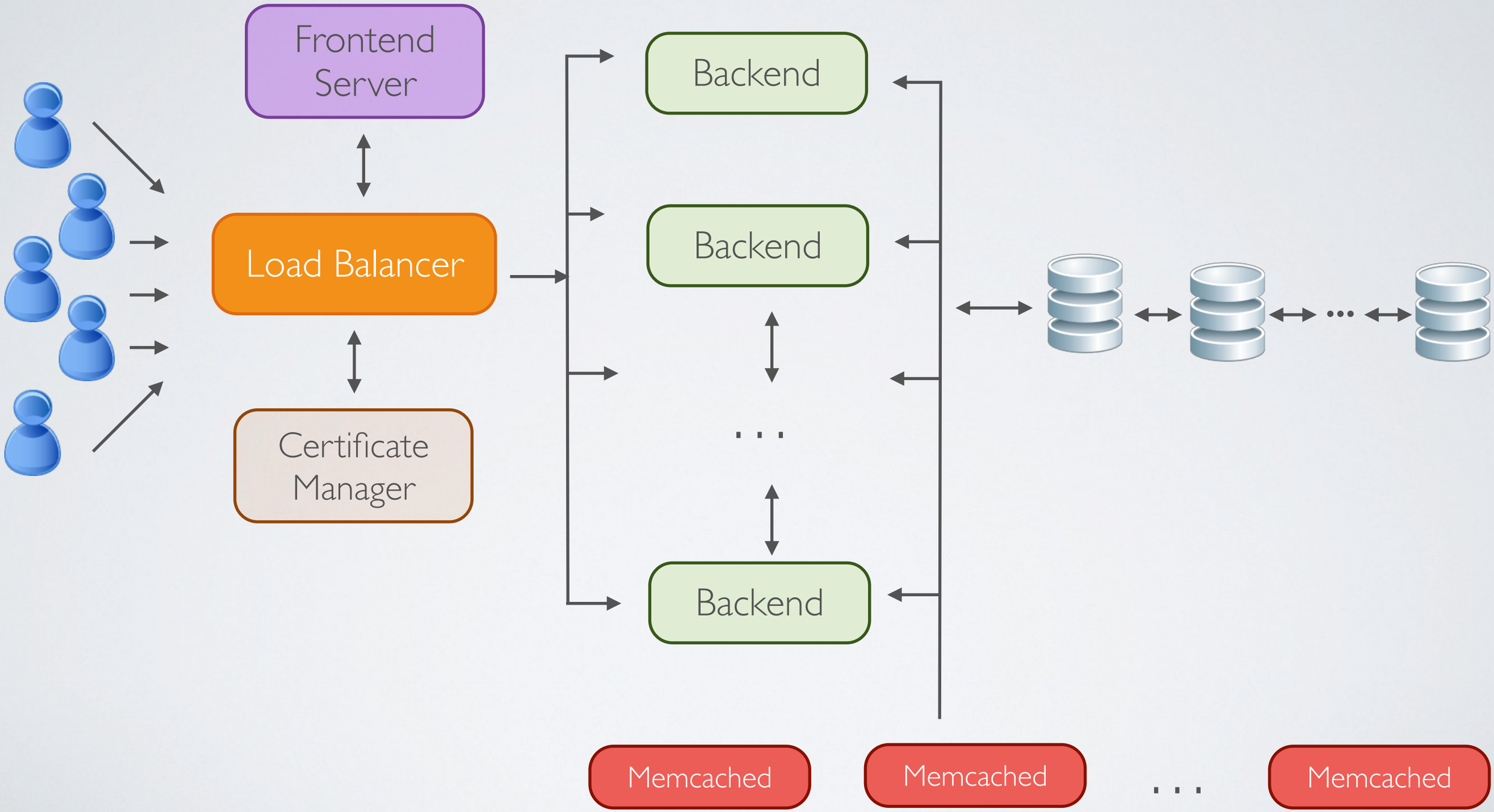
Serving multiple apps with a load balancer



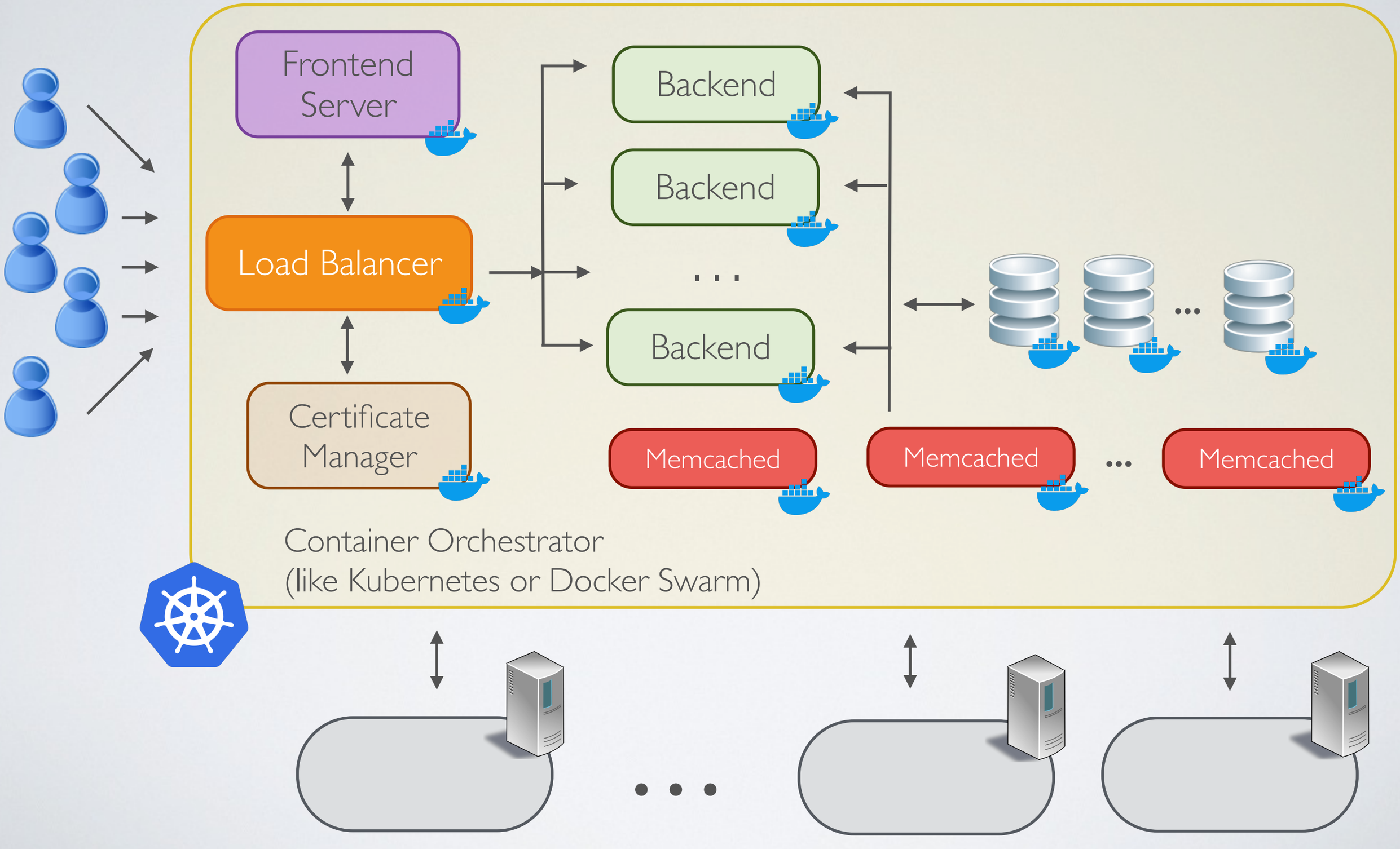
Distributed Shared Cache



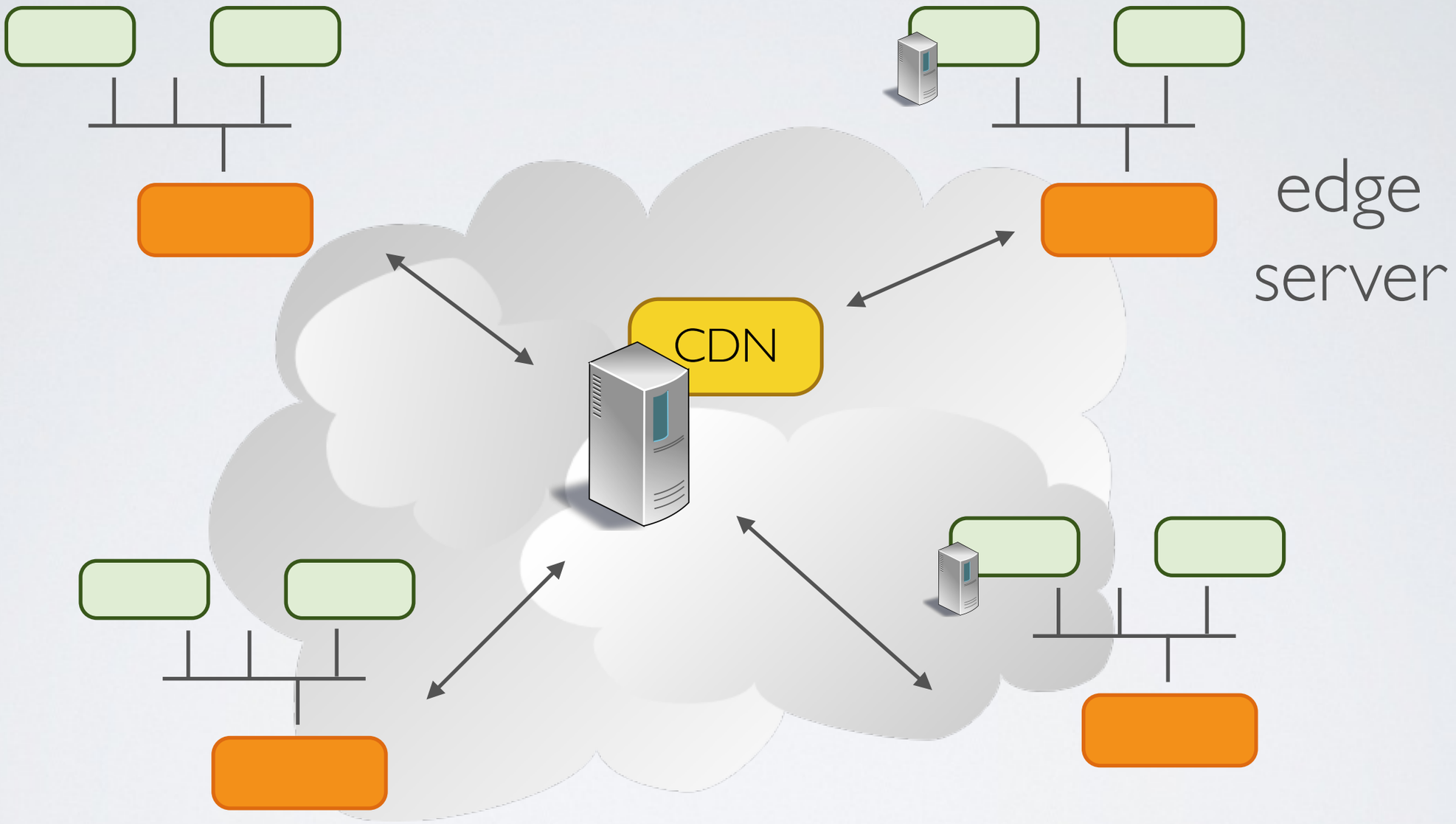
Database Sharding



Automatic Scaling with container Orchestration



CDN : Content Distribution Network



Example : Akamai, Cloudflare