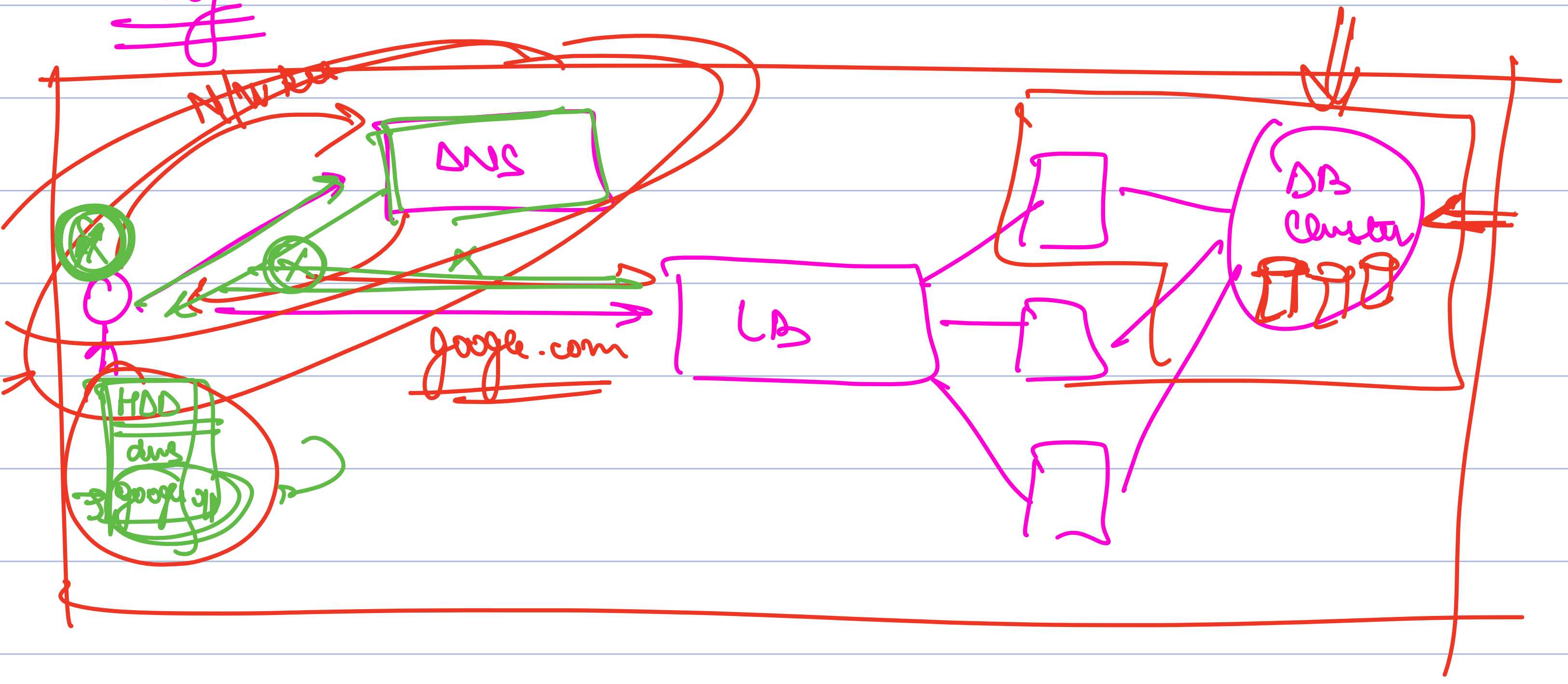
HLD: Caching

1. What is caching
2. Types of caching
3. Consistency
4. Space in Cache
   1. Eviction policies
   2. Case study

Caching



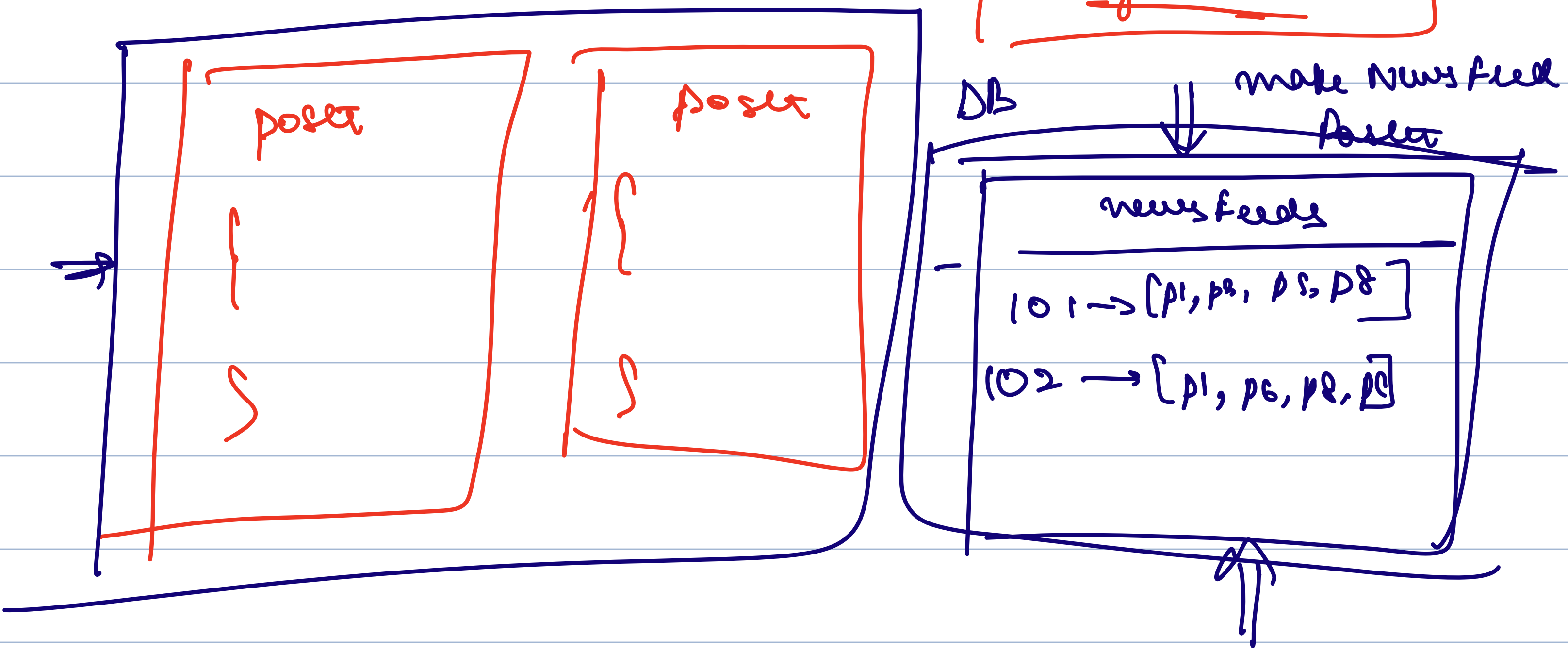
1. Where is DB storing data
   - → corresponding DB machine's Hard disk
   - → HDD is ~~for~~ slower than RAM

② getting i/p and multiple times in a day for same domain.

Cache: storing a copy of the data at some other place (→ in RAM, → in local machine) should be faster or in some other form to overall make data accesses faster

GET /newsfeed ?

Memory ⇒ RAM
Storage ⇒ Disk



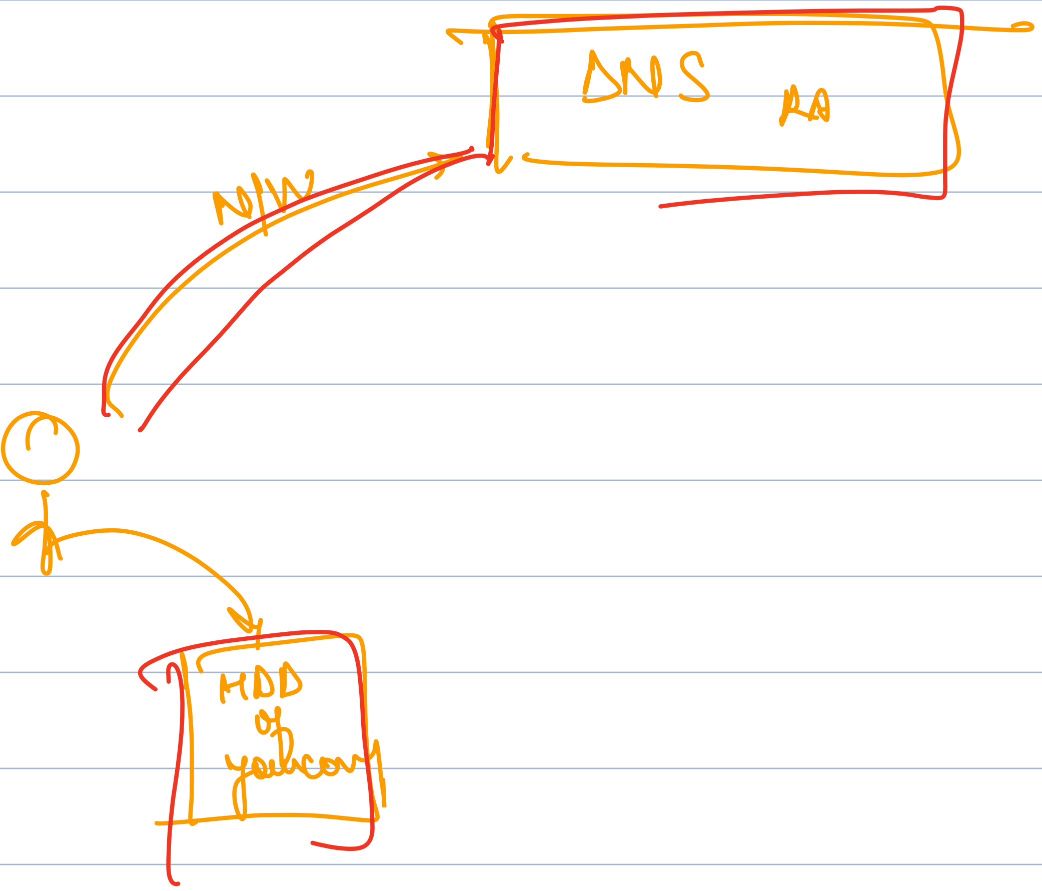
Caching

Why? : To make it faster to access data

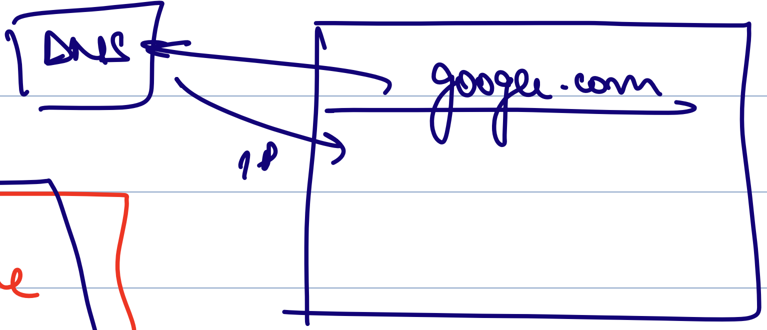
How? :

(i) Storing a copy at a place that is faster to access

(II) Storing a copy in a way that makes opn Fast



Types of Caches



1. In Browser Cache

eg
(i) IP add of domains
(ii) Scaler. com
 ↳ Static assets
(iii) google. com
 ↳ FE code caches the info on browser localstorage.

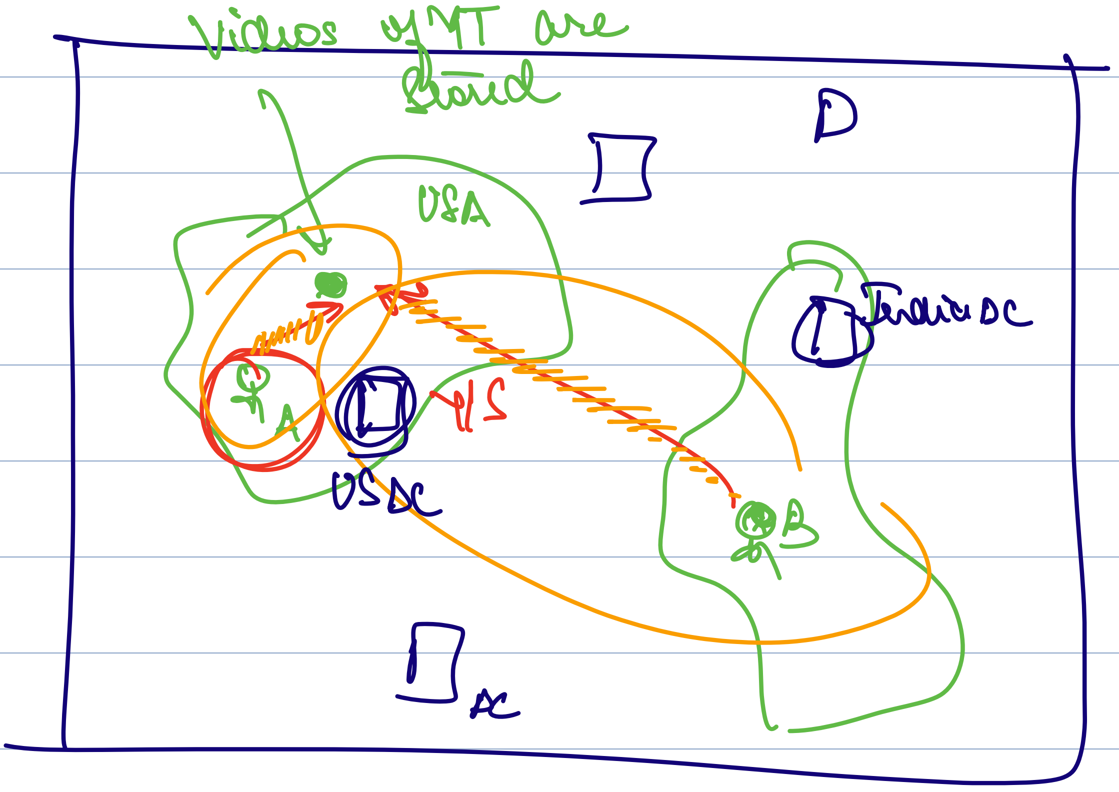
2. CDN

Content Delivery N/W

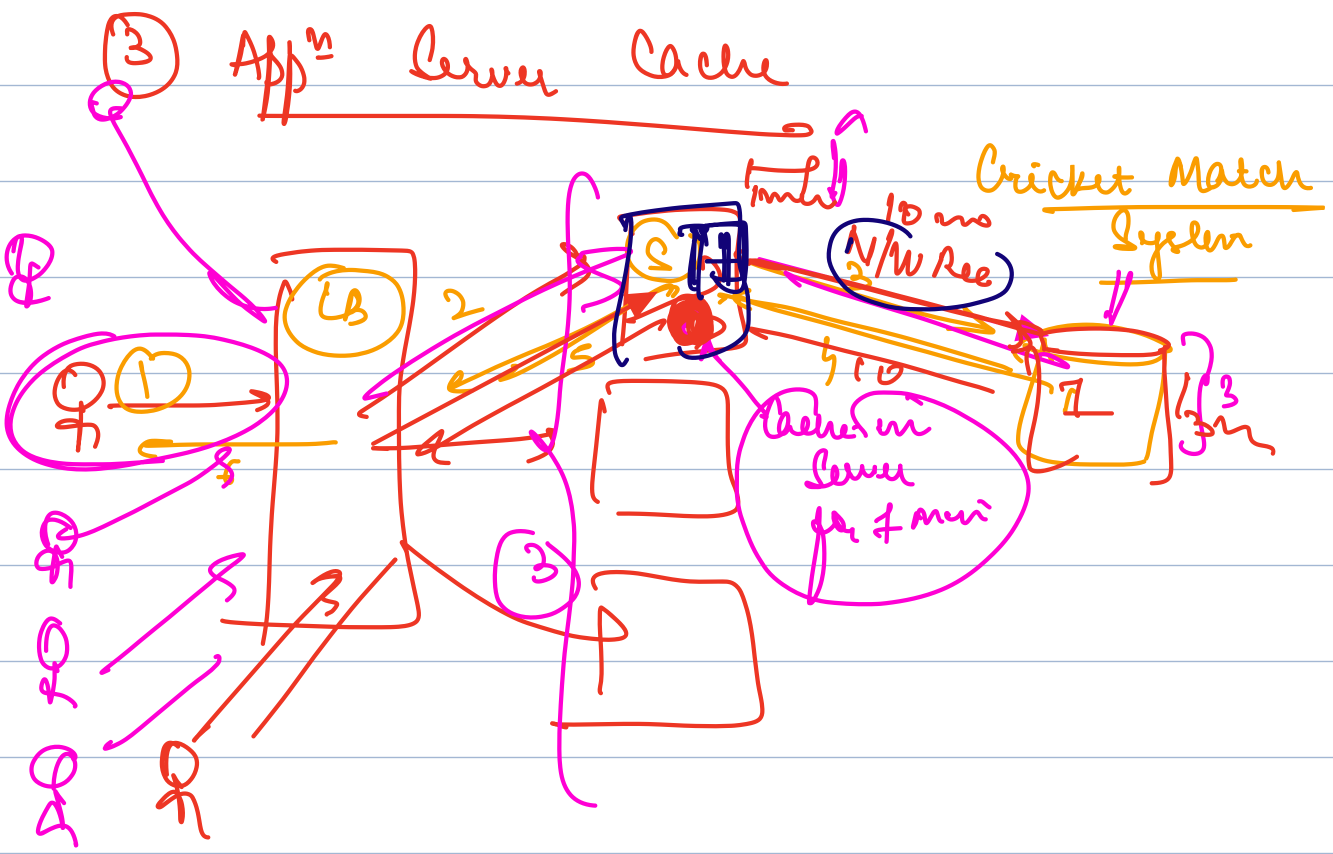
YouTube : play videos

⇒ Cloudflare
⇒ Akamai
⇒ AWS CloudFront

⇒ CDN dist your file across multiple places in the world

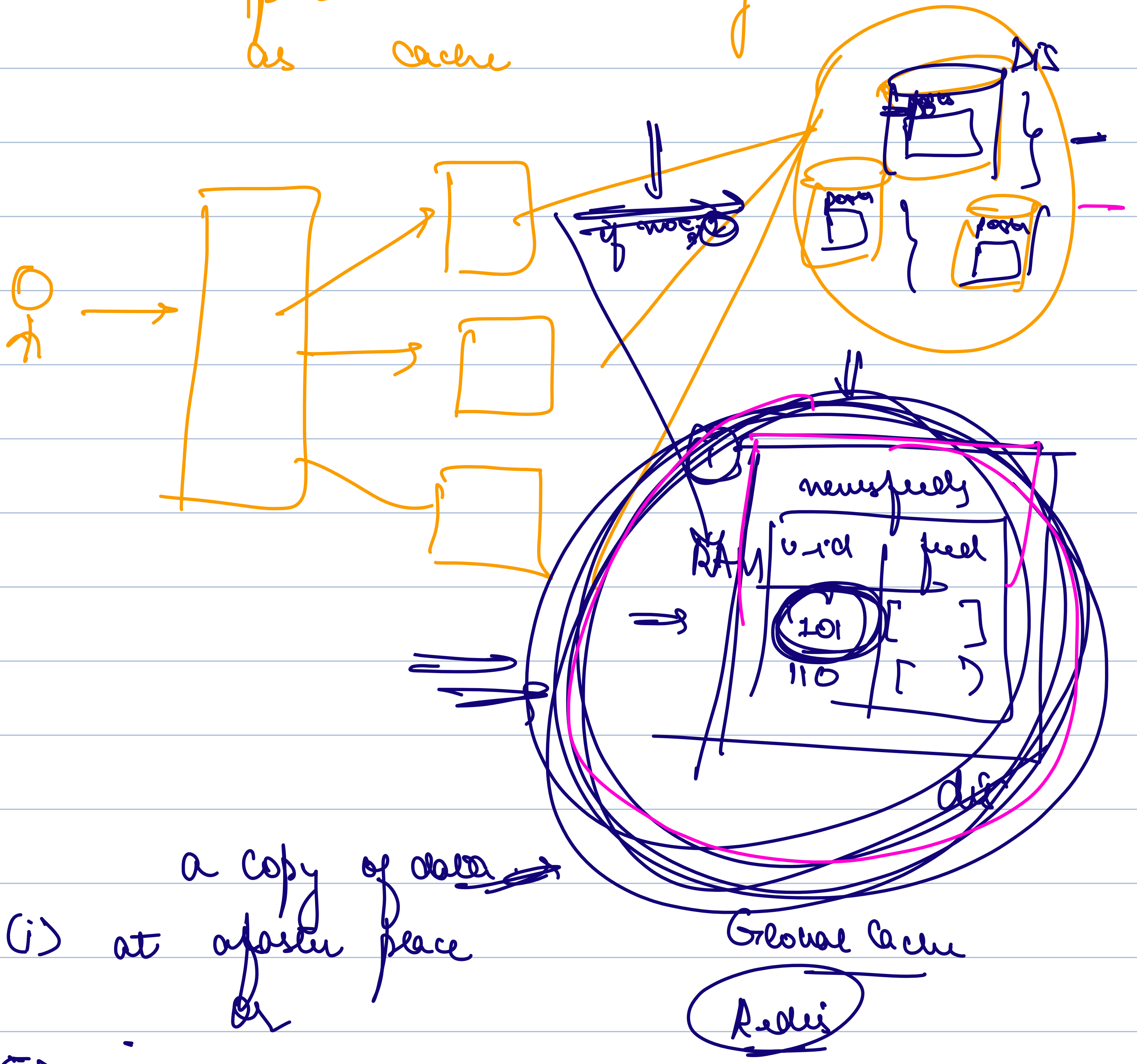


CDN URL ⇒ routed to the nearest server to the user



4 Global Cache

Separate machine only to act as cache



a copy of data →

(i) at a faster place

Or

(ii) in a better way for reads.

Cons

1.) Data can become inconsistent

→ update the data on DB but cache isn't updated.

2.)

$$\begin{tikzcd} HDD \arrow[r] & RAM \end{tikzcd}$$

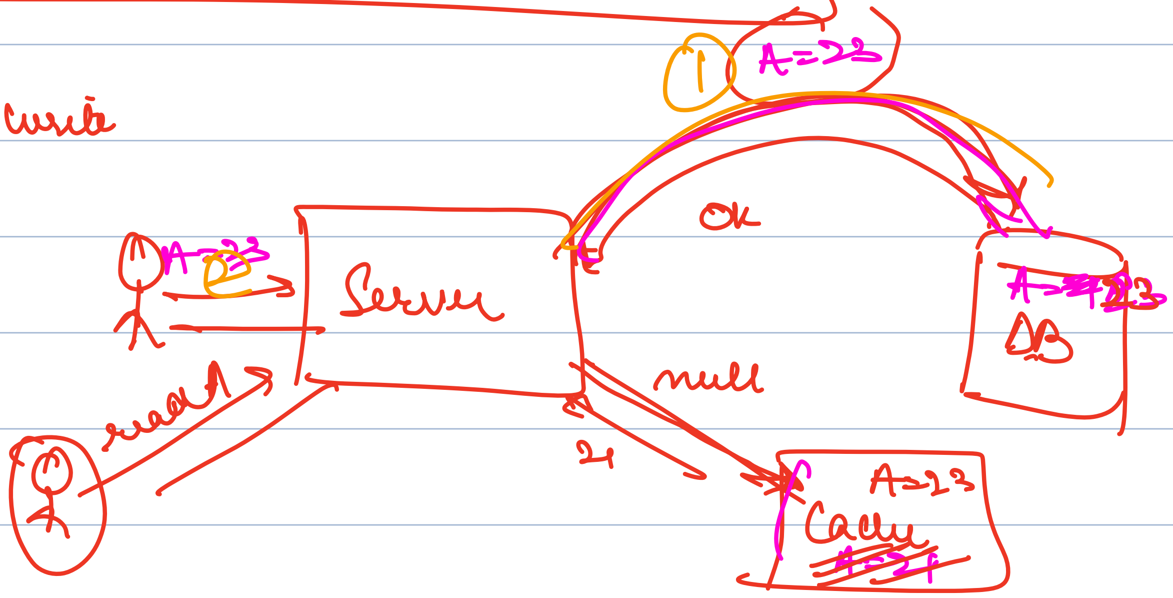
size of cache might be smaller than original source
↳ we may have cache full.

Data becoming inconsistent

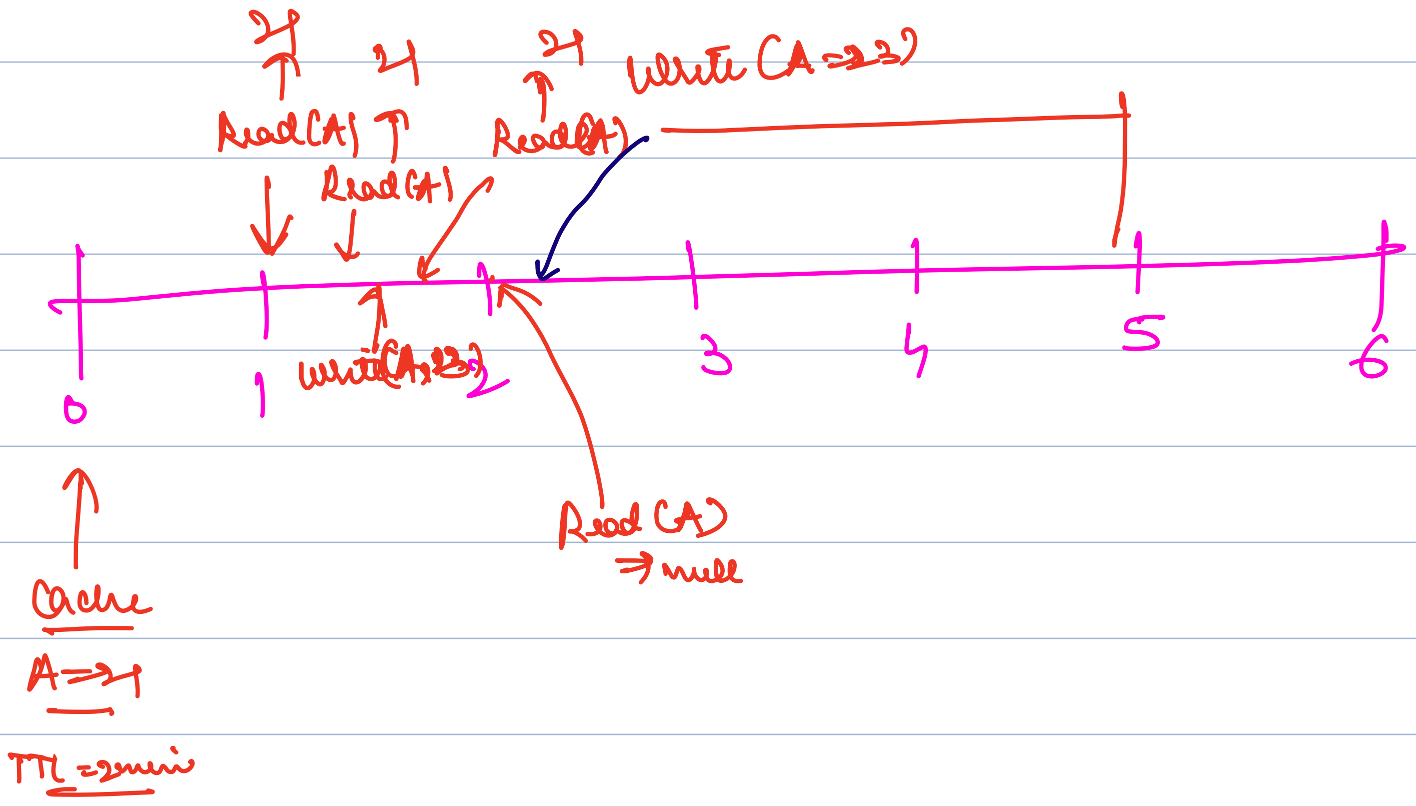
↳ How to handle write requests.

① TTL (Write Around Cache)

write



whenever we store thing to a cache, we also set a TTL(time to live) after which cache will automatically discard that value.



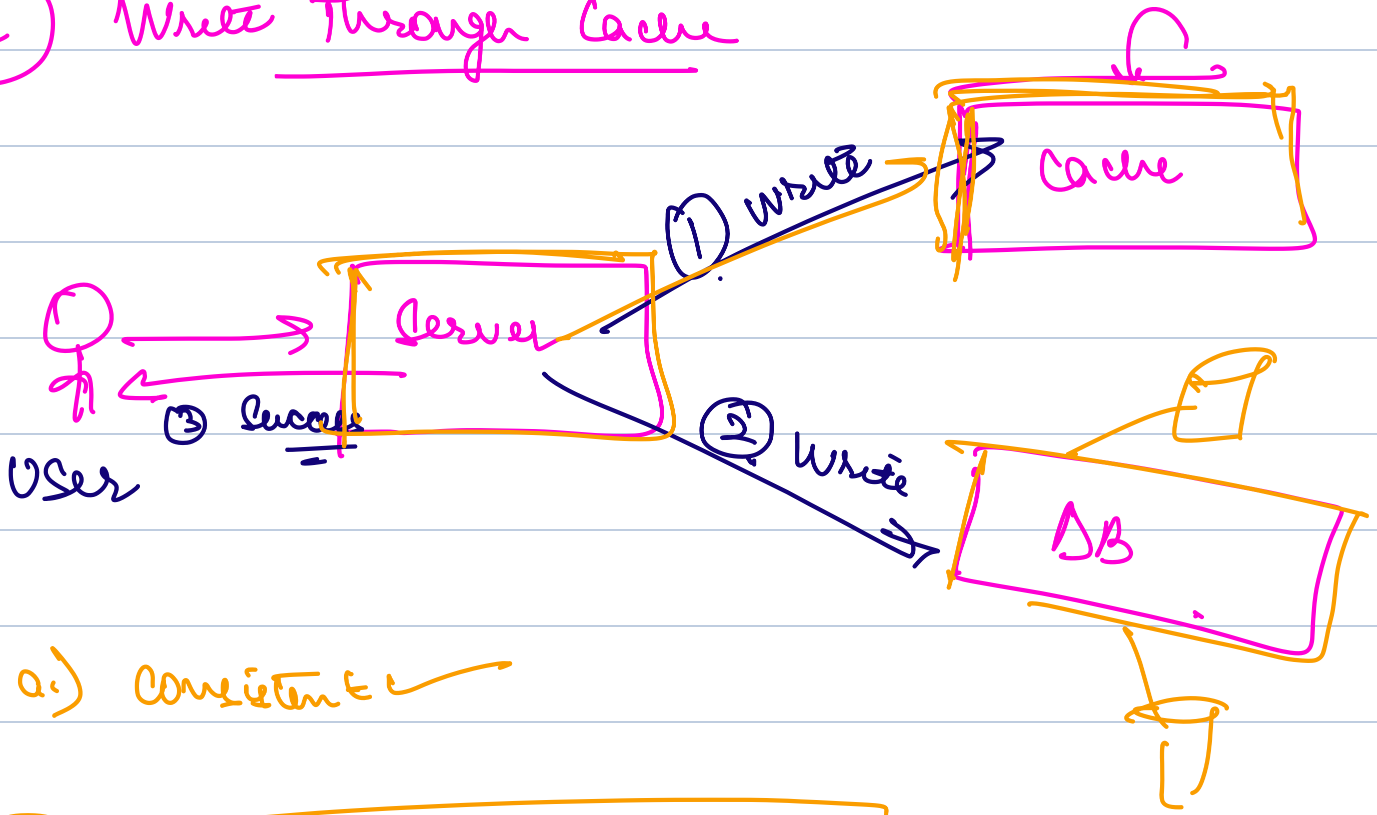
```
read(key) {
    value = cache.get(key)
    if (value == null) {
        value = db.get(key)
        cache.put(key, value, 2min)
    }

    return value;
}
```

```
write(key, value) {
    db. put (key, value)
}
```

Max_dur^n = ttl

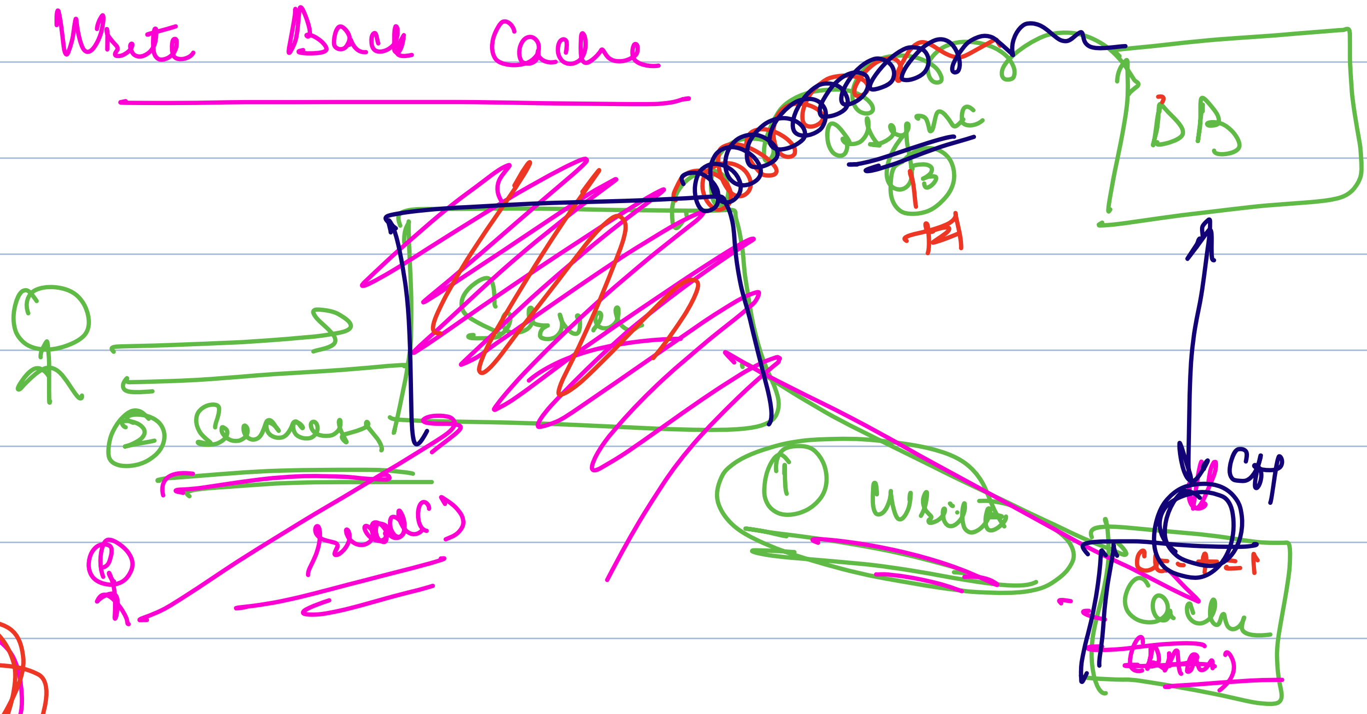
② Write Through Cache



a.) Consistent ✓

Cons: → More latency for write
→ dirty read

③ Write Back Cache



Pro:

1.) latest value read always (consistency)
2.) Fast write (only 1 write and that too cache)

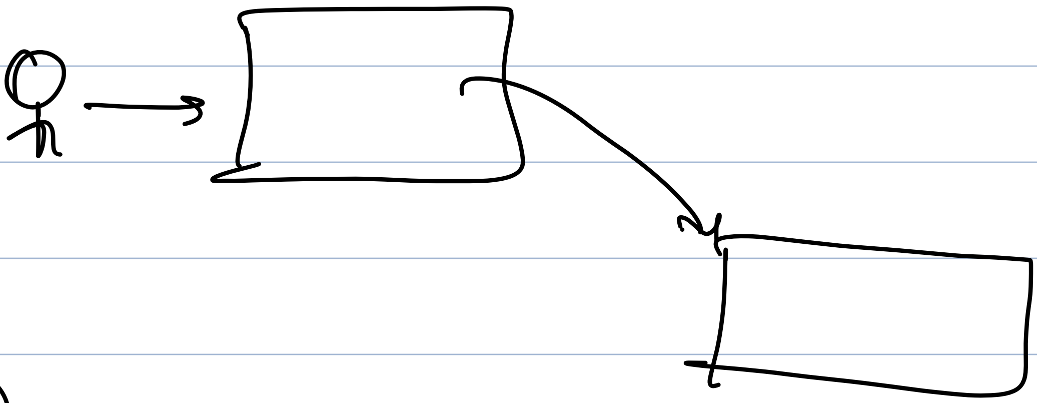
Con

1.) Data Loss.

Jio Cinema ⇒ Viewer Count

eg : analytics / Counter

	WAC	WTC	WBC
Consistent (Latest Value)	✗	✓	✓
How ?	TTL	—	—
Read Speed	same	same	same
Write Speed	middle (only DB)	slowest (Both DB+Cache)	fastest (only to cache)



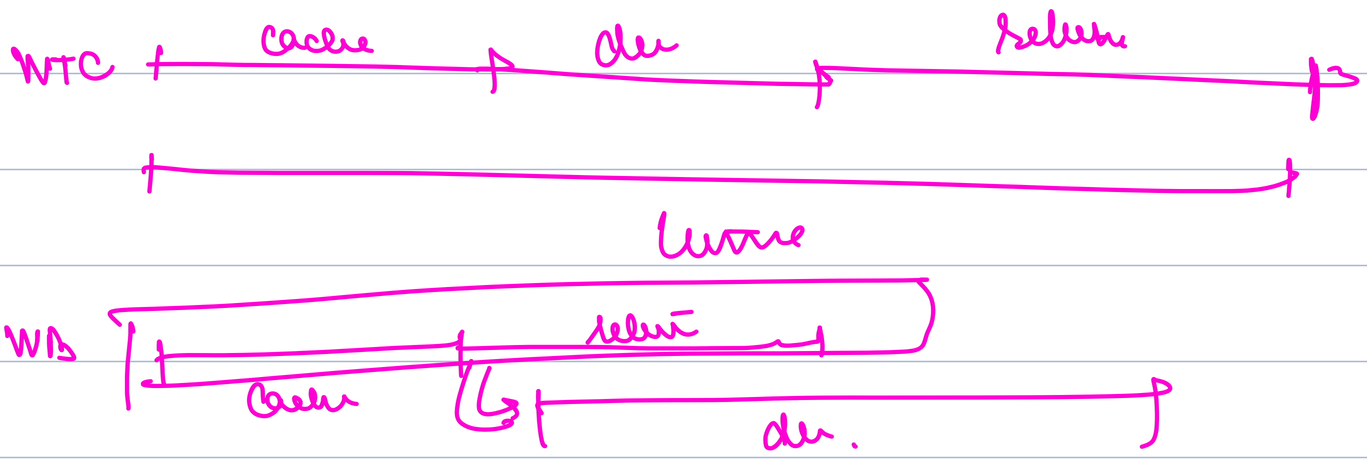
①

```
handleWrite(key, value) {
    ① cache.put(key, value)
    ② ~~return~~
    async    db.put(key, value)
}
```

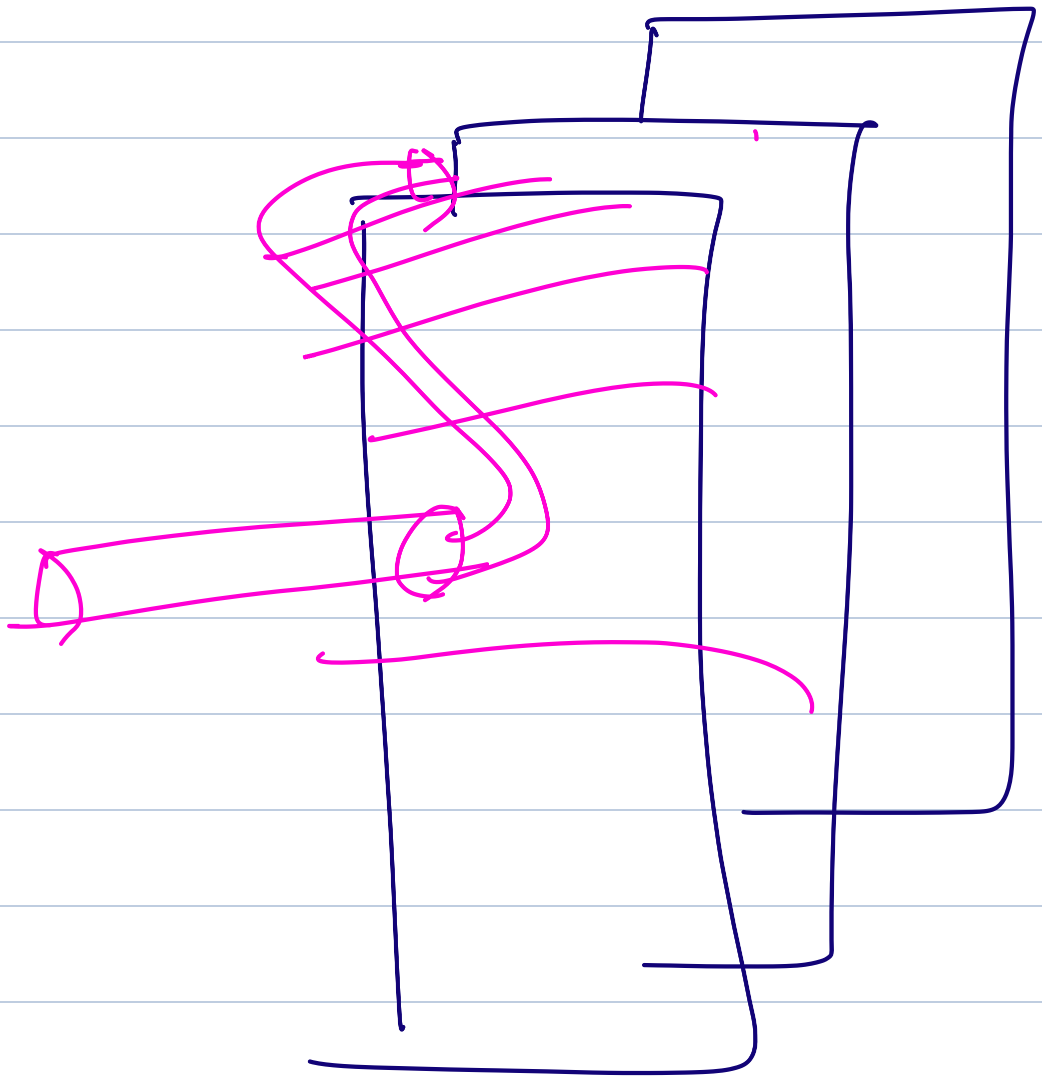
Reads are same regardless of your write policies

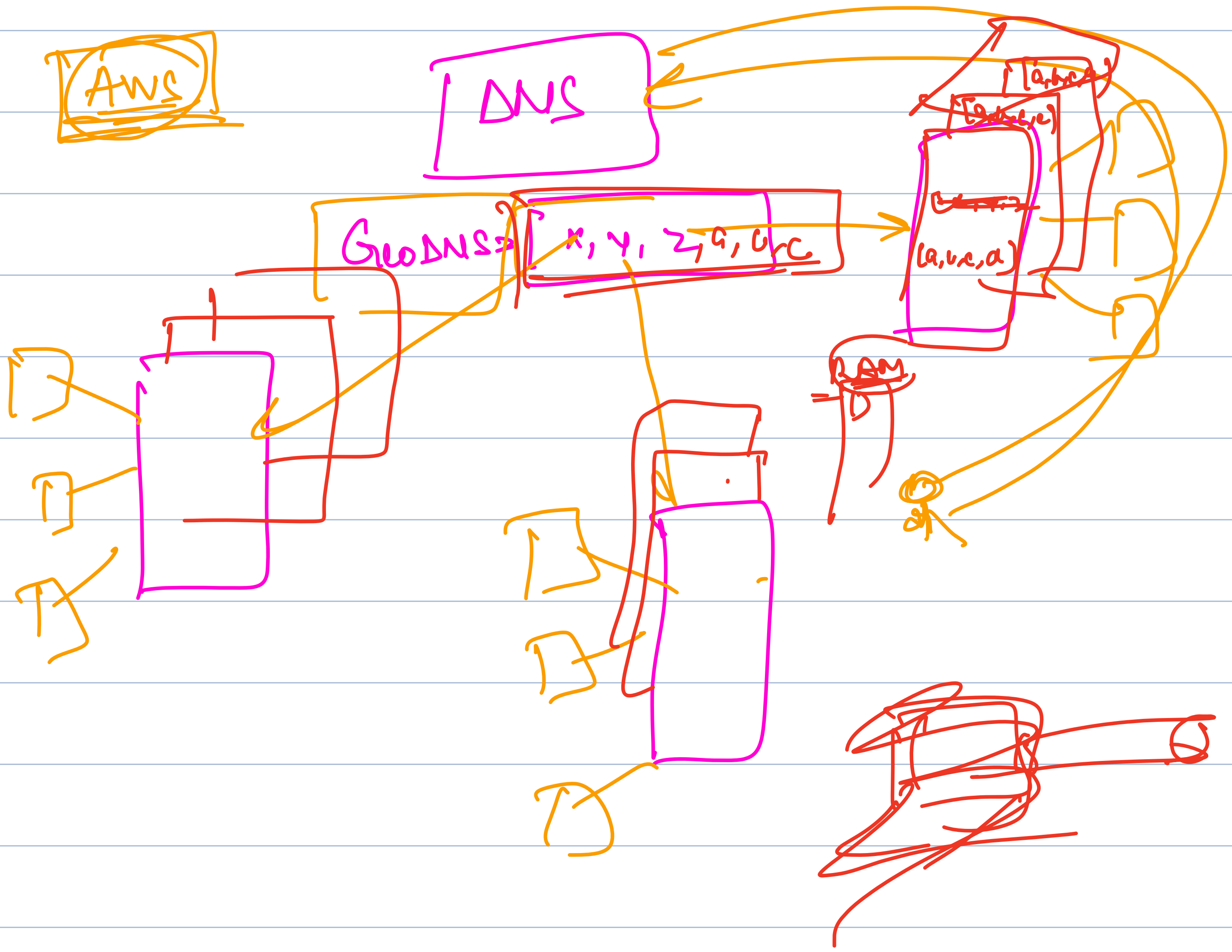
```
read (key) {
    value = cache.get (key)
    if (value == null)
        v = db.get (key)
        cache.put (key, value)

    return v;
}
```



MS ⇒ cant take locks





1 GB