Design Twitter

Saturday, March 23, 2024 2:32 PM

FUNCTIONAL REQUIREMENTS
1. Generate newsfeed
2. Tweet (Text)
3. Followers / Followee
4. Tweets may contain photos / videos

Extended Requirements
1. Share a tweet
2. Retweet
3. search for a tweet
4. Comments / likes
5. Must users be private?

NON FUNCTIONAL REQUIREMENTS
1. Available
2. Low latency (200 ms)
3. Eventual consistency

CAPACITY ESTIMATION
1. Daily active users: 200M
2. Post: 100 M tweets / day
3. Check my profile 2 times per day.
4. 5 profiles / day 3 check per day (10 times)] 10 entries.

Traffic
Writes: 100 M tweets / day
Reads: 200M × (5+2) × 20
140 × 200 M = 28000 M = 28 billion

Storage:
each character takes 2 bytes
140 characters (1 tweet size) × 2 bytes = 280 bytes
100M × (280 + 20) = 300 × 100M = 30 GB
tweet size, metadata size

20 GB × 365 (days) × 10 (years) = 1095000 GB
Every 5th tweet has a photo — (200 kb)
Every 10th tweet has video — (2 MB)

photos: $\frac{100M}{5}$ daily tweets × 200 kb = 4000 × 10^6 × 10^3 = 4 TB per day.
memory of each photo

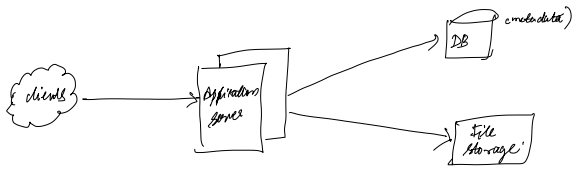
Videos: $\frac{100M}{10}$ × 2 MB = 20 × 10^6 × 10^6 = 20 × 10^{12} = 20 TB

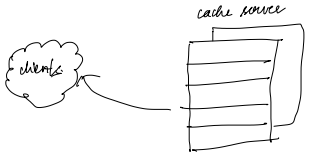
Total: 20TB + 4TB = 24 TB / day.

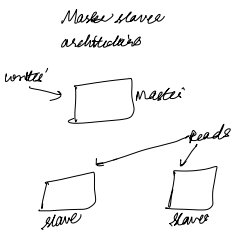
24 TB × 365 (days) × 10 (years)

Bandwidth:
$\frac{30GB}{24 \times 60 \times 60} + \frac{4TB}{86400} + \frac{20TB}{86400}$ = 25 GB/sec

Caching:

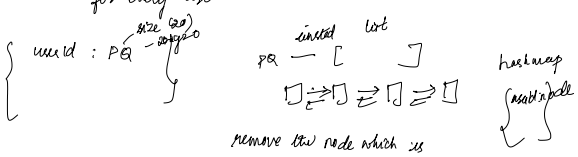




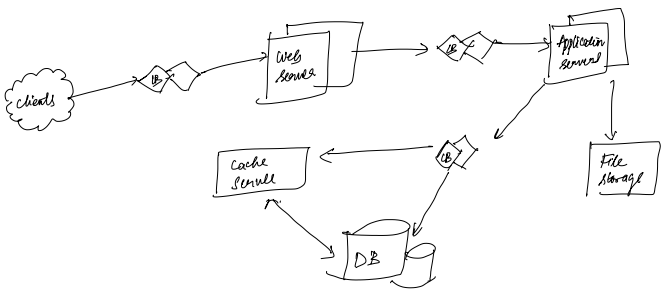


one of the slave replaces master of master goes down.

To get the top records - we need a priority queue for every user.



remove the node which is

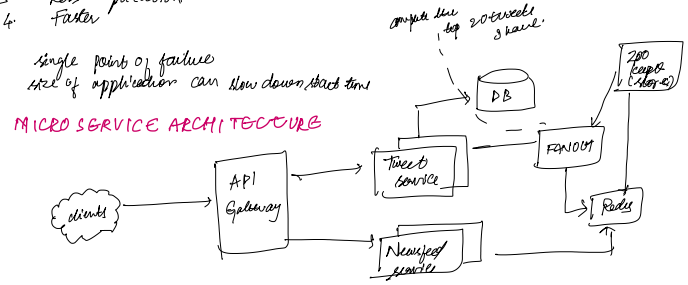


Monolithic architecture

+ves
1. Good for small teams
2. Simple for scaling horizontally
3. Less duplication
4. Faster

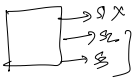
-ves
single point of failure
size of application can slow down start time

Microservices Architecture



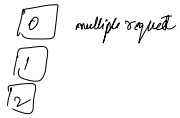
\+ Easy to add another service
single point of failures gone

-ves complex communication takes place b/w services
- REST
- RPC

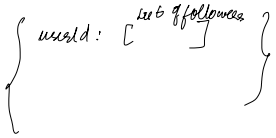


if one service goes down we can

Cache Eviction Tech: LRU
LB algorithm:
if userId is used for partitioning X
tweetId - multiple reqs
tweetId and creation time



In DB I store



components - How to generate a tweetId
timestamp + autoInc
epoch
32 bit
16 bit
48 bit

multiple tweets can come at the same time
to have it unique use autoInc for current epoch.