Facebook Capacity Estimation

1) Traffic Estimation (per sec traffic)

Total users = 1 B

Daily Active Users (DAU) = 25% of total users

$= \frac{25}{100} \times 10^9$

$= 250 \times 10^6$

$= 250$ Million users

Read & Write operations is traffic

Let's assume every user does (5 reads + 2 writes) per day = 7 queries

≈ 10 queries

So, total traffic per day = 250 Million × 10

$= 2500 \times 10^6$ queries

= 2500 Million queries per day

1 day → 2500 Million queries

1 sec. → $\frac{2500 \text{ Million}}{24 \times 60 \times 60} = \frac{2500 \times 10^6}{100000 \text{ sec}}$ (approx)

= 25000 querie/sec

= 25 K queries/sec

Storage Estimation

Assumption :-

1) Every active user does 2 posts per day.
2) 10% of users upload 1 image also.

1 post has 250 characters
1 image = 300 KB

1 post = 250 × 2 Bytes
= 500 Bytes

So, 2 post = 2 × 500 Bytes
= 1 KB

So, 250 Million users × 1 KB = 250 GB
So, 250 GB needed perday for posts.

10 % of 250 Million = $\frac{10}{100} \times 250 \times 10^6$
= 25 Million users

25 Million users × 300 KB = 7500 GB
= 7.5 TB
≈ 10 TB

So, 10 TB data is required per day for images.

So, total = (250 GB + 10 TB) per day
≈ 11 TB

RAM Estimation

Last 5 posts will be cached for each user.

Last 5 $\rightarrow$ 4 posts

Last 5 $\rightarrow$ 1 mage

4 posts = 4×250 characters

$= 4 \times 250 \times 2$ Bytes

$= 2000$ Bytes

$= 2$ KB

1 image = 300 KB

Total = 302 KB $\approx$ 300 KB

250 Million $\times$ 300 KB = 75 000 GB

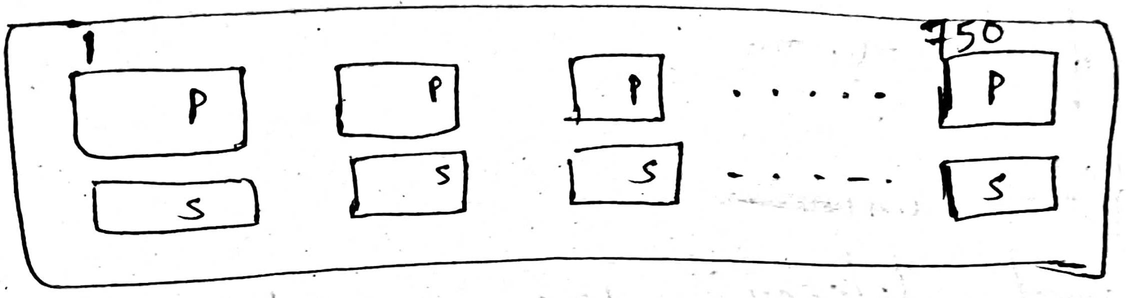
= 75 TB per day

Each machine (node) can have 128 GB RAM

$\rightarrow$ usable 100 GB (28 GB for OS and Redis Overhead)

$$\frac{75 \times 10\cancel{0}\cancel{0} \text{ GB}}{1\cancel{0}\cancel{0} \text{ GB}} = 750 \text{ machines}$$

750 master nodes



So, 750 masters + 750 replicas

= 1500 ⇒ Total redis servers,

⇓

Each having 128 GB storage (RAM)

We can implement cache eviction strategy like LRU

No. of Servers

~~Let's assume~~

I want latency = 100 ms = 0.1 sec

↳ Means time taken to process one request is 100 ms

0.1 sec ⟶ 1 request

1 sec ⟶ $\frac{1}{0.1}$ requests

= 10 requests

10 req ⟶ 1 sec

Assume 1 server can have 200 threads

Lets say each thread serves 10 req in 1 sec

So, 10 × 200 = 2000 Requests per sec

1 server ⟶ 2000 RPS

⇒ 25000 ⟶ $\frac{1}{2\cancel{000}} \times 25\cancel{000}$ servers

= 13 App servers

So, 13 app servers are required to serve 25K requests persecond

It is alway best to add 30 - 50% buffer for 1) Traffic Spike

2) Future growth

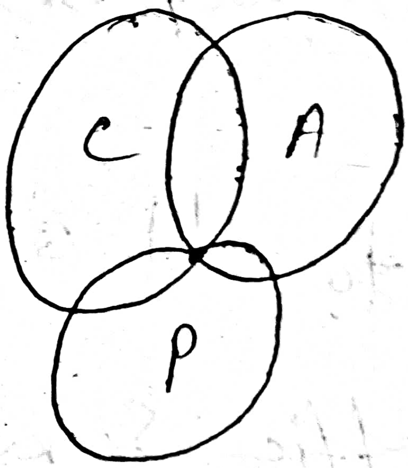
So, $\frac{30}{100} \times 13 + 13 = \frac{13}{3} + 13$

$= 4 + 13$

≈ 17 app servers

$\boxed{\approx 20 \text{ App Servers}}$

Trade off



CA
CP
AP ✓

CAP