Agenda.

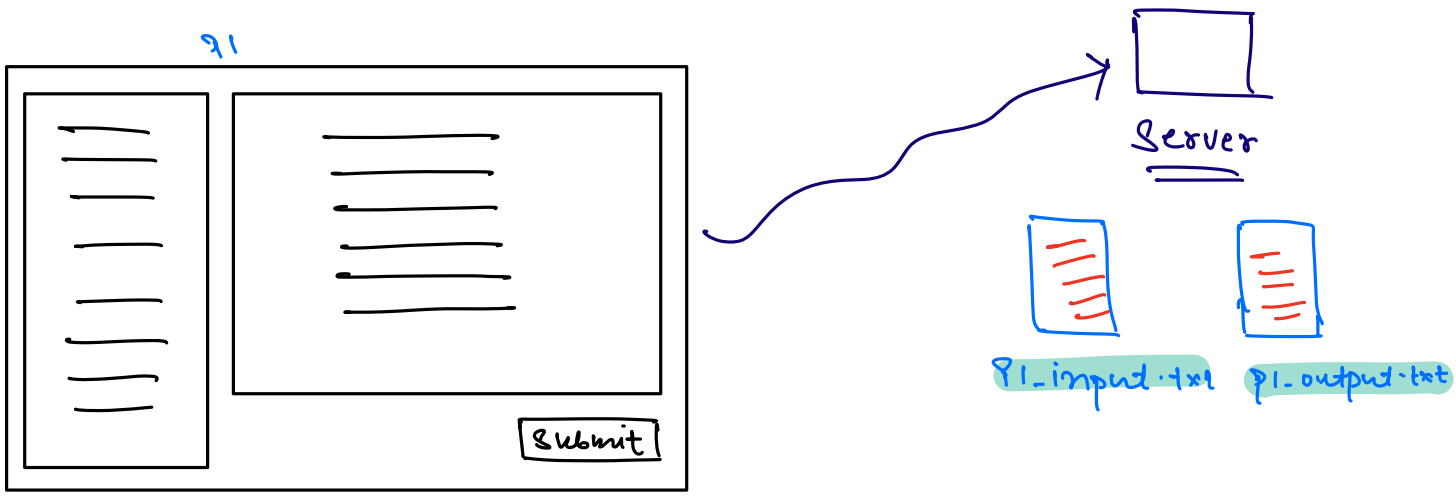
⇒ Case studies on Caching

1) Scaler Code Judge.

2) Scaler Contest Leaderboard.

Scaler Code Judge.

↳ DSA Code judge.



Size of input file

⇒ 100 Test Cases * 10^6 integers * 8B

= 800×10^6 Bytes

= 800 MB.

~ 1GB.

1 problem ~ 1-2 GB.

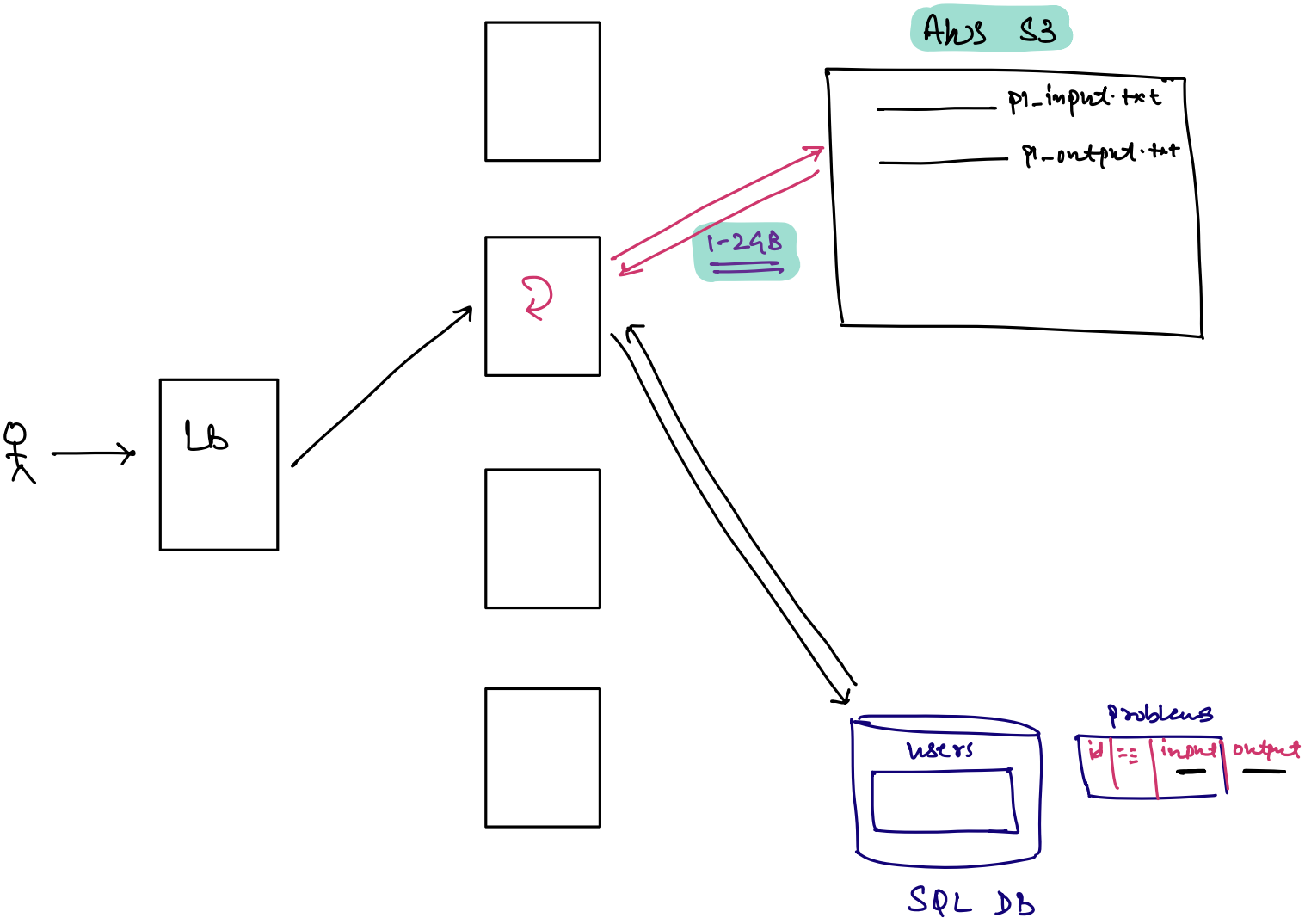
total 2000 problem.

↳ 2000 x 1.5 GB.

⇒ 3 TB.

File Storage

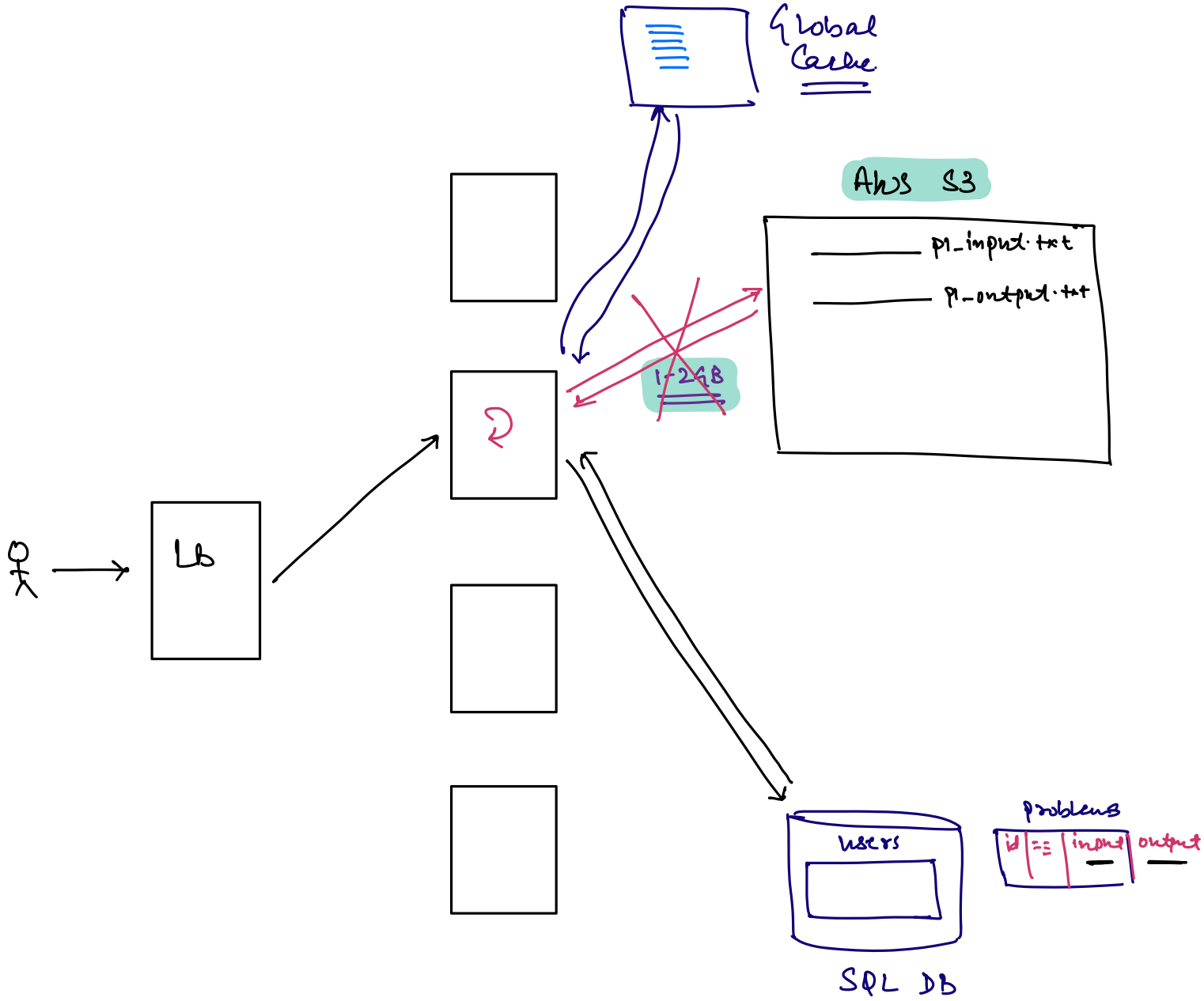
AWS S3 | Azure Blob | ...



⇒ Huge n/w bandwidth.

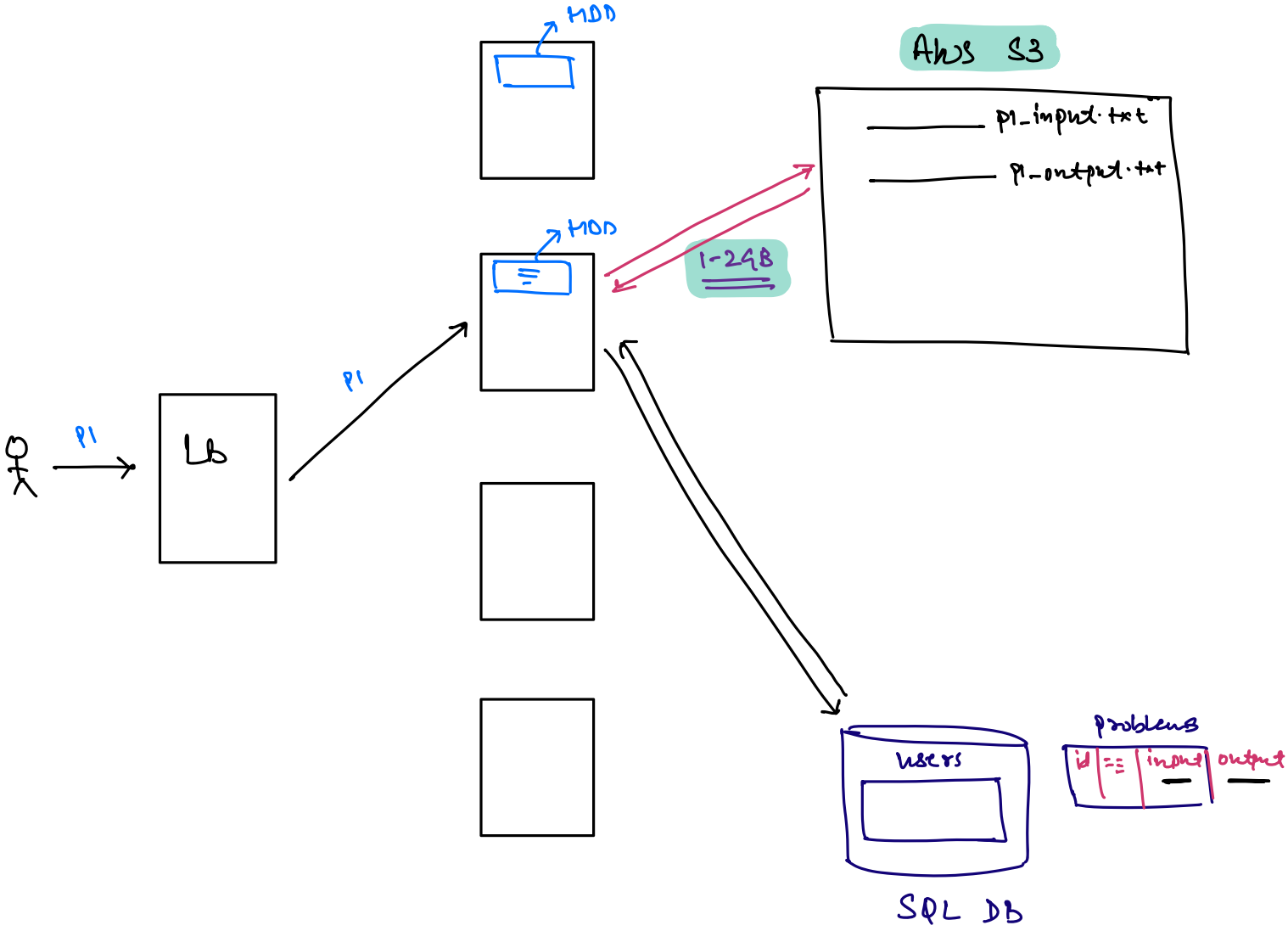
⇒ 1-2 GB data transfer per submission.

Global Cache



⇒ Not solving the problem of n/w bandwidth.

Local Cache.



Cache Invalidation

→ Can TC files change ?

↳ Extremely rare.

→ Eventual (or) Immediate Consistency. ✓

TTL ✗

↳ 1min

1 sec

1 hr

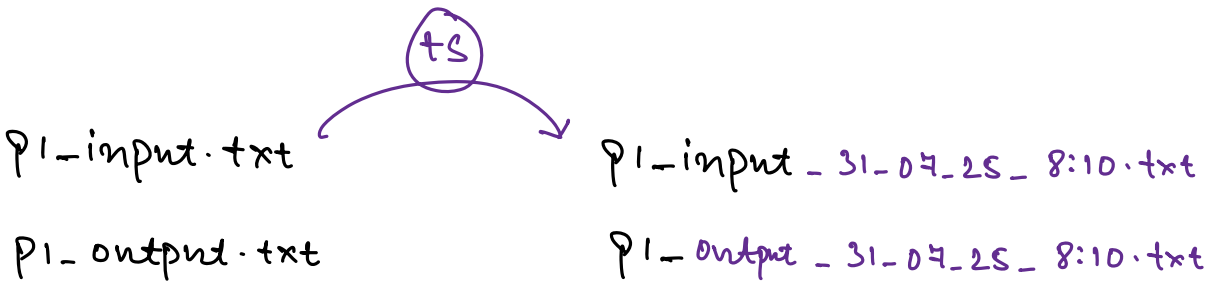
10hr

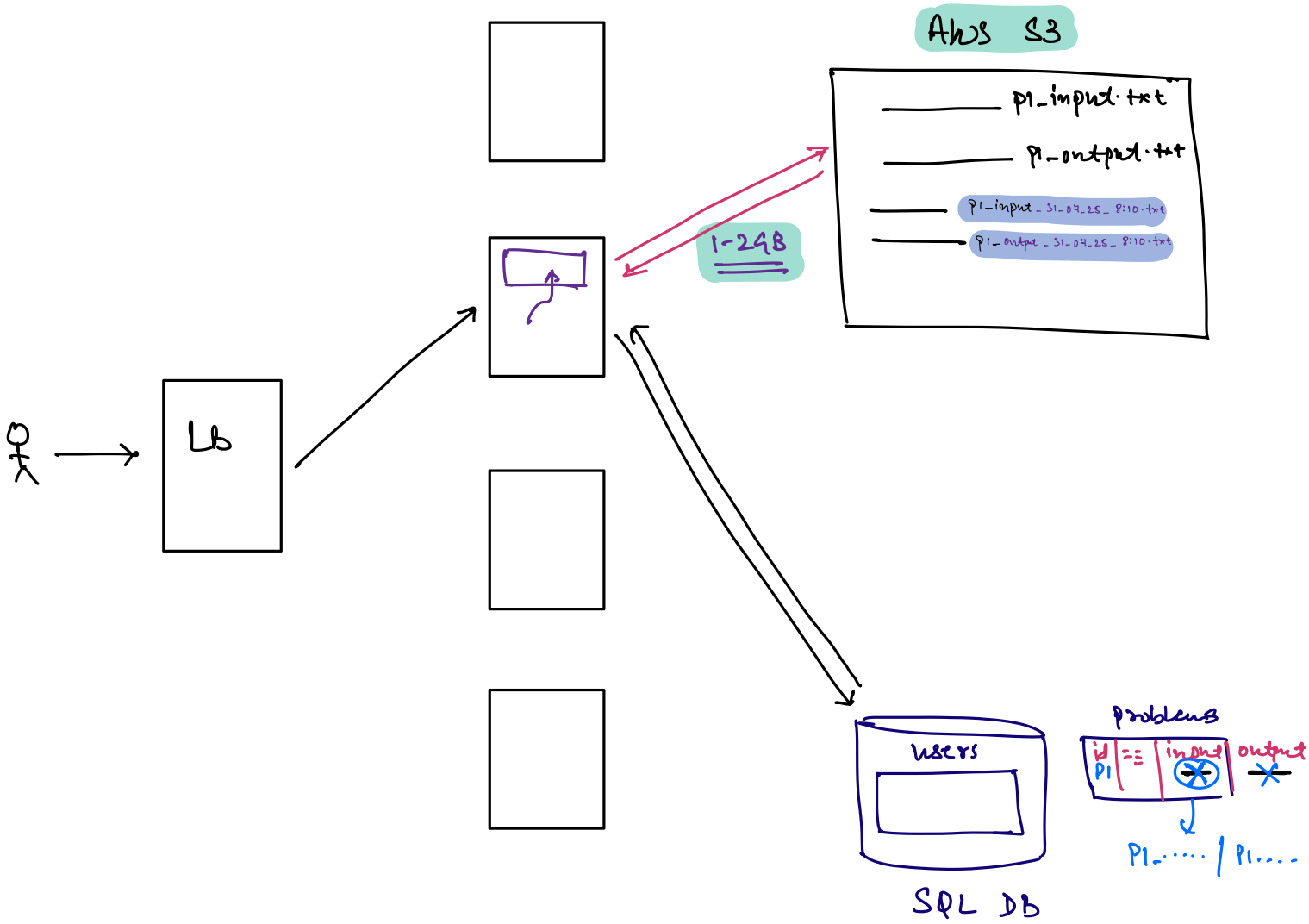
24hrs ✗

Write Around ✗

Write Back ✗

Write Through. ✗



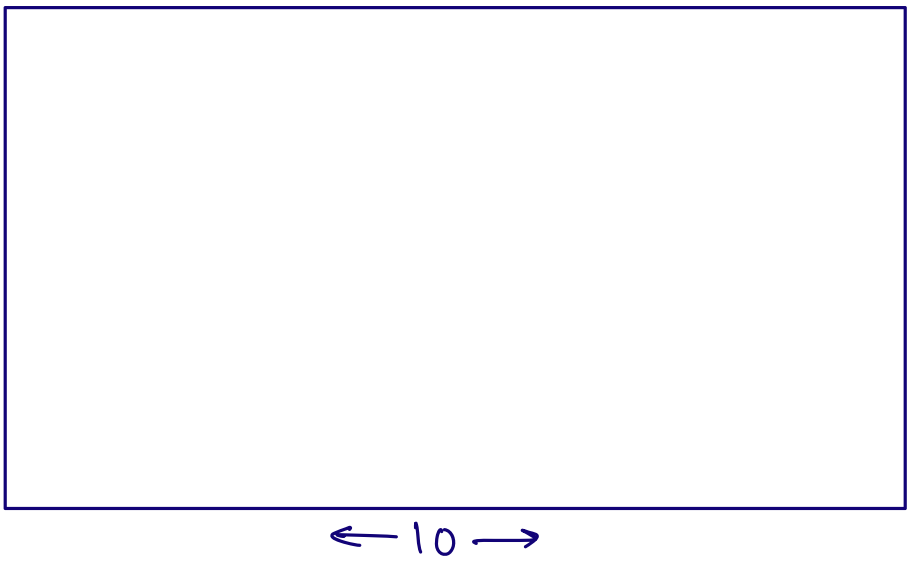


⇒ Immediate Consistency. ✓

Scaler Contest Leaderboard.

↓

100K participants.



⇒ Paginated response.

⇒ 3hrs : Contest Duration

100K users.

5 problems.

Avg Submissions

1 submission / user / problem × 5 problems

⇒ 5 submissions / user

Total Submissions = 5 × 100K Submissions

= 500K.

3 hrs → 500 K

3 × 60 × 60 sec → 500 K

1 sec ⇒ $\frac{500\,K}{3 \times 3600} = \frac{50\cancel{0} \times 1\cancel{000}}{1\cancel{0000}}$

= 50 Submissions / sec.

Data for leaderboard.

user_id ⇒ {contest_id, problem_id, marks, # of subm.

500 B.

100 K × 500 B

⇒ 500×10^5 Bytes.

⇒ 50×10^6 B.

⇒ 50 MB.

⇒ Avg # of requests for leaderboard

$$100K * \left(\frac{20}{3 \text{ hrs}}\right)$$

$$\frac{20 \times 100 \times 1000}{3 \times 60 \times 60} = \frac{20 \times 100 \times 1000}{10000}$$

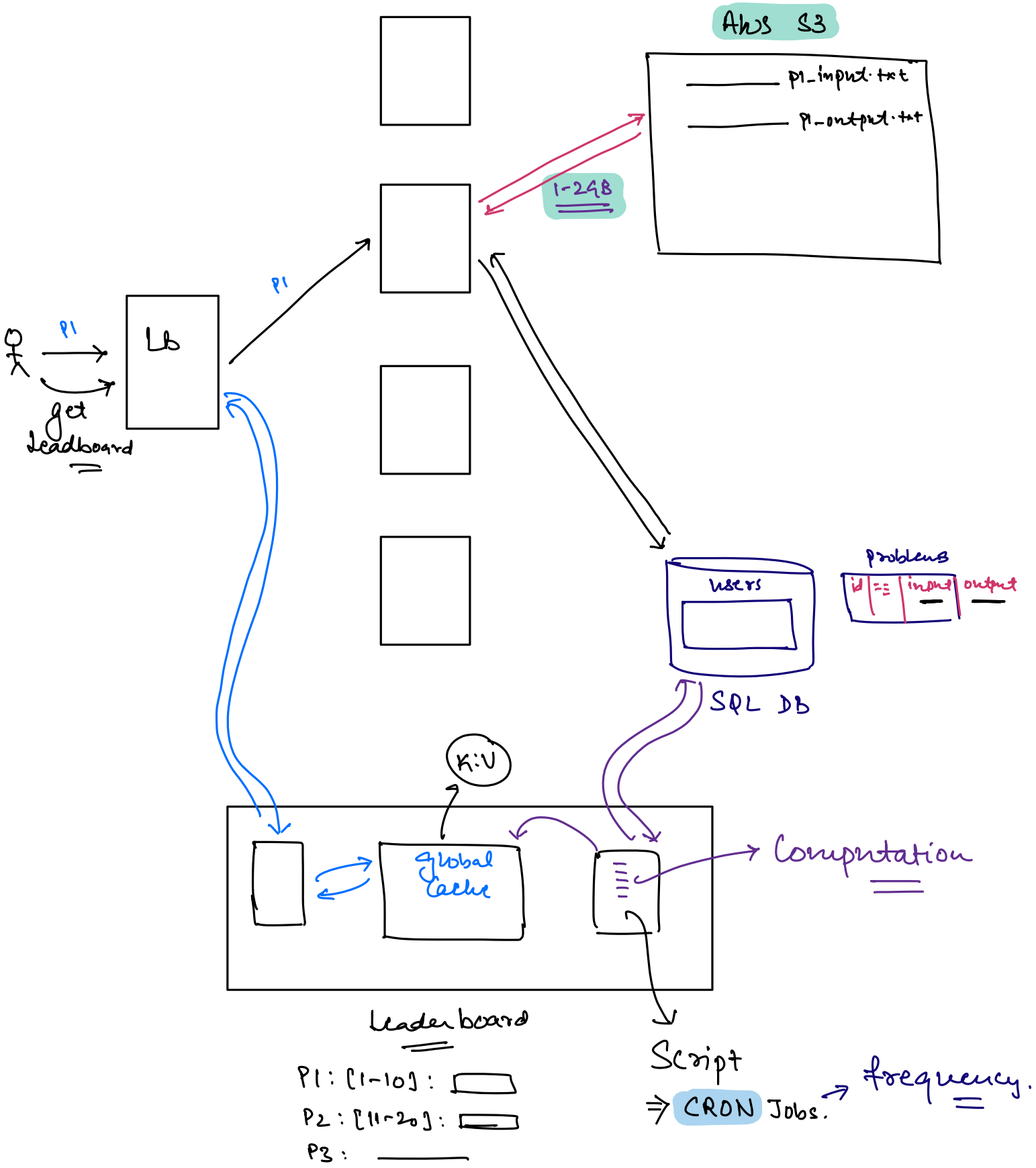
$$= 200 \text{ views/sec.}$$

⇒ Cache.

↳ Local (OR) Global. ✓ ✗

↓

Single (OR) Distributed.


Aws S3
p1_input.txt
p1_output.txt
1-2GB
P1
P1
LB
get
leadboard
users
problems
id
input
output
SQL DB
K:V
Global Cache
Computation
Leaderboard
P1 : [1-10] :
P2 : [11-20] :
P3 :
Script
CRON Jobs.
frequency.

Cache Invalidation.

Eventual (vs) Immediate Consistency.

TTL

↳ 10 mins.

Cache Eviction.

↳ No need.

———— * ————