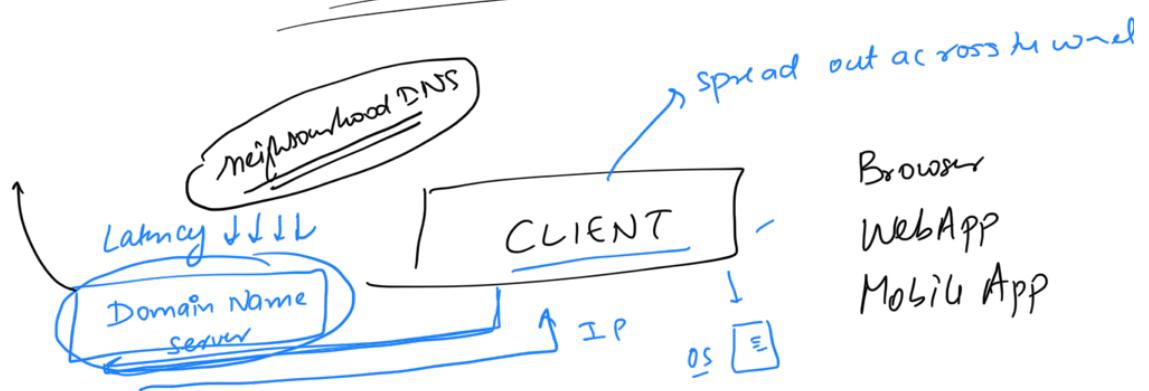
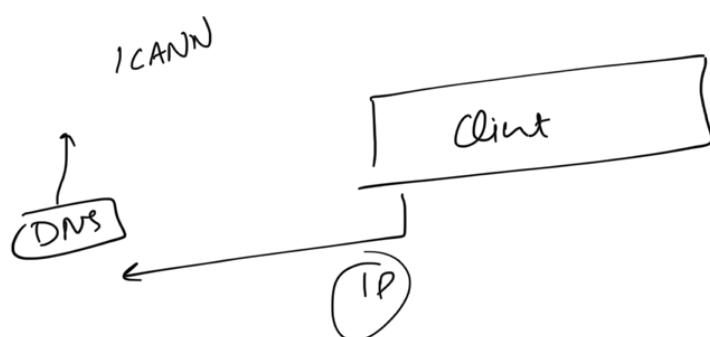
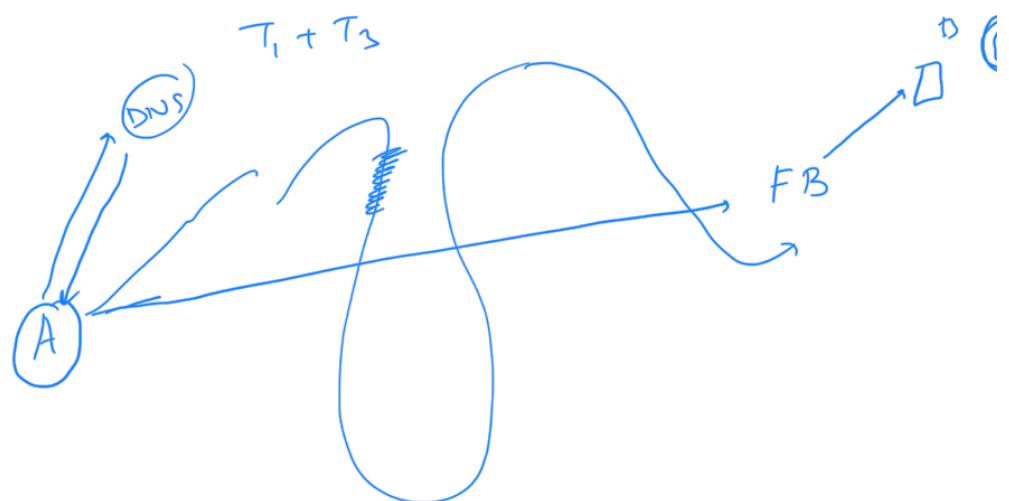
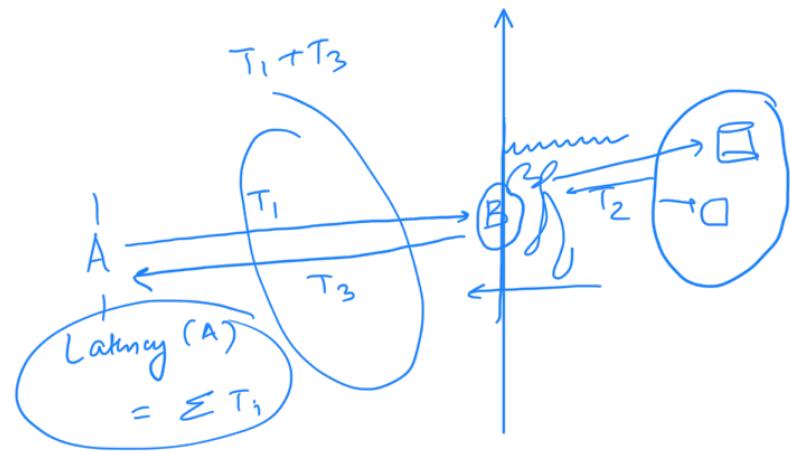
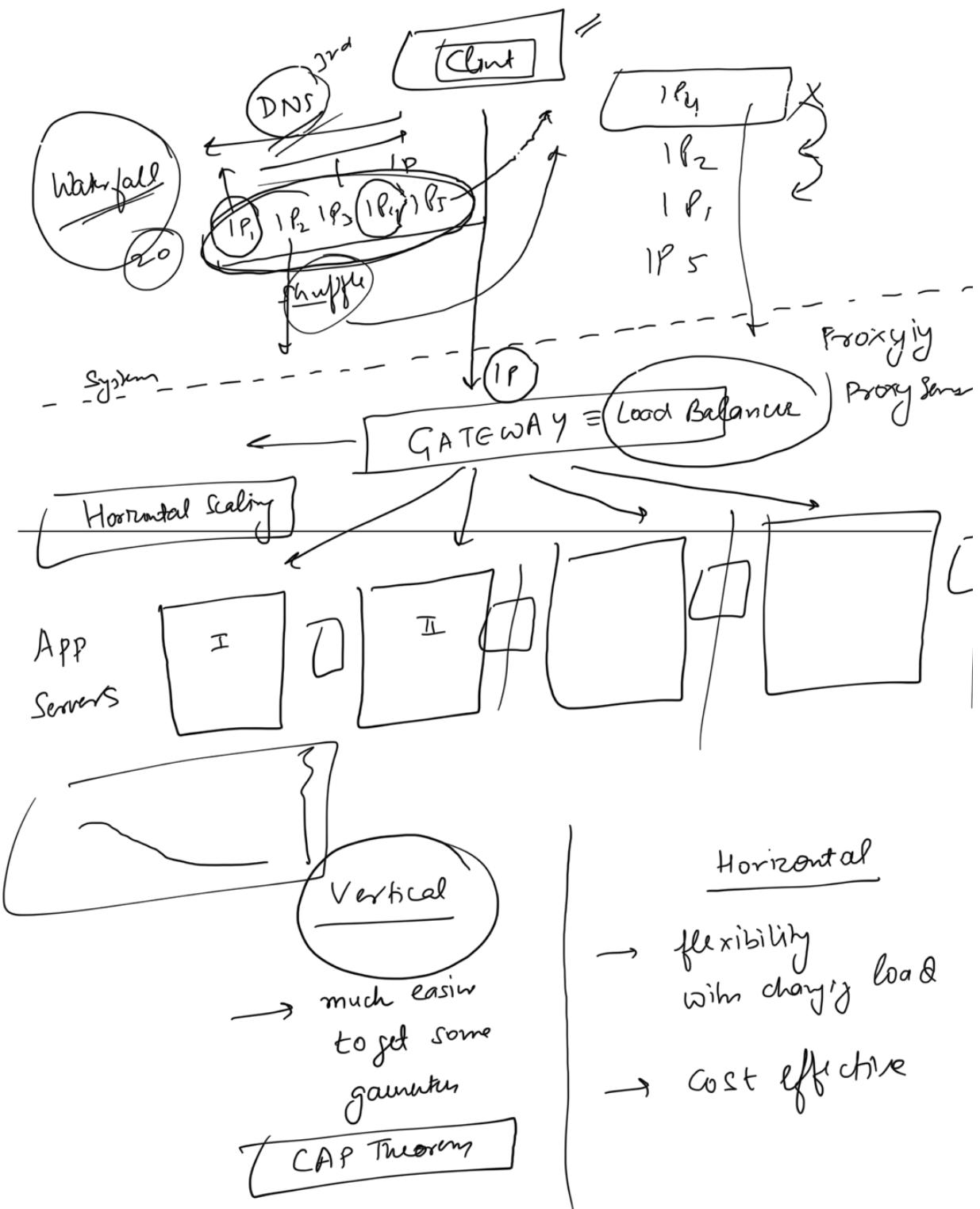
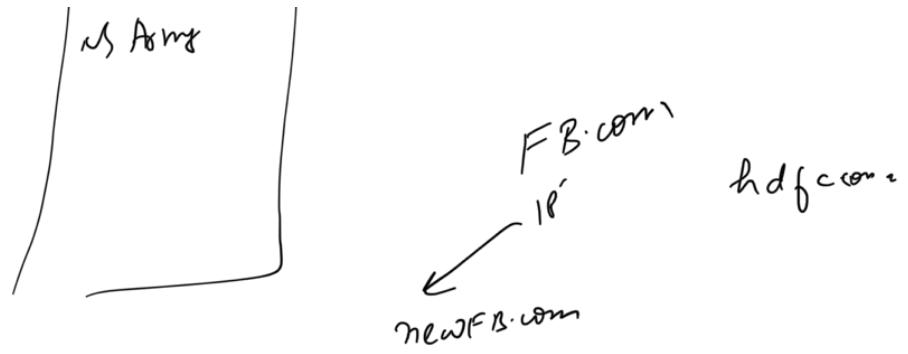
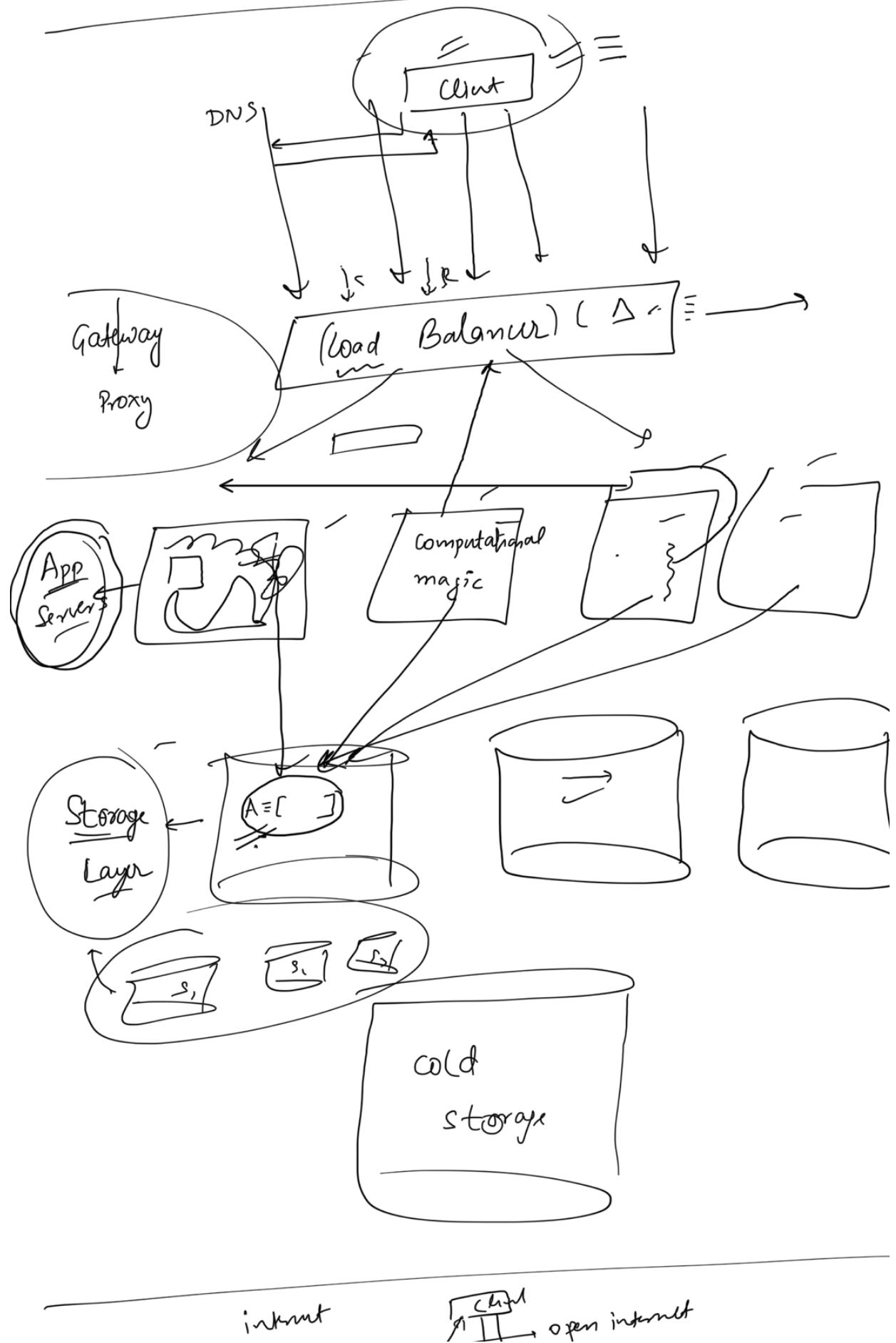


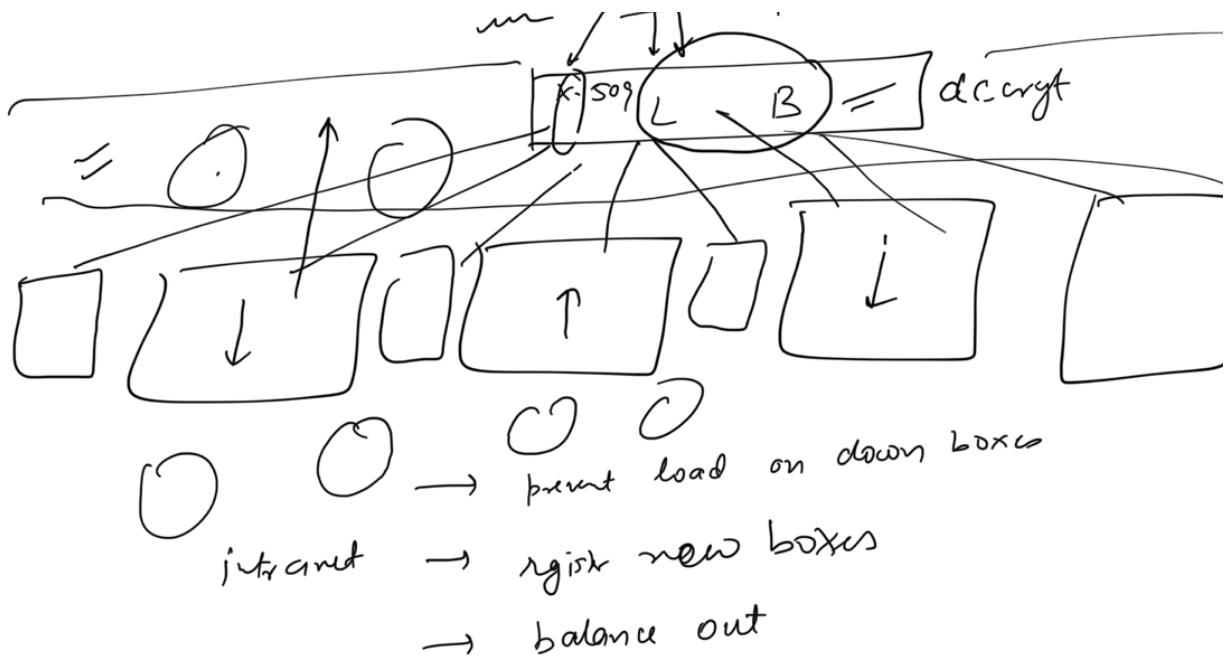
High Level Design



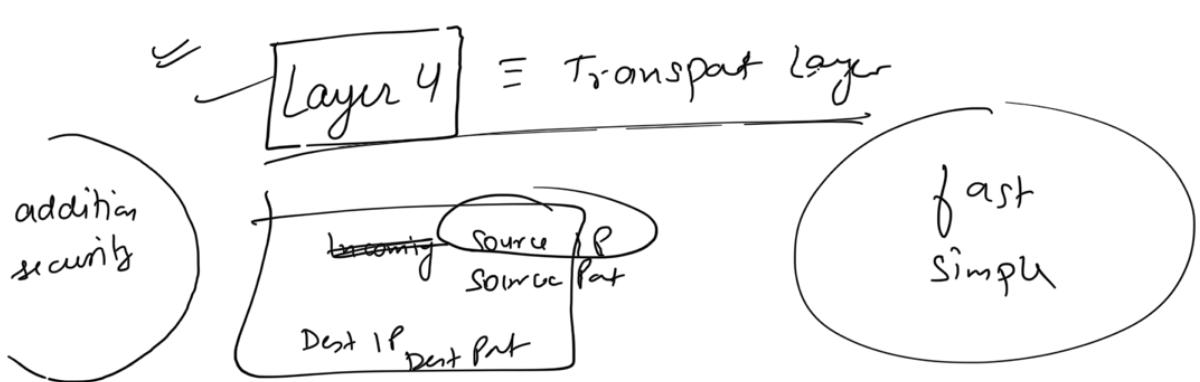








- X. 509 -

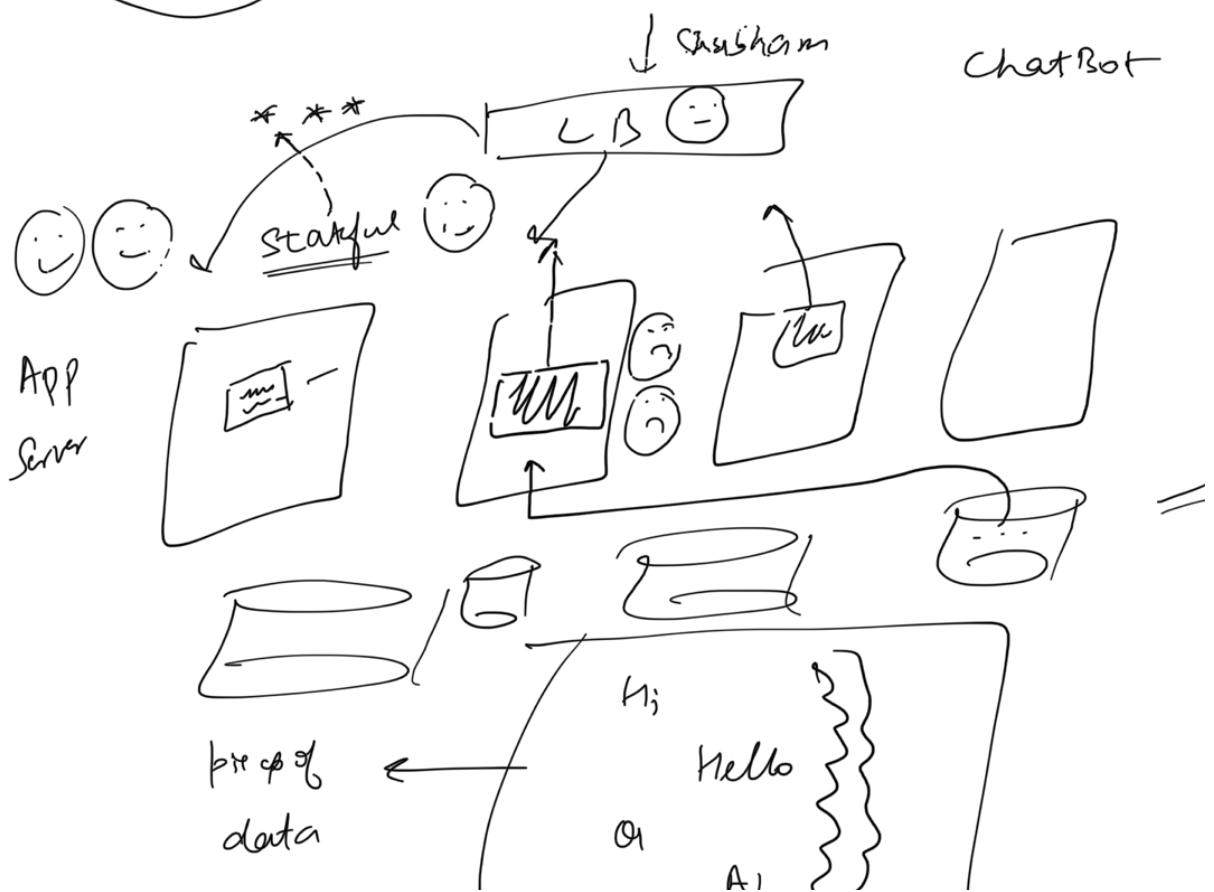
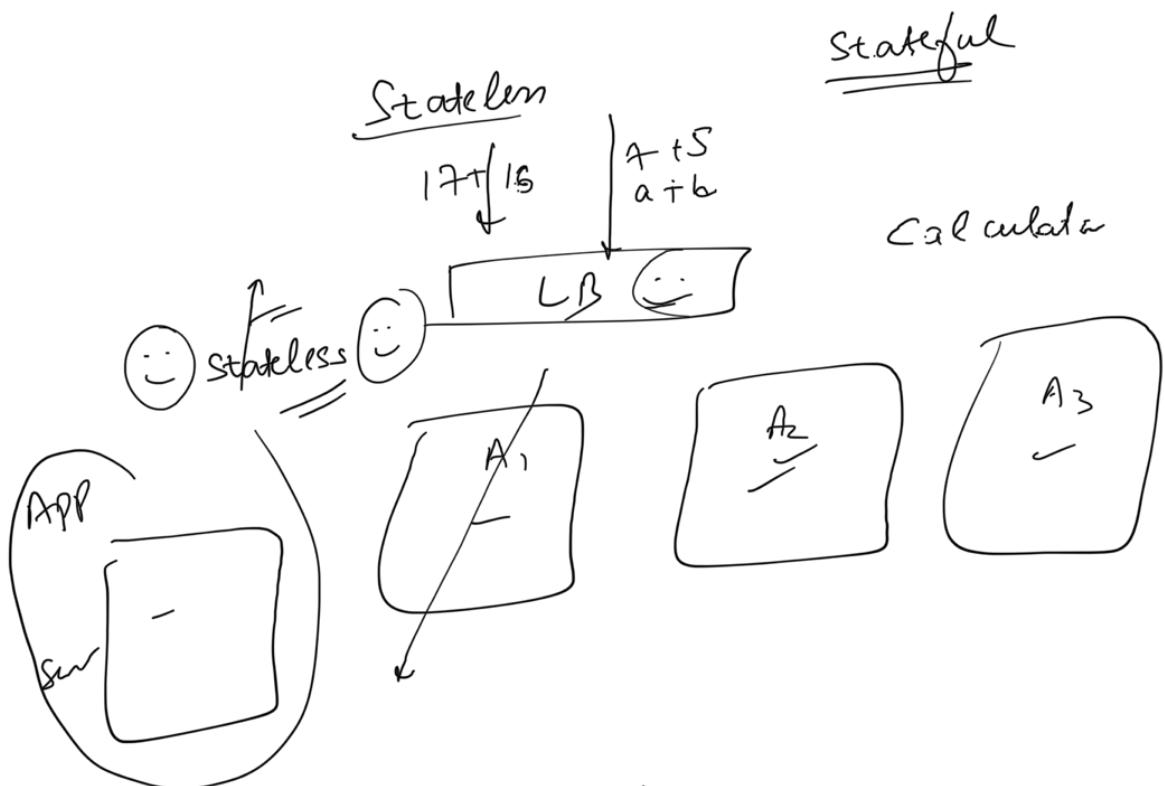


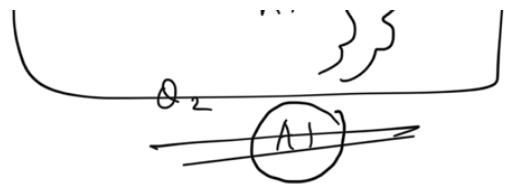
✓ Layer + LB = Application

$$\text{User ID} = 707$$
$$\text{vrid} = 107$$

much better flexibility

Slowur

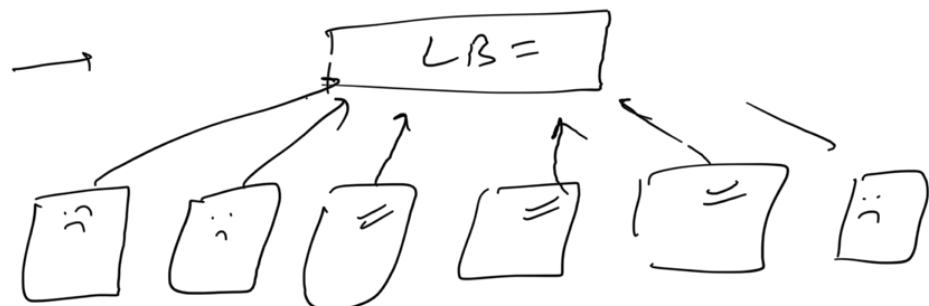
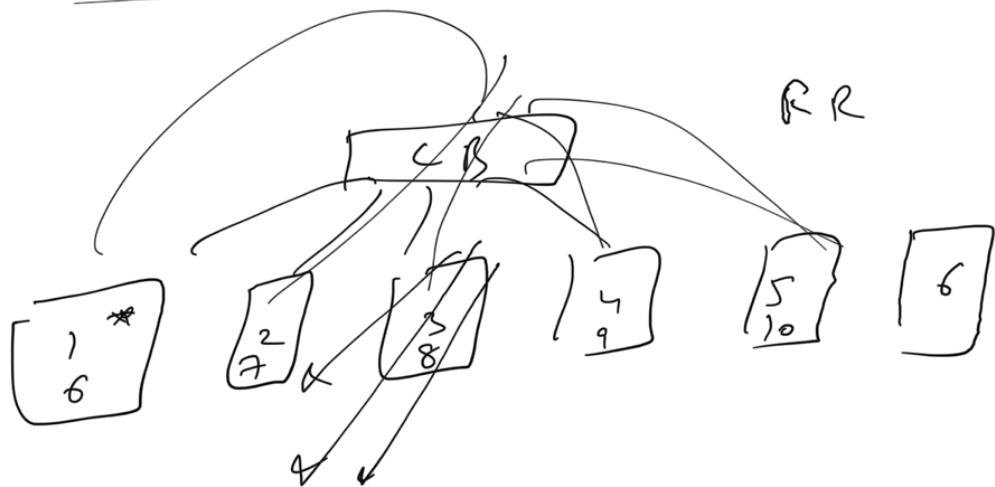




Load Balancing

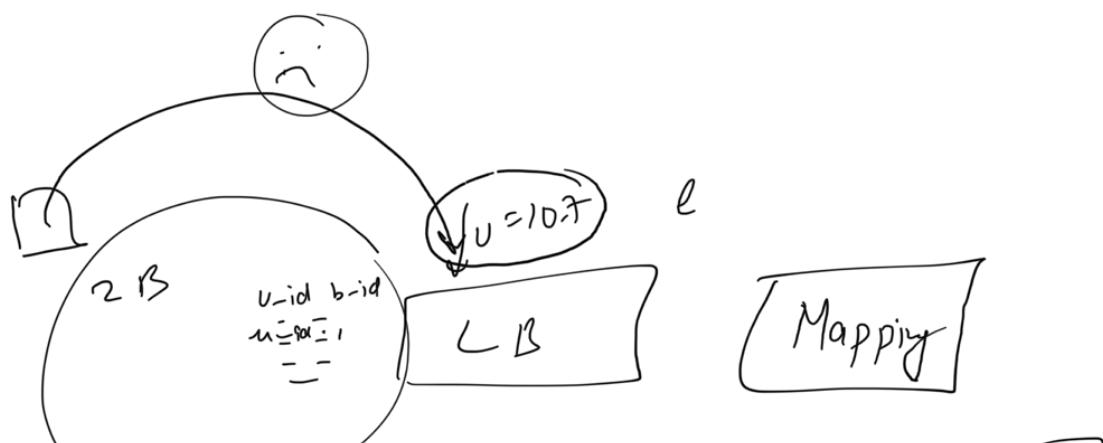
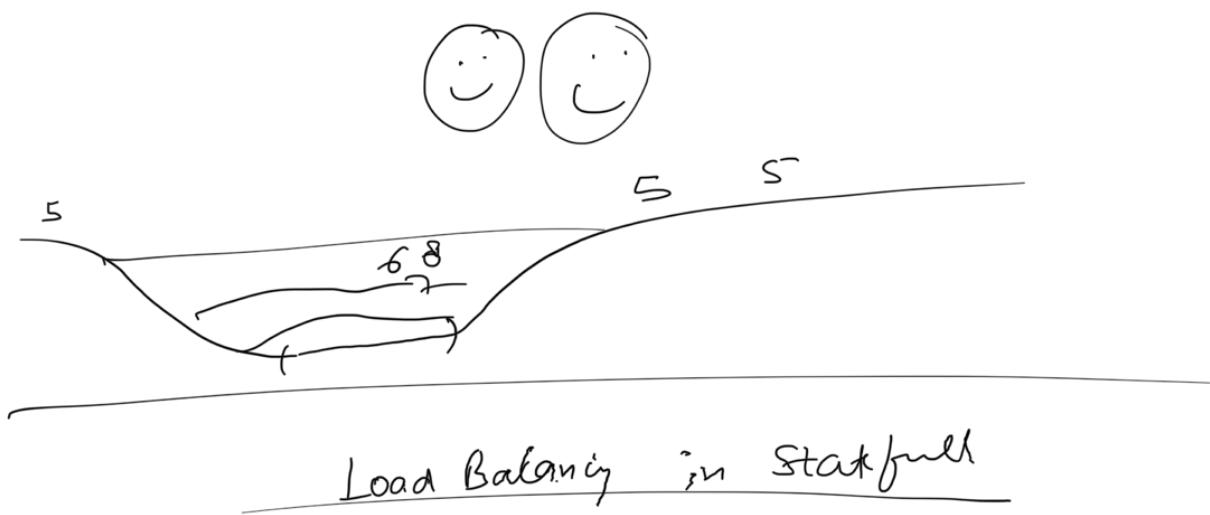
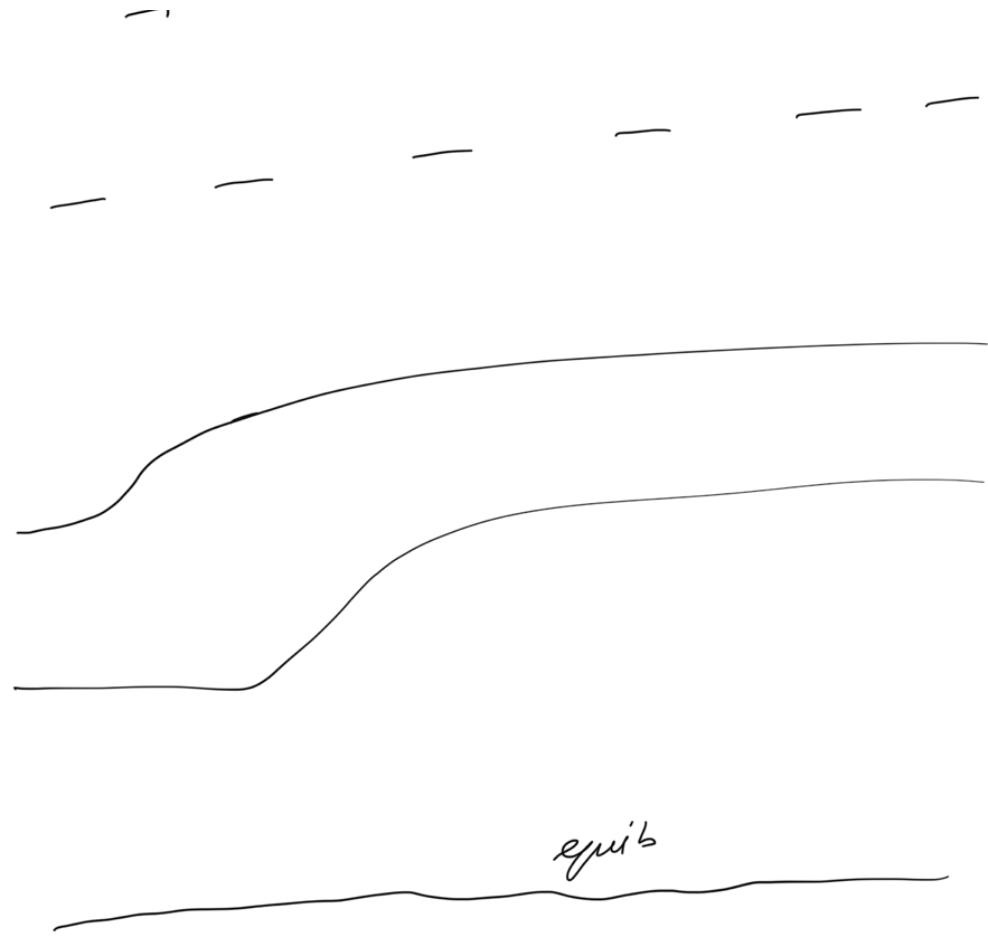
station

→ Round Robin



WRR
→ (Least connection first)

→ WRR ARR





BAD!!! 10 boxes \equiv 10 App Scans

$$\text{vid } \gamma.10 = \boxed{}^{0-9}$$

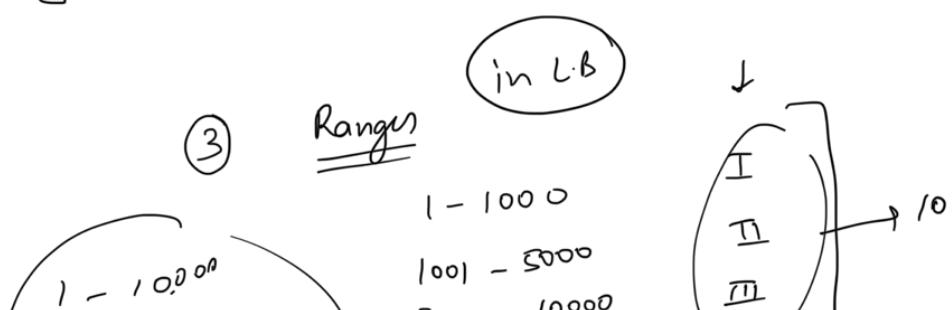
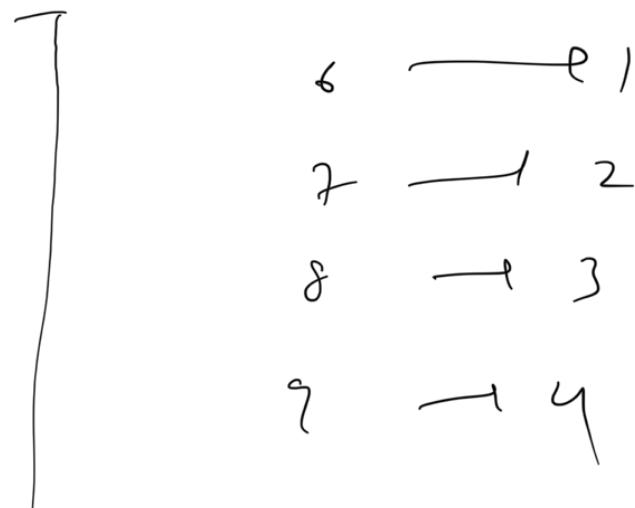
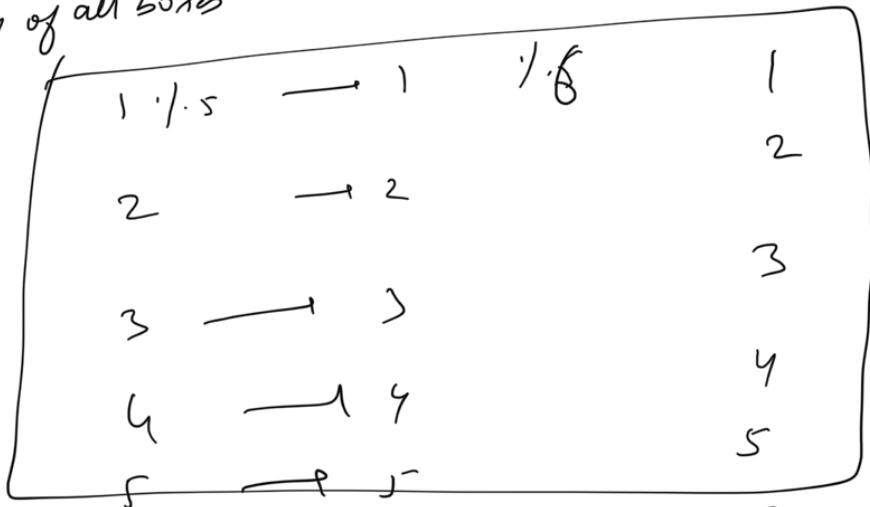
The diagram shows the construction of a surface S from a base polygon A and a boundary polygon B . The base polygon A is labeled $A_gush = 107$ and has vertices $0-1-2-3$. The boundary polygon B is labeled β and has vertices $4-5$. The surface S is formed by gluing the edges of B to the corresponding edges of A , as indicated by arrows. The resulting surface S is a genus-1 surface with two boundary components.

→ aurav of stickiness
.....
.....

0 0 using LIB

→ LIB ☺

Download
On server inc / disc
you will have 1.6
to move state of all boxes
internally



5001 - 10¹⁸

Bottleneck: Add / Removal
of servers is inefficient

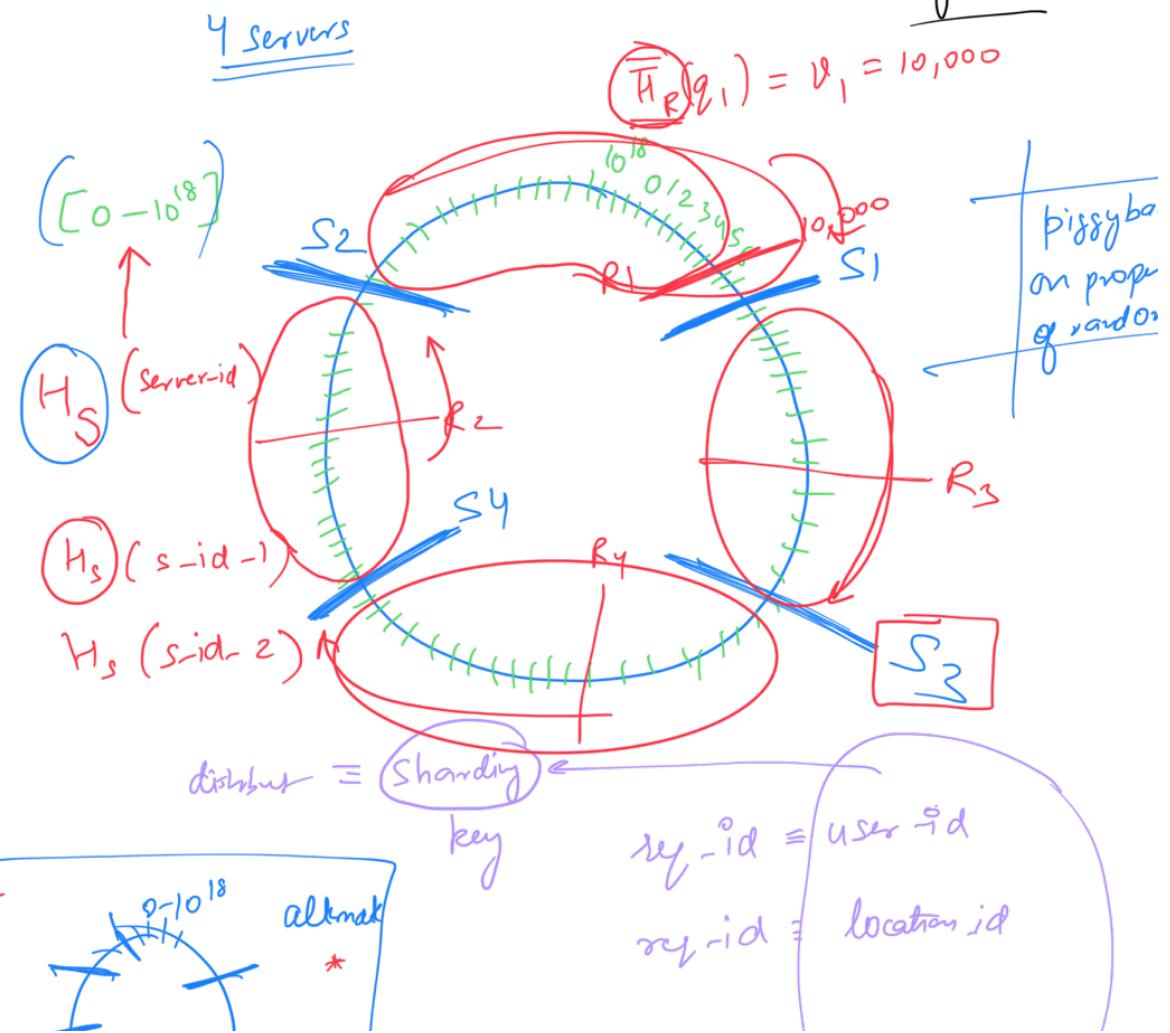


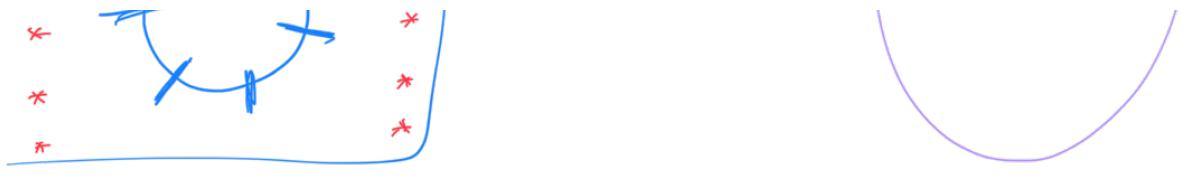
④

Consistent Hashing

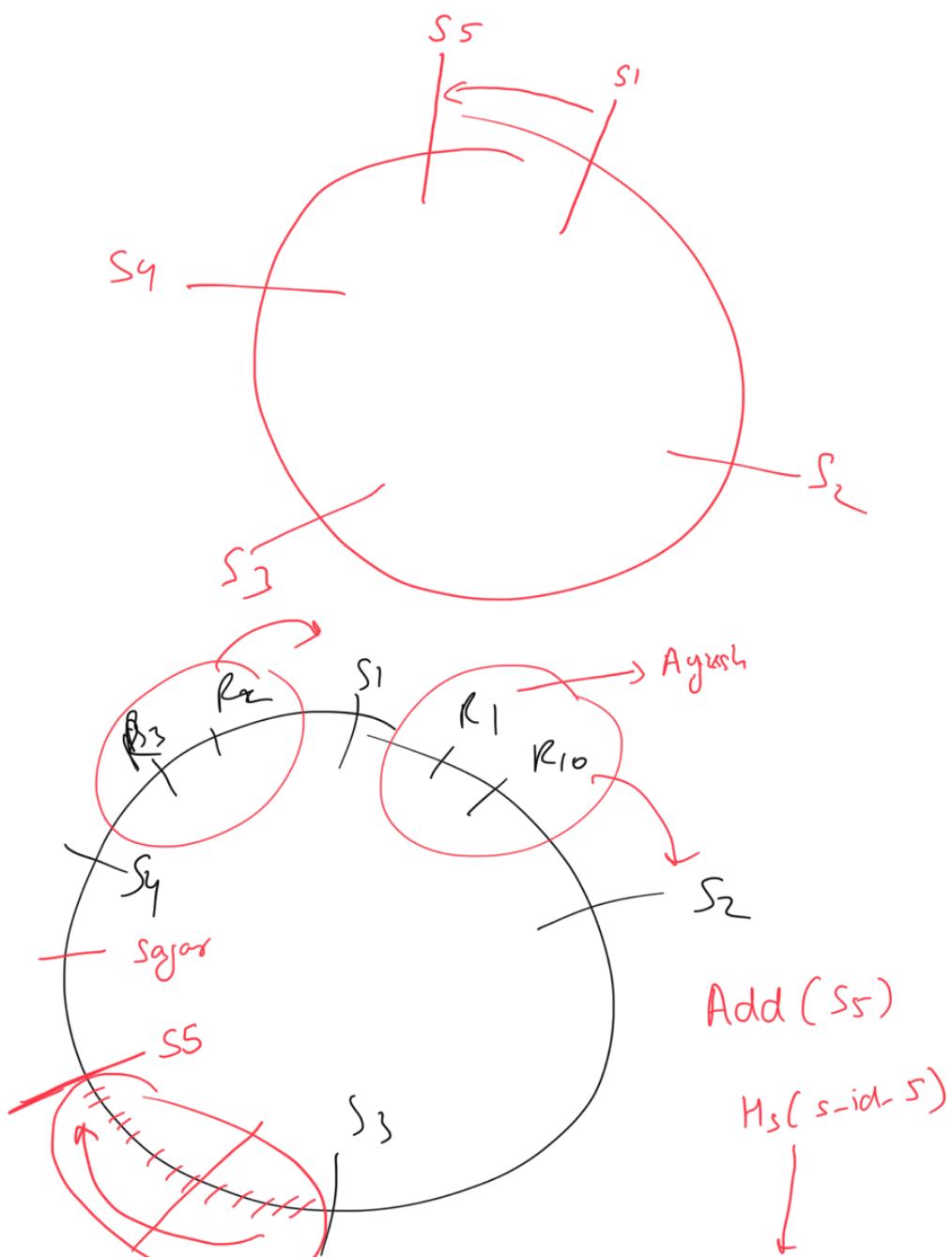
LB for stateful Application

const add/del/removal
of boxes





$H_R =$ (ref-id)
 $[0 - 10^{18}]$



Lokesh

I



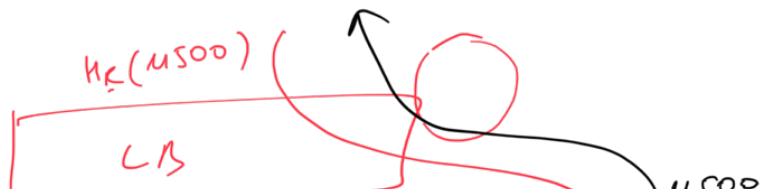
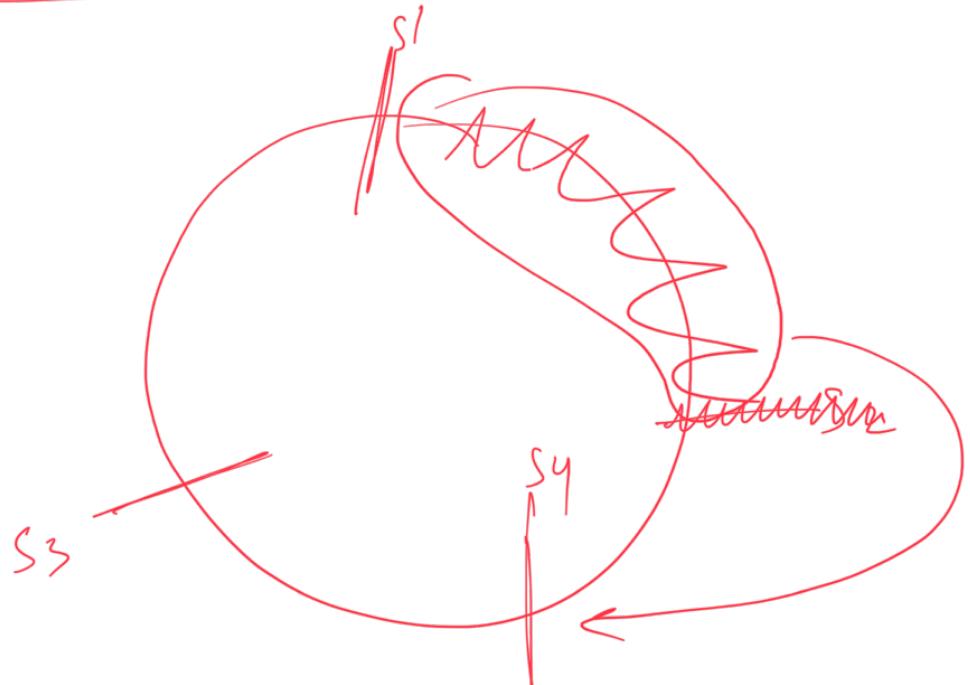
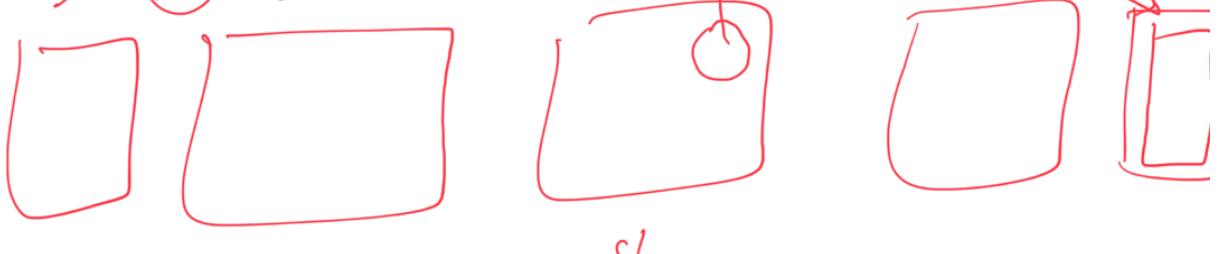
Amount of state transfer that happens
has been reduced a lot

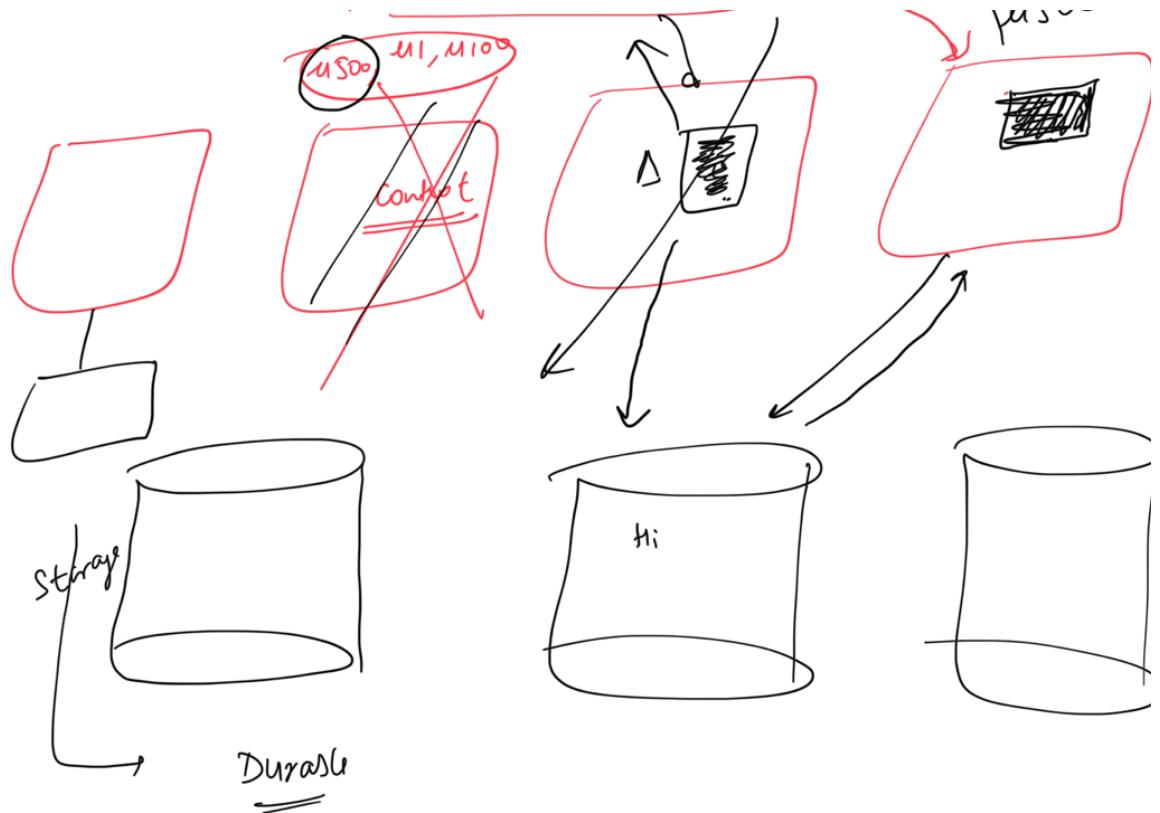
by a factor of # servers

II

III

downtime will also be low.....
you don't have downtime for every user. Small set of
users get impacted





→ we are not losing the ultimate source of truth

(A) cold start = request latency inc for some time
 //
 Hi

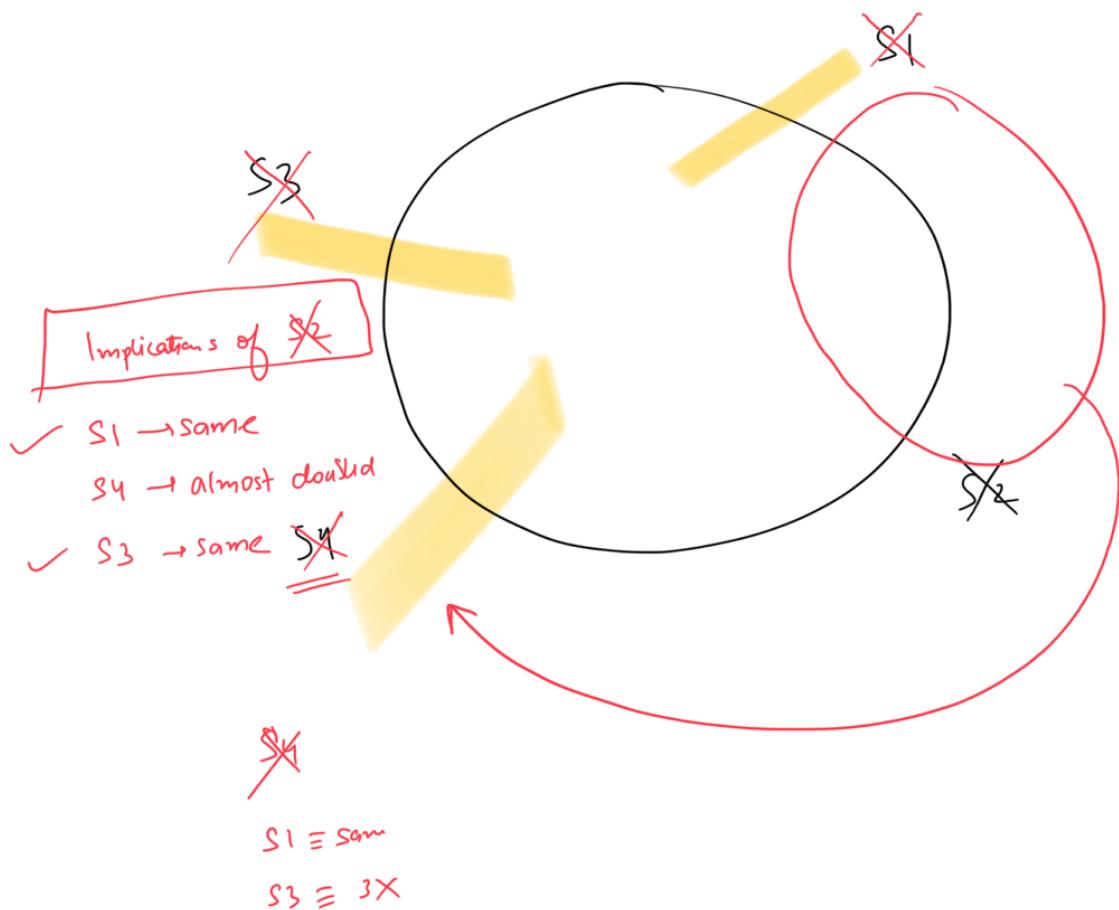
Hello
 01
 A1
 or
 Online Learnig
 Algo's =
 modul

Az

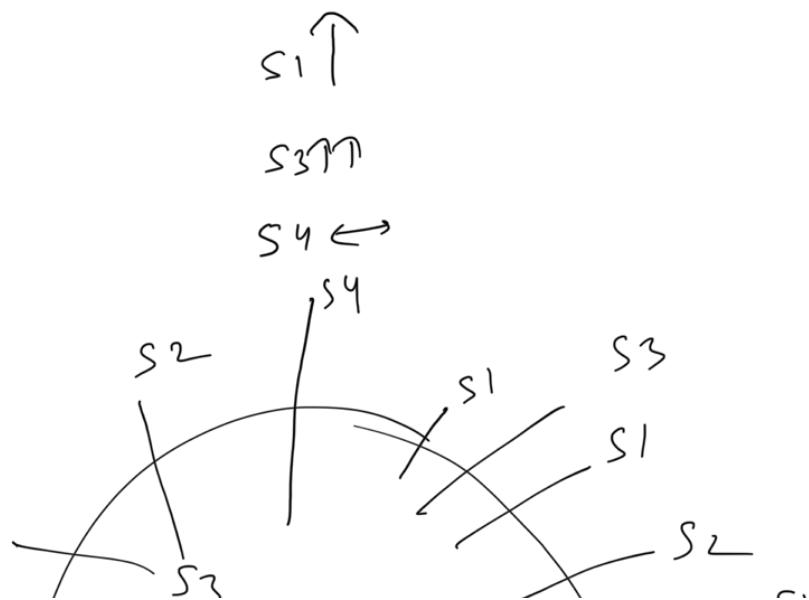
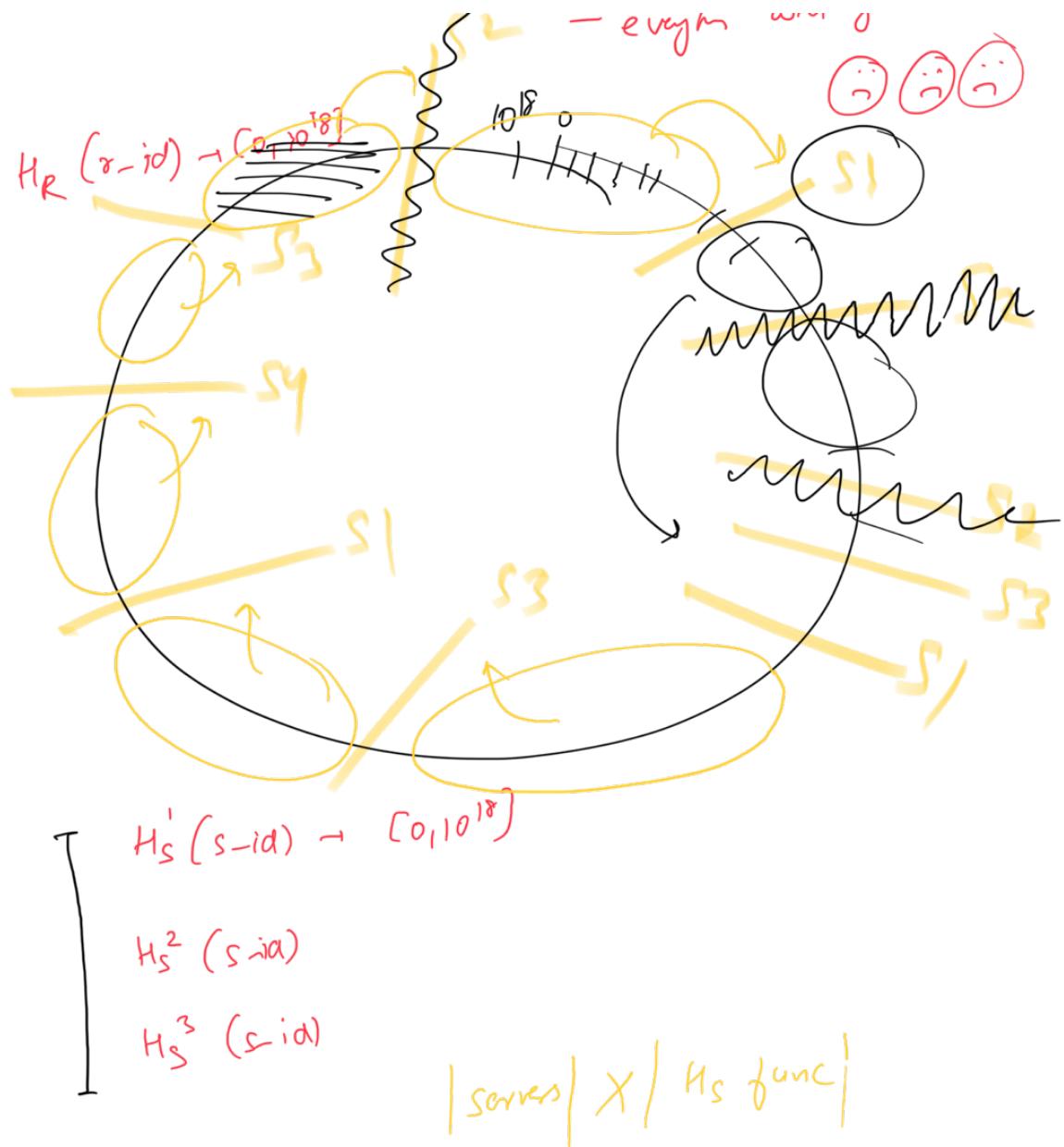
(B) pre-emptive copying

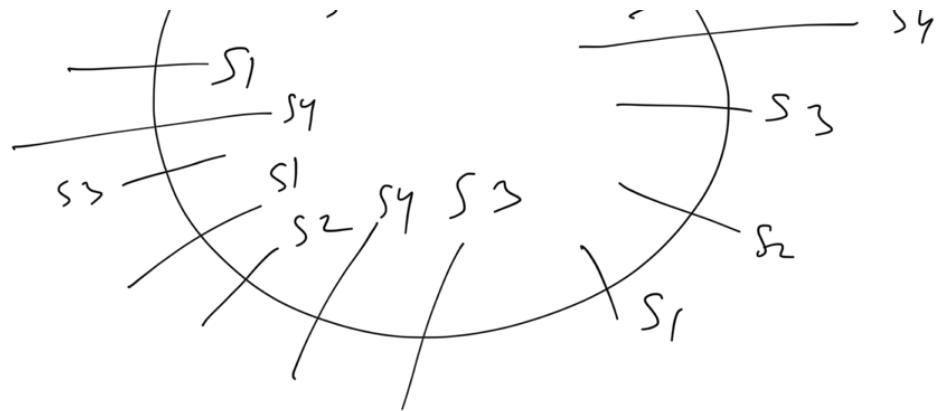


* Solved problem
 * impact of server addition or removal
 * is minimized



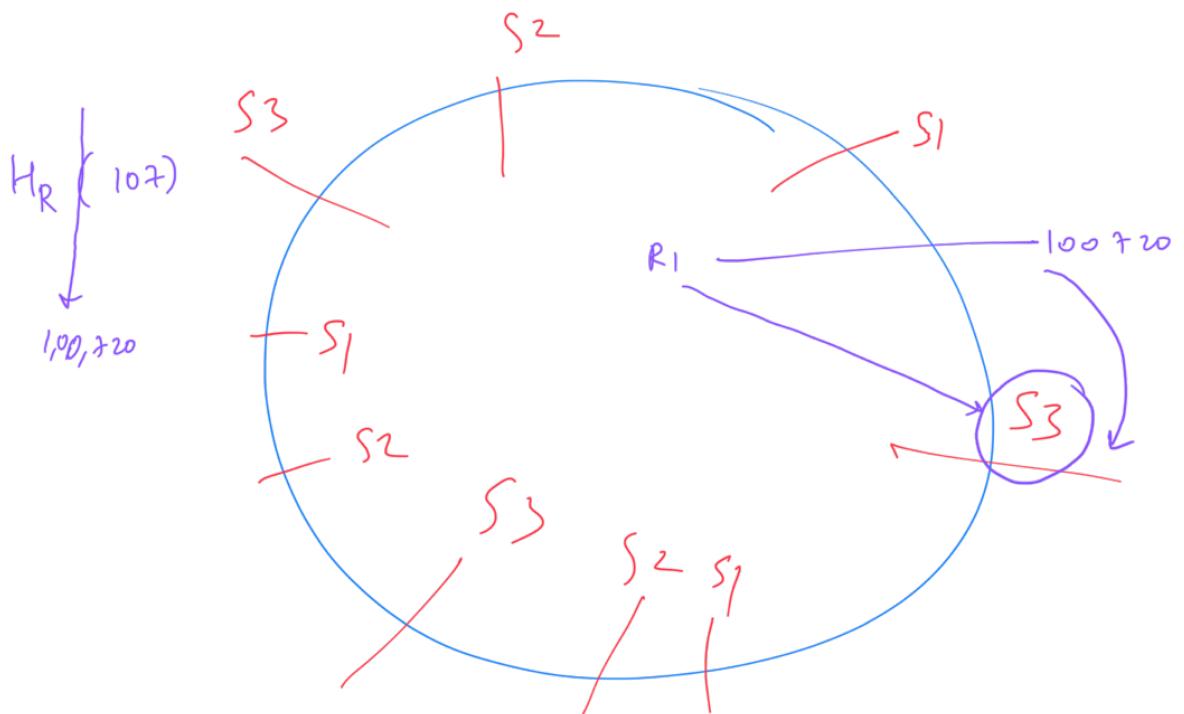
"Cascading Failures"
 until go down





Solved

Cascading Failures
[Now, the load will get divided almost equally]

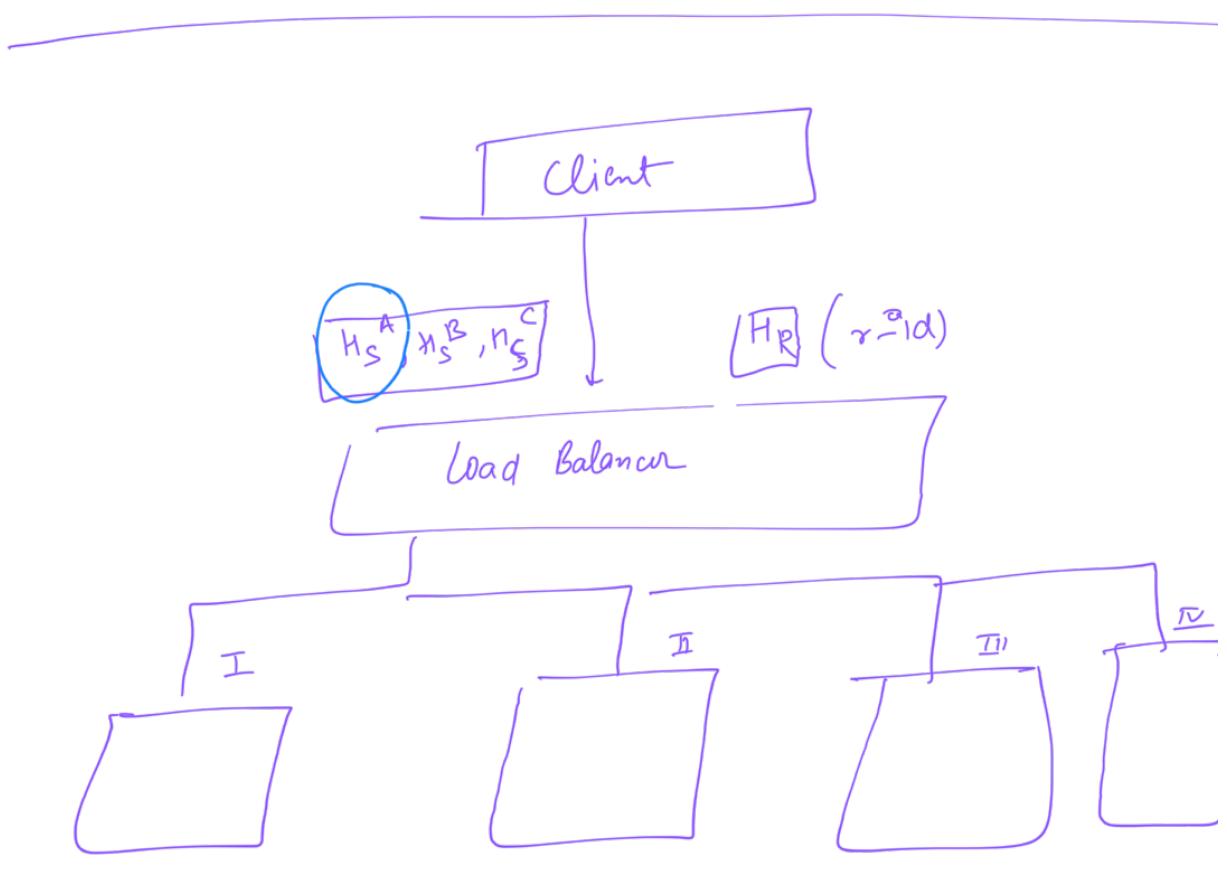


\approx $|H_s| \uparrow \uparrow \equiv$ Randomness ↑↑
 \downarrow
 Equitable Distribution ↑↑

~ L

$$(\mu_s)_{TTT} = \text{time for resolution}_{TTT}$$

ASG



Io server

↓ ↗ →

