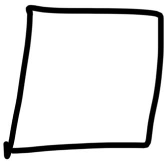
24|6|2023

Scaling

Horizontal scaling	Vertical scaling
	
→ load balancing reauired	→ N/A
→ Resilient	→ single point of failure
→ network calls (RPC)	→ Interprocess communicator
→ data inconsistency	→ consistent
→ scales well as users increase	→ hardware limit

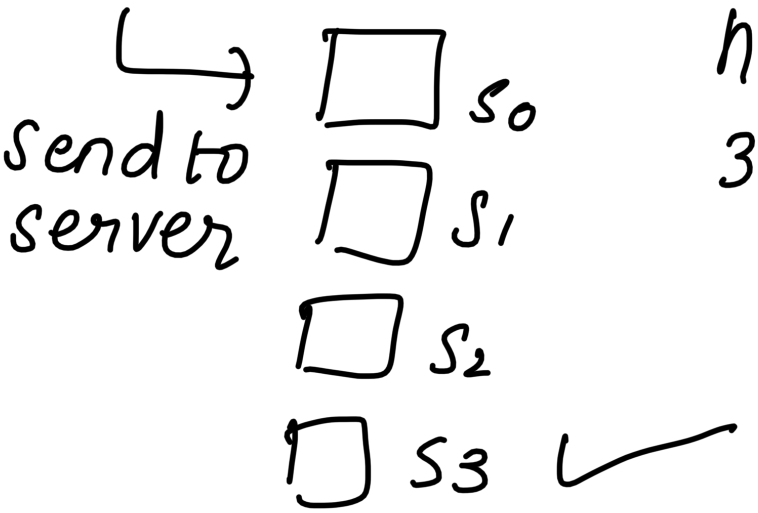
Load Balancing

→ N-servers have load onthem. Balancing load is known as load balancing.

To do that : – consistent hashing
↓
evenly distribute load

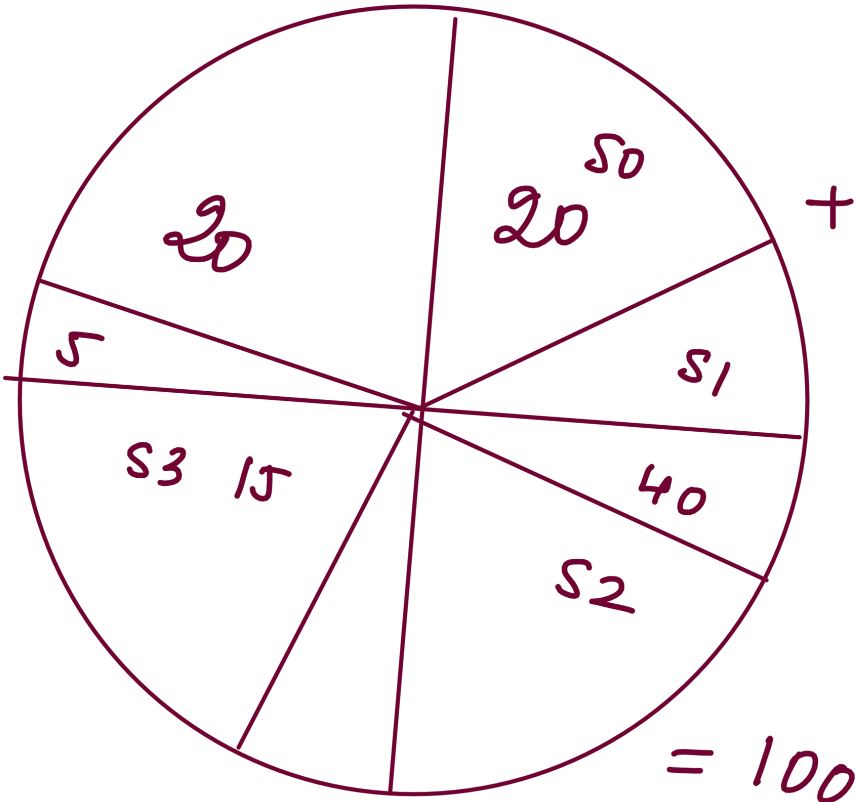
Request Id → 0 to M-1 → sent to server

Take the ID, say r_1 then hash it
$h(r_1) \rightarrow m_1 \% n \rightarrow$ server



$h(10) \rightarrow 3$
$3 \% 4 = 3$

on increasing servers, hash generated server number changes. → consistent hashing

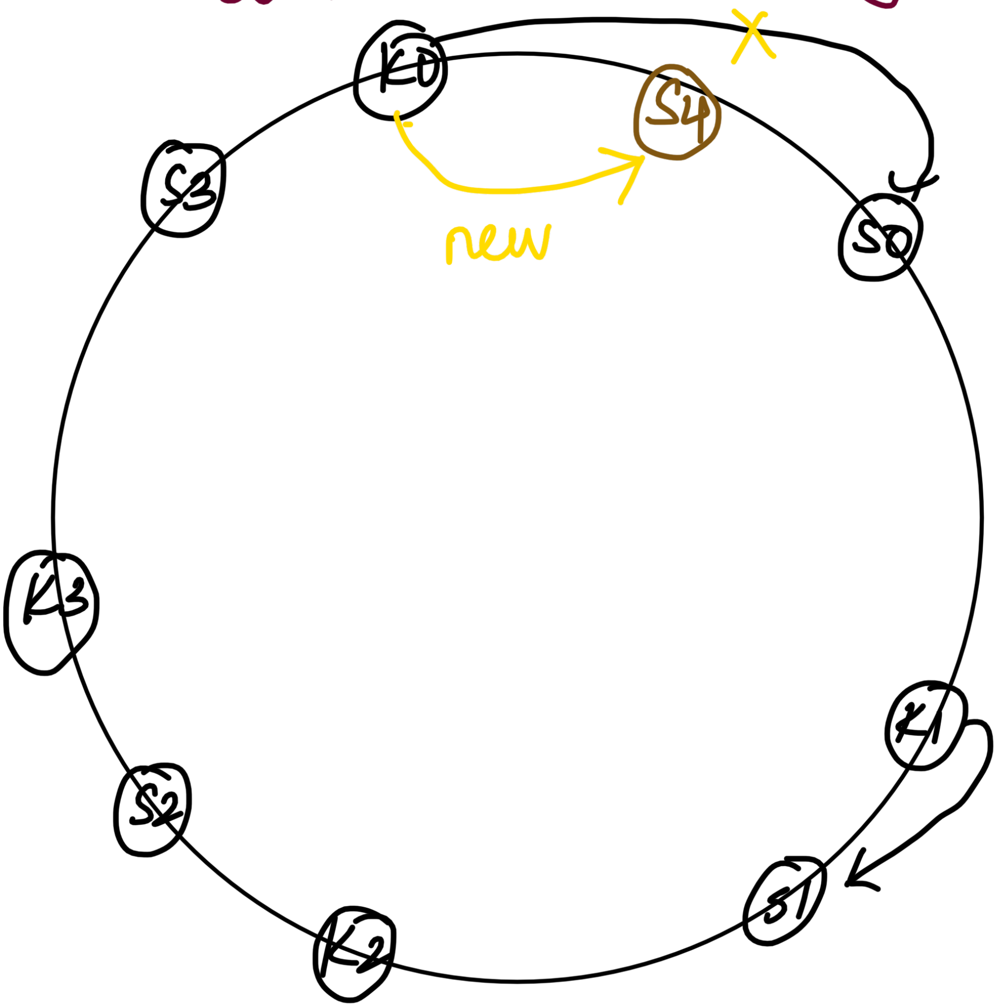


+5 +5 +10
+10 +15
+15 +20
+20

↓
S4

= 100 = M

Consistent Hashing



In simple hashing, when newserver added, almost all keys need to be remapped.

with consistent hashing, on adding new servers, only fraction of keys are relocated