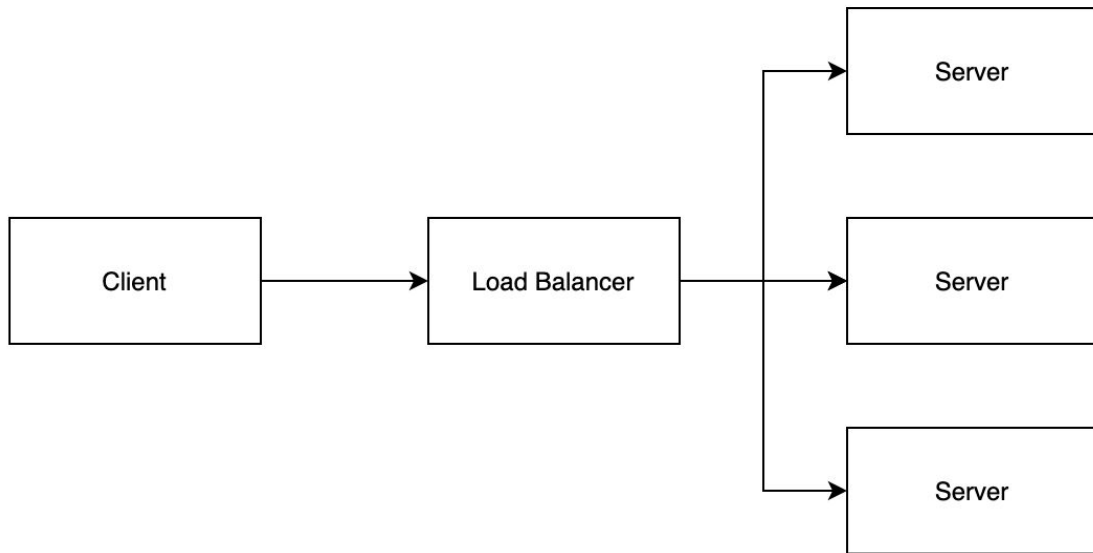# Super Load Balancer

Team 7
Dennis He
Yanchi Li

Project Goals

- Implement load balancer with weighted round robin algorithm that dispatches client requests to different servers
- Have multiple engineer threads on each server to service requests (stretch goal)
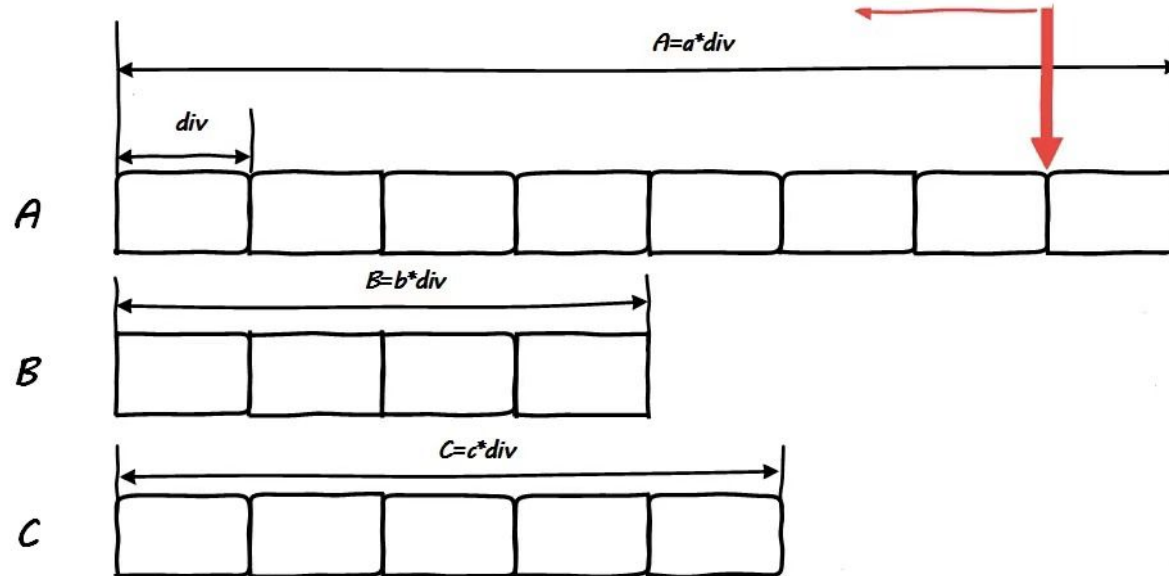- Having chain replication and recovery feature that works with load balancer (stretch goal)

Design and Implementation

Design and Implementation

Weighted Round Robin Scheduling

$$A=a*div$$

$$div$$
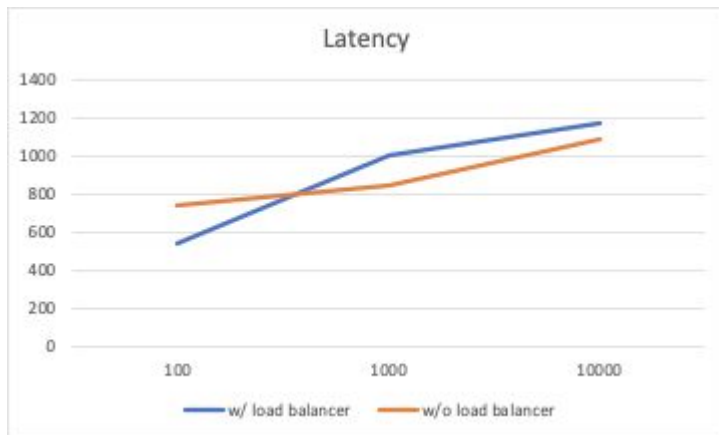
A

$$B=b*div$$

B

$$C=c*div$$

C

What we achieved and wasn't able to achieve?

- We implemented the the round robin algorithm that works with load balancer by distributing workloads to different server nodes based on weights.

- We have not yet achieved our stretch goals of having chain replication and recovery features that work with the load balancer.

Evaluation of System

Evaluation of System

- The number of requests received by each server is proportional to their weights
- Load Balancer increases workload capacity (ex: distributed to multiple server nodes)
- Load Balancer increases scalability (ex: as workload increases, more servers can be added to the system)

Key Takeaways

- Load balancer helps to distribute requests among servers
- Load balancer works as a middleware between clients and servers
- We are able to improve system performance by using load balancer