Load Balancing and Optimal Resource Allocation

Dr. Tushar Sandhan

Contents

- Introduction
    - What is load balancing?
    - How does load balancing work?
    - Hardware vs. Software Load Balancing
    - Benefits of load balancing
    - Load Balancing Methods
- Deep Learning Based Load Balancing
- Power allocation using Deep learning for D2D Communication

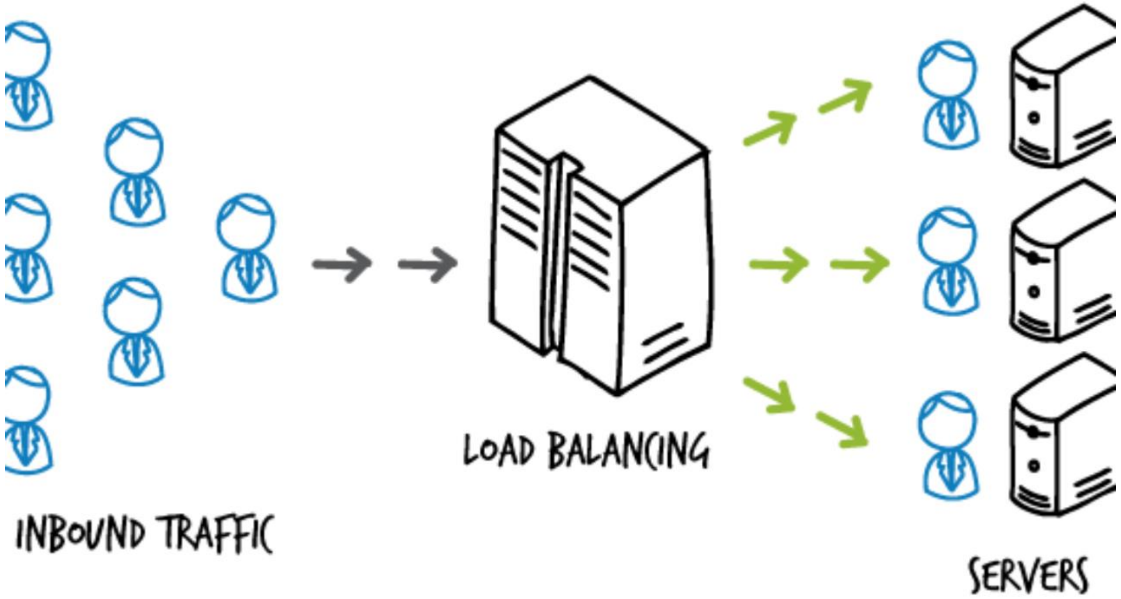
Introduction

- Wireless networks are getting more and more popular and have become an essential part of our lives with the ever-increasing use of IoT devices. The reality is that users expect high-quality connectivity in all scenarios, especially in public spaces with crowded networks and multiple concurrent users downloading and uploading content simultaneously.

- Hundreds of devices want to connect to a network comprised of multiple access points and a limited spectrum. For all of those devices to receive a decent connection quality, throughput, and delay, there shouldn’t be access points overloaded. Otherwise, it would not be easy to provide service for each client device connected to the network.

What is load balancing?

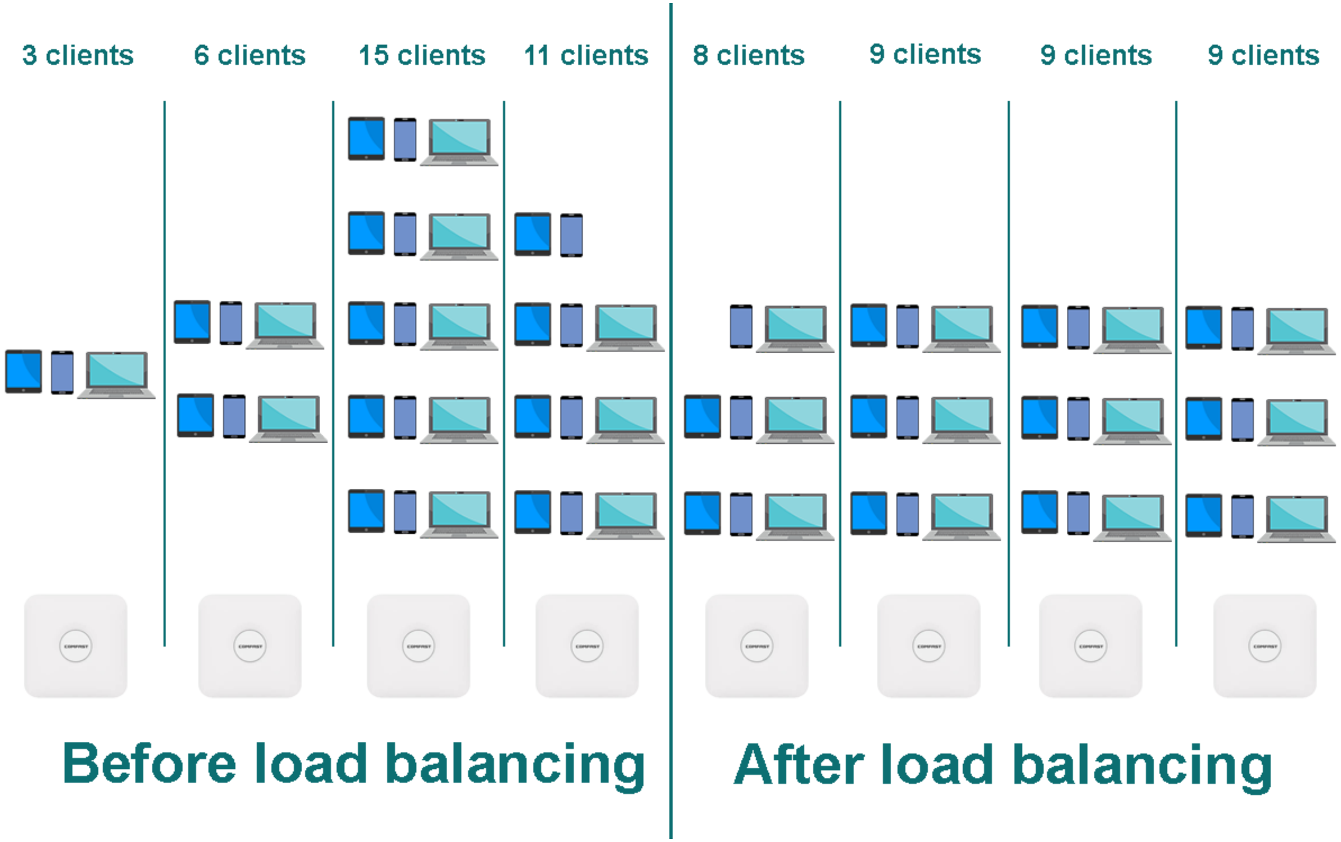
- Load balancing lets you evenly distribute network traffic to prevent failure caused by overloading a particular resource. This strategy improves the performance and availability of applications, websites, databases, and other computing resources. It also helps process user requests quickly and accurately.



How does load balancing work?

- Load balancing ensures that client devices are distributed evenly, so no single AP is simultaneously overloaded with too many client devices. Therefore, maximum number of client devices can be served by various APs, delivering better performance and an improved user experience.

- It optimizes throughput for all client devices by continually optimizing user associations to give each client device optimal throughput. This improves the throughput for each client device and dynamically balances the client load for the network.



How does load balancing work?

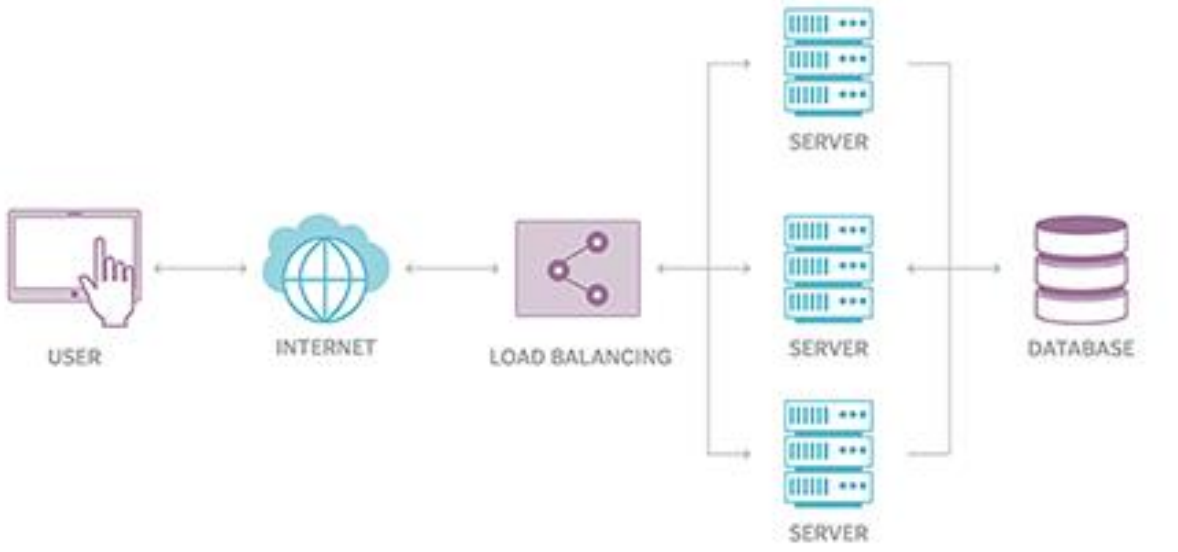
Load balancers handle incoming requests from users for information and other services.

They sit between the servers that handle those requests and the internet.

Once a request is received, the load balancer first determines which server in a pool is available and online and then routes the request to that server.

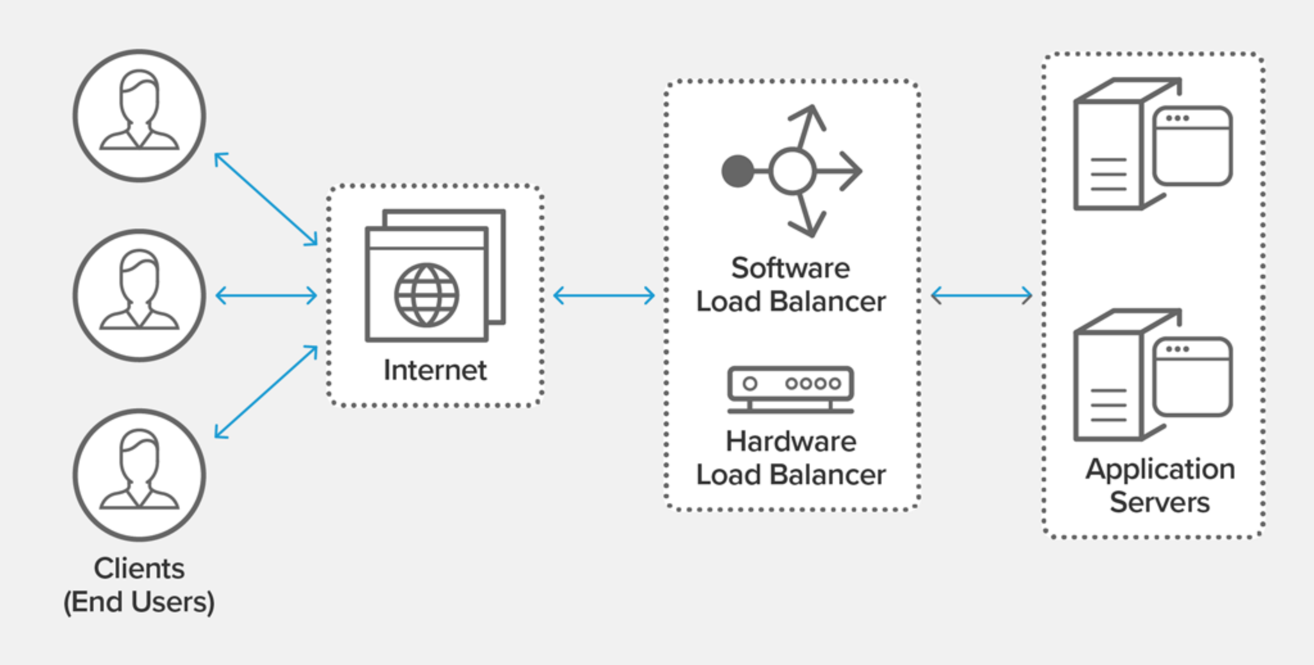
During times of heavy loads, a load balancer acts promptly and can dynamically add servers in response to spikes in traffic.

Conversely, load balancers can drop servers if demand is low.



Hardware vs. Software Load Balancing

- Both hardware and software load balancers have specific use cases. Hardware load balancers require rack-and-stack appliances.
- software load balancers are installed on standard x86 servers, VMs or cloud instances. Hardware load balancers are sized to handle peak traffic loads. Software products are typically licensed based on bandwidth consumption.



Hardware vs. Software Load Balancing

Hardware load balancers

- Pros
  - They provide fast throughput, as the software is run on specialized processors.
  - These load balancers offer better security, as they're handled only by the organization and not by any third party.
  - They come with a fixed cost at the time of purchase.

- Cons
  - Hardware load balancers require extra staff and expertise to configure and program them.
  - They can't scale when a set limit on several connections has been reached. When this happens, connections are either refused, dropped or degraded, and the only option is to purchase and install additional machines.

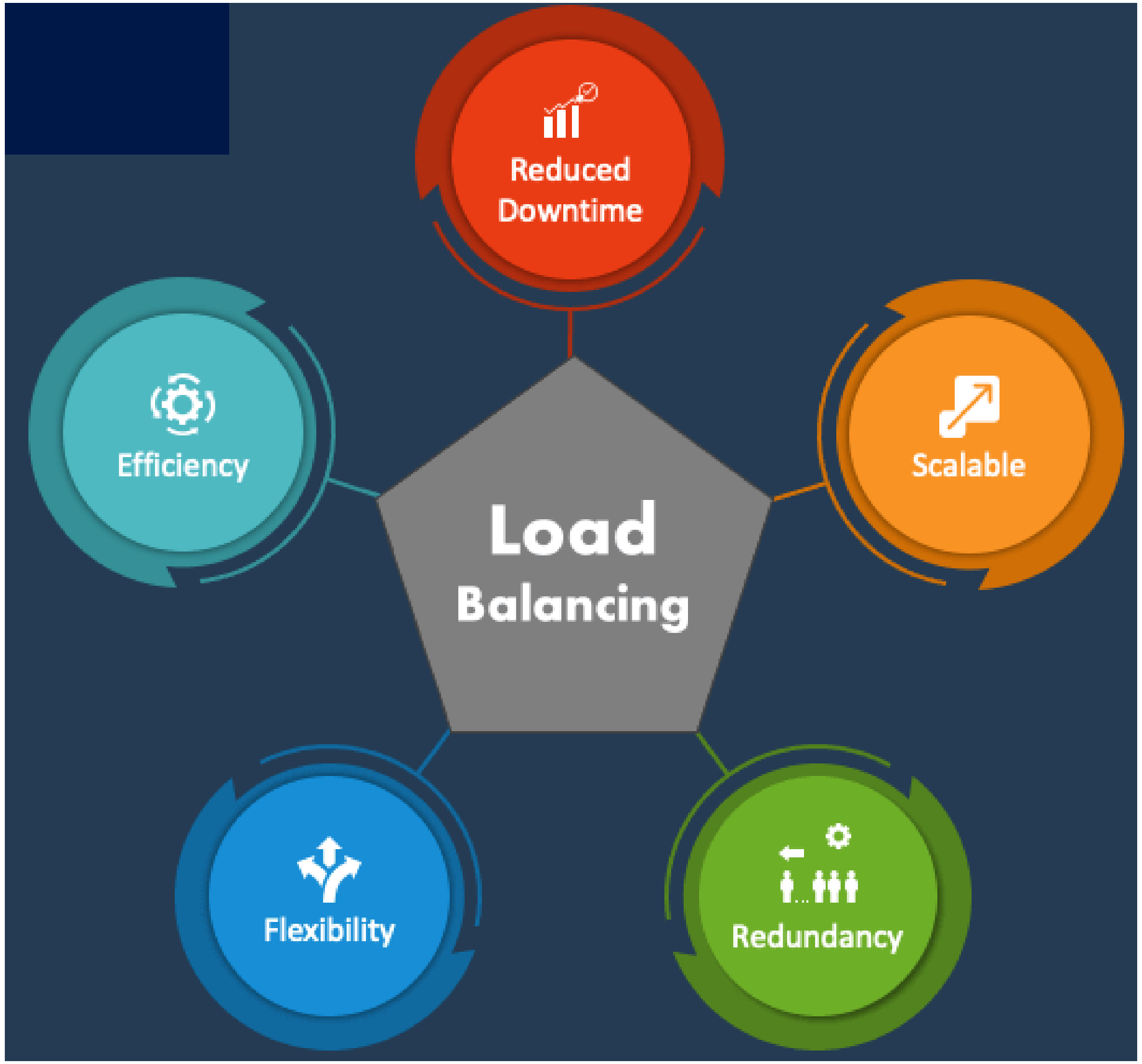
Hardware vs. Software Load Balancing

Software load balancers

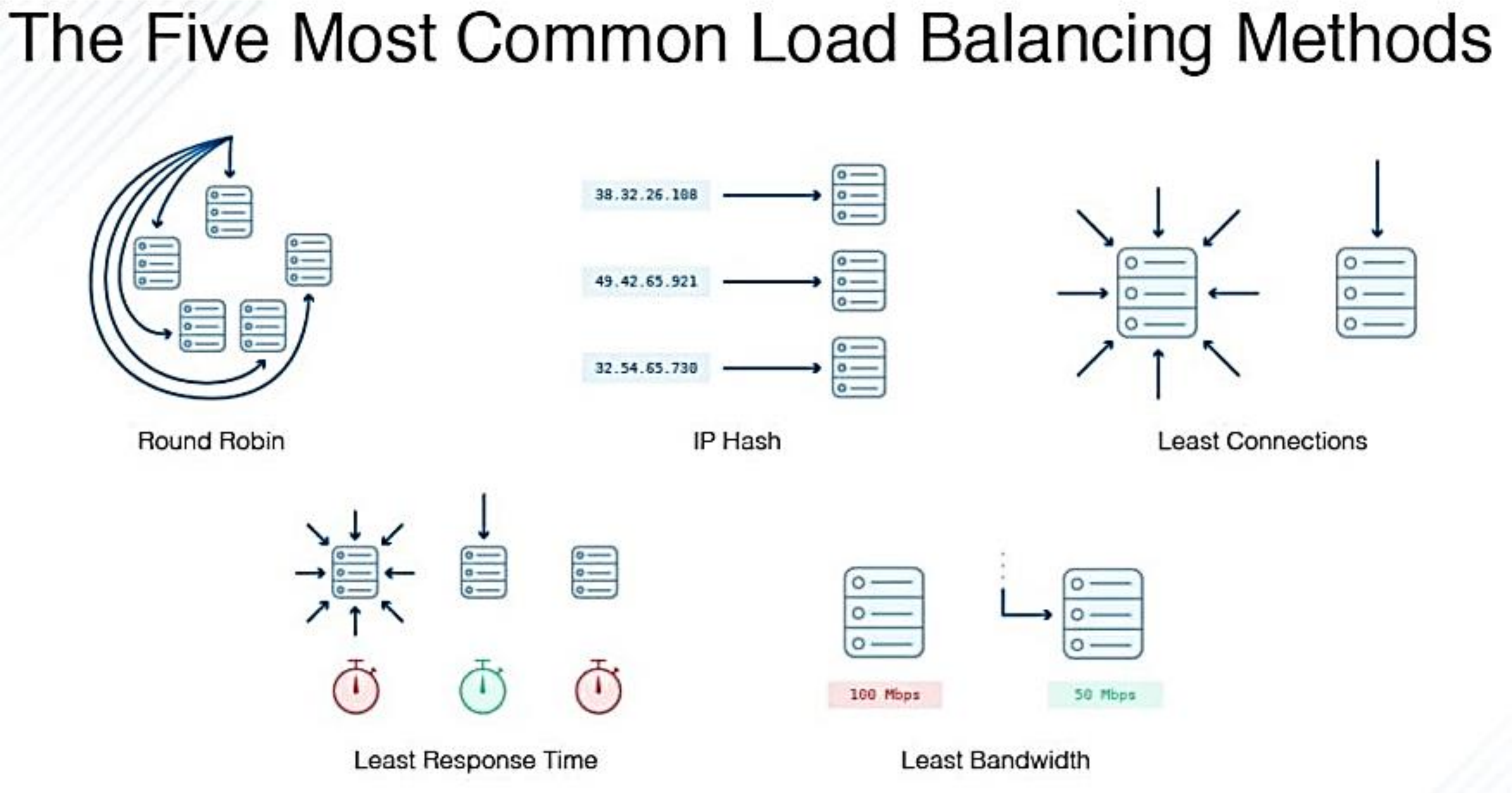
- Pros
  - They offer the flexibility to adjust to the changing needs and requirements of a network.
  - By adding more software instances, they can scale beyond the initial capacity.
  - They offer cloud-based load balancing, which provides off-site options that can operate on an elastic network of servers. Cloud computing also offers options with various combinations, such as hybrid with in-house locations.
    - For example, a company could have the main load balancer on premises, and the backup load balancer could be in the cloud.
- Cons
  - When scaling beyond capacity, software load balancers might cause an initial delay. This usually happens when the load-balancer software is being configured.
  - Since they don't come with a fixed upfront cost, software load balancers can add ongoing costs for upgrades.

Benefits of load balancing

- Improved scalability.
- Improved efficiency.
- Reduced downtime.
- Predictive analysis.
- Efficient failure management.
- Improved security.



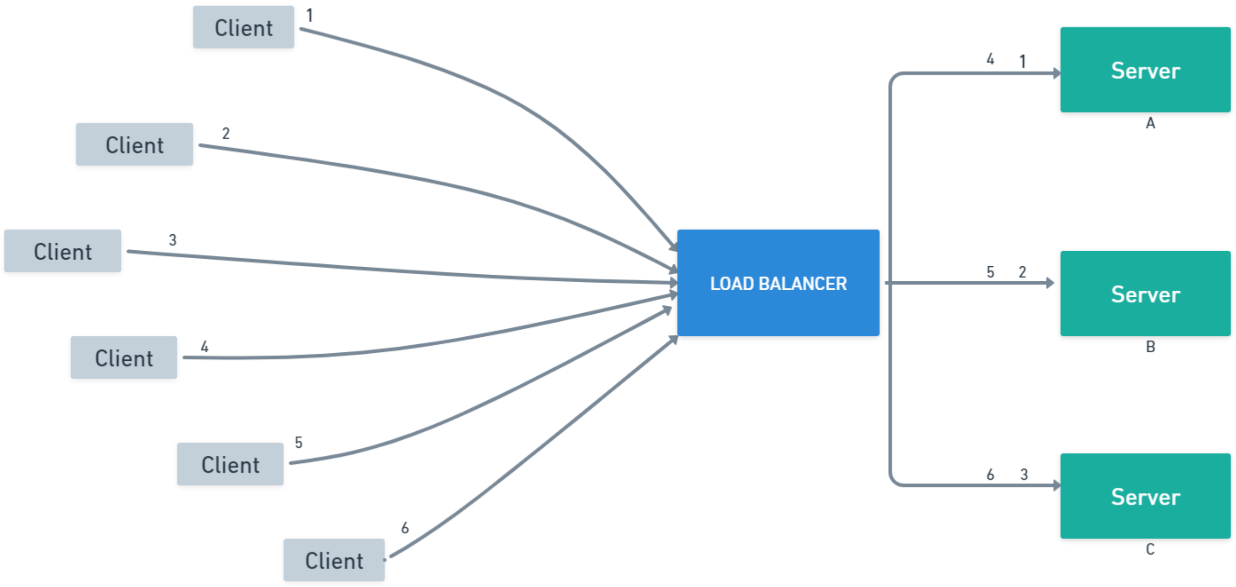
Load Balancing Methods



Load Balancing Methods

Round Robin:

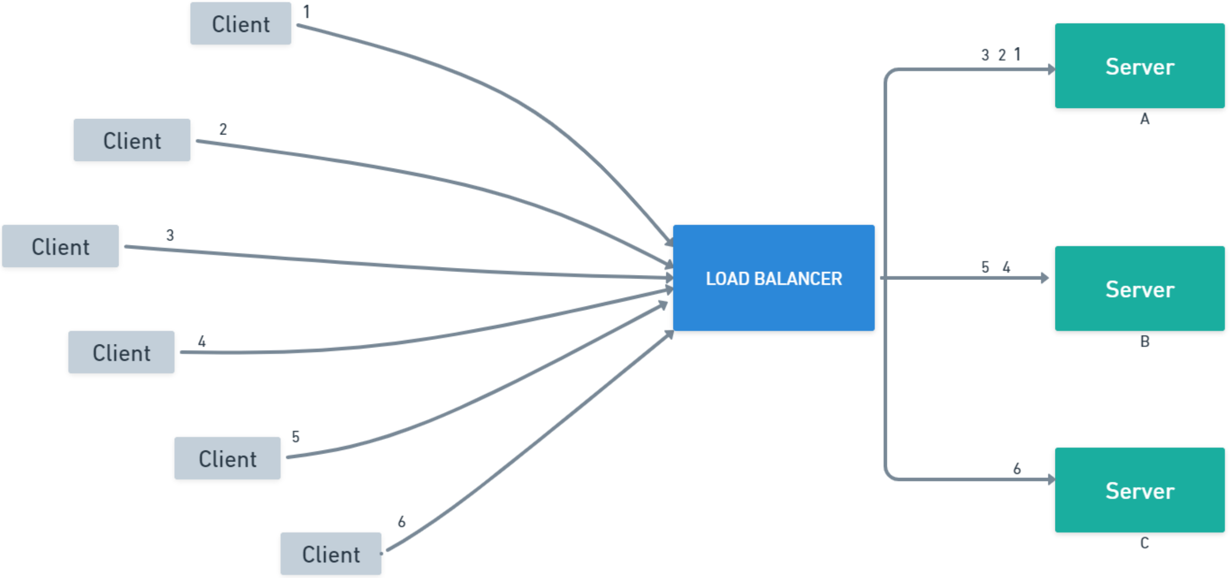
- In round robin ,the requests from the client is distributed in the cyclic manner, it mean, we will see with the help of diagram
- From the fig, let's say there are three servers A, B, and C and requests from the clients 1,2,3,4,5,6 come to load balancer. The load balancer will forward the request from client's 1 to server A, client's 2 requests to server B, client's 3 requests to server C. Now, for client's 4 requests, the load balancer will forward back to server A, client's 5 requests to server B and the same cycle repeats.
- Loads are evenly distributed which increases the responsiveness of the servers.
- But what will happen, if the server B will have higher RAM, CPU and other specs than the server A and C. In that case ,server A and C may get overloaded and fail quickly, while server B sit idle.
- This method can be preferred where the servers configuration are same.



Load Balancing Methods

Weighted Round Robin

- .Dealing with different configurations of the servers, the administrator can assign the weight or ratio to the server, depending on the request it can handle. Let say, server A can take 3 requests per second, server B can take 2 requests per second on an average, and server C can take 1 requests per second.
- So the load balancer will assign a weight to the server A=3,B=2,C=1.You can see the diagram below
- Now, if the request comes from the clients, the load balancer will forward the first three request to server A, then client's 4 and 5 request to server B and client's 6 request to server C.After this, if there will be 7,8,9 requests would be there, the same cycle will be repeated like round-robin.



Load Balancing Methods

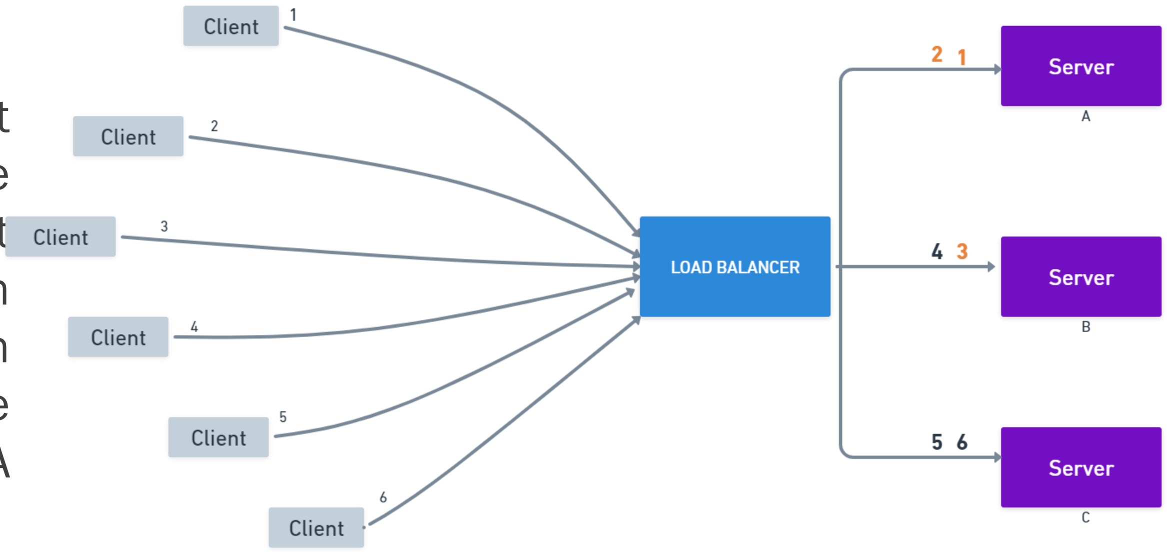
Least Connections

- In round-robin and the weighted round-robin, we can see that the load balancer is not taking consideration of the current load connections on each server.

- Let say we are taking first case where all the servers have same configuration. See the diagam,

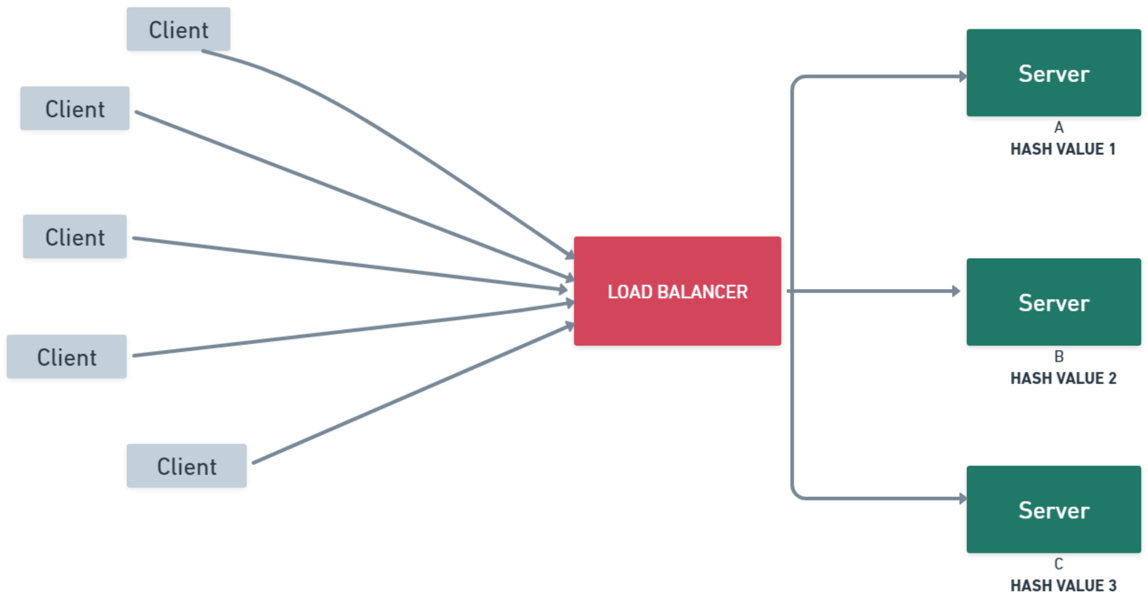
- On server A the client's 1 and 2 requests are not disconnected yet

- On server B client's 3 request are not disconnected. But the client's 4,5 & 6 requests are already disconnected. Now if the new request comes in so according to the round-robin algorithm, it will be forwarded to server A, then server B, and then server C. Now from here, we can see loads on server A to pile up and server A resources may be exhausted quickly.



Load Balancing Methods

- Hashing algorithms are used in the case of persistant connections(means stick a client to the specific server).This may be due to wide range content is serving to the clients like videos. Cache to be served,this reduces the response latency, better cpu utilization.
- Different hashing methods can be used like:
  - URL Hash method
  - Souch IP Hash method



Deep Learning Based Load Balancing

- DLB is to replace hash functions in the load balancing mechanisms with deep learning models, which are trained to be able to map different distributions of workloads and data to the servers in a uniform manner.
- Tim Kraska and et al. [1] introduced the hash model index, which reduces the total number of hash conflicts over map data set by learning a CDF(Cumulative distribution function) at a reasonable cost. However, there are still remaining challenges to leverage deep learning models to improve the effectiveness of load balancing mechanisms.
- On the one hand, how to design a neural network that can converge quickly during the training while also being able to effectively mapping large volumes of inputs to a uniformly distributed space.
- On the other hand, how to balance between the complexity and the expressiveness of the model.

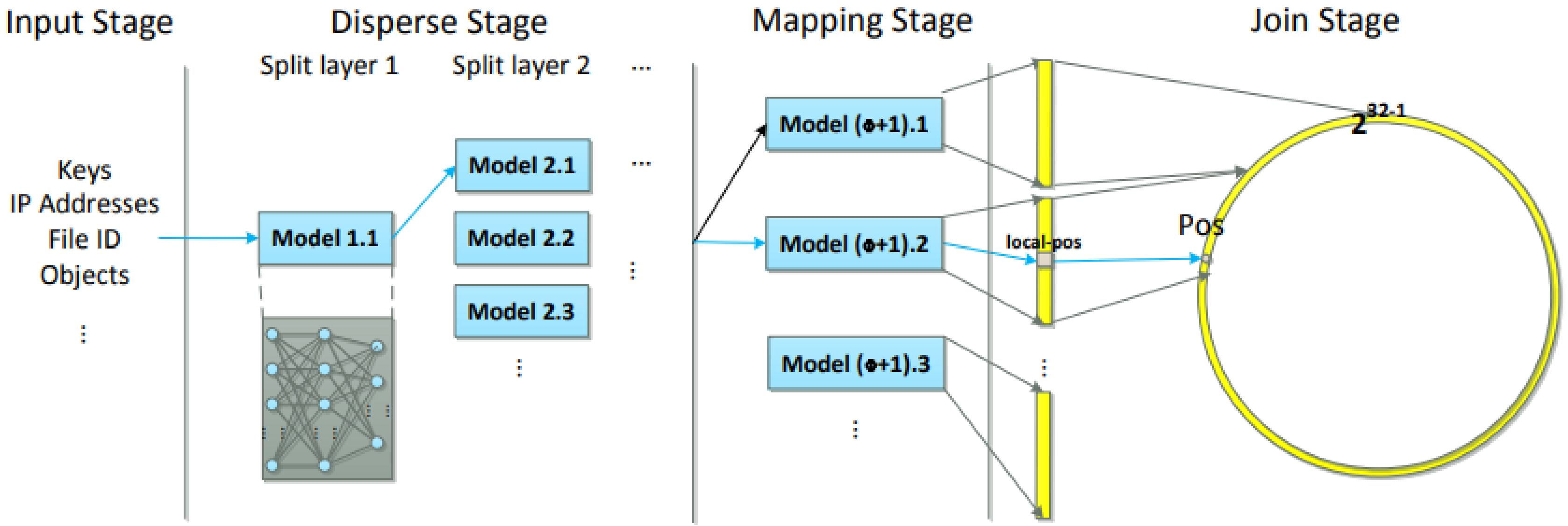
Source: https://arxiv.org/pdf/1910.08494.pdf

[1] Tim Kraska et al. The case for learned index structures. In Proceedings of the 2018 International Conference on Management of Data, SIGMOD Conference 2018, Houston, TX, USA, June 10-15, 2018, 2018.

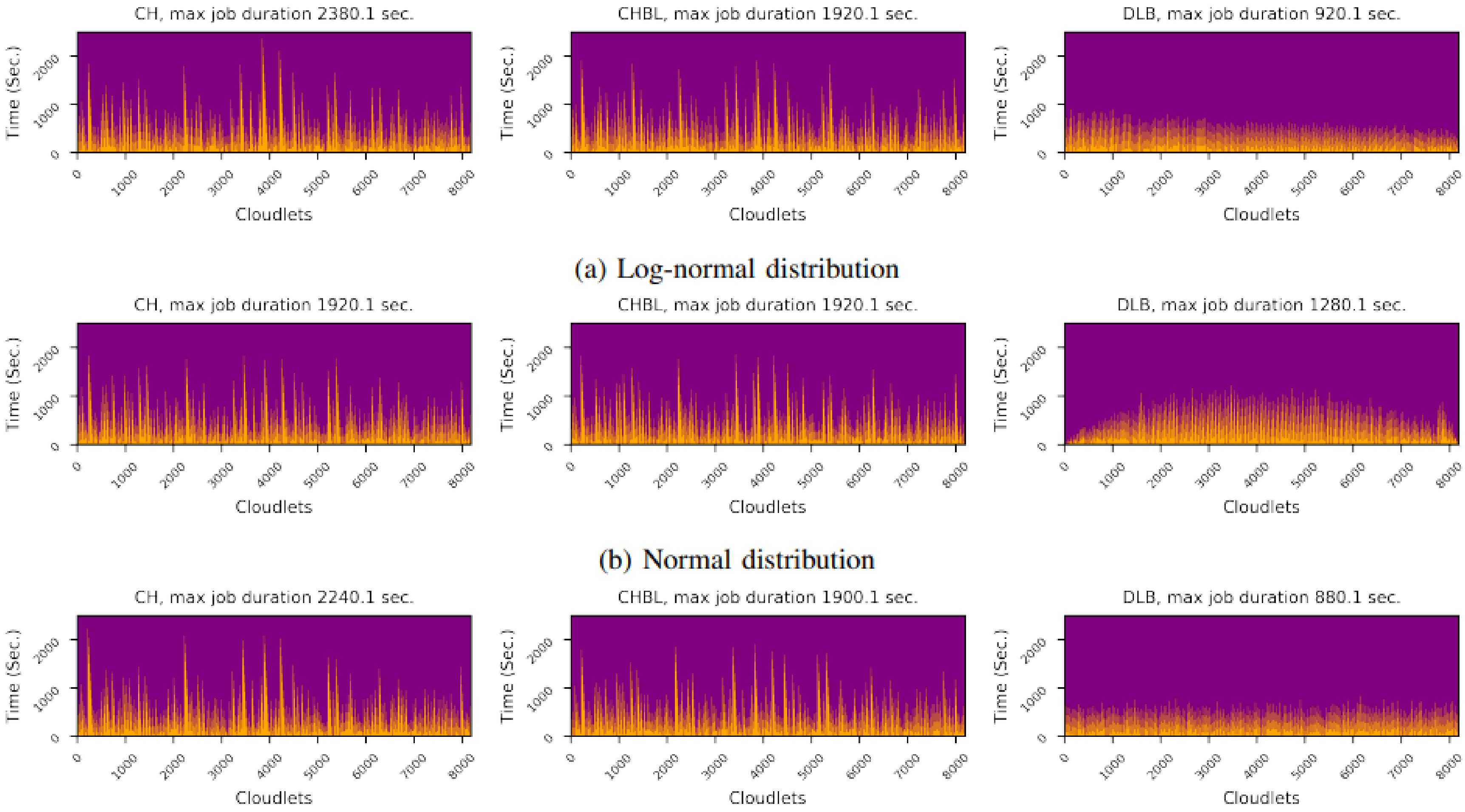
Deep Learning Based Load Balancing

- In order to solve these challenges, DLB is designed in a way that, instead of using a single end-to-end model, it organizes a set of models into a hierarchical architecture. In such an architecture, the models are organized in different connected layers.

- a deep learning based load balancing mechanism which solves the data skew problem by replacing the hash function with deep learning models.



Deep Learning Based Load Balancing



CH, max job duration 2380.1 sec.

CHBL, max job duration 1920.1 sec.

DLB, max job duration 920.1 sec.

(a) Log-normal distribution

CH, max job duration 1920.1 sec.

CHBL, max job duration 1920.1 sec.

DLB, max job duration 1280.1 sec.

(b) Normal distribution

CH, max job duration 2240.1 sec.

CHBL, max job duration 1900.1 sec.

DLB, max job duration 880.1 sec.

(c) Uniform distribution

Fig. 3: Compare the effectiveness of different load balancing mechanisms in a practical Cloud environment created by CloudSim.

Thank you