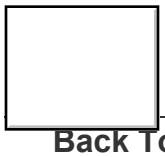


Back To Course Home

Grokking Modern System Design Interview for Engineers & Managers

0% completed

System Design Interviews

Introduction

Abstractions

Non-functional System Characteristics

Back-of-the-envelope Calculations

Building Blocks

Domain Name System

Load Balancers

Introduction to Load Balancers

Global and Local Load Balancing

Advanced Details of Load Balancers

Databases

- [ ]

Key-value Store

- [ ]

Content Delivery Network (CDN)

- [ ]

Sequencer

- [ ]

Distributed Monitoring

- [ ]

Monitor Server-side Errors

- [ ]

Monitor Client-side Errors

- [ ]

Distributed Cache

- [ ]

Distributed Messaging Queue

- [ ]

Pub-sub

- [ ]

Rate Limiter

☐

Blob Store

☐

Distributed Search

☐

Distributed Logging

☐

Distributed Task Scheduler

☐

Sharded Counters

☐

Concluding the Building Blocks Discussion

☐

Design YouTube

☐

Design Quora

☐

Design Google Maps

☐

Design a Proximity Service / Yelp

☐

Design Uber

☐

Design Twitter

☐

Mark Course as Completed

Introduction to Load Balancers

Learn about the basics of load balancers and the services offered by them.

We'll cover the following

What is load balancing?#

Millions of requests could arrive per second in a typical data center. To serve these requests, thousands (or a hundred thousand) servers work together to share the load of incoming requests.

> **Note:** Here, it's important that we consider how the incoming requests will be divided among all the available servers.

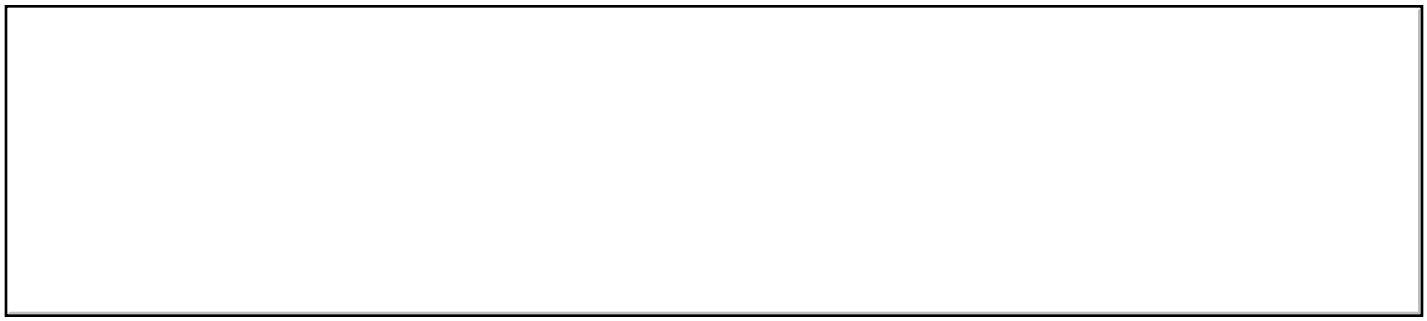
A **load balancer** (**LB**) is the answer to the question. The job of the load balancer is to fairly divide all clients' requests among the pool of available servers. Load balancers perform this job to avoid overloading or crashing servers.

The load balancing layer is the first point of contact within a data center after the firewall. A load balancer may not be required if a service entertains a few hundred or even a few thousand requests per second. However, for increasing client requests, load balancers provide the following capabilities:

- **Scalability**: By adding servers, the capacity of the application/service can be increased seamlessly. Load balancers make such upscaling or downscaling transparent to the end users.
- **Availability**: Even if some servers go down or suffer a fault, the system still remains available. One of the jobs of the load balancers is to hide faults and failures of servers.
- **Performance**: Load balancers can forward requests to servers with a lesser load so the user can get a quicker response time. This not only improves performance but

also improves resource utilization.

Here’s an abstract depiction of how load balancers work:



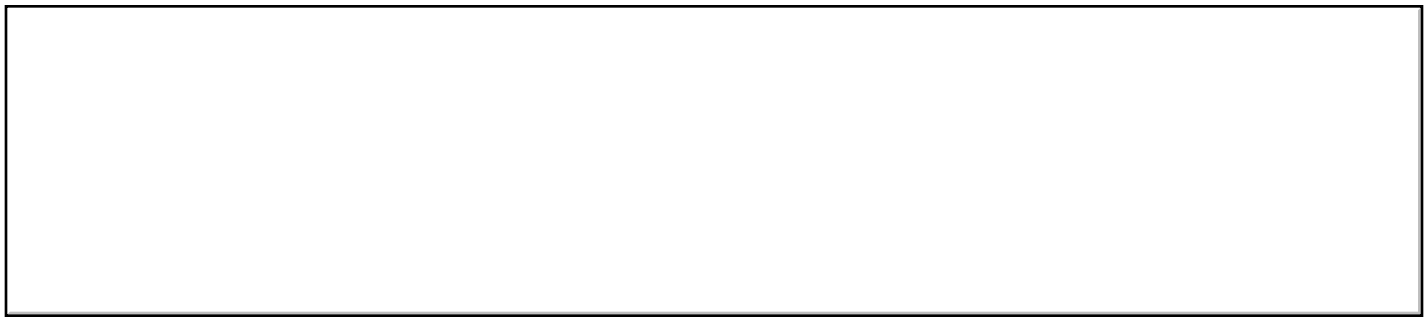
Simplified working of a load balancer

Placing load balancers#

Generally, LBs sit between clients and servers. Requests go through to servers and back to clients via the load balancing layer. However, that isn’t the only point where load balancers are used.

Let’s consider the three well-known groups of servers. That is the web, the application, and the database servers. To divide the traffic load among the available servers, load balancers can be used between the server instances of these three services in the following way:

- Place LBs between end users of the application and web servers/application gateway.
- Place LBs between the web servers and application servers that run the business/application logic.
- Place LBs between the application servers and database servers.



Possible usage of load balancers in a three-tier architecture

In reality, load balancers can be potentially used between any two services with multiple instances within the design of a system.

Services offered by load balancers#

LBs not only enable services to be scalable, available, and highly performant, they offer some key services like the following:

- **Health checking**: LBs use the heartbeat protocol to monitor the health and, therefore, reliability of end-servers. Another advantage of health checking is the improved user experience.
- **TLS termination**: LBs reduce the burden on end-servers by handling TLS termination with the client.
- **Predictive analytics**: LBs can predict traffic patterns through analytics performed over traffic passing through them or using statistics of traffic obtained over time.
- **Reduced human intervention**: Because of LB automation, reduced system administration efforts are required in handling failures.
- **Service discovery**: An advantage of LBs is that the clients' requests are forwarded to appropriate hosting servers by inquiring about the service registry.
- **Security**: LBs may also improve security by mitigating attacks like denial-of-service (DoS) at different layers of the OSI model (layers 3, 4, and 7).

As a whole, load balancers provide flexibility, reliability, redundancy, and efficiency to the overall design of the system.

Food for thought

Question

What if load balancers fail? Are they not a single point of failure (SPOF)?

Show Answer

In the coming lessons, we'll see how load balancers can be used in complex applications and which type of load balancer is appropriate for which use case.

How the Domain Name System Works

Next

Global and Local Load Balancing

Mark as Completed

Report an Issue