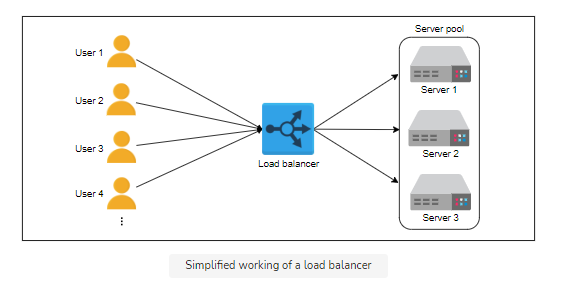
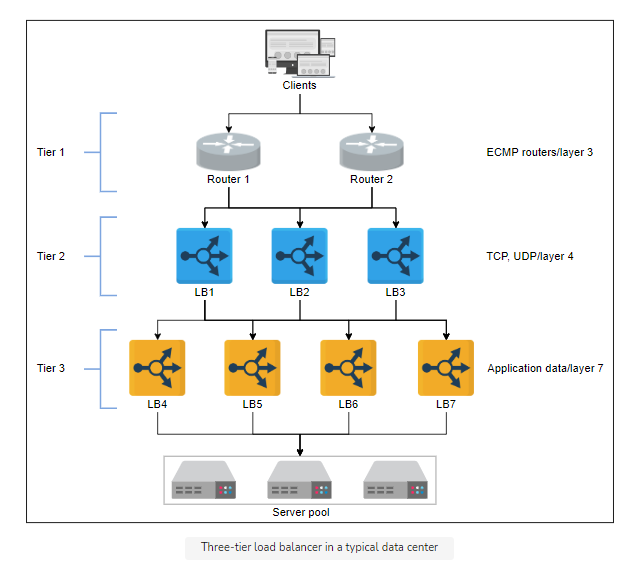
**Load Balancers**

* 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.
* Load Balancers offers:
  + Scalability
  + Availability
  + Performance



* Load Balancers are placed between:
  1. End Users and Web Servers/ Application gateways
  2. Web servers and Application servers
  3. Application servers and Database servers
* Services offered by load balancers:
  1. Health checking
  2. Predictive analytics
  3. Security
  4. Service Discovery
  5. Reduced human Intervention
* Global server load balancing (GSLB):
  1. GSLB involves the distribution of traffic load across multiple geographical regions.
  2. GSLB service can be installed on-premises or obtained through Load Balancing as a Service (LBaaS).
* Local load balancing:
  1. Local load balancers reside within a data center. They behave like a reverse proxy and make their best effort to divide incoming requests among the pool of available servers.
* Algorithms of load balancer:
  1. Round-robin scheduling
  2. Weighted round-robin
  3. Least connections
  4. Least response time
  5. IP Hash
  6. URL Hash
* Static Vs Dynamic Algorithms
  + Static algorithms don’t consider the changing state of the servers.
  + Dynamic algorithms are algorithms that consider the current or recent state of the servers. Dynamic algorithms maintain state by communicating with the server, which adds a communication overhead.
  + Dynamic algorithms require different load balancing servers to communicate with each other to exchange information.
* Stateful Vs Stateless LBs
  + A state is maintained to hold session information of different clients with hosting servers.
  + **Stateful load balancing**: As the name indicates, stateful load balancing involves maintaining a state of the sessions established between clients and hosting servers. The stateful LB incorporates state information in its algorithm to perform load balancing.
  + **Stateless load balancing**: Stateless load balancing maintains no state and is, therefore, faster and lightweight. Stateless LBs use consistent hashing to make forwarding decisions.
  + Therefore, a state maintained across different load balancers is considered as stateful load balancing. Whereas a state maintained within a load balancer for internal use is assumed as stateless load balancing.
* Types of load balancers
  + Depending on the requirements, load balancing can be performed at the network/transport and application layer of the open systems interconnection (OSI) layers.
  + **Layer 4 load balancers**: Layer 4 refers to the load balancing performed on the basis of transport protocols like TCP and UDP.
  + **Layer 7 load balancers:** Layer 7 load balancers are based on the data of application layer protocols. It’s possible to make application-aware forwarding decisions based on HTTP headers, URLs, cookies, and other application-specific data—for example, user ID.
* Load balancer deployment



1. Tier-0 and tier-1 LBs:
   * If DNS can be considered as the tier-0 load balancer, equal cost multipath (ECMP) routers are the tier-1 LBs.
   * Tier-1 LBs will balance the load across different paths to higher tiers of load balancers.
2. Tier-2 LBs:
   * The second tier of LBs include layer 4 load balancers.
   * Tier-2 LBs make sure that for any connection, all incoming packets are forwarded to the same tier-3 LBs. To achieve this goal, a technique like consistent hashing can be utilized.
3. Tier-3 LBs:
   * Layer 7 LBs provide services at tier 3.
   * Since these LBs are in direct contact with the back-end servers, they perform health monitoring of servers at HTTP level.

To summarize, tier 1 balances the load among the load balancers themselves. Tier 2 enables a smooth transition from tier 1 to tier 3 in case of failures, whereas tier 3 does the actual load balancing between back-end servers. Each tier performs other tasks to reduce the burden on end-servers.