INTRODUCTION TO LOAD BALANCING

Load balancing is used everywhere in networking. Are you logging into a Windows network? Windows Active Directory uses load balancing to sign you in through its next available domain controller. Are you shopping on Amazon? Amazon and other popular sites balance the load of incoming traffic through their server farms.

Your organization is probably using some form of load balancing for functions such as VPN, app servers, databases, and other resources. Load balancing is prebuilt into many popular software packages.

Load balancing is so pervasive that unless you are actively working on it, you may not even realize it is there. This articles introduces load balancing and how it works. We will look at:

- The concept
- Benefits
- How load balancing works
- Common algorithms
- Other load balancing scenarios

What is load balancing?

Load balancing distributes incoming network traffic among multiple servers and resources. It helps ensure that no single server becomes overworked. It also prevents:

- Slowdowns
- Dropped requests

- Server crashes

Instead, when a server is unable to handle incoming requests, a load balancing server will direct incoming traffic to another available server.

Load balancers sit in front of your network servers. Your network must contain one or more redundant servers or resources that it balances incoming traffic for. A load balancer receives incoming requests from endpoint devices (laptops, desktops, cell phones, IoT devices, etc.) and uses algorithms to route each request to one or more servers in its server group.

When a server goes down, the load balancer redirects the traffic to the remaining servers in the group. When a server is added to the group, the load balancer will start sending traffic to that server as part of its balancing algorithm.

Load balancing can be performed:

- By physical servers: hardware load balancers
- By virtualized servers: software load balancers
- As a cloud service: Load Balancer as a Service (LBaaS), such as AWS Elastic

Load balancing can be performed by dedicated load balancers or in an Application Delivery Controller (ADC) with load balancing capabilities.

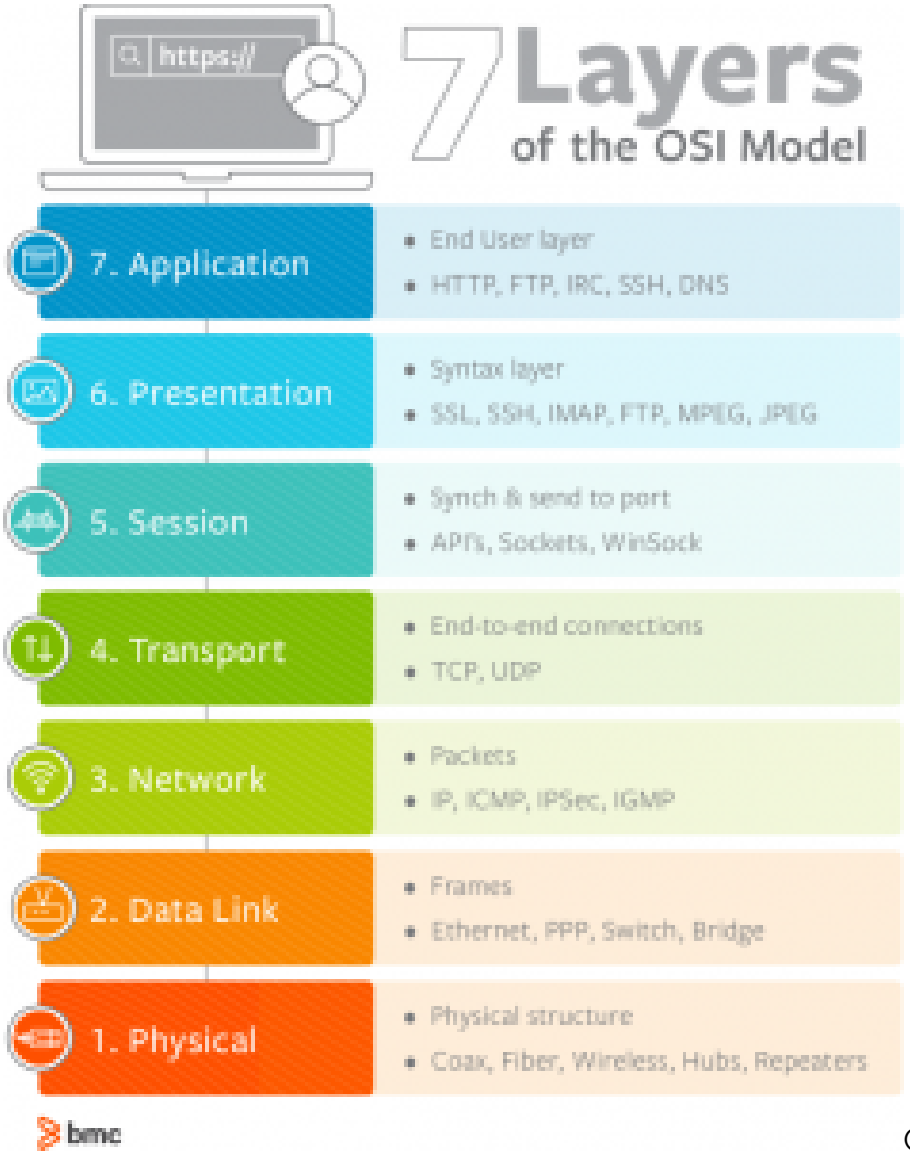
Load balancers can reside on premise, in a regional or global data center, or in the cloud, making it easy to set up load balancing services residing anywhere in the world.

How load balancing works

Load balancers route traffic at Layer 4 or Layer 7 of the Open Systems Interconnection (OSI) model. They advertise their address as the destination IP address for a service or Web site. Balancers receive incoming client requests and select servers to process each request:

- **Layer 4 (L4 OSI Transport layer) balancers do not inspect the contents of each packet.** They make routing decisions based on the port and IP addresses of the incoming packets and use Network Address Translation (NAT) to route requests and responses between the selected server and the client.
- **Layer 7 (L7 OSI Application layer) balancers route traffic at the application level.** They inspect incoming content on a package-by-package basis. L7 balancers route client requests to selected servers using different factors than an L4 balancer, such as HTTP headers, SSL session IDs, and types of content (text, graphics, video, etc.).

L7 balancers use more computational power than an L4 server. They can be more efficient because they route based on context-based factors.



Global server load balancing (GSLB) is also available. GSLBs can route traffic between geographically dispersed servers located in on premise data centers, in the public cloud, or in private clouds. GSLBs are generally configured to send client requests to the closest geographic server or to servers that have the shortest response time.

Benefits of load balancing

Load balancing offers many advantages, including:

- **Efficiency.** Load balancers distribute requests across the WAN and the internet, preventing server overload. They also increase response time by using multiple servers to process many requests at the same time.
- **Flexibility.** Servers can be added and removed from server groups as needed. Individual servers can be brought down for maintenance or upgrade without affecting processing.
- **High availability.** Load balancers only send traffic to servers that are currently online. If one server fails, others are still available to handle requests. Large commercial Web sites such as Amazon, Google, and Facebook deploy thousands of load balancing and associated app servers worldwide. Smaller organizations may also employ load balancers to route traffic to redundant servers.
- **Redundancy.** Multiple servers ensure that processing will continue, even when a server failure

occurs.
- **Scalability.** When traffic increases, new servers can be automatically added to a server group without bringing down services. When high-volume traffic events end, servers can be removed from the group without disrupting service.

GSLB provides several additional benefits over traditional load balancing setups, including:

- **Disaster recovery.** If a local data center outage occurs, other load balancers in different centers around the world can pick up the traffic
- **Compliance.** Load balancer settings can be configured to conform to local regulatory requirements
- **Performance.** Closest server routing can reduce network latency.

Common load balancing algorithms

Load balancers use algorithms to determine where to route client requests. Some of the more common load balancing algorithms include:

- **Least Connection Method.** Clients are routed to servers with the least number of active connections.
- **Least Bandwidth Method.** Clients are routed to servers based on which server is servicing the least amount of traffic, measured in bandwidth.
- **Least Response Time.** Server routing occurs based on the shortest response time generated for each server. Least response time is sometimes used with the least connection method to create a two-tiered method of load balancing.
- **Hashing methods.** Linking specific clients to specific servers based on information in client network packets, such as the user's IP address or another identification method.
- **Round Robin.** Clients are connected to servers in a server group through a rotation list. The first client goes to server 1, second to server 2, and so on, looping back to server 1 when reaching the end of the list.

Load balancing scenarios

Using the techniques outlined here, load balancing can be applied in many different scenarios. Some of the more common load balancing use cases include:

- **App servicing.** Improving overall on premise, mobile, and web performance.
- **Network load balancing.** Evenly distributing requests to commonly used internal resources that are not cloud based, such as email servers, file servers, video servers, and for business continuity.
- **Network adapters.** Using load balancing techniques to direct traffic to different network adapters servicing the same servers.
- **Database balancing.** Distributing data queries to different servers, increasing reliability, integrity, and response time.

Load balancing is a core networking function that can be used anywhere to uniformly distribute workloads across different computing resources. It is a key component of any network.

Additional resources

For related reading, explore these resources:

- BMC IT Operations Blog
- BMC Mainframe Blog
- Building an IT Network for a Remote Facility
- Introduction to BYON (Bring Your Own Network)
- BMC Solutions for Infrastructure Automation: How They Can Help You