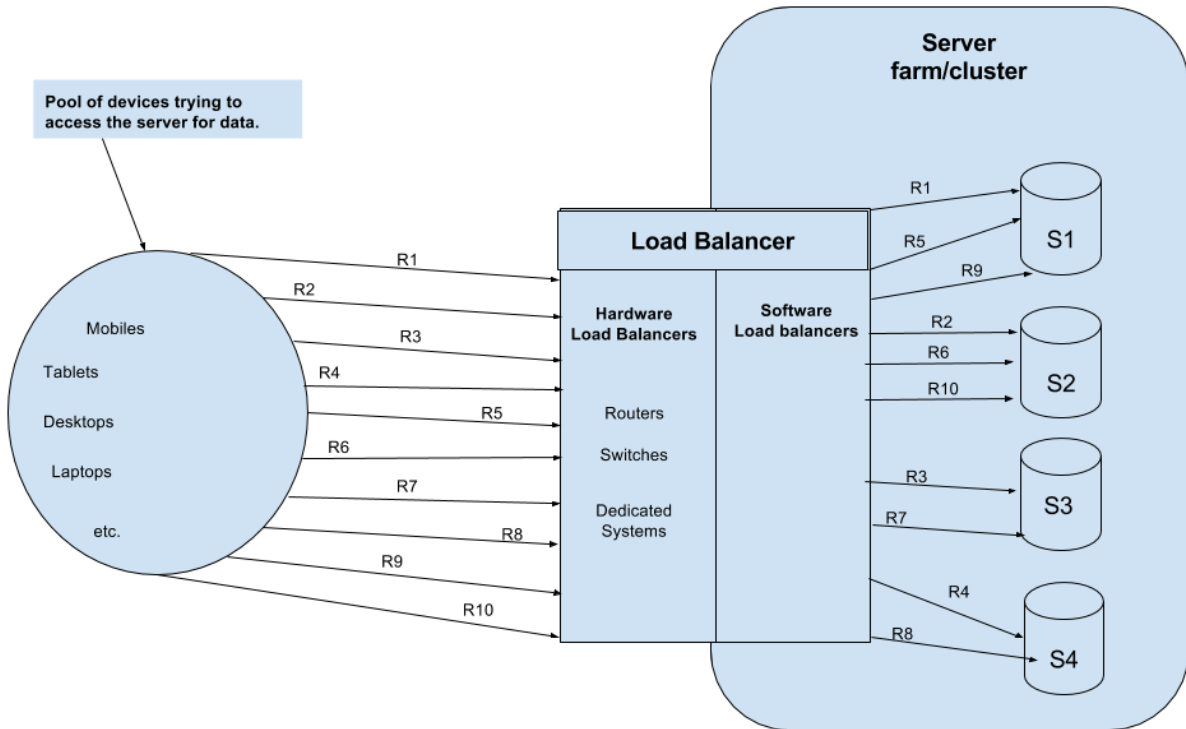


- [Articles](#)
- [People](#)
- [Learning](#)
- [Jobs](#)
- [Games](#)

[Join now](#)[Sign in](#)



Load Balancing, Terminology And Best Practices

[RadhaKrishna Prasad](#)

RadhaKrishna Prasad

PerformanceEngineeringSME | SRE | Corporate Trainer - Performance Engineering |
CloudPerformanceTesting | Chaos and Resilience | Observability | DevOps |

Published Sep 20, 2018

[+ Follow](#)

Introduction:

Load balancing is a key component of highly-available infrastructures commonly used to improve the performance and reliability of web sites, applications, databases and other services by distributing the workload across multiple servers in the infrastructure

Terminology:

- All load-balancing examples will use TCP/IP as the transport protocol. This could easily be any other protocol (e.g., SCTP).
- SLB = Server Load-Balancer
- Virtual Server = Virtual instance of application running on SLB device
- Real Server = physical machine with application instances

Goals Of Load Balancing:

- Achieve optimal resource utilization
- Maximize throughput
- Minimize response time
- Avoid overload
- Avoid crashing

Why To Load Balance?

- To scale the applications and services very easily (High Availability)
- Ease of administration and maintenance. This clearly remove physical servers from rotation in-order to perform any type of maintenance on that server
- Can run multiple instances of application / service on a server. Could be running on a different port for each instance. Can load balance to different port based on data analyzed.

Types Of Load Balancing:

Static Load Balancing

- Balance load prior to the execution.
- Several fundamental flaws with static load balancing
- Very difficult to estimate accurately the execution times of various parts of a program without actually executing the parts.
- Communication delays that vary under different circumstances
- Some problems have an indeterminate number of steps to reach their solution.

Dynamic Load Balancing

- Vary load during the execution of the processes.

- All previous factors are taken into account by making the division of load dependent upon the execution of the parts as they are being executed.
- Does experience an additional overhead during execution, but it is much more effective than static load balancing

Load-Balancing Algorithms:

Most Majorly Used:

Least Connections - server with fewest number of flows gets the new flow request.

Weighted Least Connections - associate a weight / strength for each server and distribute load across server farm based on the weights of all servers in the farm.

Round Robin - round robin thru the servers in server farm.

Weighted Round Robin - Give each server 'weight' number of flows in a row. Weight is set just like it is in weighted least flows.

There are other algorithms that took at or try to predict server load in determining the load of a real server

What Does A Server Load-Balancer Do:

Gets user to needed resource:

- Server must be available
- Users "session" must not be broken

- If user must get to same resource over and over, the SLB device must ensure that happens (ie, session persistence)

In Order to do work, Server Load-Balancer is must:

- Know servers – IP/port, availability
- Understand details of some protocols (e.g., FTP, SIP, etc)

Network Address Translation, NAT:

Packets are rewritten as they pass through Server Load Balancer device

How Server Load-Balancer Devices Make Decisions:

The Server Load-Balancer device can make its own load-balancing decisions based on several factors. Some of these factors can be obtained from the packet headers (i.e., IP address, port numbers etc., Other factors are obtained by looking at the data beyond the network headers. Examples below;

- HTTP Cookies
- HTTP URLs
- SSL Client certificates

The decisions can be based strictly on flow counts or they can be based on knowledge of application.

For some protocols, like FTP, you have to have knowledge of protocol to correctly load-balance (i.e., control and data connection must go to same physical server).

When a New Flow Arrives:

Determine if virtual server exists:

- If so, make sure virtual server has available resources.
- If so, then determine level of service needed by that client to that virtual server.
- If virtual machine is configured with particular type of protocol support of session persistence, then do that work

Pick a real server for that client:

- The determination of real server is based on flow counts and information about the flow.
- In order to do this, the Server Load-Balancer may need to proxy the flow to get all necessary information for determining the real server. This will be based on the services configured for that virtual server.

Server Load-Balancer Architectures:

Centralized:

Server Load-Balancer device sits between the Clients and the Servers being load-balanced. The centralized strategies are only useful for small distributed systems.

Distributed:

Server Load-Balancer device sits off to the side, and only receives the packets it needs to based on flow setup and tear down. It work well in large distributed systems.

Few Best Practices That Will Avoid Performance Issues:

- Use HTTP/2 - is a major update to the older HTTP/1.x protocol
- Monitor the Performance of Your Droplets
- Scale Droplets Horizontally or Vertically
- Choose the Right Load Balancing Algorithm
- **Enable Timed Tasks** - As a minimum ensure **timedtasks.enabled** property in the server.init file is set to true
- **The configuration database** - Use the Web GUI itself to change the configuration. The Web GUI always checks that the database is available before you are permitted to save any changes to the configuration.
- **The list of files to maintain in the database**
- **Custom web content**
- **IP spoofing is required** - We need as many IP as the number of users we are testing for depending on the algorithm of the Load balancer. If Algorithm is using Source IP, then it will route the requests based on Source IP.
- If your sole objective is to test Load Balancer , make sure you run only single script and not multiple scripts, as this will rule out the connection imbalance issues

normally called as network bottlenecks. Not all connection exit gracefully across all layers at TCP Stack

This will make your job extremely easy to identify and say whether loadbalancer is doing its job correctly or not.

Hope this article helps. Thanks for reading the article. Happy Learning:)

[Like](#)

[Comment](#)

[Share](#)

[1288 Comments](#)



[Durga Prasad](#)

Performance test lead@ standard chartered Bank

5y

Good article....

[Like](#)

[Reply](#)

[1 Reaction](#)

RadhaKrishna Prasad

PerformanceEngineeringSME | SRE | Corporate Trainer - Performance Engineering | CloudPerformanceTesting | Chaos and Resilience | Observability | DevOps |

5y

Andrew L. Thank you. Just changed the title now

[Like](#)

[Reply](#)

Andrew L.

Splunk Architect

5y

Can't get past the title, Parctices, is not best spelling.

[Like](#)

[Reply](#)

[2 Reactions](#)

RadhaKrishna Prasad

PerformanceEngineeringSME | SRE | Corporate Trainer - Performance Engineering | CloudPerformanceTesting | Chaos and Resilience | Observability | DevOps |

5y

Thanks **Juan Jimenez**

[Like](#)

[Reply](#)

Kosuri Suryanarayana Raju

Quality Specialist - Performance Engineer

5y

Informative article

[Like](#)

[Reply](#)

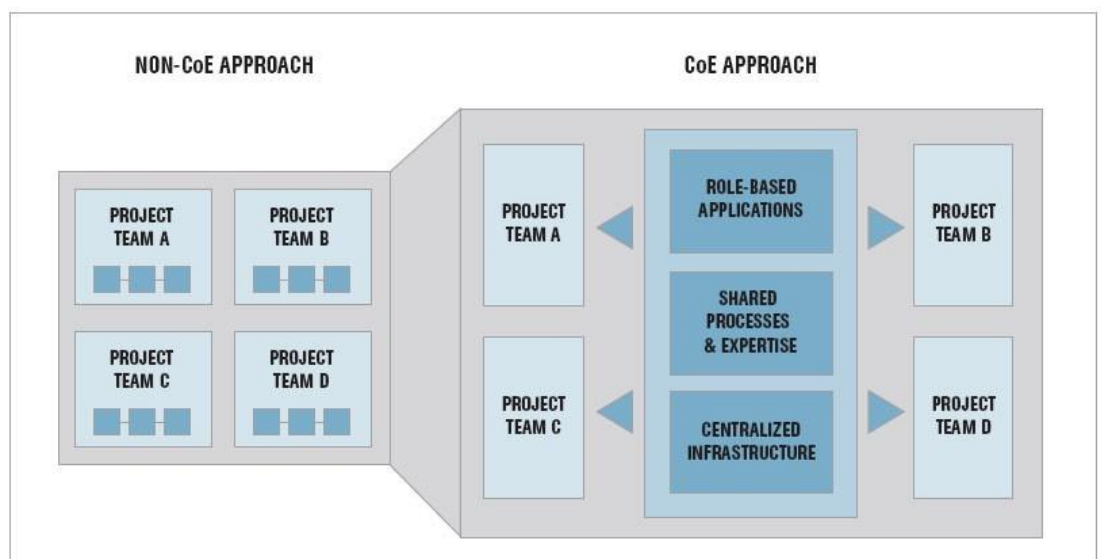
[3 Reactions](#)

[See more comments](#)

To view or add a comment, [sign in](#)

More articles by this author

-

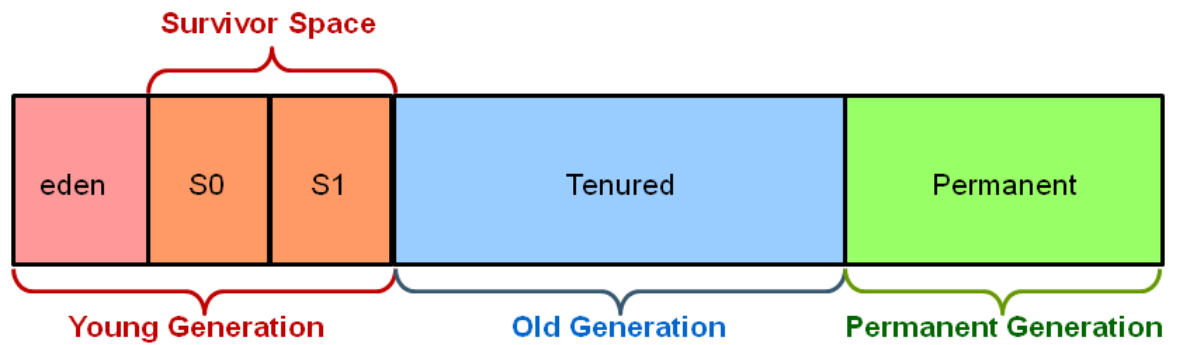


How to Build a Successful Performance CoE for Any Organization

Jun 10, 2020

-

Hotspot Heap Structure



Heap Memory In Java Performance Testing

May 25, 2020

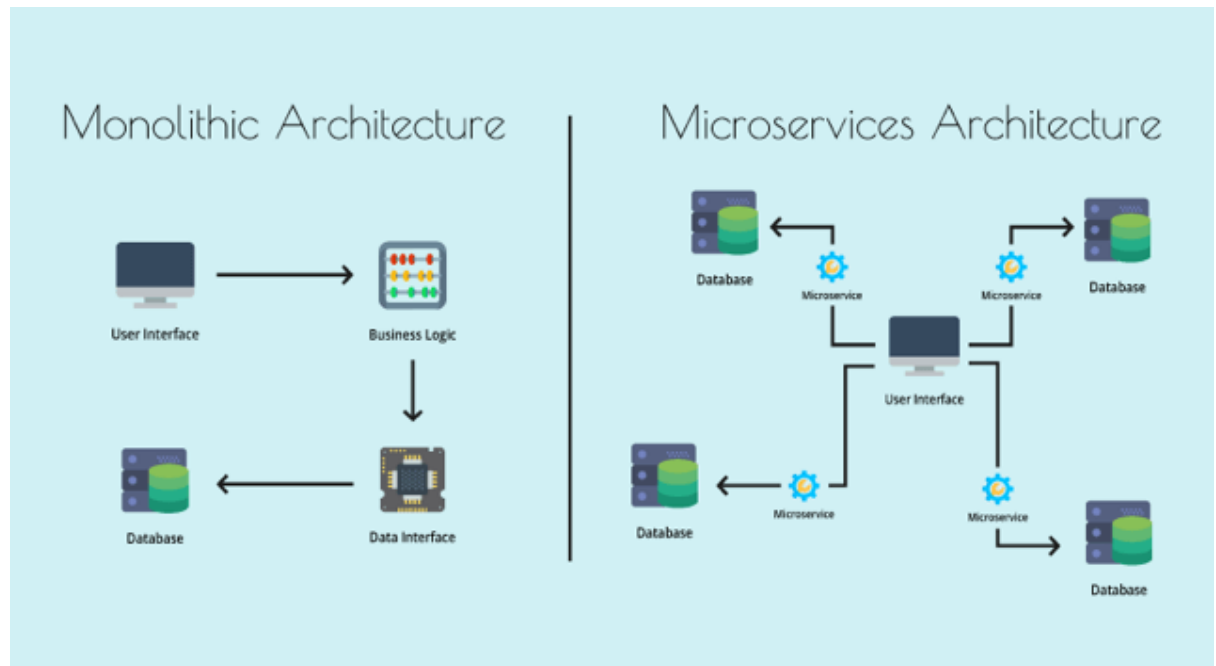
-



Choosing the Best Garbage Collection Algorithm for Better Performance In Java

May 14, 2020

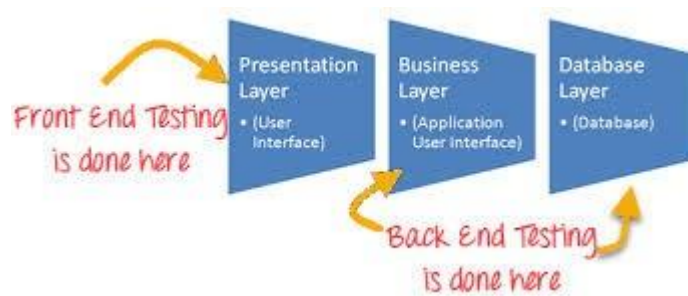
-



Micro-services based Applications - Performance Testing

Feb 27, 2020

-



Client-Side Performance Testing

Nov 28, 2019

-



Skills Required To Be A Perfect Performance Engineer - Performance Testing To Performance Engineering

Jun 19, 2019

-

Fundamentals Of Performance Engineering

Jun 16, 2019

-

WorkLoad Analysis In Performance Testing

Jun 13, 2019

-

Application Scalability

May 21, 2019

-

Common Mistakes In Performance Test Execution & Monitoring

Mar 7, 2019