**Selenium Grid:** (Purpose and main functions)

* Central entry point for all tests.
* Management and control of the nodes / environment where the browsers run.
* Scaling.
* Running tests in parallel.
* Cross platform testing.
* Load balancing.

**Components of Grid:**

**Hub:**

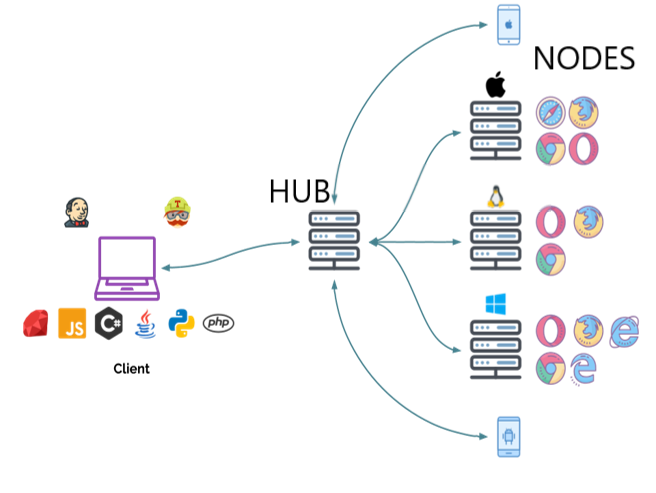
* Intermediary and manager
* Accepts requests to run tests
* Takes instructions from client and executes them remotely on the nodes
* Manages threads

A Hub is a central point where all your tests are sent. Each Selenium Grid consists of exactly one hub. The hub needs to be reachable from the respective clients (i.e. CI server, Developer machine etc.) The hub will connect one or more nodes that tests will be delegated to.

**Nodes:**

* Where the browsers live
* Registers itself to the hub and communicates its capabilities
* Receives requests from the hub and executes them

Nodes are different **Selenium instances** that will **execute tests** on **individual computer systems**. There **can be many nodes in a grid**. The machines which are nodes do not need to be the same platform or have the same browser selection as that of the hub or the other nodes. A node on Windows might have the capability of offering Internet Explorer as a browser option, whereas this wouldn’t be possible on Linux or Mac.



**Main purpose:**

* To run your tests against multiple browsers, multiple versions of browser, and browsers running on different operating systems.
* To reduce the time it takes for the test suite to complete a test pass.

**Setting up the Grid:**

Step 1:

**Start the Hub:** Hub is the central point that will receive test request and distribute the to the right node.

[Distribution is done on Capabilities basis, means a test requiring a set of capabilities will only be distributed to nodes offering that set/subset of capabilities.

* **java -jar selenium-server-standalone.jar -role hub**

[ Hub will listen to port 4444 by default].

Step 2:

**Start the Nodes:**

* **java -jar selenium-server-standalone.jar -role node -hub** [**http://localhost:4444**](http://localhost:4444)**/grid/register**

**Configuration of Node with options:**

Passing JVM properties ( using -D flag before the -jar argument)

-Dwebdriver.chrome.driver=chromedriver.exe.

**Using log file:**

* **java -jar selenium-server-standalone.jar -role hub -log log.txt**

**Using debug argument:**

* **java -jar selenium-server-standalone.jar -role hub -debug**

**@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@**

**Setting up the Hub and node architecture. (working commands)**

**Selenium Grid**

Download Selenium Standalone Server

**To Run the Hub:**

command prompt > **java -jar selenium-server-standalone-3.14.0.jar -role hub**

> then Grid console should up and running

>open browser enter URL : **localhost:4444/grid/console**

Next > to connect Node to Hub we need to provide BrowserWebDriver and port number.

**To Create a Node:**

for chrome browser >

open another command prompt > **java -Dwebdriver.chrome.driver=C:\Users\sball\Downloads\chromedriver\_win32\chromedriver.exe -jar selenium-server-standalone-3.14.0.jar -role node -hub** [**http://localhost:4444/grid/register**](http://localhost:4444/grid/register)

for ie browser >

open another command prompt > **java -Dwebdriver.ie.driver=C:\Users\sball\Downloads\MicrosoftWebDriver.exe -jar selenium-server-standalone-3.14.0.jar -role node -hub** [**http://localhost:4444/grid/register**](http://localhost:4444/grid/register) **-port 4549**

similar commands to other browsers as well

**To create Node: we can use below command**

> **java -jar selenium-server-standalone-3.14.0.jar -role node -hub** [**http://localhost:4444:4444/grid/register**](http://localhost:4444:4444/grid/register)**/ -port 4547**

**For other machines:** we need to change remote url ( http:// different ip address )

**Note : whenever we want to run test cases in Selenium Grid**

we have to set Desired Capabilities

and use RemoteWebDriver

--------------

coding in the test script

>>>to set Desiredcapabilities

**dcap.setPlatform(Platform.WINDOWS);**

>>>to use RemoteWebDriver

**URL url =new URL ( "** [**http://192.168.0.13:4444/wd/hub**](http://192.168.0.13:4444/wd/hub) **" );**

**WebDriver driver=new RemoteWebDriver( remoteaddress , dcap);**

To run test cases on another machine :

> install selenium-standalone-server from Seleniumhq

> install relevant browserDrivers.

> on remote machine to run hub > command prompt

**> java -Dwebdriver.ie.driver=C:\Users\sball\Downloads\microsoftwebdriver.exe -jar selenium-server-standalone-3.14.0.jar -role node -hub** [**http://ipaddress**](http://ipaddress) **of hub :4444/grid/register**

**++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++**

**To find ip address > ipconfig > then IPv4 is ip address.**

**BrowserStack and Saucelab > cloud based company that provides multiple test environments. ( provide virtual machines to tests )**

**Browserstack**

**to use browserstack we need to create an account.**

**> dashboard > docs > browser and platform**

**we need API key > to obtain the key**

we need to change URL in our test script

and set DesiredCapabilities