GRID

Grid when implemented with WebDriver is called as GRID-2; When it was implemented with RC, it was called as GRID-1.

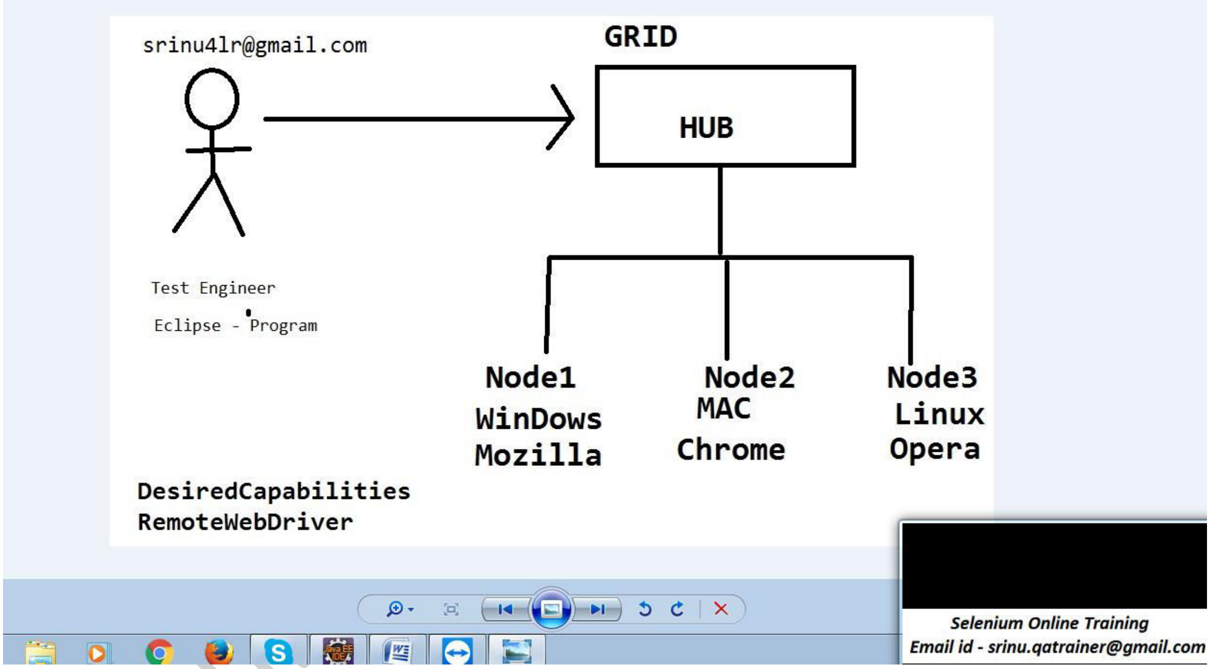
Grid is used to perform distributed Testing.

In Grid we have a Server called as HUB. This HUB is connected to multiple terminals called as NODES.

Each of these NODES can be running on the same OS or different OS.

Similarly they can also be using the same Browser or different Browser.

Using GRID it is possible to perform Cross Browser and Cross Platform Testing, either Serially or Parallely.



Note: Grid is most computable with the TestNG frame work because TestNG Supports both serial and parallel Testing.

Creating a Hub Server

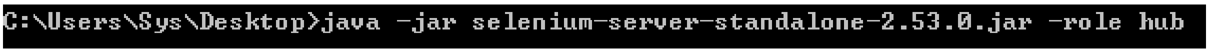
Note: The Hub and the Nodes should mandatorily be running with JDK and they should also have a copy of Selenium Server Standalone jar file.

To create a HUB server goto Command prompt in that machine

Change the directory to the location where a copy of selenium server standalone jar is present.

```
C:\Users\Sys>cd desktop
```

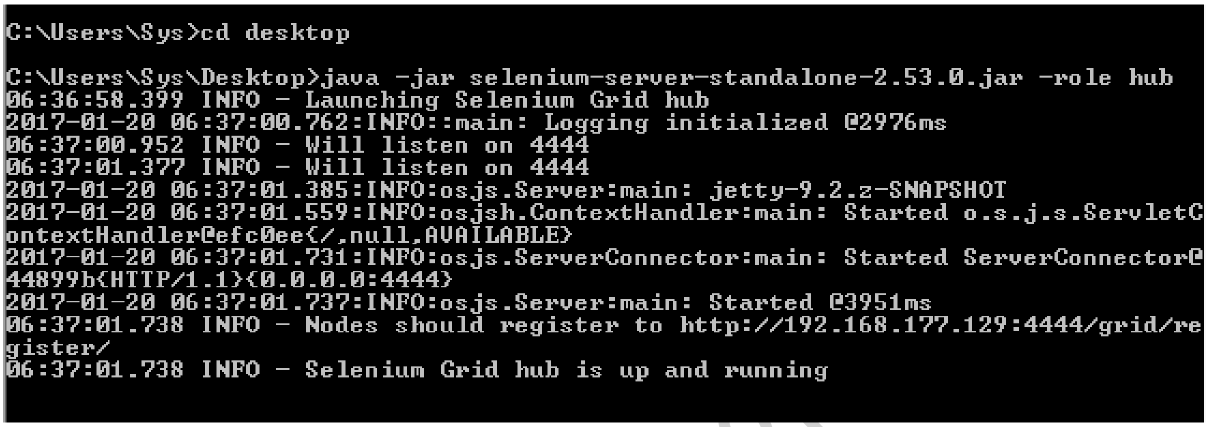
Now trigger the command for creating this machine as HUB server



```
C:\Users\Sys\Desktop>java -jar selenium-server-standalone-2.53.0.jar -role hub
```

Hit enter

We will see the message in command prompt as “ Selenium Grid Hub is up and Running”



```
C:\Users\Sys>cd desktop

C:\Users\Sys\Desktop>java -jar selenium-server-standalone-2.53.0.jar -role hub
06:36:58.399 INFO - Launching Selenium Grid hub
2017-01-20 06:37:00.762:INFO::main: Logging initialized @2976ms
06:37:00.952 INFO - Will listen on 4444
06:37:01.377 INFO - Will listen on 4444
2017-01-20 06:37:01.385:INFO:osjs.Server:main: jetty-9.2.z-SNAPSHOT
2017-01-20 06:37:01.559:INFO:osjsh.ContextHandler:main: Started o.s.j.s.ServletC
ontextHandler@efc0ee{/,null,AVAILABLE}
2017-01-20 06:37:01.731:INFO:osjs.ServerConnector:main: Started ServerConnector@
44899b{HTTP/1.1}{0.0.0.0:4444}
2017-01-20 06:37:01.737:INFO:osjs.Server:main: Started @3951ms
06:37:01.738 INFO - Nodes should register to http://192.168.177.129:4444/grid/re
gister/
06:37:01.738 INFO - Selenium Grid hub is up and running
```

This statement indicates the the Hub server is Ready to use.

The default port not of Hub server is 4444

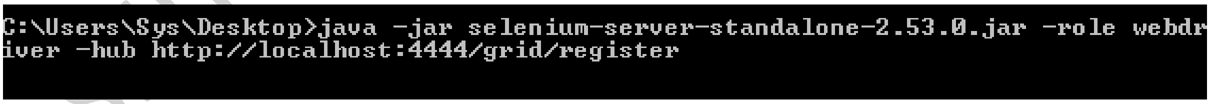
To view the console of the Hub Server -→ open the browser in the HUB machine and navigate to the URL as “IP address of the Hub machine:PortNumber”

Process of creating a NODE

Open the command prompt in the External machine which acts as a NODE

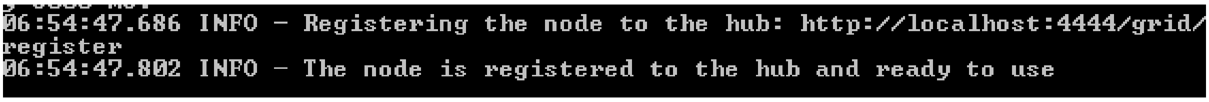
Change the directory to the location where we have selenium server standalone jar file

And now trigger the command as below to convert this machine as NODE and connecting to Hub server



```
C:\Users\Sys\Desktop>java -jar selenium-server-standalone-2.53.0.jar -role webdr
iver -hub http://localhost:4444/grid/register
```

Hit enter and wait for the statement as below

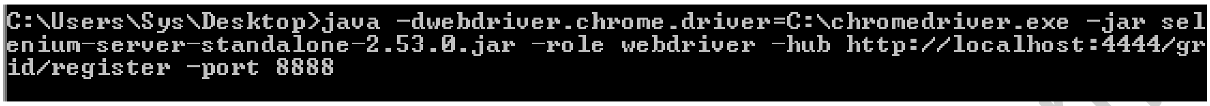


```
06:54:47.686 INFO - Registering the node to the hub: http://localhost:4444/grid/
register
06:54:47.802 INFO - The node is registered to the hub and ready to use
```

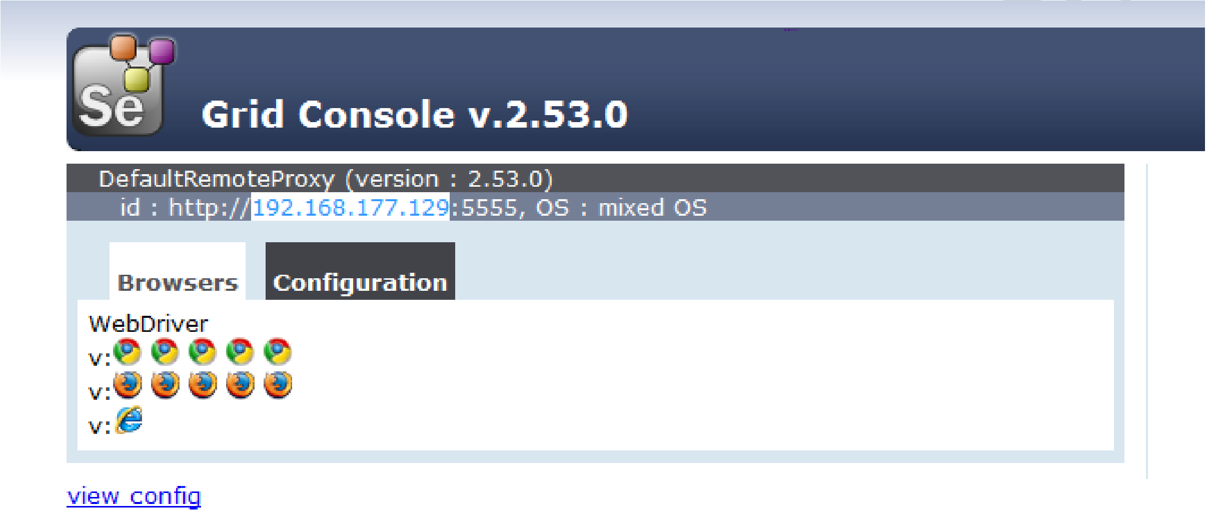
If we have to see whether the Node is registered to the HUB server --- goto HUB server machine – open the console of the GRID HUB server and check whether the node is registered or NOT

The default port number of NODE is 5555

Creating a node with chrome as driver

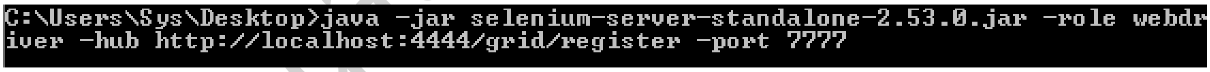


```
C:\Users\Sys\Desktop>java -dwebdriver.chrome.driver=C:\chromedriver.exe -jar sel
enium-server-standalone-2.53.0.jar -role webdriver -hub http://localhost:4444/gr
id/register -port 8888
```

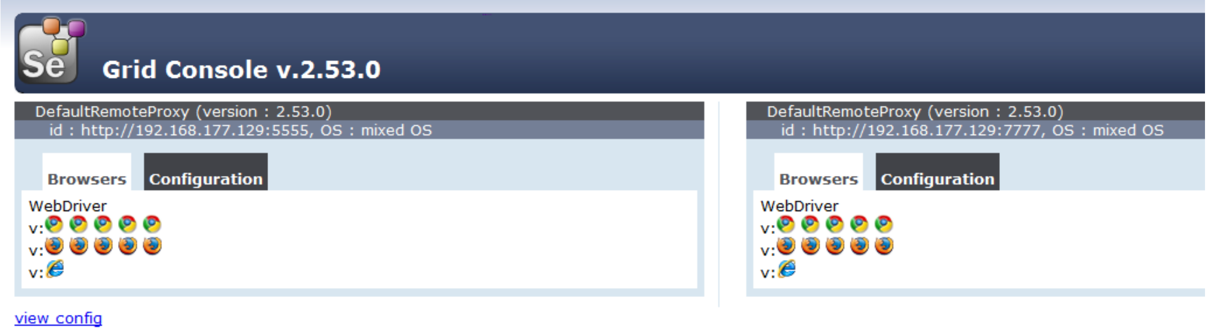


```
DefaultRemoteProxy (version : 2.53.0)
  id : http://192.168.177.129:5555, OS : mixed OS
```

Creating Node with our own port number and connecting to the HUB server



```
C:\Users\Sys\Desktop>java -jar selenium-server-standalone-2.53.0.jar -role webdr
iver -hub http://localhost:4444/grid/register -port 7777
```



```
DefaultRemoteProxy (version : 2.53.0)
  id : http://192.168.177.129:5555, OS : mixed OS
```

```
DefaultRemoteProxy (version : 2.53.0)
  id : http://192.168.177.129:7777, OS : mixed OS
```

Desired Capabilities

This class is used in Selenium for specifying the Browser on which the Automation should be performed. Similarly the version of the Browser that should be used. The OS that should be used.

Remote WebDriver

This class is used in selenium GRID for Perfoming Testing on remote Machine. It accepts 2 parameters

1. IP address of the HUB server
2. Capability object

POM

POM is designing Program / pattern for creating automation Programs

Here the Programs are designed based on the Pages present in the Web Application.

Each page in th Web Application is automated as a sperate Java Class

This class will contain information about the elements present in the Web Page and also the Operations performed on these elements.

These opertions methods are called as Business Logics.

We maintain an Object repository for storing Web element information.

The names of the Java Classes and the names of the corresponding Web Pages should be same.

The names of the the Business Logics are given based on the Functionality they are performing.

- PageFactory actually uses a class in Selenium WebDriver called as the WebFactory class.

  In WebFactory we have an Annotation called as @FindBy ,used to locate an Element.

  Advantages:

  POM Speerates the test case executio flow from the automation programs. This makes the code cleaner and esay to understand.Since the Object repository is maintained independent of the test Case.

  Thse same Object repository can be used for Regression Testing with TestNG.

  It can be used for performing User Acceptance Testing

  Since methods are given realistic names it becomes easy to map the methods with the actions they perform on the Web Application.