Total-QA
Future of Software Testing

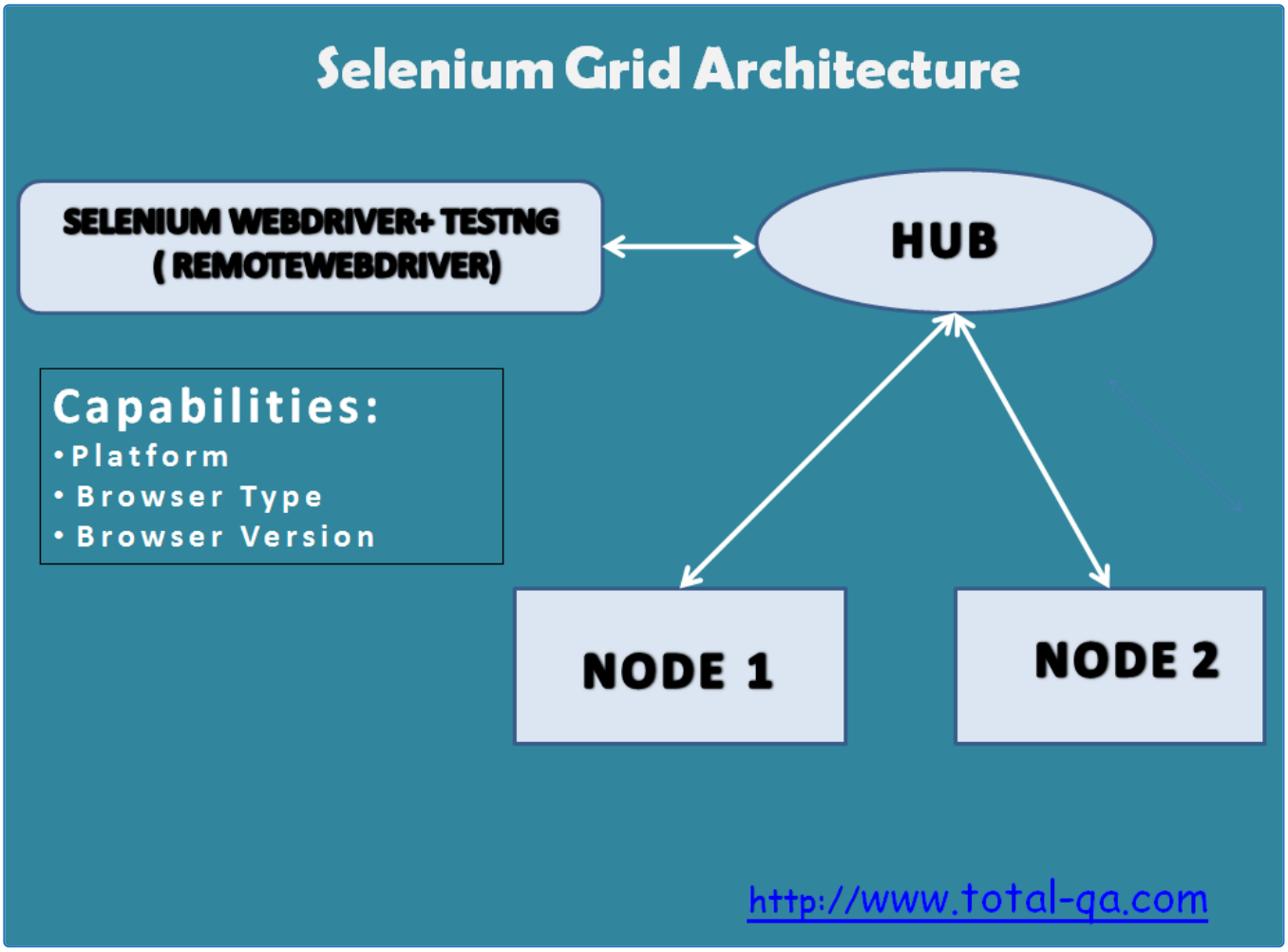
How to use Selenium Grid easily in 3 Steps?

total-qa January 22, 2019 No Comments

Selenium-Grid allows you run your tests on remote desktop against different Operating Systems like Windows,Linux,MAC and against different browsers like Opera,Safari,IE,Firefox and Chrome in parallel. Selenium-Grid allows running your tests in a distributed test execution environment.

Grid Architecture:

In Selenium-Grid, the Component Hub acts as a server receives the requests from Selenium WebDriver program and redirects requests to the Nodes which are configured to Hub. Refer to the details in the diagram mentioned below:



Grid Architecture

Installation:

Ensure the Selenium Standalone Server Jar is downloaded in Hub and Node Systems as well.
Download the Selenium Standalone Server Jar from this link -> http://selenium-release.storage.googleapis.com/index.html

Starting Hub:

The Hub is the central point that will receive all the test request and distribute them the the right nodes.
Open the new command line and execute the below command:

```
java -jar selenium-server-standalone-xxx.jar -role hub -port 4545
```

Starting Node 1:

Use the same http://selenium-release.storage.googleapis.com/index.html jar file to start the nodes and connect to the Hub.

Capabilities for Node 1:

- Platform : WINDOWS
- Browser Name: firefox
- Browser Version:123

Open the new command line and execute the below command:

```
java -Dwebdriver.gecko.driver="C:\\XXXX\\geckodriver-v0.23.0-win64\\geckodriver.exe"
-jar selenium-server-standalone-3.141.59.jar
-role webdriver
-browser "browserName=firefox,version=123,maxInstances=1,platform=WINDOWS"
-hubHost localhost
-hubPort 4444
-port 4546
```

Starting Node 2:

Use the same http://selenium-release.storage.googleapis.com/index.html jar file to start the nodes and connect to the Hub.

Capabilities for Node 2:

- Platform : WINDOWS
- Browser Name: chrome
- Browser Version:40

Open the new command line and execute the below command:

```
java -Dwebdriver.chrome.driver="C:\\XXXX\\chromedriver_win32\\chromedriver.exe"
-jar selenium-server-standalone-3.141.59.jar
-role webdriver
-browser "browserName=chrome,version=40,maxInstances=1,platform=WINDOWS"
-hubHost localhost
-port 4547
```

Selenium WebDriver Testng Program:

```
1  package totalqa;
2  import java.net.MalformedURLException;
3  import java.net.URL;
4  import org.openqa.selenium.Platform;
5  import org.openqa.selenium.remote.DesiredCapabilities;
6  import org.openqa.selenium.remote.RemoteWebDriver;
7  import org.testng.annotations.Test;
8
9  public class GridEx {
10
11
12      @Test
13      public void executeTest() throws MalformedURLException
14      {
15           DesiredCapabilities caps = new DesiredCapabilities();
16           caps.setBrowserName("firefox");
17           caps.setVersion("123");
18           caps.setPlatform(Platform.WINDOWS);
19 //               Comment below lines of code to execute tests on Chrome
20 //        caps.setBrowserName("chrome");
21 //        caps.setVersion("40");
22 //        caps.setPlatform(Platform.WINDOWS);
23           RemoteWebDriver driver = new RemoteWebDriver(new URL("http://localhost:4444/wd/h
24           driver.get("http://www.google.com");
25           System.out.println(driver.getTitle());
26
27      }
28 }
```

Results:

To check the ports which are reserved

To verify that the port number 4444 is in use. Execute the below command in the command line:
netstat -a -n | findStr 4444

Conclusion:

Selenium Grid is useful to run test cases remotely with specific capabilities.

Selenium Hub, Node, Selenium Grid