

11 Selenium Grid

本章大纲

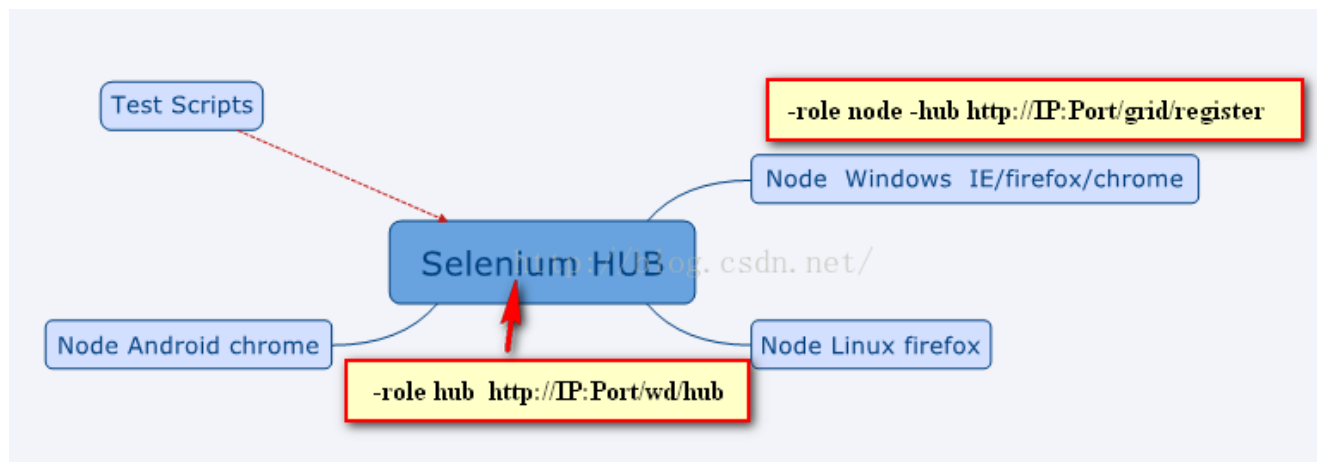
11.1 Selenium Grid简介

11.2 Selenium Grid的使用方法

Selenium Grid简介

- 专门用于**分布式测试**的组件，可以在不同机器上测试不同浏览器
- 使用Hub和Node模式

一台计算机作为Hub（管理中心）管理其他计算机。Hub负责将测试用例分发给多台Node（节点）执行，并收集Node执行结果的报告，汇总后提交一份总的测试报告。



Hub

- ✓ 在分布式测试模式中，只能有一台作为Hub计算机
- ✓ Hub负责管理测试脚本，并负责发送脚本给其他的Node
- ✓ 所有的Node会在Hub计算机中先进行注册，注册成功后再和Hub计算机通信，Node会告之Hub自己的相关信息，例如操作系统和浏览器的相关版本
- ✓ Hub计算机可以给自己分配执行测试用例的任务
- ✓ Hub分发的测试用例会在多个Node中执行。

Node

- ✓在分布式测试模式中，可以有一个或多个Node节点
- ✓Node节点会打开本地的浏览器完成测试任务并返回结果给Hub
- ✓Node的操作系统和浏览器不需要与Hub一致
- ✓在Node可以同时打开多个浏览器并执行测试任务

本章大纲

11.1 Selenium Grid简介

11.2 Selenium Grid的使用方法

Selenium Grid 的使用步骤

1. 找到两台计算机A作为Hub， B作为Node
2. 两台机器访问<http://www.seleniumhq.org/>， 下载 selenium-server-standalone-3.6.0.jar， 并保存在两台计算机的D:/demo目录
3. 在计算机A中打开cmd窗口， 在D:/demo输入语句：
`java -jar selenium-server-standalone-3.6.0.jar -role hub`
使用java命令把jar包文件作为程序， 并且将role参数传递给jar文件的参数， 以此来启动管理中心
4. 在计算机A访问<http://localhost:4444/grid/console>， 出现“view config”链接， 表示hub启动成功

Selenium Grid 的使用方法（续）

5. 在计算机B打开窗口，打开cmd窗口，在D:/demo下输入语句：
`java -jar selenium-server-standalone-3.6.0.jar -role node -browser
browserName=chrome,maxInstances=5 -hub http://(计算机A的IP) 10.7.10.233:4444/grid/register
-port 6655`

role: 参数值webdriver 表示Node节点的名字。

hub: 参数值表示管理中心URL地址， Node会连接这个地址进行注册。

port: 参数值表示Node节点服务的端口是6655， 建议使用大于5000的端口

C:\WINDOWS\system32\cmd.exe - java -jar selenium-server-standalone...

Microsoft Windows [版本 6.3.9600]

(c) 2013 Microsoft Corporation。保留所有权利。

C:\Users\think>D:

D:\>java -jar selenium-server-standalone-2.53.0.jar -role hub

15:21:59.618 INFO - Launching Selenium Grid hub

2016-09-18 15:22:01.740:INFO::main: Logging initialized @3560ms

15:22:01.754 INFO - Will listen on 4444

15:22:01.823 INFO - Will listen on 4444

2016-09-18 15:22:01.827:INFO:osjs.Server:main: jetty-9.2.z-SNAPSHOT

2016-09-18 15:22:01.890:INFO:osjs.ContextHandler:main: Started o.s.j.s.ServletContextHandler@1d371b2d</,null,AVAILABLE>

2016-09-18 15:22:01.990:INFO:osjs.ServerConnector:main: Started ServerConnector@47b120c5<HTTP/1.1><0.0.0.0:4444>

2016-09-18 15:22:01.991:INFO:osjs.Server:main: Started @3812ms

15:22:01.992 INFO - Nodes should register to http://10.7.10.233:4444/grid/register/

15:22:01.993 INFO - Selenium Grid hub is up and running

16:03:01.433 INFO - Registered a node http://10.7.10.212:6655

搜狗浏览器地址: http://www.sogou.com/

Selenium Grid 的使用方法（续）

6. 在计算机A中访问

<http://localhost:4444/grid/console>，验证Node是否注册成功

7. 编写分布式执行的测试脚本



```
//Node节点
```

```
String nodeUrl = "http://10.7.10.212:6655/wd/hub";
```

```
//被测网址
```

```
String baseUrl = "http://10.7.10.190:8032/MyMovie/admin.php/Login/index.html";
```

```
@Test
```

```
public void testLogin() throws MalformedURLException {
```

```
    // 访问远程节点的操作系统和浏览器，需要设定DesiredCapabilities对象
```

```
    DesiredCapabilities capability = DesiredCapabilities.internetExplorer();
```

```
    WebDriver driver = new RemoteWebDriver(new URL(nodeUrl), capability);
```

```
    driver.get(baseUrl);
```

```
    driver.findElement(By.name("username")).sendKeys("admin");
```

```
    driver.findElement(By.name("password")).sendKeys("admin");
```

```
    driver.findElement(By.xpath("//input[@type='submit']")).click();
```

```
    try {
```

```
        Thread.sleep(10000);
```

```
    } catch (InterruptedException e) {
```

```
        // TODO Auto-generated catch block
```

```
        e.printStackTrace();
```

```
    }
```

```
    driver.quit();
```

```
}
```

使用TestNG远程并发自动化测试

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE suite SYSTEM "http://testng.org/testng-1.0.dtd" >
<suite name="Suite1" parallel="methods" thread-count="3">
    <test name="FirefoxTest">
        <classes>
            <class name="com.edu.grid.testGrid" />
        </classes>
    </test>
</suite>
```

怎么用控制多台node呢

- 大家考虑一下怎么控制多台机子的不同浏览器呢？

注意

- Dwebdriver.chrome.driver="D:/demo/chromedriver.exe"
- Dwebdriver.ie.driver="D:/demo/IEDriverServer.exe"
- maxSession 在一台节点计算机，同时允许打开多少浏览器窗口
- browser
"browserName=firefox,maxInstances=3,platform=ANY"
- firefox_binary:设定浏览器firefox启动路径