Atitit WebDriver技术规范原理与概念

[1. Book haosyo ma 1](#_Toc20795)

[2. WebDriver是W3C的一个标准，由Selenium主持。 1](#_Toc18518)

**[3. WebDriver如何工作 (zhuan)](#_Toc18662)** [1](#_Toc18662)

# Book haosyo ma

# WebDriver是W3C的一个标准，由Selenium主持。

W3C的[浏览器测试和工具(Browser Testing and Tools)工作组](http://www.w3.org/testing/browser/" \t "http://www.chinaw3c.org/archives/84/_blank)发布了[WebDriver](http://www.w3.org/TR/2013/WD-webdriver-20130117/" \t "http://www.chinaw3c.org/archives/84/_blank)的工作草案。该规范定义了WebDriver的应用编程接口。WebDriver是一个平台和语言中立的访问接口，允许应用程序或脚本获得Web浏览器的状态并控制浏览器的行为。WebDriver API目前主要用于开发人员编写测试脚本，通过一个独立的控制流程实现基于浏览器的自动测试，未来该规范还可能使运行在浏览器中的脚本程序对浏览器行为进行控制。

该规范还包括一个基于API调用和响应时，基于JSON的输入输出参数的序列化建议，可能对浏览器提供商提供帮助。

作者:: 绰号:老哇的爪子 （ 全名：：Attilax Akbar Al Rapanui 阿提拉克斯 阿克巴 阿尔 拉帕努伊 ）

汉字名：艾提拉（艾龙），   EMAIL:1466519819@qq.com

转载请注明来源： http://blog.csdn.net/attilax

# **[WebDriver如何工作 (zhuan)](http://www.cnblogs.com/backpacker/archive/2012/11/22/2782631.html)**

WebDriver是W3C的一个标准，由Selenium主持。

具体的协议标准可以从http://code.google.com/p/selenium/wiki/JsonWireProtocol#Command\_Reference   查看。

从这个协议中我们可以看到，WebDriver之所以能够实现与浏览器进行交互，是因为浏览器实现了这些协议。这个协议是使用JOSN通过HTTP进行传输。

它的实现使用了经典的Client-Server模式。客户端发送一个requset，服务器端返回一个response。

我们明确几个概念。

Client

调用 WebDriverAPI的机器。

Server

运行浏览器的机器。Firefox浏览器直接实现了WebDriver的通讯协议，而Chrome和IE则是通过ChromeDriver和[InternetExplorerDriver](http://code.google.com/p/selenium/wiki/InternetExplorerDriver)实现的。

Session

服务器端需要维护浏览器的Session，从客户端发过来的请求头中包含了Session信息，服务器端将会执行对应的浏览器页面。

WebElement

这是WebDriverAPI中的对象，代表页面上的一个DOM元素。

举个实际的例子，下面代码的作用是”命令”firefox转跳到google主页：

       WebDriver driver = new FirefoxDriver();  
        //实例化一个Driver  
   
        driver.get("http://www.google.com");

在执行driver.get("http://www.google.com")这句代码时，client，也就是我们的测试代码向remote server发送了如下的请求：

POSTsession/285b12e4-2b8a-4fe6-90e1-c35cba245956/url  post\_data{"url":"http://google.com"}

通过post的方式请求localhost:port/hub/session/session\_id/url地址，请求浏览器完成跳转url的操作。

如果上述请求是可接受的，或者说remote server是实现了这个接口，那么remote server会跳转到该post data包含的url,并返回如下的response

{"name":"get","sessionId":"285b12e4-2b8a-4fe6-90e1-c35cba245956","status":0,"value":""}

该response中包含如下信息

name：remote server端的实现的方法的名称，这里是get，表示跳转到指定url；

sessionId：当前session的id；

status：请求执行的状态码，非0表示未正确执行，这里是0，表示一切ok不许担心；

value：请求的返回值，这里返回值为空，如果client调用title接口，则该值应该是当前页面的title；

如果client发送的请求是定位某个特定的页面元素，则response的返回值可能是这样的：

{"name":"findElement","sessionId":"285b12e4-2b8a-4fe6-90e1-c35cba245956","status":0,"value":{"ELEMENT":"{2192893e-f260-44c4-bdf6-7aad3c919739}"}}

name,sessionId，status跟上面的例子是差不多的，区别是该请求的返回值是ELEMENT:{2192893e-f260-44c4-bdf6-7aad3c919739}，表示定位到元素的id，通过该id，client可以发送如click之类的请求与 server端进行交互。

IE11 开始支持WebDriver标准，IE WebDriver Tool for Internet Explorer 11下载地址：

# WebDriver API 说明

<https://www.w3.org/TR/webdriver/#commands>

[Conformance:](http://dict.youdao.com/search?q=Conformance&keyfrom=chrome.extension&le=eng" \t "https://www.w3.org/TR/webdriver/_blank) 一致性

2. Terminology

3. Interface

4. Protocol

5. Capabilities

6. Sessions

7. Navigation

8. Command Contexts

9. Elements

10. Element Retrieval

11. Element State

12. Element Interaction

13. Document Handling

14. Cookies

15. Actions

16. User Prompts

17. Screen Capture

# 参考资料

WebDriver如何工作 (zhuan) - allback - 博客园.html

WebDriver API 提交 W3C 标准化 · LinuxTOY.html

微软：Win10全新Edge浏览器支持WebDriver测试 - Edge浏览器,Win10系统 - IT之家.html

Atiend