atitit.[Selenium RC 与 Web Driver 的区别](http://www.baidu.com/link?url=OJHRdGeMtnKdEMjI71qHPzPN3XZuuicobF71igeMSOZxcxu1Z3__csIqtcrnzf1QyqBKCy4U_5uF0QsR2XYZ6_" \t "https://www.baidu.com/_blank)

[1. Selenium2= Selenium1（有名Selenium RC）,+webdriver, 1](#_Toc24890)

[1.1. :selenium-java是必须要导入到eclipse中的,selenium-server-standalone是启动seleniumserver用 2](#_Toc20820)

[2. WebDriver 是… …？ 3](#_Toc2680)

[2.1.1. Selenium vs WebDriver 3](#_Toc29478)

[2.2. 定位页面元素 3](#_Toc11128)

[2.3. 从 Selenium RC 迁移到 WebDriver 4](#_Toc24413)

[2.3.1. 步骤 4](#_Toc20812)

 Selenium 2(又名 WebDriver)

# Selenium2= Selenium1（有名Selenium RC）,+webdriver,

我们平时所说的selenium说的都是selenium1（有名Selenium   
RC）,他曾经有个非常强的竞争对手就是Webdriver,后来Selenium团队发布了Selenium2.Selenium2=   
Selenium1+webdriver,这样的组合使Selenium的功能增强了n倍,用户也可以很自由的选择自己喜欢的工具.  
2、 什么是Webdriver?  
WebDriver是一个用来进行复杂重复的web自动化测试的工具.意在提供一种比Selenium1.0更简单易学,有利于维护的API.它没  
有和任何测试框架进行绑定,所以他可以很好的在单元测试和main方法中调用.一旦创建好一个Selenium工程,你马上会发现WebDriver和其  
他类库一样：它是完全独立的,你可以直接使用而不需要考虑其他配置,这个Selenium RC是截然相反的.  
3、 Selenium1.0的缺点?  
Selenium1.0不能处理以下事件：  
1) 本机键盘和鼠标事件  
2) 同源策略XSS/HTTP（S）  
3) 弹出框,对话框（基本身份认证,自签名的证书和文件上传/下载）  
4、 Selenium2有什么好处?  
当这两个框架被合并后,一个框架的缺陷被另一个框架所弥补.WebDriver对浏览器的支持需要对应框架开发工程师做对应的开发；同样  
Selenium必须操作真实浏览器,但是WebDriver可以HTML unit   
Driver来模拟浏览器,在内存中执行用例,更加的轻便.Selenium1.0解决了自动化测试中的一些常见问题,WebDriver更好的解决了沙  
箱限制.WebDriver不支持并行,但是Selenium Grid解决了这个问题.

那么remote server端的这些功能是如何实现的呢？答案是浏览器实现了webdriver的统一接口，这样client就可以通过统一的restful的接口去进行浏览器的自动化操作。目前webdriver支持ie, chrome, firefox, opera等主流浏览器，其主要原因是这些浏览器实现了webdriver约定的各种接口。

## :selenium-java是必须要导入到eclipse中的,selenium-server-standalone是启动seleniumserver用

# WebDriver 是… …？

Selenium 2，又名 WebDriver，它的主要新功能是集成了 Selenium 1.0 以及 WebDriver​（WebDriver 曾经是 Selenium 的竞争对手）。也就是说 Selenium 2 是 Selenium 和 WebDriver 两个项目的合并，即 Selenium 2 兼容 Selenium，它既支持 Selenium API 也支持 WebDriver API。  
那 Selenium 团队为什么会将两个项目合并呢？我们通常认为其中部分原因是 WebDriver 解决了 Selenium 存在的缺点（比如，能够绕过 JS 沙箱），部分原因是 Selenium 解决了 WebDriver 存在的问题（比如，支持更广泛的浏览器和编程语言），不论真正的原因是什么两个项目的合并为用户提供了一个优秀的自动化测试框架。

### Selenium vs WebDriver

从上述用 Selenium 和 WebDriver 构建的两个测试工程来看，WebDriver 工程在构建之后不需要其他的配置我们便可以直接使用，这一点和 Selenium 是截然不同的。因为 Selenium 还需要安装并启动 Selenium Server 才能运行测试程序。

另外，我们可以看出 WebDriver 是基于面向对象的 API，它更多的是从用户角度出发。反之 Selenium 提供的是基于字典的 API，用户可以很方便的看到所以支持的方法。毋庸置疑的是，WebDriver 提供的 API 更为简洁，对用户更加的友好。但从另一个角度来看，就是用户不能很直观的看到 WebDriver 提供了哪些 API，可能需要通过官网提供的 JavaDoc（来源：http://selenium.googlecode.com/svn/trunk/docs/api/java/index.html）的协助来找寻一些方法。

下面我们看下 WebDriver 支持的浏览器类型，以及一些基本操作的使用方法。

## ****定位页面元素****

Selenium 对元素的处理是基于 Action，Object（操作，对象）的方式，使用 id、name，或是 XPath 来定位页面元素。当使用的是 XPath 时在某种程度上对 Internet Explorer 的测试速度有很大的影响，因为 IE 浏览器没有内置对 XPath 的支持，所以在 IE 上测试时会特别慢，当然现在已经有办法解决这个问题了，虽然比不上 Firefox 的测试速度但也是有了相当大的提高。

WebDriver 则使用 findElement 方法来查找到页面元素，它定位页面元素可以通过 id、name、xpath、className、link text、CSS 等等查找。

## 从 Selenium RC 迁移到 WebDriver

WebDriver 有更简洁的 API，它针对各个浏览器而开发，取代了嵌入到被测 Web 应用中的 JavaScript。它与浏览器的紧密集成支持创建更高级的测试，避免了 JavaScript 安全模型导致的限制。除了来自浏览器厂商的支持之外，WebDriver 还可以利用操作系统级的调用模拟用户输入，例如鼠标和键盘操作，这些都是 Selenium 所不能比拟的。

而 Selenium 可以使用任何支持 HTTP 的编程语言，它基于 Javasript 并支持多数浏览器，但它也不是完美的，由于浏览器对于 Javascript 的安全策略的加强导致某些情况下不能使用 Selenium。而且随着 WebDriver 的逐步完善，Selenium 可能将完全不受支持，所以选择 WebDriver 必将成为一个必然趋势。但毋庸置疑的是两个项目的将为用户提供了一个更为通用的强大的 Web 测试框架。

**[回页首](http://www.ibm.com/developerworks/cn/web/1309_fengyq_seleniumvswebdriver/" \l "ibm-pcon)**

### 步骤

Selenium VS Webdriver.html