

Appium梳理

京东商城-平台技术服务部@何小伟

什么项目适合做自动化测试

软件需求变动不频繁

测试脚本的稳定性决定了自动化测试的维护成本。如果软件需求变动过于频繁，测试人员需要根据变动的需求来更新/维护测试用例以及相关的测试脚本。

项目周期较长

如果项目的周期比较短，没有足够的时间去支持这样一个过程，那么自动化测试便成为笑谈。

自动化测试脚本可重复使用

移动应用测试框架

1. Appium(ios/android)
2. Calabash(ios/android)
3. KIF(ios)
4. Robolectric(android)
5. Flip Test(ios)
6. Gitorious(ios)
7. Frank(ios)
8. Kiwi(ios)
9. App Grader(android)
10. Cedar(ios)

如何选择适合自己的框架

1. 它应能同时支持Android/ios
2. 支持尽可能多的编码语言
3. 能与多数主流开发/测试工具集成(eclipse...)
4. 不要求测试人员修改或者重新编译待测应用
5. 应该有一个活跃的社区

为什么是Appium

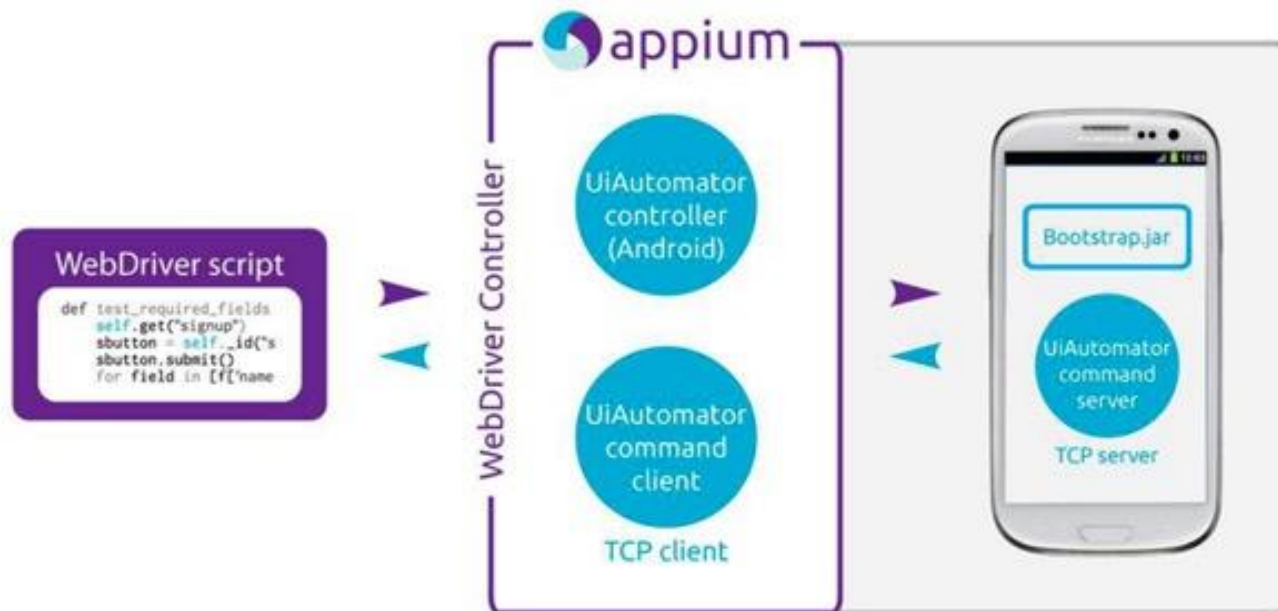
1. 支持更多地编程语言（ PHP, Python, [Ruby](#), C#, Clojure, or Per ）
2. 不要求重新编译应用或者修改，Calabash测试ios应用需要引入第三方包。
3. Appium社区更活跃，可能纳入Selenium_WebDriver
4. 支持iOS，Android，Windows，FirefoxOS

Appium简介

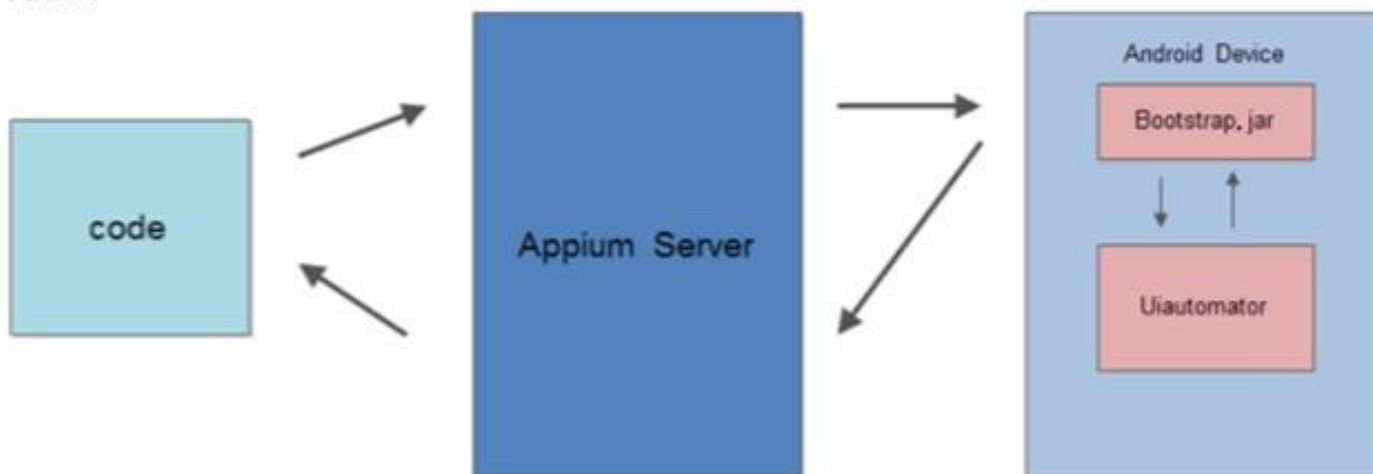
- 由Sauce labs提供的一款开源测试框架，针对原生（Native）,混血（hybrid）和移动Web应用。使用WebDriver协议来驱动IOS和Android应用。
- 理念：测试原生应用不应该要求用户引入额外的SDK或者重新编译应用。
-

Appium Android原理架构

- 在Android端， appium基于WebDriver,并利用Bootstrap.jar,最后通过调用UiAutomator的命令， 实现App的自动化测试。
- UiAutomator测试框架是Android sdk自带的App Ui自动化测试java库。另外由于UiAutomator对WebView的支持有限，Appium引入了Chromedriver来实现基于Webview的自动化。



简易图:



Appium Android的主要原理

- 左边的WebDriver script是我们的测试脚本
- 中间是Appium服务，Appium启动一个服务（4723端口），与selenium-Webdriver测试框架类似。Appium server接收WebDriver标准请求，解析请求内容，调用对应的框架相应操作。
- Appium server会把请求转发给中间件Bootstrap.jar它使用java编写，安装在手机上。Bootstrap接收Appium的命令（4724），最终通过调用UiAutomator的命令来实现。
- 执行结果由Bootstrap返回给我Appium Server.
- Appium还用到了Chromdriver来支持基于Webview的测试

Appium win运行环境准备

- 安装node.js
- Android Sdk （包含tools,platform-tools它添加到环境变量。）
- Java JDK
- 安装Apache Ant 或者使用eclipse/plugins的ant
- Apache Maven.并设置M2HOME和M2环境变量。将M2HOME设置maven安装目录，并设置M2为%M2HOME\bin。并将M2添加到PATH中。
- 安装Git
- 安装cURL