

## 关于 Android 下的自动化测试 (转载)

原文作者: 宋立新 Email: [zjujoe@yahoo.com](mailto:zjujoe@yahoo.com)

### 前言:

现在 Android 开发非常红火, Java 环境下敏捷开发是不二选择。而敏捷开发都是测试驱动。所以,最近研究了一下 Android 下的各种自动化测试手段。本文重点在于面上的比较而非点上的细节。

### 测试手段 1: CTS

CTS 用来确保设备符合 Android 兼容性规范。

CTS 原来只对 OHA 联盟开放。最近 Google 把它 Release 出来了。似乎做过一些裁剪,比如针对 Java 虚拟机的测试,似乎被删除了,但我们一般用不着这么高深的。

针对每个版本,比如 2.1, 2.2, Google 发布了一个兼容性规范,而 CTS 测试就是用来确保设备或模拟器符合该兼容性规范。

CTS 测试基于 Android instrumentation 测试,其又基于 JUnit 测试。说白了,CTS 就是一堆单元测试用例。这也是 Java 语言的擅长部分。

在 2.1 模拟器上试验了一下,有少数没有通过。

目前 CTS 主要包括功能方面的测试,有少数的性能方面的测试。性能测试未来会越来越多。

总的来说,CTS 跟 WM 的 LTK 测试还是弱了一些,毕竟还年轻。它只包括自动化测试,目的主要是保证 API 的兼容性。由于基于单元测试,CTS 本身不能用于测试多应用交互的情况。

对我们的帮助:

- 1) 应用程序的开发者可以开发出自己应用的单元测试,并将其加入 CTS 测试集。
- 2) 设备制造商可以通过周期性运行 CTS 测试,确保没有对 Android 伤筋动骨。

### 测试手段 2: Monkey

猴子测试本身非常简单,就是模拟用户的按键输入,触摸屏输入,手势输入等,看手机多长时间会出异常。执行简单,效果明显。

猴子测试既可以针对全局,也可以正对某个局部(某个 Category, package 等等)。可以设置让 Monkey 只测试某个应用,从而辅助应用程序的开发。

对我们的帮助:

- 1) 应用程序的开发者可以测试自己应用的鲁棒性。

2) 设备制造商可以使用猴子对自己的测试施行压力测试。看设备能坚持多久。

### 测试手段 3: ASE

ASE 意思为 Android 脚本环境，即我们可以通过脚本（比如 Python）调用 Android 的功能，从而定制一些测试。比如打电话，发短信，浏览网页，等。

我们可以扩充它的 API（Java 部分），并用 python 脚本调用这些 API，从而实现丰富的测试功能。

用于 API 部分可以访问到 Android 全部 API, python 又能灵活部署测试，所以 ASE 的扩展性非常好。

个人觉得这对复现某些偶发故障非常有帮忙。

目前 ASE 还处于它的成长期，希望它不断成熟，为开发者提供更多便利。

### 测试手段 4: Robotium

该工具用于黑盒的自动化测试。可以在有源码或者只有 APK 的情况下对目标应用进行测试。

Robotium 提供了模仿用户操作行为的 API，比如在某个控件上点击，输入 Text 等等。

Robotium 模仿普通用户行为，可以试着把一些原来由测试工程师做的测试变成 Robotium 自动化实现。

### 测试手段 5: 单元测试

Android 本身带有很多单元测试例子，我们可以按需要模仿它们，针对某个应用进行单元测试。

注意 Android 的 Instrument 机制功能非常强大，可以测试 UI。

### 测试手段 6: 其它

可以写一个 Android 应用程序，命令行脚本等，在其他方法不能实现时，就只能用这些方法了。

## 总结

对于 CTS/Monkey，我们不需要开发，只要执行测试就可以了。

对于 ASE，我们可以扩充它的现有 API(Java)，用 Python 调用这些 API 实现丰富的测试功能。

一方面我们要充分利用 Android 提供的现成测试，密切关注其进展，另一方面，我们要学习好 JAVA/PYTHON 编程，必要时，自己开发自己的测试用例。

作为模块开发者，一定要写自己的单元测试，一方面可以保证自己的代码没有缺陷，另一方面，也为系统级测试提供素材。

当然，自动化测试关键还在于意识，创意比实现更重要。