

终端应用程序分布式自动化测试策略探讨

Testing strategies for mobile applications against distributing automated testing framework

邹一心¹, 宋光照²

1. 中国电信股份有限公司上海研究院, 上海, 中国, 200122

2. 上海博为峰软件技术有限公司, 上海, 中国, 200001

【摘要】随着移动互联网业务的兴起和智能移动终端操作系统的开放, 终端应用复杂性快速增加。用户对产品功能要求和质量越来越高。实际上, 终端应用开发技术因操作系统、开发语言差异而不同, 终端应用开发周期较短、版本迭代频繁, 本文尝试引入分布式自动化测试工具, 实现功能回归测试自动化, 以提高测试效果和效率。

【关键词】移动互联网; 移动应用; 测试策略; 自动化测试; 关键字驱动;

【Abstract】Along with mobile internet service and the opening of smart phone OS, the complexity of apps is increasing rapidly. The functional requirement and quality is getting higher and higher for end user. Actually, the development is different for the OS and specified programming language. Its version control and iteration is not stable. This paper attempts to introduce a distributed and automated software testing tool, and improve test effectiveness and efficiency.

【Key Words】Mobile Internet; Mobile Application; Test Strategy; Automation Testing; Keyword-driven Testing

1. 引言

以智能终端操作系统为基础, 结合多种基础中间件、业务中间件、通信中间件来实现对应用的支撑。其中应用又可分为本地应用和 Web 应用两类。本地应用体系以 iOS+App Store+NativeApp、Android+Market+NativeApp 为代表, Web 应用以 HTML5/Widget+Web Store+Web App 为代表。

随着 Android 开源手机系统的逐步普及, 逐渐受到手机生产厂商、移动运营商、手机应用开发商的广泛关注, 而随着 Android 4.0 手机、平板和电视版本的统一, 相关产品出现以后, 必将出现井喷式终端应用软件, 大量网络化终端应用软件面临挑战, 如终端平台软件、移动互联网业务应用后台服务和移动应用客户端, 对测试人员来说, 这些软件的测试工作是一个重大的课题, 特别需要通过自动化测试技术来帮助提高效率。

2. 终端应用以用户为中心的质量要求

移动互联网环境下的应用服务, 云是服务的核心, 端是服务的载体与呈现途径, 同时也是用户参与业务互动的重要渠道, 端对用户的服务体验、业务的推广都影响巨大。因此, 发展移动互联网应用更需要关注“端”, 即需重点关注终端应用质量控制。

当前, 终端应用软件仅是终端表现, 其后台业务逐渐在云端, 移动互联网业务端到端的可用与否、系统部署和用户使用的质量高低与否, 运营商和最终用户都非常重视。从软件质量角度看, 其主要的质量要求列举如下:

- 功能性: 终端上移动应用功能越来越复杂, 测试难度、周期和工作量逐步加大, 测试成本快速上升
- 稳定性: 用户使用移动应用时, 与终端的电话、短信、浏览器等背景业务经常产生功能交互, 增加了移动应用的不稳定性
- 可维护性: 用户越来越关注应用业务的用户体验, 在应用上线后需要持续对业务运营质量进行测试和监控
- 性能: 终端上移动应用与终端、网络和服务的性能都有关系, 性能遭遇瓶颈时, 定位需围绕应用关联的整个链路来开展, 导致应用业务优化的成本在不断提高

终端应用软件的最使用者是移动互联网用户, 因此提高软件质量要以用户满意为中心, 以影响软件质量特性为基础开展测试活动, 方可有效对质量进行控制。

3. 终端应用软件测试受影响的因素和自动化测试的必要性

应用程序无论是后台还是客户端，都跟终端平台、应用本身采用的软件架构和特性影响因素有关。测试策略将根据这些特点灵活进行调整。常见的应用软件特性影响因素，比较如下：

移动应用的特性	本地化应用	移动 web 应用
在应用开发方面	较高级、复杂的编程语言编写，对不同系统、硬件及其不同版本分别开发、编译	采用标准 web 语言，如 html5、css、javascript 等编写，适合于符合规范的所有 web 终端
在应用发布方面	从终端下载并安装运行，更新或升级需用户主动认知并获取新资源	开发者实时维护，用户在启动或运行时自动同步加载，任何更新变动无需用户交互而无缝升级
在用户界面方面	操控响应灵敏，对动画、图形处理性能高	受限于浏览器引擎处理能力，时而存在响应延迟
在多媒体方面	音视频格式处理效率高、稳定性强，对 flash 处理需设备支持时才可用	浏览器支持 flash，对各种音视频格式兼容性仍存在些问题
在设备能力方面	能调用硬件设备上各种传感器	通过浏览器引擎目前能调用的设备能力有限

显然，应用软件测试的重点，本地化应用需为不同终端设备考虑适配性测试。其适配性测试因终端版本、机器型号、显示分辨率和用户交互的不同导致了困难和差异。与之类似的移动 web 应用需考虑因网络环境覆盖、连接速率和稳定性等情况考虑性能测试。

因此，为了提高测试覆盖，降低手工测试工作量，考虑利用平台 API，开发相应测试程序，对其被测对象进行模拟客户端请求进行测试，要求自动化测试工具能支持被控对象分布式执行和交互控制。由此，支持分布式执行、跨不同语言的 Agent 是必要的。支持远程交互控制、执行过程中资源监控是终端应用软件测试值得改进的新特性要求。

4. 终端应用以平台 API 为基础的自动化测试

笔者了解到，Android/Phone 终端平台软件自动化测试话题已多次在技术期刊论文中提到，如《Phone 自动化测试技术概述》、《基于 Android 的自动化测

试的设计与实现》，然而探讨业务应用后台和应用客户端测试自动化测试技术较少。平台软件采用的几种自动化测试框架总结如下：

第一，代码驱动测试的测试框架。Android 平台整合了 JUnit 框架和 Instrumentation 机制，调用系统 API 和应用层 Java API，适合对单个应用程序开展自动化测试，基本可实现对某个应用大部分 UI 操作和数据库操作，提供 TestSuite 和 TestRunner 执行器，分别组织、运行测试用例并反馈测试结果。然而，框架局限性在于跨应用测试难度大，复杂交互测试实现困难。

方式二：GUI 录制回放的测试框架。基于 keyevent 键盘事件和图像识别开展自动化测试，实现方式不受进程间安全机制限制，实现多终端交互和交互性系统测试自动化，适合代替压力类型的手动测试。然而，缺点在于靠同步图像进行结果比对，其执行效率降低，另因终端屏幕规格及其分辨率差异，自动化脚本维护成本高。

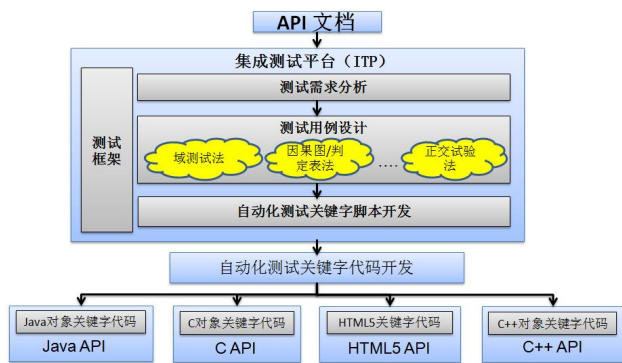
方式三：合理利用 Android 开源测试框架，自带的 CTS 解决兼容性测试困难，自带的 MonkeyRunner 解决压力测试问题，开源项目 Robotium 适用于无源码有 .apk 的情况，提供模拟用户操作行为的 API 实现测试自动化。

4.1. 接口集成测试和分布式测试整合

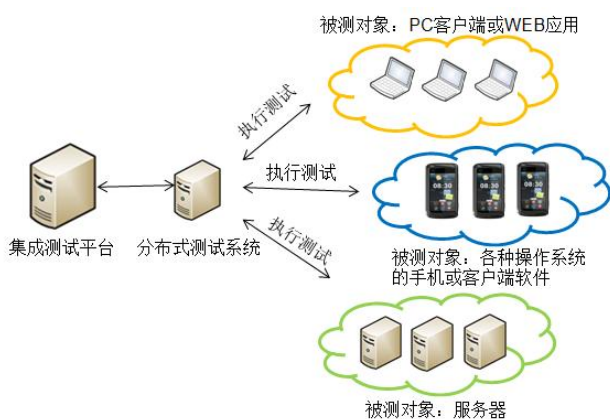
iOS 和 Android 等平台终端的应用软件只是迈向全面移动互联网的阶段性步骤。未来的应用模式是互联网应用模式，应用程序将依互联网而生。软件开发模式从传统模式演变成敏捷开发，快速迭代优化。从需求到发布的生命周期中，快速回归自动化测试势在必行，测试面临迭代频繁、效率低、工作量大等问题，需配合高效的自动化测试手段来完成。即通过接口层开展功能测试自动化，配合持续集成每日构建自动化冒烟测试。

集成测试平台目标可实现测试代码集成、版本生成、被测对象版本更新、测试运行及测试结果反馈的全流程自动化。

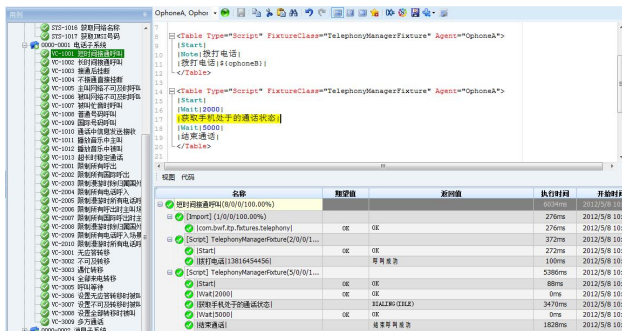
自动化测试框架可满足用例管理、对应用例的脚本管理、对应脚本的执行管理、对应执行的报告管理，为提高自动化测试可维护性和可扩展性，选择的关键字驱动的自动化测试框架如下：



集成自动化测试平台与持续集成系统及分布式测试系统的整合，被测对象扩展到所有手机操作系统、PC 软件和服务器软件，示意图如下：



通过集成自动化测试和分布式测试工具的整合，统一自动化测试平台、统一脚本界面、统一 Agent 配置。如图所示



基于被测对象平台 API 的封装实现，快速转化为自动化测试脚本，在测试框架上进行优化和创新，适合解决以业务为中心的功能测试自动化，通过往对被测对象植入 Agent 实现分布式部署和交互式控制，实现快速回归测试，结合任务调度等改进，实现测试任务自动化执行。

4.2. 自动化测试的适用场景

从应用推广角度，最终用户使用质量，因网络环境差异、设备类型复杂、用户人群不同，致使最终用户对其使用质量诉求不同。以 Android 应用测试为例，预计将覆盖如下测试类型：

- 功能测试：需根据应用程序本身功能场景和 UI 界面，验证其正确性和适合性
- 性能测试：需根据应用程序客户端和服务端验证最终用户的响应时间，与应用相对应的软件路径上的资源消耗和时间消耗
- 终端适配测试：需要根据应用程序在不同系统、不同版本、不同机型、不同分辨率机型上进行兼容性确认测试，实际上也是功能测试的延伸
- 稳定性测试：需要根据应用程序的大小和复杂度来定义稳定性衡量指标
- 安全性测试：需要根据开发、部署和发布等安全性要求进行验证和确认

5. 分布式自动化测试典型应用

全球市场上存在两种较实用的基于真实终端环境的网络化测试解决方案，即美国 Devicely 和以色列 Perfecto Mobile，在功能测试领域获得了成功，其不足之处在于：缺乏利用系统级 API 能力，完全依赖界面交互录制回放的自动化方式；不易于分布式和交互式控制被测对象，支持远程监测软件运行状态。以上问题限制了这种方案的适用性。本文提出分布式执行和 Agent 交互式控制能很好地解决这些困难。

基于集成测试自动化平台和满足不同语言 Agent 进行分布式部署，可实现对移动互联网业务应用软件的功能测试自动化的闭环管理，不需要手工参与，实现功能、兼容性、性能和稳定性测试的自动化功能回归测试。

功能测试	借助工具对移动应用进行安装卸载测试
	在针对移动应用测试的过程中，构造背景业务，验证移动应用的正确性（移动应用运行中，借助工具随时构造接听电话、短信、定时提醒、弹出消息等，能否返回继续移动应用） 移动应用所运行的手机的关闭和重启
兼容性测试	针对大批量的手机，借助工具同时检测不同的手机操作系统版本（Android 1.5、1.6、2.0、2.1、2.2 和 2.3）对移动应用的影响，该款移动应用软件运行的终端适配或版本兼容性
性能测试	借助工具对移动应用对内存的使用情况监测
	借助工具移动应用对处理器的使用情况监测

	借助工具移动应用对网络的使用情况监测
稳定性测试	借助工具构造移动应用运行过程中脱网,重连的情况
	借助工具构造手机没有网络的时候,启动移动应用
	借助工具构造移动应用运行中,手机屏保
	借助工具构造中断恢复
	借助工具构造多手机并行测试
其它测试特性	借助工具构造移动应用启动和运行时耗电量检查

6. 结论

综上所述,本文提出的智能移动终端应用软件分布式自动化测试工具,采用的关键字驱动的自动化框架不仅仅能实现把数据与脚本彼此独立,还实现了数据与逻辑的隔离,从而提高了测试脚本的可维护性和重用度。测试场景关键字代码和关键字脚本界面相互独立,具体实现细节和测试描述相互独立,数据和脚本相互独立。统一平台、统一界面、统一配置,实现对应用程序进行跨平台测试,进行频繁重复性测试,

最大限度解决终端应用软件生命周期中大量回归测试带来的重复工作量问题。

我们提出了在分布式和持续集成整合的自动化测试工具的优化和创新,并结合案例进行了讲解,是一次有意思的测试策略探讨,以此可类推至其它智能平台移动应用开发环节,进一步保障质量,提高测试效率方面起到作用。

参考文献

- [1] 金镛. OPhone 自动化测试技术概述.程序员, 2010(4)
- [2] 谢红霞. 基于 Android 的自动化测试的设计与实现.计算机时代, 2012(2)
- [3] http://en.wikipedia.org/wiki/Mobile_Application_Testing

作者简介

宋光照,男,1978年生,籍贯(江西),上海博为峰软件技术有限公司,解决方案事业部总监,《手机软件测试最佳实践》第一作者,主要兴趣方向:软件分析与测试、数据挖掘与软件测试