调查，比较和评价跨平台移动应用开发工具

Isabelle Dalmasso, Soumya Kanti Datta, Christian Bonnet, Navid Nikaein

移动通信部门，厄尔电信

法国索菲亚科技园

{dalmasso, dattas, bonnet, [nikaeinn}@eurecom.fr](mailto:nikaeinn%7d@eurecom.fr)

摘要：  
因为多样化的平台和软件开发工具包移动应用程序开发变得更加具有挑战性。为了减少开发成本，接触更多的用户，跨多个平台,开发者开始转向到跨平台的应用程序开发工具。在这篇文章中,我们为移植性应用程序开发问题选择恰当的跨平台工具提供了一些新的决策标准。在一个跨平台的框架上需求是明确的。我们对总体跨平台应用程序架构开发进行了探讨。然后用一个分类和比较工具提供一次编写到处运行工具(PhoneGap,Titanium,Sencha Touch)的调查结果。检查Android测试应用程序是使用这样的工具开发的性能CPU,内存使用,耗电量。它是发现,PhoneGap消耗更少的内存、CPU和电量,因为它不包括专门的UI组件。最后,论文总结了贡献,总结一些未来的方向。

关键词：

移动应用发展;跨平台工具;  
PhoneGap;Titanium;Android。

1.介绍

在最近过去的发展中移动平台上已经看到了很好的前景。而黑莓,Bada和  
塞班并不能接地气,iOS和Android赢得了战争的移动平台。在智能手机和平板电脑的时代,移动应用程序给许多行业带来附加值包括交通、电子商务、网络银行、旅游、零售和企业服务。  
开发人员利用先进的功能和智能设备提供革命性的用户体验。使它们成为创新的引擎。因此对移动平台提供者来说最重要的是吸引越来越多的开发人员，为了扩大，丰富，并投入到这个平台的开发中来。不仅移动平台所有者和手机制造商还有网络服务提供者和芯片组制造商正在大力投资开发和发布软件开发工具包来讨好开发人员。也有几个工具厂商喜欢跨平台工具(PhoneGap、Titanium),应用诊断工具厂商(BugSense)和更多的人也在努力博得移动应用程序开发人员的关注。但是移动平台的多样性和各种sdk和其他工具带来了独特的挑战。它们包括选择SDK,用户体验,框架的稳定性、易用性、更新多个平台的成本和部署到应用商店需要的时间。大多数开发人员希望发布的移动应用平台是iOS和Android,并提供一个一致的用户体验(UX)整个平台。为不同的移动平台开发一个应用程序需要深入的了解他们和他们的sdk。这就增加了开发成本、更新和投放市场的时间成本。这就是跨平台的开发工具出现的原因。表中我提供了一个深入的比较原生,Web和跨平台的应用程序开发方法。跨平台工具(例如PhoneGap,Titanium,Rhomobile)允许使用Web技术例如HTML，CSS来开发移动应用程序和用户界面(UI)。然后应用程序可以构建在几个移动平台上(例如iOS、Android、Windows  
黑莓手机7日)。这个过程将非常有用只要开发人员愿意牺牲一些用户体验来换取更大的用户量。这种方法允许为多个移动平台在同一时间开发应用程序。因此应用的开发成本和时间减少了。

表一 选择因素

|  |  |  |  |
| --- | --- | --- | --- |
| 标准 | 原生 | Web | 跨平台混合式 |
| UX的质量 | 优秀 | 好 | 比原生稍微差 |
| 应用的质量 | 高 | 中 | 中偏下 |
| 潜在用户 | 被平台所限制 | 所有智能手机，平板，笔电等 | 范围较大，由于跨平台的缘故 |
| 应用开发代价 | 高 | 低 | 中等偏低 |
| 应用的安全性 | 优秀 | 取决于浏览 | 不是很好 |
| 支持效果 | 复杂 | 简单 | 较复杂 |
| 更新的代价 | 高 | 低 | 较高 |
| 发布到市场的耗时 | 高 | 中 | 低 |
| 应用的拓展 | 可以 | 可以 | 可以 |

本文提出了几种标准，跨过可移植性更关注选择一个合适的跨平台工具来开发。目前有几个这样的工具。我们提出了跨平台框架的要求和高层次的体系结构。测评一些像这样的一次编译处处运行的工具（WORA）的API和文档，开发环境，部署，优势和弱点。为了测试这些工具性能，我们已经开发了专门的安卓应用程序用来测量处理器的使用，内存使用和功率消耗。在测试过程中，我们发现PhoneGap消耗更少的CPU，内存和电源相较于其他工具。但使用PhoneGap开发的应用程序并没有很好的用户界面。要创建一个更好的用户界面，就必须求助其他工具。在这种情况下，Sencha2就比较出色。论文的其余部分如下。第二节概述跨平台开发框架的基本要求。第三节描绘了一个移动平台应用程序开发的基本架构。第四节介绍WORA工具的详细调查。第五部分对WORA工具进行分类，并提供了一个工具间的比较结果。第六部分演示性能测试结果。在本文的最后，我们将得出结论与一些未来的方向。

2.跨平台框架的需求

我们已经确定了任何跨平台框架的要求如下：

多平台支持：框架必须支持多个移动平台。支持Android和iOS是非常重要的因为他们有最大的应用市场份额。

·丰富的用户界面：目前的跨平台工具不能像本地应用程序那样提供丰富的用户界面（UI）。因为一个应用程序的成功取决于界面的用户体验，丰富的视觉界面开发应纳入进来。支持复杂的图形（二维，三维），动画，多媒体是必要的。

·后台通信：移动设备推动一个“始终连接”的模式，在该模型中，用户全天候在社交网站上分享材料，观看视频，实时聊天，接收并交流信息。需要有后台支持，通信协议和数据格式是绝对必须的。

·安全：应用程序的跨平台开发工具是不安全的。这些工具和应用还是需要适当的安全研究的。

·支持应用程序的扩展：在现有的应用程序上进行拓展像应用内购/计费能力。

·功耗：这是个重要的问题，如今随着智能手机和平板电脑的人每天被使用。生成的应用程序必须为电池优化。

·访问内置的功能：工具必须能够访问智能设备的内置功能。使用相机，传感器，地理定位和更多功能有助于提供更好的用户体验。

·开源：它吸引了更多的应用开发者和开发者社区可以参与错误修正和进一步发展。需要注意的是，这不是一个技术要求。

3.跨平台应用程序开发的通用体系结构

本节提供了一个通用的跨平台应用程序开发的架构如图1所描绘的。

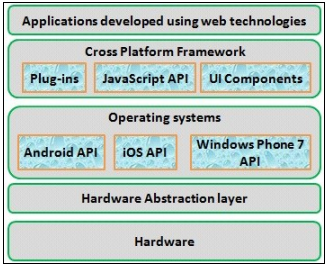


图1：跨平台移动应用开发的一般体系结构。

应用程序开发人员实现业务逻辑或使用网络技术的应用功能。这个跨平台框架允许实现用户界面，访问存储设备和设备功能（传感器，摄机和联系人）与JavaScript API进行交互。API将依次与本地接口的移动平台进行交互。该应用程序将单独建立并生成不同平台上的可执行文件。移动平台的APIs实际上是产生各自的应用程序的根本。由此产生的应用程序可以运行在相应的移动设备。

4.一次编译处处运行（WORA）工具的调查

在市场上有几个该工具如PhoneGap，Titanium，Rhomobile，jQuery Mobile等。本文给出了这些工具的特性为应用程序开发提供了参考。

A. PhoneGap

PhoneGap的开源框架为建原生应用提供了一个恰当的工具集，包括HTML，JavaScript和CSS [ 1 ]、[ 11 ]、[ 12 ]、[ 13 ]。这是相当流行的在用户中主要是因为它的灵活性，简单的架构和易用性。这是用户中相当流行的主要是因为它的灵活性，简单的架构和易用性。

·API和文档：它提供了一个纯JavaScript / HTML / CSS API（WebKit框架）一个包括material资源的库。PhoneGap 还提供了专利的文件。

·开发环境/建立：它提供了一个Eclipse插件，但也可以使用其他IDE构建

移动应用程序。

·部署：在编译生成的二进制文件不能在任何移动应用市场上发布。开发者获得最终可以发布版本需要通过PhoneGap的收费构建服务。

·框架稳定性：它是一个成熟的框架。

·优点：提供所有本地包装的源代码，所以它可以进一步定制。PhoneGap支持广阔的应用平台。应用程序是完全建立在HTML，JavaScript和CSS上，使网页开发人员很容易地使用该工具。

·弱点：有些原生UI组件，设计模式和开发工具的缺乏。然而, 开发商可以自由结合PhoneGap与另一个工具像jQuery或Sencha来制作更好的用户界面的应用程序。

B. Sencha Touch 2.0

**·**API和文档：Sencha Touch 2.0是一个强大而复杂的框架。HTML代码通过JavaScript API生成。在理论上Sencha SDK工具是独立的开发平台提供一个访问本地接口的一个子集（如照相机、通知、连接、方向）。它还提供了建立本地的包装并且可部署iOS和Android应用市场[ 4 ]。

**·**框架稳定性：它是一个成熟的框架。

**·**另外Sencha也可以作为PhoneGap项目的JavaScript / CSS资源。这是一个变通方法，但允许开发用户界面使用Sencha的API。

C. PhoneGap + Sencha Touch 2

**·**API和文档：一个完整的MVC设计模式的JavaScript 框架和完整的API文档可供开发人员使用。

**·**开发环境/构建：采用引入的主要的CSS /

JavaScript文件。

**·**框架稳定性：它是一个成熟的框架。

D. PhoneGap + jQuery mobile

jQuery Mobile是一个轻量级的API和开发任何查询网络服务的应用是很容易的[ 3 ]。

**·**API和文档：jQuery mobile建立在 jQuery API上并且提供了一个包含CSS和其他组件的JS库。mobile APIs良好的文档让开发人员熟悉工具很快。

**·**开发环境/构建：环境使用引入的CSS和JavaScript文件建立常规的安卓应用程序。

**·**框架稳定性：它是一个成熟的框架。

E.应用制作

它是一个为移动端，平板电脑和台式电脑设计的基于云的开发环境。应用制作可以设计一个通过浏览器连接平台的网页应用。生成的代码被存储到服务器，用私人或公共的地址来运行应用程序。生成的代码可以下载。

F. 应用加速 Titanium Studio 2

**·**API和文档：它提供了丰富的API和底层的对象像TCP套接字。UI对象可以通过JavaScript API来定制。并没有没有HTML和CSS编码。该工具有很好的API文档，但是SDK工具的使用（编译器等）不暴露[ 2 ]。

**·**开发环境/构建：IDE Titanium Studio（基于Eclipse的IDE）是必须的。它包括SDK和工具。它是用Python脚本生成原生代码和原生应用程序的编译和封装都来自该脚本（Android）。

**·**框架稳定性：它是一个成熟的框架。

**·**优点：原生代码的输出是非常快的并能流畅的运行移动设备上。安装过程对于新的开发者来说非常的容易。开发文档非常的优秀并且还将支持平板电脑应用的开发。

**·**弱点：有限的API和小部分手机支持。

5.对WORA（一次编译处处运行）工具的分类

在本文中，我们专注于WORA工具因为他们实现了一个应用在多个平台上运行。我们做了很大的努力基于这些工具所提供的功能将他们进行分类。五个主要的因素，可供开发者选择一个可能的类别。

A.应用开发工具

·Hybrid App：这类工具提供了一个特殊的平台，应用程序具有HTML页面渲染能力。它扩展的HTML功能，通过API允许访问设备的特定功能。一些工具包括库，以提供特定平台的用户界面。appMobi、PhoneGap等就是这类工具。

·Mobile Web：这些工具主要是JavaScript库和HTML恰当的配合和相关的CSS在不同类型的设备上渲染你的Mobile网站。一些这样的工具还可以与混合应用程序结合将应用打包成本地应用程序。例如，jQTouch，jqurey mobile和Sencha Touch。

·Generator：这些工具，开发人员编写在一个特定的语言（JavaScript或PHP）并且工具把它转换成一个适用于各种平台的可部署的原生应用。这个可部署的应用程序可能包括一个运行时引擎或虚拟机。Generator和Hybrid之间的主要区别在于使用生成的本机代码的性能增益而不是HTML渲染。Appcelerator和rhomobile [ 9 ]是典型的这种工具的例子。

B.用户界面工具

·视觉工具：他们提供了一个可视化的界面，界面元素/小部件主需要拖拽在屏幕上就可以内部处理由工具来完成。结果是怎么样取决于该工具是一个本地应用程序或移动网站。一些可视化工具可以联合应用开发工具（主要是混合型应用工具）。Dragon rad[ 5 ]，[ 6 ]Application Craft，July Systems[ 7 ]，[ 8 ]是一些视觉工具的例子。

C.开发人员指南

我们为开发人员提供一个选择一个上面的指导方针。

·原生的用户体验：如果得到应用希望拥有类似于原生的用户体验，则必须选择“Generator”。当用户体验不那么重要的时候，其他类型的工具也是可以的。

·离线/在线使用：Mobile Web应用程序允许通过浏览器在线。Generator和混合应用程序创建安装程序允许脱机使用。

·兼容性：移动设备的发展非常快，在硬件、移动平台和SDK上。对一个跨平台的应用程序需要长期保持下去而言，Mobile Web是最安全的，所有移动设备都支持HTML5和更新的版本。

·有限的内置功能访问：在这种情况下，混合应用程序可以优选。但是，如果应用程序需要频繁访问内置的功能，应选择Generato，虽然原生应用更适合。

·安全：Hybrid和Generator可能比纯移动网络应用程序更安全。通过原生的方法可以更好的提供安全性。

D.用WORA工具的应用的类型

几种类型的移动应用程序使用的该工具：商务、游戏、多媒体、工具，娱乐，电子商务和社交。一些应用程序开发的PhoneGap是：维基百科（多媒体），BBC Olympics（多媒体），untapp（社交）， Facebook（社交）和Zinga游戏黑手党战争手机游戏（游戏）。

一些由Titanium开发的应用程序：Legoland Parc California（多媒体/娱乐/电子商务），易趣网企业（业务）。纽约参议院（多媒体）和Zipcar公司（电子商务）。

6.性能评价

检查性能，已开发利用跨平台的办法的安卓测试应用。他们有下面的功能，可以被归类为一个“商务”式应用。

·UI上表现就是一些按钮，每一种都有要求类型。当单击按钮时，应用程序查询免费的网络服务（存在于互联网上的）通过Ajax，REST和SOAP技术。

·解析和显示的不同格式的内容：文本，XML，JSON。

应用程序的安装和测试都在一个运行Ice Cream Sandwich（Android 4.0.4）的Archos平板上进行。测试环境包括Linux Fedora，Eclipse IDE Indigo Classic，Web开发插件和Android ADT 16。 图2描述了测试程序的用户界面。值得注意的是，没有采取措施使用复杂的用户界面而是采用默认的版本。以上应用为代表的移动端机器对机器的应用。在框架内项目WL-Box 4G [ 19 ]，我们正在开发的移动应用程序接收来自不同传感器的数据通过一个设备入口。数据被飞快的处理和应用的用户界面相应刷新。

A.内存使用

我们发现得用Titanium开发这种类型的应用程序。但当“开发者指南”是深思熟虑的，选择PhoneGap使因为应用的需求。因此，应用程序开发利用以下的工具，然后，内存和处理器的使用并且注意到电力消耗。

·只有PhoneGap。

·PhoneGap和jQuery Mobile

·PhoneGap和Sencha Touch 2

·Titanium

内存使用的信息是从ADT的工具DDMS获得的。

·实际使用的物理内存（PSS）：PSS是与其他进程共享的内存，在某种程度上认为内存的分配是平均的。如果这个进程被终止内存不会被释放，但是表示这个进程的数额 “贡献”整体内存负荷。

·进程独自占用的物理内存（USS）：是页面的设置独特的过程。这是内存的量如果应用程序被终止，将被释放。

表二提供了内存使用率。

|  |  |  |
| --- | --- | --- |
| 开发工具 | PSS | USS |
| PhoneGap only | 12091 | 6036 |
| PhoneGap+jQuery Mobile | 14730 | 9424 |
| PhoneGap+Sencha Touch 2.0 | 24526 | 20164 |
| Titanium | 17500 | 8676 |

表二。内存使用

从上面的表格可以清楚的是，用PhoneGap写的应用只有最小的PSS和USS值。这是由于PhoneGap的设计不使用任何的设计类型或工具来改进的用户界面。所以我们可以得出这样的结论：内存的使用量将增加在产生更好的用户界面下。PhoneGap结合jQuery mobile或Sencha Touch 2才是一个完整的环境为更好的用户界面开发应用。在这种情况下，内存使用量增加因为HTML和JavaScript文件的使用。Titanium包括全SDK，所以内存使用量更高一些。

B. CPU使用率

要测量的处理器的使用，该应用程序开发使用PhoneGap和其他工具。代码段通过PhoneGap被添加到Android Activity来度量处理器的使用。但没有应用程序可以使用Titanium开发因为它不允许添加Activity。一个插件可以开发用于此目的，但这是耗时的。我们已经有了2个不同的途径来记录中央处理器的使用如下：

 ·第一种方法需要CPU在每个状态的快照在Activity的生命周期中（例如onCreate，onStart，onPause，onStop和onDestroy）在应用中。

·另外一种方法是读'TOP'结果每一秒在整个生命周期的应用程序。然后计算在生命周期中每一种状态的平均值。开发的应用程序执行相同的功能用来做前期的测试。第三表描绘了测试结果。从CPU快照方法获得的值是计算应用程序正在做大量的计算，并因此用非常高的值表示。还需要注意的是这些值可能会有很大的不同，从一毫秒到另一个因为快照在截取的是CPU非常短的时间。

从“top”的结果获得的值显示不同的处理器使用量。最小值是0 -一旦应用程序读取请求页面显示，该应用程序不使用任何处理器等待下一个系统的输入。平均值是计算使用的总时间。对于这种方法显然的是，第一个应用程序利用非常少的处理器，但用户体验不是很复杂。当Sencha Touch 2 和PhoneGap一起使用，CPU使用率更多但用户体验明显好。

表三 cpu使用量

|  |  |  |
| --- | --- | --- |
| 开发工具 | CPU使用量（CPU快照） | CPU使用量（“Top”命令） |
| PhoneGap+HTML+CSS tools | 81.92771% | Max: 10% Min: 0% Average: 2% |
| PhoneGap+jQuery+HTML | 80.26316% | Max: 42 Min: 0% Average: 10% |
| PhoneGap+Sencha Touch 2.0 | 44.0% | Max: 32% Min: 0% Average: 8% |

C.功耗

最近移动应用程序的功耗已得到研究人员的注意[ 14 ]，[ 15 ]，[ 16 ]。为了有效地使用电池的移动设备，使用跨平台工具开发的应用程序应该更高效。我们使用“Power Tutor”测得同样的应用程序的功率消耗[ 17 ]，[ 18 ]。这是一个非常受欢迎的Android APP，是一款安装在移动设备上能够报告功耗的个人应用程序。表四列表的就是测试结果。值得注意的是，报告的值是应用程序消耗的平均功率。

表四 耗电量

|  |  |
| --- | --- |
| 开发工具 | 功率（mW） |
| PhoneGap+HTML+CSS tools | 107 |
| PhoneGap+HTML+JQuery | 168 |
| PhoneGap+Sencha Touch 2.0 | 120 |

同样的结果指出，在三个程序中第一个程序消耗最少的电量。原因是在于其用户界面非常简单。使用PhoneGap和Sencha Touch 2开发的应用程序也很高效因为它消耗了120MW的电量。从以上的评价，很显然，PhoneGap与Sencha Touch 2比较高效从CPU使用率和耗电量可以看出来。但目前还没有龙头跨平台开发工具能够符合市场上的所有要求。然而在开发者之间最流行的工具是PhoneGap和Titanium。我们确定它们主要的差异就在表5。在第二节的对跨平台的需求分析也有提到上述的结果分析。所选择的工具都是开源的，可以构建多种操作系统的应用。本文介绍了安卓应用的性能并将开发苹果手机应用程序且在未来的作品中评价。PhoneGap不包含复杂的UI构建工具。因此对内存，处理器和耗电量的要求小于PhoneGap和jQuery的组合或Sencha Touch 2。所有工具都支持后端通讯。其余需求，如安全性、应用程序扩展将在今后的工作中进行评估。

表五 PhoneGap和Titanium的区别

|  |  |
| --- | --- |
| PhoneGap | Titanium |
| PhoneGap应用是用HTML5写的并且运行在原生的容器中  开发者可以利用HTML5，CSS和JavaScript  UI元素可能由jQuery Mobile，Sencha Touch 或其他 JS框架来渲染。 | Titanium应用是用JavaScript写的编译成原生应用并利用原生的API来控制移动设备  并不是一个运行在Web容器里面的H5应用  从设计方面来讲，这个工具能够提供更好的性能和用户体验。 |

7.结论

简言之，我们已经很关注跨平台开发工具，因为他们可以构建不同平台的应用程序而且开发的成本和发布到市场的时间代价很小。虽然用户体验不如原生应用程序，但应用程序可以在几个平台上发布一次，以达到最大的用户数量。我们已经描述了一般跨平台框架的要求及其一般的构建。涵盖几个开发工具的详细调查，让开发人员更好的认识这些工具。为开发者将这些工具进行分类并指导什么样的需求需要用什么样的工具。提供了用PhoneGap和Titanium开发的应用例子。我们测试了安卓测试程序的性能在内存，CPU的使用量和功率消耗方面都进行了分析。我们发现用PhoneGap开发的应用占用内存，CPU和耗电量是最少的，但只提供了一个非常简单的用户体验。报告结果也表面，如果你觉得UI体验比内存占用率更优先，PhoneGap和Sencha Touch 2.0组合能够更有效。 将来，我们的目标是确定几个其他性能测试案例。同时，其它的跨平台工具也应该用本文的方法来测试一下性能。应用程序开发将为iOS平台。进一步实验将检查安全性和应用程序跨平台工具的扩展能力。还存在一个尚未解决的重要问题，我们尚未进入从传感器因为当前移动设备严重依赖于它。这样的利用跨平台的方法的信息挖掘必须进一步研究。

引用  
[1] http://phonegap.com  
[2] http://www.appcelerator.com  
[3] http://jquerymobile.com  
[4] http://www.sencha.com  
[5] http://dragonrad.com  
[6] http://www.applicationcraft.com  
[7] http://julysystems.com  
[8] http://www.netbiscuits.com  
[9] http://www.rhomobile.com  
[10] http://www.appmobi.com  
[11] https://github.com/remy/PhoneGap-Plugin-WebSocket  
[12] https://github.com/phonegap/phonegapplugins/tree/master/iPhone/InAppPurchaseManager  
[13] https://github.com/anismiles/websocket-android-phonegap  
[14] S. K. Datta, “Android stack integration in embedded systems,” in  
International Conference on Emerging Trends in Computer &  
Information Technology, Coimbatore, India, 2012.  
[15] Datta, S.K.; Bonnet, C.; Nikaein, N., "Android power management:  
Current and future trends," First IEEE Workshop on Enabling  
Technologies for Smartphone and Internet of Things (ETSIoT),  
pp.48,53, 18 June 2012.  
[16] Lee, J.; Hyunwoo Joe; Hyungshin Kim, "Smart phone power model  
generation using use pattern analysis," 2012 IEEE International  
Conference on Consumer Electronics (ICCE), pp. 412,413, 13-16 Jan.  
2012.  
[17] L. Zhang, et al. “Accurate online power estimation and automatic battery  
behavior based power model generation for smartphones.” In Proc. Of  
ACM CODES+ISSS’10, Arizona, USA, 2010, pp. 105-114.  
[18] https://play.google.com/store/apps/details?id=edu.umich.PowerTutor  
[19] http://www.pole-scs.org/projet/wl-box4g