

Android

韩迪 / 主编

创意实例详解

软件工程规范化的标准组织内容,并配有“搜索关键字和参考文献”

提高读者信息挖掘的能力和扩展其知识面



每个章节遵循技术研发知识体系的严密性

在容易产生错误、不易理解的环节配以详细的开发截图

将重要的知识点和开发技巧以“知识点”、“注意”、“小技巧”等活泼形式呈现给读者,保证不会见到一大堆概念和代码



北京邮电大学出版社
www.buptpress.com

TN929.53/262

2012

Android 创意实例详解

主编 韩 迪

REID

北方工业大学图书馆



C00303235



北京邮电大学出版社
www.buptpress.com

内 容 简 介

全书分为3个部分,共12章,全面涵盖了Android底层框架、通信应用程序开发、本地数据应用、网络数据应用、盈利模式分析以及云计算服务应用等多个开发领域。1~5章以知识点训练为主,6~10章以作者上传到Android市场上的真实项目进行剖析,其中部分案例曾获得了省级、校级和企业级等奖项,而最后两章融入云计算服务,以软件工程的规范化行文。

本书深入浅出,适合有面向对象开发经验和希望深入了解Android软件开发的读者阅读。

图书在版编目(CIP)数据

Android 创意实例详解/韩迪主编. --北京:北京邮电大学出版社,2012.1

ISBN 978-7-5635-2899-8

I. ①A… II. ①韩… III. ①移动终端—应用程序—程序设计 IV. ①TN929.53

中国版本图书馆 CIP 数据核字(2012)第 005641 号

书 名: Android 创意实例详解

主 编: 韩 迪

责任编辑: 王丹丹

出版发行: 北京邮电大学出版社

社 址: 北京市海淀区西土城路 10 号(邮编:100876)

发 行 部: 电话: 010-62282185 传真: 010-62283578

E-mail: publish@bupt.edu.cn

经 销: 各地新华书店

印 刷: 北京联兴华印刷厂

开 本: 889 mm×1 194 mm 1/16

印 张: 26.5

字 数: 853 千字

版 次: 2012 年 1 月第 1 版 2012 年 1 月第 1 次印刷

ISBN 978-7-5635-2899-8

定 价: 55.00 元

• 如有印装质量问题,请与北京邮电大学出版社发行部联系 •

前 言

随着计算机及相关技术的发展,通信能力和计算能力的价格正变得越来越便宜,所占用的体积也越来越小,各种新形态的传感器、计算/联网设备蓬勃发展;同时由于人类对生产效率、生活质量的不懈追求,人们开始希望能随时随地,无困难地享用计算能力和信息服务,由此带来了计算模式的新变革。

普适计算是计算机技术计算模式的新时代,直接称为普适计算(Pervasive Computing 或 Ubiquitous Computing)时代。普适计算是指在普适环境下使人们能够使用任意设备、通过任意网络、在任意时间都可以获得一定质量的网络服务的技术。

所以将来智能手机市场发展的趋势并不是在于 Android 提供了新功能,苹果又有了新的用户体验,或者其他手机平台又提供了更有趣的软件市场(更长远的来说并不是手机市场发展,而普适计算设备发展,这其中包括计算机、手机、汽车、家电等能够通过任意网络上网的设备)。竞争的重点在于提供面向客户的、统一的、自适应的网络服务。服务内容包括计算、管理、控制、资源浏览等。所以本书以此为前提,并不在炫耀 Android 架构的优点,也不强调如何弥补其他系统的软肋,更多是结合 Android 自身的开源平台和 Google 这个强有力的后盾(不光只有搜索引擎),设计出更多方便生活、服务社会的应用程序。

本书以 Android 应用程序实例开发为主线,通过由浅入深的 12 个单元的真实项目,全面涵盖了 Android 底层框架、通信应用程序开发、本地数据应用、网络数据应用、盈利模式分析以及云计算服务应用等多个开发领域。

本书分为 3 个部分。

第 1 部分:知识点应用——1~4 章以知识点训练为主,分别融合了 Android 常用组件、提高用户体验(最新 SDK 的功能)、Java I/O 操作、电话与短信服务、使用网络数据以及 Android 市场的运营模式,包括如何在自己开发的程序中插入广告,通过反编译理解 Android 底层框架的原理。

第 2 部分:项目实训——5~10 章以作者上传到 Android 市场上的真实项目进行剖析,其中部分案例曾获得了省级、校级和企业级等奖项,值得一提的是“Android 移动体感”获得了粤港澳软件设计大赛移动组的二等奖。为了更好地让读者能够理解项目精髓,作者将项目拆分成主要功能模块,将项目实际开发经验和建构主义教学思想融入其中,培养读者分析问题的能力解决问题的能力。

第 3 部分:系统实战——为了更好地突出普适计算的重点,本书最后 2 章以 Android 平台为基础,分别讲解城市公交线路查询系统和 LBS 地理位置服务的移动社交应用系统。这 2 个工程不仅只有 Android 客户端,还需要服务器的支持,整个系统的讲解依照软件工程的规范化形式组织内容。第 3 部分代码较多,知识点涉及范围广,而且包含了一定的算法,所以难度也较大。

本书 3 个部分环环相扣,注重对实际动手能力的指导,在遵循技术研发知识体系的严密性的同时,在容易产生错误、不易理解的环节配以详细的开发截图,并将重要的知识点和开发技巧以“知识点”、“注意”“小技巧”等活泼形式呈现给读者。所有程序实例的讲解方面,按照“搜索关键字”(挖掘本章中在搜索引擎中的需要的关键字)、“本章难点”(帮助读者把握重点)、“项目简介”(项目的功能介绍)、“案例的设计与实现”(如何将功能需求进行分析、拆解最终实现)、“项目心得”(笔者的心得体会)、“参考资料”(笔者在解决问题时候查阅的网页、书籍或者其他资料)和“常见问题”(初学者会出现的问题)。

本书特色如下:

对大脑中已经存储的信息越熟悉,那么对新输入进来的内容理解起来也就越容易,简单点来说就是熟能生巧。根据作者的一些学习经验以及对以往一些案例讲解模式进行修改:

(1) 提高教学效率。作者将项目分别拆成单独功能 Demo,这并不只是简单的分别叙述子功能模块,而

是为程序中的功能,单独做一个案例。比如这个程序用到4个知识点,然后分别为4个知识点各做一个案例,案例与案例之间并无联系,其目的是:尽量避免很多实训类的教材,案例非常好,但是读者无法实现或者理解其中一部分功能,而不能整体实现,最终不得不放弃整个项目的问题,将项目拆开虽然会带来更多的工作量,但是读者学习的效率会更高。

(2) 错误教学。因为本书大部分项目已经放上了市场,通过市场的反映、用户体验、自身检查,将软件中不恰当的地方重新修改,并将这些经验和技巧反馈融入教材中。

(3) 培养分析问题和解决问题的能力。作者并不仅教读者第一步怎么做,第二步这么做,而且思考这个项目该如何拆分,大的问题该如何变成小的问题,小的问题如何去寻找答案,解决的方法有多少种,哪一种更好。此外,这个问题的解决方法还可以应用其他什么方面。希望能够达到授人鱼不如授人以渔,授人以渔不如授人以欲的目的。

(4) 提高信息挖掘的能力。现在搜索引擎提供的内容已经很丰富了,问题的解决方法一般网上都会有,但是关键是:如何找到这些信息,然后如何整理。所以本书通过开始的“搜索关键字”和课后提供的“参考资料”的超链接及其注解,辅导读者能够更好的利用搜索引擎,提高自学能力。

致谢:

衷心感谢在本书编写过程中帮助我的每一个人,包括在技术群和论坛中热心网友。

特别感谢广州大学软件学院提供移动互联网实验室。

同时感谢刘瑞斌,一位非常低调、工作认真的Android工程师、移动互联网爱好者,在案例的设计和建议上,对我帮助良多。

感谢吴溢彬,一位技术面面俱到、力求完美的阳光宅男,无论是UI设计、软件测试都能给予我灵感和启发。

感谢丁文浩,在软件调试过程中的细心、踏实的工作,和较强责任心,值得我借鉴。将来必定前途无量。

感谢翁耀彬,对Android新技术执着和积极向上的态度,总能激励着我。

感谢吴川原、王庆光、薛向强、丘志仕和扣叮小组的每一位优秀成员,和你们交流,我总觉得我有很大的进步空间。

最后感谢黄丽芳女士,让我在坚持笔耕之时,心中总有依靠。

由于书中内容比较多,难免有所疏漏,诚挚感谢读者指出书中不足,这样我就能和读者共同进步和提高。

作 者

目 录

第 1 章	学以致用	1
第 2 章	用户体验	16
第 3 章	简单文件管理器	40
3.1	代码布局	40
3.2	Java 操作	52
第 4 章	通信功能多样化	79
第 5 章	如虎添翼	96
5.1	二维码和翻译机	96
5.2	天气预报	116
5.3	谷歌卫星地图	139
第 6 章	音乐播放器	152
6.1	淡妆浓抹 Activity	152
6.2	任劳任怨的 Service	166
6.3	双管齐下的 Handler	182
第 7 章	电池挂件	197
7.1	关闭后台程序	198
7.2	电量的获得	208
7.3	整合 App Widget	216
第 8 章	放心行程表	228
8.1	界面与数据库	229
8.2	数据库	253
8.3	闹钟与情景模式	260
第 9 章	手机小秘	275
9.1	交互设计	277
9.2	架构	293
9.3	功能实现	302
第 10 章	Android 移动体感	316
10.1	搭建程序框架	317
10.2	实现控制和通信	335
10.3	传感器开发	347
10.4	远程控制	351
10.5	服务器设置	364
第 11 章	基于 Android 平台公交线路查询应用系统	380
第 12 章	基于 Android 平台与 LBS 地理位置服务的移动社交应用系统	399

第1章 学以致用

搜索关键字

Android Market;eomarket;Andorid 反编译;Android 广告。

本章难点

本章涉及 3 个部分：

- (1) 将文件打包,做成可安装文件 apk。
- (2) 如何反编译 apk 文件,并且阅读反编译后的文件?
- (3) 如何在项目中插入广告?

1. 项目简介

本章演示的案例是一个计算体重的例子,例子的选取和本章讲解的内容联系并不大,读者可以根据自己的兴趣选择项目。本章中将项目源代码进行打包,然后在演示反编译的过程,最后在项目中插入广告。感觉难度好像不高,其实每个细节都十分重要,稍有差错都不能成功。

2. 案例设计与实现

(1) 打包签名

在打包签名前,可以将所选择的工程打开进行观察,如图 1-1 所示。无须特别设置(只要编译成功运行在模拟器之后)就会发现已经生成了一个 apk 文件,这个文件经过测试后,既可以安装到自己调试的机器上,又可以安装到其他的 Android 设备上。既然已经有了 apk 文件,那么还需要打包做什么呢?

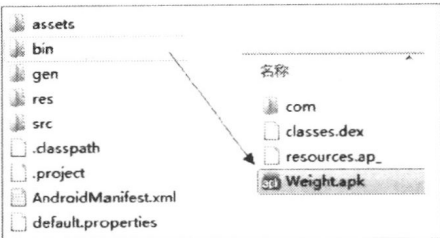


图 1-1 项目结构图

图 1-1 中的 Weight.apk 属于 Debug 版本,而传到 Android 市场上“必须”是 Release 版本,那么 Debug 和 Release 之间有什么本质的区别呢?

【知识点】Debug 和 Release 的本质区别

Debug 通常称为调试版本,它包含调试信息,并且不作任何优化,便于程序员调试程序。Release 称为发布版本,它往往是进行了各种优化,使得程序在代码大小和运行速度上都是最优的,以使用户很好地使用。

Debug 和 Release 的真正区别在于一组编译选项。下面列出了分别针对二者的选项如表 1-1 和表 1-2 所示。(当然除此之外还有其他一些,如/Fd/F0,但区别并不重要,通常它们也不会引起 Release 版错误,在此不讨论。)

表 1-1 Debug 版本

参 数	含 义
/MDd/MLd 或/MTd	使用 Debug runtime library(调试版本的运行时刻函数库)
/Od	关闭优化开关
/D“_DEBUG”	相当于 #define _DEBUG,打开编译调试代码开关(主要针对 assert 函数)
/ZI	创建 Edit and continue(编辑继续)数据库,这样在调试过程中如果修改了源代码不需重新编译
/GZ	可以帮助捕获内存错误
/Gm	打开最小化重链接开关,减少链接时间

表 1-2 Release 版本

参 数	含 义
/MD/ML 或/MT	使用发布版本的运行时刻函数库
/O1 或/O2	优化开关,使程序最小或最快
/D"NDEBUG"	关闭条件编译调试代码开关(即不编译 assert 函数)
/GF	合并重复的字符串,并将字符串常量放到只读内存,防止被修改

实际上,Debug 和 Release 并没有本质的区别,它们只是一组编译选项的集合,编译器只是按照预定的选项行动。事实上,程序员甚至可以修改这些选项,从而得到优化过的调试版本或是带跟踪语句的发布版本。

如果不能理解它们之间的根本区别也不重要,因为原理上的区别平时也用得比较少,接下来就着重叙述为什么必须在 Android 市场上使用 Release 版本。

① Android APK 签名的作用可以防止出现“李鬼”现象,比如说恶意使用其他公司的 package name 代替正规的应用;同时,相同签名应用之间的数据共享不受限制,例如项目发布第 2 个版本,只要是相同签名签出来的文件,用户下载安装会自动覆盖老版本。反之,则不会(这是最重要的),而且使用特殊签名会得到不同的权限。

【注意】Google 在 Android 系统方面的策略是非营利性质的,所以其签名作用和其他智能手机(如 sym-bian)签名是完全不同的。

② bin 目录的 apk 文件可以安装,但是不能作为 Release 发布到市场,因为它使用的是 Debug 签名,该文件名为 Android.debug.keystore,如图 1-2 所示。如果要发布 Release 版本,就必须新建 Release 的keystore。

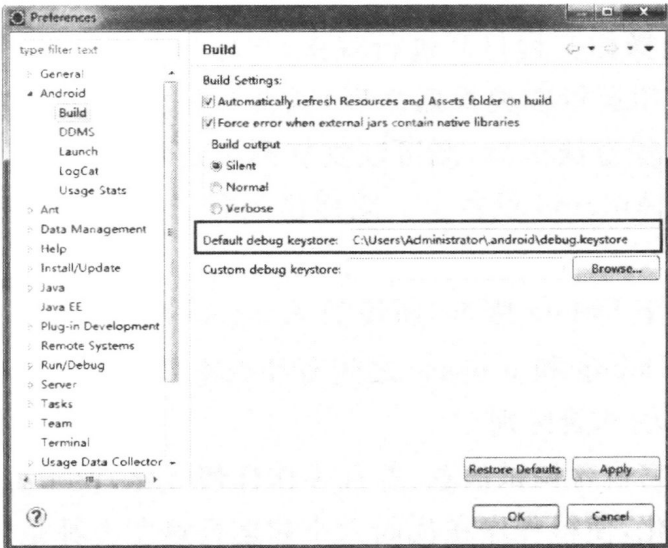


图 1-2 环境变量

③ 应该如何做 Release 版的 apk 发布呢?

关于如何发布 Release 版的 apk,Android 官方给出了 3 种打包方式,最后的结果都是一样的。

方式 1:命令行手动编译打包。

方式 2:使用 ant 自动编译打包。

方式 3:使用 Eclipse+ADT 编译打包。

前面 2 种打包方式流程如图 1-3、图 1-4 所示。在这里推荐采用第 3 种比较简单且高效的方法。对于前 2 种打包方式,有兴趣的读者可以查阅搜索引擎。

下面就介绍目前最常用的编辑方法,通过 Eclipse 和 ADT 插件的向导即可完成。

当 SDK 和 ADT 版本很旧时(1.5 版本以前)对工程打包,通常在图形化界面下右击工程,选择 Android Tools,然后选择 Export Signed Application Package 进行打包,如图 1-5 所示。

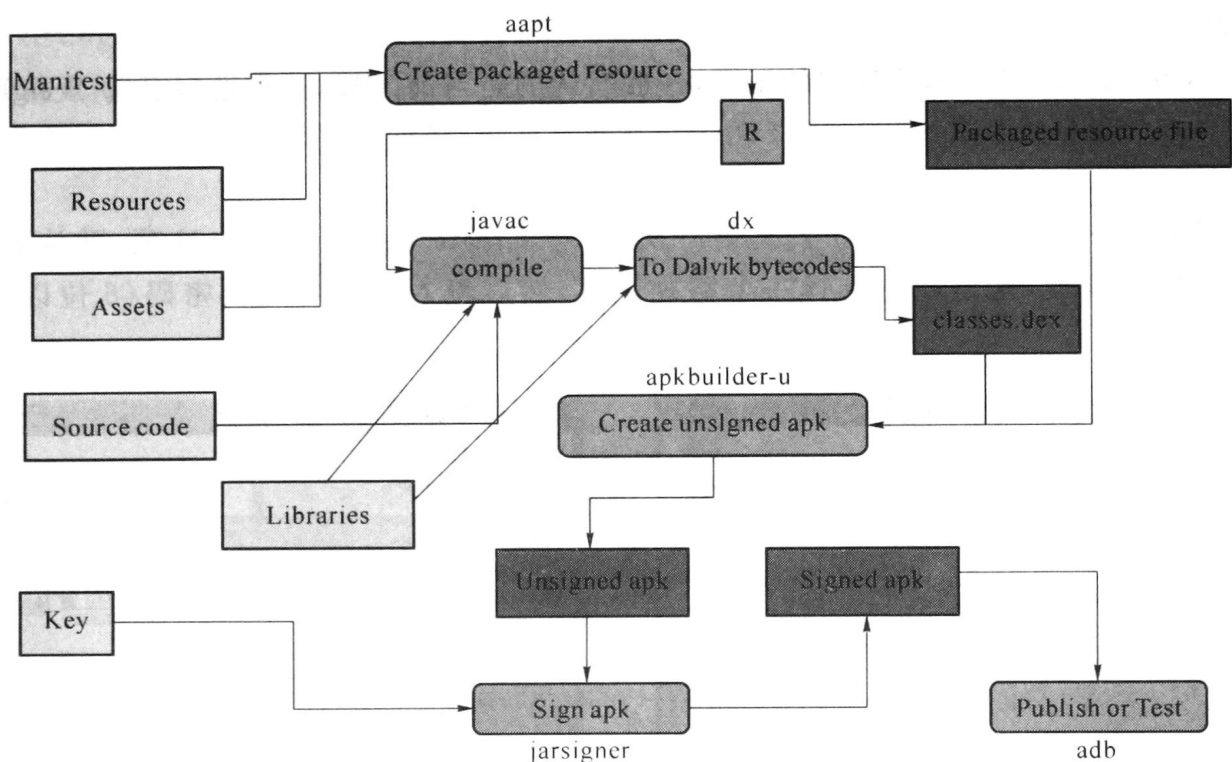


图 1-3 手动打包

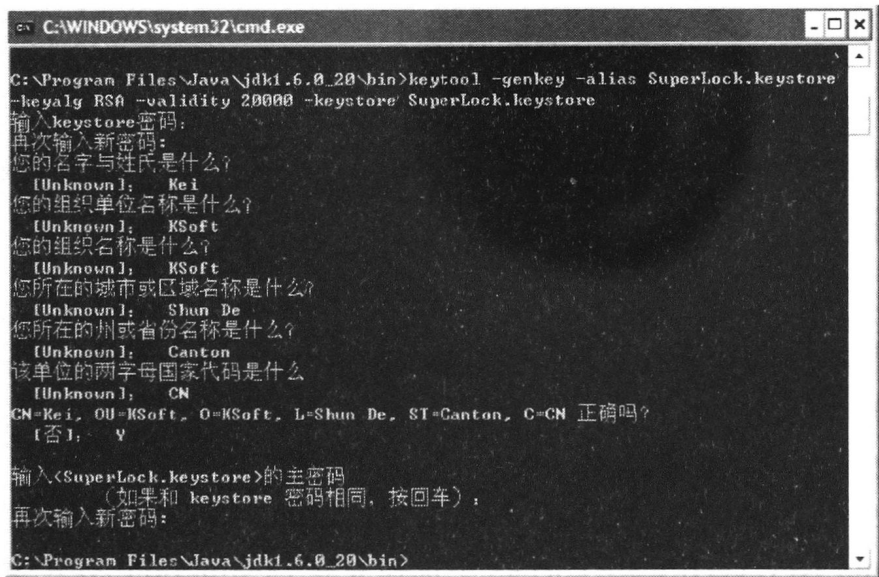


图 1-4 命令行打包

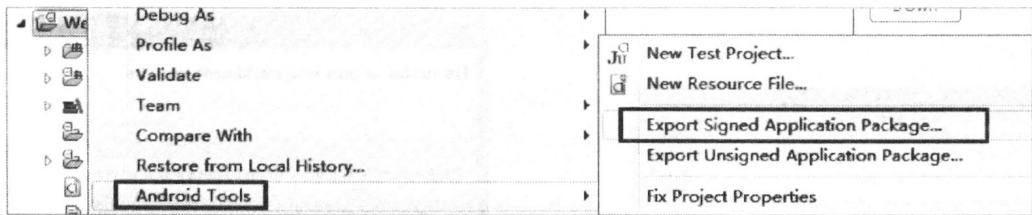


图 1-5 旧版本打包

① Use the Export Wizard

Android 1.5 SDK 以后,ADT 0.9.1 版自带的 Use the Export Wizard。首先在 Package Explorer 中选择工程的 AndroidManifest.xml 文件,可以看到右边默认的 manifest 模式中有个 exporting 功能,选择 Use the Export Wizard,如图 1-6 所示。

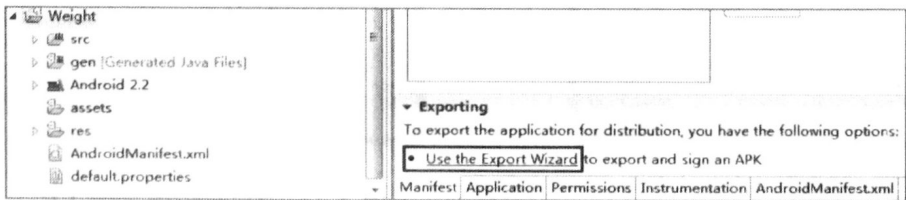


图 1-6 Use the Export Wizard

② Project Checks

在 Project Checks 中,选择需要导出的工程,一般默认的是当前工程,这里选择名为 Weight 的工程,如图 1-7 所示。

③ Keystore selection

接下来在 Keystore selection 中,如果是第一次,选择 Create new keystore 这项,默认的 Location 为 keystore 文件的保存位置,这里随意选择一个路径即可,然后输入密码和确认密码(6 位以上),如图 1-8 所示。

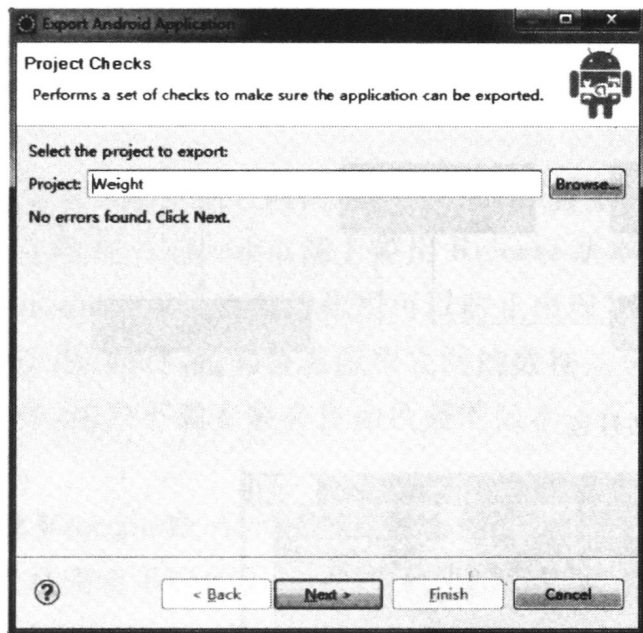


图 1-7 Project Checks



图 1-8 Keystore selection

④ Key Creation

在 Key Creation 选项中,需要输入一些简单信息,比如 Alias(别名),这里就以 andy 为例,密码要和前面 keystore 中输入的一样,整个签名过程其实是一个 RSA 加密过程,最后的 Validity(years)是有效期,这里输入推荐的 25 年即可,其他的内容为选填,如图 1-9 所示。

⑤ Destination and Key/certificate checks

最后在 Destination and Key/certificate checks 中选择只有 Destination APK 文件,这是保存的最终 apk 文件的路径,最后签名后的 apk 文件就保存到这个位置中,同时下面有一些描述信息,如图 1-10 所示。



图 1-9 Key Creation

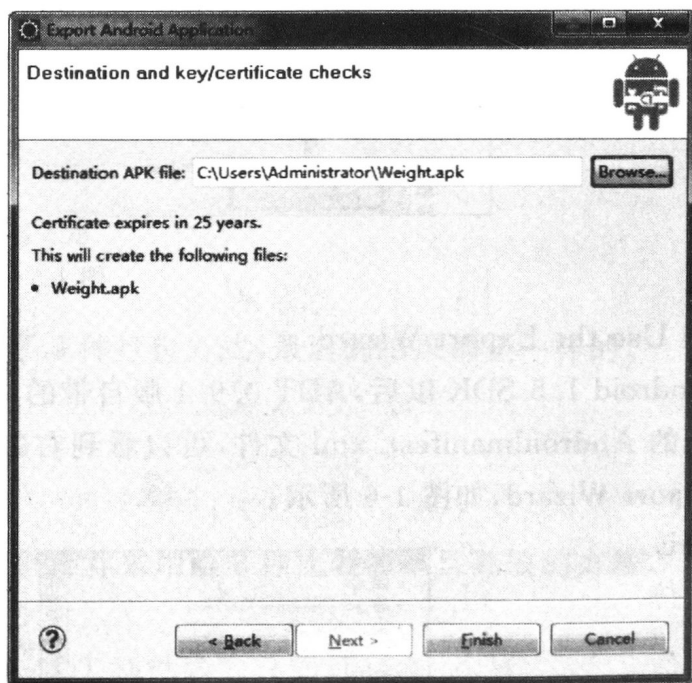


图 1-10 Destination and Key/certificate checks

⑥ Use existing keystore

这里操作就全部完成了,在相应的目录下就生成对应的 Release 版本的 apk 程序。如果以后需要对程序升级,或者发布新的程序,就不需要这么复杂,直接选择已有的 Use existing keystore 进行签名,如图 1-11 和图 1-12 所示。



图 1-11 选择已经存在的 keystore

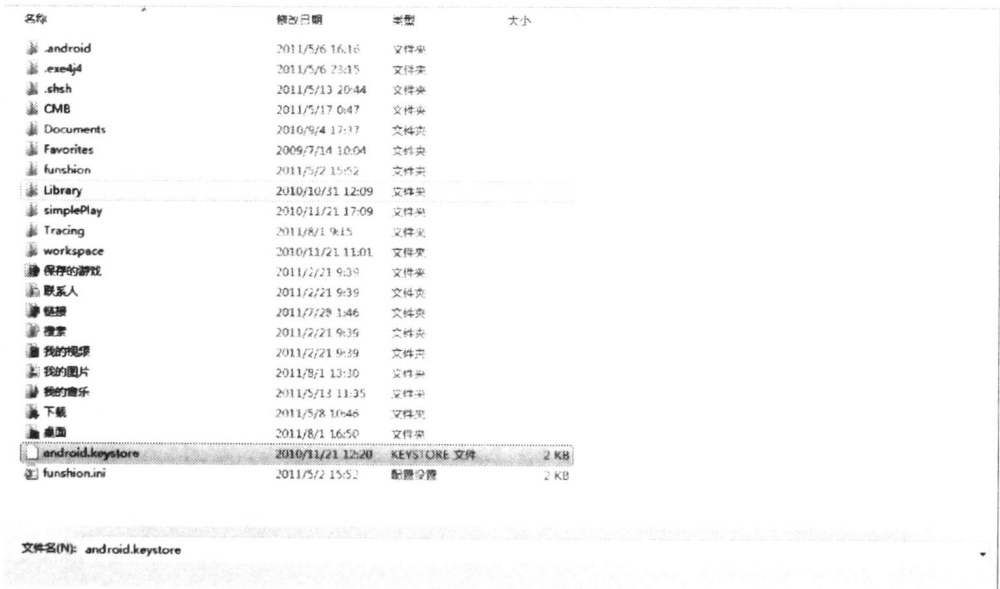


图 1-12 keystore 存放位置

⑦ Key alias selection

选择好相应的 existing key,如图 1-13 所示。成功生成 apk,如图 1-14 所示。

(2) 反编译

反编译有时候就像一把双刃剑,试想程序员辛辛苦苦开发的程序,过了几天在市场上出现了类似的山寨程序,通常会令软件开发者厌恶。

反过来说,不可能每一个 Android 的初学者一开始都能马上独立开发出一个较好的程序,所有的学习都是从模仿开始,都希望能够有机会学习到优秀软件的精髓,达到提高。就像练习书法的人,都希望一睹大师的真迹,加以模仿达到提高。各种各样的编程语言都有对应的反编译工具,当然 Android 也不例外。

【知识点】什么是反编译?

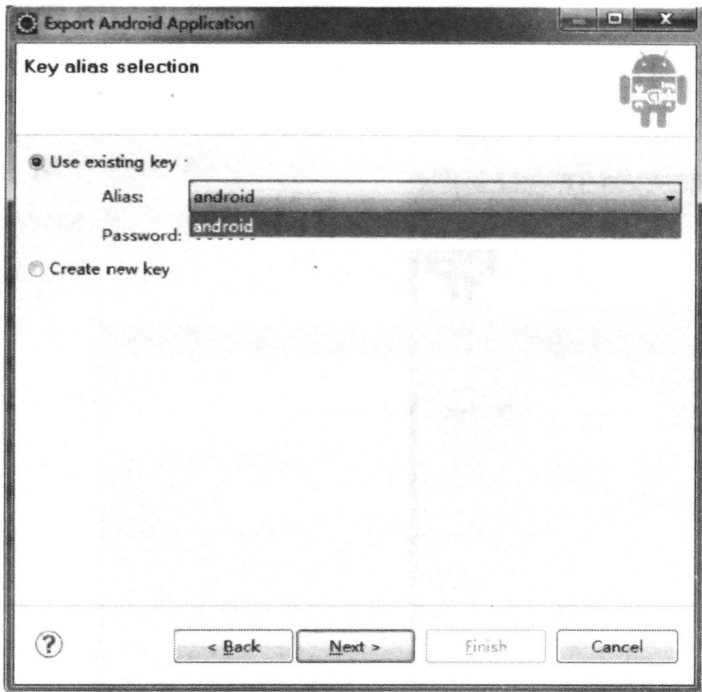


图 1-13 Use existing key

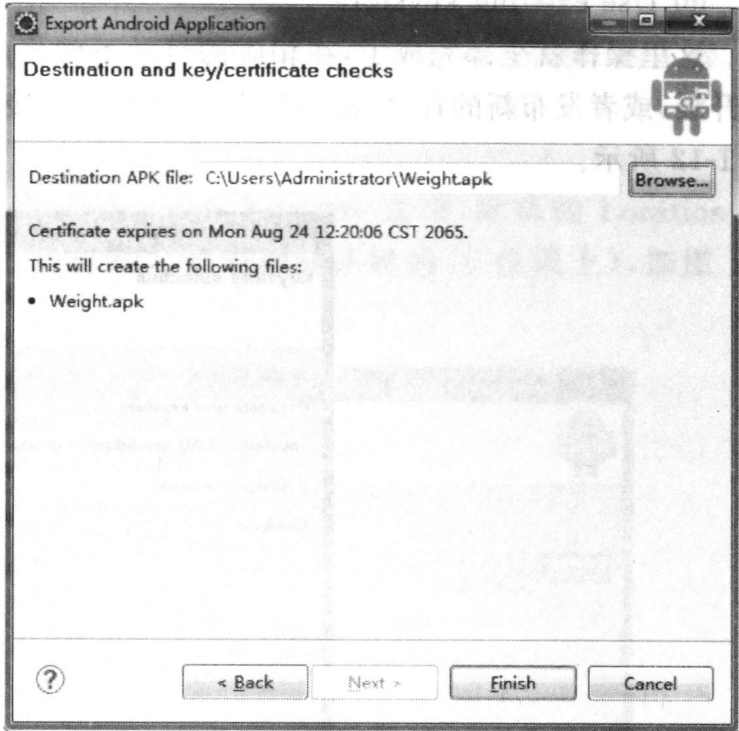


图 1-14 生成 apk

高级语言源程序经过编译变成可执行文件，反编译就是逆过程。但是通常不能把可执行文件变成高级语言源代码，只能转换成汇编程序。

计算机软件反向工程(Reverse engineering)也称为计算机软件还原工程，是指通过对他人软件的目标程序(可执行程序)进行“逆向分析、研究”工作，以推导出他人的软件产品所使用的思路、原理、结构、算法、处理过程、运行方法等设计要素，作为自己开发软件时的参考。

简单地 Java 为例,Java 一直以跨平台著称，一次编译处处运行。利用 javac 命令编译出来的 class 文件，用类似记事本的软件打开，显示的结果好像乱码，是看不明白的，如图 1-15 所示。因为它已经由高级语言编译成中间代码，其目的可以运行在不同平台的 Java 虚拟机上，从而达到控制底层硬件的目的。这个编译的结果，即图 1-15 中的 class 文件无论在哪个平台执行结果都是一样的，但是用不同的虚拟机去执行它就可以运行在不同平台的硬件上。举个形象的例子，这就好像一个平日生活中计算机的读卡器，编译的 class 文件就好比是 USB 接口，是统一的，任何计算机都可以用，而同一个读卡器可以读 TF 卡、SD 卡，就好比不同的平台，通过 USB 接口可以执行这些不同平台的数据，这就是跨平台的原理。而反编译的目的就是将图 1-15 中的乱码还原成原始的高级语言。



图 1-15 class 文件

① 需求分析

目前来说 Google Android 平台选择了 Java Dalvik VM 的方式,使其程序很容易破解和被修改,首先 apk 文件其实就是一个 MIME 为 ZIP 的压缩包,读者可以修改 zip 后缀名方式可以看到内部的文件结构,类似 Sun JavaMe 的 Jar 压缩格式一样,不过比较区别的是Android上的二进制代码被编译成为 dex 的字节码,所有的 Java 文件最终会编译进该文件中,作为托管代码既然虚拟机可以识别,那么就可以很轻松地反编译。所有的类调用涉及的方法都在里面体现到,至于逻辑的执行可以通过实时调试的方法来查看,当然这需要借助一些第三方编写的跟踪程序。Google 最后在 Android Market 上设置了权限保护 app-private 文件夹的安全,但是最终读者使用修改定值的系统仍然可以获取到需要的文件。

反编译所需工具:

- AXMLPrinter2.jar。
- dex2jar。
- 查看 Jar 包的 GUI 工具。

② 功能实现

- 解压缩

依旧以上面打包好的 Weight.apk 程序进行反编译,用自己的案例进行反编译,更有对比性,方便学习。

首先把.apk 文件改名为.zip,然后将其解压缩。找到解压缩包中的 classes.dex 文件,如图 1-16 所示。

classes.dex 就是 Java 文件编译再通过 Dalvik 虚拟机提供的 dx 工具打包而成的(和 Java 相比,为什么这么多类,却只有一个文件,请参考本章最后的常见问题)。接下来,就用上述提到的 2 个工具来逆向导出 Java 源文件。

- 生成 Jar 包

将 classes.dex 复制到工具 dex2jar.bat 所在目录,如图 1-17 所示。



图 1-16 核心程序



图 1-17 copy 文件

Windows 下用命令行模式下定位到 dex2jar.bat 所在目录运行:dex2jar.bat(空格)classes.dex,生成 classes.dex.dex2jar.jar,如图 1-18 和图 1-19 所示。

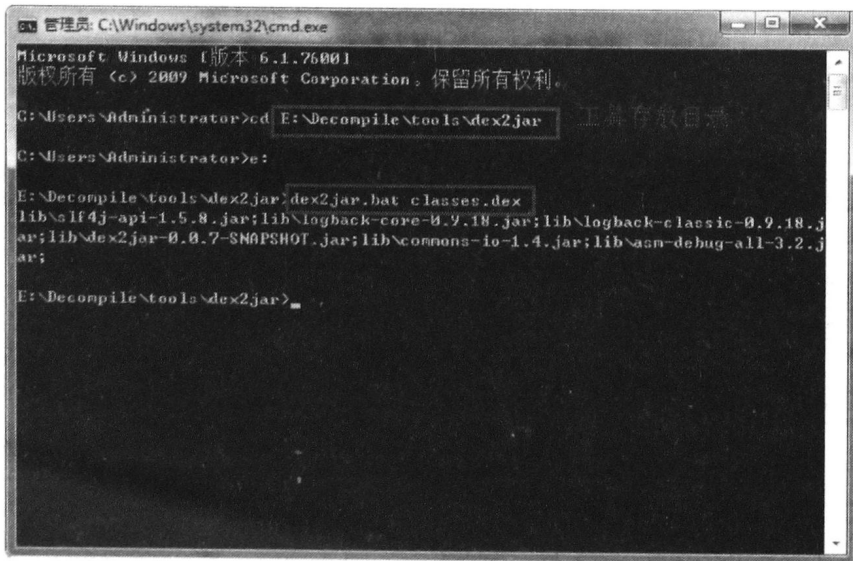


图 1-18 反编译 Jar 包



图 1-19 生成 Jar 包

• JD-GUI 查看源码

JD-GUI 是绿色软件,用它打开上面的 classes.dex.dex2jar.jar 文件,即可看到源代码,如图 1-20 所示。

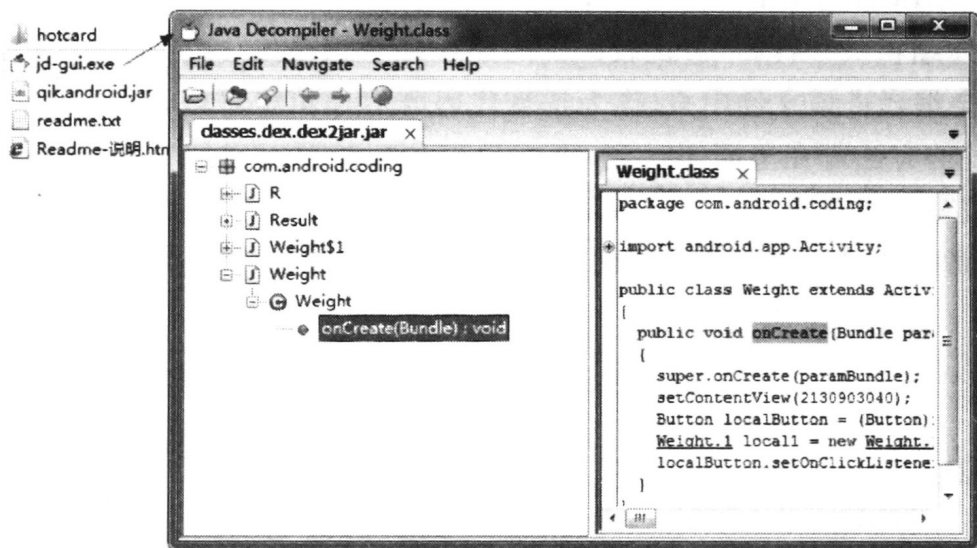


图 1-20 反编译成功

【知识点】“不是每一个程序都可以被反编译”和“不是每一个反编译程序都能看懂”,这两句话该如何理解呢?

首先,很多大公司放在市场上的程序已经用加密算法加密过了,很可能反编译结果是空的。

其实不要奢望反编译的程序和源程序是一模一样的。可以从图 1-21 中看出一些端倪,右边的树状目录为类名,和源程序的类比较一致(\$表示内部类)。

但仔细观察图 1-22 和图 1-23,发现和原来的程序大相径庭。观察图 1-24 不免发现,反编译程序将每一个内部类都编译成一个单独的类。并且布局文件中每一个组件都是有 ID 的,而反编译之后所有的 ID 都是十进制数字。

【小技巧】本例反编译后的结果是很容易理解的,原因是这个例子前面已经做过,知道主文件是哪一个,而且重点是代码很少。一个复杂程序反编译后,会有几百个类,每个类里面有上千行的代码,其实根本无法下手。

这时候需要先找到对应 main.xml 文件的类。可以利用 JD-GUI 中的查找功能找到 setContentView。然后在一步步地利用反编译的信息往下查找,例如图 1-22 中 setContentView(2130903040),在利用 2130903040 就找到 layout 布局文件夹,如图 1-25 所示,然后又得到了另外布局文件 result 的代码,依此类推。



图 1-21 对比

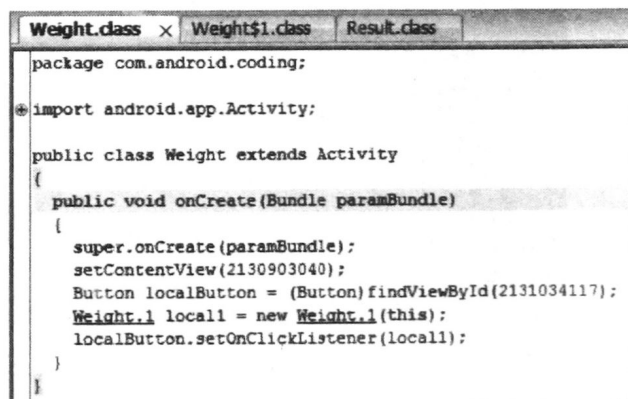


图 1-22 反编译程序

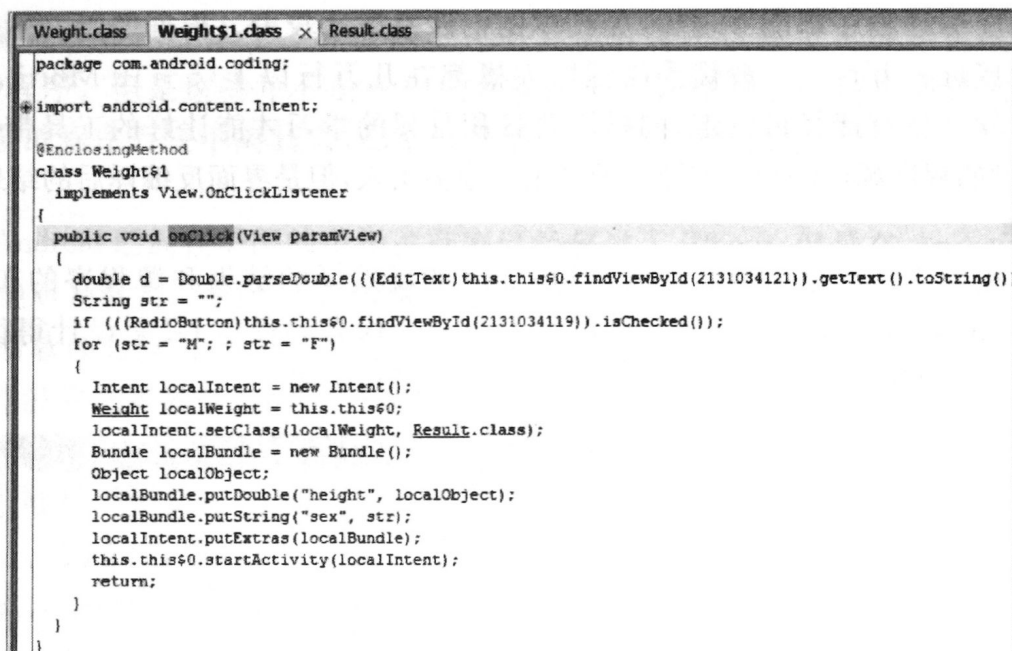


图 1-23 反编译内部类

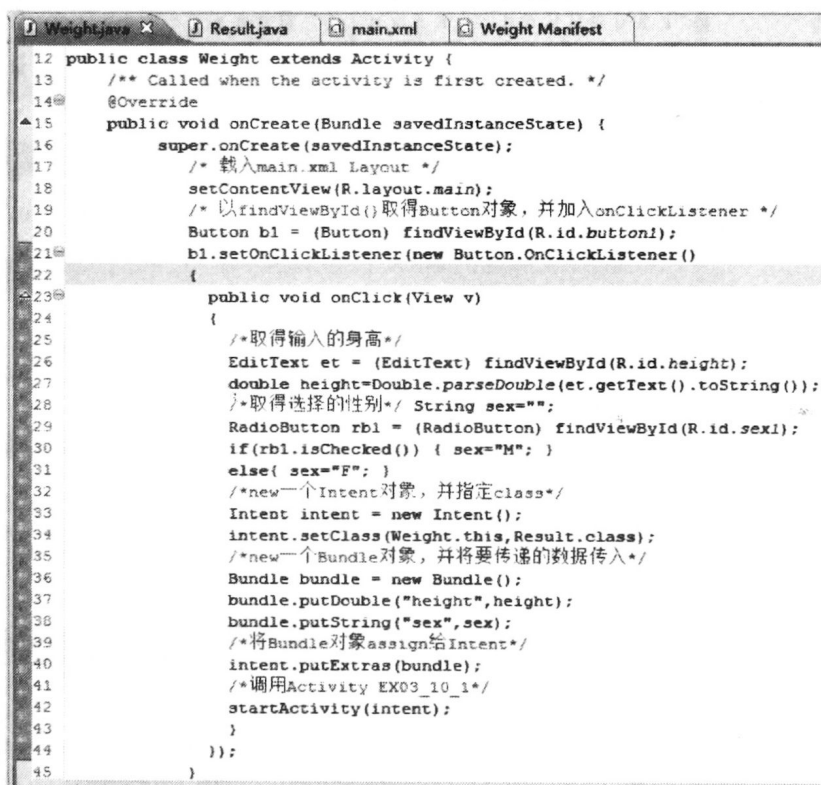


图 1-24 源程序

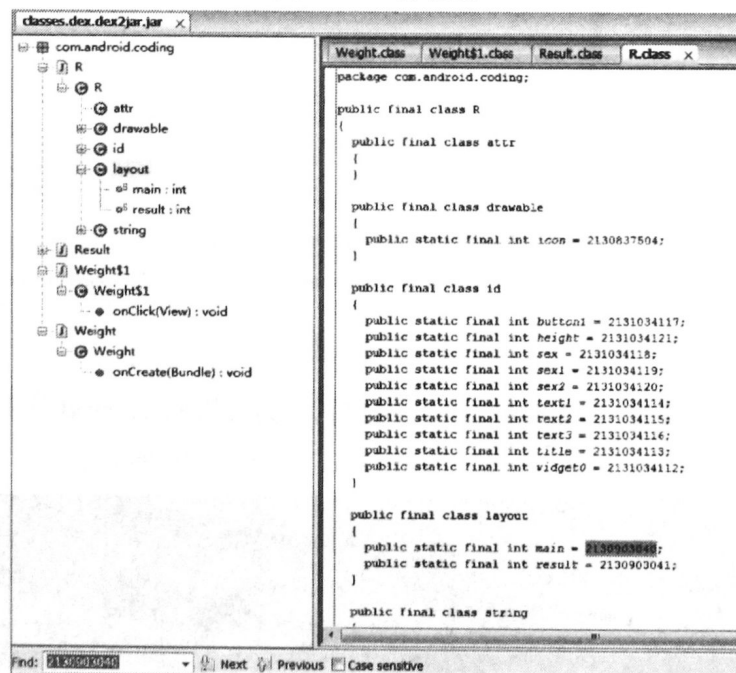


图 1-25 反编译查找

所以并非所有的反编译程序都能够理解,况且本例中涉及类比较少。根据经验一般来说超过 1 Mbit 的项目(无图片)代码量接近一万行。一般优秀的程序大概都在几万行以上(5~10 Mbit),游戏就更多。所以不要以为通过反编译软件就有捷径可以走,同样需要日积月累的学习才能让好的工具带来锦上添花的效果。

【小技巧】虽然大型的程序源代码和反编译后的结果有很多出入,但是界面反编译后的结果还是比较相似的。

• xml 的反编译

虽然功能代码反编译学习起来需要点功底,但是 xml 的反编译的结果和源程序的布局就比较接近。根据经验,从反编译学习最多内容多半是布局文件的学习。一个优秀的程序 UI 是设计得很精巧的。

而直接打开刚刚解压缩的 xml 文件,如图 1-26 所示,全部是乱码。

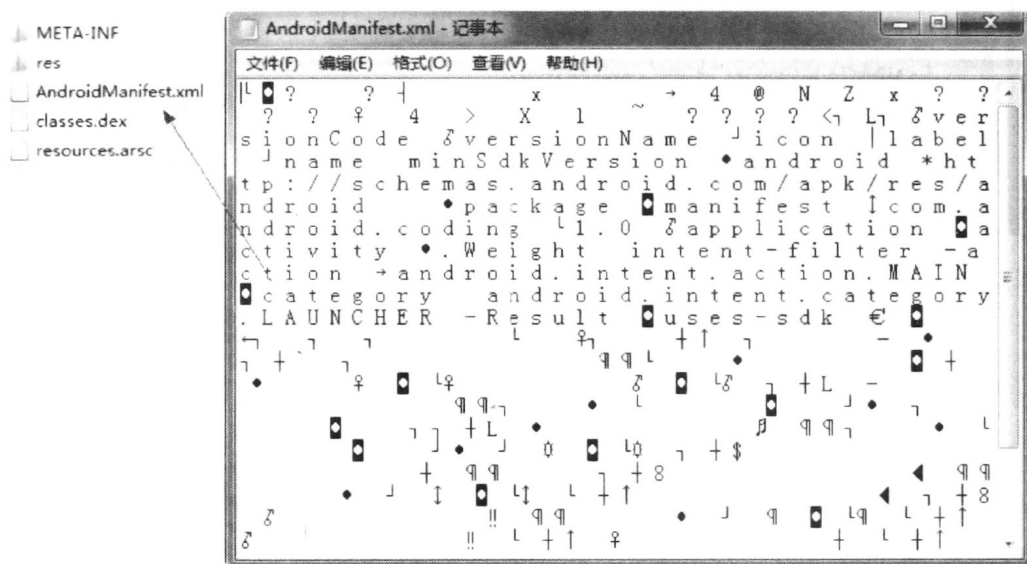


图 1-26 乱码

用 AXMLPrinter2.jar 查看 apk 中的布局 xml 文件。

方法一:把 AXMLPrinter2.jar 复制到 C 盘,在控制台 cd 到要解压出来的 apk 文件夹,执行下面的命令就可以把所有的.xml 还原文本格式了,如图 1-27 所示。

```
for/r. %a in( *.xml)do @java -jar c:\AXMLPrinter2.jar "%a">>"%a".txt
```

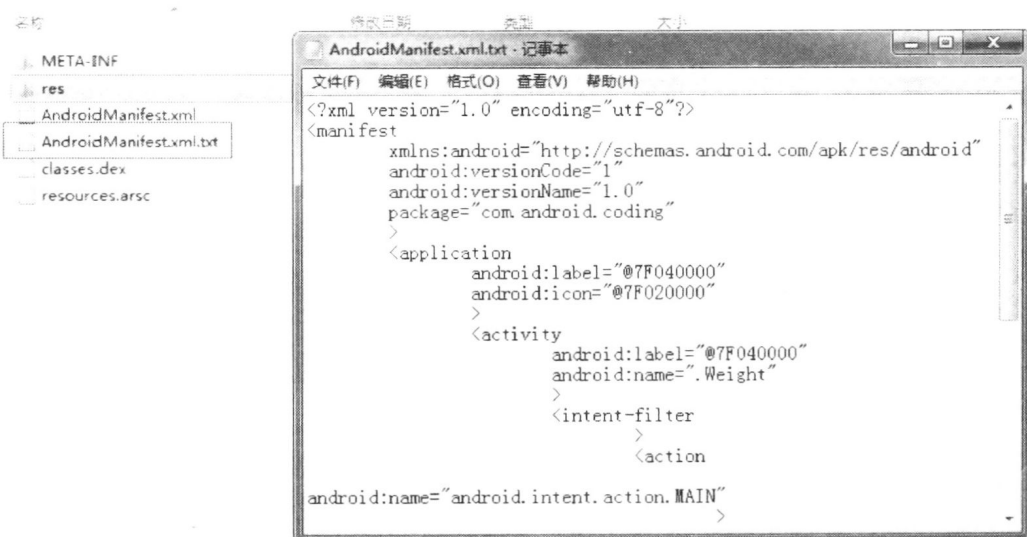


图 1-27 方法一

方法二:如果不想全部还原,而有针对性地查看 xml 文件。也可以打开 cmd 终端,一直进入到 tools 目录下,输入如下命令:java -jar AXMLPrinter2.jar main.xml>main.txt.,如图 1-28 所示。

```
Microsoft Windows [版本 6.1.7600]
版权所有 (c) 2009 Microsoft Corporation. 保留所有权利。

C:\Users\frankie>cd D:\Android\eclipse-for-android-win32\sdk\tools
D:\Android\eclipse-for-android-win32\sdk\tools>java -jar AXMLPrinter2.jar main.xml > main.txt
```

图 1-28 方法二

至此反编译的流程就介绍完毕了。互联网上还有其他软件可以通过反编译更改其他软件的图片、文字,即可以生产新的程序。但是这些并不是一个优秀程序员应该做的。任何事还是需要一步一个脚印,多看多吸收。所有复杂的程序万变不离其宗,都是由很基础的内容组合而成。

(3) 植入广告

目前为止尽管 Android 已经在美国的市场占有率已经超过了 iPhone,但是在 Android Market 赚钱不如在 App Store 上容易这绝对是事实。原因很多,互联网上也较多分析的原因,有兴趣的读者可以参考。从数字上面来看,iPhone 的付费软件卖得比 Android Market 好,很大程度和国外用户使用正版软件的习惯有关。

如果用户的软件没有金蝶软件“随手记”功能丰富,能够切入理财市场,建议不要采用付费下载。那么在 Android Market 的盈利通常有几种方式:企业外包(定制服务)、植入广告、产品平台。作为个人开发初学者通常是通过程序中植入广告达到受益效果。

那么 Android Market 目前为止并不支持大陆地区上传,但是国内也有很多类似于 Android Market 的市场,例如“有米”、“eoe 优亿”、“wooboo”、“架势”等移动传媒的门户网站,给移动终端的开发爱好者提供了免费的平台。下面就介绍一下如何在已有的程序中植入广告,因为 apk 文件中植入广告,并非像在 Web 网站中植入广告那样简单。演示以 wooboo 平台为例,如图 1-29 所示,其他平台的广告植入大同小异。

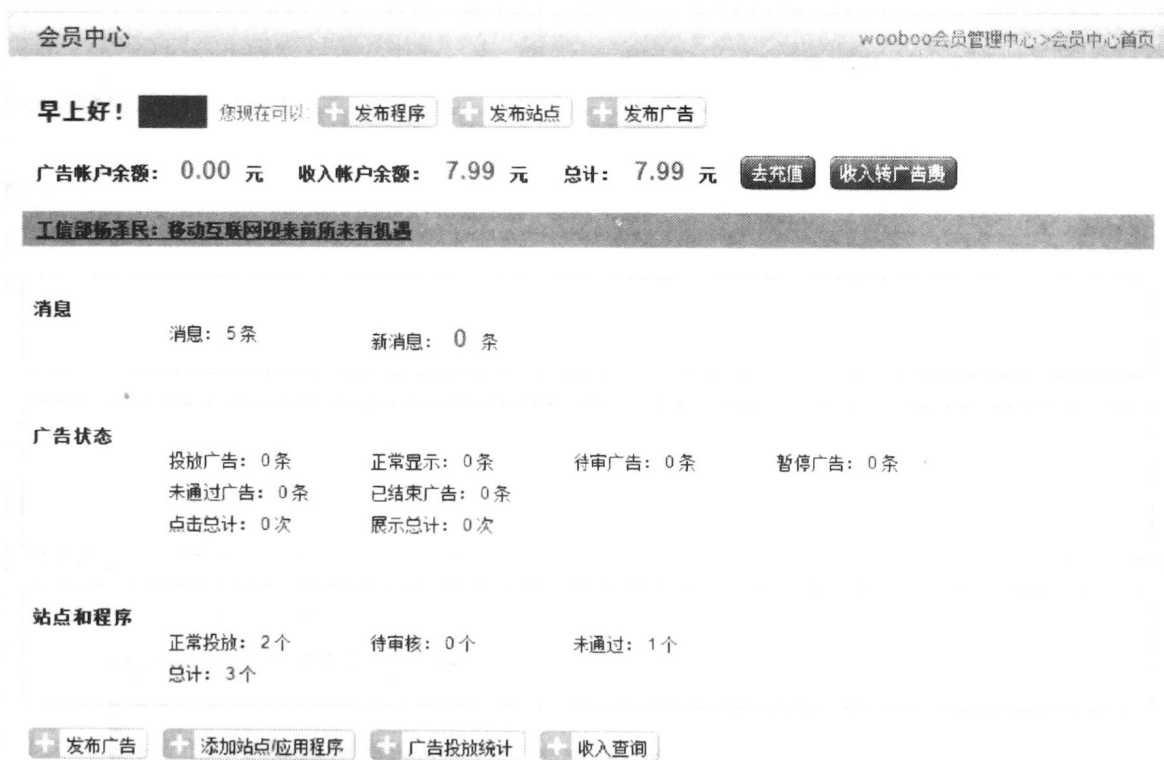


图 1-29 wooboo

① 添加访问权限

如图 1-30 方框 2 所示,植入广告必须有访问网络和得到电话信息的权限,所以需要添加 21、22 行代码。值得一提的是 READ_PHONE_STATE,不光只有访问广告才需要,例如返回设备的电话号码、返回注册的网络运营商的名字、返回 SIM 卡运营商的国家代码都需要此权限。

② 申请 ID

当在开源平台注册好账号为程序申请好 ID 后,就需要在 manifest 里面填入注册信息,如图 1-30 方框 1 所示。

简单来说,当其他用户使用了用户的程序,程序就会报告 wooboo 服务器:ID 为 023f129c6a70413f8 23b394b0a52a7fa 程序的广告被他人阅读或者点击,广告费才会充入申请的账号。

③ 广告参数

在 res/values 目录下面新建一个 attrs.xml 文件,如图 1-31 所示。此文件是配置广告的参数,代码如图 1-32 所示。