

移动开发系列丛书

eeeAndroid 51CTO

ZD.NET 机锋网 博客园

鼎力推荐



Google Android 应用开发与系统改造实战

王保卫 申波 编著

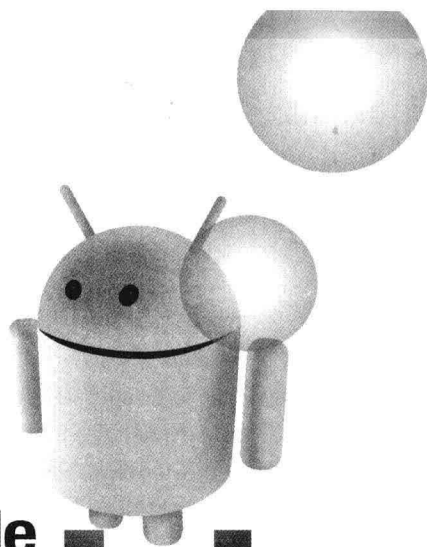
- 详细讲解改造 Android 系统的实例，如状态栏定制、开机动画、系统服务、系统应用改造
- 深入剖析如何编译 Android 源程序工程及 Android 编译系统原理
- 全面讲解了 Android 系统框架层各个部分的基本原理，如系统架构、系统服务模型、启动过程、图形系统、蓝牙系统、电话系统、多媒体系统、电源管理及系统通信机制等



人民邮电出版社
POSTS & TELECOM PRESS



移动开发系列丛书



Google Android

应用开发与系统改造实战

王保卫 申波 编著

人民邮电出版社
北京

图书在版编目 (C I P) 数据

Android应用开发与系统改造实战 / 王保卫, 申波编
著. — 北京: 人民邮电出版社, 2012. 2
ISBN 978-7-115-27272-0

I. ①A… II. ①王… ②申… III. ①移动电话机—应
用程序—程序设计 IV. ①TN929.53

中国版本图书馆CIP数据核字(2011)第262934号

内 容 提 要

本书共分 25 章, 对 Android 系统的各个层面进行了详细讲解, 旨在让读者在尽量短的时间内对 Android 系统的各个方面有一个全面的了解, 为进一步学习开发和研究 Android 操作系统源程序打下坚实的基础。首先, 在 Android 应用程序层面, 详细讲解了应用程序开发的各项技术, 着重讲解了应用程序的开发基础、应用程序的结构、4 大组件工作原理与功能, 以及它们之间通信的基础 Intent 类。此外, 给出了一些实例让读者能够更深刻地理解这些知识并加以应用。然后, 讲解了 Android NDK 开发的方方面面, 为了更好地开发出高质量的应用程序, 详细讲解了 Android 调试技术, 包括普通 Android 应用程序和 NDK 应用程序调试。

当然, 为满足一些有着丰富应用程序开发经验的读者和对 Android 系统底层有很大兴趣的读者的学习需求, 本书还详细讲解了如何编译 Android 源程序工程, 并对 Android 编译系统进行了深入剖析, 让读者对 Android 工程的高效组织和自动编译有更深刻的理解。此外, 本书还结合着源程序深入讲解了 Android 系统中的某些子系统, 包括子系统的功能、结构和工作原理。

本书还着重讲解了 Android 系统改造的思路, 详细讲解了一些改造 Android 系统的实例, 如状态栏定制、开机动画、系统服务、系统应用改造, 使读者通过动手实践来真正将所学知识融会贯通。

本书适合作为 Android 应用程序开发者的实践教材, 也适合对 Android 系统原理有极大兴趣的爱好者阅读, 还可供 Android 系统改造人员作为参考书使用。

Android 应用开发与系统改造实战

-
- ◆ 编 著 王保卫 申 波
责任编辑 张 涛
 - ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号
邮编 100061 电子邮件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
三河市潮河印业有限公司印刷
 - ◆ 开本: 800×1000 1/16
印张: 29
字数: 675 千字 2012 年 2 月第 1 版
印数: 1—4 000 册 2012 年 2 月河北第 1 次印刷

ISBN 978-7-115-27272-0

定价: 59.00 元

读者服务热线: (010)67132692 印装质量热线: (010)67129223

反盗版热线: (010)67171154

广告经营许可证: 京崇工商广字第 0021 号

前 言

随着 Android 系统版本的不断升级，给很多学习以及开发的人带来很多的困扰，目前为止，虽然系统依然在有序升级，但是系统已经趋于成熟，只是做了少许的性能改进以及 UI 界面的更换。已经很少改动系统的 API，这对于我们的学习和开发带来了很大的好处。

从 2008 年初接触 Android 以来，一直从事 Android 方面的开发，对 Android 的发展和学习参杂着很多的感受，从学习系统编译到学习系统部分移植；从学习系统的架构到应用架构的开发，以及对系统各部分 API 的深入学习，让我们逐步地对 Android 系统有了全面的理解。目前，虽然已经有很多 Android 方面的书籍，但不能满足用户深入学习系统底层知识需要，以及着手去尝试着对系统做些改造的体验。所以此书应时而生。

本书概述

全书共 25 章，分为 4 个部分。

第一部分从第 1 章至第 7 章。主要从 Android 应用开发的角度来讲解 Android 基础知识的入门。这一部分主要介绍了开发环境的搭建、应用开发所遇到的基本概念、各种控件及其使用方法、应用程序结构、应用程序组件、应用数据存储、应用通信机制以及应用的开发实例，最后讲解了高性能应用开发所需要注意的事项。这一部分旨在让读者能够对 Android 系统有一个直观的理解，能够开发基本的应用程序。

第二部分从第 8 章至第 12 章。主要从 Android 调试的角度来讲解 Android 的基础工具以及 Android 的编译系统。这一部分主要介绍了 4 个方面，第一个方面是介绍 Android 应用开发的基本工具，如模拟器、adb 工具、Hierarchy Viewer、layoutopt、DDMS、aapt、sqlite3 以及 Traceview 等；第二个方面是介绍应用调试技术、系统源码的调试技术等；第三个方面主要介绍系统源码的编译系统以及编译过程；第四个方面介绍了 NDK 的编译调试技术。这一部分旨在让读者能够从 Android 系统开发的角度来理解系统的各个部分是怎么开发出来的，并且能够使用这些工具来进行应用开发和调试，并且能够自己去接触系统源程序，并对其中的各个部分进行编译，尝试亲手编译系统的体验。

第三部分从第 13 章至第 21 章。主要从 Android 系统框架层来讲解系统应用的各个部分的基本原理，主要包括系统架构、系统服务模型、启动过程、图形系统、蓝牙系统、电话系统、多媒体系统、电源管理以及系统通信机制等方面。旨在让读者能够从系统应用的角度来了解系统应用存在于虚拟机之上是怎么调用和工作的。了解了原生的系统应用，就可以编写自己的系统应用了。

第四部分从第 22 章至第 25 章。这一部分主要介绍了几个系统改造的实例，让读者能够以创造者的身份去创造自己的系统。从系统开机动画的设计到系统服务编写以及系统应用的开发，让读者能够看到其实对系统的改造并不是那么的深奥，完全可以自己按照这种方式在系统级别上去

做一些事情。

致谢

在写这本书的过程中，深深地感受到写书的不易，虽然参加过很多的项目，也自己开发过很多项目，对 Android 的理解也已经不浅，但是真正提笔写起来，总是感觉功底不够，笔力有限，很多知识不知道怎么去组织、去介绍才能让读者更加轻松地理解和学习。

这期间，非常感谢陈榕老师，在科泰华捷的那段日子，我接触到了 Android，也是在这个公司的时候，我们深入学习和理解了整个 Android 的框架基础知识以及应用开发。很感激裴喜龙老师和顾伟楠老师一直以来对我们的指导，还有很多同学对我们的帮助。

非常感谢人民邮电出版社的张编辑，一直默默地在支持着我们写作，才让我们有足够的勇气去组织材料编写本书。源程序下载网址：www.ptpress.com.cn。编辑联系邮箱：zhangtao@ptpress.com.cn。

编 者

目 录

第一部分 Android 应用程序开发

第 1 章	Android 开发环境	2
1.1	初识庐山真面目——Android 开发环境概述	2
1.2	Android 开发系统环境要求	2
1.2.1	操作系统配置	2
1.2.2	开发环境配置要求	2
1.3	Android 开发所需软件的下载	3
1.3.1	Eclipse	3
1.3.2	ADT	3
1.3.3	Android SDK	3
1.4	Android SDK 开发环境配置	9
1.4.1	安装已下载的软件	9
1.4.2	本地安装 ADT	10
1.4.3	网络安装 ADT	14
1.4.4	创建 AVD	15
1.4.5	新建工程 HelloWorld	18
1.4.6	运行 Android 工程	20
1.5	Android NDK 开发环境搭建	22
1.5.1	Android NDK 简介	22
1.5.2	开发环境配置	23
1.5.3	NDK 的实例开发	28
1.6	小结	32
第 2 章	Android 基本应用开发与解析	33
2.1	应用程序结构	33
2.1.1	应用程序目录结构	33
2.1.2	知其然,知其所以然——Hello Wrold 程序结构讲解	35
2.2	Android 资源系统 (Android resource system)	37
2.2.1	资源系统中的基本概念	37
2.2.2	Android 资源系统引用	39
2.3	Android 布局	46
2.3.1	线性布局 (LinearLayout)	46
2.3.2	相对布局 (RelativeLayout)	50
2.3.3	帧布局 (FrameLayout)	52
2.3.4	表格布局 (TableLayout)	53
2.3.5	绝对布局 (AbsoluteLayout)	55
2.4	Android ViewGroup	56
2.4.1	TabWidget 和 TabHost	57
2.4.2	TabWidget 和 TabHost 的应用	57
2.4.3	ListView (列表视图)	60
2.4.4	实现九宫图首选——GridView	64
2.5	Android View (视图)	66
2.5.1	文本框 (TextView)	66
2.5.2	AutoCompleteTextView	69
2.5.3	编辑框 (EditText)	70
2.5.4	下拉列表 (Spinner)	74
2.5.5	拖动条 (SeekBar)	75
2.5.6	评分条 (RatingBar)	77
2.5.7	按钮 (Button)	79
2.5.8	图片按钮 (ImageButton)	80
2.5.9	图片框 (ImageView)	82
2.5.10	画廊 (Gallery)	82
2.6	Android UI 事件处理	85
2.6.1	Android UI 概述	85
2.6.2	事件监听器和事件处理	85
2.6.3	监听器和事件处理实例	86
2.7	小结	88
第 3 章	Android 应用程序清单	89
3.1	应用程序结构	89
3.1.1	Manifest 文件作用	89
3.1.2	元素顺序问题	90
3.1.3	AndroidManifest.xml 的功能介绍	90

3.1.4	AndroidManifest.xml 的结构和规则	90	4.2	服务 (Service) 应用	122
3.1.5	结合实例综述说明	91	4.2.1	Service 概念及使用实例	122
3.2	Manifest 文件结构	92	4.2.2	Service 的生命周期	123
3.3	Manifest 文件中各个元素及属性介绍	92	4.2.3	Service 与 Activity 通信	125
3.3.1	<action>	92	4.2.4	Service 与 Activity 通信实例	125
3.3.2	<activity>	93	4.3	存储与访问	131
3.3.3	<activity-alias>	95	4.3.1	文件进行数据存储	131
3.3.4	<application>	95	4.3.2	SharedPreferences	134
3.3.5	<category>	97	4.3.3	使用 SQLite 数据库存储数据	137
3.3.6	<data>	97	4.3.4	内容提供者——Content provider	142
3.3.7	<grant-uri-permission>	98	4.4	广播 (Broadcast) 与接收 (Receiver)	147
3.3.8	<instrumentation>	99	4.4.1	概述	147
3.3.9	<intent-filter>	99	4.4.2	广播的生命周期	147
3.3.10	<manifest>	100	4.4.3	广播实例	147
3.3.11	<meta-data>	100	4.5	小结	149
3.3.12	<path-permission />	101	第 5 章	Android 应用层通信机制	150
3.3.13	<permission>	102	5.1	Intent 通信机制	150
3.3.14	<permission-group>	103	5.1.1	Intent 概述	150
3.3.15	<permission-tree>	103	5.1.2	Intent 对象	151
3.3.16	<provider>	104	5.1.3	Intent 数据传递 Bundle	153
3.3.17	<receiver>	106	5.1.4	Intent 过滤器——Intent filters	153
3.3.18	<service>	107	5.1.5	一般案例	157
3.3.19	<supports-screens>	108	5.1.6	如何利用 Intent 来匹配	157
3.3.20	<uses-configuration>	108	5.1.7	Intent 的实例	158
3.3.21	<uses-feature>	109	5.2	Handler 消息通信机制	158
3.3.22	<uses-library>	109	5.2.1	Handler 机制概述	158
3.3.23	<uses-permission>	110	5.2.2	Handler 发送消息的方法列表	159
3.3.24	<uses-sdk>	110	5.2.3	Handler 实例	159
3.4	Android permission 列表	110	5.3	小结	161
3.5	小结	114	第 6 章	综合案例——多线程下载器开发	162
第 4 章	Android 的 4 大组件	115	6.1	多线程下载概述	162
4.1	Activity 简介和应用实例	115	6.2	Android 多线程下载	162
4.1.1	Activity 简介	115	6.3	小结	169
4.1.2	Activity 的生命周期	115	第 7 章	Android 应用程序设计与优化	170
4.1.3	Activity 堆栈 (Stack)	117	7.1	UI 设计	170
4.1.4	Activity 使用实例	118			
4.1.5	多个 Activity 之间的数据传递	119			
4.1.6	Activity 的生命周期实例	122			

7.2	性能设计	171	7.4	无缝性设计	177
7.3	针对响应的设计	174	7.5	小结	180

第二部分 Android 调试技术与编译系统

第 8 章	Android 工具介绍	182	8.9.1	DDMS 工作原理	220
8.1	模拟器 Emulator 命令	182	8.9.2	启动 DDMS	221
8.2	Android 模拟器	199	8.9.3	DDMS 功能	222
8.2.1	启动和关闭模拟器	199	8.10	资源打包工具——aapt	227
8.2.2	操作模拟器	199	8.11	IDL 语言——aidl	227
8.2.3	模拟器启动选项	200	8.11.1	用 aidl 实现 IPC	228
8.2.4	使用模拟器控制台	200	8.11.2	调用的 IPC 方法	232
8.2.5	使用模拟器皮肤	203	8.12	sqlite3	236
8.2.6	运行多个模拟器实例	204	8.13	Traceview	236
8.2.7	在模拟器上安装应用程序	204	8.13.1	创建 Trace 文件	236
8.2.8	SD 卡模拟	204	8.13.2	将 Trace 文件复制到主机	237
8.2.9	故障排除	205	8.13.3	使用 Traceview 查看跟踪文件	237
8.2.10	模拟器的限制	205	8.13.4	Traceview 文件格式	238
8.3	adb	206	8.13.5	Traceview Known Issues	240
8.3.1	发出 adb 命令	206	8.13.6	dmtracedump 用法	240
8.3.2	查询模拟器/设备	207	8.14	mksdcard	241
8.3.3	向特定的模拟器/设备发送命令	207	8.15	bat 脚本——dx	242
8.3.4	安装软件	208	8.16	压力测试工具——Monkey	243
8.3.5	转发端口	208	8.16.1	Monkey 简介	243
8.3.6	从模拟器/设备中导入导出文件	208	8.16.2	Monkey 的基本用法	243
8.3.7	adb 命令列表	208	8.16.3	命令选项详解	244
8.3.8	启动 shell 命令	210	8.16.4	实例	245
8.3.9	启动 logcat	210	8.17	android 工具	246
8.4	ADT 插件	213	8.18	优化 APK 新工具——zipalign	246
8.5	Android 虚拟设备	213	8.19	小结	247
8.5.1	界面方式	214	第 9 章	调试技术	248
8.5.2	命令行方式	215	9.1	Android 应用程序调试	248
8.6	设计用户界面利器——Hierarchy Viewer	216	9.1.1	日志式调试	248
8.7	layoutopt	219	9.1.2	Eclipse 调试	249
8.8	Draw 9-patch	220	9.1.3	TraceView 跟踪	250
8.9	调试工具——DDMS	220	9.1.4	单元测试 (JUNIT)	252
			9.2	Web 应用程序调试	264

9.2.1	在 Android 浏览器中用 控制台 API	264	11.3	获取源代码	289
9.2.2	在 WebView 中用 控制台 API	265	11.4	编译源代码	290
9.3	NDK 调试	265	11.5	模块编译	292
9.3.1	日志式调试	266	11.6	编译 Android 内核	293
9.3.2	ndk-gdb 调试	266	11.7	编译问题	295
9.4	系统源代码调试	270	11.7.1	Git 工具详解	295
9.4.1	编译 Android 源代码	270	11.7.2	repo 工具详解	297
9.4.2	导入 Android 源代码工程	270	11.7.3	32 位操作系统无法编译 问题	298
9.4.3	调试程序	272	11.7.4	JDK 版本	298
9.4.4	调试说明	275	11.7.5	arm-eabi-4.4.3 版本问题	299
9.5	Android 程序调试原理	275	11.7.6	libOpenSLES.so 问题	299
9.6	小结	276	11.7.7	libclearsilver-jni.so 问题	300
第 10 章	Android 编译系统	277	11.7.8	LOCAL_MODULE_TAGS 问题	300
10.1	Android 编译系统概述	277	11.8	小结	300
10.2	编译系统入口	278	第 12 章	NDK 开发	302
10.3	Makefile 文件	278	12.1	NDK 开发概述	302
10.3.1	理解 Makefile 文件	278	12.2	Android.mk 语法规范	303
10.3.2	简单 APK 的 Makefile	279	12.2.1	NDK 提供的变量	304
10.3.3	使用 jar 文件的 APK 的 Makefile 文件	280	12.2.2	NDK 提供的宏	305
10.3.4	平台密钥签名的 APK 的 Makefile 文件	280	12.2.3	NDK 模块描述变量	306
10.3.5	特定厂商签名的 APK 的 Makefile 文件	280	12.3	Application.mk 语法规范	310
10.3.6	增加已编译好的 APK 的 Makefile 文件	281	12.4	导入模块功能	312
10.3.7	增加静态 Java 库	281	12.4.1	NDK_MODULE_PATH 变量	312
10.4	编译层次结构	282	12.4.2	编写导入模块	313
10.5	配置新产品的 Makefile	282	12.4.3	命名导入模块	313
10.5.1	配置步骤	282	12.4.4	一些建议	314
10.5.2	新产品的文件结构树	284	12.5	NDK 预编译功能	315
10.5.3	产品定义文件	284	12.5.1	声明预编译库模块	315
10.6	编译系统的结构	286	12.5.2	引用预编译模块	315
10.7	小结	287	12.5.3	导出预编译模块的头 文件	316
第 11 章	Android 系统编译环境搭建	288	12.5.4	调试预编译模块	316
11.1	系统要求	288	12.5.5	预编译模块的 ABI	316
11.2	安装工具	288	12.6	NDK 编译工具 ndk-build	317
			12.7	NDK 调试工具 ndk-gdb	318
			12.8	小结	321

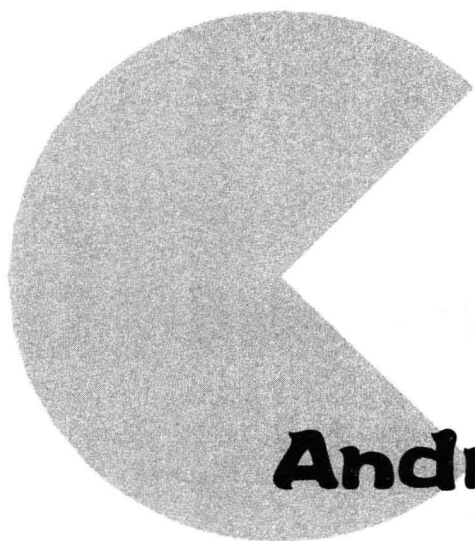
第三部分 Android 子系统分析

第 13 章 Android 系统架构	324	16.3 OpenGL ES 分析	355
13.1 Android 概念	324	16.4 Skia 图形库分析	357
13.2 Android 平台特性	324	16.5 SurfaceFlinger 服务	358
13.3 Android 架构	325	16.6 Surface 显示过程	360
13.3.1 Android 应用程序层	325	16.7 小结	364
13.3.2 Android 应用程序框架层	325	第 17 章 蓝牙系统	365
13.3.3 Android 程序库	326	17.1 蓝牙系统概述	365
13.3.4 Android 运行时库	326	17.2 蓝牙系统架构	365
13.3.5 Linux 内核	327	17.3 蓝牙系统源代码分析	366
13.4 Android 版本演化	327	17.3.1 蓝牙服务的启动和关闭	366
13.5 小结	329	17.3.2 蓝牙系统与蓝牙耳机的 连接	369
第 14 章 系统服务模型	330	17.4 移植和编译	371
14.1 系统服务模型概述	330	17.4.1 移植	371
14.2 Android 系统服务启动过程	332	17.4.2 编译	371
14.3 Android 系统服务注册	334	17.4.3 遇到的问题	372
14.4 Android 系统服务请求	335	17.4.4 工具	372
14.5 小结	335	17.5 蓝牙新特性	372
第 15 章 Android 启动过程	336	17.6 小结	374
15.1 Android 初始化语言	336	第 18 章 电话系统	375
15.1.1 Actions (行动)	336	18.1 电话系统概述	375
15.1.2 Services (服务)	337	18.2 Android 无线接口层	376
15.1.3 Options (选项)	337	18.2.1 Android 无线接口总述	376
15.1.4 Triggers (触发器)	337	18.2.2 RIL 初始化	377
15.1.5 Commands (命令)	338	18.2.3 RIL 交互	377
15.1.6 Properties (属性)	339	18.2.4 RIL 实现	379
15.1.7 initot.conf 实例	339	18.3 GSM 驱动模块	381
15.1.8 Android 调试记录	340	18.3.1 GSM 基本架构及初始化	381
15.2 Android 启动过程	341	18.3.2 请求流程	383
15.2.1 Android 概述	341	18.3.3 响应流程	385
15.2.2 Android 启动过程	341	18.4 电话和短信	387
15.2.3 init.rc 文件解析过程	345	18.5 小结	388
15.3 小结	353	第 19 章 多媒体系统	389
第 16 章 图形系统	354	19.1 多媒体概述	389
16.1 图形系统概述	354	19.2 多媒体系统架构	389
16.2 驱动程序接口之一 ——Framebuffer 分析	354	19.3 多媒体系统源代码分析	390

19.3.1 系统共享库架构及关系	390	20.3 Binder 源代码分析	405
19.3.2 系统框架重要头文件	392	20.3.1 Binder 源代码文件及其解析	405
19.3.3 MediaPlayer 分析	394	20.3.2 源代码分析	408
19.4 OpenCore 概述	399	20.4 小结	419
19.5 小结	400	第 21 章 电源管理	420
第 20 章 Binder 通信机制	401	21.1 电源管理概述	420
20.1 Binder 通信机制概述	401	21.2 电源管理源代码分析	422
20.2 Binder 通信机制工作原理	402	21.3 系统休眠与唤醒源代码分析	423
20.2.1 Binder 组织结构	402	21.4 小结	426
20.2.2 Binder 通信时序	402		
20.2.3 Binder 类继承关系	404		

第四部分 Android 系统改造实战

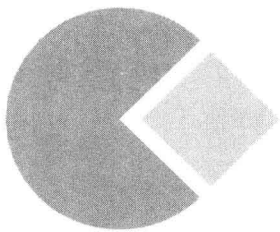
第 22 章 StatusBar 改造	428	23.6 小结	441
22.1 StatusBar 概述	428	第 24 章 系统服务改造指南	442
22.2 自定义 StatusBar 图标	428	24.1 自定义 Native 服务	442
22.2.1 制作图标	428	24.1.1 自定义服务	442
22.2.2 布局选择文件	428	24.1.2 注册服务	444
22.2.3 修改布局文件	429	24.1.3 调用服务	445
22.3 修改 Status Bar 图标默认值	429	24.1.4 运行测试	446
22.4 增加触摸事件	430	24.2 自定义 Android 服务	447
22.5 小结	432	24.2.1 自定义服务	447
第 23 章 开机动画改造	433	24.2.2 注册服务	449
23.1 开机动画概述	433	24.2.3 调用服务	450
23.2 开机图片	434	24.2.4 运行测试	450
23.3 开机文字	434	24.3 小结	451
23.4 开机动画	435	第 25 章 构建自己的系统应用	452
23.5 开机动画定制	437	25.1 系统应用的概述	452
23.5.1 制作 initlogo.rle	437	25.2 编写系统应用	452
23.5.2 修改开机文字	438	25.3 模块化编译系统应用	453
23.5.3 制作开机动画 bootanimation	440	25.4 运行系统应用	453
		25.5 小结	454



第一部分

Android 应用程序开发

- 第 1 章 Android 开发环境
- 第 2 章 Android 基本应用开发与解析
- 第 3 章 Android 应用程序清单
- 第 4 章 Android 的 4 大组件
- 第 5 章 Android 应用层通信机制
- 第 6 章 综合案例——多线程下载器开发
- 第 7 章 Android 应用程序设计与优化



第1章 Android 开发环境

1.1

初识庐山真面目——Android 开发环境概述

“巧妇难为无米之炊”，要想在 Android 平台上开发软件，必须装备好必备的工具，一个好的开发环境能够帮助我们快速地开发出想要的应用软件。伴随着 Android 的开放，Google 也提供了一套开发工具，包括设备模拟器、Android 资源打包工具 aapt、Dalvik 调试监控工具 DDMS，adb 调试桥和字节码转换工具 Dx 工具。

这些工具我们在开发基本应用时一般是用不着的，在后面的一些部分，会专门讲解 Android 系统中自带工具的用处。下面将进入 Android 开发环境搭建之旅。

1.2

Android 开发系统环境要求

1.2.1 操作系统配置

在你的计算机上具备以下其中的一个条件方能正常地开发 Android 程序。

- Windows XP (32-bit)、Windows Vista (32- or 64-bit) 或 Windows 7 (32- or 64-bit)。
- Mac OS X 10.5.8 或 later (x86 only)。
- Linux (Ubuntu Linux, Lucid Lynx)。GNU C 库 (glibc) 2.11 或者最新的。Ubuntu Linux、Lucid Lynx 新版本。

1.2.2 开发环境配置要求

(1) Eclipse 工具。

- Eclipse 3.4 (Ganymede) 或者更新版本。
- Eclipse JDT 插件 (包含在大多数的 Eclipse IDE 包中)。
- 如果你想安装或者更新 Eclipse，可以从下面的网址中下载：<http://www.eclipse.org/downloads/>。
- JDK 5 or JDK 6 (仅安装 JRE 是不够的)。
- Android Development Tools plugin (ADT)。

- 不适宜用 GNU 编译器来编译 Java。

- Apache Ant 1.8 或者最新版本。

(2) 注意。

如果 JDK 已经安装好了, 请确信它是满足上面所列的系统要求。特别要注意的是, 在 Linux 的某些版本的系统中, 使用的是 JDK1.4 或者 GNU 编译器来编译 Java, 这两种都不能支持 Android 开发环境。

1.3 Android 开发所需软件的下载

1.3.1 Eclipse

目前, Android 官方已经给出的最新 ADT 集成开发环境的插件已经开始支持 Eclipse 的 3.6 (Helios) 版本, 也可以使用 Eclipse3.4 或者 3.5 版本。Eclipse 的下载网址: <http://www.eclipse.org/downloads/>。

1.3.2 ADT

ADT 是 Eclipse 的一个插件, 全称为 Android Development Tools。是 Google 开发用来给 Android 开发人员开发 Android 应用软件的集成开发工具。你可以轻松地通过 ADT 工具来快速建立一个新的工程, 创建应用程序界面。还可以通过使用 Android SDK 工具来调试你的应用程序, 为应用程序签名。这些功能都会在今后的章节中介绍。下面通过网址下载 ADT。

ADT 下载网址: <http://developer.android.com/sdk/eclipse-adt.html>。

1.3.3 Android SDK

安装完整 Android SDK。(官网上下载的 SDK 只是一个框架, 并不是完整的开发包, 所以需要进一步下载完整) 安装步骤如下。

(1) 进入官方网址 <http://developer.android.com/sdk/index.html>, 如图 1.1 所示。

Platform	Package	Size	MD5 Checksum
Windows	android-sdk_r08-windows.zip	32696391 bytes	3e0b08ade5bfa9624bce9ddc164a48cb
	installer_r08-windows.exe (Recommended)	32746192 bytes	04ce87b10a8361a1f63cf2238bbc1ee3
Mac OS X (intel)	android-sdk_r08-mac_86.zip	28797617 bytes	d2e392c4e4680cbf2dfd6dbf82b662c7
Linux (i386)	android-sdk_r08-linux_86.tgz	26817291 bytes	3b626645b223d137d27beefbda0c94bc

图 1.1 SDK 安装包

选择第一个 android-sdk_r08-windows.zip 下载, 下载完成后解压, 会看到目录如图 1.2 所示。

(2) 运行 SDK Manager.exe 文件, 如图 1.3 所示, 弹出一个对话框, 显示进行下载 SDK 列表。

add-ons	2010/11/30 19:34	文件夹
platforms	2010/11/30 19:34	文件夹
tools	2010/11/17 18:51	文件夹
SDK Manager.exe	2010/11/17 18:51	应用程序
SDK Readme.txt	2010/11/30 19:34	文本文档

图 1.2 SDK Manager 工具

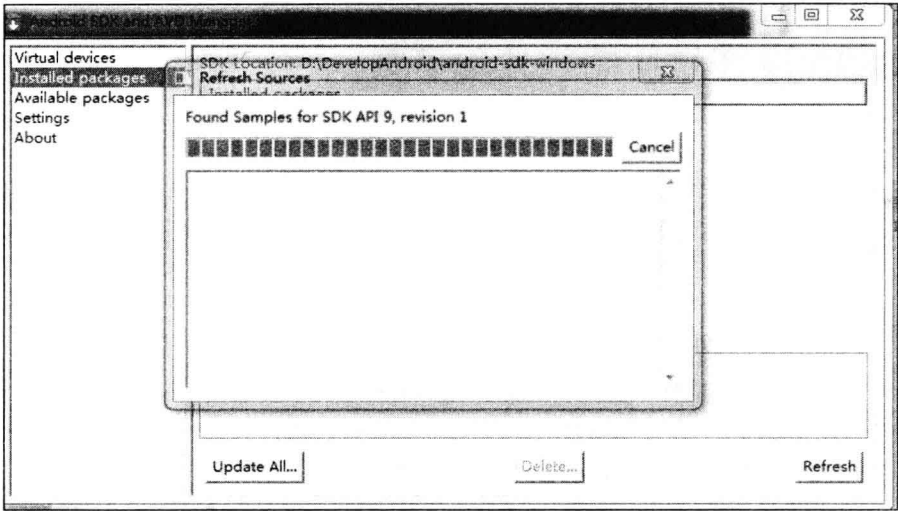


图 1.3 运行 SDK Manager

列表下载完成后会显示图 1.4 所示界面。

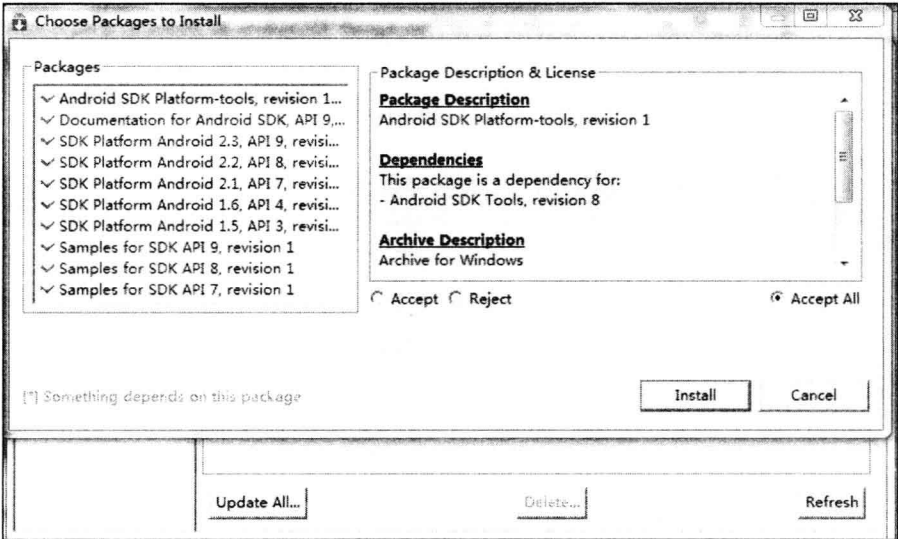


图 1.4 SDK 目录列表

选择 Accept All 项，并点击 Install 按钮，SDK Manager 就会下载所有的 SDK1.5 以上版本的 Android 以及 SDK 文档，如图 1.5 所示。

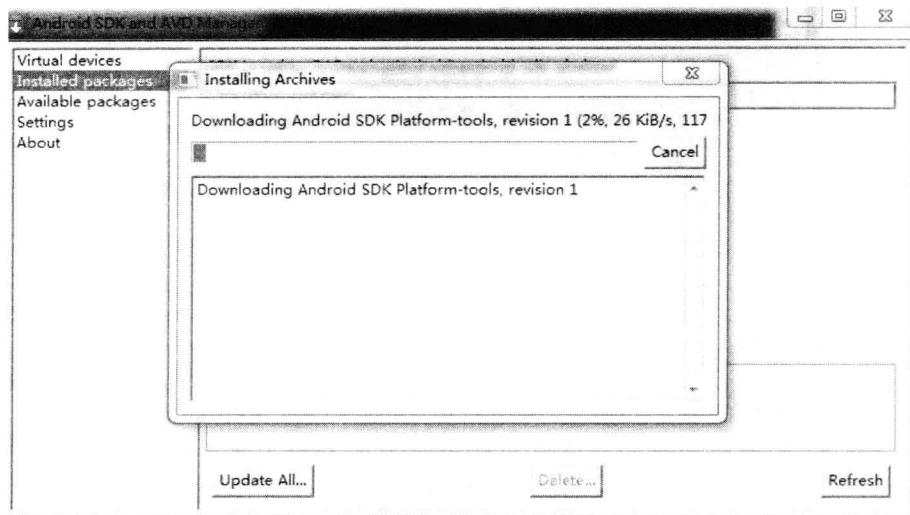


图 1.5 开始下载 SDK

(3) 安装完毕后，点击“Close”按钮，关闭窗口，进入你的 SDK 目录，其中多了 4 个文件夹，如图 1.6 所示。

文件夹 platforms 是下载的各个版本的 Android 系统 SDK。platform-tools 文件夹下载的是各种工具。docs 文件夹是下载的最新的 Android 开发文档。temp 文件夹是空的。

(4) 在 Eclipse 中，依次选中菜单 Window→Preferences 项，如图 1.7 所示。

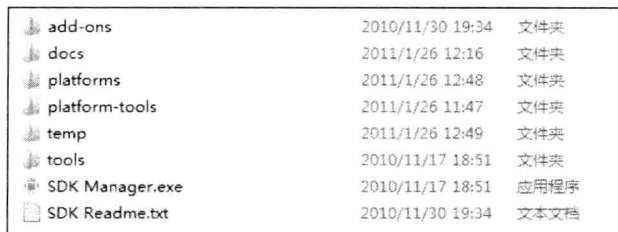


图 1.6 SDK 目录结构

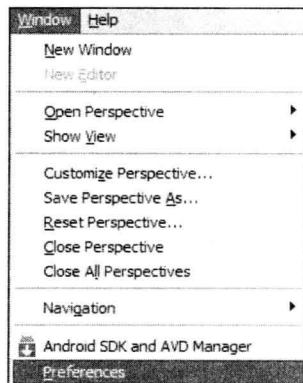


图 1.7 SDK 设置步骤 1

(5) 弹出“Preferences”对话框，可能会立即弹出“Android SDK Location”的错误对话框，如图 1.8 所示。

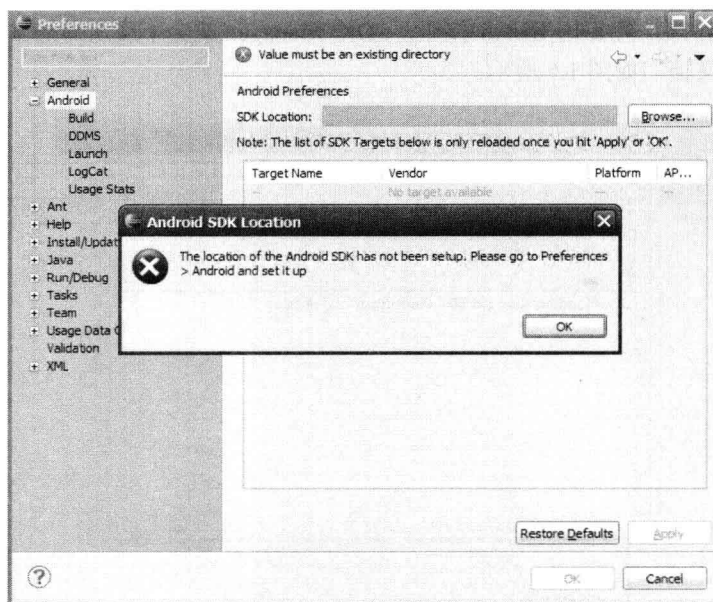


图 1.8 SDK 设置步骤 2

(6) 关掉上述错误对话框，在 SDK Location 中输入你的 SDK 文件所在的位置，这里是 E:\Android\android-sdk_r06-windows，点击“OK”按钮，如图 1.9 所示。

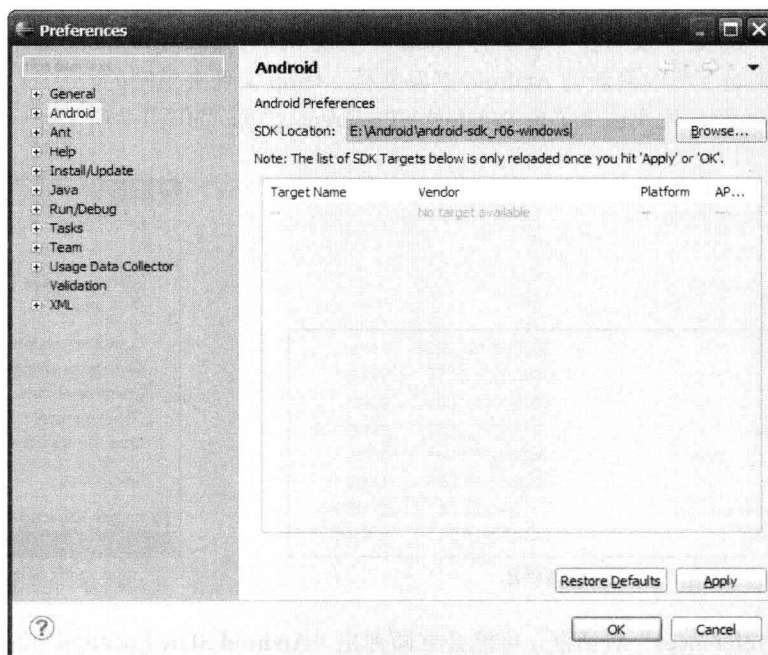


图 1.9 SDK 设置步骤 3