

TURING

图灵程序设计丛书



深入浅出 Google Android

E2ECloud 工作室 编著



人民邮电出版社
POSTS & TELECOM PRESS



TURING

图灵程序设计丛书

深入浅出 Google Android

E2ECloud 工作室 编著

人民邮电出版社
北 京

图书在版编目 (CIP) 数据

深入浅出 Google Android /E2ECloud 工作室编著. —北京: 人民邮电出版社, 2009.8
(图灵程序设计丛书)
ISBN 978-7-115-20069-3

I. 深… II. E… III. 移动通信-携带电话机-应用程序-程序设计 IV. TN929.53

中国版本图书馆CIP数据核字 (2009) 第114106号

内 容 提 要

本书基于 Google Android SDK 1.5 编著。在全方位讲解 Google 开放移动应用平台 Android 各种令人兴奋的特性的基础上, 全书以简单易懂的实例为依托, 深入探讨了应用程序的基本组件、界面布局, 通过 Intent 实现拨打电话和发送短信, 利用 Service 创建音乐播放器, 使用外部数据和服务, 处理本地数据, 发布和销售 Android 应用程序等内容。

本书适合有面向对象开发经验的读者阅读。此外, 本书还提供了配套的在线视频教程, 以便读者快速掌握 Android 开发。

图灵程序设计丛书

深入浅出Google Android

-
- ◆ 编 著 E2ECloud工作室
责任编辑 傅志红
执行编辑 陈兴璐
 - ◆ 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 电子函件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
北京隆昌伟业印刷有限公司印刷
 - ◆ 开本: 800×1000 1/16
印张: 19
字数: 449千字
印数: 1-4 000册

2009年8月第1版
2009年8月北京第1次印刷

ISBN 978-7-115-20069-3/TP

定价: 49.00元

读者服务热线: (010)51095186 印装质量热线: (010)67129223

反盗版热线: (010)67171154

序

Android一词的英文原义是“机器人”，而本书介绍的Android则是由开放手机联盟（Open Handset Alliance, OHA）于2007年11月5日发布的基于Linux的开源手机平台。该平台由操作系统、中间件、用户界面和应用软件组成，是首个为移动终端打造的真正开放和完整的移动软件平台。Android平台采用了WebKit浏览器引擎，具备触摸屏、高级图形显示和上网功能，用户能够在手机上查看电子邮件、搜索网址和观看视频节目等，同时Android还具有比iPhone等其他手机更强的搜索功能，可以说是一种融入全部Web应用的平台。根据市场研究机构信息通信与媒体研究公司Informa的预测，Google Android手机平台将在三年内超越苹果iPhone操作系统。我们完全相信未来Android的发展前景必将越来越广阔。

为了帮助国内的开发人员能快速掌握Android开发，基于Google于2009年4月15日最新发布的Android SDK 1.5，我们编写了这本Android开发指南。本书一共分为13章，以大量详实的实例为基础，全面系统地向读者介绍了Android开发的各个方面。本书第1~3章为全书的基础，为读者介绍了Android的诞生、开发环境的构建以及Android应用程序开发的基本组件：Activity、Intent、Service等。通过对前面三章内容的学习，读者将对Android应用开发有一个基本了解。接下来在本书的第4~9章，我们将带领读者一步步构建电话应用、音乐播放器应用以及地震观测器应用，使读者深入掌握Android平台下的UI开发、电话应用开发、多媒体操作、数据库使用，以及基于位置的应用开发方法。本书的第10、11章将重点向读者介绍Android SDK 1.5引入的最为重要的三个特性：桌面部件、实时文件夹、虚拟软键盘与输入法应用的开发方法。通过这三章的学习，读者将掌握如何利用这些新特性来开发引人入胜的应用。最后，本书的第12、13章将向读者介绍Android程序的发布流程，包括了程序的签名、使用Android Market等内容。

为了配合本书的学习，我们还提供了配套的视频教程^①。本视频教程一共8讲，可以帮助读者快速掌握Android开发基础，读者可以首先进行视频教程的学习，然后再利用本书进行更深入的学习。在本书的配套网站（<http://www.5billion.com.cn>）上，读者还可以下载对应的源代码。同时Android 2.0也将在2009年的第三季度推出，因此我们将利用网站来不断完善和扩充本书中没有讲到的内容。读者有任何的疑问和心得也可以到网站上与我们交流。

^① 配套资料也可在图灵教育网站（www.turingbook.com）下载。——编者注

目 录

第1章 千呼万唤始出来

——Android的前世今生..... 1

- 1.1 Android的诞生..... 1
- 1.2 Android的优点..... 3
 - 1.2.1 开放性..... 3
 - 1.2.2 应用程序平等..... 3
 - 1.2.3 应用程序间无界限..... 4
 - 1.2.4 快速方便的应用程序开发..... 5
- 1.3 Android系统架构..... 6
 - 1.3.1 应用程序..... 6
 - 1.3.2 应用程序框架..... 7
 - 1.3.3 Android运行时..... 8
 - 1.3.4 系统库..... 9
 - 1.3.5 Linux内核..... 10
- 1.4 Android与移动因特网..... 10
 - 1.4.1 诺基亚的“四大支柱”..... 10
 - 1.4.2 苹果公司的“大超市”..... 11
 - 1.4.3 Google的“云”..... 12
- 1.5 Android的发展..... 12
 - 1.5.1 Android开源应用..... 12
 - 1.5.2 Android设备..... 13
 - 1.5.3 Android平台..... 14
- 1.6 小结..... 15

第2章 而今迈步从头越

——构建 Android 开发环境..... 16

- 2.1 系统需求..... 16
- 2.2 安装软件..... 17
 - 2.2.1 安装JDK..... 17
 - 2.2.2 安装Android SDK..... 17
 - 2.2.3 安装Eclipse以及ADT..... 18
 - 2.2.4 创建AVD..... 22

- 2.3 Android SDK 1.5初体验..... 23
- 2.4 构建第一个Android程序..... 26
- 2.5 解析Hello Android..... 29
 - 2.5.1 应用程序构成..... 30
 - 2.5.2 代码分析..... 31
- 2.6 调试程序..... 34
- 2.7 小结..... 38

第3章 一片冰心在玉壶

——剖析 Android 应用程序核心..... 39

- 3.1 应用程序的基本组件..... 39
 - 3.1.1 Activity简介..... 41
 - 3.1.2 Service简介..... 42
 - 3.1.3 BroadcastReceiver简介..... 43
 - 3.1.4 ContentProvider简介..... 43
 - 3.1.5 Intent——连接组件的纽带..... 44
- 3.2 应用程序的生命周期..... 44
- 3.3 近看Activity及其生命周期..... 48
 - 3.3.1 创建Activity..... 48
 - 3.3.2 Activity的生命周期..... 49
- 3.4 资源..... 57
 - 3.4.1 创建、管理资源..... 58
 - 3.4.2 使用资源..... 58
- 3.5 AndroidManifest.xml文件..... 61
 - 3.5.1 AndroidManifest.xml文件的结构..... 61
 - 3.5.2 在AndroidManifest.xml中声明权限..... 62
- 3.6 小结..... 64

第4章 淡汝浓抹总相宜

——创建用户界面 65

- 4.1 Android用户界面初体验 65
 - 4.1.1 View与ViewGroup 65
 - 4.1.2 可视化控件 66
 - 4.1.3 布局 69
 - 4.1.4 音乐播放器用户界面(1) 82
- 4.2 菜单和对话框 85
 - 4.2.1 菜单 85
 - 4.2.2 对话框 91
 - 4.2.3 音乐播放器用户界面(2) 97
- 4.3 进阶 99
 - 4.3.1 响应用户界面的事件 99
 - 4.3.2 用户界面的数据绑定 104
 - 4.3.3 更改控件外观 109
 - 4.3.4 音乐播放器用户界面(3) 112
- 4.4 小结 114

第5章 青鸟殷勤为探看

——电话与短信应用 115

- 5.1 什么是Intent 115
- 5.2 解析Intent 118
 - 5.2.1 显式Intent和隐式Intent 119
 - 5.2.2 IntentFilter 119
- 5.3 用Intent激活电话拨号程序 121
 - 5.3.1 拨号程序雏形 121
 - 5.3.2 可输入电话号码的拨号程序 125
 - 5.3.3 使用IntentFilter的拨号程序 128
- 5.4 用Intent实现一个简单的短信程序 129
 - 5.4.1 创建TinySMS界面 129
 - 5.4.2 设置权限 131
 - 5.4.3 发送短信 131
- 5.5 Android中电话和短信服务的包 133
- 5.6 小结 133

第6章 仙乐风飘处处闻

——构建你的音乐播放器 134

- 6.1 Service简介 134
- 6.2 使用BroadcastReceiver 138
- 6.3 让TinyPlayer播放音乐 140

- 6.3.1 Service部分 140

- 6.3.2 Activity部分 144

- 6.4 小结 146

第7章 乱花渐欲迷人眼

——使用外部数据和服务 147

- 7.1 实例1: 通过URL获取网络资源 147
- 7.2 实例2: 用HTTP POST提交表单 150
- 7.3 实例3: 用SAX解析XML文件 153
- 7.4 实例4: 利用Google API完成天气
预报 162
 - 7.4.1 用户界面设计与实现 163
 - 7.4.2 解析天气信息 165
- 7.5 实例5: 用DOM解析XML文件——构
建实时地震信息列表 170
 - 7.5.1 XML解析器: DOM和SAX 170
 - 7.5.2 通过DOM构建实时地震信息
列表 171
 - 7.5.3 使用DOM解析 176
- 7.6 小结 179

第8章 日暮乡关何处是

——使用位置服务和地图API 180

- 8.1 位置服务 180
 - 8.1.1 获取位置信息 180
 - 8.1.2 构造LocationProvider查询
条件 183
 - 8.1.3 跟踪位置变化 183
- 8.2 基于地图的应用 186
 - 8.2.1 获取Map API密钥 187
 - 8.2.2 使用MapView下载显示地图 189
 - 8.2.3 在地图上标记当前位置 191
- 8.3 用地图显示地震信息 197
- 8.4 小结 201

第9章 为有源头活水来

——使用SQLite、ContentProvider
与Preferences处理数据 202

- 9.1 文件I/O 202
- 9.2 SQLite数据库 206
 - 9.2.1 SQLite数据库介绍 206

9.2.2 创建/打开SQLite数据库	207	11.2.4 根据目标输入类型调整输入 法布局	264
9.2.3 利用SQLiteDatabase对象操 作数据库	210	11.2.5 获取输入内容	266
9.2.4 Cursor的使用	216	11.2.6 截获硬键盘按键事件	267
9.3 ContentProvider	220	11.2.7 输入候选View	267
9.3.1 定义ContentProvider	220	11.3 小结	268
9.3.2 使用ContentResolver查询、 更改数据	227	第 12 章 试借君王玉马鞭 ——准备发布应用程序	269
9.4 使用Preferences存储应用程序数据	230	12.1 Android应用程序发布概览	269
9.5 让TinyPlayer播放本地音乐文件	233	12.2 最终用户许可协议	269
9.6 小结	236	12.3 应用程序签名	274
第 10 章 珠箔银屏迤迳开 ——桌面组件的开发	237	12.3.1 概述	274
10.1 桌面组件简介	237	12.3.2 签名策略	275
10.2 快捷方式	238	12.3.3 为应用程序签名	275
10.2.1 在应用程序中添加快捷方式	238	12.4 应用程序版本	277
10.2.2 在Launcher添加应用程序 的快捷方式	242	12.4.1 版本管理	277
10.3 实时文件夹	245	12.4.2 应用程序升级	278
10.3.1 定义创建实时文件夹的 Activity	245	12.5 小结	279
10.3.2 定义支持实时文件夹的 ContentProvider	248	第 13 章 书中自有黄金屋 ——销售 Android 应用程序	280
10.4 桌面部件	251	13.1 Sun开发人员的苹果App Store传奇	280
10.5 小结	256	13.2 Android Market简介	280
第 11 章 欲书花叶寄朝云 ——虚拟软键盘与输入法	257	13.2.1 Android Market是什么	281
11.1 软键盘输入法简介	257	13.2.2 Android Market的优势	281
11.2 创建软键盘输入法	259	13.2.3 在Android Market上发布 应用程序	282
11.2.1 软键盘输入法的 AndroidManifest.xml	259	13.3 发布Android应用程序的其他途径	284
11.2.2 输入法和软键盘的核心类	260	13.4 小结	285
11.2.3 输入法	264	附录 A 常用的 Android 常量	286
		附录 B ADB 的使用	291
		附录 C 在模拟器中虚拟 SD 卡	293

第1章

千呼万唤始出来

——Android的前世今生

Android平台于2007年11月一经推出即获得了手机厂商、运营商和芯片厂商的大力支持。这些厂商继而组成一个全球性的联盟组织——开放手机联盟（Open Handset Alliance, OHA）。如摩托罗拉、三星、LG电子、中国移动、日本NTT DoCoMo、Intel以及TI等都相继加入了OHA。

那么，是什么原因导致众多厂商都对Android有如此大的兴趣呢？本章我们将通过介绍Android系统的起源、特点以及系统架构，剖析推出Android平台对于Google的重大意义，以期揭示这一战略举措背后的故事。

1.1 Android 的诞生

Android的诞生还要从Andy Rubin（见图1-1）说起。Rubin是硅谷著名的极客，他家的“门铃”是硅谷最昂贵的玩具：视网膜扫描仪。Rubin很喜欢机器人，所以就可以理解为什么Rubin为创立的新公司取名叫Android了。Rubin的最初目标是想把Android打造成一个可以对任何软件设计人员开放的移动终端平台。很快这个公司就获得了青睐，很多人表示打算买下他的公司。而Rubin发了一封电子邮件给拉里·佩奇，告诉拉里·佩奇有人要跟他合伙的事情。几周之后，Google就抢先把他的公司买下。Google收购Android的时候没有宣布任何计划，只是向《商业周刊》表示：“我们收购Android是因为它拥有天才般的工程师，这些工程师具有非常棒的技术。我们非常兴奋让他们加入Google。”

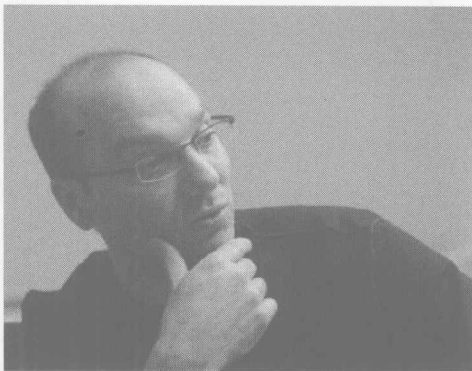


图1-1 Andy Rubin

随着Rubin加入Google，2007年网络上就盛传全球最大的在线搜索服务商Google公司将进军

移动通信市场，并推出自主品牌的移动终端产品。Google手机的照片更是满天飞，光外形就有翻盖、滑盖、旋屏、触控等多种版本。更有人将其与苹果公司于2007年年初推出的iPhone相提并论，取名为“Gphone”，如图1-2所示。

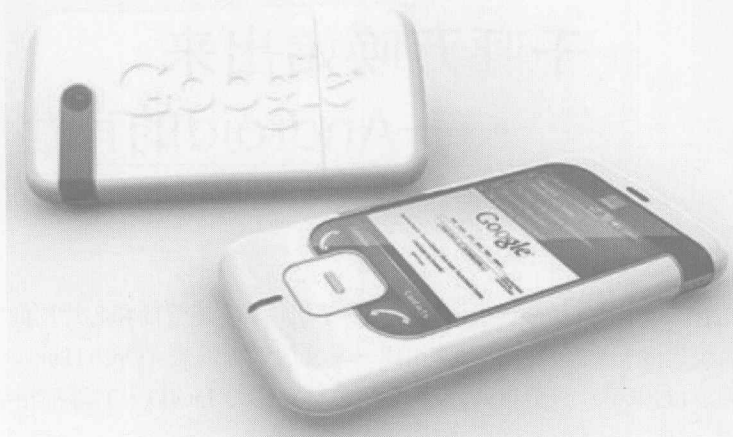


图1-2 Google手机（触控型）

2007年11月5日，Google终于揭开了谜底。Google宣布与其他33家手机厂商（包括摩托罗拉、华为、宏达电、三星、LG等）、手机芯片供货商、软硬件供货商、移动运营商联合组成开放手机联盟（Open Handset Alliance，OHA），并发布了名为Android的开放手机软件平台。参与开放手机联盟的这些厂商，都会基于Android平台来开发新的手机业务。Android向手机厂商和移动运营商提供一个开放的平台，供它们开发创新性的应用软件。Android基于Linux技术，由操作系统、用户界面和应用程序组成，允许开发人员自由获取、修改源代码，也就是说这是一套具有开源性质的手机终端解决方案。

Android作为Google企业战略的重要组成部分，将进一步推进“随时随地为每个人提供信息”这一企业目标的实现。Google的目标是让移动通信不依赖于设备甚至平台，基于此Android将进一步补充Google长期以来的移

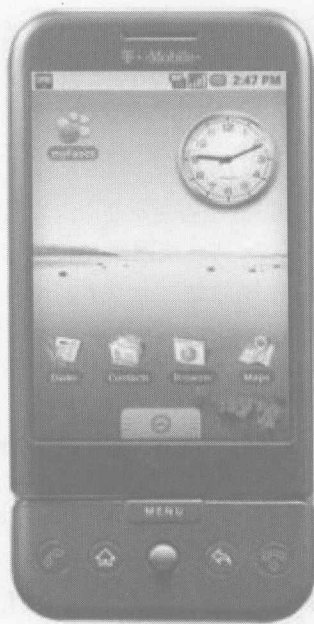


图1-3 第一款Google手机——T-Mobile G1

动发展战略：通过与全球各地的手机厂商和移动运营商结成合作伙伴，开发既有用又有吸引力的移动服务，并推广这些产品。2008年9月22日，美国移动运营商T-Mobile USA在纽约正式发布了第一款Google手机——T-Mobile G1，如图1-3所示。该款手机为宏达电制造，是世界上第一部使用Android操作系统的手机，支持WCDMA/HSPA网络，理论下载速率为7.2Mbit/s，并支持Wi-Fi。我们相信未来将会有越来越多的Android手机握在消费者的手中。

1.2 Android 的优点

与其他手机操作系统相比，Android有4个无可比拟的优点。

- 开放性。
- 所有的应用程序是平等的。
- 应用程序间无界限。
- 快速方便的应用程序开发。

下面我们分别对这4个优点进行讨论。

1.2.1 开放性

Android是一个真正意义上的开放性移动开发平台。它同时包含底层操作系统以及上层的用户界面和应用程序——移动电话工作所需的全部软件，而且不存在任何以往阻碍移动产业创新的专有权障碍。Google与OHA合作开发Android，目的是通过与运营商、设备厂商、开发商等结成深层次的合作伙伴关系，来建立标准化、开放式的移动电话软件平台，在移动产业内形成一个开放式的生态系统，这样应用程序之间的通用性和互联性将在最大程度上得到保持。另一方面，Android平台的开放性还体现在不同的厂商可以根据自己的需求对平台进行定制和扩展，以及使用这个平台无需任何授权许可费用上面。

1.2.2 应用程序平等

所有的Android应用程序之间是完全平等的。在开发之初，Android平台就被设计成由一系列应用程序所组成的平台。所有的应用程序都运行在一个核心引擎上面，这个核心引擎其实就是一个虚拟机，它提供了一系列用于应用程序和硬件资源间通信的API。抛开这个核心引擎，Android的所有其他的东西，包括系统的核心应用和第三方应用都是完全平等的。因此，用户甚至可以将系统中默认的电话拨号软件替换成其他第三方的电话拨号软件。你也可以改变主界面显示窗口的内容，或者将手机中任意的应用程序替换成你所需要的其他应用程序。如图1-4所示就是Android手机中默认的拨号程序，开发人员可以很容易地开发自己拨号程序，然后用自己的拨号程序替代系统的拨号程序（如图1-5所示），这些功能在其他手机平台几乎是不可能实现的。

站在开发者的立场来看,这将大大拓宽可开发应用程序的范围。这样的自由度在Android出现之前是不存在的,之前绝大多数的移动平台内都被固化了一套厂家定制的应用程序,它们不能被替换或删除。Android的改进就在于此,你将不会再面对一堆枯燥无味的固化应用程序而感到无奈。



图1-4 系统默认的拨号程序

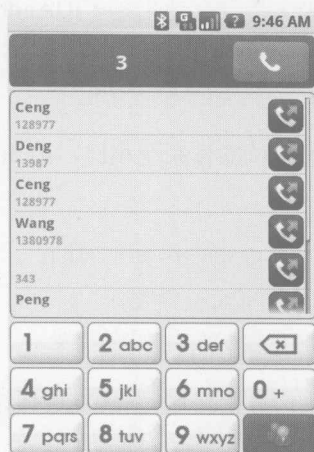


图1-5 替代后的拨号程序

1.2.3 应用程序间无界限

Android打破了应用程序之间的界限,开发人员可以把Web上的数据与本地的联系人、日历、位置信息结合起来,创造全新的用户体验。例如图1-6所示的Google Latitude应用程序就将联系人和位置应用结合在了一起。

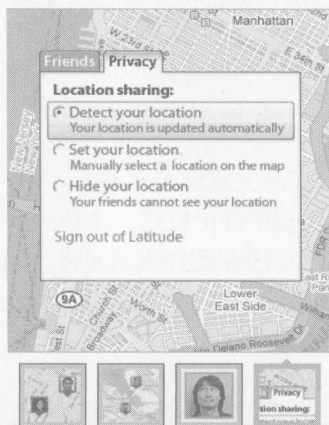


图1-6 Google Latitude应用程序

此外，应用程序不仅可以通过标准API访问核心移动设备功能和因特网，应用程序还可以声明它们的功能可以供其他应用程序使用，比如前面图1-5中的拨号程序就是通过声明其具有拨号功能来实现的，而这一实现只需要一句简单的声明就可以，具体如下所示（更详细的内容将在第3章中讲到）。

```
<action android:name="android.intent.action.DIAL" />
```

1.2.4 快速方便的应用程序开发

Android平台为开发人员提供了大量的实用库和工具，开发人员可以快速地创建自己的应用程序。例如在别的手机平台上要开发基于位置的应用是相当复杂的，而Android将Google Map集成了进来，开发人员通过简单的几行代码就可以实现如图1-7所示的地图应用程序。关于地图应用程序更详细的介绍可以参考第8章。

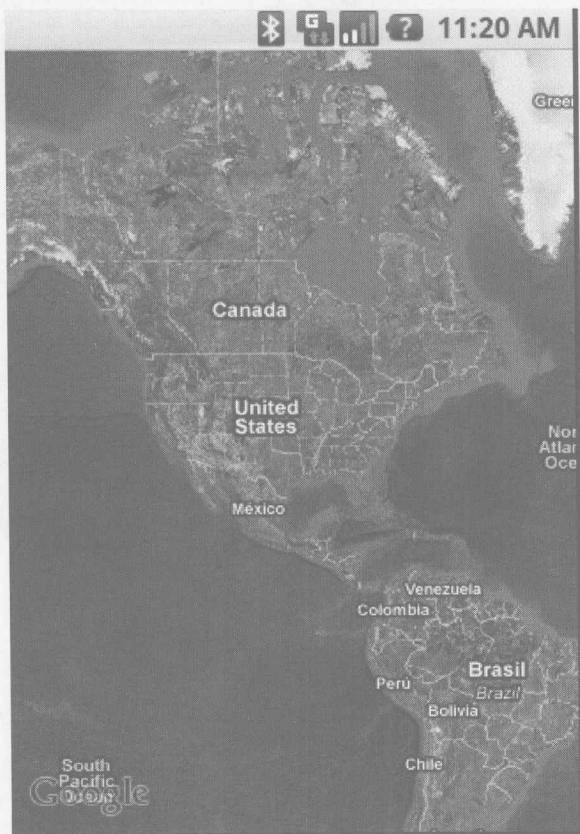


图1-7 地图应用程序

1.3 Android 系统架构

上一节我们提到了Android平台的优越性,那么这些优越性是由什么来保证的呢?这就引出了我们对Android系统架构的分析。从软件分层的角度来看,Android平台由应用程序、应用程序框架、Android运行时、库以及Linux内核共5部分构成,如图1-8所示,在后续的章节我们将向读者详细介绍每一部分的内容。

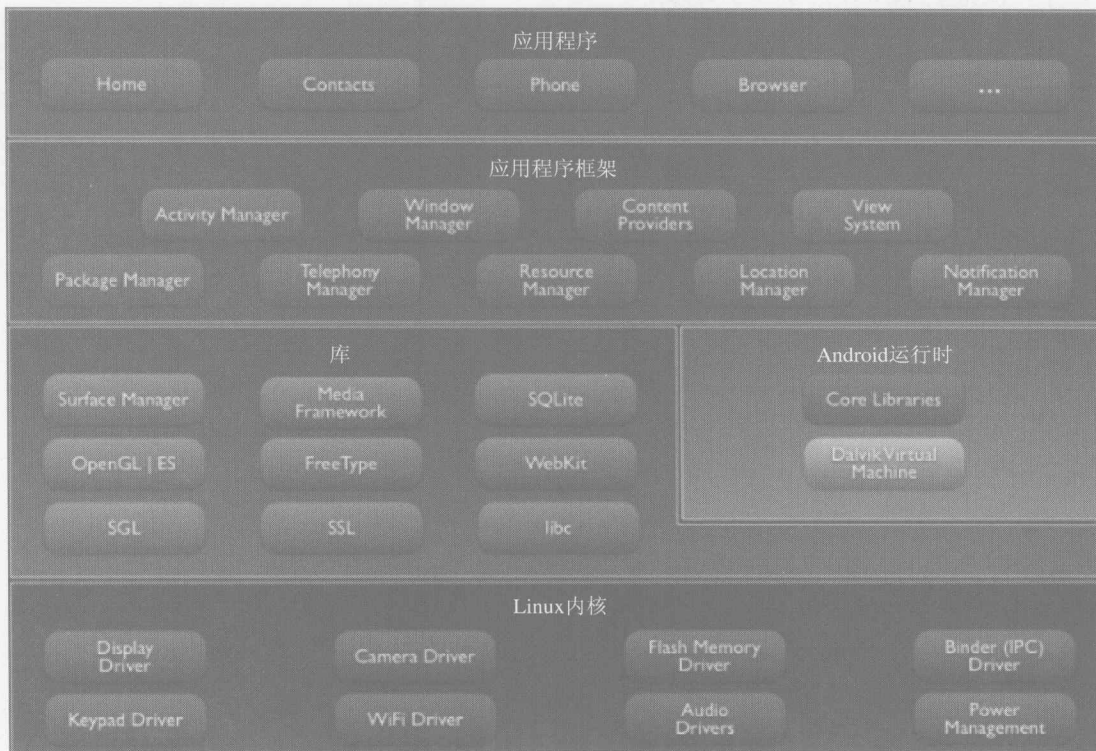


图1-8 Android平台架构图

1.3.1 应用程序

Android平台默认包含了一系列的核心应用程序,包括电子邮件、短信、日历、地图、浏览器、联系人管理程序等,如图1-9所示。这些应用程序都以Java程序语言编写,你也可以用自己编写的应用程序来替换Android提供的应用程序,这个替换的机制实际是由下一节讲述的应用程序框架来保证的。

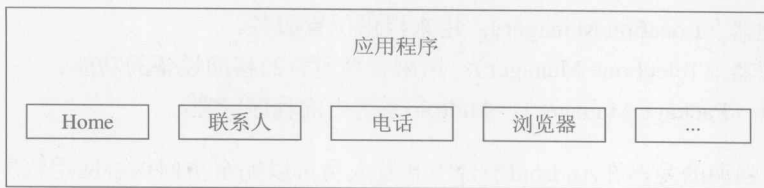


图1-9 Android平台的核心应用程序

1.3.2 应用程序框架

应用程序框架层是我们进行Android开发的基础，开发人员大部分情况下也是和应用程序框架层打交道。应用程序框架层包含了视图系统、内容提供器、窗口管理器、活动管理器、通知管理器、位置管理器、资源管理器、电话管理器和包管理器9大部分，如图1-10所示。

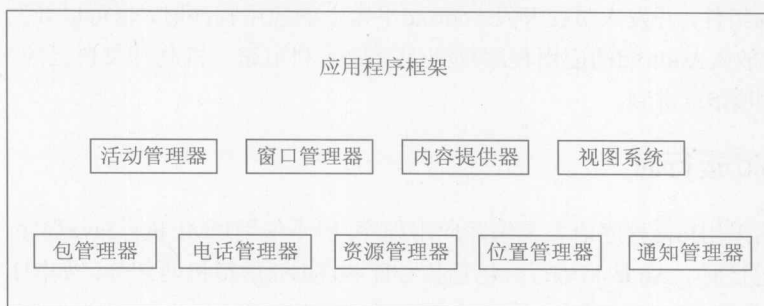


图1-10 应用程序框架层

应用程序框架层的各部分具体功能如下所示。

- ❑ 视图系统 (View System): 用来构建应用程序的基本组件，包含了列表、网格、文本框和按钮等。
- ❑ 内容提供器 (Content Provider): 用来让应用程序之间互相存取/分享数据，例如某个应用程序可以存取联系人应用程序内的联系人数据。
- ❑ 窗口管理器 (Window Manager): 管理所有的窗口程序。
- ❑ 活动管理器 (Activity Manager): 管理所有的应用程序生命周期以及通常的导航返回栈 (navigation backstack)。
- ❑ 通知管理器 (Notification Manager): 使应用程序可以在状态栏应用程序中显示警告信息。状态栏通常在手机的顶部，比如短信、语音邮件 (voice mail) 提示就会出现在这里。
- ❑ 资源管理器 (Resource Manager): 提供各种资源让应用程序去使用，比如本地化字符串、图片、布局文件。

- 位置管理器 (Location Manager): 用来提供位置服务。
- 电话管理器 (Telephone Manager): 用来管理所有的移动设备的功能。
- 包管理器 (Package Manager): Android系统内的程序管理。

这里有必要强调的是,在Android平台中开发人员可以完全访问核心应用程序(即应用程序框架)所使用的API框架,同时Android平台在设计时就考虑了组件的重用。如1.2.2节和1.2.3节提到的应用程序平等和应用程序无界限等特性就是由应用程序框架来保证的。

在Android中,任何一个应用程序都可以发布自身的功能块,而其他应用程序则可以使用(不过得遵循框架的安全性限制)已发布的功能块。基于这样的重用机制,用户就可以方便地替换平台本身的各种应用程序组件,这在Symbian以及Windows Mobile平台中都是无法想象的。例如,虽然Android本身已在框架中提供了许多软件组件,不过这并不表示所有的应用程序一定要调用Android所提供的组件,开发人员在开发Android平台上的应用程序时,也可以开发新的软件组件,并将该软件组件放入Android的应用程序框架中。如何利用这一机制开发自己的应用程序的具体内容将在第3章和第5章讲到。

1.3.3 Android 运行时

Android虽然采用Java程序语言来编写应用程序,但不使用J2ME执行Java程序,而是用Android自有的Android运行时。Android运行时包括核心库和Dalvik虚拟机两部分,如图1-11所示。



图1-11 Android运行时

这两部分具体功能如下所示。

- 核心库。核心库已经包含了两部分内容:一部分为绝大多数Java程序语言所需要调用的功能函数,另一部分为Android的核心库如android.os、android.net、android.media等。与标准Java不一样的是,每个Android应用程序都有一个自有的进程,Android不是用一个Dalvik虚拟机来同时执行多个Android应用程序,而是每个Android应用程序都用一个自有的Dalvik虚拟机来执行。

- Dalvik^①虚拟机。Dalvik是一种基于寄存器的Java虚拟机。Dalvik虚拟机是专门为移动设备而设计的，它在开发时就考虑到了用最少的内存资源来执行，以及支持前面提到的同时执行多个虚拟机的特性。在设计方面，Dalvik虚拟机有许多地方参考了Java虚拟机的设计，不过Dalvik虚拟机所执行的中间码并非是Java虚拟机所执行的Java字节码，但也不直接执行Java的类文件，而是依靠转换工具dx将Java字节码转换为dex（Dalvik EXcutable）格式，称为.dex。Dalvik虚拟机与Java虚拟机最大的不同在于Java虚拟机基于栈（stack-based），而Dalvik基于寄存器（register-based）。从技术层面来考虑，基于寄存器的虚拟机的一个优点就是所需要的资源也相对较少，而且用硬件实现虚拟机也会比较容易。我们知道Java程序的运行比较慢不单单只是因为虚拟机，需要将Java程序编译成字节码也是关键原因之一。

1.3.4 系统库

应用程序框架是贴近于应用程序的软件组件服务，而更底层则是Android的函数库，这一部分是应用程序框架的支撑，其架构如图1-12所示。



图1-12 Android系统库架构

系统库各部分的功能如下。

- 媒体函数库：以PacketVideo公司的OpenCORE为基础所发展成的，该函数库可以播放、录制多种常见的影音格式。
- Surface Manager：在同时执行多个应用程序时，Surface Manager会负责管理显示与存取操作间的互动，另外也负责将2D绘图与3D绘图进行显示合成。
- WebKit：一套网页浏览器的软件引擎，该引擎的功能不仅可供Android内建的网页浏览器所调用，也可以提供具内嵌性网页显示效果。WebKit是一个开源项目，许多浏览器也都是用WebKit引擎所开发成，如Apple的Safari、Nokia S60手机内的浏览器等。
- SGL：提供Android在2D绘图方面的绘图引擎。

^① Dalvik得名于创始人Dan Bornstein的祖先曾经居住过的冰岛小渔村。

- OpenGL ES: Android是依据OpenGL ES 1.0 API标准来实现的其3D绘图函数库, 该函数库可以用软件方式执行也可以用硬件加速方式执行, 其中3D软件光栅处理方面已进行高度优化。
- FreeType: 提供点阵字、向量字的描绘显示。
- 媒体框架: 提供了对各种音频、视频的支持。Android支持多种音频、视频、静态图像格式, 如MPEG-4、H.264、MP3、AAC、AMR、JPG、PNG、GIF等。
- SQLite: 一套轻量级的数据库引擎, 可供其他应用程序调用。
- Libc: 提供了针对移动设备而优化了的C库。

1.3.5 Linux 内核

在1.2.1节我们提到了Android平台的开放性, 采用Linux内核则是Android平台开放性的基础。Android平台中的操作系统采用了Linux 2.6版的内核, 它包括了显示驱动、摄像头驱动、Flash内存驱动、Binder (IPC) 驱动、键盘驱动、Wifi驱动、Audio驱动及电源管理部分, 如图1-13所示。Linux内核层为我们在软件层和硬件层建立了一个抽象层, 使得应用程序开发人员无需关心硬件细节。对于手机的开发商而言, 如果想要Android平台运行到自己的硬件平台上就必须对Linux内核层进行修改, 通常要做的工作就是为自己的硬件编写驱动程序。

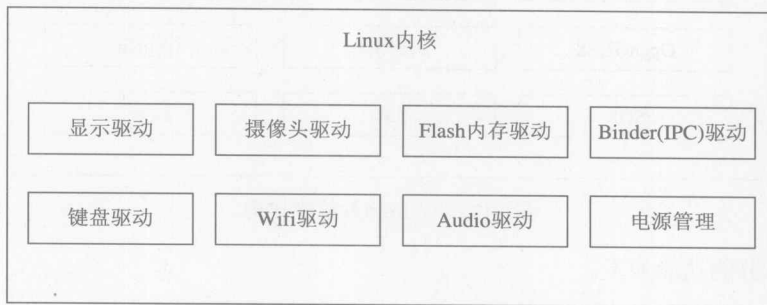


图1-13 Android的Linux内核

1.4 Android 与移动因特网

前面两节分别介绍了Android产生的背景、目的及其架构, 本节我们将比较几大厂商的商业模式, 由此出发分析Android是如何实现Google的移动因特网战略的。

1.4.1 诺基亚的“四大支柱”

诺基亚很早就意识到了手机将成为下一代的PC, 其行业利润将越来越微薄。于是它开始发展