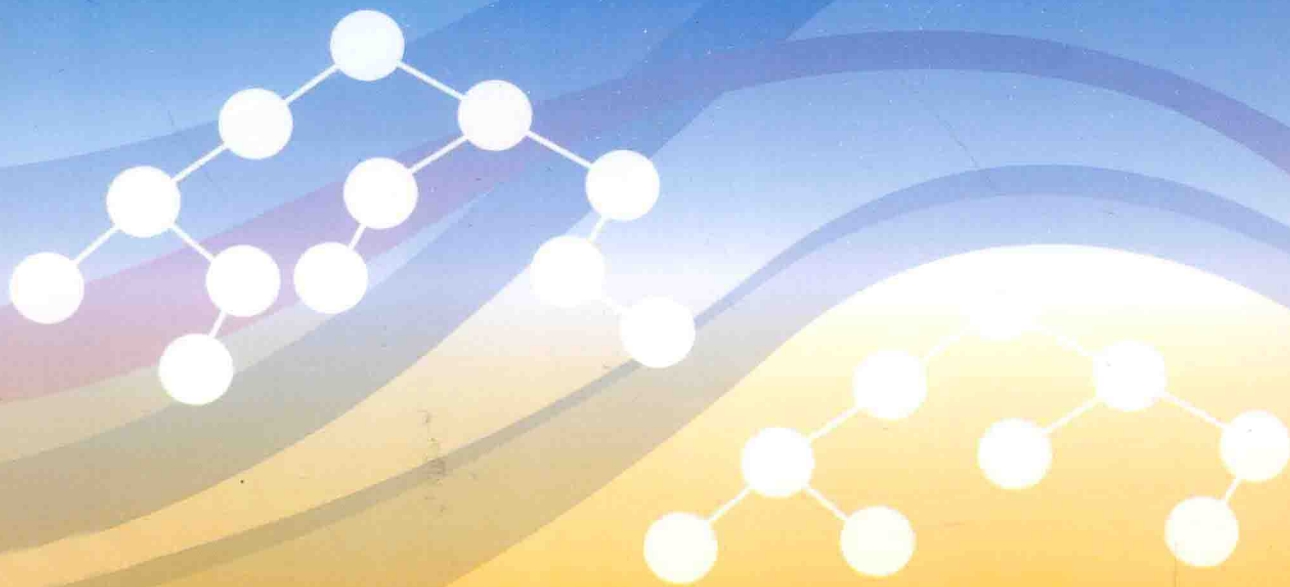


高等学校计算机专业规划教材

Android程序设计



范永开 许林 编著

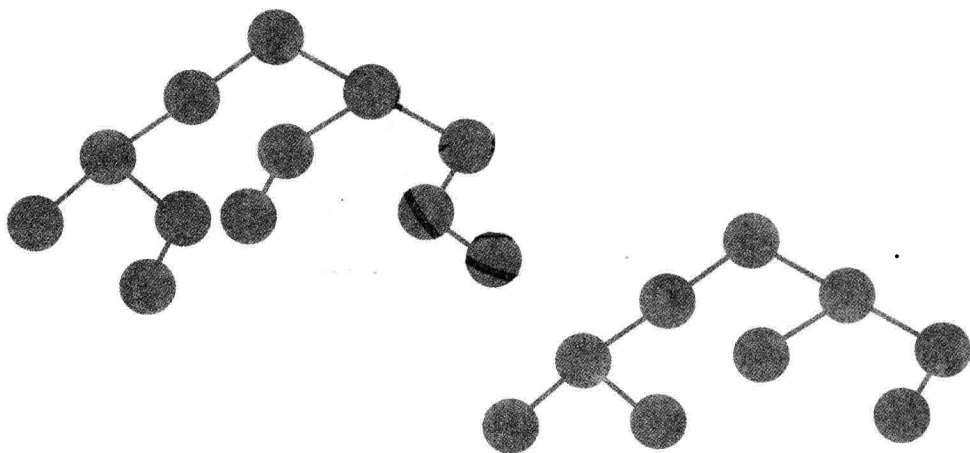
清华大学出版社



高等学校计算机专业规划教材

Android程序设计

范永开 许林 编著



清华大学出版社

内 容 简 介

随着移动应用的普及,Android 程序设计演变成高等院校计算机学科的一门基础课程,许多高校将其列入必修或选修课环节。

本书以基础知识结合实际案例的方式,由浅入深地讲解 Android 开发技术。本书采用多例子式论述展开,考虑高等院校的教学需求,对 Android 程序设计中的核心知识点进行说明,通过简单示例学习重要知识点。全书分为 10 章,内容包括 Android 程序设计的基础知识、Android 界面控件的介绍、Android 事件的详细阐述、Android 程序设计的界面布局、Android 的弹出信息与资源、Android 数据存储与网络应用。同时,对书中内容以程序实例的方式进行阐述,语言通俗易懂,示例丰富实用,能帮助读者拨开晦涩难懂的术语迷雾,一步一步地进行详细指导式学习。

本书从书写与内容设计方面着重考虑适合作为 64 学时的高等院校理工类学生的教材,同时也可作为 Android 程序开发者的参考书。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

Android 程序设计/范永开,许林编著. —北京:清华大学出版社,2014
高等学校计算机专业规划教材
ISBN 978-7-302-34502-2

I. ①A… II. ①范… ②许… III. ①移动终端—应用程序—程序设计—高等学校—教材
IV. ①TN929.53

中国版本图书馆 CIP 数据核字(2013)第 274260 号

责任编辑:龙启铭 王冰飞

封面设计:常雪影

责任校对:时翠兰

责任印制:王静怡

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦 A 座 邮 编:100084

社 总 机:010-62770175 邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质量反馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

印 装 者:北京国马印刷厂

经 销:全国新华书店

开 本:185mm×260mm 印 张:16 字 数:367 千字

版 次:2014 年 6 月第 1 版 印 次:2014 年 6 月第 1 次印刷

印 数:1~2000

定 价:29.50 元

产品编号:053510-01



自从 Google 公司在 2007 年推出 Android 以来,Android 已经经历了 6 年多的发展。从版本 1.5 开始,Android 崭露头角,也开始进入了高速发展阶段。基于 Android 的智能设备是移动领域最具有人气的设备之一。Android 以免费、开源为特点,并且 Google 没有限制使用什么语言或技术在 Android 上开发软件。这就意味着任何企业、组织和个人都可以使用 Android 系统,这也使 Android 的市场占有率节节攀升。无论在国内还是国外,Android 都受到极大的关注和重用,在可预期的未来,在全球范围内的移动领域,Android 将扮演一个重要的角色。

作为传播知识与教育的核心领域——高等院校顺应时代发展的趋势,紧跟信息领域发展的脉搏,许多高校及时调整自己的培养体系,已将 Android 相关知识作为本科生教育的一部分,这部分内容以必修课或选修课的形式出现在课程体系结构中。本书是作者作为一线教师及一线的开发者近几年来讲课与开发的结晶,书中所出现的代码是作者书写程序近 20 年的个人心得与新技术出现的熔融。本书克服文字说教的方式,通过实例来论述知识点,强调在学习过程中提高编程能力,真正回归语言学习的真谛。

本书旨在帮助读者认识 Android,并从各个角度介绍 Android 的核心理念和学习方法,从多个方面介绍和阐述对 Android 架构的理解。只有真正理解 Android 的设计理念和思想,才能更快地掌握 Android 开发。本书深入阐述了 Android 最核心的控件与事件机制,详细介绍了控件的特点和使用方式,以 Android 的交互界面开发为目的,讲解了 Android 的控件框架,结合实际的项目,对重要控件的实现和使用逐一进行了分析与讲解,其中包含了最新的 Android 界面开发的一些实践精华。本书还阐述了 Android 的数据存储结构,不同的数据存储模式的使用要点以及简单的网络应用。

本书非常适合于 64 学时的高等院校理工类学生学习 Android 程序设计,对于具有丰富 Android 开发经验、对 Android 基础框架有很深认识的资深开发者而言,本书可作为一本参考用书。

本书得到中国石油大学(北京)优秀青年教师基金(2462012KYJJ0518)资助,在此表示感谢!

编者
2014 年 5 月



第 1 章 Android 概述 /1

1.1	Android 的由来和发展	1
1.1.1	Android 的由来	1
1.1.2	Android 的发展过程	2
1.2	Android 的体系结构	3
1.2.1	应用程序	3
1.2.2	应用程序框架	3
1.2.3	系统运行库	4
1.2.4	Linux 内核	4
1.3	Android 的优劣	6
1.3.1	Android 平台手机的五大优势	6
1.3.2	Android 的五大不足	7

第 2 章 Android 初探 /8

2.1	准备相关软件	8
2.1.1	操作系统	8
2.1.2	JDK	8
2.1.3	IDE 开发环境	8
2.1.4	软件开发包 Android SDK	9
2.1.5	Android 插件 ADT	10
2.2	Windows+Eclipse 开发环境配置过程	10
2.2.1	安装 JDK	10
2.2.2	安装 Android SDK	11
2.2.3	升级 SDK 版本	15
2.2.4	新建 AVD	15
2.3	Windows+MyEclipse 开发环境配置过程	17
2.3.1	Android ADT 离线安装	17
2.3.2	Android ADT 在线安装	18
2.4	其他开发环境配置过程	22
2.4.1	安装 JDK	22



2.4.2	安装 Android SDK 并创建 AVD	22
2.4.3	安装 Eclipse	23
2.5	第一个 Android 程序——Hello World	26
2.5.1	创建 Android 项目	26
2.5.2	项目目录结构	29
2.5.3	运行项目	33
2.6	Android 测试	35
2.6.1	Log 类和方法	35
2.6.2	LogCat 页面	36
2.7	Activity	37
2.7.1	基本用法	37
2.7.2	常用设置	37
2.7.3	生命周期	38
2.7.4	Activity 加载模式	38
2.7.5	Activity 切换	39
2.7.6	其他常用的 Activity	42

第 3 章 基本界面控件 /43

3.1	TextView	45
3.2	EditText	56
3.3	AutoCompleteTextView	61
3.4	Button	64
3.5	CheckBox	66
3.6	RadioButton	68
3.7	ToggleButton	70
3.8	ImageView	72
3.9	ImageButton	73
3.10	ImageSwitcher 和 Gallery	74
3.11	DigitalClock	77
3.12	AnalogClock	78
3.13	TimePicker	79
3.14	DatePicker	81
3.15	ProgressBar	82
3.16	SeekBar	83
3.17	RatingBar	85
3.18	Spinner	87
3.19	实现注册界面	89



第 4 章 Android 事件 /94

4.1	事件的过程及原理	94
4.1.1	事件的过程	94
4.1.2	事件机制原理	95
4.2	事件处理模型	96
4.2.1	接口实现事件处理模型	98
4.2.2	内部类事件处理模型	99
4.2.3	匿名内部类事件处理模型	100
4.3	Android 事件处理机制	101
4.3.1	回调机制	101
4.3.2	监听机制	104
4.4	常见事件	105
4.4.1	触摸屏事件	105
4.4.2	手势识别	109
4.4.3	键盘事件	113
4.4.4	模拟鼠标与按键事件	117
4.4.5	菜单事件	119

第 5 章 Android 布局 /123

5.1	布局概述	123
5.2	LinearLayout	124
5.3	RelativeLayout	127
5.4	TableLayout	129
5.5	AbsoluteLayout	132
5.6	FrameLayout	133
5.7	GridView	134
5.8	ListView	136
5.9	计算器的实现	140

第 6 章 Android 弹出信息 /147

6.1	Toast	147
6.2	AlertDialog	154

第 7 章 Android 资源 /167

7.1	res/values	167
7.1.1	strings.xml	168
7.1.2	arrays.xml	170



7.1.3	Bools.xml	171
7.1.4	colors.xml	172
7.1.5	dimens.xml	173
7.1.6	ids.xml	175
7.1.7	styles.xml	176
7.2	res/drawable	177
7.3	res/xml	178
7.4	res/menu	180
7.5	res/raw	183
7.6	res/assets	184
7.7	资源的国际化	186

第 8 章 Android 菜单 /189

8.1	选项菜单	190
8.2	子菜单	198
8.3	上下文菜单	203
8.4	实例代码	206

第 9 章 数据存储 /210

9.1	使用 SharedPreferences 存储数据	210
9.1.1	获得 SharedPreferences	211
9.1.2	增加或者更新数据	211
9.1.3	读取数据	211
9.1.4	清空数据	211
9.1.5	PreferenceActivity	212
9.2	文件存储数据	214
9.3	SQLite	217
9.3.1	SQLiteOpenHelper 类	219
9.3.2	SQLiteDatabase 类	220
9.3.3	Cursor 接口	222
9.3.4	标准数据库 adapter 类的实现代码	223
9.3.5	注意事项	225
9.4	使用 ContentProvider 存储数据	226
9.4.1	使用 ContentProvider 共享数据	227
9.4.2	Uri 介绍	234
9.5	网络存储数据	238
9.6	实现方式总结	239

第 10 章 网络应用 /240

10.1	Android 的 HTTP 通信	240
10.1.1	Java.net.HttpURLConnection 的 get 方式	241
10.1.2	Java.net.HttpURLConnection 的 post 方式	241
10.1.3	org.apache.http 的 get 方式	243
10.1.4	org.apache.http 的 post 方式	243
10.2	设置代理	244
10.2.1	HttpURLConnection	244
10.2.2	HttpClient	244

第 1 章

Android 概述

当前参与移动业务领域的有四类人群：移动用户、移动运营商、移动开发人员和移动产品制造商。移动用户需要更多的、更实用的、更个性化的功能。移动运营商需要更易于管理、获利更多的增值服务。移动开发人员需要可以自由开发的开发环境，不要有太多限制。移动产品制造商需要一个稳定、安全、廉价的系统平台，在充分发挥硬件的性能的基础上降低成本。Android 系统是唯一可以满足以上四类人群需要的移动开发平台。

未来 IT 行业中，移动业务将占有举足轻重的地位。越来越多的公司，包括电商、金融、餐饮等行业的公司都在实现自己的移动业务系统，以方便客户使用。Android 系统是普遍采用的移动操作系统。

本书所介绍的 Android 系统是基于 Android 4.0 版本的，这是当前移动产品采用比较广泛的平台版本。

作为学习和了解 Android 的第一步，在这一章中，我们将简要地介绍 Android 的由来、发展和优势。

1.1 Android 的由来和发展

1.1.1 Android 的由来

Android 一词最先出现在法国作家利尔亚当在 1886 年发表的科幻小说《未来夏娃》中，作者将外表像人类的机器起名为 Android，如图 1.1 所示。

Google 将其基于 Linux 平台的开源手机操作系统命名为 Android，该平台由操作系统、中间件、用户界面和应用软件组成，并采用图 1.1 作为 Logo。

Android 的 Logo 是由 Ascender 公司设计的，其中的文字使用了 Ascender 公司专门制作的称为 Droid 的字体。Android 是一个全身绿色的机器人，绿色也是 Android 的标志。颜色采用了 PMS 376C 和 RGB 中十六进制的 #A4C639 来绘制，这是 Android 操作系统的品牌形象。有时候，它们还会使用纯文字的。



图 1.1 Android 的 Logo



1.1.2 Android 的发展过程

2003年10月,Andy Rubin 等人创建 Android 公司,并组建 Android 团队。

2005年8月17日,Google 低调收购了成立仅22个月的高科技企业 Android 及其团队。安迪·鲁宾成为 Google 公司工程部副总裁,继续负责 Android 项目。

2007年11月5日,Google 公司正式向外界展示了这款名为 Android 的操作系统,并且在这天 Google 宣布建立一个全球性的联盟组织,该组织由34家手机制造商、软件开发商、电信运营商以及芯片制造商共同组成,并与84家硬件制造商、软件开发商及电信运营商组成开放手持设备联盟(Open Handset Alliance)来共同研发改良 Android 系统,这一联盟将支持 Google 发布的手机操作系统以及应用软件,Google 以 Apache 免费开源许可证的授权方式发布了 Android 的源代码。

2008年,在 Google I/O 大会上,Google 提出了 Android HAL 架构图,在同年8月18日,Android 获得了美国联邦通信委员会(FCC)的批准,在2008年9月,Google 正式发布了 Android 1.0 系统,这也是 Android 系统最早的版本。

2009年4月,Google 正式推出了 Android 1.5 这款手机,从 Android 1.5 版本开始,Google 开始将 Android 的版本以甜品的名字命名,Android 1.5 命名为 Cupcake(纸杯蛋糕)。该系统与 Android 1.0 相比有了很大的改进。

2009年9月,Google 发布了 Android 1.6 的正式版,并且推出了搭载 Android 1.6 正式版的手机 HTC Hero(G3),凭借着出色的外观设计以及全新的 Android 1.6 操作系统,HTC Hero(G3)成为当时全球最受欢迎的手机。Android 1.6 也有一个有趣的甜品名称,它被称为 Donut(甜甜圈)。

2010年2月,Linux 内核开发者 Greg Kroah-Hartman 将 Android 的驱动程序从 Linux 内核“状态树”(“staging tree”)上除去,从此,Android 与 Linux 开发主流分道扬镳。在同年5月,Google 正式发布了 Android 2.2 操作系统。Google 将 Android 2.2 操作系统命名为 Froyo(冻酸奶)。

2010年10月,Google 宣布 Android 系统达到了第一个里程碑,即电子市场上获得官方数字认证的 Android 应用数量已经达到了10万个,Android 系统的应用增长非常迅速。在2010年12月,Google 正式发布了 Android 2.3 操作系统 Gingerbread(姜饼)。

2011年1月,Google 称每日的 Android 设备新用户数量达到了30万部,到2011年7月,这个数字增长到55万部,而 Android 系统设备的用户总数达到了1.35亿,Android 系统已经成为智能手机领域占有量最高的系统。

2011年8月2日,Android 手机已占据全球智能机市场48%的份额,并在亚太地区市场占据统治地位,终结了 Symbian(塞班系统)的霸主地位,跃居全球第一。

2011年9月,Android 系统的应用数目已经达到48万,而在智能手机市场,Android 系统的占有率已经达到43%,继续排在移动操作系统首位。在本月19日,Google 发布了全新的 Android 4.0 操作系统,这款系统被 Google 命名为 Ice Cream Sandwich(冰激凌三明治)。

2012年1月6日,Google Android Market 已有10万开发者推出超过40万活跃的应

用,大多数的应用程序为免费。Android Market 应用商店在新年首周末突破 40 万基准,距离突破 30 万应用仅 4 个月。在 2011 年早些时候,Android Market 从 20 万增加到 30 万应用也花了 4 个月。

1.2 Android 的体系结构

Android 的系统架构和其操作系统一样,采用了分层的架构,如图 1.2 所示,分为 4 个层,从高层到低层分别是应用程序层、应用程序框架层、系统运行库层和 Linux 核心层。



图 1.2 Android 系统架构图

1.2.1 应用程序

Android 会同一系列核心应用程序包一起发布,该应用程序包包括 E-mail 客户端、SMS 短消息程序、日历、地图、浏览器、联系人管理程序等。所有的应用程序都是使用 Java 语言编写的。

1.2.2 应用程序框架

开发人员可以完全访问核心应用程序所使用的 API 框架。该应用程序的架构设计简化了组件的重用;任何一个应用程序都可以发布它的功能块并且任何其他的应用程序都可以使用其所发布的功能块(不过要遵循框架的安全性限制)。同样,该应用程序重用机制也使用户可以方便地替换程序组件。

隐藏在每个应用后面的是一系列的服务和系统,其中包括:

- 丰富而又可扩展的视图(Views),可以用来构建应用程序,它包括列表(lists)、网格(grid)、文本框(text boxes)、按钮(buttons),甚至可嵌入的 Web 浏览器。
- 内容提供者(Content Providers),使得应用程序可以访问另一个应用程序的数据

(如联系人数据库), 或者共享它们自己的数据。

- 资源管理器(Resource Manager), 提供非代码资源的访问, 如本地字符串、图形和布局文件(layout files)。
- 通知管理器(Notification Manager), 使得应用程序可以在状态栏中显示自定义的提示信息。
- 活动管理器(Activity Manager), 用来管理应用程序生命周期并提供常用的导航回退功能。

1.2.3 系统运行库

1. 程序库

Android 包含一些 C/C++ 库, 这些库能被 Android 系统中不同的组件使用。它们通过 Android 应用程序框架为开发者提供服务。程序库包括以下核心库。

- 系统 C 库: 一个从 BSD 继承来的标准 C 系统函数库(libc), 它是专门为基于嵌入式 Linux 的设备定制的。
- 媒体库: 基于 PacketVideoOpenCORE。该库支持多种常用的音频、视频格式回放和录制, 同时支持静态图像文件。编码格式包括 MPEG4、H. 264、MP3、AAC、AMR、JPG 和 PNG。
- Surface Manager: 对显示子系统的管理, 并且为多个应用程序提供了 2D 和 3D 图层的无缝融合。
- LibWebCore: 一个最新的 Web 浏览器引擎, 支持 Android 浏览器和一个可嵌入的 Web 视图。
- SGL: 底层的 2D 图形引擎。
- 3D 库: 基于 OpenGL ES 1.0 APIs 实现。该库可以使用硬件 3D 加速(如果可用)或者使用高度优化的 3D 软加速。
- FreeType: 位图(bitmap)和矢量(vector)字体显示。
- SQLite: 一个对于所有应用程序可用、功能强劲的轻型关系型数据库引擎。

2. Android 运行库

Android 包括一个核心库, 该核心库提供了 Java 编程语言核心库的大多数功能。

每一个 Android 应用程序都在它自己的进程中运行, 都拥有一个独立的 Dalvik 虚拟机实例。Dalvik 被设计成一个设备可以同时高效地运行多个虚拟系统。Dalvik 虚拟机执行的 Dalvik 可执行文件(.dex), 该格式文件针对小内存使用做了优化。同时虚拟机是基于寄存器的, 所有的类都经由 Java 编译器编译, 然后通过 SDK 中的 dx 工具转化成 .dex 格式由虚拟机执行。

Dalvik 虚拟机依赖于 Linux 内核的一些功能, 比如线程机制和底层内存管理机制。

1.2.4 Linux 内核

Android 的核心系统服务依赖于 Linux 2.6 内核, 如安全性、内存管理、进程管理、网络协议栈和驱动模型。Linux 内核也同时作为硬件和软件栈之间的抽象层。

1. 系统内核

Android 运行于 Linux Kernel 之上,但并不是 GNU/Linux。因为在一般的 GNU/Linux 中支持的功能,Android 大都没有支持,包括 Cairo、X11、Alsa、FFmpeg、GTK、Pango 及 Glibc 等都被移除掉了。Android 又以 Bionic 取代 Glibc,以 Skia 取代 Cairo,再以 opencore 取代 FFmpeg,等等。Android 为了达到商业应用,必须移除被 GNU GPL 授权证所约束的部分,例如 Android 将驱动程序移到 Userspace,使得 Linux 驱动与 Linux 内核彻底分开。Bionic/Libc/Kernel 并非标准的内核头文件。Android 的内核头是利用工具由 Linux 内核头产生的,这样做是为了保留常数、数据结构与宏。

Android 的 Linux 内核控制包括安全 (Security)、存储器管理 (Memory Management)、程序管理 (Process Management)、网络堆栈 (Network Stack) 和驱动程序模型 (Driver Model) 等。下载 Android 源码之前,先要安装其构建工具 Repo 来初始化源码。Repo 是 Android 用来辅助 Git 工作的一个工具。

2. 硬件抽象层

Android 的 HAL(硬件抽象层)以封闭源码形式提供硬件驱动模块。HAL 的目的是为了把 Android 框架与 Linux 内核隔开,让 Android 不至于过度依赖 Linux 内核,以达成内核独立的概念,也让 Android 框架的开发能在不考虑驱动程序实现的前提下进行发展。

HAL Stub 是一种代理人 (Proxy) 的概念,Stub 是以 *.so 档的形式存在的。Stub 向 HAL“提供”操作函数 (Operations),并由 Android runtime 向 HAL 取得 Stub 的 Operations,再 Callback 这些操作函数。HAL 中包含了许多 Stub(代理人)。Runtime 只要说明“类型”,即 Module ID,就可以取得操作函数。

3. 中介软件

中介软件是操作系统与应用程序的沟通桥梁,分为两层:函数层 (Library) 和虚拟机 (Virtual Machine)。

Android 采用 OpenCORE 作为基础多媒体框架。Open CORE 可分为七大块:PVPlayer、PVAuthor、Codec、PacketVideo Multimedia Framework (PVMF)、Operating System Compatibility Library (OSCL)、Common 和 OpenMAX。

Android 使用 Skia 作为核心图形引擎,搭配 OpenGL/ES。Skia 与 Linux Cairo 功能相当,但与 Linux Cairo 相比,skia 功能还只是雏形的。2005 年 Skia 公司被 Google 收购,2007 年初,Skia GL 源码被公开,Skia 也是 Google Chrome 的图形引擎。

Android 的多媒体数据库采用 SQLite 数据库系统。数据库又分为共用数据库和私有数据库。用户可通过 ContentResolver 类 (Column) 取得共用数据库。

Android 的中间层多以 Java 实现,并且采用特殊的 Dalvik 虚拟机 (Dalvik Virtual Machine)。Dalvik 虚拟机是一种“暂存器形态” (Register Based) 的 Java 虚拟机,变量皆存放于暂存器中,虚拟机的指令相对减少。

Dalvik 虚拟机可以有多个实例 (Instance),每个 Android 应用程序都用一个自属的 Dalvik 虚拟机来运行,让系统在运行程序时可达达到优化。Dalvik 虚拟机并非运行 Java 字节码 (Bytecode),而是运行一种称为 .dex 格式的文件。

4. 安全权限机制

Android 本身是一个权限分立的操作系统。在这类操作系统中,每个应用都以唯一

的一个系统识别身份运行(Linux 用户 ID 与群组 ID)。系统的各部分也分别使用各自独立的识别方式。Linux 也是这样将应用与应用、应用与系统隔离开的。

系统更多的安全功能通过权限机制提供。权限可以限制某个特定进程的特定操作,也可以限制每个 URI 权限对特定数据段的访问。

Android 安全架构的核心设计思想是:在默认设置下,所有应用都没有权限对其他应用、系统或用户进行较大影响的操作。这其中包括读写用户隐私数据(联系人或电子邮件),读写其他应用文件,访问网络或阻止设备待机等。

安装应用时,在检查程序签名提及的权限,且经过用户确认后,软件包安装器会给予应用权限。从用户角度看,一款 Android 应用通常会要求如下的权限:拨打电话、发送短信或彩信、修改/删除 SD 卡上的内容、读取联系人的信息、读取日程信息,写入日程数据、读取电话状态或识别码、精确的(基于 GPS)地理位置、模糊的(基于网络获取)地理位置、创建蓝牙连接、对互联网的完全访问、查看网络状态、查看 WiFi 状态、避免手机待机、修改系统全局设置、读取同步设定、开机自启动、重启其他应用、终止运行中的应用、设定偏好应用、震动控制、拍摄图片等。

一款应用应该根据自身提供的功能,要求合理的权限。用户也可以分析一款应用所需权限,从而简单判定这款应用是否安全。例如,一款应用是不带广告的单机版,也没有任何附加的内容需要下载,那么它要求访问网络的权限就比较可疑。

1.3 Android 的优劣

1.3.1 Android 平台手机的五大优势

1. 开放性

Android 平台的最大优势就是其开放性,Android 平台允许任何移动终端厂商加入到 Android 联盟中来。显著的开放性可以使其拥有更多的开发者,随着用户和应用的日益丰富,一个崭新的平台也将很快走向成熟。

开放性对于 Android 的发展而言,有利于积累人气,这里的人气包括消费者和厂商。对于消费者来说,最大的受益是丰富的软件资源。开放的平台也会带来更大竞争,如此一来,消费者将可以用更低的价格购得心仪的手机。

2. 挣脱运营商的束缚

在过去很长的一段时间,特别是在欧美地区,手机应用往往受到运营商的制约,使用什么功能、接入什么网络,几乎都受到运营商的控制。从 iPhone 上市后,用户可以更加方便地连接网络,运营商的制约减少。随着 EDGE、HSDPA 这些 2G 至 3G 移动网络的逐步过渡和提升,手机随意接入网络已不是运营商口中的笑谈,当你通过手机 IM 软件方便地进行即时聊天时,再回想不久前天价的彩信和图铃下载业务,是不是像噩梦一样?

互联网巨头 Google 推动的 Android 终端天生就有网络特色,这让用户离互联网更近。

3. 丰富的硬件选择

这一点还是与 Android 平台的开放性相关,由于 Android 的开放性,众多的厂商会推

出千奇百怪、各具特色的产品。功能上的差异和特色却不会影响到数据同步,甚至软件的兼容,好比从诺基亚 Symbian 风格手机一下改用苹果 iPhone,同时还可将 Symbian 中优秀的软件带到 iPhone 上使用,联系人等资料更是可以方便地转移,是不是非常方便呢?

4. 不受任何限制的开发商

Android 平台提供给第三方开发商一个十分宽泛、自由的环境,不会受到各种条条框框的阻挠,可想而知,会有多少新颖别致的软件诞生。但这个特点也有其两面性,对血腥、暴力、情色方面的程序和游戏如何控制,正是留给 Android 的难题之一。

5. 无缝结合的 Google 应用

如今叱咤互联网的 Google 已经走过 10 多年的历史,从搜索巨人到全面的互联网渗透,Google 服务如地图、邮件、搜索等已经成为连接用户和互联网的重要纽带,而 Android 平台手机将无缝结合这些优秀的 Google 服务。

1.3.2 Android 的五大不足

1. 安全和隐私

由于手机与互联网的紧密联系,个人隐私很难得到保护。除了上网过程中经意或不经意留下的个人足迹,Google 这个巨人也时时站在你的身后,洞穿一切。因此,互联网的深入将会带来新一轮的隐私危机。

2. 首先开卖 Android 手机的不是最大运营商

众所周知,T-Mobile 在 2008 年 9 月 22 日,于美国纽约发布了 Android 首款手机 G1。但是在北美市场,最大的两家运营商是 AT&T 和 Verizon,而目前所知,取得 Android 手机销售权的仅有 T-Mobile 和 Sprint,其中,T-Mobile 的 3G 网络相对于其他三家也要逊色不少,因此,用户可以买账购买 G1,能否体验到最佳的 3G 网络服务则要另当别论了。

3. 运营商仍然能够影响到 Android 手机

在国内市场,不少用户对购得移动定制机不满,感觉所购的手机被人涂画了广告一般。这样的情况在国外市场同样出现。Android 手机的另一发售运营商 Sprint 就将在其机型中内置其手机商店程序。

4. 同类机型用户减少

在不少手机论坛都会有针对某一型号的子论坛,对一款手机的使用交流心得,并分享软件资源。而对于 Android 平台手机,由于厂商丰富,产品类型多样,因此使用同一款机型的用户较少,缺少统一机型的程序强化。举个稍显不当的例子,现在山寨机泛滥,品种各异,就很少有专门针对某个型号山寨机的讨论和群组,除了那些功能异常抢眼、颇受追捧的机型以外。

5. 过分依赖开发商,缺少标准配置

在使用 PC 端的 Windows XP 系统时,都会内置微软 Windows Media Player 这样一个程序,用户还可以选择更多样的播放器,如 RealPlay 或暴风影音等。但入手开始使用默认的程序同样可以应付多样的需要。在 Android 平台中,由于其开放性,软件更多依赖第三方厂商,例如 Android 系统的 SDK 中就没有内置音乐播放器,全部依赖第三方开发,缺少了产品的统一性。

通过第 1 章的学习,我们知道了 Android 的基本情况。下面将指导大家准备 Android 开发环境,并指引大家编写一个简单的 Android 程序以及如何进行简单的测试。

2.1 准备相关软件

下面介绍安装 Android 需要的相关软件。

2.1.1 操作系统

支持 Android 应用程序的操作系统有以下几种:

- (1) Windows XP、Windows 7 或 Windows Vista。
- (2) Linux。
- (3) Mac OS X 10.4.8 及新版本。

2.1.2 JDK

Android 开发环境是基于 Java 的。JDK 是整个 Java 的核心,包括 Java 运行环境、Java 工具和 Java 基础类库。这是搭建平台所必需的。

JDK 的下载地址是 <http://www.oracle.com/technetwork/Java/Javase/downloads/index.html>,下载界面如图 2.1 所示。

选择 JDK DOWNLOAD,只下载 JDK,无须下载 JRE。

2.1.3 IDE 开发环境

在 IDE 开发环境中,开发人员多数使用免费的 Eclipse 集成开发环境。Eclipse 的下载地址是 <http://www.eclipse.org/downloads/>,下载界面如图 2.2 所示。

首先在 Eclipse Juno (4.2) SR1 Packages for 后面的下拉列表框中选择使用的操作系统,然后选择第一个(即 Eclipse IDE for Java EE Developers)下载即可。解压到文件夹之后即可使用,无须安装。

当然,也可以选择 MyEclipse,它是对 Eclipse IDE 的扩展。MyEclipse 的下载地址是 <http://www.myeclipseide.com/>。下载完成后按照操作提示安装即可。