

Android

开发技术

● 许 超 主编

● 张晓军 赖 炜 副主编



化学工业出版社

Android 开发技术

许 超 主编

张晓军 赖 炜 副主编



化学工业出版社

· 北 京 ·

第 00.88 号 第 00 页

本书是校企合作开发的教材,以 Android 市场发布过的手机游戏作为教学案例,将案例中涉及的相关知识点,有机地融入教学过程中,并按照 Android 平台体系特征,详细介绍了各类 Android 项目开发所用的共性技术。

本书主要介绍了 Android 入门基础、Android 开发环境、Android 开发准备、Android 基本组件、Intent 和 Broadcast 应用、Android 的数据存储操作以及 Service 应用,最后通过《贪吃虎》游戏案例,综合介绍了 Android 技术的应用方法,并附有详细的开发源代码。读者通过本书的学习,将全面、系统地掌握 Android 平台相关开发技术,同时还将深入了解这些技术如何具体应用到企业开发实践中去。

本书可供高职高专计算机应用技术、计算机网络技术、软件工程、物联网等相关专业教学使用,也可供相关工程技术人员参考。

图书在版编目(CIP)数据

Android 开发技术 / 许超主编. —北京: 化学工业出版社, 2018.2

ISBN 978-7-122-31255-6

I. ①A… II. ①许… 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 (2017) 第 330451 号

责任编辑: 王听讲

装帧设计: 刘丽华

责任校对: 王 静

出版发行: 化学工业出版社 (北京市东城区青年湖南街 13 号 邮政编码 100011)

印 刷: 三河市航远印刷有限公司

装 订: 三河市宇新装订厂

787mm×1092mm 1/16 印张 14¼ 字数 364 千字 2018 年 3 月北京第 * 版藏 1 次印刷 *



购书咨询: 010-64518888 (传真: 010-64519686)

售后服务: 010-64518899

网 址: <http://www.cip.com.cn>

凡购买本书,如有缺损质量问题,本社销售中心负责调换。

定 价: 48.00 元

版权所有 违者必究

前 言

市场调查数据显示,在全球智能手机操作系统中,Android 系统占大多数,中国市场占有率更是超过了 85%,促使了就业市场对 Android 开发人员的需求量猛增。

为了更好地培养 Android 开发人才,我们与湖南东君科技实行校企深度合作,组建了课程研发团队,通过长期教学实践,逐步完成了本书的构思和编写任务。在本书编写过程中,重点体现本课程的专业基础平台课的性质,在内容的安排和深度的把握上,坚持传授 Android 应用开发技能,培养学生运用基础知识解决实际问题的能力。本书有以下几方面的特色。

(1) 内容先进。本书重点介绍当前市场主流的 Android 操作系统,详细地叙述了 Android 项目开发的整个流程,引导学生了解最新的 Android 平台应用开发技术。

(2) 代码丰富。本书根据高职教学实际情况,文字表达浅显易懂,案例有趣,并配有大量的开发源代码,方便教师授课,同时也便于学生理解。

(3) 突出实用。本书强调实用性,以培养学生完成实际工作能力为重点,紧密联系企业实际,增加了用户需求分析、UI 设计、市场发布等内容。

(4) 结构合理。本书在内容编排上紧密结合职业教育实际,符合职业院校学生的认知规律。

本书主要介绍了 Android 入门基础、Android 开发环境、Android 开发准备、Android 基本组件、Intent 和 Broadcast 应用、Android 的数据存储操作以及 Service 应用,最后通过《贪吃虎》游戏案例,综合介绍了 Android 技术的应用方法,并附有详细的开发源代码。读者通过本书的学习,将全面、系统地掌握 Android 平台相关开发技术,同时还将深入了解这些技术如何具体应用到企业开发实践中去。

本书可供高职高专计算机应用技术、计算机网络技术、软件工程、物联网等相关专业教学使用,也可供相关工程技术人员参考。

本书由许超主编,并对全书进行统稿;张晓军、赖炜担任副主编,李合军参加编写。在本书编写过程中,东君科技公司廖彦高级工程师、熊曼青高级工程师为本书编写提供了资料,刘欢、江海潮完成部分图表绘制及文档排版任务,在此对他们表示衷心的感谢。

限于编者的水平和经验,书中难免存在不妥之处,恳请读者提出批评和修改意见。

编 者
2018 年 1 月

目 录

第 1 章 Android 入门基础	1
1.1 Android 语言概述	1
1.1.1 Android 的概念	1
1.1.2 Android 的发展简史	1
1.2 Android 的体系结构	2
1.2.1 应用程序 (Application)	2
1.2.2 应用程序框架	2
1.2.3 库 (Libraries) 和 Android 运行环境 (Run-time)	3
1.2.4 操作系统 (OS)	3
1.3 Android SDK	4
1.3.1 Android SDK 基础	4
1.3.2 Android SDK 目录结构	4
1.3.3 Android.jar 及内部结构	5
1.3.4 Android API 核心开发包	5
1.3.5 Android SDK 1.5 的新特性	6
第 2 章 Android 开发环境	7
2.1 Android 开发环境搭建	7
2.1.1 Android 开发系统要求	7
2.1.2 下载所需软件包	7
2.1.3 安装 Android SDK	9
2.1.4 安装 ADT	9
2.1.5 设置 SDK	13
2.1.6 验证开发环境	15
2.2 Android 模拟器	21
2.2.1 模拟器概述	21
2.2.2 使用命令行工具管理模拟器	21
2.2.3 操作模拟器	21
2.2.4 模拟器与真机的区别	22
2.2.5 使用模拟器的注意事项	22
2.3 创建 Android 工程	22
2.3.1 创建 HelloAndroid 项目	22
2.3.2 Android 项目调试	25
2.3.3 Android 工程目录	25
第 3 章 Android 开发准备	27
3.1 Android 应用程序的组成	27

3.1.1	Activity	27
3.1.2	Broadcast Intent Receiver	29
3.1.3	Service	29
3.1.4	Content Provider	29
3.2	Android 的事件处理	29
3.2.1	事件监听简介	29
3.2.2	常用的事件监听	30
3.3	Intent 的简单应用	31
3.3.1	Intent 概述	31
3.3.2	Intent 实现多个 Activity 直接跳转的步骤	31
3.4	Android 应用程序的线程模型	31
第 4 章	Android 基本组件	33
4.1	UI 的基本元素	33
4.1.1	视图组件 (View)	33
4.1.2	视图容器组件 (Viewgroup)	33
4.1.3	布局组件 (Layout)	33
4.1.4	布局参数 (LayoutParams)	33
4.2	Android 中的 UI 布局	34
4.2.1	声明布局的方式	34
4.2.2	布局属性	34
4.2.3	Android 中的盒子模型	35
4.2.4	Android 中常见的布局	35
4.3	常用的 Widget 组件	38
4.4	菜单 (Menu)	44
4.4.1	菜单 (Menu) 简介	44
4.4.2	菜单 (Menu) 的创建方法	49
4.4.3	菜单 (Menu) 的事件处理	50
4.5	列表 (ListView)	52
4.5.1	列表 (ListView) 简介	52
4.5.2	简单 ListView 的创建方式	52
4.5.3	Adapter 接口	54
4.6	对话框 (Dialog)	57
4.6.1	对话框 (Dialog) 简介	57
4.6.2	创建 AlertDialog 解析常用的对话框方法	59
4.6.3	创建对话框 (Dialog)	61
4.6.4	对话框 (Dialog) 应用实例	61
4.7	Toast 和 Notification 的应用	65
4.7.1	Toast	65
4.7.2	Notification	65

4.7.3 Toast 与 Notification 应用实例	66
第 5 章 Intent 和 Broadcast 应用	75
5.1 Intent 简介	75
5.1.1 Intent 基础	75
5.1.2 用 Intent 启动新的 Activity	75
5.2 Intent 详解	78
5.2.1 操作 (Action)	79
5.2.2 数据 (Data) (与动作相关联的数据)	79
5.2.3 类型 (Type)	80
5.2.4 类别 (Category)	80
5.2.5 附件信息 (Extras)	80
5.2.6 目标组件 (Component)	81
5.3 解析 Intent	81
5.3.1 显式 Intent 与隐式 Intent	81
5.3.2 IntentFilter	81
5.4 Android 中的广播机制	84
5.5 Intent 实现广播案例	85
第 6 章 Android 的数据存储操作	89
6.1 Android 数据存储概述	89
6.2 Shared Preferences 存储	89
6.3 Files 存储	94
6.4 Network 存储	97
6.5 Android 数据库编程	100
6.5.1 SQLite 简介	100
6.5.2 SQLite 编程详解	100
6.6 Content Provider	108
6.6.1 数据模型	108
6.6.2 URI	108
6.6.3 查询	109
6.6.4 修改记录	110
6.6.5 添加记录	110
6.6.6 删除记录	111
6.6.7 创建 Content Provider	111
第 7 章 Service 应用	116
7.1 Service 概述	116
7.2 Service 的生命周期	116
7.3 Service 的使用	120
第 8 章 案例实践:《贪吃虎》游戏设计	129
8.1 构思	129
8.1.1 游戏的整体框架	129

8.1.2 游戏用到的 API	130
8.2 绘图	131
8.2.1 游戏 LOGO 的绘制	131
8.2.2 游戏菜单的绘制	132
8.2.3 游戏背景的绘制	133
8.2.4 游戏元素块的绘制	134
8.2.5 游戏人物的绘制	136
8.2.6 道具的绘制	138
8.3 逻辑	139
8.3.1 游戏 LOGO 的逻辑	139
8.3.2 游戏菜单的逻辑	139
8.3.3 游戏背景的逻辑	140
8.3.4 游戏元素块的逻辑	140
8.3.5 游戏人物的逻辑	142
8.3.6 道具的逻辑	145
8.4 游戏按键	146
8.4.1 游戏菜单的按键处理	146
8.4.2 游戏人物的按键处理	147
8.5 附件：源代码	148
8.5.1 GameActivity 类	148
8.5.2 GameView 类	148
8.5.3 Map 类	186
8.5.4 Npc 类	198
8.5.5 Bonus（道具）类	200
8.5.6 Hero 类	200
8.5.7 Tools（工具）类	205
8.5.8 Music 类	215
8.5.9 AndroidManifest.xml 文件	216
8.5.10 string.xml 文件	216
参考文献	217

第 1 章 Android 入门基础

1.1 Android 语言概述

1.1.1 Android 的概念

Android（安卓）是一套用于移动设备的软件平台，其中包括操作系统、中间件以及一些关键应用。Android SDK 基于 Java 开发语言，提供了在 Android 平台上进行应用开发的工具和相应的 API。其应用程序由用户利用 Java 自行开发。

1.1.2 Android 的发展简史

谈到 Android，首先需要了解“开发手机联盟（Open Handset Alliance）”。这是美国于 2007 年 11 月宣布组建的一个全球性联盟，这个联盟由包括中国移动、摩托罗拉、高通、宏达电子和 T-Mobile 在内的 30 多家技术和无线应用的领军企业组成。这一联盟共同开发名为 Android 的开放源代码的移动操作系统。

Android 是基于 Linux 平台的开源手机操作系统。它包括操作系统、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。通过运营商、设备制造商、开发商和其他有关各方的合作，建立了标准化、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。

Android 系统采用软件堆层（software stack，又名软件叠层）的架构，主要分为 3 个部分：低层以 Linux 核心工作为基础，只提供基本功能；其他应用软件由各公司自行开发，以 Java 作为编写应用软件的基础工具。

Android 的发展历程简述如下。

1. 开发手机联盟成立

2007 年 11 月 5 日，34 家公司宣布成立开发手机联盟（OHA）。

2. 发布第 1 版 Android SDK

2007 年 11 月 12 日，第 1 版 Android SDK 发布。

3. Android Market 上线

2008 年 8 月 28 日，为 Android 平台手机提供软件分发和下载的 Market 正式上线，并且迅速积累了大量应用。

4. T-Mobile G1 上市

2008 年 9 月 22 日，美国移动电话运营商 T-Mobile USA 在纽约正式发布了 T-Mobile G1 手机。该款手机为台湾宏达电（HTC）代工制造，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 7.2Mbps，并支持 Wi-Fi。

5. Android 1.0 SDK R1 发布

2008 年 9 月 23 日，Android 1.0 SDK R1 发布，标志着 Android 系统趋于稳定和成熟。

6. Android 宣布开放源代码

2008 年 10 月 21 日, Android 宣布开放源代码, 后来又发布了 Android 1.5 SDK Release 3。

1.2 Android 的体系结构

Android 作为一个移动设备的开发平台, 其软件层次结构分为操作系统 (OS)、中间件 (Middle-Ware) 和应用程序 (Application), 如图 1-1 所示。中间件包括应用程序框架 (Application Framework)、各种库 (Libraries) 和 Android 运行环境(Run-time), 简单说明如下。

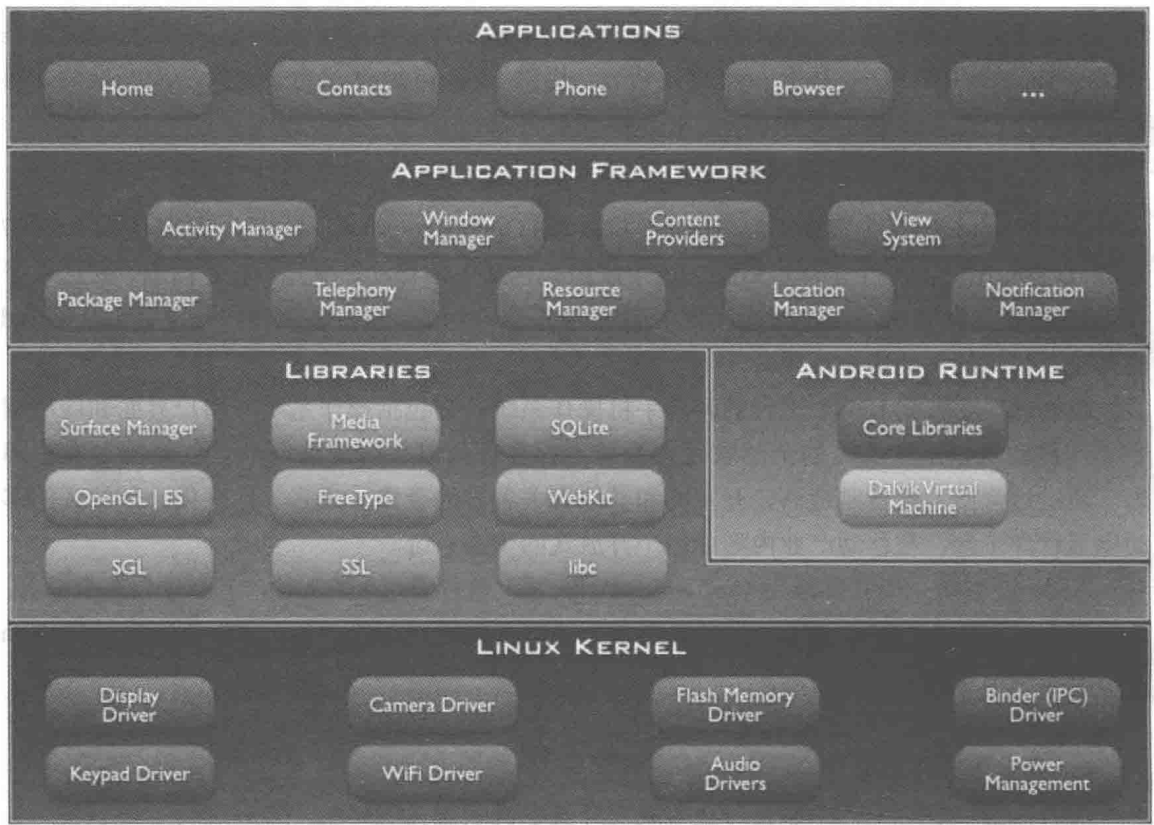


图 1-1

1.2.1 应用程序 (Application)

Android 的应用程序通常涉及用户界面和用户的交互, 是用户能够直接接触的部分。Android 通常将一系列核心应用程序包一起发布, 包括 email 客户端、SMS 程序、日历、地图、浏览器、联系人管理程序等。Android 平台下的所用应用程序都使用 Java 语言编写, 开发人员也可以使用应用程序框架的 API 开发自己的程序, 显示出 Android 巨大的应用潜力。

1.2.2 应用程序框架

应用程序框架 (Application Framework) 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, 并且任何其他的应用程序都可以使用这些功能模块 (应遵循框架的安全性限制)。同样地, 应用程序重用机制使用户可以方便地替换程序组件。

每个应用程序可能用到的应用框架列举如下。

(1) 丰富而又可扩展的视图 (View): 用于构建应用程序, 包括列表 (list)、网格 (grid)、文本框 (text box)、按钮 (button) 等, 以及可嵌入的 Web 浏览器。

(2) 内容提供者 (Content Provider): 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), 或者共享自己的数据。

(3) 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 file)。

(4) 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。

(5) 活动管理器 (Activity Manager): 用来管理应用程序生命周期, 并提供常用的导航返回功能。

在 Android 应用中, 每个应用程序一般由多个页面组成, 每个页面对应的 xml 文件对应一个 Activity。因此可以说, Android 的应用程序是由多个 Activity 的交互构成的。

1.2.3 库 (Libraries) 和 Android 运行环境 (Run-time)

Android 系统中包含一些 C/C++ 库, 能被不同的组件使用。它们通过 Android 应用程序框架为开发者提供服务。一些核心库简述如下。

(1) 系统 C 库: 一个 BSD 继承来的标准 C 系统函数库 (libc), 为基于 embedded Linux 的设备定制。

(2) 媒体库: 基于 PacketVideo OpenCORE, 支持多种常用的音频、视频格式回返和录制, 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。

(3) Surface Manager: 管理显示子系统, 并且为多个应用程序提供 2D 和 3D 图层的无缝融合。

(4) 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。

(5) SGL: 底层的 2D 图形引擎。

(6) 3D Libraries: 基于 OpenGL ES1.0 API 实现, 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。

(7) FreeType: 位图 (bitmap) 和矢量 (vector) 字体显示。

(8) SQLite: 一个对于所有应用程序可用, 且功能强劲的轻型关系数据库引擎。

Android 还有一个核心库, 提供 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 Dalvik 可执行文件 (.dex), 该格式的文件针对小内存使用做了优化。同时, 虚拟机基于寄存器, 所有的类都经由 Java 编译器编译, 然后通过 SDK 中的 dx 工具转换成 dex 格式, 由虚拟机执行。

1.2.4 操作系统 (OS)

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。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 内核同时作为硬件和软件栈之间的抽象层。

1.3 Android SDK

1.3.1 Android SDK 基础

Android SDK(Software Development Kit)提供在 Windows/Linux/Mac 平台上开发 Android 应用的组件，其中包含在 Android 平台上开发移动应用的各种工具集，不仅有 Android 模拟器和用于 Eclipse 的 Android 开发插件 ADT，而且有各种用于调试、打包和在模拟器上安装、应用的工具。

Android SDK 主要以 Java 语言为基础。通过 SDK 提供的一些工具，将其打包成 Android 平台使用的 apk 文件，然后使用 SDK 中的模拟器 (Emulator) 来模拟和测试该软件在 Android 平台上运行的情况和效果。

1.3.2 Android SDK 目录结构

Android SDK 1.1 的目录结构如图 1-2 所示。

Android SDK 1.5 的目录结构如图 1-3 所示。

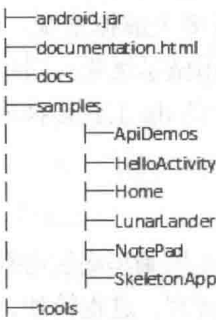


图 1-2

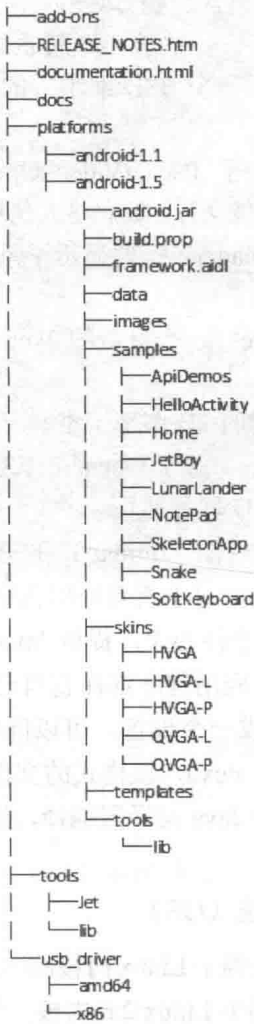


图 1-3

(1) Add-ons 目录下的 google_apis3 提供的 API 包, 主要包括 API Documentation.html 和 docs 目录下的文档

(2) RELEASE_NOTES.html 是 SDK 的发布说明。

(3) usb_driver 目录下包含 amd64 和 x86 的驱动文件。

(4) Tools 目录下包含一些通用的工具文件。

(5) 在 Platforms 目录下, 针对每个版本的 SDK, 提供了对应的 API 包及其示例。

① android.jar 是包含全部 API 的压缩包;

② samples 目录下是 SDK 附带的一些示例;

③ skins 目录下是其支持的几种外观像素;

④ templates 目录下是一些常用的文件模板;

⑤ tools 目录下是一些常用的辅助工具。

1.3.3 Android.jar 及内部结构

Android.jar 是一个标准 jar 压缩包, 其内部是编译后的 class 文件, 包含全部的 API, 结构如图 1-4 所示。

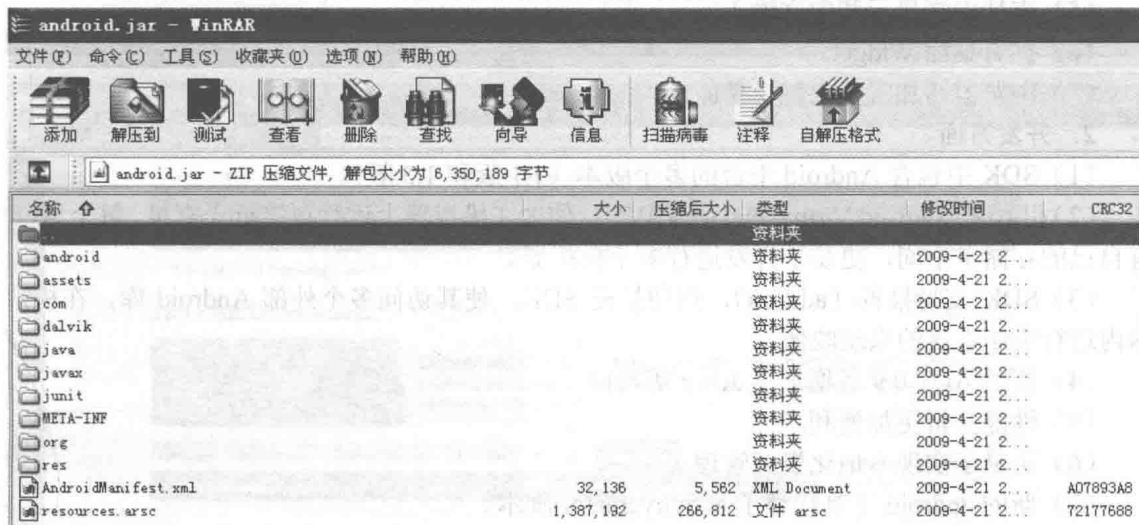


图 1-4

通过图 1-4 可以了解其模块的划分和结构, 有助于用户阅读和查找 SDK 文档。

1.3.4 Android API 核心开发包

SDK 参考文档是按照包结构组织的, 以使用户很清晰地看到 API 的结构。其核心包中的模块主要包括以下内容。

(1) android.app: 提供高层的程序模型, 提供基本的运行环境。

(2) android.content: 包含各种访问和发布设备上的数据的类。

(3) android.database: 通过内容提供者浏览和操作数据库, 包含底层 API 处理数据库。

(4) android.graphics: 底层图形库, 包含画布, 通过过滤、点、矩形, 将图形直接绘制到屏幕上, 并作为核心渲染包, 提供图形渲染功能。

(5) android.location: 定位和提供相关服务的类。

(6) android.media: 提供外挂多种音频、视频等媒体接口的类。

- (7) `android.net`: 通过的 `java.net.*` 接口, 提供帮助网络访问的类。
- (8) `android.os`: 提供系统服务、消息传输和进程间通信 (IPC)。
- (9) `android.provider`: 提供访问 Android 内容提供者的类。
- (10) `android.telephony`: 提供与拨打电话相关的 API 交互。
- (11) `android.view`: 提供基础的用户界面接口框架。
- (12) `android.util`: 涉及工具性的方法, 例如时间、日期的操作。
- (13) `android.widget`: 包含各种 UI 元素 (大部分是可见的), 在应用程序的屏幕中使用。
- (14) `android.webkit`: 包含一系列基于 Web 内容的 API。

1.3.5 Android SDK 1.5 的新特性

1. 系统方面

- (1) 采用当时最新的 Linux 内核 2.6.27 版本。
- (2) 精简了用户界面。
- (3) 拥有全新的视频录制功能, 可以上传视频到 Youtube, 上传照片到 Picasa。
- (4) 支持软键盘。
- (5) 支持中文显示和中文输入。
- (6) 拥有桌面 Widget。
- (7) 浏览器增加了多点触摸功能。

2. 开发方面

- (1) SDK 中包含 Android 平台的多个版本 (1.1 版和 1.5 版)。
- (2) 引入了 Android Virtual Devices (AVD), 使之在模拟器上运行更接近于真机。每个 AVD 有自己的存储卡空间, 更易于开发运行多个模拟器。
- (3) SDK 支持插件 (add-on), 以便扩充 SDK, 使其访问多个外部 Android 库, 在模拟器内运行定制系统的系统映像。
- (4) 新的 ADT 0.9 版增加了 JUnit 等功能。
- (5) 性能分析更加便利。
- (6) 更易于实现本地化资源管理。
- (7) 新的 `android` 工具代替了 `activitycreator` 脚本。

通过以上对 Android SDK 文档的介绍, 分析了 `android.jar` 文件, 读者应该大致了解了其内部 API 的结构和组织方式。如果想深入了解各个文件包含的 API 及其用法, 必须学会阅读和查找 SDK 文档。读者使用浏览器打开 SDK 目录下的 `documentation.html` 文件, 可详细阅读其内容。

第 2 章 Android 开发环境

2.1 Android 开发环境搭建

第 1 章介绍了 Android 的基础知识, 本章详细介绍搭建 Android 开发环境的步骤。

2.1.1 Android 开发系统要求

Android 开发系统要求如下:

- (1) Windows XP (32 位) 或 Vista (32 或 64 位)。
- (2) Mac OS X 10.4.8 或更高版本 (仅对 x86 版)。
- (3) Linux (在 Linux Ubuntu Dapper Drake 上经过测试)。

2.1.2 下载所需软件包

在 <http://java.sun.com/javase/downloads/index.jsp> 下载 JDK, 如图 2-1 所示。



图 2-1

在 <http://www.eclipse.org/downloads/> 下载 Eclipse, 如图 2-2 所示。

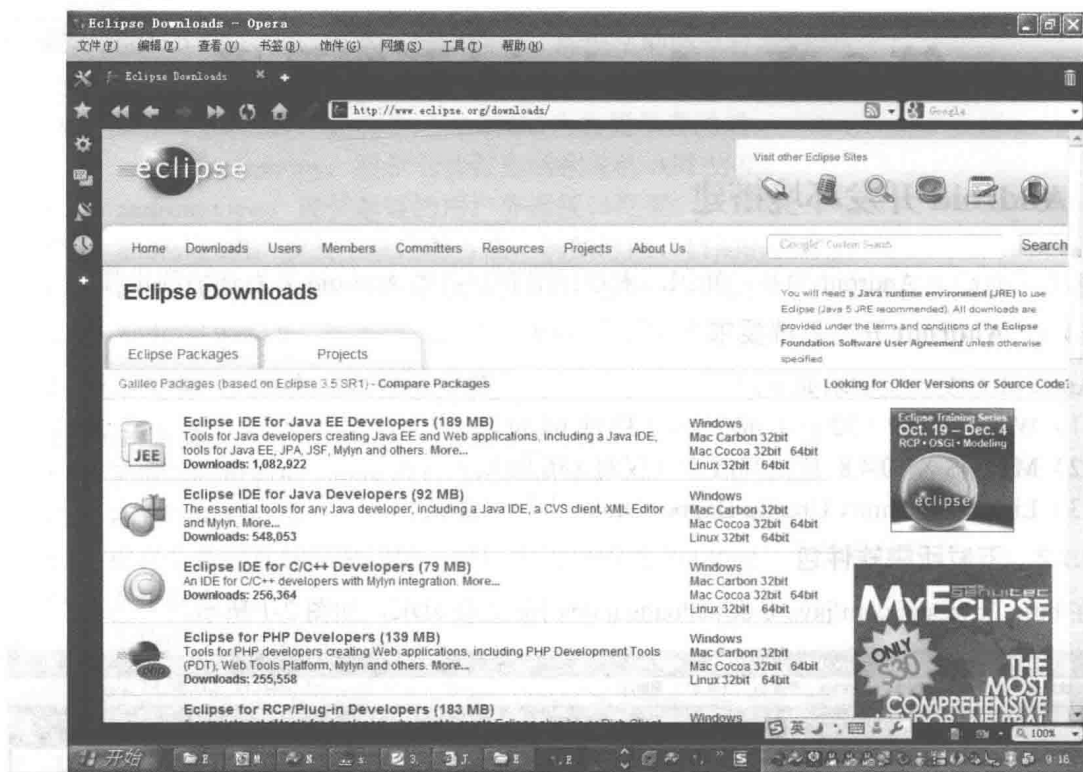


图 2-2

在 <http://developer.android.com/sdk/index.html> 下载 Android SDK, 如图 2-3 所示。



图 2-3

2.1.3 安装 Android SDK

Android SDK 同 Eclipse 一样，直接解压缩就可以打开。假定下载的压缩包解压缩到文件夹 F:\Android 中，如图 2-4 所示。



图 2-4

将 Android SDK 中 tools 目录的绝对路径添加到系统 PATH 中。然后，打开“系统属性”对话框，单击“环境变量”按钮，如图 2-5 所示。

在“环境变量”对话框中，将 PATH 的值设置为 SDK 中 tools 的绝对路径，如图 2-6 所示。

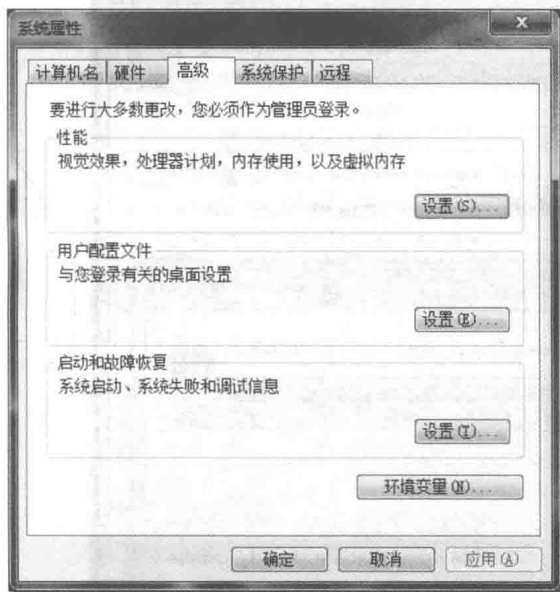


图 2-5



图 2-6

单击“确定”按钮，重新启动计算机，进入 cmd 命令窗口，检查 SDK 是否安装成功。运行 `android -h`，如果输出如图 2-7 所示，表明安装成功。

2.1.4 安装 ADT

(1) 启动 Eclipse，然后单击菜单中的 Help→Install New Software，弹出 Available Software 对话框，如图 2-8 所示。