



普通高等教育“十二五”计算机类规划教材

Android

程序设计教程

© 丁山 编

*Android Chengxu
Sheji Jiaocheng*



附赠光盘



机械工业出版社
CHINA MACHINE PRESS

普通高等教育“十二五”计算机类规划教材

Android 程序设计教程

丁 山 编



机械工业出版社

本书介绍了基于 Android 系统的程序设计技术, 内容涵盖了 Android 相关的各个领域, 本书大致可以分成两个部分, 第一部分为理论篇, 主要介绍了 Android 操作系统概况、Android 生命周期与组件通信、Android 用户界面设计、Android 数据存储与交互、Android 后台服务与事件广播、媒介与网络、Android NDK 等内容。第二部分为实践篇, 主要介绍了 Android 通信应用、定位与 Google 地图开发、语音与短信服务、Android 传感器应用等内容。

本书内容丰富, 叙述浅显易懂, 程序实例具有典型性, 随书光盘中收录了本书所有例题的源代码, 可供读者参考和学习。本书配有免费的教学课件, 欢迎选用本书作为教材的教师登录机械工业出版社教材服务网 www.cmpedu.com 下载或发邮件到 wangkang_maizi9@126.com 索取。

本书可作为高等院校电子信息类、计算机科学与技术等专业高年级学生和研究生教材, 也可以作为学习 Android 系统程序设计的工程技术人员参考用书。

图书在版编目 (CIP) 数据

Android 程序设计教程/丁山编. —北京: 机械工业出版社, 2014. 12
普通高等教育“十二五”计算机类规划教材
ISBN 978-7-111-48823-1

I. ①A… II. ①丁… 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 (2014) 第 290190 号

机械工业出版社 (北京市百万庄大街 22 号 邮政编码 100037)

策划编辑: 王 康 责任编辑: 王 康 刘丽敏

版式设计: 常天培 责任校对: 王 欣

封面设计: 张 静 责任印制: 刘 岚

涿州市京南印刷厂印刷

2015 年 2 月第 1 版第 1 次印刷

184mm×260mm·20.5 印张·496 千字

标准书号: ISBN 978-7-111-48823-1

ISBN 978-7-89405-589-7 (光盘)

定价: 42.00 元 (含 1DVD)

凡购本书, 如有缺页、倒页、脱页, 由本社发行部调换

电话服务

网络服务

服务咨询热线: 010-88379833 机工官网: www.cmpbook.com

读者购书热线: 010-88379649 机工官博: weibo.com/cmp1952

教育服务网: www.cmpedu.com

封面无防伪标均为盗版

金书网: www.golden-book.com

前 言

Android 是 Google 于 2007 年 11 月推出的一款开放的嵌入式操作系统平台，由于其完全开源的特性，广泛应用于手机、平板电脑、家电及其他嵌入式系统设计中，车载设备、智能电视、VoIP 电话和医疗设备等厂商纷纷推出 Android 系统产品。Android 正以空前的速度吸引着大批开发者的加入，尤其是应用开发工程师。本书以 Android 系统的程序设计开发为主体，并结合真实的案例向读者介绍 Android 的基本组件的使用及程序开发的整个流程，内容涵盖了所有和 Android 相关的领域。

本书力求全面、实用，对例题做到详细分析和解释，既可以帮助读者学习理解知识和概念，降低学习难度，又具有启发性，帮助读者更加轻松、迅速地理解和掌握本书内容。

本书在内容的组织上分为理论篇和实践篇共 11 章，其中第 1~7 章为理论篇，第 8~11 章为实践篇。各章的具体内容如下：

第 1 章主要对 Android 的发展、特点、环境搭建和体系结构进行简要介绍。并且讲解了 JDK、Eclipse、Android SDK 软件的下载及安装的基本知识。对 Android 应用程序进行解析，提高读者对程序的创建、目录的结构、资源的管理以及对程序权限的理解。最后讲解如何调试 Android 程序。

第 2 章主要讲述了 Android 生命周期和组件之间的通信。生命周期主要讲述了 Android 四大组件之一的 Activity 生命周期，包括生命周期函数、栈结构和基本状态三方面。组件的通信靠 Intent 实现，以及 Intent 基本构成。

第 3 章主要从 Android 用户界面开发出发，讲述了开发过程中经常使用到的控件，包括菜单、常用基础控件、对话框与消息框。界面中控件的结构及位置等需要通过有效的界面布局控制，Android 中提供了 5 种界面布局格式，即线性布局、相对布局、表格布局、绝对布局和框架布局。界面中还有一种必要的操作处理——外部操作的响应，通过有效的事件机制完成。

第 4 章主要讲述了 Android 数据存储与交互方面的内容，系统中数据交互主要通过五种方式实现，共享优先数据机制、SQLite 数据库、File 文件机制、内容提供者控件和网络存储。其中在应用程序中最常用也是最有效的数据交互方式是使用 SQLite 数据库。

第 5 章主要讲述了 Android 后台服务、事件广播和常驻程序。后台服务由系统提供的 Service 组件实现，可分为本地服务和远程服务。事件广播机制主要依靠 Broadcast Receiver 组件实现。常驻程序 AppWidget 又称为窗口小部件，是在 HomeScreen 上显示的小部件，开发时常用 AppWidgetProvider 和 AppWidgetProviderInfo 类实现。

第 6 章介绍了 Android 平台下通过程序实现音频、视频播放等操作。分别介绍了从源文件播放、文件系统播放和流媒体播放等方式。并且介绍了 Android 图形绘制与特效，包括图形的平移、旋转及缩放等操作，保存指定格式图形文件。编写专业的绘图或控制图形动画的应用程序。以及如何使用 Android 手机中内置的高性能 WebKit 内核浏览器浏览网页，使用 HTTP 和 URL 获得网络资源等内容。

第 7 章介绍了 Android NDK 的相关知识，从 NDK 的简单介绍到开发环境的配置，以及开发流程。使用 NDK 实现一些对代码性能要求较高的模块并将这些模块嵌入到 Android 应用程序中会大大地提高程序效率，比如用 NDK 开发 OpenGL。此外，如果项目中包含了大量

的逻辑计算或者是 3D 特效，这时 Android NDK 便会显示出它超强的功能。

第 8 章介绍了 Android 平台下的几种通信方式，即 Socket 通信、蓝牙及 WiFi。其中对它们通信方式，通信中所需的各种 API 及其使用方法进行了介绍。在 Socket 通信中主要介绍了它的通信模型以及通信各部分的实现并通过实例展示了其具体的通信过程。在蓝牙中主要介绍了蓝牙系统的基本构成，在 Android 下的各种 API 及通信方式。最后对 WiFi 的操作做了详细的介绍。

第 9 章介绍了 GPS 的概念、系统架构以及底层驱动的编写，并通过例子讲解了 GPS 在 Android 上的应用。

第 10 章介绍了 Android 中对语音及短消息的访问。其中重点介绍了利用 Telephony 类来监听来电与去电信息；利用 SMSManager 来发送和接收短消息，并用 PendingIntent 对发送消息进行跟踪。

第 11 章介绍了 Android 系统所支持的传感器类型，如何使用传感器 API 来获取传感器数据，如何通过 SensorManager 来注册传感器监听器，如何在 SensorEventListener 中对传感器进行监听，如何使用几种常用的传感器等。最后通过使用最常用的加速度传感器开发有趣的应用来进一步介绍传感器开发的流程。

本书内容充实，系统全面，重点突出。阐述循序渐进，由浅入深。书中所有例题均在 Eclipse + ADT 环境下运行通过。本书配有免费的电子课件，欢迎选用本书作为教材的老师登录 www.cmpedu.com 下载或发邮件到 wangkang_maizi9@126.com 索取。

参加本书编写、校对及程序测试工作的还有研究生刘海瑞与马文翼，在此表示感谢。

由于作者水平有限，书中难免有错误和不足之处，恳请各位专家和读者批评指正。

编 者

目 录

前言

理 论 篇

第 1 章 Android 操作系统	1
1.1 Android 简介	1
1.1.1 Android 的起源	1
1.1.2 Android 的发展史	1
1.1.3 Android 的特点	2
1.2 Android 体系结构	3
1.3 环境搭建及环境配置	4
1.3.1 JDK 的下载和安装	4
1.3.2 下载安装 Eclipse	7
1.3.3 SDK 的下载和安装	8
1.3.4 创建 AVD	11
1.3.5 安装 ADT	13
1.4 Android 应用程序解析	14
1.4.1 创建一个 Android 应用	14
1.4.2 目录结构	16
1.4.3 Android 中的资源访问	17
1.5 Android 系统的调试与下载	27
本章小结	30
习题	30
第 2 章 Android 生命周期与组件通信	31
2.1 Android 生命周期	31
2.2 Activity 组件	32
2.2.1 Android 组件简介	32
2.2.2 Activity 生命周期	33
2.2.3 Task 与 Activity 栈	43
2.2.4 Activity 基本状态	44
2.3 Intent 信使	46
2.3.1 Intent 基本构成	46
2.3.2 Intent 形式	48
2.3.3 Intent 过滤器	50
2.3.4 Activity 信息传递	53

本章小结	59
习题	59
第 3 章 Android 用户界面设计	60
3.1 菜单	60
3.1.1 选项菜单 (Option Menu)	60
3.1.2 上下文菜单 (Context Menu)	63
3.1.3 子菜单 (Sub Menu)	65
3.2 常用基础控件	68
3.2.1 列表视图	68
3.2.2 文本框类	70
3.2.3 按钮类	77
3.2.4 时钟控件类	83
3.2.5 日期与时间类	84
3.2.6 计时控件	85
3.2.7 进度条控件	86
3.2.8 拖动条控件	89
3.2.9 下拉列表控件	92
3.3 对话框和消息框	94
3.3.1 对话框	94
3.3.2 消息框	98
3.4 界面布局	101
3.4.1 线性布局	101
3.4.2 相对布局	103
3.4.3 表格布局	105
3.4.4 绝对布局	107
3.4.5 框架布局	109
3.5 事件处理机制	110
3.5.1 事件处理模型	110
3.5.2 事件处理函数	111
本章小结	113
习题	114
第 4 章 Android 数据存储与交互	115
4.1 共享优先数据存储	115
4.2 数据库存储	121



4.2.1 嵌入式数据库 121

4.2.2 Android SQLite 数据库 125

4.3 文件存储 137

4.3.1 内部存储 137

4.3.2 SD 卡存储 139

4.3.3 资源文件访问 142

4.4 内容提供者 144

4.4.1 内容解析器 145

4.4.2 内容提供者 145

4.5 网络存储 155

本章小结 156

习题 157

第 5 章 Android 后台服务与事件

广播 158

5.1 Service 进程服务 158

5.1.1 Service 组件生命周期 158

5.1.2 Service 服务 159

5.2 BroadCast Reciver 广播 168

5.3 AppWidget 常驻程序 169

5.3.1 App Widget 框架 170

5.3.2 App Widget 创建 171

本章小结 173

习题 173

第 6 章 媒介与网络 174

6.1 Android 的音频与视频 174

6.1.1 Android 音频/视频播放状态 174

6.1.2 Android 的音频播放 177

6.1.3 Android 视频播放 185

6.2 Android 图形绘制与特效 187

6.2.1 几何图形绘制类 187

6.2.2 图形绘制过程 189

6.2.3 图形特效 195

6.3 Web 视图 197

6.3.1 浏览器引擎 WebKit 198

6.3.2 Web 视图对象 198

6.3.3 Web 视图实例 199

6.4 HTTP 和 URL 网络资源获取 201

本章小结 206

习题 207

第 7 章 Android NDK 208

7.1 Android NDK 简介 208

7.2 构建 NDK 系统 209

7.2.1 Android NDK 开发环境构建 209

7.2.2 解析 hello-jni 例程 212

7.3 NDK 开发过程详解 215

7.3.1 中间件的概念 215

7.3.2 Android 系统的中间件 216

7.3.3 使用 C/C++ 实现本地方法 218

7.3.4 依赖关系建立 220

7.3.5 NDK 程序的链接与运行 223

本章小结 225

习题 225

实 践 篇

第 8 章 Android 通信应用 226

8.1 Socket 通信 226

8.1.1 Socket 简介 226

8.1.2 Socket 通信模型及重要的 API 227

8.1.3 ServerSocket 类 227

8.1.4 Socket 连接过程 230

8.1.5 Android 中的 Socket 通信 230

8.2 蓝牙通信 237

8.2.1 蓝牙简介 237

8.2.2 蓝牙系统的组成 239

8.2.3 蓝牙技术的特点 239

8.2.4 Android 蓝牙驱动架构 240

8.2.5 蓝牙在 Android 下的应用 243

8.3 WiFi 通信 255

8.3.1 WiFi 包 255

8.3.2 网卡状态 256

8.3.3 WiFi 网卡操作权限 256

8.3.4 更改 WiFi 状态 256

本章小结 263

习题 263

第 9 章 定位与 Google 地图开发 264

9.1 使用 GPS 定位 264

9.2 Google 地图的使用 270

9.2.1 Google Maps 包 270

9.2.2 获得 Map API Key	270	习题	299
9.2.3 Android Google Map 基于程序	274	第 11 章 Android 传感器应用	300
9.3 GPS 与 Google 地图结合	277	11.1 利用 Android 传感器	300
本章小结	282	11.1.1 传感器的定义	300
习题	282	11.1.2 Android 中传感器关联类和 接口	300
第 10 章 语音与短信服务	283	11.2 Android 中常用的传感器	307
10.1 电话服务的硬件支持	283	11.2.1 感知环境	307
10.2 Android 系统电话服务框架	283	11.2.2 感知设备方向和运动	308
10.3 语音服务	284	11.3 传感器应用案例	312
10.3.1 TelephoneManager 类	285	11.3.1 Android 加速度传感器应用一—— 实现手机摇一摇控制音乐播放	312
10.3.2 访问电话服务的属性及状态	285	11.3.2 Android 加速度传感器应用二—— 重力小球	314
10.3.3 监听来电信息	288	本章小结	317
10.3.4 监听去电信息	290	习题	317
10.4 短消息服务	291	参考文献	318
10.4.1 SMS 和 MMS 简介	291		
10.4.2 SMS 消息的发送与跟踪	292		
10.4.3 SMS 消息的接收	294		
本章小结	299		

理 论 篇

第 1 章 Android 操作系统

作为首个真正完全开放的手机移动平台，Android 以其开源性及强劲的功能被称为目前世界上最为流行的手机操作系统，本章主要对 Android 的发展、特点、环境搭建和体系结构进行简要介绍。

1.1 Android 简介

1.1.1 Android 的起源

Android 系统最初由安迪·鲁宾 (Andy Rubin) 等人开发研制，最初是为创建一个数码相机的先进操作系统；但是后来发现市场需求不够大，加上智能手机市场的快速成长，于是 Android 被改造成一款面向智能手机的操作系统，2005 年 8 月 Android 被 Google 收购。

Android 一词英文本意为“人形机器人”，2007 年 11 月 5 日，Google 公司正式对外展示了基于 Linux 内核的开放源代码移动设备操作系统并宣布将其命名为 Android。自此 Android 便以最具开放性的手机平台开发系统在操作系统中崭露头角。为了更好地开发与推广 Android 手机操作系统，2007 年 11 月 Google 公司与 34 家手机相关企业携手建立了开放手机联盟 (Open Handset Alliance, OHA)。开放手机联盟的组成成员涵盖了手机和其他终端制造商、移动运营商、半导体厂商、软件厂商，例如 HTC (宏达)、SAMSUNG (三星)、MOTO (摩托罗拉)、CHINA MOBILE (中国移动) 等知名品牌公司。然而苹果、微软、诺基亚、RIM、黑莓等公司并没有在联盟成员之列。开放手机联盟建立的目的是支持谷歌可能发布的手机开发系统和应用软件，并共同开发名为 Android 的开放源代码移动系统，开发多种技术，以大幅削减移动设备和服务的开发和推广成本。随后，Google 以 Apache 免费开放源代码许可证的授权方式，发布了 Android 的源代码。让生产商推出搭载 Android 操作系统的智能手机，随着 Android 操作系统的逐步完善和发展，Android 操作系统后来逐渐扩展到平板电脑及其他领域。

1.1.2 Android 的发展史

Android 的发展与 Google 密不可分。2007 年 Google (谷歌) 在收购了 Android 公司后，将其特有的谷歌服务与 Android 手机操作系统相连，赋予了 Android 操作系统全新的灵魂。

谷歌移动服务 (Google Mobile Service, GMS) 是谷歌的一项服务，该服务致力于让用户在利用移动电话或其他移动设备时能够使用谷歌搜索、谷歌地图、Gmail、YouTube、Android Market 等谷歌服务产品。谷歌公司将 GMS 内嵌到 Android 手机系统中，并同时给予 An-



droid 手机生产商不同程度的授权。由于 GMS 的强大功能，GMS 成为 Android 操作系统的灵魂，大部分用户使用 Android 手机，其实就是为了使用谷歌所提供的各项服务。

Android 1.5 版本发布后，Android 手机操作系统便开始以甜点命名。继 2008 年 9 月 Android 1.1 发布后又陆续发布了 Android 1.5 Cupcake（纸杯蛋糕）、Android 1.6 Donut（甜甜圈）、Android 2.0/2.1 Éclair（法式奶油夹心甜点，也译为松饼）、Android 2.2 Froyo（冻酸奶）、Android 2.3 Gingerbread（姜饼）、Android 3.0/3.1/3.2 Honeycomb（蜂巢）、Android 4.0 Ice Cream Sandwich（冰激凌三明治）、Android 4.1/4.2/4.3 Jelly Bean（果冻豆）。到目前为止，最新的 Android 版本是 Android 4.4 KitKat（奇巧）。

Android 操作系统是基于 Linux 的开源操作系统。除了应用于手机移动平台外，它在平板电脑方面的应用也日趋广泛，并且有报道称 Android 开始应用于军事方面。

1.1.3 Android 的特点

Android 操作系统为何会脱颖而出，这与市场的选择性和 Android 的优越性紧密相关。

手机从其出现到目前市场上流行的形形色色的功能样式，最受人们欢迎并且被人们普遍使用的是功能手机和智能手机。功能手机包含了基本的手机应用并集成了一定的手机扩展功能，然而功能手机和智能手机并不可混为一谈。功能手机所具有的功能是手机生产厂商利用 Java 软件集成于手机之中的，而智能手机内置与计算机类似的操作系统，我们可以根据需要下载和安装不同的软件来扩展手机的功能，而不受手机生产厂商的限制。所谓智能手机，即“像个人电脑一样，具有独立的操作系统，可以由用户自行安装软件、游戏等第三方服务商提供的程序，通过此类程序来不断对手机的功能进行扩充，并可以通过移动通信网络来实现无线网络接入的这样一类手机的总称”。

智能手机的特点如下：

(1) 拥有一般手机的全部功能，如能够进行正常的手机通话、收发短信等手机通信功能，并且具备智能手机的典型功能之一，即多任务功能和复制粘贴功能。

(2) 具备手机无线网络接入能力和掌上电脑的功能。

(3) 具备手机功能扩展能力。使用者可以根据自身的需要下载安装相应功能软件，从而达到扩展手机功能的目的。

目前市场上的智能手机操作系统主要有开放手机联盟的 Android 操作系统、诺基亚的 Symbian 操作系统、苹果的 iOS 操作系统、RIM 的 Blackberry 操作系统等。自 iOS 操作系统和 Android 操作系统推进市场以后，Symbian 操作系统的市场份额已被占去大多半。

和 Symbian 和 iOS 系统比较起来，Android 系统的特点及优势如下：

(1) 可自动切换无线网络，节省上网费用。

(2) 操作界面更简洁、个性，与实际使用联系紧密；更易上手，操作方便。

(3) 互联网连接使用简单便捷，可称之为最佳的互联网移动终端。

(4) 支持多任务运行，切换简单快捷，流畅无阻。

(5) 支持与微软 Exchange 的同步，办公娱乐两不误。

(6) 全新开源系统，软件数量和增长速度远超过 Windows Mobile。无“证书”限制，安装软件更自由，系统发展更具前景。软件安装卸载更方便，无需第三方平台软件。

(7) 强大的 Linux 内核，内存管理更优秀，不容易死机。

1.2 Android 体系结构

Android 是基于 Linux 内核并开放源代码的移动平台操作系统，它采用了分层式架构，主要分为四层，从低到高依次为 Linux 核心层（LINUX KERNEL）、系统运行层（LIBRARIES）、应用程序框架层（APPLICATION FRAMEWORK）及应用程序层（APPLICATIONS）。其系统架构如图 1-1 所示。

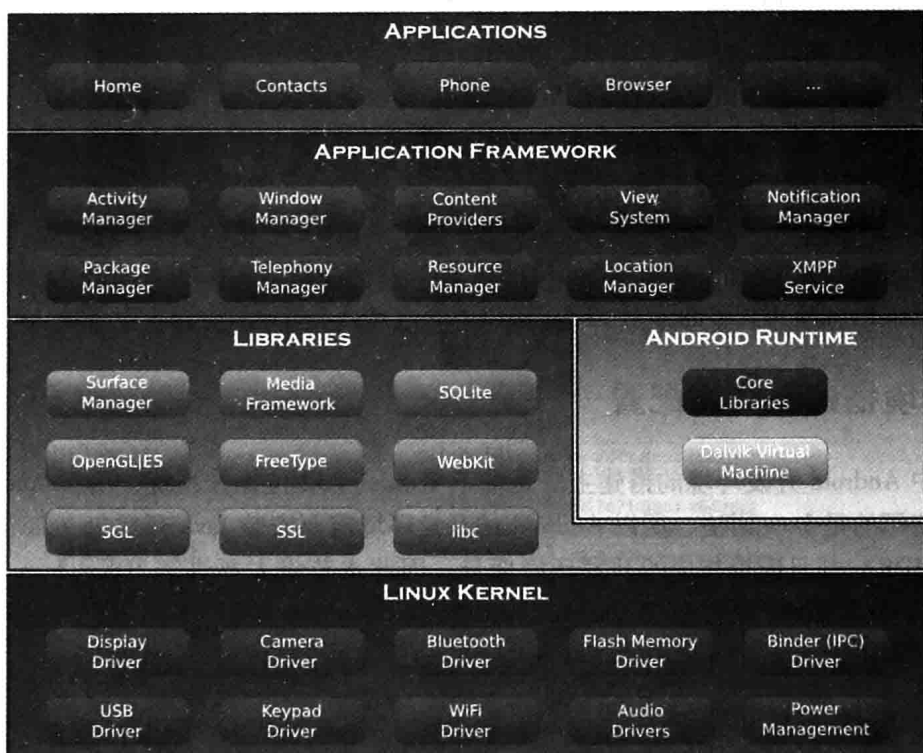


图 1-1 Android 的系统架构

Android 的分层结构特点是低层为高层提供统一的服务，高层使用低层所提供的服务，并屏蔽低层和本层的差异。当低层结构发生变化时并不会影响高层的结构及功能的实现。下面将从底层 Linux 核心层到高层应用程序层，依次讲解各层的功能及特点。

由于 Android 是以 Linux 为内核开发的。对于 Linux 核心层（LINUX KERNEL），其主要功能是隐藏硬件特性并为高层提供一系列统一的系统服务及驱动程序，例如内存管理、进程管理、网络协议栈等系统服务及 WiFi 驱动、蓝牙驱动等相关功能的驱动程序。核心层结构如图 1-1 中 LINUX KERNEL 部分所示。

Linux 核心层是用户空间和内核空间的分界线。并且仅 Linux 核心层属于内核空间，其余三层均属于用户空间。Linux 核心层是硬件和软件之间的抽象层，通过 C 语言开发并提供其基本功能。

再向上一层主要是由系统运行层（LIBRARIES）构成，还包括虚拟机（Virtual Machine）。系统运行层的基本结构如图 1-1 中 LIBRARIES 部分所示。



由于 Android 系统需要支持 Java 代码的运行，系统运行层主要包括系统运行库 LIB 和 Android 运行环境，通常称之为中间层。与移动设备的平台应用相关，并通过 Java 调用接口函数实现与上层之间的通信。该层由 C/C++ 开发，主要包括媒体库、Web 浏览引擎、关系数据库引擎、图形库等功能。Android 运行环境以虚拟机 Dalvik Virtual Machine 为主，虚拟机以编码格式运行。其中 Dalvik 虚拟机和标准的 Java 虚拟机之间的区别主要体现在，Dalvik 是基于寄存器的，而标准的 Java 虚拟机是基于堆栈的，并且 Dalvik 经过优化可在有限的内存中同时运行多个虚拟机程序实例，每一个 Dalvik 应用可作为一个独立的 Linux 进程执行。

再向上便是应用程序框架层 (APPLICATION FRAMEWORK)。该层包括所有开发所用的 SDK 类库和某些未公开的接口类库，是 Android 移动平台的核心机制的体现。应用程序框架层的结构如图 1-1 中 APPLICATION FRAMEWORK 部分所示。

最上层是应用程序层，包括各种应用软件，如通话程序、短信程序等。应用程序层结构如图 1-1 中 APPLICATIONS 部分所示。

应用软件由各公司自行开发，以 Java 作为编写程序的一部分。从低层到高层，第二层和第三层之间，是本地代码层和 Java 代码层的接口。第三层和第四层之间，是 Android 的系统 API。对于 Android 应用程序的开发，第三层以下的内容是不可见的，仅考虑系统 API 即可。

1.3 环境搭建及环境配置

本书中 Android 开发环境的搭建主要是基于 Windows 32 位操作系统，由于 Android 的应用框架层使用的是 Java 语法，所以 Android 的开发环境需要安装 Java 开发包 (Java Development Kit, JDK) 并且配置相应的开发环境变量。因此本节将主要讲解 JDK、Java 程序开发环境 Eclipse 和 Android 开发包 (Android Software Development Kit, 即 Android SDK) 的下载及安装的基本知识。

1.3.1 JDK 的下载和安装

在安装 JDK 软件前首先到相应的软件下载地址下载其对应的安装软件。下载地址为：<http://Java.sun.com/javase/downloads/index.jsp>。下载 JDK 方法如下。打开对应链接，在弹出的页面中将会看到很多下载版本，选择对应下载版本，此处选择 Java SE 7。接着将会进入 Java SE Downloads 的下载页面如图 1-2 所示。

此页面还包括其他与 Java 安装相关的下载项，可根据自身需要进行下载，主要使用的是 JDK (其中 JDK 中包含 JRE)，单击 Download 进行下载。

将会出现如图 1-3 所示下载页面。

在注册登录后便可以下载和自己计算机所安装的系统对应的 JDK 版本，在此处使用的是 Windows x86 下所对应的安装版本。下载完成后单击应用程序文件进行安装 (注意在此处程序的安装路径可在安装过程中自行设定)。

本书附有已下载好的 JDK 的安装包，此处以安装包内应用程序进行安装，双击安装包中的 JDK 安装应用程序 `jdk-7u51-windows-i586.exe`，将会弹出如图 1-4 所示的安装对话框。选择“接受”选项，然后在弹出的对话框中依次单击“下一步”。安装过程如图 1-5 ~ 图 1-8 所示。本软件默认安装在 `C:\Program Files\Java\jdk1.7.0_51\` 中。

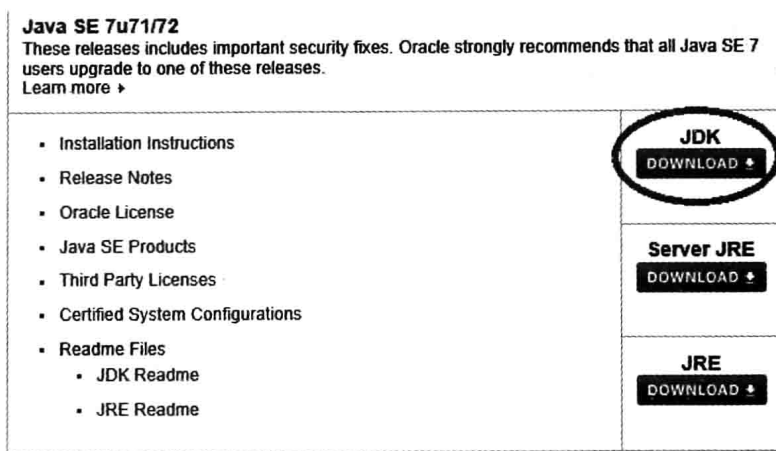


图 1-2 JDK 的下载页面 1

Java SE Development Kit 7u71

You must accept the Oracle Binary Code License Agreement for Java SE to download this software.

Thank you for accepting the Oracle Binary Code License Agreement for Java SE; you may now download this software.

Product / File Description	File Size	Download
Linux x86	119.44 MB	jdk-7u71-linux-i586.rpm
Linux x86	136.76 MB	jdk-7u71-linux-i586.tar.gz
Linux x64	120.81 MB	jdk-7u71-linux-x64.rpm
Linux x64	135.63 MB	jdk-7u71-linux-x64.tar.gz
Mac OS X x64	185.84 MB	jdk-7u71-macosx-x64.dmg
Solaris x86 (SVR4 package)	139.36 MB	jdk-7u71-solaris-i586.tar.Z
Solaris x86	95.48 MB	jdk-7u71-solaris-i586.tar.gz
Solaris x64 (SVR4 package)	24.68 MB	jdk-7u71-solaris-x64.tar.Z
Solaris x64	16.36 MB	jdk-7u71-solaris-x64.tar.gz
Solaris SPARC (SVR4 package)	138.74 MB	jdk-7u71-solaris-sparc.tar.Z
Solaris SPARC	98.62 MB	jdk-7u71-solaris-sparc.tar.gz
Solaris SPARC 64-bit (SVR4 package)	23.94 MB	jdk-7u71-solaris-sparcv9.tar.Z
Solaris SPARC 64-bit	18.35 MB	jdk-7u71-solaris-sparcv9.tar.gz
Windows x86	127.78 MB	jdk-7u71-windows-i586.exe
Windows x64	129.52 MB	jdk-7u71-windows-x64.exe

图 1-3 JDK 的下载页面 2

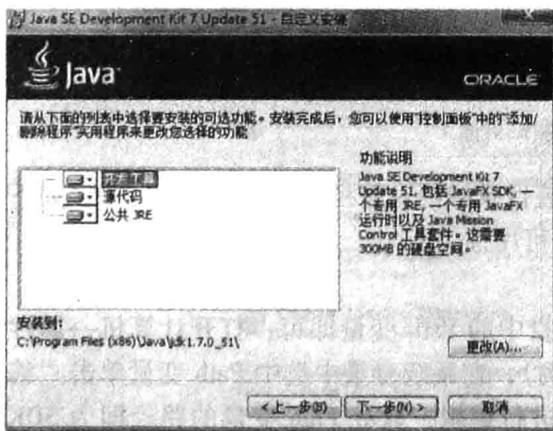


图 1-4 JDK 的安装

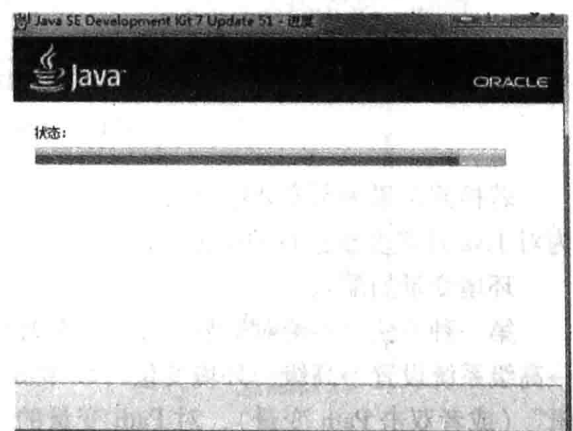


图 1-5 JDK 的安装

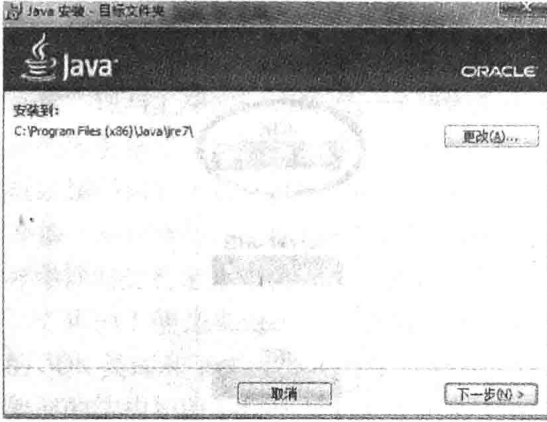


图 1-6 JDK 的安装

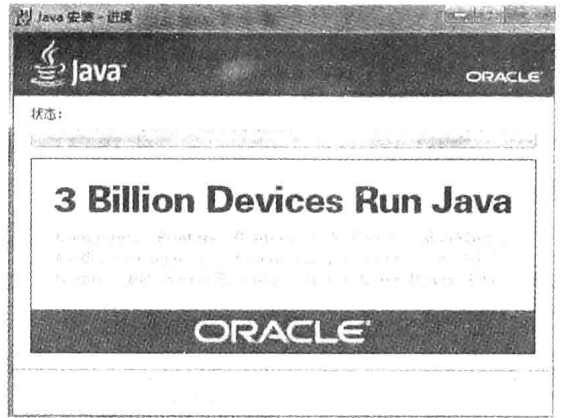


图 1-7 JDK 的安装

注意在安装过程中，将最后一步中显示上述文件前边的勾去掉，然后安装完成。

在 JDK 安装完成后需要对其进行进一步的检测以确定软件成功安装。具体的检测方法如下所述。

单击“开始”→“运行”（对于 Win7 可直接通过组合键 < Win + R > 实现），在弹出的对话框中输入“cmd”并按 < Enter > 键确定后，将会弹出 cmd 命令器窗口。在弹出的 cmd 窗口中输入 java - version（注意 java 后面有一个空格）并按 Enter 确定。若显示如下信息则表示安装成功，如图 1-9 所示。

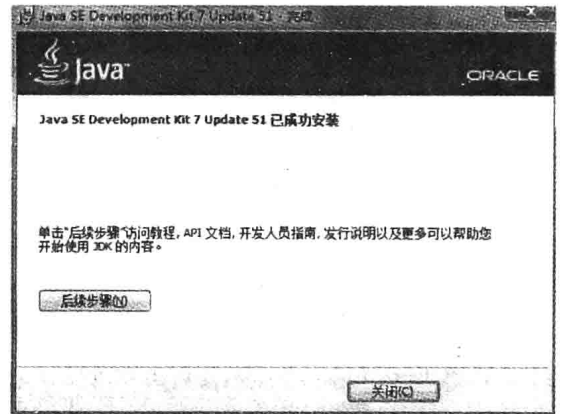


图 1-8 JDK 的安装



图 1-9 JDK 安装测试页面 1

若检测结果表示安装不成功，则需要将其安装目录的绝对路径添加到系统路径中，如下为对 Java 环境变量进行的配置以保证 Java 程序的可用性。

环境变量的配置方法有二：

第一种方法是将绝对安装地址加入到环境变量中的 Path 变量即可：打开计算机→属性→高级系统设置→高级→环境变量（以 Win7 为例），在系统变量中选中 Path 变量单击“编辑”（或者双击 Path 变量），对 Path 变量的路径进行编辑，此处 Path 变量的路径即为 SDK 的安装地址及 Java 的安装地址。例如地址(;F:\Android-sdk-windows\tools;C:\Program Files\

Java\jdk1.7.0_51\bin;)。

第二种方法是设置 JAVA_HOME、Path、ClassPath 这三个系统变量的路径（方法同上）。在系统变量处选择新建，在新建变量对话框中变量名输入 JAVA_HOME，变量值处输入 Java 的安装地址，此处为 C:\Program Files\Java\jdk1.7.0_51；新建系统变量 ClassPath 并将变量值赋为 Win7→Win7；编辑 Path 变量将其赋值为 .;%JAVA_HOME%\bin；。

此处 JAVA_HOME 表示 Java 的安装路径，即将变量值写为其安装路径即可；Path 变量的含义是系统在任何路径下都可以识别 Java 命令；ClassPath 为 Java 加载类（class or lib）路径，只有类在 ClassPath 中，Java 命令才能识别。

在配置好环境变量的前提下，再次检测环境变量是否配置成功。打开 cmd 命令器，在命令器中输入 Java-version 若出现如图 1-10 所示状态则表示配置成功且 Java 程序安装成功并可使用。



图 1-10 JDK 安装测试页面 2

1.3.2 下载安装 Eclipse

安装并测试成功 JDK 软件后，下一步要安装 Eclipse 软件。在安装 Eclipse 软件前首先到相应的软件下载地址下载软件安装包。Eclipse 的下载地址：<http://www.eclipse.org/downloads/>。输入下载地址后，进入下载页面，如图 1-11 所示。

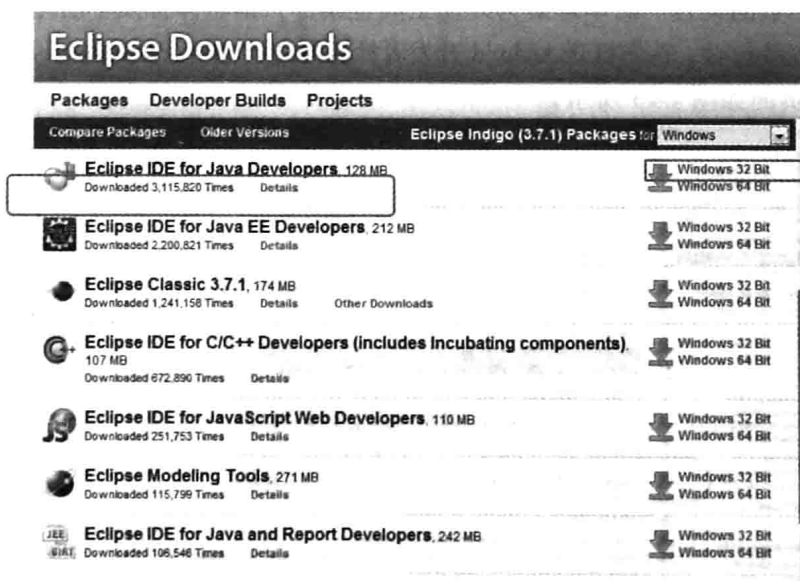


图 1-11 Eclipse 的下载页面 1

选择 Eclipse IDE for Java Developers 右侧与你的处理器位数相对应的项进行下载（此处以 32 位处理器为例）。选择完成后将进入如图 1-12 所示下载页面。

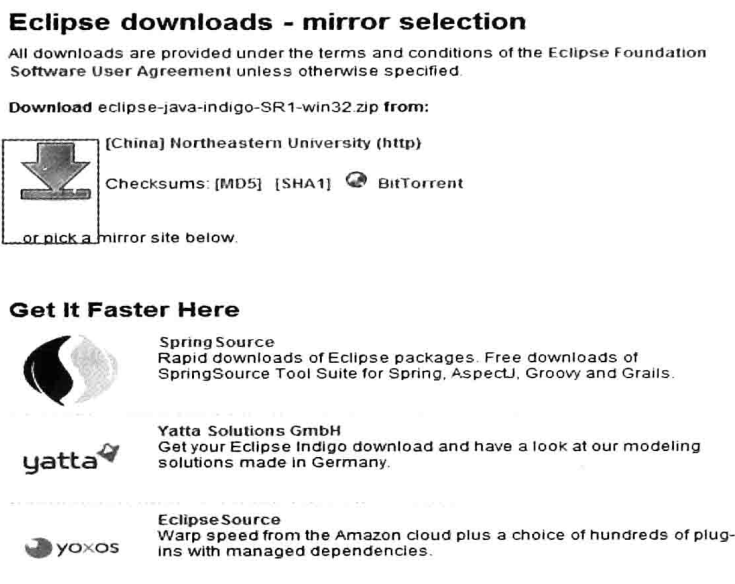


图 1-12 Eclipse 的下载页面 2

下载完成后解压即可使用，Eclipse 下载包在随书光盘中，读者可以直接从光盘获取，解压便可使用（在附录中下载其压缩包解压即可使用）。

1.3.3 SDK 的下载和安装

SDK 一般是指一些被软件工程师用于作为特定的软件包、软件框架、硬件平台、操作系统等建立应用软件的开发工具的集合。在 Android 中，它为开发者提供了库文件以及其他开发所用到的工具的集合。简单理解为开发工具包集合，是整体开发中所用到的工具包。

下载 SDK 安装包时，首先打开 Android 开发者社区网站：<http://developer.Android.com/>，并转入 SDK 下载页面如图 1-13 所示。

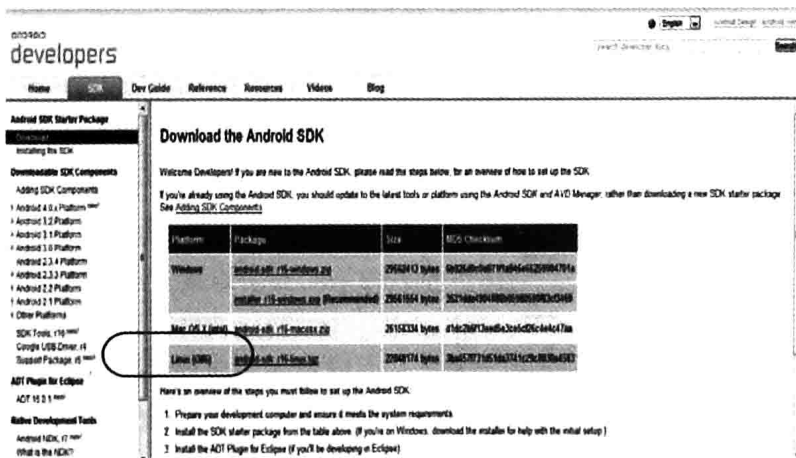


图 1-13 SDK 的下载

选择与所安装的系统一致的下载项，此处我们选择用于 Windows 平台的 Android-SDK r16-windows.zip，单击进入下载页面，下载并保存。下载完成后解压下载的文件，注意解压的路径在环境变量的设置中会涉及使用，此处假设将下载的压缩包解压在 F:\Android\ 目录下，配置环境变量方法同上，若使用第二种方法配置，此处只需更改 SDK 文件中的 tools 的绝对路径即可，具体方法如下：

计算机→属性→高级系统设置→高级→环境变量，在打开的环境变量对话框中，对系统变量进行设置，单击“新建”，在变量名处输入 SDK_HOME，变量值处输入前面提到的解压后 SDK 的路径，此处即为 F:\android-sdk-windows，设置完成后单击“确定”，如图 1-14 所示。

在此基础上，还需编辑路径的变量，即同样在打开的环境变量对话框中完成此操作，在环境变量对话框中对系统变量进行操作，选中变量 Path 对其进行编辑，将变量名改为% SDK_HOME%\tools；即可。如图 1-15 所示。



图 1-14 设置环境变量

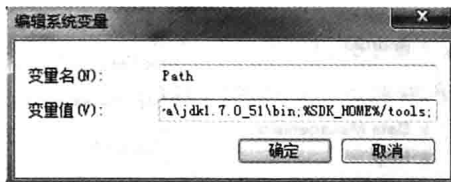


图 1-15 设置环境变量

变量值有问题也可以直接将 Path 的变量值设置为； F:\Android 光盘内容 \ DISK-ANDRI-OD \ Android-sdk-windows \ tools； C:\Program Files \ Java \ jdk1.7.0_51 \ bin；免去以上两步的繁琐。

利用 cmd 命令测试窗口进行 SDK 软件测试：打开 cmd 命令窗口，输入测试命令 android -h，若显示如图 1-16 所示信息则表示 SDK 软件安装成功。

完成 SDK 的安装后，配置 Android SDK 的大概方法如下。打开 Eclipse 开发环境，依次选择 window→Preference→Android，将会出现 SDK 的配置窗口，如图 1-17 所示。

在 Android 中，为开发者提供了库文件以及其他开发所用到的工具。简单理解为开发工具包集合，是整体开发中所用到的工具包，如果不用 Eclipse 作为开发工具，就不需要下载 ADT，只下载 SDK 即可开发。