

TURING

图灵原创



深入剖析 Android 开发

小应用里的大智慧

涵盖Android SDK最新版本

张泳 葛丽娜 编著

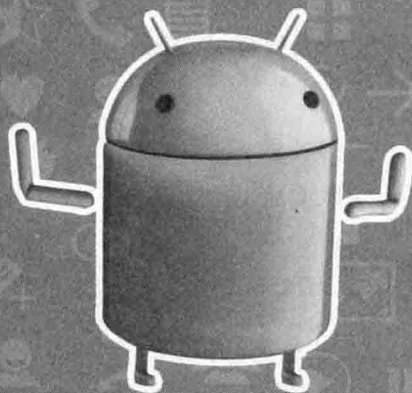
以谷歌音乐播放器为例
分析Android应用开发要点



人民邮电出版社
POSTS & TELECOM PRESS

TURING

图灵原创



深入剖析 Android开发

小应用里的大智慧

张泳 葛丽娜 编著

人民邮电出版社
北京

图书在版编目 (C I P) 数据

深入剖析Android开发：小应用里的大智慧 / 张泳,
葛丽娜编著. — 北京：人民邮电出版社, 2014. 8
(图灵原创)
ISBN 978-7-115-35856-1

I. ①深… II. ①张… ②葛… 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(2014)第115367号

内 容 提 要

本书以谷歌的音乐播放器为例,深入剖析了Android的重要组件和核心服务,书中首先介绍了Android开发环境的搭建,以及Android SDK中一些重要工具的用法,接着介绍了Android的组件及其配置,最后从源代码的角度剖析了Android的核心管理服务。

本书不仅适合从事或者正准备从事Android开发的工程师学习,也可以作为培训教材使用。

◆ 编 著 张 泳 葛丽娜

责任编辑 王军花

执行编辑 董苗苗

责任印制 焦志炜

◆ 人民邮电出版社出版发行 北京市丰台区成寿寺路11号

邮编 100164 电子邮件 315@ptpress.com.cn

网址 <http://www.ptpress.com.cn>

北京鑫正大印刷有限公司印刷

◆ 开本: 800×1000 1/16

印张: 24.75

字数: 585千字

印数: 1—3 000册

2014年8月第1版

2014年8月北京第1次印刷



定价: 69.00元

读者服务热线: (010)51095186转600 印装质量热线: (010)81055316

反盗版热线: (010)81055315

广告经营许可证: 京崇工商广字第 0021 号

前 言

Android 是一种以 Linux 为基础的开源操作系统。Android 1.1 版本于 2008 年 9 月发布。2009 年 4 月底，谷歌公司发布了 Android 1.5 版本，我正是从 1.5 版本开始接触并研究该操作系统的。彼时，在国内图书市场上，尚未有出版社推出该新生技术的图书，自学之路漫漫且艰难。幸而当年夏天，人民邮电出版社推出了《GoogleAndroid SDK 开发范例大全》一书，犹如甘霖滋润了我的求学若渴之心。此后，市场上出现了更多介绍 Android 的图书，我从中受益匪浅。

经过这几年的快速发展，Android 已经发展到目前的 4.4 版本。我在各大书店浏览时发现，真正深入剖析 Android 的图书寥寥无几，于是萌生了一个念头：何不将这几年的 Android 所学、所思以及实践经验编写成书，和广大开发者共同探讨研究 Android 的重要组件以及核心服务呢？冬去春来，工作之余，经过不懈努力，本书终于面世。希望本书对各位读者理解 Android 基本组件以及核心内容有所帮助。

本书内容

本书的各个篇章及主要内容如下。

第一篇主要介绍了 Android 开发环境的搭建，同时介绍了 Android SDK 中一些重要工具的使用方法。该篇下设如下两章。

- 第 1 章介绍了如何配置一个 Android 应用程序开发环境，以及如何使用 Android SDK 提供的强大工具来帮助我们进行开发和仿真调测。
- 第 2 章从简单的 HelloWorld 项目入手，介绍了多种测试方法及相关工具。

第二篇介绍了 Android 的组件及其配置。在这一篇中，我们以音乐播放器为例，翔实地阐述了重要组件的运用。该篇下设如下五章。

- 第 3 章以谷歌音乐播放器为例，介绍了 AndroidManifest.xml。
- 第 4 章介绍了 Activity，怎么创建 Activity，如何管理 Activity 的生命周期，以及如何实现它的声明周期等诸多内容。
- 第 5 章介绍了服务的详细情况及如何使用、声明应用程序的服务。另外，还介绍了两种服务的创建及用法，并简要介绍了服务的其他知识。
- 第 6 章介绍了最简单的布局 FrameLayout、线性布局 LinearLayout 和相对布局 RelativeLayout 这 3 种布局形式。

第三篇从源代码的角度来剖析 Android 的核心管理服务。该篇下设如下四章。

- ❑ 第 7 章讲述了启动 Android 系统的两个阶段：应用的初始化流程（init）与 system_service 进程及核心服务的创建流程。
- ❑ 第 8 章首先从启动、流程详解及使用方面介绍了备份管理服务，然后讨论了当程序被重新安装时所触发的备份和恢复操作。
- ❑ 第 9 章介绍了 Activity 管理服务的启动以及它的行为。
- ❑ 第 10 章介绍了包管理服务的启动、安装以及卸载应用程序时该服务的行为。

本书目标

本书以谷歌的音乐播放器为例，深入剖析了 Android 的重要组件和核心服务，从而帮助读者更深刻地理解 Android 的核心内容，以便融会贯通。在学习过程中，请读者按照书中的例子，自己动手进行验证。同时，结合书中对于代码以及图表的注解，理解示例，对所学内容咀嚼消化，扩宽思路，形成自己的认知，在今后的开发过程中运用自如。

本书适合从事或者正准备从事 Android 开发的工程师学习，可以帮助其深入透彻理解 Android 的一些核心知识，也可作为高级培训教材使用。

致谢

在编写过程中，本书综合并吸收了 Android 开发者的一些经验和心得，我在此对他们深表感谢。同时，感谢杨海玲老师、王军花老师在早期沟通和后续跟进过程中给予我的耐心、关心和帮助。最后，感谢为本书出版而辛勤付出的诸多工作者。

欢迎加入

图灵社区 ituring.com.cn

——最前沿的IT类电子书发售平台

电子出版的时代已经来临。在许多出版界同行还在犹豫彷徨的时候，图灵社区已经采取实际行动拥抱这个出版业巨变。作为国内第一家发售电子图书的IT类出版商，图灵社区目前为读者提供两种DRM-free的阅读体验：在线阅读和PDF。

相比纸质书，电子书具有许多明显的优势。它不仅发布快，更新容易，而且尽可能采用了彩色图片（即使有的书纸质版是黑白印刷的）。读者还可以方便地进行搜索、剪贴、复制和打印。

图灵社区进一步把传统出版流程与电子书出版业务紧密结合，目前已实现作译者网上交稿、编辑网上审稿、按章发布的电子出版模式。这种新的出版模式，我们称之为“敏捷出版”，它可以让读者以较快的速度了解到国外最新技术图书的内容，弥补以往翻译版技术书“出版即过时”的缺憾。同时，敏捷出版使得作、译、编、读的交流更为方便，可以提前消灭书稿中的错误，最大程度地保证图书出版的质量。

优惠提示：现在购买电子书，读者将获赠书款20%的社区银子，可用于兑换纸质样书。

——最方便的开放出版平台

图灵社区向读者开放在线写作功能，协助你实现自出版和开源出版梦想。利用“合集”功能，你就能联合二三好友共同创作一部技术参考书，以免费或收费的形式提供给读者。（收费形式须经过图灵社区立项评审。）这极大地降低了出版的门槛。只要有写作的意愿，图灵社区就能帮助你实现这个梦想。成熟的书稿，有机会入选出版计划，同时出版纸质书。

图灵社区引进出版的外文图书，都将在立项后马上在社区公布。如果你有意翻译哪本图书，欢迎你来社区申请。只要你通过试译的考验，即可签约成为图灵的译者。当然，要想成功地完成一本书的翻译工作，是需要有坚强的毅力的。

——最直接的读者交流平台

在图灵社区，你可以十分方便地写作文章、提交勘误、发表评论，以各种方式与作译者、编辑人员和其他读者进行交流互动。提交勘误还能够获赠社区银子。

你可以积极参与社区经常开展的访谈、乐译、评选等多种活动，赢取积分和银子，积累个人声望。

目 录

第一篇 Android 开发起航

第 1 章 环境搭建	2
1.1 搭建 Android 开发环境的需求	2
1.2 Android 开发环境配置	3
1.2.1 JDK 的下载、安装及配置	3
1.2.2 下载并安装 Eclipse	6
1.2.3 Android SDK 的安装和配置	6
1.2.4 ADT 的安装和配置	9
1.2.5 创建并运行模拟器	12
第 2 章 测试方法及工具	16
2.1 向世界问好——HelloWorld 项目	16
2.1.1 两种创建 HelloWorld 项目的方法	16
2.1.2 HelloWorld 项目中的默认配置解读	23
2.2 HelloWorld 项目运行及调试信息	26
2.2.1 运行 HelloWorld 应用程序	26
2.2.2 HelloWorld 运行过程的调试信息	29
2.3 DDMS 工具介绍	30
2.3.1 DDMS 工具及其打开方式	30
2.3.2 使用 DDMS 工具调测 HelloWorld 项目	31
2.3.3 详解 DDMS 工具界面	36
2.3.4 DDMS 工具菜单中的重要工具	41
2.4 Android 的主要工具介绍	62
2.4.1 ADB 工具	63
2.4.2 android 工具	64
2.4.3 sdcard 相关命令	65
2.4.4 模拟器的操作	65

2.4.5 LogCat 工具	67
2.4.6 数据库工具	68
2.5 Android 其他小工具简介	68
2.5.1 截屏工具	69
2.5.2 Monkey 工具	69
2.6 下载谷歌播放器源代码	72

第二篇 Android 组件及其配置

第 3 章 应用的五脏六腑——AndroidManifest.xml	78
3.1 Android Manifest.xml 文件	78
3.2 一切从<manifest>节点开始	82
3.2.1 xmlns:android 属性——定义命名空间	82
3.2.2 package 属性——应用程序的身份证	84
3.2.3 android:sharedUserId 属性——共享数据	85
3.2.4 android:versionCode 属性——内部版本号	86
3.2.5 android:versionName 属性——显示给用户的版本号	86
3.2.6 android:installLocation 属性——安装位置	87
3.2.7 HelloWorld 示例——再向世界打个招呼	89
3.2.8 动动手，验证知识	90
3.3 应用程序权限的声明	93
3.3.1 <uses-permission>——应用程序的权限申请	93

3.3.2	<permission>节点——自定义应用程序的访问权限.....	100	3.6.14	android:persistent	165
3.3.3	<uses-sdk>节点——SDK 版本限定	103	3.6.15	android:process	166
3.3.4	<instrumentation>节点——应用的监控器	105	3.6.16	android:taskAffinity	167
3.3.5	动动手, 验证知识	107	3.6.17	android:theme	168
3.3.6	<instrumentation>节点的另一种使用方法	114	3.6.18	android:uiOptions	171
3.4	应用程序的根节点——<application>	117	3.6.19	android:vmSafeMode	174
3.4.1	<application>节点配置	117	3.6.20	android:largeHeap	174
3.4.2	音乐播放器的<application>节点	118	第 4 章	让程序活动起来——Activity	176
3.4.3	如何实现 Application 类	119	4.1	什么是 Activity	176
3.4.4	Application 提供的函数及其用法	123	4.1.1	简介	176
3.5	backupAgent 的用法	134	4.1.2	解读音乐播放器中的 Activity	177
3.5.1	backupAgent 简介	134	4.2	定义 Activity	179
3.5.2	如何使用 backupAgent 来实现备份	134	4.2.1	定义 Activity 的回调方法	179
3.5.3	从备份中实现恢复	141	4.2.2	在 AndroidManifest.xml 中声明 Activity	181
3.5.4	如何使用 bmgr 工具	144	4.3	管理 Activity 的生命周期	182
3.6	<application>的属性详解	150	4.3.1	Activity 的 3 种状态	182
3.6.1	android:allowBackup	151	4.3.2	实现 Activity 的生命周期回调	182
3.6.2	allowTaskReparenting	152	4.3.3	回调方法在音乐播放器中的应用	186
3.6.3	android:killAfterRestore	152	4.4	保存和协调 Activity	189
3.6.4	android:restoreAnyVersion	153	4.4.1	保存 Activity 状态	189
3.6.5	android:debuggable	156	4.4.2	协调 Activity	191
3.6.6	android:description	158	4.5	解读关于生命周期的一个实例	192
3.6.7	android:enabled	159	4.6	<activity>节点的属性	195
3.6.8	android:hasCode	160	4.6.1	android:allowTaskReparenting	195
3.6.9	android:hardwareAccelerated	161	4.6.2	android:alwaysRetainTaskState	195
3.6.10	android:label / android:icon	162	4.6.3	android:clearTaskOnLaunch	196
3.6.11	android:logo	164	4.6.4	android:configChanges	196
3.6.12	android:manageSpace-Activity	165	4.6.5	android:enabled	197
3.6.13	android:permission	165	4.6.6	android:excludeFromRecents	197
			4.6.7	android:exported	197
			4.6.8	android:finishOnTaskLaunch	197
			4.6.9	android:hardwareAccelerated	198

4.6.10	android:icon	198	6.1.3	FrameLayout 内子视图的特色 布局参数	220
4.6.11	android:label	198	6.2	线性布局——LinearLayout	222
4.6.12	android:launchMode	198	6.2.1	LinearLayout 简介	222
4.6.13	android:multiprocess	199	6.2.2	LinearLayout 的特有属性	225
4.6.14	android:name	199	6.2.3	LinearLayout 特有的布局 参数	231
4.6.15	android:noHistory	199	6.3	相对布局——RelativeLayout	231
4.6.16	android:permission	200	6.3.1	RelativeLayout 简介	231
4.6.17	android:process	200	6.3.2	RelativeLayout 的特色属性 及其参数	233
4.6.18	android:screenOrienta- tion	200	第三篇 核心服务解析篇		
4.6.19	android:stateNotNeeded	200	第 7 章 Android 系统的启动		
4.6.20	android:taskAffinity	201	7.1	初始化流程	244
4.6.21	android:theme	201	7.1.1	应用的初始化流程	244
4.6.22	android:windowSoft- InputMode	201	7.1.2	init.rc 的用法	246
第 5 章 我会默默地为你服务—— service			7.1.3	用 init 解析整个 init.rc 文件	251
5.1	服务	202	7.2	创建 system_service 进程	254
5.1.1	何为服务	202	7.2.1	创建流程	254
5.1.2	服务可采用的方法	202	7.2.2	system_service 简介	256
5.1.3	<service>节点的属性	203	第 8 章 备份管理服务		
5.2	创建并使用服务	204	8.1	备份管理服务的启动方式和流程	258
5.2.1	创建 Service 子类的重要回调 方法	204	8.1.1	备份管理服务的启动	258
5.2.2	在 manifest 文件中声明服务	205	8.1.2	详解备份管理服务的流程	258
5.3	创建一个启动的服务	206	8.2	使用备份管理服务	263
5.3.1	继承 IntentService 类	206	8.2.1	bmgr 工具简介	263
5.3.2	继承 Service 类	207	8.2.2	使用 bmgr 工具实现备份与 恢复	264
5.3.3	启动服务	208	8.2.3	用编程的方式实现备份与 恢复	272
5.3.4	停止服务	208	8.3	应用程序在被重新安装过程中的 备份和还原	274
5.3.5	TuringService 实例	209	第 9 章 Activity 管理服务		
5.4	创建一个被绑定的服务	212	9.1	ActivityManagerService 简介	276
5.4.1	基本介绍	212	9.2	ActivityManagerService 的使用	276
5.4.2	TuringBoundService 实例	213	9.2.1	孵化进程	276
第 6 章 我可以更漂亮——布局					
6.1	最简单的布局类——FrameLayout	217			
6.1.1	FrameLayout 简介	217			
6.1.2	FrameLayout 特有的属性	218			

9.2.2	ActivityManagerService 启动的3个阶段	277
9.2.3	ActivityManagerService 的工作原理	289
9.2.4	ActivityManagerService 依赖的两个类	294
9.3	Activity的启动流程	295
9.3.1	启动Activity的方式	295
9.3.2	Activity启动的4个阶段	297
9.4	结束Activity	306
9.4.1	结束Activity的3种主要方法	306
9.4.2	结束Activity的4个阶段	308
9.5	广播接收器	315
9.5.1	注册广播接收器	315
9.5.2	ActivityManagerService 的行为	316
9.6	服务	318
9.6.1	服务的数据结构	318
9.6.2	启动服务	320
9.6.3	停止服务	327
9.6.4	以绑定的方式启动/停止服务	331
9.7	发布ContentProvider	339
9.7.1	启动ContentProvider发布 工作时的操作	339
9.7.2	解读发布流程中Activity- ManagerService的行为	342
9.8	ActivityManagerService如何应付 异常	345

第10章 包管理服务 349

10.1	PackageManagerService 概述	349
10.2	PackageManagerService 的组成和 应用	349
10.2.1	PackageManagerService 的重要组成部分	350
10.2.2	解读 PackageManager- Service 如何关注目录	351
10.2.3	PackageManagerService 定义的 PackageParser 类	354
10.3	启动 PackageManagerService	355
10.3.1	PackageManagerService 的启动流程	355
10.3.2	PackageManagerService 构造函数的流程	356
10.3.3	scanDirLI() 方法	361
10.4	解析 AndroidManifest.xml 文件	366
10.4.1	解析流程	366
10.4.2	解析音乐播放器的 AndroidManifest.xml 文件	368
10.5	安装应用程序	377
10.5.1	用 ADB 的 install 命令 安装应用程序	377
10.5.2	解析 installPackageWith- Verification() 的行为	379
10.6	卸载应用程序	383

Part 1

第一篇

Android 开发起航

想要开发 Android 应用程序，开发者首先需要完成的任务是搭建 Android 应用程序的开发环境，并且了解 Android SDK 提供给我们的一些比较重要的工具。

本章将详细讲解如何配置一个Android应用程序的开发环境,以及如何使用Android SDK提供的强大工具来帮助我们进行开发和仿真调测。

1.1 搭建 Android 开发环境的需求

在开始配置环境之前,我们首先要了解Android开发环境的需求,这里的需求包括操作系统需求、硬件需求和软件需求等。

- **操作系统需求:** Android支持的操作系统及其需求说明如表1-1所示。本书以Windows 系统(32位操作系统)为基础编写,关于Linux系统的相关内容稍后介绍。

表1-1 Android开发环境的操作系统需求

操作系统	需求说明
Windows	Windows XP (32位操作系统)、Vista (32或64位操作系统)、Windows 7 (32或64位操作系统)
Mac OS	Mac OS X 10.5.8 或更高版本 (仅支持x86)
Linux	至少需要支持GNU C库 (glibc) 2.7或更高版本,使用Ubuntu Linux,至少需要支持8.04或更高版本,64位系统必须有运行32位应用程序的能力

- **软件需求:** Eclipse IDE、JDK和ADT。对应的版本和下载地址如表1-2所示。

表1-2 Android开发环境的软件需求

软 件	版 本	下载地址
Eclipse IDE	Eclipse 3.7.2 (Indigo) 或更高版本	http://www.eclipse.org/downloads
JDK	JDK 6或更高版本	http://www.oracle.com/technetwork/java/javase/downloads/index.html
ADT	ADT 12.0.0 (以当时能下到的最新版本为准)	https://dl-ssl.google.com/android/eclipse/ (后面将描述该地址的使用方法)

- **硬件需求:** Android SDK要求开发者有足够的磁盘存储空间选择安装所需的SDK组件。表1-3描述了各个组件所需要的磁盘空间及其说明。

表1-3 Android SDK组件及其说明

组 件	组件大小（近似）	说 明
Tools	40 MB	必选。开发以及调试Android应用程序的必要工具或者一些工具的快捷方式（这里的快捷方式主要以Windows的对比处理文件或Linux的shell文件等形式出现），其中包括DDMS（Dalvik Debug Monitor Server）、Emulator（Android模拟器）、monkeyrunner、ADB等工具以及这些工具依赖的运行库
Platform-tools	4.5 MB	必选。Android的平台工具、ADB、fastboot烧录工具及其相关库文件
Platform（每一个）	150 MB	至少选择一个。Android开发平台，其中包括Android开发的基本类库（android.jar）、相关资源以及创建应用程序使用的相关模板（Android Manifest.xml文件模板、Activity代码模板以及一些默认资源）
Add-on（每一个）	100 MB	可选。第三方工具包，这里一般包含谷歌服务相关库（比例地图服务等）、文档、例子以及系统镜像
Build-tools（每一个）	54 MB	建议开发安装。编译以及打包应用程序所需要的工具及相关类库，比如aapt、aidl、dx等
Samples	10 MB	可选。这里包含每一个Android版本的API使用例子。虽然这是可选下载部分，但我还是建议开发者下载
docs	586 MB	可选。离线文档部分，这里包含了API的说明等重要信息。虽然这是可选下载部分，但我还是建议开发者下载
system-images	根据版本而定	必选。Android 3.0以后的SDK版本在这个目录下存放不同版本的配套镜像文件。启动虚拟机的时候，需要导入这些镜像文件
AVD Manager	351 KB	虚拟机管理工具，包括管理、刷新、创建、删除虚拟机
SDK Manager	351 KB	SDK管理工具，使用它可升级或者安装Android DSK

1.2 Android 开发环境配置

清楚了Android开发环境的基本需求，这一节就来介绍如何配置Android开发环境，主要包括JDK的下载、安装以及配置，Eclipse的下载和安装，Android SDK的安装和配置，ADT的安装和配置，创建并运行模拟器这5个方面的内容。

1.2.1 JDK的下载、安装及配置

JDK是整个Java的核心，包括Java运行时环境（Java Runtime Environment，JRE）和Java工具等，它们是Android开发环境的基础部分。接下来，我们就来详细演示配置Java基础平台的具体操作步骤。

(1) 登录<http://www.oracle.com/technetwork/java/javase/downloads/index.html>，下载最新版本的JDK。

(2) 安装JDK。下载完成以后，可以得到一个可执行的安装包，直接双击即可开始安装过程，如图1-1所示。

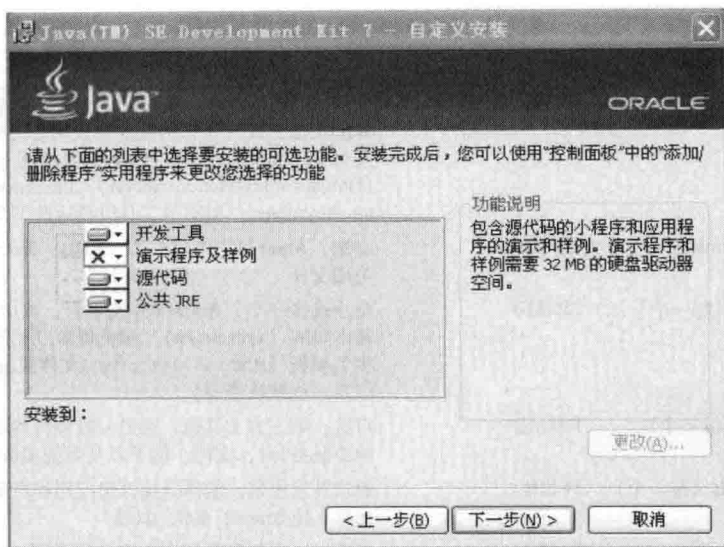


图1-1 JDK安装启动画面

这里建议不修改图1-1中展示的默认配置，一直单击“下一步”按钮即可。

安装包中包含了JDK和JRE两个部分。当我们安装完成后，可以在安装路径下看到如图1-2所示的目录结构。



图1-2 Java环境的目录结构

注意 编写此书时，JDK的版本是JDK 7。如果读者用的是更高的版本，其安装步骤与此相同。

(3) 当完成整个安装过程之后，Java的相关环境就已经基本配置好了，现在来验证一下。具体操作方法为单击“开始”→“运行”菜单，进入命令行模式，键入命令`java -version`，如果得到如图1-3所示的结果，则证明我们的安装和配置是成功的。否则，就需要回到前面的步骤进行检查，再重新验证。

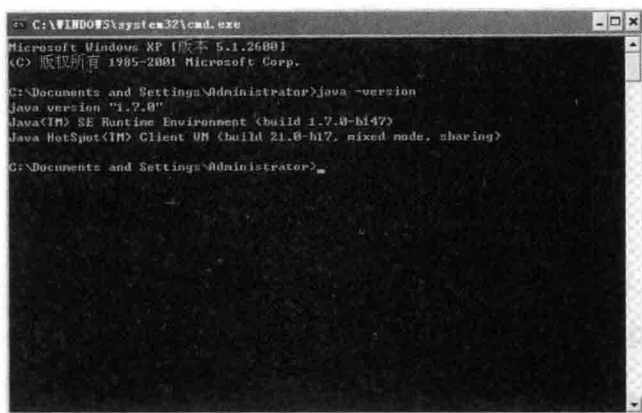


图1-3 java-version命令的测试结果

(4) 完成了上述的安装步骤, Java环境已经基本可用, 这里还需要将关键路径配置到环境变量中, 具体如下所示。

配置JAVA_HOME

右击“我的电脑”, 单击“属性”, 单击“高级”选项卡, 单击“环境变量”, 单击“系统变量”下方的“新建”按钮, 并在弹出的“新建系统变量”对话框中将JDK的路径配置其中, 完成后单击“确定”按钮以确认配置, 如图1-4所示。

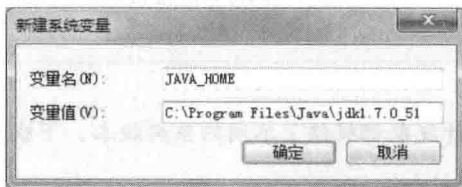


图1-4 配置JAVA_HOME系统环境变量

配置CLASSPATH

右击“我的电脑”, 单击“属性”, 单击高级系统设置选项, 在打开的“高级”选项卡中单击“环境变量”按钮, 接着单击“环境变量”对话框下方的“新建”按钮, 然后在弹出的“新建系统变量”对话框中将JDK中LIB和相关工具的路径配置其中, 完成后单击“确定”按钮, 如图1-5所示。

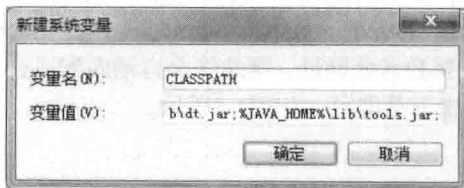


图1-5 配置CLASSPATH系统环境变量

然后将JDK和JRE的bin路径加入系统路径中，如图1-6所示。

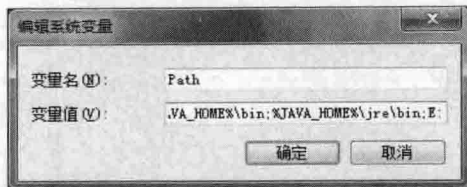


图1-6 配置Path系统环境变量

至此完成Java环境的配置。

1.2.2 下载并安装Eclipse

Eclipse的下载地址为<http://www.eclipse.org/downloads/>，它为我们提供了Eclipse包，只需要选择符合Android开发需求的下载即可。这里选择了“Eclipse Modeling Tools”包进行开发，如图1-7所示。下载后，无需安装它，直接解压运行即可（前提是Java环境正确配置）。这里我们将“Eclipse Modeling Tools”包解压到自定义的目录即可。



图1-7 Eclipse开发包

注意

- ❑ Eclipse提供的每一个开发包都提供了不同的系统版本，下载的时候开发者需要根据自身的实际情况进行选择。
- ❑ 在Eclipse的解压路径中，应尽量避免使用中文字符和空格等特殊字符。

1.2.3 Android SDK的安装和配置

至此，我们就完成了Android开发环境的基础配置。接下来，我们将讲解如何安装和配置Android开发工具包（后面简称Android SDK），具体步骤如下所示。

(1) 登录Android开发者网站(<http://developer.android.com/sdk/index.html>)，下载最新的Android SDK版本。值得一提的是，只要登录此网址，网站将会自动匹配适合开发者正在使用的操作系统的Android SDK版本，直接点击下载即可，如图1-8所示。



图1-8 下载SDK（匹配Windows系统）

(2) 解压下载好的SDK安装包，此时我们仅仅得到了如图1-9所示的基本内容的目录结构。要进行开发，还需要下载其他必要的安装包。

名称	大小	类型	修改日期
eclipse		文件夹	2013-3-9 14:28
sdk		文件夹	2013-3-9 14:29
SDK Manager	350 KB	应用程序	2013-2-19 14:54

图1-9 Android SDK的初始化目录结构

如图1-9所示，谷歌为Android开发者提供了一个已经部署好ADT的Eclipse。大家可以选择使用此Eclipse进行开发，也可以根据以下步骤继续部署一个属于自己的Android开发环境。

(3) 运行目录下的“SDK Manager.exe”文件进行安装。这里，我们选择下载并安装所有的平台版本，还要下载相关例子文档，如图1-10所示，同时还需要更新SDK工具（SDK Tools）和平台工具（Platforms Tools）。

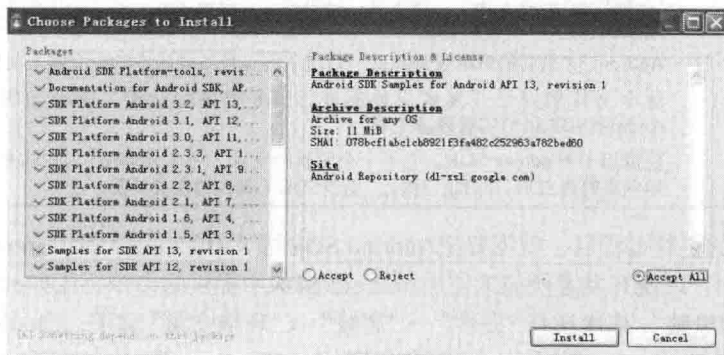


图1-10 Android SDK的安装以及更新