

信息科学与技术丛书 移动与嵌入式开发系列

汪永松 编著

Android 平台开发之旅

- Android应用程序组件及机制剖析
- 网络通信(SSL、WebKit)与无线通信(短信、蓝牙和Wi-Fi)
- 数据库应用(SQLite、JDBC API和Db4o)详解
- 电话信息、个人信息和系统信息管理实例开发
- 地图应用扩展



机械工业出版社
CHINA MACHINE PRESS

信息科学与技术丛书·移动与嵌入式开发系列

Android 平台开发之旅

汪永松 编著



机械工业出版社

本书涵盖了 Android 平台 1.5 到 2.2 版本的主要功能特性，立足实际的开发案例，介绍了 Android 手机平台开发的基础概念、实用技术和应用模式。主要内容包括：平台基础、开发环境搭建、程序框架、高级界面和底层界面设计、文件系统管理、网络通信、无线通信、多媒体编程、个人信息管理、电话系统、数据库应用、XML 应用和地图应用。开发实例多达 120 例。

本书主要面向具有一定移动平台开发经验的开发人员，以及有兴趣进行 Android 平台开发的程序员。

书中代码可从 <http://www.cmpbook.com/> 下载。

图书在版编目 (CIP) 数据

Android 平台开发之旅 / 汪永松编著. —北京: 机械工业出版社, 2010.7

(信息科学与技术丛书·移动与嵌入式开发系列)

ISBN 978-7-111-31294-9

I. ①A… II. ①汪… III. ①移动通信-携带电话机-应用程序-程序设计
IV. ①TN929.53

中国版本图书馆 CIP 数据核字 (2010) 第 134034 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: 车 忱

责任编辑: 车 忱

责任印制: 乔 宇

三河市国英印务有限公司印刷

2010 年 8 月·第 1 版第 1 次印刷

184mm×260mm·33.25 印张·822 千字

0001—3000 册

标准书号: ISBN 978-7-111-31294-9

定价: 60.00 元

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

社服务中心: (010) 88361066

门户网: <http://www.cmpbook.com>

销售一部: (010) 68326294

教材网: <http://www.cmpedu.com>

销售二部: (010) 88379649

读者服务部: (010) 68993821

封面无防伪标均为盗版

出版说明

随着信息科学与技术的迅速发展，人类每时每刻都会面对层出不穷的新技术和新概念。毫无疑问，在节奏越来越快的工作和生活中，人们需要通过阅读和学习大量信息丰富、具备实践指导意义的图书来获取新知识和新技能，从而不断提高自身素质，紧跟信息化时代发展的步伐。

众所周知，在计算机硬件方面，高性价比的解决方案和新型技术的应用一直备受青睐；在软件技术方面，随着计算机软件的规模和复杂性与日俱增，软件技术不断地受到挑战，人们一直在为寻求更先进的软件技术而奋斗不止。目前，计算机在社会生活中日益普及，随着 Internet 延伸到人类世界的方方面面，掌握计算机网络技术和理论已成为大众的文化需求。由于信息科学与技术 在电工、电子、通信、工业控制、智能建筑、工业产品设计与制造等专业领域中已经得到充分、广泛的应用。所以这些专业领域中的研究人员和工程技术人员越来越迫切需要汲取自身领域信息化所带来的新理念和新方法。

针对人们了解和掌握新知识、新技能的热切期待，以及由此促成的人们对语言简洁、内容充实、融合实践经验的图书迫切需要的现状，机械工业出版社适时推出了“信息科学与技术丛书”。这套丛书涉及计算机软件、硬件、网络和工程应用等内容，注重理论与实践的结合，内容实用、层次分明、语言流畅。是信息科学与技术领域专业人员不可或缺的参考书。

目前，信息科学与技术的发展可谓一日千里，机械工业出版社欢迎从事信息技术方面工作的科研人员、工程技术人员积极参与我们的工作，为推进我国的信息化建设作出贡献。

机械工业出版社

前 言

作者在刚接触 Android 的时候，感觉就像推开了一扇窗，与 Android 有关的信息、技术、概念纷至沓来，让人目不暇接。不可否认，无论是 Android 手机产品的定位还是其开源项目的技术背景，都使它成为业界瞩目的焦点。Android 手机的横空出世，打乱了移动平台的格局；而 Android 项目的开源举措，更是人心所向。这些因素毫无疑问地让 Android 成为移动平台开发者最看好的黑马。

但随着作者对 Android 平台认识的逐步深入，令作者震撼的不再是其表面的新奇，而是那些支撑平台蓬勃发展、来自各个领域的专业应用。在图形方面，有工业级的 OpenGL ES 库；在输入法方面，有应用广泛的 FreeType 引擎；在网络方面，除了纳入 Bouncy Castle 提供的 SSL 算法和 Apache 提供的 HTTP 开发库，还把 WebKit 项目内核作为系统浏览器引擎；在数据管理方面融入了 SQLite 数据库；在 XML 应用方面引入了 XML Pull API；在集成应用方面植入了地图应用。以上这些项目或标准，都在各个领域中得到广泛应用，而 Android 平台能把这些“习性各异”的先进技术融汇到一起，并进行协同作业，这才真正是该平台博大精深的地方。

本书的特色

作者认为本书有三个鲜明的特色。

第一点：内容全面，讲解细透。本书中的内容涵盖了 Android 1.5 到 2.2 版本主要的功能特性，除了结合其他技术透彻地讲解平台中功能的渊源和关联，还对部分变迁之后的功能进行了对比介绍，切实让读者能够结合自己的知识来理解平台中的功能，无论平台如何升级都能感受到“万变不离其宗”的技术本质。

第二点：案例丰富，易于动手。本书分为 17 章，开发实例多达 120 例。这些开发实例都是经过作者亲自进行审定和调试的，其内容不仅与章节的内容紧密相扣，而且还能方便地用于实际演练，从而激发读者的学习热情和巩固对相关知识的理解。

第三点：结构合理，深浅适度。本书内容的编排遵循“由表及里，由内而外”的形式，从功能使用到应用机制，从高级界面到底层界面，从内部存储到外部通信。在对专题的介绍中，作者结合自己的理解，采用“步步为营”的方式引导读者从了解功能到应用联想，让读者逐步形成自己的认识，再借助详细的开发实例来加深理解。

本书章节内容

本书前三章简要介绍了 Android 平台、开发环境以及 Android 应用程序组件。

第 4 章重点介绍了架构中比较常见、重要的界面元素，并通过众多实例让读者能够迅速地在 Android 平台搭建如心所愿的界面效果。

第 5 章介绍了一些用于底层用户界面控制的组件及其使用方式。

第 6 章对 Android 平台中的文件访问类型进行了详细的说明，从系统和应用程序的角度

介绍了对文件系统进行访问的过程。

第 7 章对 Android 平台支持的多种网络通信机制进行了详细的介绍。还介绍了网页浏览器的开发技术和实际的开发案例。

第 8 章对 Android 平台支持的短消息通信、蓝牙通信以及 Wi-Fi 网络连接管理等无线通信方式进行了详细的介绍。

第 9 章对 Android 平台提供的多媒体应用方式进行了实例说明。

第 10 章介绍了 Android 支持的个人信息管理内容，通过实际的开发案例，讲解了如何获取联系人信息、电话号码、公司信息等与个人有关的内容。并结合调整前后的 Android 平台的不同版本（1.5 和 2.1），以对比的方式介绍不同版本对个人信息管理的支持方式的改变。

第 11 章介绍了 Android 平台提供的电话信息系统管理功能。还介绍了如何获取呼叫日志信息。

第 12 章对 Android 平台支持的数据库类型进行了详细介绍，主要内容包括：SQLite 数据库、JDBC API 和 Db4o 数据库。

第 13 章对 Android 平台支持的 XML 应用方式进行了全面说明，主要内容包括：SAX 解析方式、DOM 解析方式、XML Pull API 以及资源解析过程分析。

第 14 章对 Android 平台提供的地图 API 的功能进行了详细的阐述，并通过开发实例详细介绍了如何控制地图以及添加地图叠加图等常用功能，同时还对地图视图的使用模式和缩放控制进行了小结。

第 15 章对 Android 平台提供的系统信息管理接口进行了全面介绍。

第 16 章对 Android 平台支持的资源类型及其定义、资源的使用模式、系统资源定义进行了全面介绍。还对 Android SDK 附带工具的常用方式进行了详细说明。

第 17 章对 Android 平台中常用视图组件的属性以及应用程序的使用许可进行了全面介绍。

附录对随书源代码的使用进行了说明。

本书中的一些约定

注意：提醒读者应该重视的内容。

提示：是对读者有所帮助的一些技巧。

书中提到的源代码可在 <http://www.cmpbook.com/> 下载。全部代码按章划分父目录，各子目录以工程为单位存放。源代码的使用可以参考附录（随书源代码说明）。

代码类型

本书中的代码分为 Java 代码和 XML 代码。

编码风格

代码 Q-1 是本书中的示例代码，本书中所有代码风格与之相同。



代码 Q-1 示例代码

```
1 public class WidgetsDemoAct extends Activity {  
2     @Override  
3     public void onCreate(Bundle savedInstanceState) {  
4         super.onCreate(savedInstanceState);  
5         setContentView(R.layout.main);  
6     }  
7 };
```

囿于篇幅，代码中部分非核心语句用省略号代替。读者可在 <http://www.cmpbook.com> 下载完整代码。

养成良好的编码风格对于任何程序员来说都是十分重要的，可以说编码风格是判断一个程序员是不是“老手”的标杆。对于刚起步的程序员，在编码过程中首先要树立编码风格的意识，再通过不断地实践和摸索，才能逐步形成自己的编码风格。

最后，希望朋友们开发成功！

目 录

出版说明

前言

第 1 章 初识 Android 平台 1

1.1 Android 平台简介 1

1.1.1 Android 发展历史 1

1.1.2 平台内涵 2

1.2 Android 平台架构 4

1.2.1 架构图 4

1.2.2 架构内容 5

1.2.3 Android 应用程序内容 6

1.3 用户界面 8

1.3.1 视图层次结构 8

1.3.2 布局——设计图 8

1.3.3 视图——整体家居 9

1.3.4 显示部件——装饰品 9

1.3.5 用户界面事件 9

1.3.6 界面风格和主题 10

1.3.7 数据绑定 10

1.4 程序资源和资产 10

1.4.1 程序资源和资产概述 11

1.4.2 资源类型及内容 11

1.5 数据存储 11

1.5.1 首选项 12

1.5.2 文件 12

1.5.3 数据库 12

1.6 平台安全和许可 12

1.6.1 Android 平台安全结构体系 12

1.6.2 应用程序签名 13

1.6.3 用户 ID 和文件存取 13

1.6.4 许可 13

第 2 章 踏上 Android 平台开发之旅 15

2.1 搭建系统环境 15

2.1.1 安装配置 J2SE 开发环境 15

2.2 Android 平台 SDK 17

2.2.1 安装配置 Android 平台 SDK 17

2.2.2 Android SDK 内容介绍 17

2.2.3 Android SDK 附带工具介绍 19

2.3 集成开发环境——Eclipse 21

2.4 应用程序调试工具插件——

ADT 22

2.4.1 获取 ADT 22

2.4.2 安装配置 ADT 22

2.5 验证开发环境 23

2.5.1 创建虚拟设备 23

2.5.2 建立 FirstActivity 工程 24

2.5.3 运行和调试 31

2.6 应用程序的开发过程 37

2.6.1 开发流程回顾 37

2.6.2 新手上路遇到的常见问题 38

第 3 章 Android 应用程序组件 41

3.1 应用程序组件 41

3.2 Android 应用程序组件 41

3.2.1 Activity（活动）——形象大使 42

3.2.2 Service（服务）——老黄牛 47

3.2.3 BroadcastReceiver（广播接收器）
——倾听者 50

3.2.4 ContentProvider
（内容提供者） 53

3.2.5 Android 应用程序组件小结 58

3.3 组件应用机制 58

3.3.1 组件与组件间的交互机制 58

3.3.2 未决意向对象 65

3.3.3 组件与线程间的交互机制 66

3.3.4 组件与服务间的交互机制 68

3.4 Android 平台应用程序组件

小结 73



第 4 章 高级用户界面设计	75	4.11.2 扩展列表 Activity (ExpandableListActivity)	149
4.1 Android 平台 UI 组件架构探讨	75	4.11.3 标签页 Activity (Tab Activity)	149
4.1.1 Android 平台 UI 组件结构层次	75	4.11.4 地图 Activity (MapActivity)	150
4.1.2 Android UI 组件结构层次质疑	76	4.12 用户界面开发问题	150
4.1.3 参考 J2SE 平台的组件结构 层次	76	4.12.1 ANR 异常	150
4.1.4 容器与组件的关系	76	4.12.2 界面组件与线程的交互	150
4.1.5 布局的角色	76	第 5 章 底层用户界面设计	153
4.2 UI 组件的定义	77	5.1 Android 底层用户界面	153
4.3 UI 组件的引用	78	5.2 底层视图绘制	154
4.4 组件属性和 ID	79	5.2.1 表面视图类 (Surface View)	154
4.5 布局组件 (Layouts)	80	5.2.2 底层视图的绘制接口	159
4.5.1 线性布局 (LinearLayout)	81	5.3 OpenGL 视图绘制	160
4.5.2 相对布局 (RelativeLayout)	84	5.3.1 OpenGL ES 概述	160
4.5.3 绝对布局 (AbsoluteLayout)	86	5.3.2 Android 平台对 OpenGL ES 的 支持	160
4.5.4 框布局 (FrameLayout)	87	5.3.3 OpenGL 表面视图的使用模式	161
4.5.5 表格布局 (TableLayout)	89	5.3.4 Android 平台中 OpenGL 使用 说明	163
4.5.6 布局的选择	91	5.4 视频视图 (VideoView)	163
4.6 视图组件 (Views)	91	第 6 章 文件系统管理	164
4.6.1 视图的使用模式	92	6.1 Android 平台中的文件	164
4.6.2 常用视图	94	6.2 原文件资源	164
4.6.3 定制视图	120	6.2.1 准备原文件资源	165
4.7 小部件 (Widgets)	122	6.2.2 使用原文件资源	165
4.7.1 小部件的使用模式	122	6.3 文件系统	166
4.7.2 文本部件	123	6.3.1 Android 平台文件系统介绍	167
4.7.3 按钮部件	126	6.3.2 文件系统操作	168
4.7.4 图片显示组件	131	6.3.3 文件浏览器	172
4.7.5 滑动条	133	6.4 应用程序文件	181
4.7.6 选取组件	134	6.4.1 程序私有文件	181
4.7.7 高级小部件	135	6.4.2 程序首选项文件	184
4.8 菜单	138	6.5 文件系统监视	185
4.8.1 选项菜单	138	第 7 章 网络通信	191
4.8.2 上下文菜单	140	7.1 Android 平台网络通信	191
4.9 对话框	141	7.2 Android 平台对网络通信的 支持	191
4.9.1 对话框的使用方式	142		
4.9.2 对话框的定义	143		
4.10 消息提示条 (Toast)	148		
4.11 定制 Activity	149		
4.11.1 列表 Activity (List Activity)	149		

7.3 流式套接字通信	192	8.3.1 Android 平台对蓝牙的支持	265
7.3.1 界面布局定义	193	8.3.2 蓝牙通信模式	266
7.3.2 Activity 定义框架	193	8.3.3 蓝牙通信实例	266
7.3.3 套接字服务端	195	8.4 Wi-Fi 网络连接管理	276
7.3.4 套接字客户端	201	8.4.1 Wi-Fi 介绍	276
7.3.5 流式套接字通信说明	206	8.4.2 Android 平台对 Wi-Fi 的支持	277
7.4 数据报 (套接字) 通信	206	8.4.3 Wi-Fi 连接管理	277
7.4.1 数据报通信程序界面	206	第 9 章 多媒体应用	281
7.4.2 Activity 定义框架	206	9.1 Android 平台对多媒体的支持	281
7.4.3 数据报服务端	208	9.2 音频播放应用	282
7.4.4 数据报工具类	210	9.2.1 音乐播放器	282
7.4.5 数据报通信配置	211	9.2.2 播放 Jet 文件	284
7.4.6 数据报客户端	211	9.2.3 录音	285
7.4.7 数据报套接字通信说明	215	9.3 视频播放应用	286
7.5 HTTP 通信	215	9.4 摄像头视频采集	292
7.5.1 套接字方式	215	9.5 照相机	293
7.5.2 URL 连接方式	218	9.5.1 照片服务接口	293
7.5.3 Apache HTTP API	220	9.5.2 照片预览	293
7.6 HTTPS	222	9.5.3 照片拍摄	294
7.7 SSL 通信	224	9.5.4 停止使用照相机	294
7.7.1 SSL 通信模式	224	9.6 流媒体	294
7.7.2 SSL 服务端	225	9.7 媒体扫描和媒体库	294
7.7.3 SSL 客户端	230	9.7.1 媒体文件的扫描	294
7.8 浏览器	234	9.7.2 获取媒体文件信息	295
7.8.1 WebKit 介绍	235	9.8 音乐盒工具	297
7.8.2 Android 平台对 WebKit 引擎的 封装	235	第 10 章 个人信息管理	307
7.8.3 网页视图 (WebView)	235	10.1 个人信息管理	307
7.8.4 浏览器开发实例	237	10.2 Android 对个人信息管理的 支持	307
7.8.5 浏览器书签信息管理	249	10.3 联系信息	308
7.8.6 浏览器搜索记录	251	10.4 联系人信息	309
第 8 章 无线通信	254	10.5 电话号码信息	311
8.1 无线通信概述	254	10.6 组织 (公司) 信息	313
8.2 短消息	254	第 11 章 电话信息系统管理	316
8.2.1 Android 平台对短消息的支持	254	11.1 电话信息系统	316
8.2.2 发送短消息	255	11.2 Android 平台对电话信息 系统的支持	317
8.2.3 查看短消息	258	11.3 电话信息	317
8.2.4 接收短消息	260	11.4 电话状态	320
8.3 蓝牙通信	265		



11.5 电话拨号	325	13.4.3 XML Pull API 应用实例	413
11.6 呼叫日志	326	13.5 XML 资源解析	416
11.7 使用许可	328	13.5.1 解析 XML 布局资源	417
第 12 章 数据库应用	329	13.5.2 解析 XML 原文件资源	418
12.1 SQLite 数据库	329	13.5.3 解析 XML 资源	420
12.1.1 SQLite 数据库介绍	329	13.6 Android 平台 XML 使用小结	423
12.1.2 Android 平台对 SQLite		第 14 章 地图应用	424
数据库的支持	330	14.1 地图应用概述	424
12.1.3 SQLite 数据库应用模式	330	14.2 Android 平台对地图应用的	
12.1.4 开发实例	332	支持	425
12.1.5 基于 SQLite 的日记账工具	342	14.3 地图视图 (MapView) 应用	425
12.2 JDBC API	361	14.3.1 地图视图组件的定义	426
12.2.1 Android 平台对 JDBC API 的		14.3.2 获取地图 API 使用密钥	426
支持	362	14.3.3 工程设置	428
12.2.2 JDBC API 应用模式	362	14.3.4 Activity 组件框架	428
12.2.3 开发实例	367	14.3.5 地图 API 库设置	429
12.3 对象数据库 Db4o	376	14.3.6 地图 API 使用许可	429
12.3.1 Db4o 对 Android 平台的支持	375	14.4 地图 API 应用	429
12.3.2 Db4o API	375	14.4.1 界面布局定义	430
12.3.3 Db4o 数据库应用	376	14.4.2 地图 Activity 组件框架	430
12.3.4 基于 Db4o 的日记账工具	382	14.4.3 获取地图当前位置	433
12.4 数据库开发总结	389	14.4.4 地图叠加图管理	434
第 13 章 XML 应用	390	14.4.5 地图 API 使用小结	436
13.1 Android 平台对 XML 应用的		第 15 章 系统信息管理	438
支持	390	15.1 系统服务	438
13.2 SAX 解析方式	390	15.1.1 Android 系统服务介绍	438
13.2.1 SAX 使用模式	390	15.1.2 Activity 管理	439
13.2.2 Android 平台中对 SAX 的		15.1.3 警报管理	443
支持	391	15.1.4 音频管理	446
13.2.3 SAX 应用实例	391	15.1.5 剪贴板管理	448
13.3 DOM 解析方式	401	15.1.6 连接管理	449
13.3.1 DOM 使用模式	402	15.1.7 输入法管理	453
13.3.2 Android 平台中对 DOM 的		15.1.8 键盘守护	454
支持	402	15.1.9 通报管理	454
13.3.3 DOM 应用实例	403	15.1.10 布局填充	456
13.4 XML Pull API	412	15.1.11 位置服务管理	458
13.4.1 XML Pull API 使用模式	412	15.1.12 电源管理	459
13.4.2 Android 平台中对 XML		15.1.13 搜索服务	460
Pull API 的支持	412	15.1.14 传感器管理	461

15.1.15	电话管理	462
15.1.16	振动器管理	463
15.1.17	Wi-Fi 管理	463
15.1.18	墙纸管理	464
15.1.19	窗口管理	464
15.2	Android 平台系统信息	465
15.2.1	进程管理	465
15.2.2	文件系统信息	467
15.2.3	环境变量	467
15.2.4	系统时间管理	469
15.2.5	构建信息	473
15.2.6	电池状态	475
15.2.7	系统设置	477
15.2.8	安全设置	478
第 16 章 Android 资源及 SDK		
	工具	480
16.1	资源类型及定义	480
16.1.1	常量值资源	480
16.1.2	绘制用资源	490
16.1.3	布局资源	491
16.1.4	动画资源	493
16.1.5	菜单资源	496
16.1.6	文件资源	498
16.1.7	备选资源	500
16.2	资源的使用模式	501
16.2.1	资源 ID	502
16.2.2	资源引用	502
16.2.3	资源属性	502
16.3	系统资源定义	503
16.4	Android SDK 工具使用	503
16.4.1	adb 工具	503
16.4.2	sqlite3 工具	504
16.4.3	keytool 工具	504
第 17 章 Android 组件属性及使用许可		
17.1	Android 常用视图组件属性	505
17.1.1	视图 (View)	505
17.1.2	线性布局 (LinearLayout)	507
17.1.3	线性布局参数 (LinearLayout_Layout)	507
17.1.4	相对布局 (RelativeLayout)	507
17.1.5	相对布局参数 (RelativeLayout_Layout)	508
17.1.6	绝对布局参数 (AbsoluteLayout_Layout)	508
17.1.7	框布局 (FrameLayout)	509
17.1.8	框布局参数 (FrameLayout_Layout)	509
17.1.9	表格布局 (TableLayout)	509
17.1.10	表格行的单元 (TableRow_Cell)	509
17.1.11	抽象列表视图组件 (AbsListView)	509
17.1.12	列表视图组件 (ListView)	510
17.1.13	格子视图组件 (GridView)	510
17.1.14	画廊视图组件 (Gallery)	510
17.1.15	文本组件 (TextView)	511
17.1.16	自动完成文本框 (AutoCompleteTextView)	512
17.1.17	图片视图 (ImageView)	513
17.2	应用程序使用许可 (Uses-permissions)	513
17.2.1	使用许可的声明	513
17.2.2	Android 平台使用许可列表	514
附录 随书源代码说明		
参考文献		

第 1 章 初识 Android 平台

本章首先从多个视角对 Android 平台进行了简要介绍，继而详细地介绍了 Android 平台的内容，包括其发展历史和内涵，特别是 Android 平台用到的一些开源项目。以此为基础，全面介绍了 Android 平台的架构和应用程序内容以及 Android 平台定义的一些核心概念。最后对用户界面、程序资源和资产、数据存储以及平台安全和许可进行了举例说明。

1.1 Android 平台简介

“Android”一词的本义是“机器人”，如果牵强一点，其含义是雄性的机器人。作为一个手机平台项目的名字，该项目的丰富内涵和有效的市场运作，让“Android”的含义也变得多样化。

对于开源爱好者而言，“Android”指的是 Android 开源项目，图 1-1 是 Android 项目的 Logo。对于手机爱好者而言，“Android”可能指的是时下流行的 Android 手机，图 1-2 是 Android 手机产品实例图。而对于开发者来说，“Android”指的可能更多的是 Android 平台提供的框架和 SDK。



图 1-1 Android 项目 Log



图 1-2 Android 手机产品实例图

1.1.1 Android 发展历史

2007 年 11 月，Google 公司宣布其基于 Linux 平台的开源手机操作系统的项目名字为“Android”。

2008 年 3 月，Android SDK 发布，代号为 m5-rc15。

2008 年 8 月，Android 0.9 SDK beta 版本发布，代号 m5-0.9。提供了 Windows、Linux 32 位版、Mac OS X Intel 版。

2008 年 9 月，美国运营商 T-Mobile USA 在纽约正式发布第一款 Android 手机——T-Mobile G1。该款手机由宏达电（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 网络，并支持 Wi-Fi。

2008 年 9 月，Android 1.0 SDK 第 1 次发布。

2008 年 10 月，T-Mobile G1 手机开始销售。

2008 年 11 月，Android 1.0 SDK 第 2 次发布。

2009 年 2 月，Android 1.1 SDK 第 1 次发布。

2009 年 2 月，T-Mobile G2 (HTC Magic) 手机发布。

2009 年 4 月，Android SDK 1.5 预览版发布，代号为 Cupcake。

2009 年 4 月，在预览版本发布 2 周后，Android SDK 1.5 正式版发布。

2009 年 6 月，Android 平台的原生 C 语言的开发包 (NDK) 发布。

2009 年 7 月，Android 1.5 SDK 第 2 次发布。

2009 年 7 月，在 Android 1.5 SDK 第 2 次发布之后，进行第 3 次发布。

2009 年 9 月，Android 1.6 SDK 正式发布。

2010 年 1 月，Android 2.1 SDK 发布。

2010 年 5 月，在 Google 公司的 I/O 大会上，代号为 Froyo (冻酸奶) 的最新版 Android 2.2 操作系统发布。新版 Android 2.2 系统使用了 JIT (即时编译技术) 编译器，可以使程序运行速度提高 2~5 倍。就框架内容而言，2.2 版本的变化趋于稳定；就功能结构而言，2.2 版本较 2.1 更多的是优化和改进。

Android 平台的版本不断“推陈出新”，给手机厂商、运营商、开发者带来了不同程度的压力。据报道，Google 公司表示从 Froyo 开始，将逐步采取系统平台与应用程序进行分离的方式来加强用户体验和降低应用开发难度。

1.1.2 平台内涵

1. Android 平台的功能

(1) 提供应用程序框架，开发者可以遵照这些框架搭建应用程序。

读者可以结合 J2SE 平台的 Applet 框架或 J2ME 平台的 MIDlet (Mobile Information Device-let, 移动信息设备套件) 框架来理解 Android 平台的应用程序框架。

(2) 定制的 Dalvik 虚拟机。

读者可以结合 J2SE 平台的 JVM (Java Virtual Machine, Java 虚拟机) 和 J2ME 平台的 KVM (Kilo-bytes Virtual Machine, 千字节虚拟机) 来理解 Dalvik 虚拟机。

无论是 JVM 还是 KVM 都是参照 Java 虚拟机的技术规范来进行设计的，而 Dalvik 虚拟机是 DalvikVM.com 公司 (<http://www.dalvikvm.com/>) 开发的，其遵照的技术规范可能与一般的 Java 虚拟机不同。

Dalvik 虚拟机所支持的字节码 (ByteCode) 是“dex”文件 (Dalvik Executable, Dalvik 可执行文件)，也就是说 Dalvik 是不支持通常的 Java 类文件 (class 文件) 字节码的。

(3) 集成了基于 WebKit 开源项目的浏览器。

WebKit 是一个开源项目 (<http://webkit.org/>)，主要由 KDE (K Desktop Environment, K 桌面环境) 的 KHTML 修改而来，并且包含了一些来自苹果公司的组件。

传统上，WebKit 包含一个网页引擎 WebCore 和一个脚本引擎 JavaScriptCore，它们分别对应的是 KDE 的 KHTML 和 KJS。不过，随着 JavaScript 引擎的独立性越来越强，现在

WebKit 和 WebCore 已经基本上混用不分。

Google 公司开发的网页浏览器产品 Google Chrome 就是基于 WebKit 开源代码，并自行开发出称为“V8”的高性能 JavaScript 引擎。读者可以将 Android 平台的浏览器视为 Google 公司的浏览器产品的移动设备版本。

(4) 2D 和 3D 图形引擎，2D 图形引擎基于 SGL，3D 图形引擎基于 OpenGL ES 1.0 规范。

SGL (Skia Graphics Library, Skia 图形库) 是一套用于绘制文本、几何图形和图片的完整的 2D 图形库，被 Google 公司用于 Android 项目。

OpenGL ES 1.0 是基于 OpenGL 1.3 规范来定义的，同时增强了软件渲染和基本的硬件加速功能。读者可以从 <http://www.khronos.org/opengles/spec/> 获取 OpenGL ES 1.0 的规范。

(5) 提供 SQLite 数据库用于结构化数据存储。

SQLite 是一个能够嵌入到进程内部的库，同时它也是一个实现了独立性、无需服务器、零配置和事务处理的 SQL 数据库引擎。其官方网站为 <http://www.sqlite.org/>。

读者也可以把 SQLite 理解为一个嵌入式 SQL 数据库引擎，其无需单独的服务器进程，开发库小巧、可靠，支持大多数的系统平台，如 Linux、Mac OS X、Windows。

(6) 提供对音频、视频和图片等媒体的支持。

Android 平台使用 PacketVideo 公司制定的 OpenCore 框架来支持各种媒体服务。该框架为移动多媒体应用程序提供了一套通用的结构。

在 PacketVideo 公司 (<http://www.packetvideo.com/index.html>) 的网站上有对 OpenCore 框架在 Android 平台的应用说明：<http://www.packetvideo.com/products/android/index.html>。

Android 平台支持的媒体类型有：MPEG4 (mp4)、H.264、MP3、JPG、PNG、GIF 等。

(7) 提供 GSM 电话控制。

(8) 支持蓝牙、EDGE、3G 和 Wi-Fi。

EDGE (Enhanced Data rate for GSM Evolution, 增强型数据速率 GSM 演进) 是一种 GSM 到 3G 的过渡技术。

(9) 支持摄像头、GPS、罗盘和加速计等设备。

2. 与 Linux 平台的渊源

(1) Android 平台是在 Linux 2.6.25 版本的基础上改造的。Linux 内核体系结构如图 1-3 所示。

Linux 内核包括：系统调用接口、进程管理、内存管理、虚拟文件系统、网络堆栈、设备驱动和体系结构代码，也就是说 Android 平台内核也会包含这些内容。

明显不同的是，Android 的目标环境是 ARM 平台，而不是通常的 i386 平台。

(2) Android 模拟器是基于 Qemu 0.8.2 和 SDL (Simple Display Layer) 进行开发的模拟环境。

Qemu (<http://www.qemu.org/>) 是由 Fabrice Bellard 开发的一款开源的、支持多种 CPU 平台 (如 x86、ARM、SPARC、MIPS 等主流类型) 的模拟器。

SDL (Simple DirectMedia Layer, 简单直访媒体层) 是一款跨平台的多媒体开发库，用于提供对音频、键盘、鼠标、游戏杆、3D 硬件 (通过 OpenGL) 和 2D 视频的帧缓冲区的底层访问。SDL 支持 Linux、Windows、Windows CE、Mac OS、Mac OS X、FreeBSD 和 Solaris 等多个系统平台，其官方主页为 <http://www.libsdl.org/>。

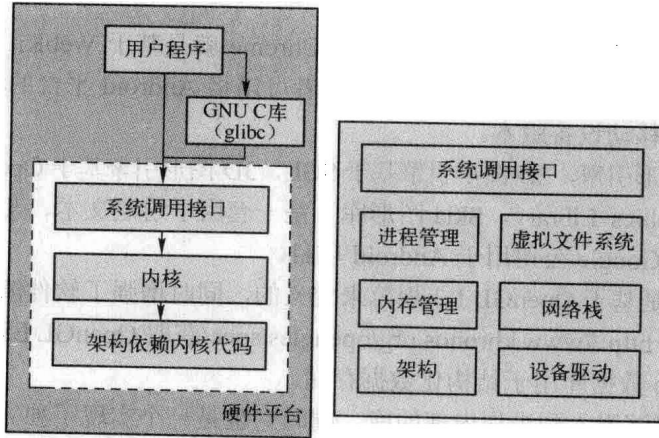


图 1-3 Linux 内核体系结构图

1.2 Android 平台架构

1.2.1 架构图

图 1-4 是 Android 平台的架构图。

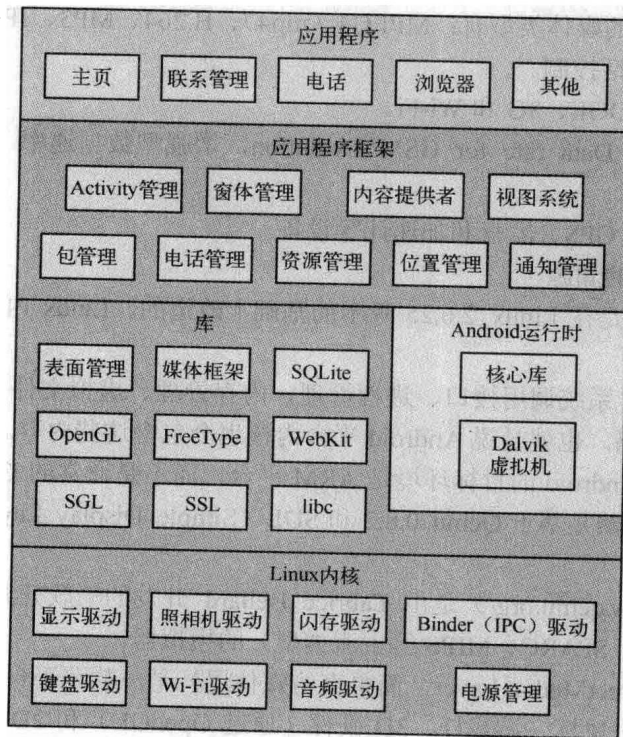


图 1-4 Android 平台架构图

1.2.2 架构内容

1. 应用程序

应用程序是包括 Android 平台配置的一套应用程序集，如短信程序、日历工具、地图浏览器、网页浏览器等工具，以及用户基于 Android 平台的应用程序框架，使用 Java 语言自行开发的程序。

2. 应用程序框架

开发者可以完全使用与那些内核应用程序相同的框架，这些框架用于简化和重用应用程序的组件。若某程序能够“暴露”其内容（如数据、功能模块），则其他程序就可以使用这些内容。

通过应用程序框架，用户自定义的程序可以执行用户程序之外的预设功能，这样可以极大减少用户程序的额外工作。如图 1-5 中，用户类（“HelloAndroidAct”）在其重载的方法（“onCreate”）中调用了 Android 应用程序框架预定义的父亲类（“Activity”）的方法。

```

HelloAndroidAct.java
package foolstudio.demo;

import android.app.Activity;

public class HelloAndroidAct extends Activity {
    /** Called when the activity is first created. */
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);
    }
}

```

图 1-5 子类通过框架执行平台预设功能

3. 系统开发库

Android 定义了一套 C/C++ 开发库供 Android 平台的其他组件使用。这些功能通过 Android 应用程序框架提供给开发者，开发者是不能直接使用这些库的。这些库包括：

(1) 系统 C 开发库

源于 BSD 的标准 C 系统库（libc）。

(2) 媒体开发库

基于 PacketVideo 的 OpenCore。

(3) 屏幕管理库

管理对显示子系统的访问或无缝衔接多个应用程序的 2D 和 3D 图形层。

(4) 网页浏览器引擎核心库

(5) SGL 库——2D 图形引擎库

(6) 基于 OpenGL ES 1.0 API 的 3D 开发库

(7) 基于开源项目 FreeType 的字体引擎开发库

FreeType (<http://www.freetype.org>) 是一款免费的、高质量的、可移植的字体引擎，其设计小巧、高效、高度可定制，并且可以产生可移植的高品质输出内容（符号图像）。该引擎已广泛用于图形库、展示服务、字体转换工具、文本图片生成工具和很多其他产品。