

Getting you the Best Books!

书中源代码下载地址: <http://www.tdpress.com/zyzx/tsscflwj>



Android

系统源代码分析

李俊 编著



全面分析和讲解Android系统源码的结构和工作机制



定位当前实践应用, 基于Android 主流版本的源码结构



书中所有概念和模块实例完全根据原生态源码来编写



从实际需要出发, 作者十年开源系统开发经验和盘托出

中国铁道出版社

CHINA RAILWAY PUBLISHING HOUSE

网络销售:

亚马逊 [amazon.cn](http://www.amazon.cn) (www.amazon.cn)

京东商城 www.jd.com)

当当网 www.dangdang.com)

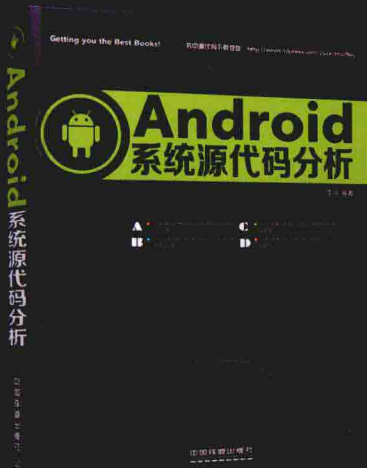
天猫 <http://zgtdcbs.tmall.com/>)



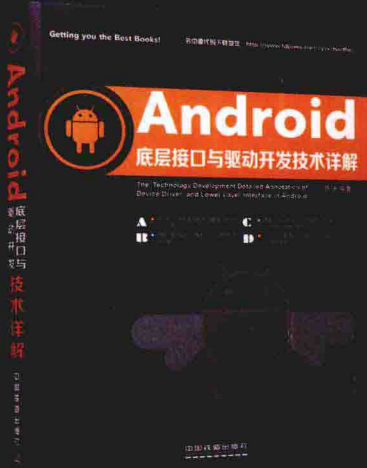
Android系统源代码分析

Source code analysis of Android

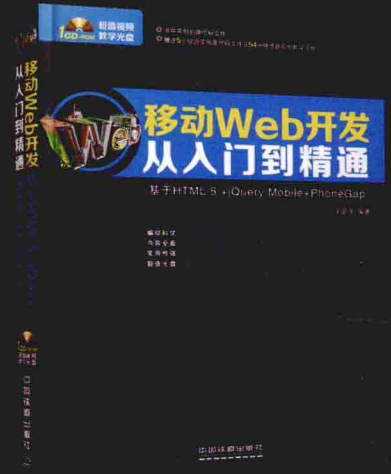
重点图书推荐:



书名: Android系统源代码分析
定价: 79.00元



书名: Android底层接口与驱动
开发技术详解
定价: 89.00元



书名: 移动Web开发从入门到精通
(基于HTML 5+jQuery Mobile+PhoneGap)
定价: 59.80元



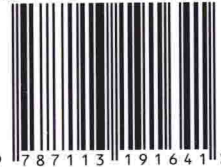
上架建议: 计算机/程序设计/Android



中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

地址: 北京市西城区右安门西街8号
邮编: 100054
网址: <http://www.tdpress.com>

ISBN 978-7-113-19164-1



9 787113 191641 >

定价: 79.00元

代码

代码

代码



Android 系统源代码分析

李俊 编著



TN 829.53
660

中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

内容简介

Android 从诞生到现在仅有短短几年的时间，它凭借其开源开放性、优异的用户体验和极为方便的开发方式，赢得了广大用户和开发者的青睐，目前已经发展成为市场占有率位居第一的智能手机操作系统。本书主要讲解 Android 系统源代码的结构和工作机制，完全基于 Android 主流版本的源代码结构来编写全书。本书内容言简意赅、通俗易懂、讲解详细。本书的内容涉及如何获取源代码、源代码的结构分析、init 解析、应用层结构分析、JNI 的机制分析、Android 中的系统服务和内存管理、硬件抽象层、Binder 机制、多媒体框架、音频系统、视频系统和传感器系统。在编写的过程中，书中的所有概念和模块实例完全根据原生态的源码来编写。

本书定位于有一定基础的 Android 程序开发人员，旨在帮助其通过细致分析 Android 系统源代码和工作机制，透彻理解 Android 系统架构，向 Android 系统开发殿堂级高手迈进。

图书在版编目 (CIP) 数据

Android 系统源代码分析 / 李俊编著. — 北京: 中国铁道出版社, 2015. 2

ISBN 978-7-113-19164-1

I. ①A… II. ①李… 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 200402 号

书 名: Android 系统源代码分析
作 者: 李 俊 编著

责任编辑: 荆 波
封面设计: 多宝格·付巍
责任印制: 赵星辰

读者热线: 400-668-0820
特邀编辑: 王惠凤

出版发行: 中国铁道出版社 (北京市西城区右安门西街 8 号 邮政编码: 100054)
印 刷: 三河市华业印务有限公司
版 次: 2015 年 2 月第 1 版 2015 年 2 月第 1 次印刷
开 本: 787mm×1092mm 1/16 印张: 35.25 字数: 834 千
书 号: ISBN 978-7-113-19164-1
定 价: 79.00 元

版权所有 侵权必究

凡购买铁道版图书, 如有印制质量问题, 请与本社教材图书营销部联系调换。电话: (010) 51873174

打击盗版举报电话: (010) 51873659

经过几年的发展，Android 已经从最初智能电话领域逐渐进入教育、医疗、军事、汽车、家居等重要领域。截至本书截稿，Android 系统已经成为移动平台领域当之无愧的王者，雄踞智能手机操作系统排行榜首位。目前，已有众多设备开始选择使用 Android 系统，比如智能手机、智能电视、平板电脑、上网本、MP3、MP4、智能相机等。相信在不久的将来，还将有更多采用 Android 系统的高科技产品进入人们的生活。这些设备将产生各种各样的应用需求，尤其是与 Android 系统底层相关的应用，这将给开发者带来很多机会，尤其是系统级应用开发工程师。

从技术角度而言，Android 是一种融入了全部 Web 应用的平台。随着版本的更新，从最初的触屏到现在的多点触摸，从普通的联系人到现在的数据同步，从简单的 Google Map 到现在的导航系统，从基本的网页浏览到现在的 HTML 5，都说明 Android 已经逐渐稳定，而且功能越来越强大。此外，Android 平台不仅支持 Java、C、C++ 等主流的编程语言，还支持 Ruby、Python 等脚本语言，甚至 Google 专为 Android 的应用开发并推出了 Simple 语言，这使得 Android 有着更加广泛的开发群体。

本书内容

本书主要分析和讲解 Android 系统源码的结构和工作机制，完全基于 Android 主流版本的源码结构来编写。本书内容言简意赅、通俗易懂、讲解详细。本书通过 12 个章节的篇幅，全面系统地讲解如何获取源码、源码的结构分析、init 解析、应用层结构分析、JNI 的机制分析、Android 中的系统服务和内存管理、硬件抽象层、Binder 机制、多媒体框架、音频系统、视频系统和传感器系统等内容，全面展现了 Android 系统架构。

本书特色

本书内容相当丰富，实例内容覆盖全面。笔者的目标是通过一本图书，提供多本图书的价值，读者可以根据自己的需要有选择地阅读。在内容的编写上，本书具有以下特色。

(1) 内容全面

本书的内容全面，主次清晰。在书中讲解了包括 Android 系统源码的各个方面，并且在编写的过程中，书中的所有概念和模块实例完全根据原生态的源码来编写。

(2) 结构合理

从用户的实际需要出发，科学安排知识结构，内容由浅入深，叙述清楚。全书精心筛选了最具代表性、读者最关心的知识点。

(3) 易学易懂

本书条理清晰、语言简洁，可帮助读者快速掌握各个知识点。使读者既可以按照本书编排的章节顺序进行学习，也可以根据自己的需求对某一章节进行针对性的学习。

(4) 实用性强

本书彻底摒弃枯燥的理论和简单的操作，注重实用性和可操作性，详细讲解了各个部分的源码知识，使用户掌握相关的操作技能的同时，还能学习到相应的基础知识。

读者对象

本书的读者对象包括以下人员。

- 从事 Android 系统开发的程序员。
- 研究 Android 原生态系统的开发者。
- 编程爱好者。
- 相关培训机构的老师和学员。

笔者在编写过程中，得到了中国铁道出版社工作人员的大力支持，正是各位编辑在工作中的求实、耐心和高效率才能使本书成功出版。另外也十分感谢笔者的家人，在笔者写作时给予了巨大的支持。同时，由于笔者水平有限，书中如有纰漏和不尽如人意之处在所难免，恳请读者提出意见或建议，以便修订并使之更臻完善。

笔 者

2014 年 12 月

第 1 章 Android 系统介绍

1.1 来到智能手机世界.....	1
1.1.1 何谓智能手机.....	1
1.1.2 百家争鸣.....	2
1.2 Android 的优势.....	3
1.2.1 优点一——系出名门.....	3
1.2.2 优点二——强大的开发团队.....	3
1.2.3 优点三——诱人的奖励机制.....	4
1.2.4 优点四——开源.....	5
1.3 剖析 Android 系统架构.....	5
1.4 五大组件.....	7
1.4.1 用 Activity 来表现界面.....	7
1.4.2 用 Intent 和 Intent Filters 实现切换.....	8
1.4.3 Service 服务.....	8
1.4.4 用 Broadcast Intent Receiver 发送广播.....	9
1.4.5 用 Content Provider 存储数据.....	9
1.5 进程和线程.....	9
1.5.1 先看进程.....	9
1.5.2 再看线程.....	9
1.6 获取 Android 源码.....	10
1.6.1 在 Linux 平台获取 Android 源码.....	10
1.6.2 在 Windows 平台获取 Android 源码.....	11
1.7 分析 Android 源码结构.....	13
1.8 编译源码.....	18
1.8.1 Ubuntu 系统编译源码.....	19
1.8.2 常见错误.....	19
1.8.3 运行 Android 源码.....	20
1.9 编译 Android Kernel.....	21
1.9.1 获取 Goldfish 内核代码.....	21
1.9.2 获取 MSM 内核代码.....	24
1.9.3 获取 OMAP 内核代码.....	24
1.9.4 编译 Android 的 Linux 内核.....	24
1.10 Android 模拟器 (AVD).....	26

1.10.1	在 Linux 环境下运行模拟器	26
1.10.2	模拟器辅助工具——adb	27
1.10.3	在 Windows 中创建 Android 模拟器	29
1.10.4	在模拟器上运行 Android 系统	30

第 2 章 源码结构和 SDK 解析

2.1	Android 源码的目录结构	32
2.2	目录结构在系统中的体现	33
2.2.1	应用程序	33
2.2.2	应用程序框架	35
2.2.3	系统服务	35
2.2.4	系统程序库	38
2.2.5	系统运行库	41
2.2.6	硬件抽象层	41
2.3	编译源码生成 SDK	42
2.4	源码中提供的接口	46
2.4.1	暴露的接口和隐藏的接口	46
2.4.2	开发中如何调用隐藏接口	52

第 3 章 分析 JNI 和 Media

3.1	JNI 基础	56
3.2	分析 Java 层的 MediaScanner	57
3.2.1	加载 JNI 库	57
3.2.2	实现扫描工作	59
3.2.3	读取并保存信息	60
3.2.4	删除不是 SD 卡中的文件信息	62
3.2.5	processDirectory	63
3.2.6	扫描函数 scanFile	64
3.2.7	JNI 中的异常处理	64
3.3	分析 JNI 层的 MediaScanner	65
3.3.1	将 Native 对象的指针保存到 Java 对象	65
3.3.2	创建 Native 层的 MediaScanner 对象	66
3.4	Native 层的 MediaScanner	66
3.4.1	JNI 函数的注册	66
3.4.2	完成注册工作	68
3.4.3	动态注册	70
3.4.4	processDirectory	72
3.4.5	client.scanFile	73
3.4.6	添加 TAG 信息	73

3.4.7	总结函数 JNI_OnLoad()与函数 JNI_OnUnload()的用途.....	74
3.4.8	Java 与 JNI 基本数据类型转换.....	75
3.4.9	JNIEnv 接口.....	76
3.4.10	JNI 中的环境变量.....	77
3.5	解析 android.process.media.....	78
3.5.1	MediaScannerReceiver.....	78
3.5.2	MediaScannerService.....	81
3.5.3	MediaProvider.....	86
3.6	MediaScanner 中的重要函数.....	99
3.6.1	seekTo.....	99
3.6.2	doProcessDirectory.....	103
3.6.3	doScanFile.....	105
3.6.4	android_media_MediaScanner_processFile.....	106
3.6.5	processFile.....	107
3.6.6	endFile.....	108
3.6.7	handleStringTag.....	108
3.6.8	fileMatchesExtension.....	109
3.7	Camera 模块 JNI 实例分析.....	110
3.7.1	Java 层预览接口.....	110
3.7.2	注册预览的 JNI 函数.....	111
3.7.3	C/C++层的预览函数.....	115

第 4 章 分析 init 进程

4.1	init 基础.....	116
4.2	分析入口函数.....	117
4.3	分析配置文件.....	120
4.3.1	init.rc 简介.....	121
4.3.2	分析 init.rc 的过程.....	123
4.4	解析 Service.....	127
4.4.1	zygote 对应的 service action.....	127
4.4.2	init 组织 Service.....	128
4.4.3	解析 Service 用到的函数.....	129
4.5	解析 on.....	134
4.5.1	zygote 对应的 on action.....	134
4.5.2	init 组织 on.....	135
4.5.3	解析 on 用到的函数.....	136
4.6	init 控制 Service.....	136
4.6.1	启动 zygote.....	137
4.6.2	启动 Service.....	137

4.6.3	总结四种启动 Service 的方式.....	142
4.7	控制属性服务.....	147
4.7.1	引入属性.....	147
4.7.2	初始化属性服务.....	150
4.7.3	启动属性服务.....	151
4.7.4	处理设置属性的请求.....	155

第 5 章 深入分析 HAL 层

5.1	初识 HAL 层.....	157
5.1.1	HAL 层简介.....	158
5.1.2	HAL_legacy 和 HAL 的对比.....	159
5.2	分析 HAL 层源码.....	159
5.2.1	分析 HAL module.....	159
5.2.2	文件 hardware.h.....	160
5.2.3	文件 hardware.c.....	162
5.3	分析硬件抽象层的加载过程.....	165
5.4	分析硬件访问服务.....	169
5.4.1	定义硬件访问服务接口.....	169
5.4.2	实现硬件访问服务.....	170
5.5	分析 mokoid 工程.....	171
5.5.1	直接调用 service 方法的实现代码.....	172
5.5.2	通过 Manager 调用 service 的实现代码.....	177
5.6	举例说明 Sensor (传感器) 在 HAL 层的表现.....	180
5.6.1	HAL 层的 Sensor 代码.....	180
5.6.2	Sensor 编程的流程.....	182
5.7	HAL 和系统移植.....	183
5.7.1	移植各个 Android 部件的方式.....	183
5.7.2	辅助工作.....	183

第 6 章 分析 Android 的内存机制

6.1	内存和进程的关系.....	190
6.1.1	进程管理工具的纷争.....	190
6.1.2	Android 系统内存设计.....	191
6.2	分析 Android 的进程通信机制.....	192
6.2.1	Android 的进程间通信 (IPC) 机制 Binder.....	192
6.2.2	Service Manager 是 Binder 机制的上下文管理者.....	193
6.2.3	分析 Server 和 Client 获得 Service Manager 的过程.....	211
6.3	分析 Android 系统匿名共享内存 C++调用接口.....	215

6.3.1	Java 程序	217
6.3.2	相关程序	226
6.4	内存优化	229
6.4.1	sp 和 wp 简析	229
6.4.2	详解智能指针 (android rebase 类 (sp 和 wp))	231

第 7 章 分析 ActivityManagerService

7.1	AMS 基础	233
7.1.1	进程数据类 ProcessRecord	234
7.1.2	数据类 HistoryRecord	234
7.1.3	类 TaskRecord	235
7.1.4	AMS 中的一些重要调度相关变量	235
7.2	Activity 的调度	237
7.2.1	startActivity() 的启动流程	237
7.2.2	stopActivityLocked() 停止 Activity	278

第 8 章 分析 Binder 和 MessageQueue

8.1	分析 Binder	285
8.1.1	Client、Server 和 ServiceManager	285
8.1.2	MediaServer	286
8.1.3	ServiceManager	312
8.1.4	分析 MediaPlayerService 和 Client	316
8.2	初始化 Java 层 Binder 框架	323
8.2.1	Binder 类的初始化	324
8.2.2	addService 实例分析	325
8.3	分析 MessageQueue	334
8.3.1	创建 MessageQueue	335
8.3.2	提取消息	335
8.3.3	分析函数 nativePollOnce	340

第 9 章 分析 SystemServer

9.1	分析 SystemServer	350
9.1.1	分析主函数 main	350
9.1.2	分析函数 init2	353
9.2	分析 EntropyService	353
9.3	分析 DropBoxManagerService	356
9.3.1	分析 DBMS 构造函数	356
9.3.2	添加 dropbox 日志文件	358

9.3.3	DBMS 和 Settings 数据库	362
9.4	分析 DiskStatsService	363
9.5	分析 DeviceStorageManagerService	368
9.6	分析 SamplingProfilerService	371
9.6.1	分析 SamplingProfilerService 构造函数	371
9.6.2	分析 SamplingProfilerIntegration	372
9.7	分析 ClipboardService	382
9.7.1	复制数据到剪贴板	383
9.7.2	从剪贴板粘贴数据	384
9.7.3	CBS 中的权限管理	387

第 10 章 分析 WindowManagerService

10.1	WindowManagerService 概述	391
10.2	分析计算 Activity 窗口大小的过程	393
10.2.1	概览	393
10.2.2	分析 Activity 计算窗口大小的过程	394
10.3	分析 WindowManagerService 对窗口的组织方式	419
10.3.1	概述	419
10.3.2	分析操作源码	420

第 11 章 分析安装应用程序的过程

11.1	WindowManagerService 概述	439
11.2	分析安装过程	441
11.2.1	主函数 main	441
11.2.2	初始化函数	442
11.2.3	初始化服务	442
11.2.4	调用静态成员函数	446
11.2.5	创建了 ServerThread 线程	447
11.2.6	启动服务	447
11.2.7	创建 PackageManagerService 服务实例	452
11.2.8	扫描函数 scanDirLI	460
11.2.9	归档处理	464
11.2.10	解析 application 标签	475
11.2.11	保存解析后的程序信息	477

第 12 章 分析核心框架系统

12.1	分析音频系统	479
12.1.1	Audio 系统的层次介绍	479

12.1.2	Media 库中的 Audio 框架	481
12.1.3	本地代码	488
12.1.4	JNI 代码	491
12.1.5	Java 代码	492
12.2	视频输出系统	492
12.2.1	Overlay 系统的基本层次结构	492
12.2.2	Overlay 系统硬件抽象层的接口	494
12.2.3	实现硬件抽象层	496
12.2.4	实现接口	497
12.3	分析 OpenMax 多媒体框架	498
12.3.1	分析 OpenMax IL 层的接口	499
12.3.2	OpenMax 适配层	504
12.4	分析多媒体插件框架	506
12.4.1	基本层次结构	507
12.4.2	分析库 libopencorecommon.so 的结构	508
12.4.3	分析库 libopencoreplayer.so 的结构	509
12.4.4	分析库 libopencoreauthor.so 的结构	510
12.4.5	其他的库	511
12.4.6	OpenCore OSCL	511
12.4.7	实现 OpenCore 中的 OpenMax 部分	513
12.5	分析 Camera 照相机系统	525
12.5.1	基本层次结构	525
12.5.2	分析 V4L2 驱动程序	527
12.5.3	硬件抽象层	534
12.5.4	实现 Camera 系统的硬件抽象层	535
12.6	分析传感器系统	542
12.6.1	基本层次结构	542
12.6.2	分析硬件抽象层	544
12.6.3	实现上层部分	545

第 1 章 Android 系统介绍

Android 是一种智能手机系统，它是建立在 Linux 基础之上的，能够迅速建立手机软件的解决方案。虽然 Android 的外形比较简单，但是它的功能十分强大。自从 2007 年诞生以来，Android 迅速成为新兴的热点，并在 2011 年的智能手机占有率中位居第一。本章将简单介绍 Android 的发展历程和背景，让读者了解 Android 的发展之路，真切体会 Android 如此火爆的原因。

1.1 来到智能手机世界

在 Android 系统诞生之前，智能手机这个新鲜事物大大丰富了人们的生活，并得到了广大手机用户的青睐。各大手机厂商在利益的驱动下，纷纷建立了各种智能手机操作系统，并且大肆招兵买马来抢夺市场份额。Android 系统就是在这个风起云涌的历史背景下诞生的。

1.1.1 何谓智能手机

智能手机就是指聪明的手机，聪明得让人无法想象它的功能究竟有多大。现在的计算机处理速度非常快，能够每秒钟计算几百亿次。可以说现在的智能手机就是一个移动计算机，能够完成大多数计算机可以实现的功能。

究竟如何才能算作智能手机呢？以往没有标准，直到一个号称权威数据中心的统计机构做了一份市场调查，根据调查结果得出了智能手机的“门槛”。要想成为智能手机，就必须具备“门槛”中的如下 5 个标准。

- (1) 操作系统必须支持新应用的安装。
- (2) 高速度处理芯片。
- (3) 支持播放式的手机电视。
- (4) 大存储芯片和存储扩展能力。
- (5) 支持 GPS 导航。

制订上述标准的机构毕竟不是官方组织，于是产生了各抒己见的情况，有的部门建议应该将拍照功能列入智能手机的标准，有的建议应该将录像功能列为标准……为了解决各方意见不统一的问题，手机界的官方组织“手机联盟”出面制订了一个标准，其中总结了如下几条智能手机的特点。

- (1) 具备普通手机的全部功能，例如可使用户进行正常的通话和发短信等手机应用。
- (2) 是一个开放性的操作系统，在系统平台上可安装更多的应用程序，从而实现功能的无限扩充。
- (3) 具备上网功能。
- (4) 具备 PDA 的功能，可实现个人信息管理、日程记事、任务安排、多媒体应用、浏览网页。
- (5) 可根据个人需要扩展设备的功能。
- (6) 扩展性能强，并且可支持很多第三方软件。

1.1.2 百家争鸣

当今市面中有很多智能手机系统，形成了百家争鸣的局面。但是最受用户欢迎的当属微软、塞班、PDA、黑莓、苹果和本书的主角 Android。

1. 微软的 Windows Mobile

微软是 PC 系统的巨头，也是软件产业的革新者。正是因为拥有 PC 系统和出色开发团队的支持，其智能手机系统 Windows Mobile 也是一款深受用户欢迎的产品，在 Windows Mobile 中 Windows 桌面扩展到了个人设备中。使用 Windows Mobile 操作系统的设备主要有 PPC 手机、PDA、随身音乐播放器等。Windows Mobile 操作系统有三种，分别是 Windows Mobile Standard、Windows Mobile Professional 和 Windows Mobile Classic。

当前微软最新的版本是 Windows Phone 7，下一代版本是 Windows Phone 8。

2. 塞班系统 Symbian

塞班系统是由诺基亚、索尼爱立信、摩托罗拉、西门子等几家大型移动通信设备商共同出资组建的一个合资公司专门研发的手机操作系统。该合资公司现已被诺基亚全额收购。Symbian 具有良好的界面，采用内核与界面分离技术，对硬件的要求比较低，支持 C++、Visual Basic 和 J2ME。目前根据人机界面的不同，Symbian 体系的 UI (User Interface 用户界面) 平台分为 Series 60、Series 80、Series 90、UIQ 等。其中 Series 60 主要用于数字键盘手机，Series 80 是为完整键盘的手机而设计的，Series 90 则是为触控笔方式而设计的。

在过去的 8 年里，诺基亚凭借塞班系统的杰出表现，一直雄踞智能手机销量的榜首。无论是诺基亚还是塞班，在各自的领域都是毫无争议的老大。但是随着 Android 和 iPhone 的崛起，塞班入不敷出。2011 年，诺基亚宣布将和微软的 Windows Phone 系统合作。

3. Palm

Palm 是流行的个人数字助理 (PDA，又称掌上电脑) 的传统名字。从广义上讲，Palm 是 PDA 的一种，是 Palm 公司发明的。而从狭义上讲，Palm 是 Palm 公司生产的 PDA 产品，区别于 SONY 公司的 Clie 和 Handspring 公司的 Visor/Treo 等其他运行 Palm 操作系统的 PDA 产品。其显著特点之一是写入装置输入数据的方法，能够点击显示器上的图标选择输入的项目。2009 年 2 月 11 日，Palm 公司 CEO Ed Colligan 宣布以后将专注于 WebOS 和 Windows Mobile 的智能设备，而将不会再推出基于“Palm OS”的智能设备，而 Palm Centro 还会在以后和其他运营商合作时继续推出。

4. 黑莓 BlackBerry

BlackBerry 是加拿大 RIM 公司推出的一种移动电子邮件系统终端,其特色是支持推动式电子邮件、手提电话、文字短信、互联网传真、网页浏览及其他无线资讯服务,其最大优势在于收发邮件。

5. iPhone

iPhone 是由苹果公司(Apple,Inc.)首席执行官史蒂夫·乔布斯在 2007 年 1 月 9 日举行的 Macworld 宣布推出的,并于 2007 年 6 月 29 日在美国上市。iPhone 将创新的移动电话、可触摸宽屏 iPod 及具有桌面级电子邮件、网页浏览、搜索和地图功能的突破性因特网通信设备这三种产品完美地融为一体。可以毫不夸张地说每一件 iPhone 产品都是独具匠心,无论是外观还是品质和性能,都拥有超高的人气。

6. Android

Android 是本书的主角,是于 2007 年 11 月 5 日宣布的基于 Linux 平台的开源手机操作系统的名称,该平台由操作系统、中间件、用户界面和应用软件组成,号称是首个为移动终端打造的真正开放和完整的移动软件。

在 Android 推出之前,智能手机系统领域塞班、苹果、微软互不相让,三足鼎立之势日渐明朗。2007 年下半年 Android 突然神秘崛起,并宣称完全免费,它的强势大有颠覆三足鼎立之势。虽然崛起较晚,但 Android“系出名门”,出身于“豪门望族”的 Linux 家族。Android 采用了 WebKit 浏览器引擎,具备触摸屏、高级图形显示和上网功能,用户能够在手机上查看电子邮件、搜索网址和观看视频节目等,同时 Android 还具有比 iPhone 等其他手机更强的搜索功能,可以说是一种融入全部 Web 应用的平台。

1.2 Android 的优势

Android 在 4 年的时间内超越了各个“大佬”,而成为智能手机占有率最高的一款系统。为何 Android 能在这么多的智能系统中脱颖而出,成为市场占有率位居第一的手机系统呢?要想分析其原因,需要了解它的巨大优势,分析究竟是哪些优点吸引了厂商和消费者的青睐。

1.2.1 优点一——系出名门

Android 出身于 Linux 世家,是一款开源的手机操作系统。Android 功成名就后,各大手机联盟纷纷加入。这个联盟由中国移动、摩托罗拉、高通、宏达电和 T-Mobile 在内的三十多家技术和无线应用的领军企业组成。通过与运营商、设备制造商、开发商和其他有关各方结成深层次的合作伙伴关系,Android 希望借助建立标准化、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。

1.2.2 优点二——强大的开发团队

Android 的研发队伍阵容强大,包括摩托罗拉、Google、HTC(宏达电子)、PHILIPS、T-Mobile、高通、魅族、三星、LG 及中国移动在内的 34 家企业,这些都是在手机江湖中享誉盛名的“大佬”。其研发队伍将基于该平台开发手机的新型业务,应用之间的通用性和互联性,并且还成立了手机开放联盟,联盟中的成员名单如下。

(1) 手机制造商

联盟中的手机制造商包括台湾宏达国际电子 (HTC) (Palm 等多款智能手机的代工厂), 摩托罗拉 (美国最大的手机制造商), 韩国三星电子 (仅次于诺基亚的全球第二大手机制造商), 韩国 LG 电子, 中国移动 (全球最大的移动运营商), 日本 KDDI (2900 万用户), 日本 NTT DoCoMo (5200 万用户), 美国 Sprint Nextel (美国第三大移动运营商, 5400 万用户), 意大利电信 (Telecom Italia) (意大利主要的移动运营商, 3400 万用户), 西班牙 Telefónica (在欧洲和拉美有 1.5 亿用户), T-Mobile (德意志电信旗下公司, 在美国和欧洲有 1.1 亿用户)。

(2) 半导体公司

联盟中的半导体公司包括 Audience Corp (声音处理器公司), Broadcom Corp (无线半导体主要提供商), 英特尔 (Intel), Marvell Technology Group, Nvidia (图形处理器公司), SiRF (GPS 技术提供商), Synaptics (手机用户界面技术), 德州仪器 (Texas Instruments), 高通 (Qualcomm), 惠普 HP (Hewlett-Packard Development Company, L.P.)。

(3) 软件公司

联盟中的软件公司包括 Aplex, Ascender, eBay 的 Skype, Esmertec, Living Image, NMS Communications, Noser Engineering AG, Nuance Communications, PacketVideo, SkyPop, Sonix Network, TAT-The Astonishing Tribe, Wind River Systems。

1.2.3 优点三——诱人的奖励机制

Google 为了提高程序员的开发积极性, 不但为他们提供了一流的硬件设置, 提供了一流的软件服务, 而且还实行了振奋人心的奖励机制, 例如定期召开开发比赛, 其创意和应用夺魁的程序员将会得到重奖。

1. 开发 Android 平台的应用

在 Android 平台上, 程序员可以开发出各式各样的应用。Android 应用程序是通过 Java 语言开发的, 程序员只要具备 Java 开发基础, 就能很快上手并掌握。单独的 Android 开发对 Java 的编程门槛并不高, 即使没有编程经验的门外汉, 也可以在突击学习 Java 后来学习 Android。另外, Android 完全支持 2D、3D 和数据库, 并且和浏览器实现了集成。所以通过 Android 平台, 程序员可以迅速、高效地开发出功能强大的应用, 例如常见的系统工具、文件管理和互联网游戏等。

2. 奖金丰厚的 Android 大赛

为了吸引更多的用户使用 Android 开发, 相关部门已成功举办了奖金为 1000 万美元的开发者竞赛, 鼓励开发人员开发出创意十足、十分有用的软件。这种大赛有利于开发人员提高其开发水平, 并且高额的奖金成为他们的动力。

3. 在 Android Market 中获取收益

为了能让 Android 平台得到更多的关注, Google 开发了自己的 Android 软件下载店 Android Market。Android Market 地址是 <http://www.Android.com/market/>, 允许开发人员将应用程序在其中发布, 也允许 Android 用户随意下载自己喜欢的程序。作为开发者, 需要申请开发者账号, 申请后才能将自己的程序上传到 Android Market, 并且可以对自己的软件进行定价。只要开发者的软件程序足够吸引人, 开发者就可以获得很好的金钱回报, 从而学习、赚钱两不误。