

以Android源代码为主，SDK为辅，以提升Android开发工程师的应用开发水平为目标，深入解析了必须掌握的Android底层核心技术，并展示了大量最佳实践，为开发高质量的Android应用提供了绝佳指导

移动开发



苗忠良 曾旭 宛斌◎著

Pro Android Programming: Key Points and Best Practice

深入

应用开发

核心技术解析与最佳实践



机械工业出版社
China Machine Press

Pro Android Programming: Key Points and Best Practice

深入Android应用开发

核心技术解析与最佳实践

苗忠良 曾旭 宛斌◎著



机械工业出版社
China Machine Press

如何才能真正进阶为 Android 应用开发高手？必须深入理解 Android 核心技术的底层原理和在开发中总结并使用各种最佳实践，别无他法！本书以 Android 的源代码为主，SDK 为辅，针对应用开发者的需求，对各种核心技术的使用方法、底层原理和实现细节进行了深入而详细的讲解，同时辅之以大量案例和最佳实践，为开发者的进阶修炼和开发出高质量的应用提供了绝佳指导。

第 1 章深入介绍了 Android 的 4 大核心组件和 9 大主要应用框架；第 2 章透彻讲解了资源、组件、事件处理、应用风格和数据适配等核心知识；第 3 章系统地展示了各种常用的 UI 控件的使用方法及原理；第 4 章深入解析了 Android 的数据存储与管理；第 5 章分析了 Android 在 UI 层、框架层、内核层的通信机制和原理；第 6 章深入讲解了 Android 实现多线程的两种主要方式（原生 Java 线程和 Android 对 Java 线程的封装）的原理，并对它们进行了全方位的比较；第 7 章深入介绍了 Android 网络编程的核心技术；第 8 章详细讲解了 Android 中的 2D 图像、3D 图像、2D 图形以及动画方面的知识；第 9 章全面介绍了 Android 的多媒体编程，包括音频、视频和 Camera 服务等，并对 TTS 的实现进行了剖析；第 10 章详细讲解了 Android 的跨语言调用；第 11 章深入剖析了 Android 的安全框架和安全机制；第 12 章针对商业应用的需要讲解了 Android 的调试、测试与性能优化；第 13 章全面地剖析了 Android 的编译系统，包括编译工具、映像文件编译、SDK 编译、NDK 编译和目标系统的配置等；第 14 章分析了 Android 系统的启动过程；第 15 章讲解了 Android 的系统管理原理，包含内存管理、应用管理、电源管理、系统管理，以及系统的还原、升级、配置和备份等多方面的内容，能帮助开发者从一个更高的视角去理解 Android 的运行机制。

封底无防伪标均为盗版

版权所有，侵权必究

本书法律顾问 北京市展达律师事务所

图书在版编目（CIP）数据

深入 Android 应用开发：核心技术解析与最佳实践 / 苗忠良，曾旭，宛斌著. —北京：机械工业出版社，2012.5

ISBN 978-7-111-37957-7

I. 深… II. ①苗… ②曾… ③宛… III. 移动电话机—应用程序—程序设计 IV. TN929.53

中国版本图书馆 CIP 数据核字（2012）第 063938 号

机械工业出版社（北京市西城区百万庄大街 22 号 邮政编码 100037）

责任编辑：孙海亮

北京京师印务有限公司印刷

2012 年 6 月第 1 版第 1 次印刷

186mm×240mm·29.25 印张

标准书号：ISBN 978-7-111-37957-7

定价：79.00 元

凡购本书，如有缺页、倒页、脱页，由本社发行部调换

客服热线：(010) 88378991；88361066

购书热线：(010) 68326294；88379649；68995259

投稿热线：(010) 88379604

读者信箱：hzjsj@hzbook.com



前言

为什么写作本书

作为一个应势而生的移动终端操作系统，Android 适应了移动互联网的发展和“4C”融合的趋势，为产业从以硬件为重心向以内容和服务为重心转型提供了一个绝佳的平台。更由于其发起者 Google 的正确商业推广策略，自发布之日起，Android 便赢得了众多开发者和 OEM 厂商的青睐与支持，支持者如滚雪球般不断扩大，最终后来居上。由于其面向互联网设计的特点，Android 适用于数字家庭、远程医疗、物联网、交通监控、移动终端、机器人等多种应用领域，能很好地适应移动互联网的发展，其前景被越来越多的人看好。Android 的出现，加速了以“内容”和“服务”为重心的新一代信息产业革命的发展。

随着移动互联网的发展，科技的魅力逐渐渗透到世界的每个角落，但是随着科技的普及，至少在消费类电子领域，以技术驱动为核心的价值观理念正逐渐让位于以消费需求为核心的价值观理念。衡量产品成功的标准不再局限于技术是否先进，而是能否最大限度地满足消费者需求。Android 的出现正逢其时。

然而 Android 庞大的架构背后蕴含的设计理念和逻辑细节异常复杂，入门容易，深入却难，需要开发者仔细钻研。Android 的知识点跨越了 Linux、Java、C/C++、XML、移动终端、无线接入技术、用户研究和计算机网络等多个领域，整合了计算机网络、嵌入式系统、移动通信、集成电路、工业设计、人机交互、设计艺术学等多个学科。

在本书中，根据以往和读者交流的经验，笔者摒弃了以前侧重理论和理念引导的写作方法，更注重实战经验的总结。从哲学角度看，这是一种退步，但是就具体的开发和与技术的适配来看，这种写作方法更加有利于将笔者的心得分享给读者，帮助读者迅速提高自己的开发水平。

然而必须申明，任何平台和工具均是开发者表达自己理念的一种手段，永远无法也不应该替代开发者头脑中的应用的设计理念，尤其是在进行商业开发的过程中，应始终明确设计理念和商业模式远比工具本身更重要。API 仅是表达设计意图的一个途径，必须将个人的视

野拔高到设计的层次。在商务实践中，战略和执行力构成个人和企业成功的两大基石，自身的技术能力和能够利用的资源则决定着产品的质量。只懂使用工具的人称不上优秀的人才，相比内外兼修的“中华功夫”高手，不过徒然练就了“一身外功”而已。只有理论与实践兼修，才能够深刻了解产业的发展趋势，才能找到正确的切入点，才能体会到工程实践的重要意义，才能理解优秀的产品其实是一种平衡者才能获得的更高成就，而前进的每一步都是在不断否定自我的过程中实现的。

秉持一贯的作风，本书少有完整的例子供开发者参考，因为在 Android SDK 所携带的文档中，已经有足够多的示例供开发者学习，笔者没有必要在此着墨太多。

在具体的内容上，本书以源代码为主、SDK 为辅的方式与大家分享经验心得。希望本书能够帮助 Android 爱好者一窥 Android 内在的神秘。

本书面向的读者

本书适合所有有一定基础的 Android 应用开发工程师和系统开发工程师阅读。

本书主要内容

本书在逻辑上力避耦合，读者可以根据自己的需要调整阅读顺序。缺乏 Android 基础的读者请务必先详细阅读本书的附录。

本书第 1 ~ 4 章介绍 Android 基础开发涉及的核心组件、应用框架、资源框架、UI 控件、数据存储等，这是所有 Android 开发者都应掌握的；第 5 ~ 10 章介绍 Android 框架层的通信机制、多线程编程、网络编程、图像与动画、多媒体编程、跨语言调用等内容，这些是中等水平的 Android 开发者必须掌握的；第 11 ~ 15 章则介绍了 Android 的安全框架、调试与优化、编译环境、启动过程和系统管理，以帮助开发者完成更高层次的积累并为商业级产品开发做好准备。

支持和勘误

除了苗忠良、曾旭、宛斌之外，周红恩、王振伟、李德伟、张婷婷、刘存青、王珊珊等还参与了资料的整理和示例的收集等工作。

由于笔者水平有限，书中难免还存在一些不足，请大家见谅。若您对本书有任何意见和建议，欢迎发邮件到 mzl626@163.com 和笔者讨论。

本书中所涉及的源代码可在 <http://source.android.com> 处下载。

致谢

在本书撰写过程中，得到了公司同事管艳丽、陈文、唐晓军、何伟伟等和家人的大力鼓励和支持，他们为本书提供了很多好的意见和建议，使本书得以不断完善，在此对他们表示深深的谢意，我很高兴能在创业的道路上和他们一路同行；另外，我的儿子苗思源的降生带给了我极大的快乐，每天看到他天真无邪的笑脸，都觉得是一种莫大的幸福，这种幸福激励着我不断奋斗。

另外，还要感谢机械工业出版社的编辑、美工和校对人员，尤其是策划编辑杨福川先生，及姜影和白宇两位编辑。在本书撰写过程中，他们提出了不少真知灼见，很高兴能和他们一起推动本书的出版。

苗忠良



前言

第 1 章 深入解析 Android 核心组件和应用框架 / 1

- 1.1 核心组件详解 / 1
 - 1.1.1 Activity 组件 / 1
 - 1.1.2 Service 组件 / 19
 - 1.1.3 Intent 组件 / 25
 - 1.1.4 BroadcastReceiver 组件 / 31
- 1.2 应用框架解析 / 31
 - 1.2.1 Service 框架 / 32
 - 1.2.2 Activity 管理机制 / 38
 - 1.2.3 Broadcast 机制 / 41
 - 1.2.4 对话框框架 / 42
 - 1.2.5 标题栏框架 / 46
 - 1.2.6 状态栏框架 / 48
 - 1.2.7 通知机制 / 52
 - 1.2.8 搜索框架 / 56
 - 1.2.9 ActionBar 框架 / 59

第 2 章 Android 资源框架详解 / 64

- 2.1 布局文件 / 64
- 2.2 值文件 / 67
- 2.3 创建菜单 / 75

- 2.3.1 选项菜单 / 75
- 2.3.2 上下文菜单 / 77
- 2.3.3 子菜单 / 79
- 2.3.4 弹出菜单 / 80
- 2.4 断言的处理 / 81
- 2.5 Jar 包和共享库 / 83
- 2.6 系统资源 / 84

第3章 Android UI 控件精要 / 89

- 3.1 布局控件 / 89
 - 3.1.1 LinearLayout / 89
 - 3.1.2 RelativeLayout / 92
 - 3.1.3 TableLayout / 93
 - 3.1.4 FrameLayout / 94
- 3.2 显示控件 / 97
 - 3.2.1 TextView / 97
 - 3.2.2 ProgressBar / 99
 - 3.2.3 WebView / 100
 - 3.2.4 ViewSwitcher / 101
 - 3.2.5 TextSwitcher / 102
 - 3.2.6 ViewFlipper / 103
 - 3.2.7 SeekBar / 104
 - 3.2.8 ViewStub / 106
- 3.3 容器控件 / 106
 - 3.3.1 SlidingDrawer / 106
 - 3.3.2 ListView / 107
 - 3.3.3 ExpandableListView / 111
 - 3.3.4 GridView / 112
 - 3.3.5 ScrollView / 113
 - 3.3.6 TabHost / 114
- 3.4 输入控件 / 114
 - 3.4.1 EditText / 114
 - 3.4.2 Spinner / 116

- 3.4.3 AutoCompleteTextView / 117
- 3.5 控件 / 118
 - 3.5.1 Button / 118
 - 3.5.2 CheckBox / 119
 - 3.5.3 RadioButton / 119
 - 3.5.4 RatingBar / 120
 - 3.5.5 ZoomControls / 121
- 3.6 多媒体控件 / 122
 - 3.6.1 ImageView / 122
 - 3.6.2 ImageSwitcher / 122
- 3.7 特殊控件 / 125
- 3.8 桌面部件 / 126
 - 3.8.1 App Widgets 的实现 / 126
 - 3.8.2 Live folders 的实现 / 128
- 3.9 自定义控件 / 129
- 3.10 数据适配 / 130
 - 3.10.1 视图适配 / 130
 - 3.10.2 标准化解析 / 133
- 3.11 Fragment / 135
 - 3.11.1 Fragment 生命周期 / 135
 - 3.11.2 ListFragment / 137
 - 3.11.3 DialogFragment / 137
 - 3.11.4 PreferenceFragment / 138
 - 3.11.5 WebViewFragment / 138
- 第 4 章 深入解析 Android 数据存储与管理 / 139**
 - 4.1 XML 文件管理 / 139
 - 4.2 内部文件管理 / 140
 - 4.2.1 写入数据 / 141
 - 4.2.2 读取数据 / 141
 - 4.3 外部文件管理 / 142
 - 4.4 数据库管理 / 143
 - 4.4.1 Android 封装接口 / 144

- 4.4.2 原生方法处理 / 145
- 4.5 数据处理 / 145
 - 4.5.1 Set 类 / 146
 - 4.5.2 Map 类 / 148
 - 4.5.3 list 类 / 150
 - 4.5.4 流类 / 150
 - 4.5.5 正则表达式 / 153
 - 4.5.6 I/O 管理 / 153
 - 4.5.7 JSON 实现 / 155

第 5 章 深入解析 Android 通信机制 / 159

- 5.1 Intent 通信 / 159
 - 5.1.1 Intent 的常见用法 / 159
 - 5.1.2 Serializable 接口 / 165
 - 5.1.3 Parcelable 接口 / 165
- 5.2 UI 事件处理 / 166
 - 5.2.1 事件监听器、事件句柄及焦点处理 / 167
 - 5.2.2 触控事件处理 / 168
 - 5.2.3 按键事件处理 / 170
 - 5.2.4 轨迹球事件处理 / 171
- 5.3 任务调度 / 172

第 6 章 深入解析 Android 多线程编程 / 174

- 6.1 Java 线程实现 / 174
- 6.2 Android 线程封装 / 177
- 6.3 线程间的消息通信 / 180
 - 6.3.1 消息队列 / 181
 - 6.3.2 消息分发 / 183
 - 6.3.3 消息接收 / 183
- 6.4 线程安全处理 / 184
 - 6.4.1 synchronized 同步 / 184
 - 6.4.2 RPC 通信 / 186
 - 6.4.3 SQLite 调用 / 186

第7章 深入解析 Android 网络编程 / 187

- 7.1 无线接入技术概述 / 187
- 7.2 基础协议封装 / 190
- 7.3 Java 网络编程接口 / 194
- 7.4 Apache 网络编程接口 / 195
- 7.5 Android 网络编程接口 / 196
- 7.6 Web 服务实现 / 199
 - 7.6.1 Web 服务概述 / 199
 - 7.6.2 KSOAP2 的实现 / 199
- 7.7 XML 解析器 / 200
 - 7.7.1 Pull 解析器 / 201
 - 7.7.2 DOM 解析器 / 202
 - 7.7.3 SAX 解析器 / 203
- 7.8 套接字编程 / 203
- 7.9 Web 应用实现 / 205
- 7.10 SIP 服务 / 212
- 7.11 NFC 通信 / 214
- 7.12 RIL 层处理 / 216
- 7.13 报文分析 / 220

第8章 Android 图形、图像与动画精要 / 222

- 8.1 2D 图像处理 / 222
 - 8.1.1 基本接口 / 222
 - 8.1.2 缩略图 / 228
 - 8.1.3 图像浏览 / 228
 - 8.1.4 人脸检测 / 230
- 8.2 3D 图像处理 / 233
 - 8.2.1 OpenGL ES 的实现 / 233
 - 8.2.2 RenderScript 的实现 / 236
- 8.3 图形处理 / 239
 - 8.3.1 基本接口 / 239
 - 8.3.2 Surface 渲染系统 / 244
- 8.4 动画处理 / 247

8.4.1 补间动画 / 248

8.4.2 帧动画 / 249

8.4.3 属性动画 / 250

第 9 章 深入解析 Android 多媒体编程 / 257

9.1 音频处理 / 257

9.1.1 音频播放 / 258

9.1.2 音频录制 / 264

9.1.3 音频管理 / 265

9.1.4 音效处理 / 267

9.2 视频处理 / 269

9.2.1 视频播放 / 270

9.2.2 视频录制 / 273

9.3 Camera 服务 / 274

9.4 TTS 的实现 / 276

第 10 章 Android 跨语言调用详解 / 279

10.1 C 语言与汇编语言的相互调用 / 279

10.2 C++ 与 C 语言的相互调用 / 280

10.3 Java 对 C/C++ 的调用 / 282

10.4 C/C++ 对 Java 的调用 / 287

第 11 章 Android 安全框架解析 / 292

11.1 Java 混淆器 / 293

11.2 接入权限 / 294

11.2.1 创建接入权限 / 294

11.2.2 应用权限 / 294

11.2.3 权限验证 / 296

11.2.4 接入服务 / 297

11.2.5 框架层接入限制 / 298

11.3 数字证书 / 298

第 12 章 Android 的调试、测试与性能优化 / 300

- 12.1 Android 调试 / 300
 - 12.1.1 Logcat 日志调试 / 300
 - 12.1.2 dmtracedump 跟踪 / 300
 - 12.1.3 Dev Tools 调试 / 301
 - 12.1.4 屏幕截图分析 / 301
 - 12.1.5 内存调试 / 302
- 12.2 Android 布局优化 / 305
 - 12.2.1 Layoutopt 优化 / 305
 - 12.2.2 Hierarchyviewer 优化 / 306
- 12.3 Android 测试 / 307
 - 12.3.1 Monkey 压力测试 / 307
 - 12.3.2 JUnit 回归测试 / 308
 - 12.3.3 CTS 兼容性测试 / 312
 - 12.3.4 目标环境测试 / 313
- 12.4 Android 性能优化 / 314
 - 12.4.1 优化资源读取 / 314
 - 12.4.2 优化 APK 加载 / 315
 - 12.4.3 Dalvik 虚拟机 / 315
 - 12.4.4 TraceView 性能分析 / 315
 - 12.4.5 运行效率的优化 / 317

第 13 章 深入解析 Android 编译系统 / 318

- 13.1 源代码编译 / 318
 - 13.1.1 映像文件 / 318
 - 13.1.2 编译方法 / 319
 - 13.1.3 主要脚本 / 322
 - 13.1.4 环境变量 / 332
 - 13.1.5 目标环境 / 333
- 13.2 SDK 编译 / 333
 - 13.2.1 Linux 下的 SDK 编译 / 334
 - 13.2.2 Windows 下的 SDK 编译 / 334
 - 13.2.3 ADT 插件的编译 / 334

- 13.3 NDK 编译 / 335
 - 13.3.1 编译脚本 / 336
 - 13.3.2 配置脚本 / 337
 - 13.3.3 GDB 调试 / 340
 - 13.3.4 NativeActivity 实现 / 340
- 13.4 应用程序编译 / 342
 - 13.4.1 本地环境变量 / 342
 - 13.4.2 在 Eclipse 下编译 / 343
- 13.5 目标系统配置 / 343
 - 13.5.1 自定义模拟器配置 / 343
 - 13.5.2 目标板配置 / 348
 - 13.5.3 目标环境配置 / 350
- 13.6 文件系统配置 / 350
- 13.7 编译工具 / 350
- 13.8 fastboot 模式 / 352

第 14 章 Android 启动过程详解 / 353

- 14.1 系统的启动过程 / 353
 - 14.1.1 系统属性配置 / 354
 - 14.1.2 文件系统挂载 / 355
 - 14.1.3 守护进程启动 / 356
- 14.2 应用的启动过程 / 362
 - 14.2.1 应用的启动配置 / 362
 - 14.2.2 应用的启动过程 / 364

第 15 章 深入解析 Android 系统管理 / 369

- 15.1 内存管理 / 369
 - 15.1.1 对象引用 / 369
 - 15.1.2 垃圾回收策略 / 370
- 15.2 应用管理 / 373
 - 15.2.1 应用的配置 / 373
 - 15.2.2 应用的启动 / 374
 - 15.2.3 应用的销毁 / 375

- 15.3 电源管理 / 376
 - 15.3.1 电源管理框架 / 376
 - 15.3.2 应用层策略 / 379
- 15.4 下载管理 / 380
 - 15.4.1 Gingerbread 前的下载 / 380
 - 15.4.2 Gingerbread 后的下载 / 382
- 15.5 系统配置 / 383
 - 15.5.1 系统数据库 / 383
 - 15.5.2 系统属性 / 385
- 15.6 数据管理 / 386
 - 15.6.1 数据备份 / 386
 - 15.6.2 剪切板管理 / 387
- 15.7 设备管理 / 388
 - 15.7.1 管理工具 / 388
 - 15.7.2 传感器管理 / 395
 - 15.7.3 USB 管理 / 398
- 15.8 应用发布 / 399
 - 15.8.1 应用商店概述 / 399
 - 15.8.2 目标环境的适配 / 401
 - 15.8.3 发布应用 / 410

附录 A Android 开发环境搭建 / 413

附录 B 常用 Linux 命令 / 436

附录 C 编码规范 / 441

附录 D 学习资源 / 447

参考文献 / 451

深入解析 Android 核心组件和应用框架

本章将详细介绍 Android 的四大核心组件并深入剖析九大主要应用框架，旨在帮助读者更好地理解 Android 的系统框架。

1.1 核心组件详解

Android 有 4 个核心组件，分别是：提供界面显示的 Activity、提供后台计算的 Service、提供进程间通信的 Intent 和提供广播接收的 BroadcastReceiver。本节将详细介绍这些核心组件的使用方法和技巧，以及隐藏在背后的一些逻辑和原理。

1.1.1 Activity 组件

Activity 是 Android 的最重要组成部分之一，是实际与用户交互的组件。对于开发者而言，有几个非常重要的 Activity 子类需要特别注意，如 ListActivity、PreferenceActivity 和 TabActivity。Activity 的类图如图 1-1 所示，Activity 的生命周期的相关内容将会在 1.2.2 节中介绍。

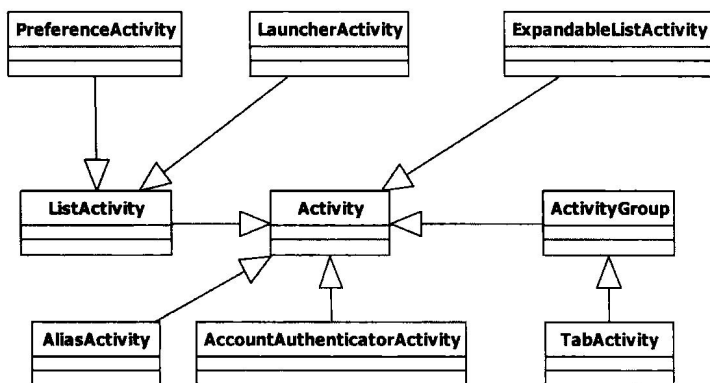


图 1-1 Activity 的类图

下面介绍图 1-1 所示类的用途和实现原理。

1. ListActivity

ListActivity 可以用来实现列表功能。列表可以说是终端应用最基本的控件，几乎每个应用中都会存在列表。但 ListActivity 也是最复杂的控件之一，其丰富的使用场景使很多初学者需要很长的时间才能熟练使用它。

在 Android 中，ListActivity 提供了对基本的单行、双行列表的封装，同时还支持用户自定义列表。自定义列表主要是基于 ListView 来实现的，为了简便起见，在此一并介绍。

就实现而言，Android 充分利用了设计模式中的适配器模式（Adaptor Pattern），无论数据多么复杂，总可以做到以不变应万变，极大地减小了实现的复杂度。

实现一个列表包括 3 步：选择或自定义列表项布局文件、实现适配器并加载数据、为 ListActivity 设置适配器。适配器的实现将在 3.10.1 节中重点介绍。

在整个列表中，列表项布局文件决定了列表的布局风格，很多初学者容易在实现自定义列表时犯布局兼容性的错误，即在选择布局控件时，不考虑所显示数据长度是基本固定的还是变化的，导致目标环境的适配出现问题。事实上，只要熟练掌握各布局控件的特点，就可轻松做出正确的选择。最基本、最常用的布局控件只有 RelativeLayout 和 LinearLayout 两个。

适配器的选择也非常简单，目前 Android 支持两种基本类型的适配器：基本适配器（BaseAdapter）和游标适配器（CursorAdapter）。基本适配器是最通用的适配器，游标适配器是用来适配数据库的数据流的，其他的系统级适配器都是继承自这两种适配器。

（1）系统列表项布局文件

在 Android 中，目前已经实现了基本的列表项布局文件，分别是基于单行布局的 simple_list_item_1 布局文件、基于简单双行布局的 simple_list_item_2 布局文件、基于单行单选布局的 simple_list_item_single_choice 布局文件、基于单行多选布局的 simple_list_item_multiple_choice 布局文件、类似树状图的 simple_expandable_list_item_1 和 simple_expandable_list_item_2 布局文件等。出于介绍方便的考虑，本书将这些基于系统已有风格的列表称为系统列表。

基于 simple_list_item_1 布局文件的一个实现如下：

```
// mStrings 为数组类型的数组
setListAdapter(new ArrayAdapter<String>(this,
    android.R.layout.simple_list_item_1, mStrings));
```

下面以 ArrayAdapter 为例来介绍适配器加载布局文件的方法，具体如代码清单 1-1 所示。