

# 基于Android环境的应用程序开发与研究

张 晶

(肇庆工商职业技术学院 广东 肇庆 526020)

**摘 要:** 主要介绍Google Android平台以, 讨论Android平台的系统结构, 应用程序的构成, 并介绍在Windows平台下面Android开发环境的搭建, 项目新建及运行的过程。

**关键词:** Android; 开发环境; 应用程序

**中图分类号:** TP311.1 **文献标识码:** A **文章编号:** 1671-7597 (2012) 0410080-02

随着移动网络的发展, 第三数字通信(3G)技术获得了非常广泛的使用, 它为使用移动终端来冲浪的用户带来了更快的数据传输速率。伴随微电子技术的发展, 移动终端不仅仅存在于通讯网络的移动电话, 还将成为互联网的终端, 比如平板电脑。因此, 在移动终端上运行的各式软件和为移动终端提供的服务具有非常大的发展。Google公司在2007年11月推出了基于移动终端的操作系统平台——Android。

Android是是基于移动终端的综合数据处理平台, 是真正的开放性平台, 它包括移动终端的操作系统以及中间件还有一些关键的应用。Android最显著的一个特点是: 体系结构的开放性, 正因如此, Android系统具有非常好的开发和调试环境, 同时支持各种扩展的用户体验。另外, Android系统对图形和多媒体的支持功能也具有独到之处。

Android平台的开放性以及其他的特点在某种程度上对技术能够祈祷一定的创新推进作用, 同时又有助于降低应用开发成本, 电信运营商也能非常方便地定制特色化的产品, 因此, 它具有很大的市场发展潜力。

## 1 Android软件平台的架构

Android平台从用户看到的表示层到底层的硬件, 总共分成了四个层次: Linux内核层(Linux Kernel)、Android运行库及函数(Android Runtime/Libraries)、应用框架层(Application Framework)和应用程序层(Application), 如图1所示:

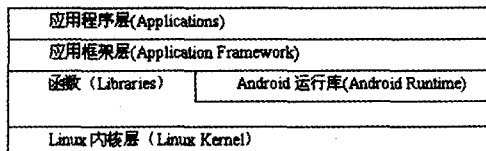


图1 Android软件平台的架构

1) Linux内核层(Linux Kernel): Android基于Linux 2.6的内核开发的, 提供核心系统服务, 例如: 安全、内存管理、进程管理、网络堆栈、驱动模型。Linux Kernel也作为硬件和软件之间的抽象层, 它隐藏具体硬件细节而为上层提供统一的服务。如果你只是做应用开发, 就不需要深入了解Linux Kernel层。

2) Android运行库及函数(Android Runtime/Libraries): Android包含一个核心库的集合, 提供大部分在Java编程语言核心类库中可用的功能。Android同时也包含一个C/C++库的集合, 供Android系统的各个组件使用。这些功能通过Android的应用程序框架(application framework)暴露给开发者。比如: 二维和三维的图形库, 位图和矢量字体的渲染, 应用程序需要是使用的数据库等等。

3) 应用框架层(Application Framework): Android开

发人员可以完全使用核心应用程序所使用的API, Android提供开发使用的API, 使得Android的开发者能够根据自己的创意编制出特色的应用程序, 比如利用设备移动设备的硬件优势、访问终端使用者的位置信息、程序进行后台服务、设置闹钟、向状态栏添加通知等等。应用框架层的体系结构旨在简化组件的重用, 任何的应用程序都可以使用组件的任何功能, 应用程序只需要服从框架执行的安全限制就可以了, Android的这一机制导致开发者可以将自己开发的程序替换系统的组件以实现更加强大的功能。

4) 应用程序层(Applications): Android提供一个核心应用程序集合, 包括电子邮件客户端、SMS程序、日历、地图、浏览器、联系人和其他设置。所有应用程序都是用Java编程语言写的, 当然我们也可以编写更加丰富的应用程序。

## 2 Android平台的组件及使用

Android应用程序主要是以各种组件的API为基础, API是Android软件的主要组成部分, Android主要由四个部分组成, 即Android四大组件, 分别是: Activity: 用于表现功能。Service: 运行在后台的Activity, 用于数据的处理。Intent: 用于传输数据。内容提供商(Content Provider): 支持在多个应用中存储和读取数据, 相当于数据库。

1) Activity: 在Android程序中, Activity是所有程序的门面, 所有程序的流程都运行在Activity之中, Activity可以算是开发者遇到的最频繁, 也是Android当中最基本的模块之一。Activity是由Android系统进行维护的, 它也有自己的生命周期, 即它的一个产生、运行、销毁的一个周期。对于Activity, 最关键的地方在于Activity的生命周期, 对Activity生命周期的掌握可以决定程序的基本流程, 其次就是Activity的各个状态的保存和恢复, 我们一般使用这2个方法来完成onSaveInstanceState和onRestoreInstanceState, 还有就是在不同的Activity之间进行跳转和Activity之间的数据传输。

2) Service: Service的作用是为Android应用程序提供适当的服务, 他存在在android系统中的某一种组件, 期作用和Activity差不了多少, 但是Service不能直接运行因为他没有自己的界面。所以在级别上来看, 他和Activity是一样的, 但是他必须依赖其他的界面来运行, 只有自己的生命周期, 但是他可以很长时间运行。

3) Intent: 抽象描述了想要执行的某种操作, 它与startActivity一起使用可以开启新的activity, 也可以开启后台服务。它是不同的Activity之间的桥梁, 是一种行为的数据结构, 主要包括action和data两部分。

4) Content Provider: Android提供的第三方应用数据的访问方案。实现应用程序的数据存取及不同应用程序间的数

