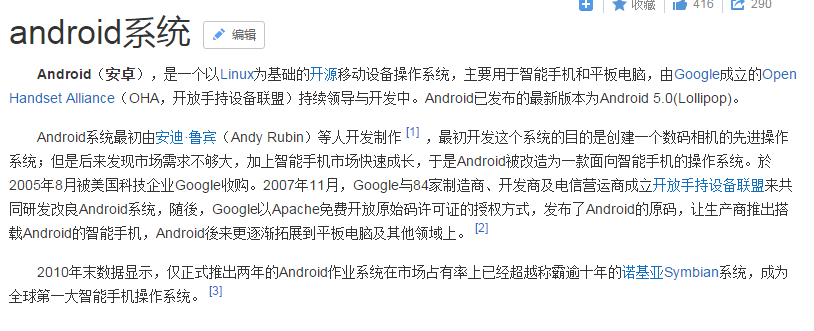
android系统



android的开发不单单只是android手机应用的开发



首先从最上面看：apk

application 分为两种，一种时我们系统自带的应用程序，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

另外一种就是互联网公司以及其他公司开发的自定义的应用程序，开发上层的界面应用程序也就是我们最熟悉的可以直接在手机看到的应用程序，他的开发语言使用的时JAVA语言进行的开发，大部分开发人员使用的开发工具：集成ecplise adt-bundle另外一款就是比较新的

android studio两款工具进行开发。

第二层开发时 application framework 框架层，他也是使用java语言进行的开发，他的开发基本都是一些手机厂商定制自己的一些功能与组件

第三层开发就是C库的开发主要的作用就是对 在这里厂商比较多的开发就是书写一些自己的算法库，还有就

在3层与4层之间许多厂商为了避免自己的程序遵循开源协议自是视频，音频的优化等功能，比如openGL 2D 3D引擎库等。定义的一层叫做HAL层 简称hardware abstract layer 这一层时驱动层的抽象，定制一些自己写的驱动对上层提供接口的功能。

Android运行环境：

Android包含一个核心库的集合，提供大部分在[Java编程语言](http://baike.baidu.com/view/53201.htm)核心类库中可用的功能。每一个Android应用程序是Dalvik[虚拟机](http://baike.baidu.com/view/1132.htm)中的实例，运行在他们自己的进程中。[Dalvik](http://baike.baidu.com/view/1551869.htm)虚拟机设计成，在一个设备可以高效地运行多个虚拟机。Dalvik虚拟机可执行文件格式是.dex，dex格式是专为Dalvik设计的一种压缩格式，适合内存和处理器速度有限的系统。 大多数虚拟机包括JVM都是基于栈的，而Dalvik虚拟机则是基于寄存器的。两种架构各有优劣，一般而言，基于栈的机器需要更多指令，而基于寄存器的机器指令更大。dx 是一套工具，可以将 Java .class 转换成 .dex 格式。一个dex文件通常会有多个.class。由于dex有时必须进行最佳化，会使文件大小增加1-4倍，以ODEX结尾。 Dalvik[虚拟机](http://baike.baidu.com/view/1132.htm)依赖于Linux 内核提供基本功能，如线程和底层[内存管理](http://baike.baidu.com/view/4541016.htm)。

**Linux Kernel**

Android基于Linux 2.6提供核心系统服务，例如：安全、[内存管理](http://baike.baidu.com/view/4541016.htm)、[进程管理](http://baike.baidu.com/view/364947.htm)、网络[堆栈](http://baike.baidu.com/view/93201.htm)、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。 如果你学过计算机网络知道OSI/RM，就会知道分层的好处就是使用下层提供的服务而为上层提供统一的服务，屏蔽本层及以下层的差异，当本层及以下层发生了变化不会影响到上层。也就是说**各层各尽其职，各层提供固定的SAP（**[Service](http://baike.baidu.com/view/3862339.htm)Access Point），专业点可以说是**高内聚、低耦合。**如果你只是做应用开发，就不需要深入了解Linux Kernel层。

android的编译与生成应用程序的流程：

