华清远见 FARSIGHT 嵌入式培训专家

3G学院 华清远见旗下品牌

Android上层应用开发

蒲洪涛

版权

- 华清远见嵌入式培训中心版权所有；
- 未经华清远见明确许可，不能为任何目的以任何形式复制或传播此文档的任何部分；
- 本文档包含的信息如有更改，恕不另行通知；
- 保留所有权利。

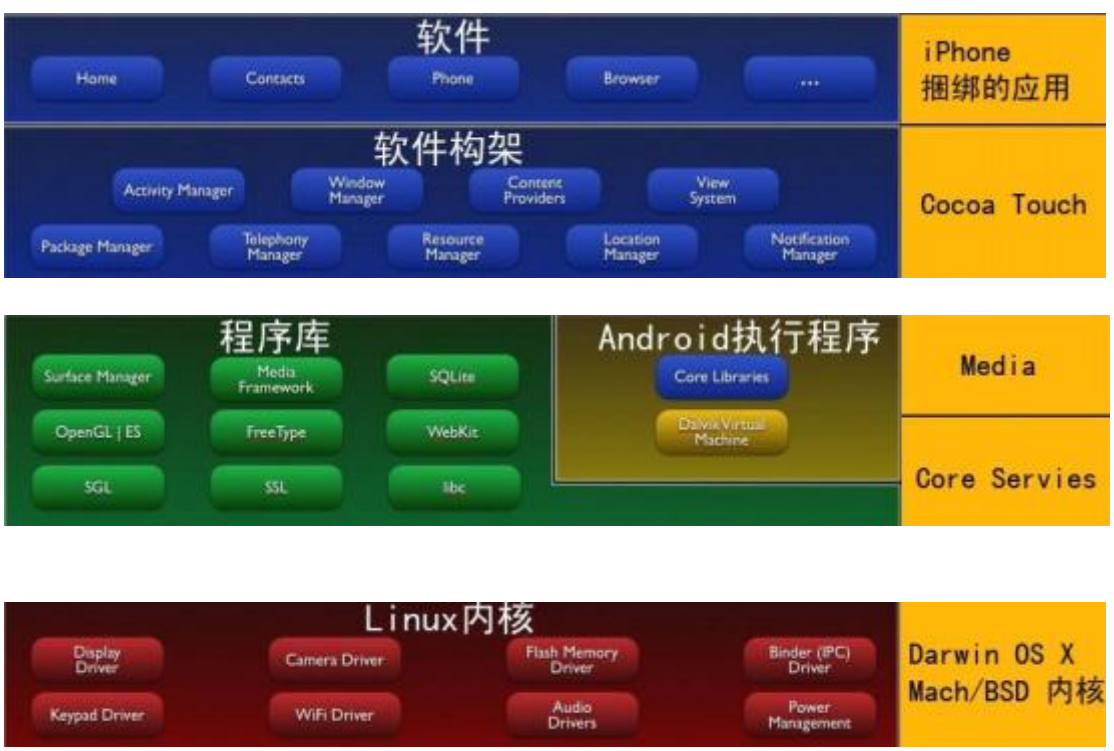
内容提纲

- 1）Android 简介
- 2）Android 开发技术
- 3）Android 实例
- 4）Android 学习资源

1)Android是什么？ —软件平台

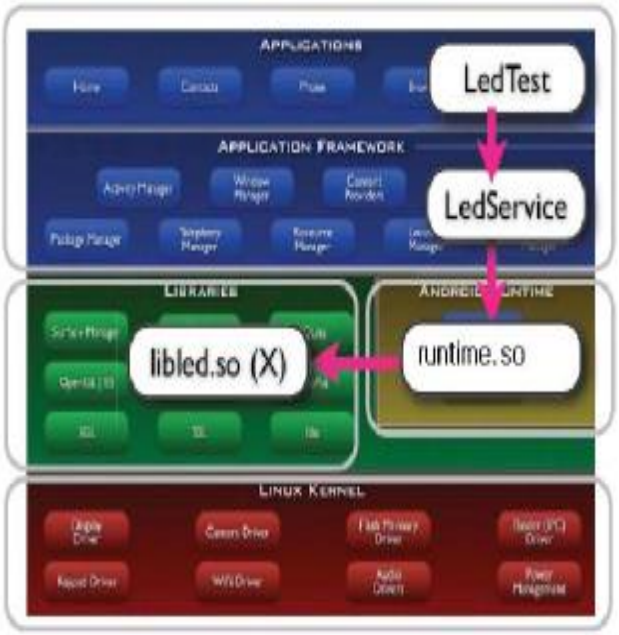
- Android 是包括一个操作系统、中间件和关键应用的移动设备的一个软件堆(软件平台)。
- 高度支持java开发
- 优化的Dalvik虚拟机
- 开源软件的大量应用
- 开源linux内核

1)对比其它的软件平台—Apple VS Android



1)Android软件平台—分层的好处

- 分层的好处是让开发者仅需专注于学习如何利用Android SDK, 编写高级语言构建其应用程序,而无需考虑底层的运作与执行
- 分层的本质是什么?--分工。分工是劳动生产力上最大的改良,由于各司其职,每个人可以从事其**最擅长的**劳动,再加上单纯劳动所带来的劳动熟练度提升和减少了更换劳动时的损失,使得劳动**生产率**大幅提升。



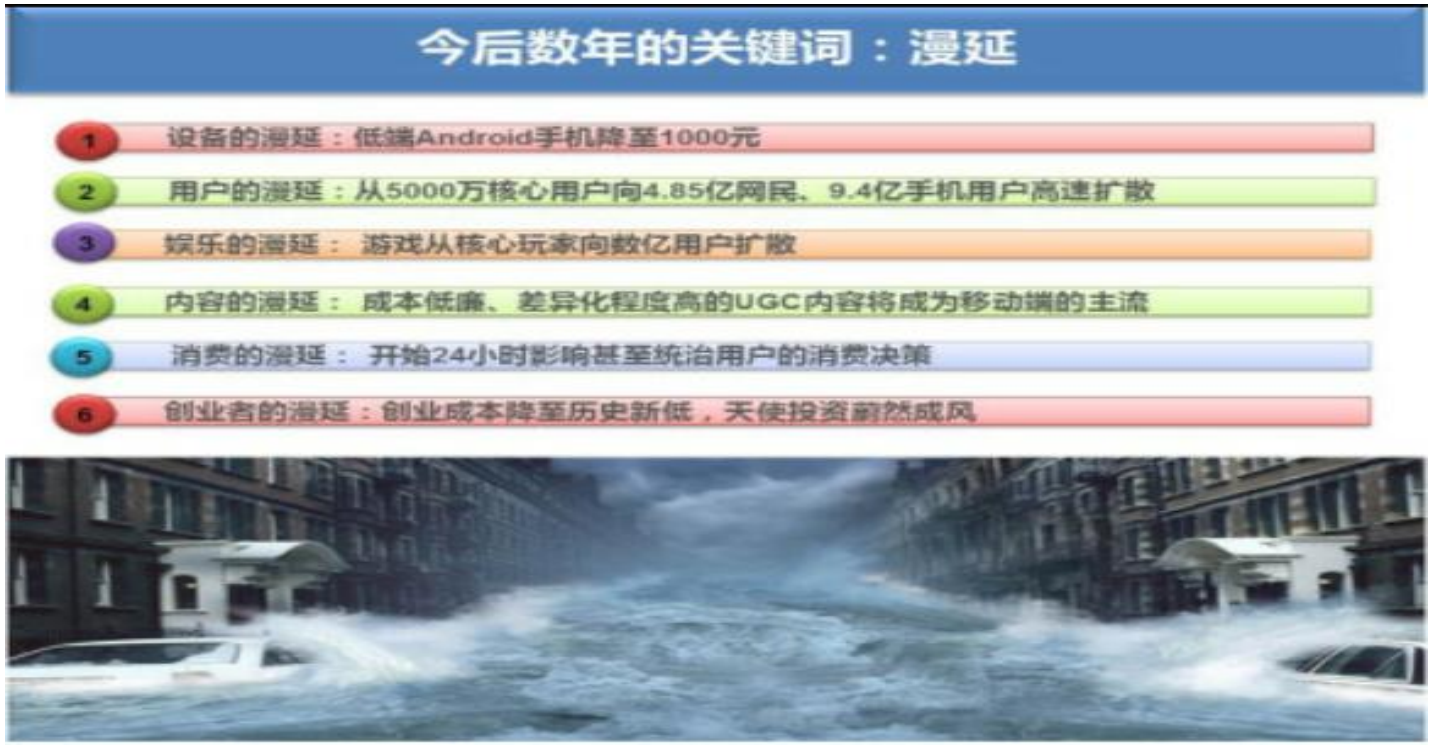
1)Android平台上能做什么?

Android 的双重开发模式:

- 各厂商在统一开放平台开发手机
- 第三方开发手机应用

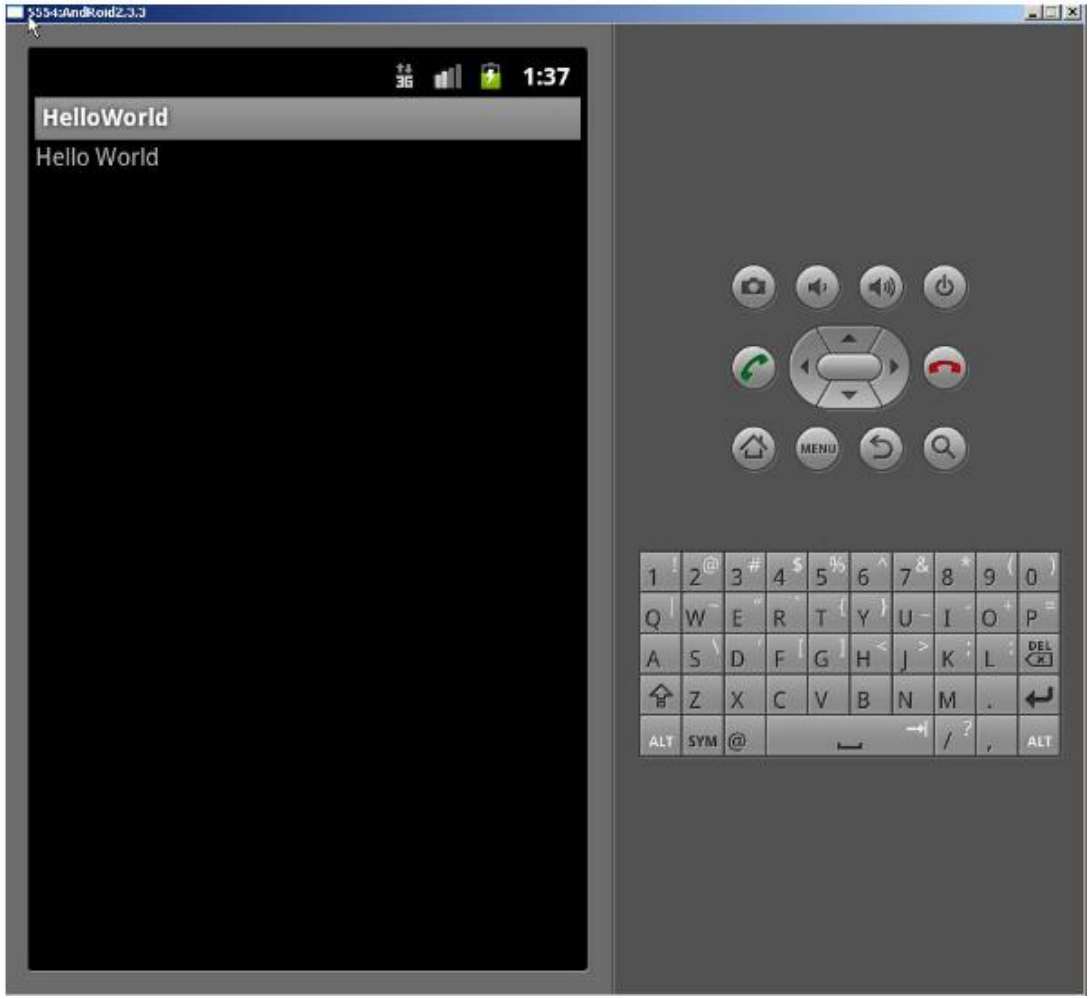
1)Android前景如何?

- 李开复在移动开发者大会的完整演讲PPT

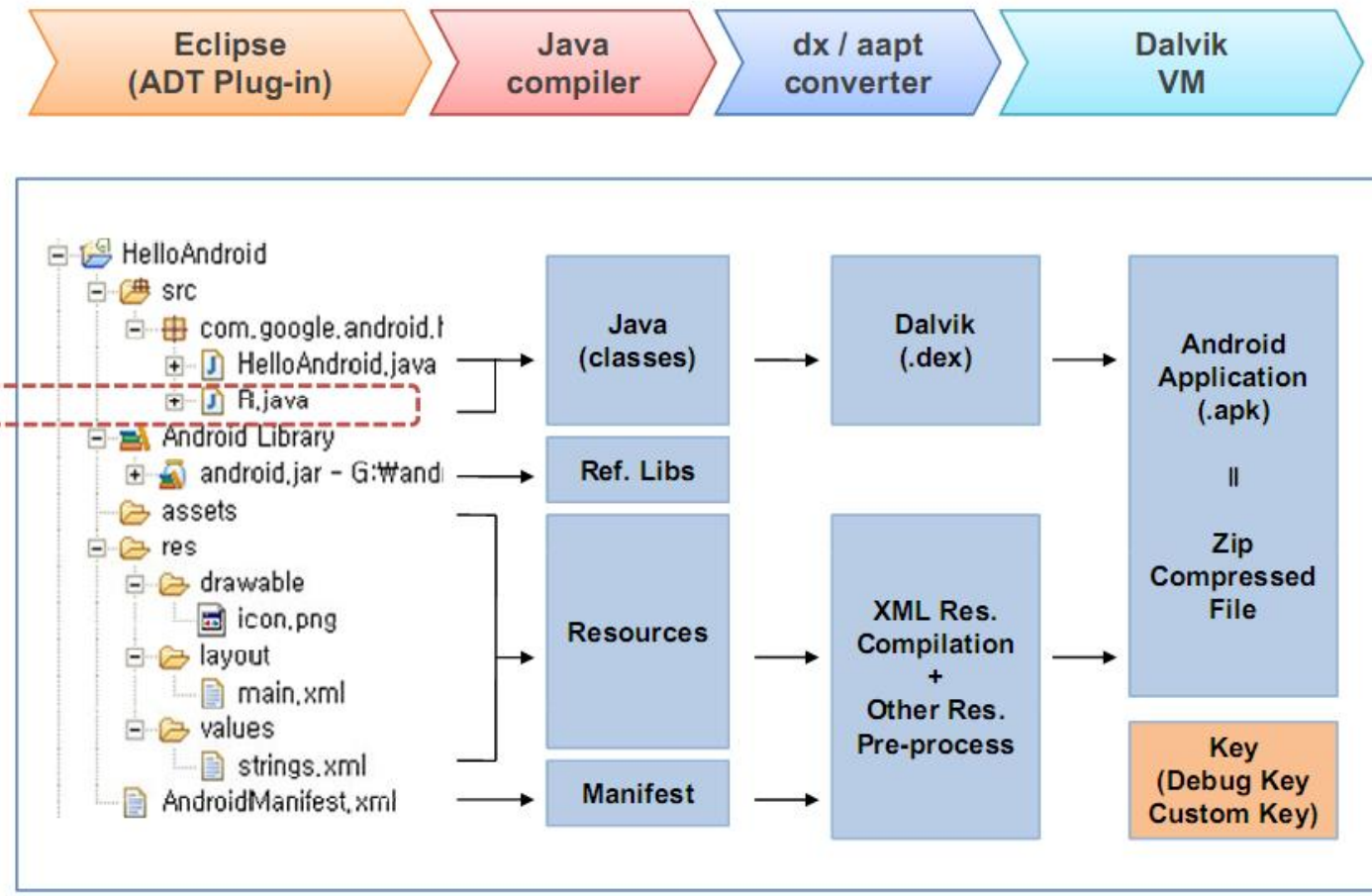


- 2011年11月17日外媒消息，激活的Android设备数量达到2亿台，每天激活的Android设备数量超过55万台

1)Hello world !



1)Hello world !



1)Android应用开发—环境搭建

- Java SE Development Kit (JDK)
  - jdk-6u21-windows-i586.exe
- Android SDK
  - android-sdk_r6-windows.zip
    - android.jar（编译后的Java应用，它包含了核心的SDK库和**API**）
    - DOCS 包括全部附带的Android文档
    - Samples 包括六个你可以通过Eclipse编译和测试的示例应用
    - Tools 包括全部开发、编译和调试工具，在开发Android应用的整个过程中你都需要他们
- Eclipse
  - eclipse-java-helios-win32.zip
- ADT Plugin for Eclipse
  - ADT-12.0.0.zip

1)Android上层应用开发—技能要求

- 编写 **à** 编译 **à** 运行 **à** 调试
- 编程语言 : Java
- 编程API : Android Framework
- 模型/编程元素—一切都是component
  - 用component(组件/元素)来开发
- 编程工具 : IDE--Eclipse+ADT插件, Android SDK
- 编译工具 : dx
  - 可执行文件: dex **à** APK
- 调试工具: ADB, DDMS, Logcat

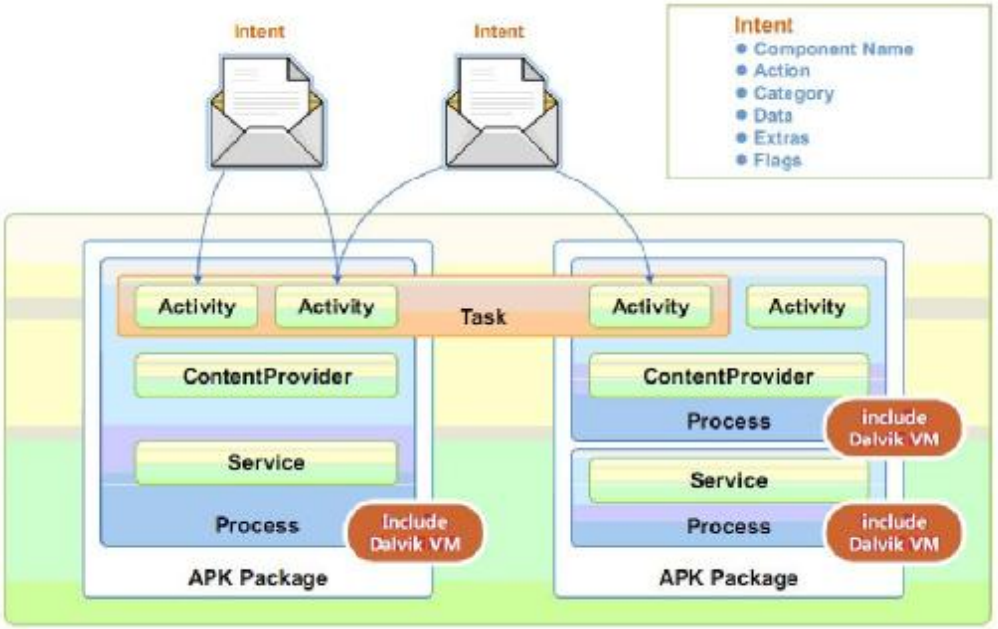
1)Android底层系统开发—技能要求

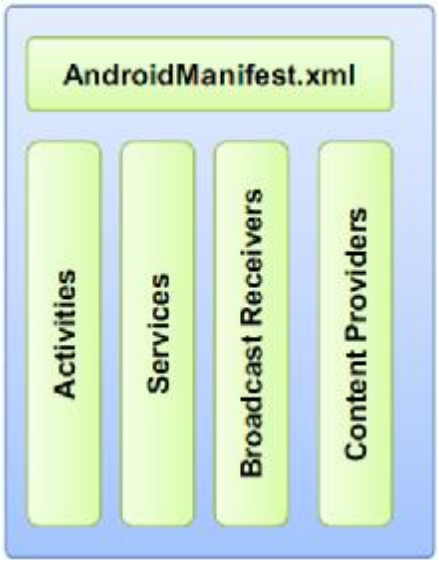
- 编写 **à** 编译 **à** 运行 **à** 调试
- 编程语言 : C/C++
- 编程API :
  - Linux用户空间编程: 库函数,libc,系统调用.
  - Linux内核空间编程: Kernel API
- 模型/编程元素—一切都是文件.
- 编程工具 : SourceInsight
- 编译工具： 交叉编译器.
- 可执行文件 : so , a, .o, ko
- 调试工具: gdb,Trace

2)Android软件堆/软件平台—应用的组件化开发方式

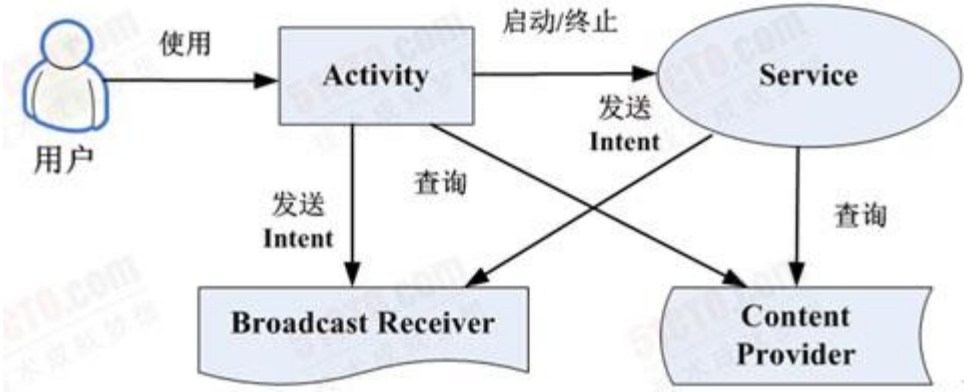
- 一个应用程序由以下组件构成：
- Acti ities 用户界面
- • Services：无用户界面的后台任务。可以提供调用接口
  - 例如：音乐后台播放 输入法 o 例如：音乐后台播放、输入法
- • Broadcast listeners：接收并响应广播通知
  - 例如：响应屏幕打开/关闭、时间变化、电池事件 o 例如　　响应屏幕打开/关闭、时间变化、电池事件
- • Content providers：通过预定义的 data provider 接口，提供
  - 特定数据的增、删、改、查。 特定数据的增 删 改 查
  - 例如：Contacts、Applications、SMS

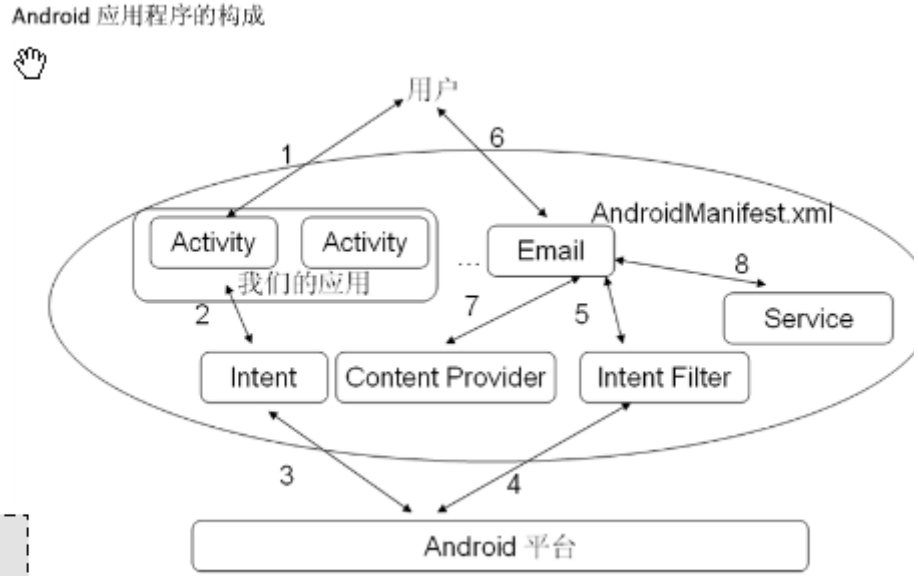
2)Android软件堆/软件平台—应用的组件化开发方式

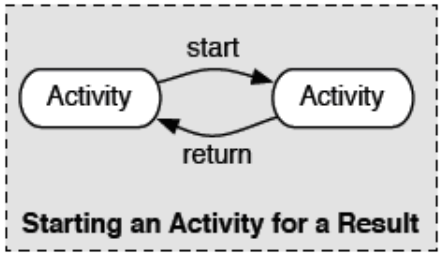


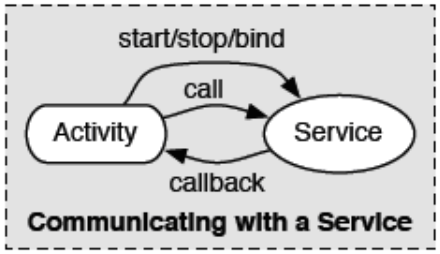


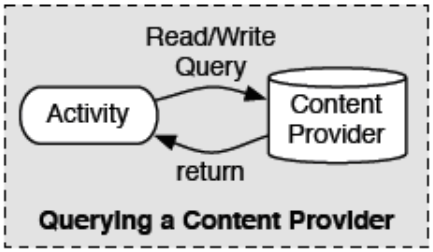
2)Android软件堆/软件平台—应用的组件化开发方式

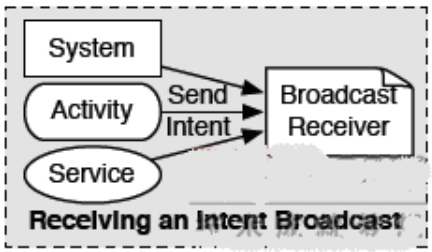






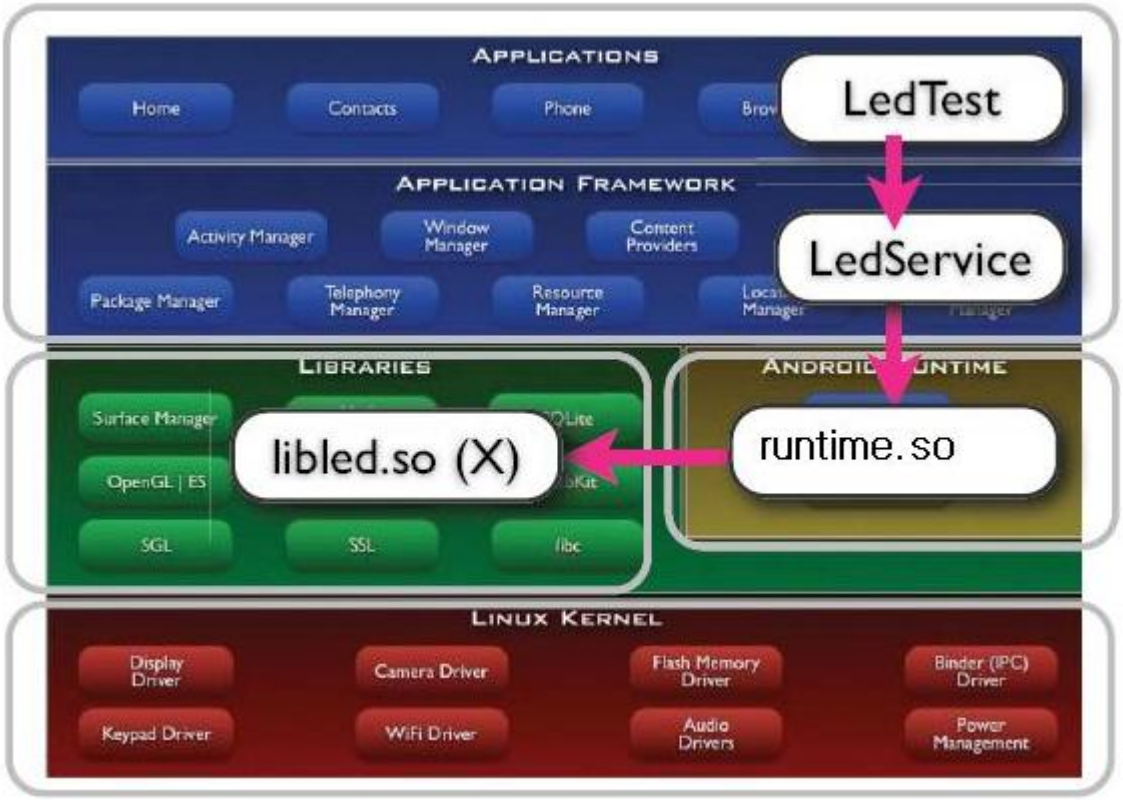






2)Android软件堆/软件平台—最终的层次调用图

Application è Application Framework **è** Runtime **è** libraries **è** Linux Kernel



2)Android软件堆/软件平台-- Framework

- 隐藏在每个应用后面的是一系列的服务和系统，其中包括：
- • 丰富而又可扩展的视图（ Views ），可以用来构建应用程序， 它包括列表（ lists ），网格（ grids ），文本框（ text boxes ），按钮（ buttons ）， 甚至可嵌入的 web 浏览器。
- • 内容提供器（ Content Providers ）使得应用程序可以访问另一个应用程序的数据（如联系人数据库）， 或者共享它们自己的数据
- • 资源管理器（ Resource Manager ）提供 非代码资源的访问，如本地字符串，图形，和布局文件（ la you files ）。
- • 通知管理器 （ Notification Manager ） 使得应用程序可以在状态栏中显示自定义的提示信息。
- • 活动管理器（ Activity Manager ） 用来管理应用程序生命周期并提供常用的导航回退功能。

2)Android软件堆/软件平台--程序库

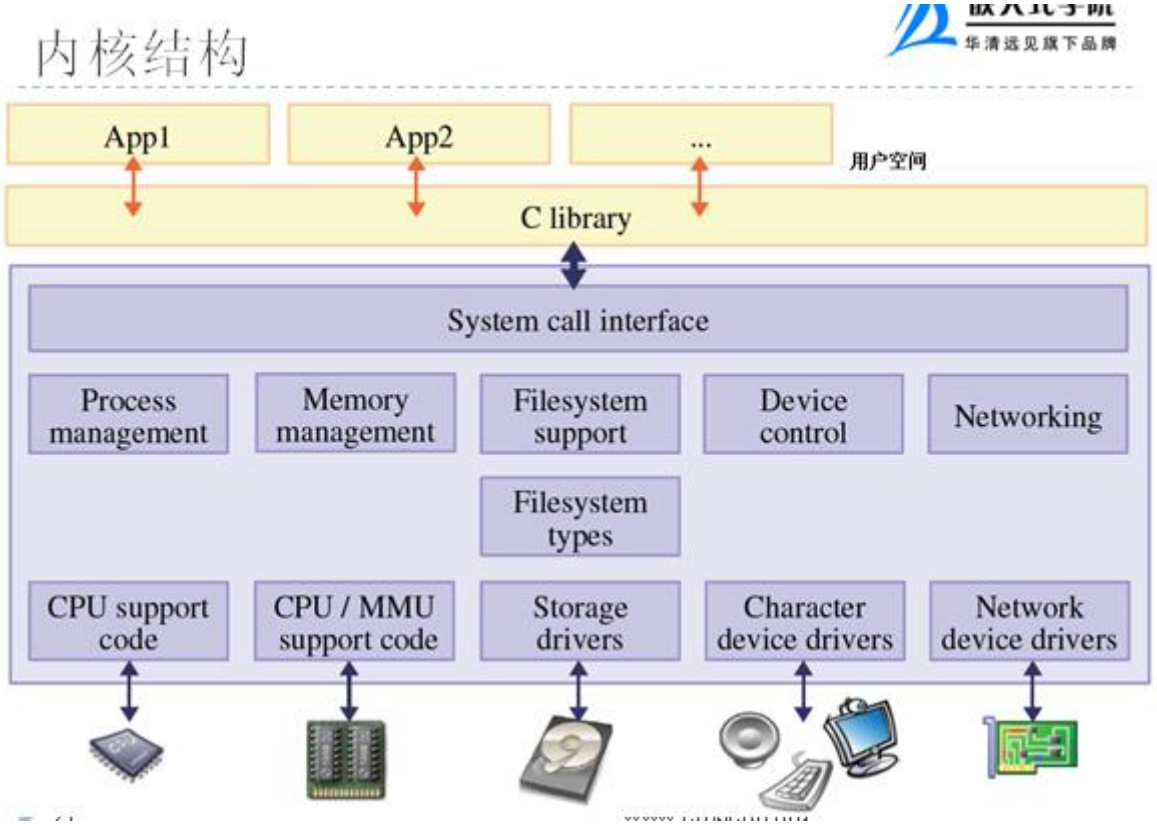
- Android 包含一些 C/C++ 库，这些库能被 Android 系统中不同的组件使用。它们通过 Android 应用程序框架为开发者提供服务。以下是一些核心库：
- • 系统 C 库 - 一个从 BSD 继承来的标准 C 系统函数库（ libc ）， 它是专门为基于 embedded linu x的设备定制的。
- • 媒体库 - 基于 PacketVideo OpenCORE ； 该库支持多种常用的音频、视频格式回放和录制， 同时支 持静态图像文件。编码格式包括 MPEG4, H.264, MP3, AAC, AMR, JPG, PNG
- • Surface Surface Surface Surface Manager Manager Manager Manager - 对显示子系统的管理，并且为多个应用程序提 供了 2D 和 3D 图层的无缝融合。
- • LibWebCore - 一个最新的 web 浏览器引擎用， 支持 Android 浏览器和一个可嵌入的 web 视图。
- • SGL - 底层的 2D 图形引擎
- • 3D libraries - 基于 OpenGL ES 1.0 APIs 实现； 该库可以使
- 如果可用）或者使用高度优化的 3D 软加速。
- • FreeType - 位图（ bitmap ）和矢量（ vector ）字体显示。
- • SQLite - 一个对于所有应用程序可用，功能强劲的轻型关

2)Android软件堆/软件平台—Android运行库

- Android 包括了一个核心库，该核心库提供了 JAVA 编程语言核心库的大多数功能。
- Dalvik 虚拟机执行（ .dex ）的 Dalvik 可执行文件，该格式 文件针对小内存使用做了 优化。同时虚拟机是基于寄存器的，所有的类都经由 JAVA 编译器编译，然后通过 SDK中 的 "dx" 工具转化成 .dex 格式由虚拟机执行。
- Dalvik 虚拟机依赖于 linux 内核的一些线程机制和底层内存管理机制。

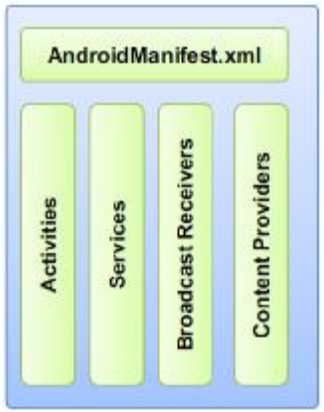
2)Android软件堆/软件平台—Linux 内核

- Android 的核心系统服务依赖于 Linux 2.6 内核，如安全性，内存管理，进程管理， 网络协议栈和驱动模 型 。
- Linux 内核也同时作为硬件和软件栈之间的抽象层。

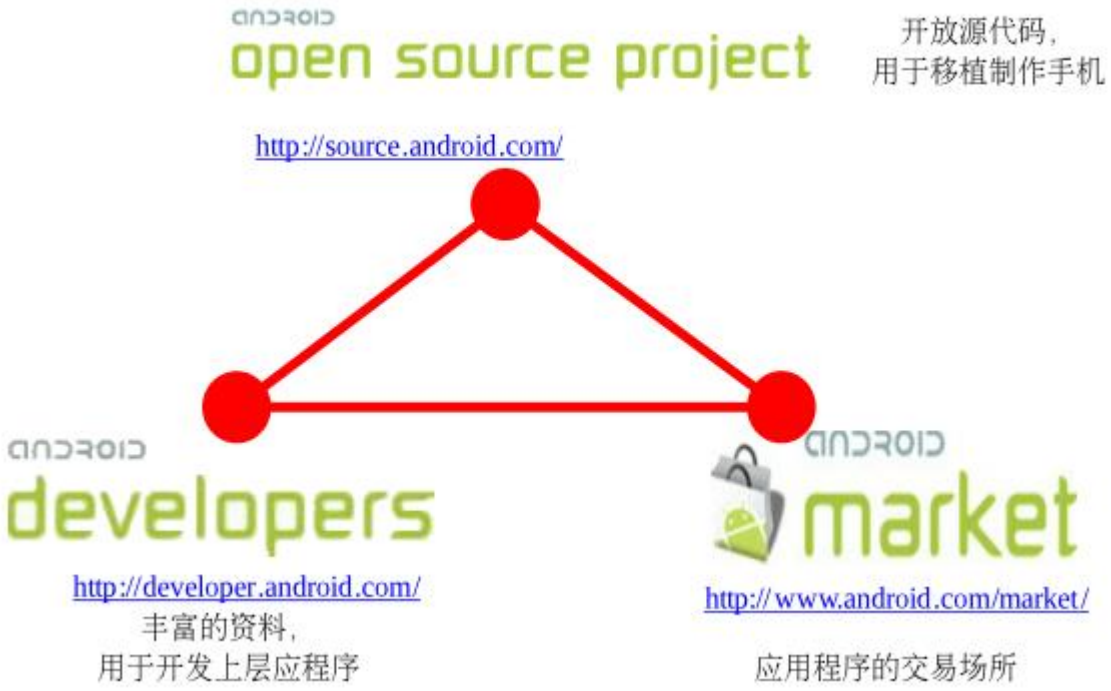


3)Android开发实例—短信发送&监听

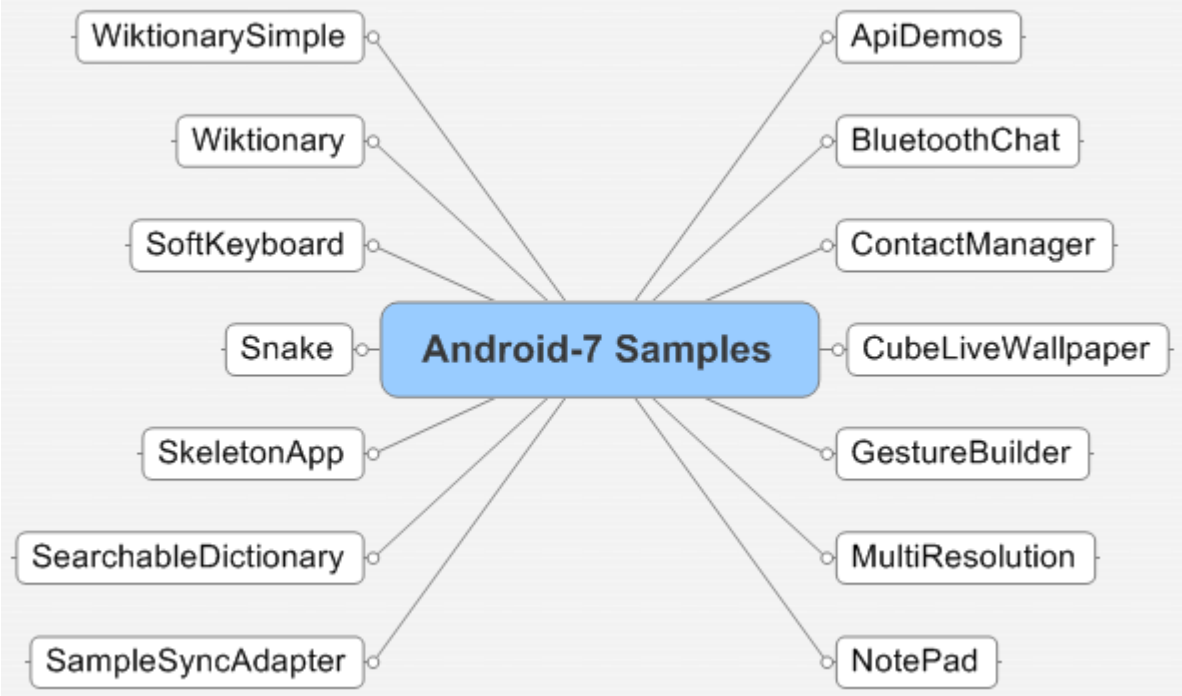
- 编写 **à** 编译 **à** 运行 **à** 调试
- 编写
  - 画一个界面— 告诉用户可以做哪些操作/功能?
    控件.
  - 响应界面事件— 用户希望做哪些事?
    响应的ONxx事件
  - 调用系统资源— 帮用户做事?
    Intent.
  - 反馈界面— 告诉用户办事结果!
    吐司/弹出框等等.
  - AndroidManifest.xml—Component组装清单
- 编译
- 运行



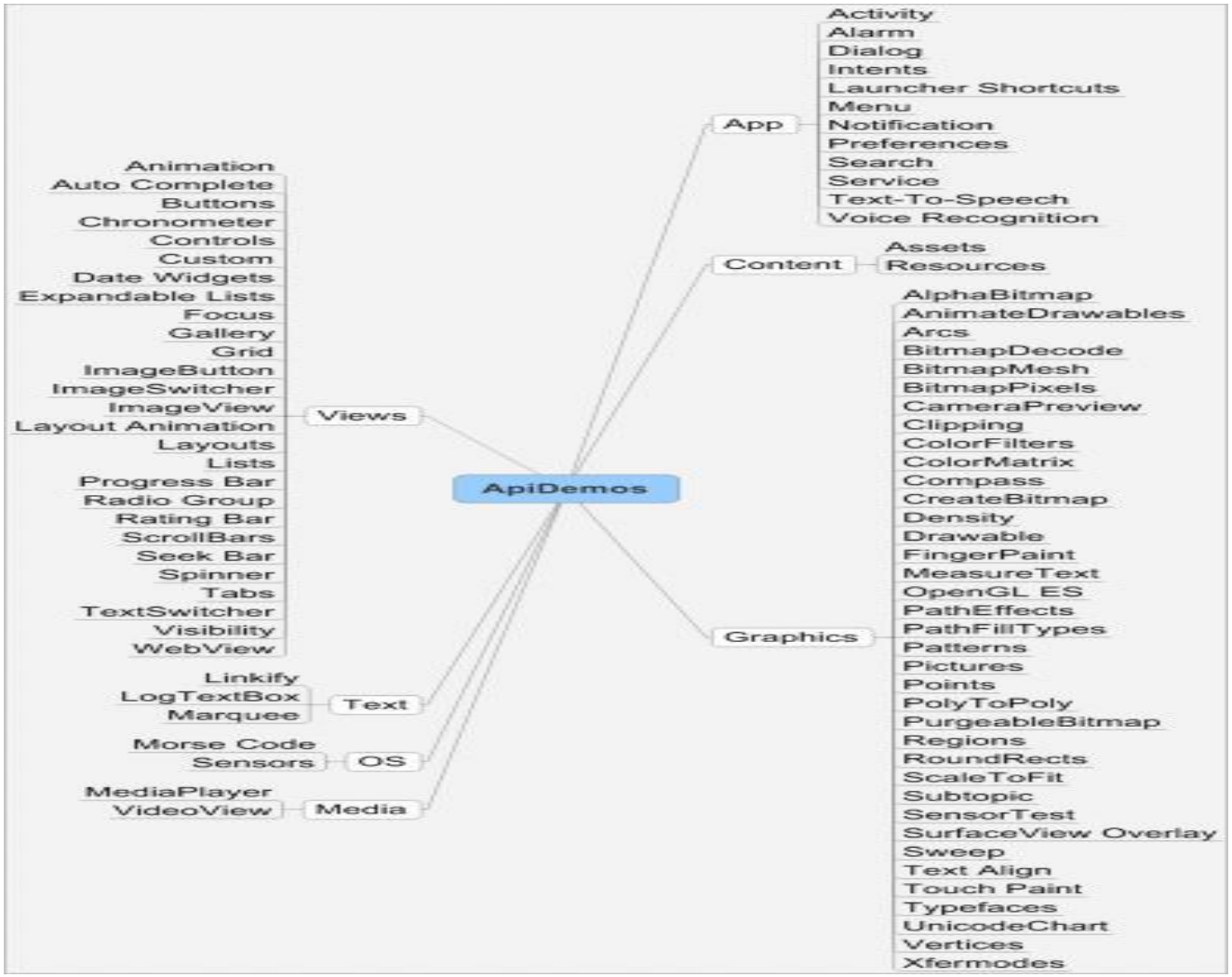
4)学习资源—官方资源


android
open source project
开放源代码，
用于移植制作手机
http://source.android.com/
android
developers
http://developer.android.com/
丰富的资料，
用于开发上层应程序
android
market
http://www.android.com/market/
应用程序的交易场所

4)学习资源--Samples

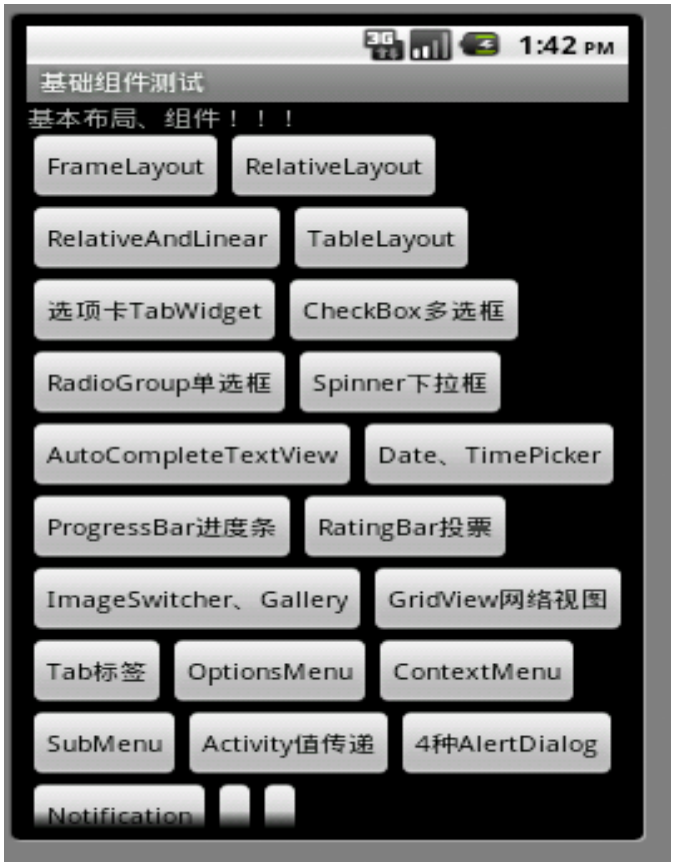


3)Android开发实例--ApiDemos



3)Android开发实例—Activity控件大集合

- 21个组件,5个布局



4)学习资源—华清学习资源

- （游戏开发）华清远见3G移动开发名家大讲堂全国开讲--再掀Android开发学习风暴--嵌入式学院（华清远见旗下品牌）

  http://www.embedu.org/news/TS111116.htm

- （视频讲座列表）名家讲堂 - 3G学院（华清远见旗下品牌）
  - http://www.3g-edu.org/lectures/index.htm
- （学习方法）Android开发入门之路（初学者必读） - 3G学院（华清远见旗下品牌）
  - http://www.3g-edu.org/news/3G-69.htm

4)学习建议

- 学习方针
  - 可以从应用开发作为切入点，逐渐熟悉其整个体系，并慢慢往其底层渗透，从而能做到软硬兼备
- 基础学习
  - 第一步，http://developer.android.com上的**Dev Guide**
  - 第二步，下载**eclipse, SDK,**搭建开发环境**.**
  - 第三步， 练习**SDK**里的**samples,**首先练习 **ApiDemos**这个例子
- 扩展学习**:**
  - 自己做一个感兴趣的应用程序**.**
  - 找一些开源项目来学习
  - 感兴趣的话可以研读**Android**源代码

Q&A

谢谢！