GETTING STARTED WITH EMBEDDED OPERATING SYSTEM

- These operating systems are designed to be compact, efficient, and reliable.

- E-OS are frequently also real-time operating systems, and the term RTOS is often used as a synonym for embedded operating system.

CONT....

- An important difference between most embedded operating systems and desktop operating systems is that the application, including the operating system, is usually statically linked together into a single executable image.

- Unlike a desktop operating system, the embedded operating system does not load and execute applications. This means that the system is only able to run a single application.

SOME EXAMPLES ARE...

- IOS (MAC OS)
- Pen Point OS
- Palm OS from Palm, Inc.
- Symbian OS
- Windows CE from Microsoft
- Android
- Metano GNU/Linux from Pynell - Embedded Systems.

WHAT IS ANDROID?

- Android is an OS that was developed by Google for use on mobile devices.
- Android is used in systems with little memory and processor.
- The Android SDK provides the tools and APIs necessary to begin developing applications on the Android platform using the Java programming language.
- Based on the Linux kernel

Why Android ?!

1 – Open Source

2 – Flash Player Support

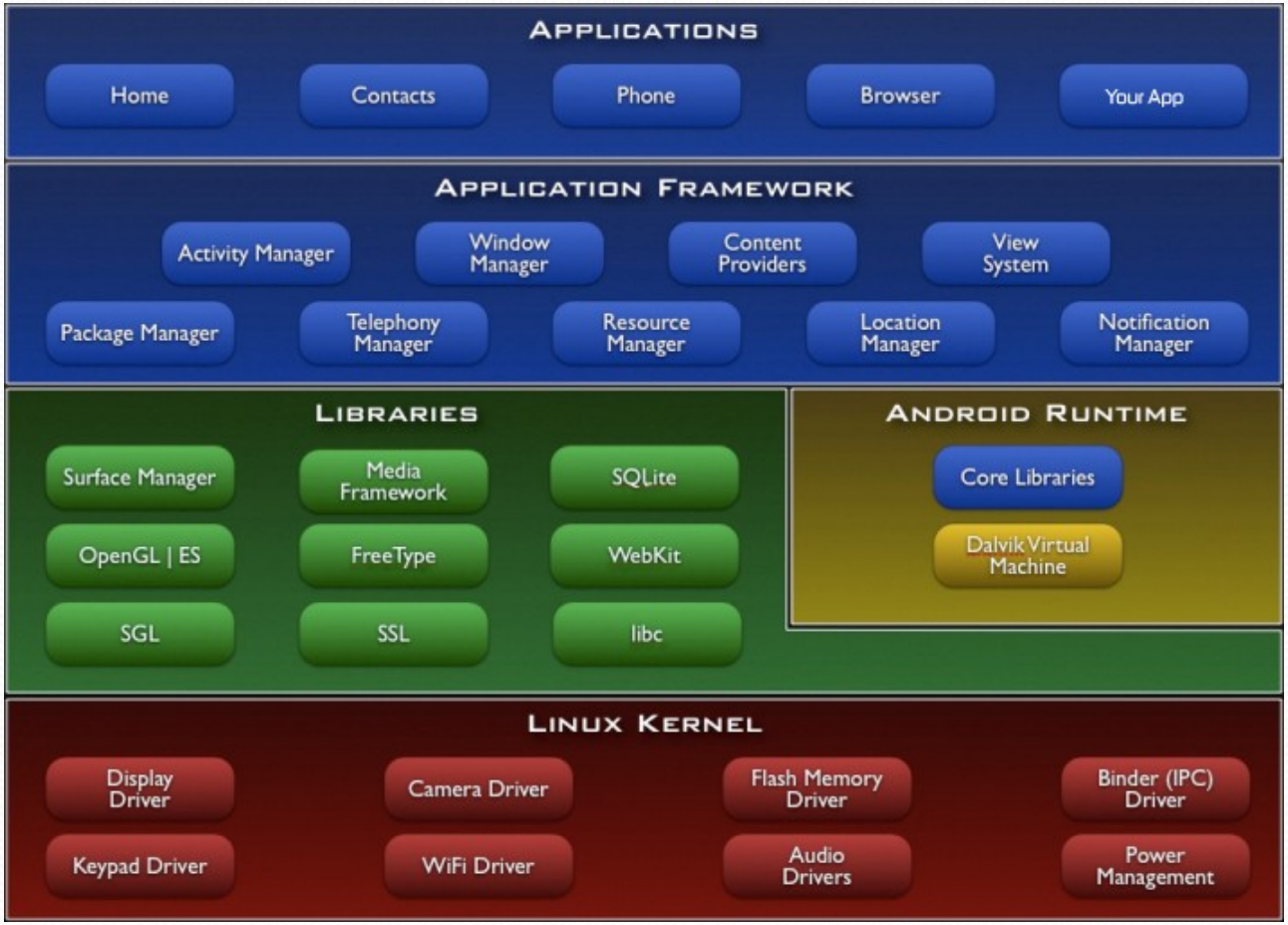
3 – Different Models

4 – Reasonable prices

5 – Google Apps

6 – Android Apps like (Astro file manager, Taskiller etc.)

Android Architecture



Applications

- Set of core applications including an email client, SMS program, calendar, maps, browser, contacts.
- All applications are written using the Java programming language.
- Built in and user apps.
- Can replace built in apps.

Application Framework

- Access location information, run background services, set alarms, add notifications to the status bar, and much, much more.
- Build an application, including lists, grids, text boxes, buttons, and even an embeddable web browser
- Providing access to non-code resources such as localized strings, graphics, and layout files.
- Activity manager – manages application life cycle.

Android Runtime

- Includes a set of core libraries of JAVA that provides most of the functionality
- Runs in its own process, with its own instance of the Dalvik Virtual Machine
- The Dalvik VM executes files in the Dalvik Executable (.dex)
  - Compact and efficient than class files
  - Limited memory and battery power

Libraries

- **System C library** - implementation of the C library (libc).
- **Media Libraries** - based on PacketVideo's Open CORE.
- **Surface Manager** - composites 2D and 3D graphic layers
- **LibWebCore** - a modern embeddable web view.
- **SGL** - the underlying 2D graphics engine.
- **3D libraries** - based on OpenGL ES 1.0 APIs; the libraries use hardware 3D acceleration.
- **FreeType** - bitmap and vector font rendering
- **SQLite** - a powerful and lightweight relational database engine .

Linux Kernel

- Acts as an abstraction layer between the hardware and the rest of the software stack.
- Relies on Linux version 2.6 for core system services such as security, memory management, process management, network stack, and driver model.
- It is open source

Dalvik Virtual Machine

- **Dalvik** is a process virtual machine (VM) in Google's Android operating system that executes applications written for Android. This makes Dalvik an integral part of the Android software stack (in Android versions 4.4 "KitKat" and earlier) that is typically used on mobile devices such as mobile phones and tablet computers, as well as more recently on devices such as smart TVs.
- Programs are commonly written in Java and compiled to bytecode for the Java virtual machine, which is then translated to Dalvik bytecode and stored in **.dex** (**Dalvik EXecutable**) and **.odex** (**Optimized Dalvik EXecutable**) files; related terms **odex** and **de-odex** are associated with respective bytecode conversions. The compact Dalvik Executable format is designed for systems that are constrained in terms of memory and processor speed.

Android Boot Sequence

- Boot Loader
- U-boot(Optional)
- Kernel
- Android

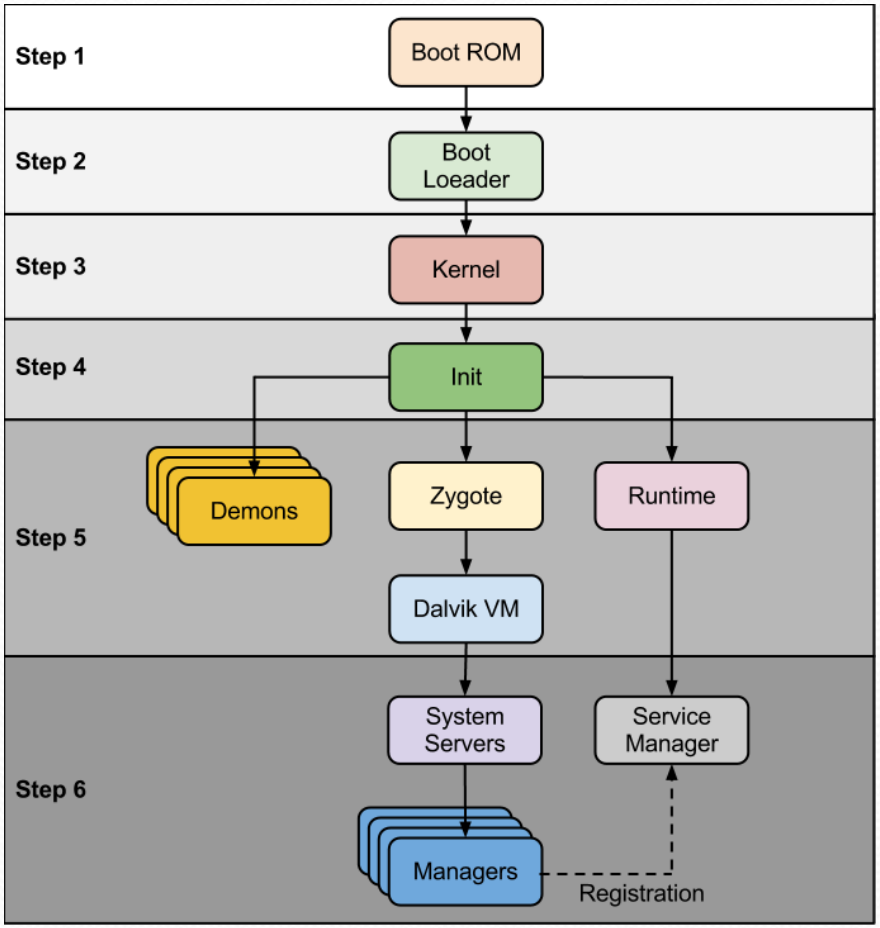
Linux Porting

- Adapting to the new environment
- Embedded Linux Porting: Adapting Linux to work on an Embedded Platform
- Architecture Porting
  - Processor Management
  - Memory Management
- Board Porting
  - I/O Management
  - Storage Management
  - Network Management

RTLinux

- Hard realtime RTOS microkernel
- Applications:
  - Control Robots,
  - data acquisition system systems,
  - manufacturing plants,
  - time sensitive machines & instruments.
  - Etc.

Android Boot Flow Diagram



Boot Loader

- When power is on the processor boots from a ROM area
- Determine Boot Media and load boot loader from media
- Boot Loader initialize DRAM and load Linux kernel
- It is generally dependent on the processor architecture and implementation.

U-Boot(Optional)

- It is first or second level boot loader
- It reads the Linux and ramdisk images from boot media and validates them.
- Passing arguments to kernel, fast boot modes etc.are applications of U-Boot

Kernel

- Heart of Android
- For process creation, IPC, Device drivers , File system management etc.
- The kernel can be either loaded as uncompressed image or as a compressed image.
- It mounts the root file system and starts the first application in user space

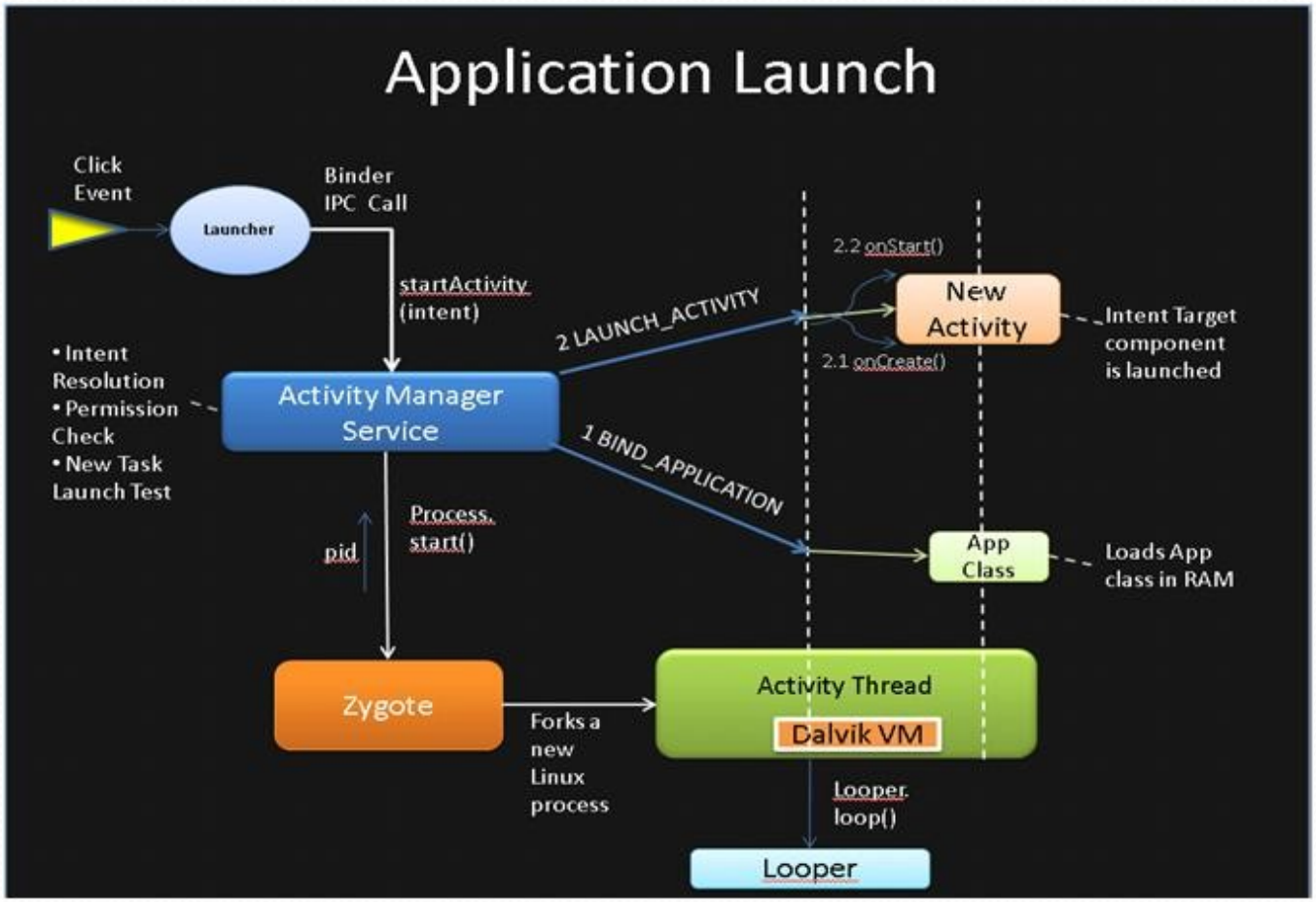
ANDROID PROCESS

Android typically operates wholly on user space. The android application are executed over a Virtual machine called Dalvik.

Android Process has following Sequence

- Init
- Zygote
- System Server
- Service Manager
- Other Daemons & processes
- Application

Android Boot Sequence



Android Kernel

- During kernel initialization:
  - Core kernel is initialized
    - Memory & I/O areas are initializes
    - Interrupts are started
    - Process table initialized
  - Drivers initialized
  - Kernel daemons (threads) are started
  - Root file system is mounted
  - The first user space process is started (/init)

Android init

- Specialized program,
- Used to initialize elements of the Android system
- Android uses its own initialization program.
- ‘init’ processes two files
  - init.rc
    - Generic initialization instructions
  - init.<machine_name>.rc
    - Machine specific initialization instructions

Zygote

- A daemon whose goal is to launch Apps
- Parent of all the app processes
- process starts when init executes C++ program from /system/bin/app_process
- service zygote /system/bin/app_process -Xzygote /system/bin --zygote --start-system-server
- Once launched, creats Dalvik VM & calls Zygote's main() method
- Once started,
  - it preloads all necessary Java classes and resources
  - Starts 'System Server' and
  - Opens a socket /dev/socket/zygote to listen for request for starting applications

System Server

- The **System Services** is a set of about 60-80 services that provide the user applications with the information and capabilities necessary to work.
- It is the ruler of the system
- This service starts all the other services

Activity Manager

- Zygote: Launch application
- Necessity of request
- Tap on App icon from home screen
- Home Screen- application listening for onClick()
- When this happens launcher contact the ActivityManager i.e. request a handler through **Binder** & call startActivity() method from Activity Manager
- After this, Activity Manager contacts Zygote to fork itself and start new VM to run the application via startViaZygote() method.

Launcher (Home)

- It is used to accept request from user.
- The home screen is an application and the only one listening for onClick().
- By tapping App icon user can send request.

CASE STUDY :-HOW ANDOID IS BETTER??

- When Android Compared with iPhone (Apple)
  - Browsing
  - Desktop
  - Connectivity
  - PC Connection
  - Multi-notification
  - Market
  - Google Integration
  - Open Source
  - Open to carriers

Android Software Updates

- Android 1.0-Alpha
- Android 1.1-Beta
- Android 1.5-Cupcake
- Android 1.6 – Donuts (CDMA)
- Android (2.0-2.1) – Éclair
- Android (2.2-2.2.3) – Froyo
- Android (2.3-2.3.7) - Gingerbread
- Android (3.0-3.2.6) – Honeycomb
- Android (4.0-4.0.4)-Ice Cream Sandwitch
- Android (4.1-4.3.1)-Jelly Bean
- Android (4.4-4.4.4)-KitKat
- Android(5.0)-Lollipop

Embedded Android Applications

- Calculator
- Tweeter Search App
- Slide Show App

CONCLUSION

In the small and rapidly growing world the need of mobiles and its applications are obvious. To meet the users embedded OS are playing the vital. To speak in a sentence about Android is much advanced and provides more value to the end users.

REFERENCES

[1] Karim Yaghmour, “Embedded Android”,O'R*eilly, ISBN: 978-1-449-30829-2, Pdf*

[2] www.android.com

[3] Lyla B. Das, “Embedded Systems: An Integrated Approach” Pearson,