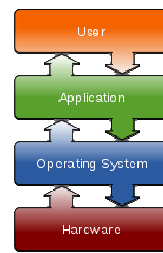# ENCE360 Operating Systems

Android

Brief History - Android

- 2003
  - Android, Inc. founded by Andy Rubin, Rich Miner, Nick Sears and Chris White
- 2005
  - Google acquires Android Inc. to start Android platform
- 2007
  - Open Handset Alliance announced
- 2008
  - SDK 1.0 released
  - Android released open source (Apache License)
- 2009
  - SDK 2.0 (Eclair)
- 2011
  - SDK 3.x (Honeycomb)
  - SDK 4.0 (Ice Cream Sandwich)
- 2015
  - SDK 5.0 (Lollipop)

Cupcake Android 1.5 **Donut** Android 1.6 **Eclair** Android 2.0/2.1 **Froyo** Android 2.2

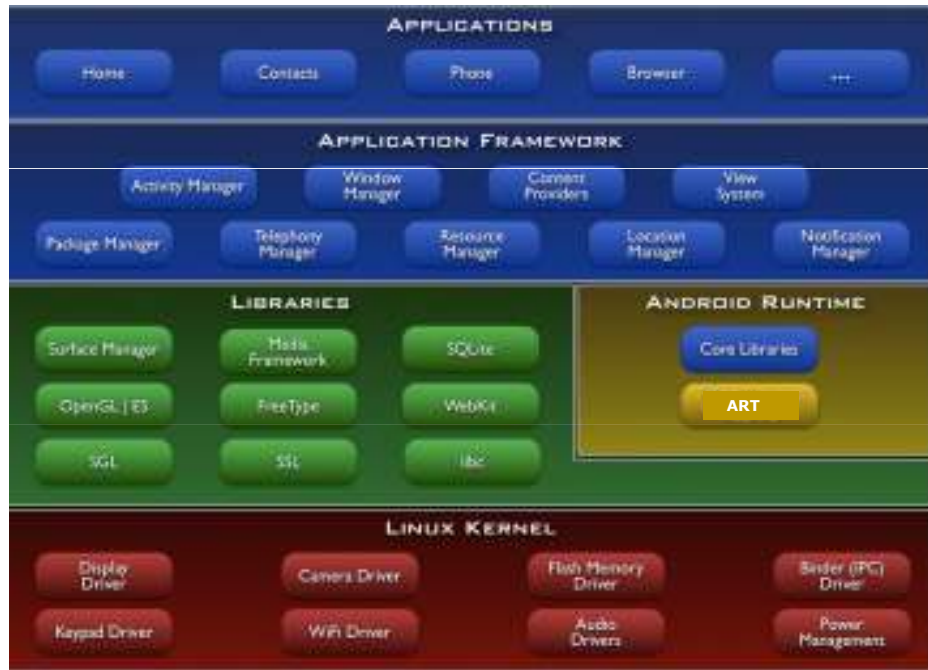**Honeycomb** Android 3.0-3.2 **Jelly Bean** Android 4.1.1

Brief History - Android

- Android is now the most popular mobile OS

- Since 2013, Android devices have been selling more than Windows, iOS, and Mac OS devices combined, with sales in 2012, 2013, 2014 and 2015 close to the installed base of all PCs

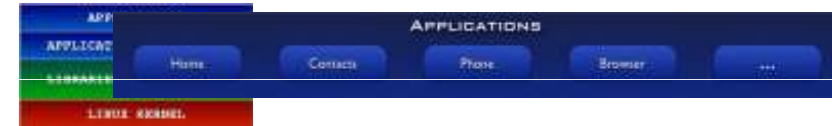- 71% of mobile developers develop for Android

---

What is Android?

- Android is a software stack for mobile devices that includes an operating system, middleware and key applications.

Architecture

Android Software Stack - Applications

- Android provides a set of core applications:
  - ✓ Email Client
  - ✓ SMS Program
  - ✓ Calendar
  - ✓ Maps
  - ✓ Browser
  - ✓ Contacts
  - ✓ Etc

- Most applications are written using the Java language.

Android Software Stack - Applications

- Android's standard C library, Bionic, was developed by Google specifically for Android, as a derivation of the BSD's standard C library code.
- Bionic itself has been designed with several major features specific to the Linux kernel.
- The main benefits of using Bionic instead of the GNU C Library (glibc) are its different licensing model, smaller runtime footprint, and optimization for low-frequency CPUs.

- Libraries written in C can also be used in a Java application by injection of a small Java shim and usage of the JNI.

Android S/W Stack – App Framework

- Enabling and simplifying the reuse of components
  - ✓ Developers have full access to the same framework APIs used by the core applications.
  - ✓ Users are allowed to replace components.
  - ✓ Android applications run in a sandbox, an isolated area of the system that does not have access to the rest of the system's resources (unless access permissions granted when installed)

Android S/W Stack – App Framework

□ Features

Feature	Role
View System	Used to build an application, including lists, grids, text boxes, buttons, and embedded web browser
Content Provider	Enabling applications to access data from other applications or to share their own data
Resource Manager	Providing access to non-code resources (localized strings, graphics, and layout files)
Notification Manager	Enabling all applications to display customer alerts in the status bar
Activity Manager	Managing the lifecycle of applications and providing a common navigation backstack

Android Software Stack - Libraries

□ Including a set of C/C++ libraries used by components of the Android system

□ Exposed to developers through the Android application framework

Android Software Stack – Runtime
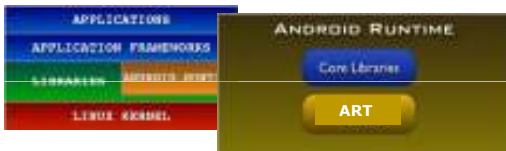
□ Core Libraries
   ✓ Providing most of the functionality available in the core libraries of the Java language
   ✓ APIs
      ➢ Data Structures
      ➢ Utilities
      ➢ File Access
      ➢ Network Access
      ➢ Graphics
      ➢ Etc

Android Software Stack – Runtime

□ **Android Runtime (ART) replaces Dalvik Virtual Machine**
   ✓ Providing environment on which most Android applications runs
      ➢ Most applications are written in Java and compiled on installation - **however, there is no Java Virtual Machine in the platform and Java byte code is not executed**
      ➢ Each such Android application runs in its own process, with its own instance of ART.
      ➢ ART is a specialized virtual machine designed specifically for Android and optimized for battery-powered mobile devices with limited memory and CPU.
      ➢ ART uses the Linux kernel to handle low-level functionality including security, threading, process and memory management
   ✓ Register-based virtual machine

Android Software Stack – Runtime

□ **Android Runtime** (**ART**)

  ✓ Unlike Dalvik, ART introduces the use of ahead-of-time (AOT) compilation by compiling entire applications into native machine code upon their installation.

  ➢ ART improves the overall execution efficiency and reduces power consumption, which results in improved battery autonomy on mobile devices (x86, ARM, and MIPS)

  ➢ ART brings faster execution of applications, improved memory allocation and garbage collection (GC) mechanisms, new applications debugging features, and more accurate high-level profiling of applications

  ✓ Relying on the Linux Kernel for:

  ➢ Threading
  ➢ Low-level memory management

Android Software Stack – Linux Kernel

- Android is a Linux distribution according to the Linux Foundation
- As of January 2014, current Android versions are built upon Linux kernel 3.4 or newer
- Relying on Linux Kernel for core system services
  ✓ Memory and Process Management
  ✓ Network Stack
  ✓ Driver Model
  ✓ Security