Overview

• History of Android Architecture
• Five Layers
  ▫ Linux Kernel
  ▫ Android Runtime
  ▫ Libraries
  ▫ Application Framework
  ▫ Applications
• Summary
History

- 2003 – Founded
  - No product for two years, funded by Andy Rubin
    - Planned the next generation of smartphones
    - **Open source** evolution of “Danger”
- 2005 – Purchased by Google
  - Sooner or G1?
- 2007 – Publically announced
- 2008 – Sold first phone
Previous Versions
Previous Versions

- Unnamed (1.0 + 1.1)
- Cupcake (1.5)
- Donut (1.6) – Quick Search Box
- Éclair (2.1) – High Density Displays, Traffic + Navigation
- Froyo (2.2) – Voice Control, Hotspot, Speed
- Gingerbread (2.3) – Simpler, Battery Life, More apps
- Honeycomb (3.0) – Flexible interface, tablets
- Ice Cream Sandwich (4.0) – Customization
- Jelly Bean (4.1) – Google Now, actionable notifications
- KitKat (4.4) – “Ok Google”, voice control variety
- Lollipop (5.0) – fluid tactile screens
- Marshmallow (6.0) – battery life, app permissions, UI
Linux Kernel

- 3.6 with ~115 patches
- Generic System Services
  - Permissions
  - Memory and Process management
  - File & Network I/O
  - Device Drivers
- Preemptive Multitasking
- Lean, efficient, and secure
- Open Source
Hardware Abstraction Layer (HAL)

- Software hooks between stack and hardware
- Hardware Specific
  - Allows Applications to be hardware ignorant
Android Runtime

- Dalvik Virtual Machine
- Core Java libraries
  - Specific to Android development
    - Apple: Swift (Objective C)
    - Windows: Visual C++ (C++), Changes with OS
  - Wrappers around C/C++ libraries
- ART (Android Runtime VM)
  - Replaced Dalvik in Lollipop (Android 5.0)
  - Advantages over Dalvik
    - AOT (Ahead of Time) Compilation
    - Improved Garbage Collection
Dalvik Virtual Machine

- Executes Android Applications
  - Each Application runs within its own VM
    - Each app is “sandboxed”
- Memory Management
- Multi-threading
Libraries

- C/C++
- Play and record audio and video
- Internet Security
- User interface building
- Graphics
- Database access
Library Examples

- **WebKit**
  - Web Browser Engine
- **OpenGL**
  - High Performance Graphics
  - Render 2D or 3D Graphic Content
- **libc**
  - Generic C library
- **SQLite**
  - Storage and sharing of application data
Library Examples Cont.

- **Surface Manager**
  - **Off-screen buffering**
    - Apps can’t directly draw into screen
    - Drawings go to off-screen buffer
    - Combined with other drawings
    - Reason behind window transparency

- **Media Framework**
  - Provides media codecs allowing recording and playback of different types of media
Application Framework

- Higher Level Services to Applications
- Environment in which applications are run and managed
- Package Manager
  - Keeps track of installed Applications
  - Apps can communicate with other Apps on device
- Window Manager
  - Manages main window that comprises Application
Application Framework Cont.

- **View System**
  - Provide Common User Interface Elements
    - Icons
    - Buttons
    - Text Entry
    - Etc.

- **Content Providers**
  - Databases that allow application to store and share structured info
Application Framework Cont.

• Location Manager
  ▫ Allows application to receive location and movement info generated by GPS

• Activity Manager
  ▫ Manages activity life cycle of applications

• Telephony Manager
  ▫ Manages all voice calls
Application Framework Cont.

- **Resource Manager**
  - Manage various types of resources used in applications
  - Allows access to non-code embedded resources
    - Strings
    - Color settings
    - UI Layout

- **Notifications Manager**
  - Allows applications to display alerts
Applications

- Hosts Android Applications
- Written in Java
  - Access to all Android APIs
- Executed in the VM (Dalvik or ART)
- Examples
  - SMS client app
  - Dialer
  - Web Browser
  - Contact manager
Conclusion

• **Designed** for mobile and flexibility
  ▫ Both in software and hardware
• 5 Layers
• Application Development
  ▫ **Simple**
    • Java
  ▫ **Access to all aspects of the Kernel**
    • Open Source
    • APIs