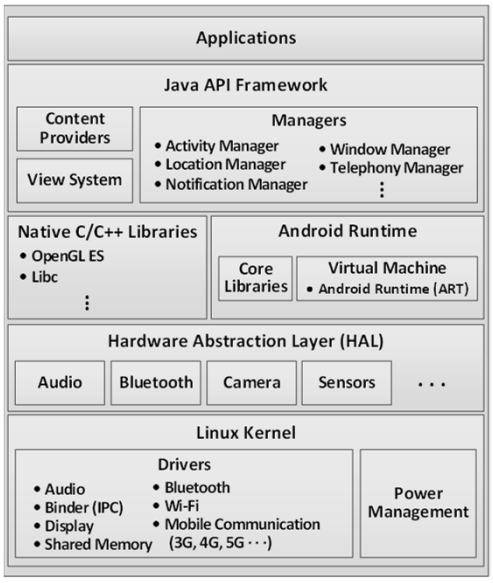
Smartphone OSs

Android Architecture

Android Architecture

❖ **Android OS**

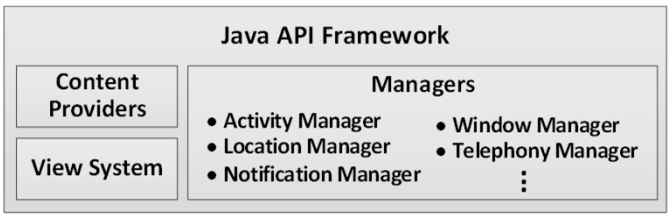
- Android is the most popular OS for smartphones
- Android includes
  - OS
  - Middleware
  - Key applications



Android Architecture

❖ JAVA API Framework

- All features of the Android OS are available through APIs written in Java
  - API: Application programming interface
- Java APIs are provided for developing and using apps



Android Architecture

❖ JAVA API Framework

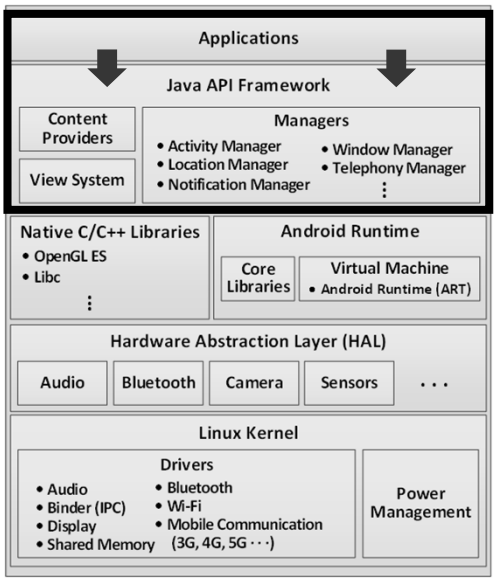
- Building blocks of the Java API framework
  - View System
    - Building UI of apps
    - Includes text boxes, buttons, lists, etc.
  - Content Provider
    - Enables apps to access data from other apps
      - Contacts, Gallery, etc.

Android Architecture

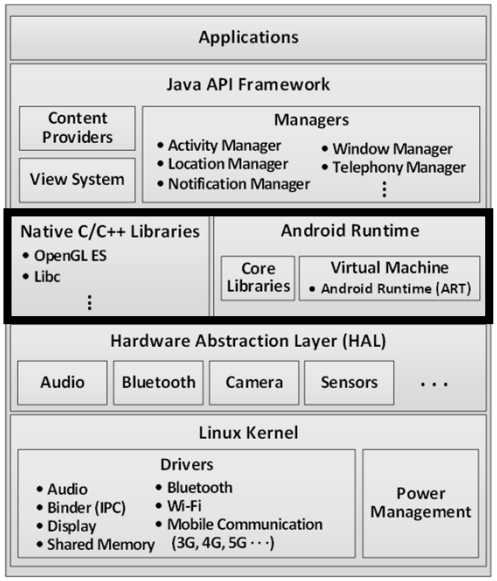
❖ JAVA API Framework

- Building blocks of Java API framework
  - Managers
    - Provide core system service and hardware system service
    - Core system service
      - Activity, Window, View, etc.
    - Hardware system service
      - Telephony, Location, Wi-Fi, USB, etc.

Android Architecture



Android Architecture



Android Architecture

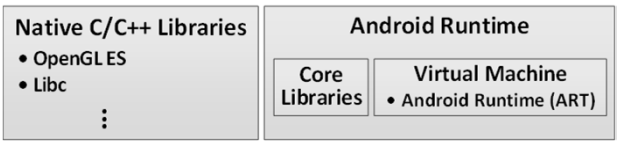
❖ Native C/C++ Libraries

- Supports programs written in C and C++ in the Android System (e.g., ART, HAL)
- Popular libraries
  - OpenGL ES
    - 2D and 3D graphic library with OpenGL
  - OpenMAX AL
    - Companion API to OpenSL ES for multimedia video & audio, supporting native multimedia handling
  - Libc
    - C runtime library

Android Architecture

❖ Android Runtime

- Running environment instance for Android applications
- VMs (Virtual Machines) are used for Java (JVM) apps that will run on different Android devices
  - Java Dalvik Virtual Machine (before API 21 Lollipop)
  - Android Runtime (ART, API 21 and beyond)



Android Lollipop

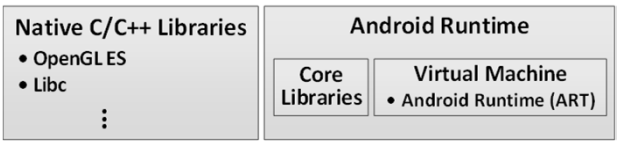
❖ Android 5.0~5.0.2 Lollipop (API level 21)

- Released in November 2014
- 64 bit CPU support
- Audio Input and Output through USB devices
- Memory of recently used applications
- Addition of 15 new languages

Android Architecture

❖ Android Runtime (ART, API 21 and beyond)

- Java Dalvik Virtual Machine
  - JIT (Just-In-Time) complier used
  - JIT compiles the code when the application starts
  - JIT uses partial section of the code in case of a request



Android Architecture

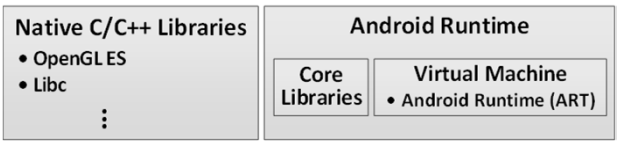
❖ Android Runtime (ART, API 21 and beyond)

- ART (Android Runtime) support
  - ART supports both the JIT (Just-In-Time) compiler and the AOT (Ahead-of-Time) compiler
  - Improved debugging environment
  - Improved GC (Garbage Collection)
    - For example, when GC compacts the heap, the display becomes choppy and the UI responsiveness is poor
    - ART provides improved automatic memory management to prevent this problem

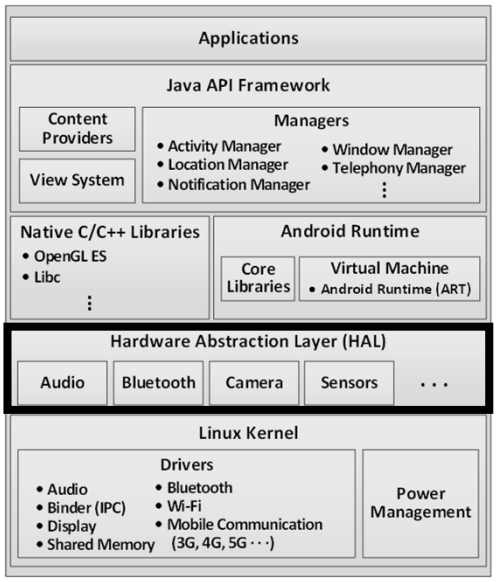
Android Architecture

❖ Android Runtime (ART, API 21 and beyond)

- AOT compiler
  - AOT (Ahead-of-Time) compiler compiles the code ahead of initial running when the application is installed
  - Improvements in app speed and power consumption performance are obtained
  - Requires a longer app installation time



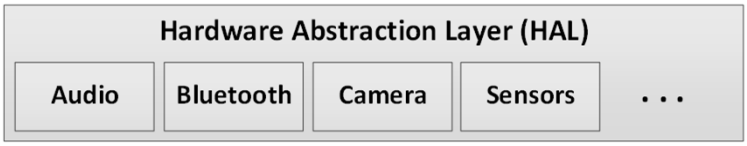
Android Architecture



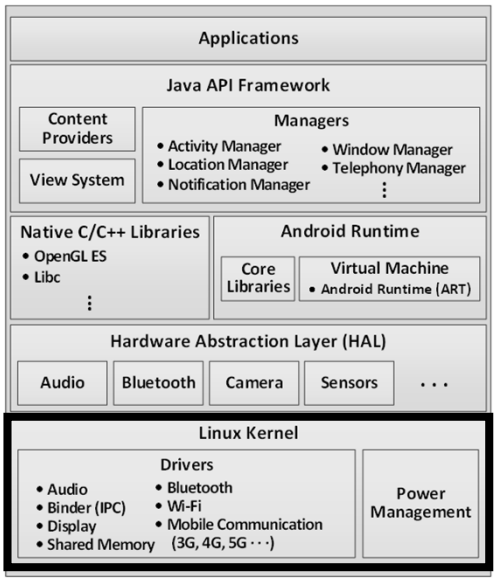
Android Architecture

❖ HAL (Hardware Abstraction Layer)

- Includes standard interfaces for higher-level Java API framework to use the device's hardware & processors
  - Consist of multiple library modules
- Customized to specific types of hardware components
  - Bluetooth module, camera module, etc.



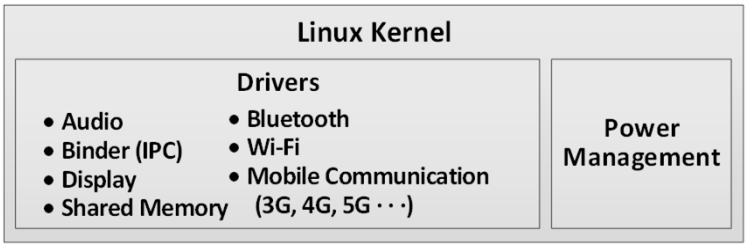
Android Architecture



Android Architecture

❖ Linux Kernel

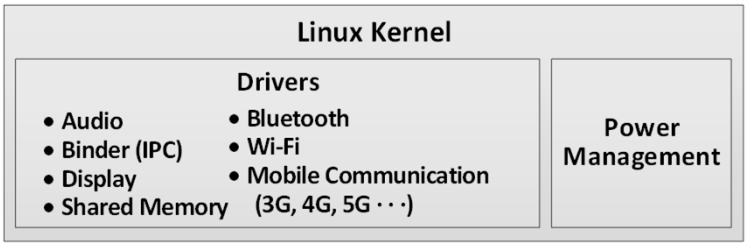
- Linux kernel is the processing foundation of the Android OS
- Executes low-level functionalities
  - Processing, Threading, Power management, etc.



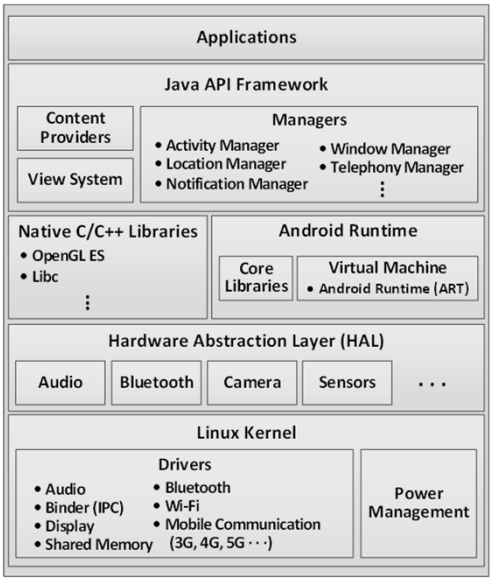
Android Architecture

❖ Linux Kernel

- Linux Kernel Components
  - Linux Kernel drivers
  - Android shared memory
  - Power management
  - Wi-Fi, Bluetooth, Mobile Com., etc.



Android Architecture



Smartphones OSs

References

References

- Techotopia, http://www.techotopia.com/index.php/IOS_6_Architecture_and_SDK_Frameworks
- Techotopia, http://www.techotopia.com/index.php/The_iPhone_OS_Architecture_and_Frameworks
- Exploring the iOS SDK, https://code.tutsplus.com/tutorials/exploring-the-ios-sdk--mobile-13959
- Android Platform Architecture, https://developer.android.com/guide/platform/index.html
- Comparint JIT and AOT Compilation, https://www.ibm.com/support/knowledgecenter/en/SSSTCZ_3.0.0/com.ibm.wrt.rtlinux.doc.30/realtime/aot.html
- DotNetTricks, http://www.dotnettricks.com/learn/xamarin/understanding-xamarin-ios-build-native-ios-app