Android OS Design

Hasitha Ch
IS201301009

Overview

This Report highlights the design details of Google's Android Operating System. Prominent features have been explained and work flow for every feature has been run through 5 main layers, namely the Application layer, the Application Framework Layer, the Libraries Layer, the Linux Kernel and the Hardware Layer.

Goals

1. To Understand the design details in the Operating System
2. To Emphasize on work flow for few prominent features in the Operating System

Introduction

Android is a mobile operating system (OS) , based on the Linux kernel and designed primarily for handheld mobile devices. As of 2015, Android has the largest installed user base of all mobile operating systems. The Operating system mainly uses versions 3.4 or 3.10 of the Linux kernel. On top of the Linux kernel, there are the middleware, libraries written in C, and applications running on an java based framework. Android used Dalvik as a process virtual machine. Programs written in the java applications are first compiled into java bytecode and then translated into Dalvik bytecode. This vm makes execution faster and also isolates applications by creating a new instance of the virtual machine for each application.

For a Mobile Operating System like Android, the functional requirements would be as follows,

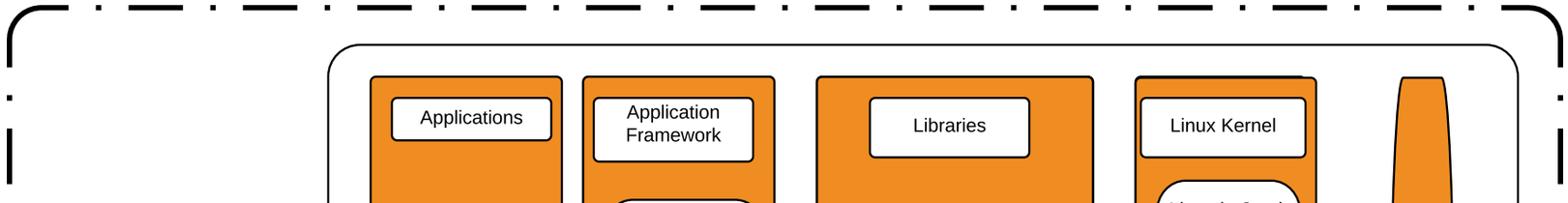
1. To be able to use lesser power
2. To support primary requirements of a mobile phone like calling, texting etc..
3. To be able to run multiple applications using lesser memory
4. To give the best performance with tight constraints like size, memory, power, etc..

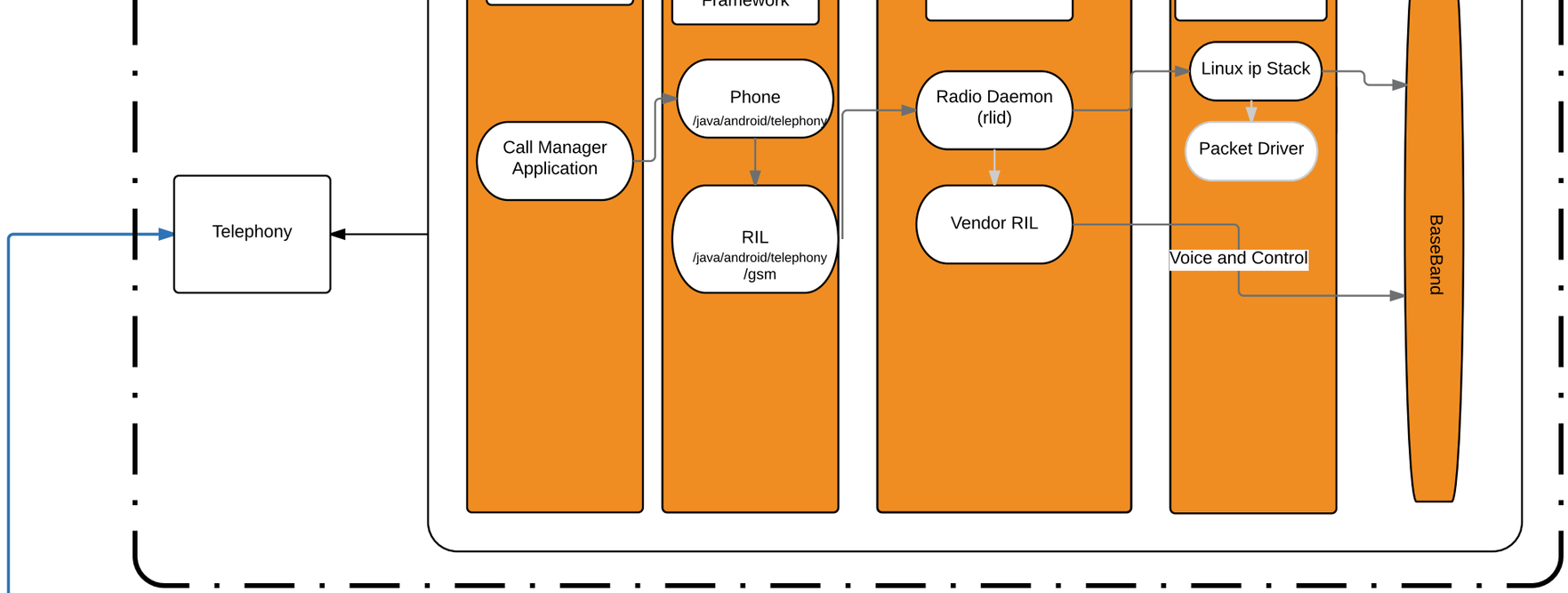
The non-functional requirements would be ,

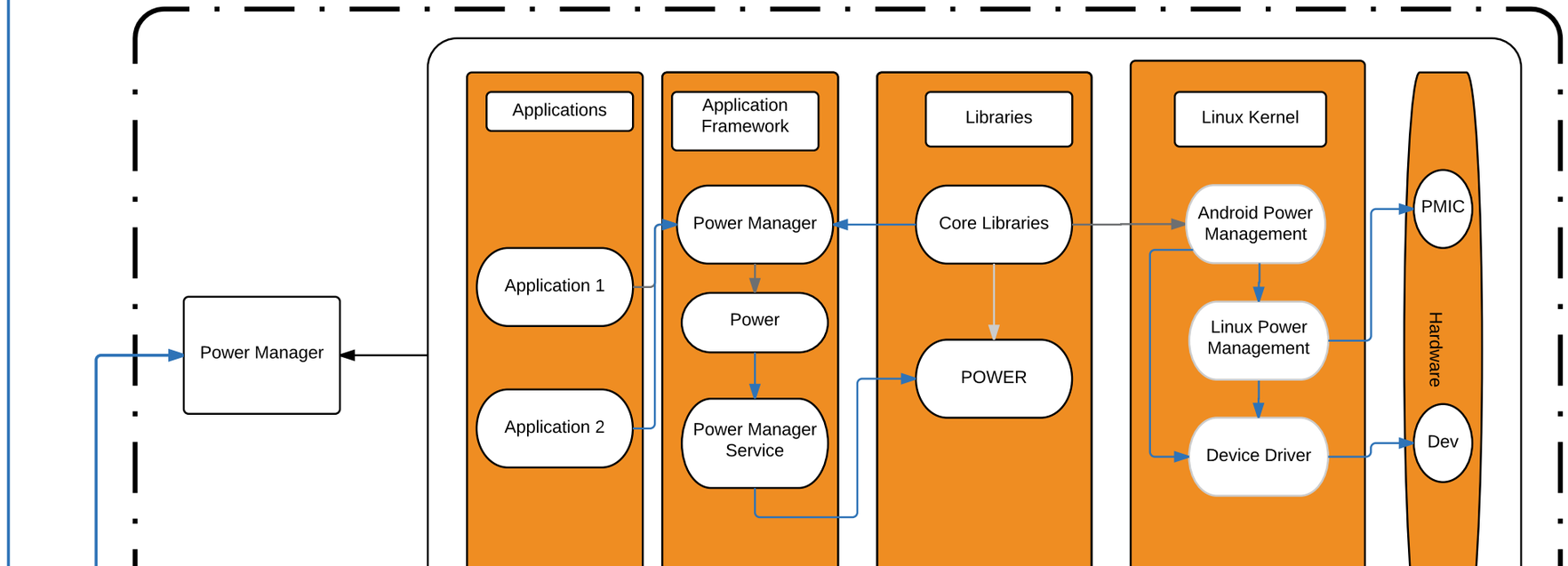
1. To be able to work on a wide range of devices.
2. To be able to make the best use of the limited memory and provide faster performance
3. To be able to run a wide range of applications
4. To function efficiently without facing any application crashes
5. To allow scope for developers to tamper with the source code but still be stable enough (open sourced)
6. To support high range of network bandwidth efficiently.

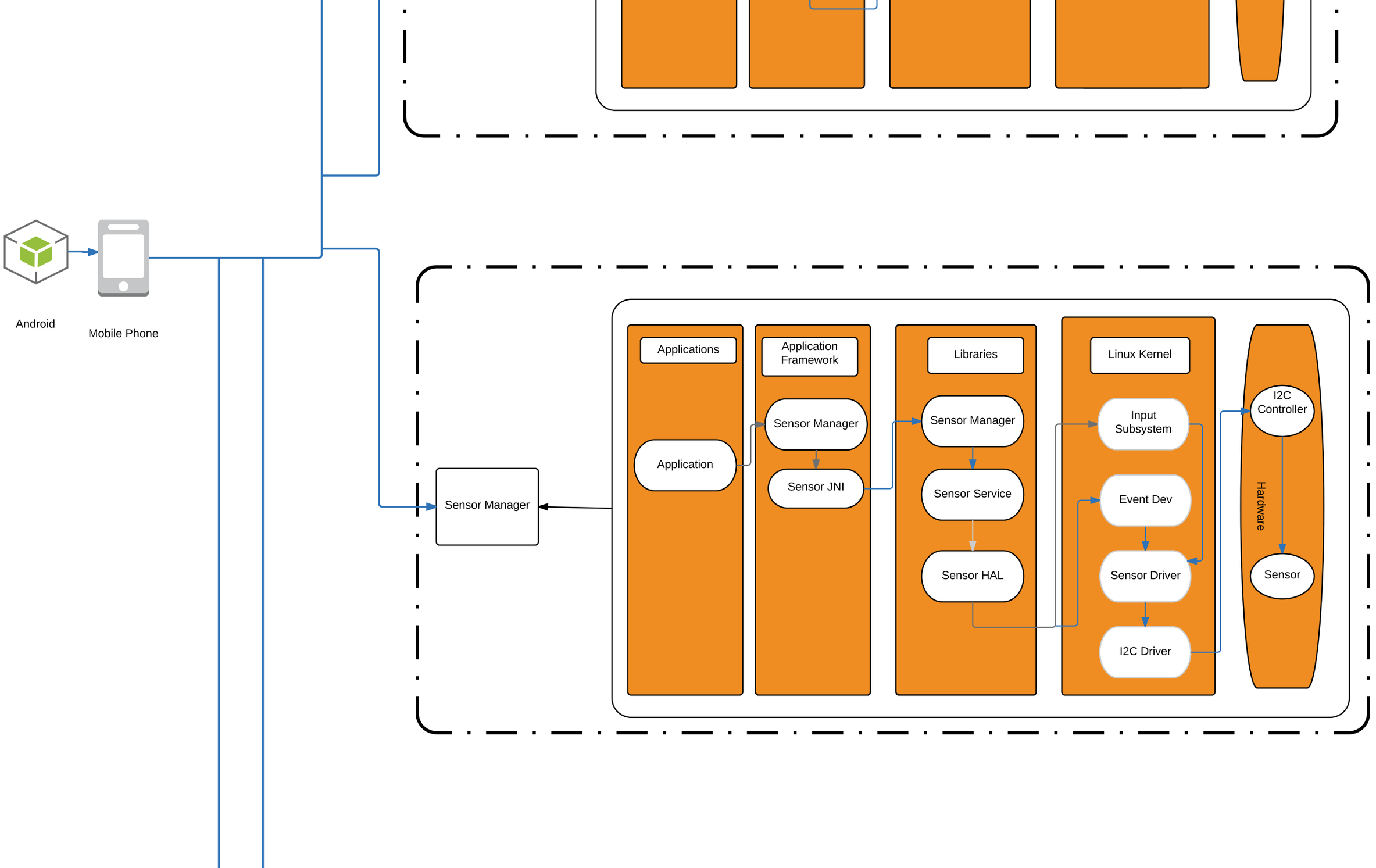
However, There can never be a strict line between the functional and nonfunctional requirements. With the increasing advances in technology and competition, the non-functional requirements become crucial and thus functionally required.

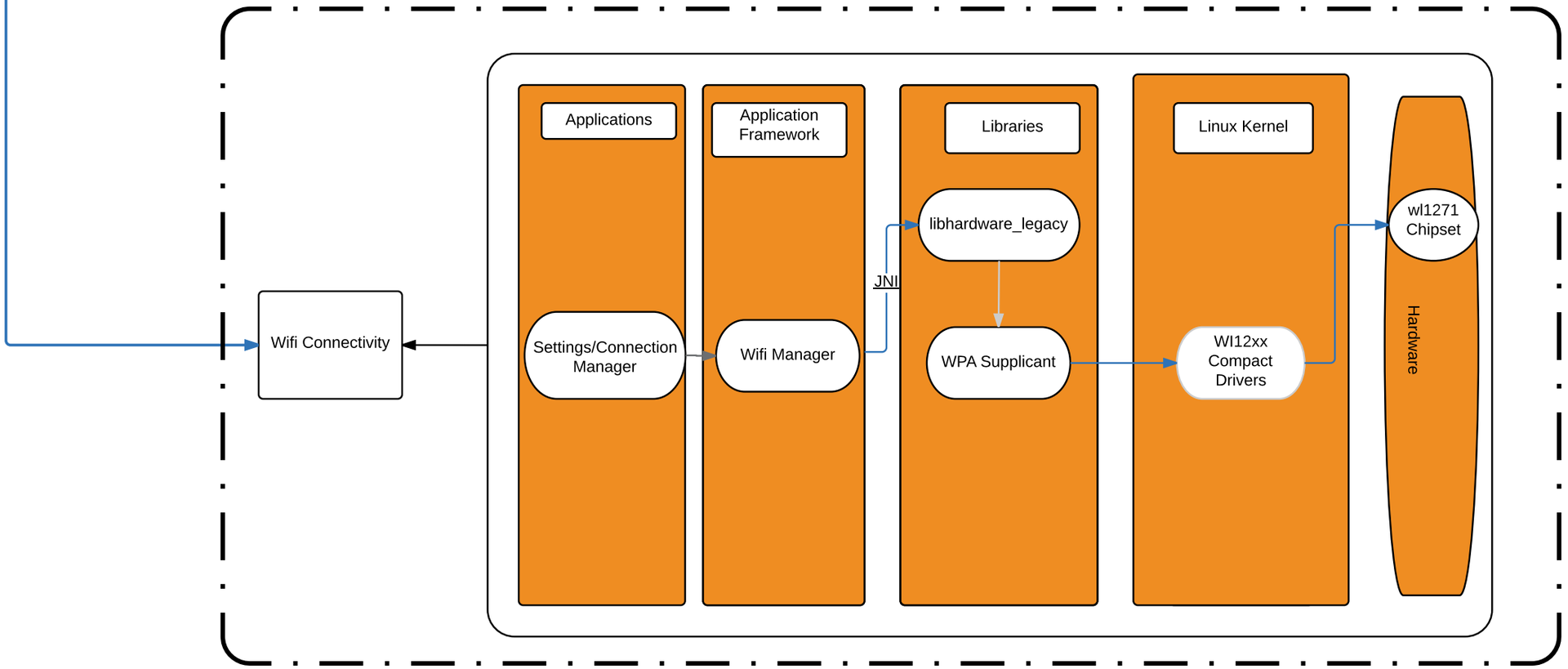
Android Operating System Work Flow

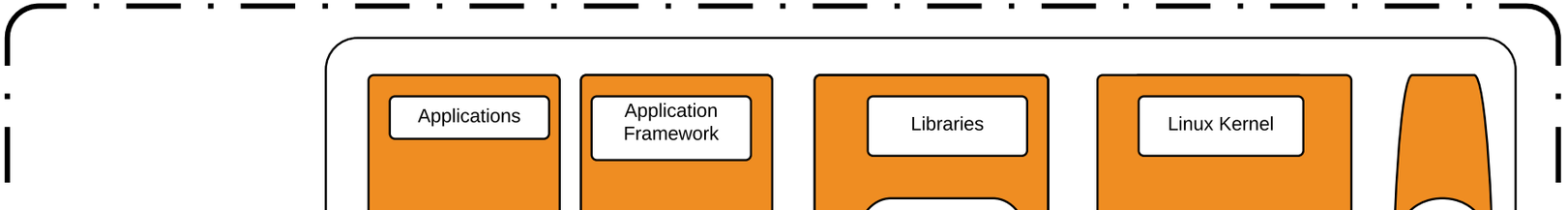


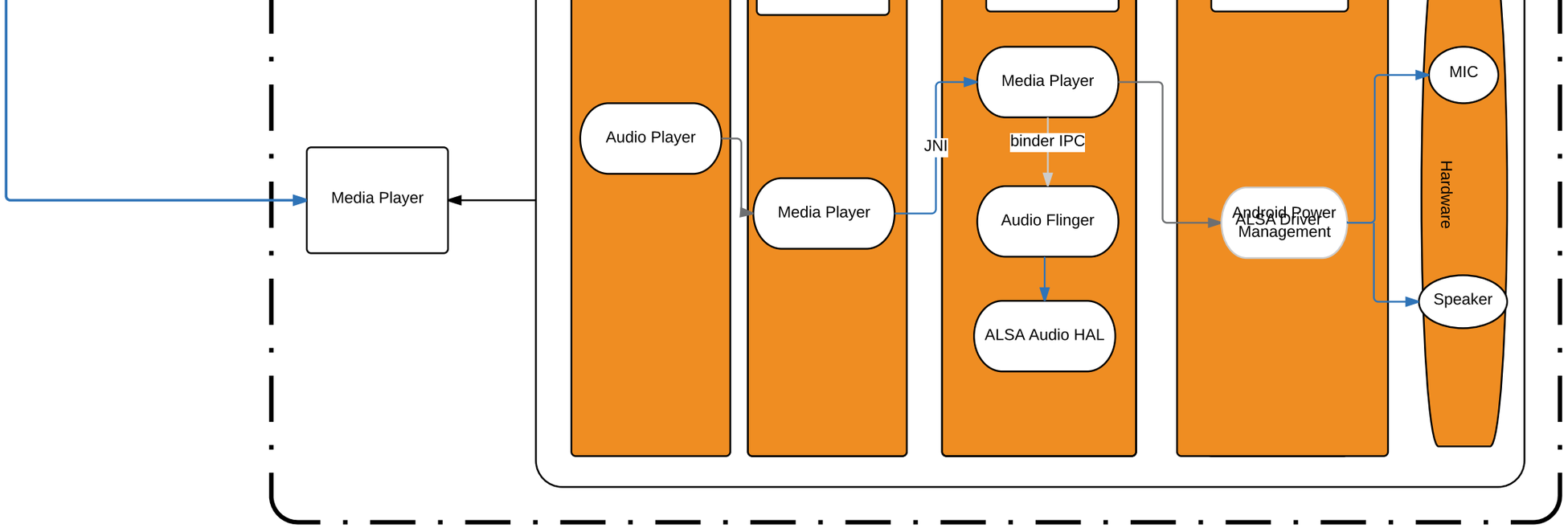

Framework
Phone
/java/android/telephony
Call Manager
Application
RIL
/java/android/telephony
/gsm
Radio Daemon
(rlid)
Vendor RIL
Linux ip Stack
Packet Driver
Voice and Control
BaseBand
Telephony


Applications
Application
Framework
Libraries
Linux Kernel
Application 1
Application 2
Power Manager
Power
Power Manager
Service
Core Libraries
POWER
Android Power
Management
Linux Power
Management
Device Driver
PMIC
Hardware
Dev
Power Manager


Android
Mobile Phone
Sensor Manager
Applications
Application
Application Framework
Sensor Manager
Sensor JNI
Libraries
Sensor Manager
Sensor Service
Sensor HAL
Linux Kernel
Input Subsystem
Event Dev
Sensor Driver
I2C Driver
Hardware
I2C Controller
Sensor


Applications
Application Framework
Libraries
Linux Kernel
Hardware
Wifi Connectivity
Settings/Connection Manager
Wifi Manager
JNI
libhardware_legacy
WPA Supplicant
WI12xx Compact Drivers
wl1271 Chipset


Applications
Application Framework
Libraries
Linux Kernel


Media Player
Audio Player
Media Player
Media Player
JNI
binder IPC
Audio Flinger
ALSA Audio HAL
ALSA Driver
Android Power Management
MIC
Hardware
Speaker