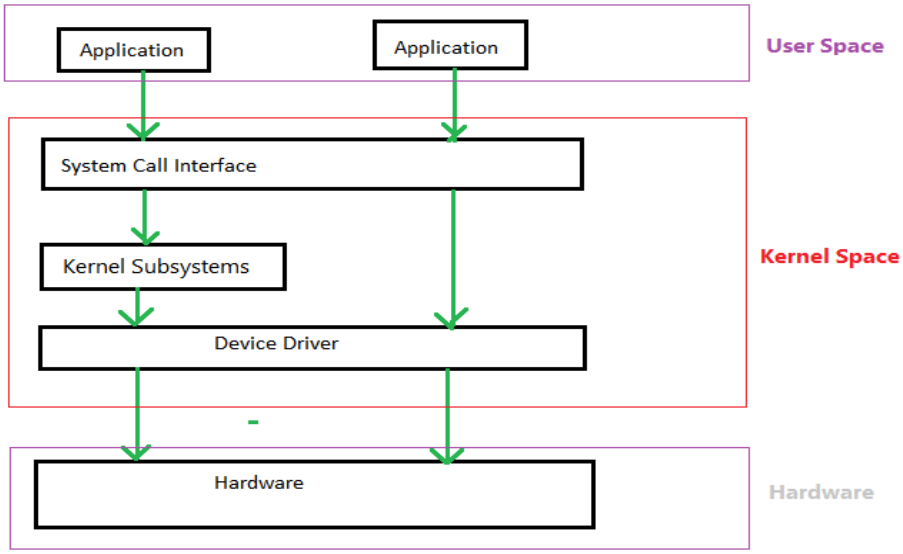
# KERNEL, PROCESS & THREAD OVERVIEW

## Kernel




**What is kernel?**
Kernel is the core program of operating system (OS).
It is loaded into memory at startup and keep alive until OS shut down.
Kernel is located at a separate area in memory to prevent accessing from user process.

**What does kernel do?**
Kernel is mainly in charge of managing four general ereas:
- Manage process and thread
- Manage memory
- Interact with hardware and process
- Support system calls

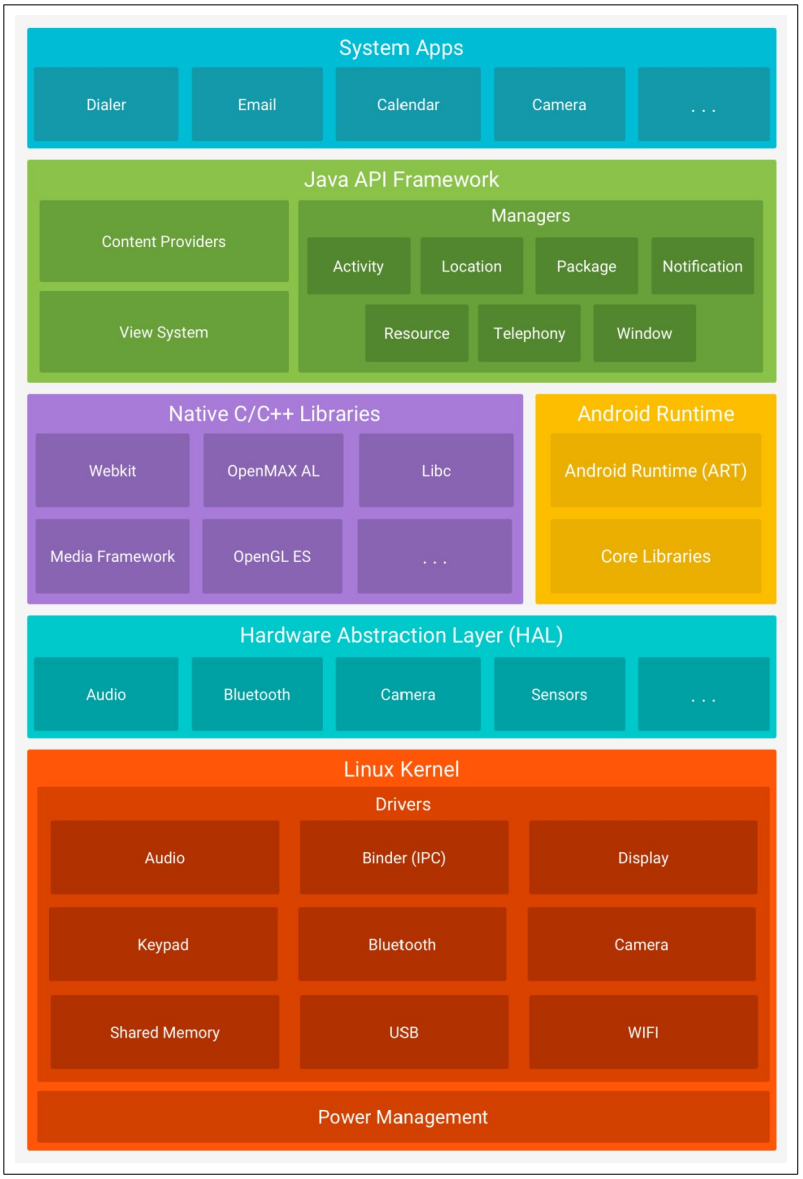
*For short, kernel acts as the interface between application and hardware.*

**How does kernel boot into memory?**
- The boot firmware or BIOS loads and runs the boot loader
- The boot loader finds the kernel image on disk, loads it into memory and starts it
- The kernal initializes the devices and drivers, mount the root filesystem and the rest of system process.

**How boot loader finds the kernel image on disk?**
The location is defined in kernel parameter.
In Ubuntu, kernel parameter is located at: */proc/cmdline.*

```
invistd@server:/$ cat /proc/cmdline
BOOT_IMAGE=/boot/vmlinuz-4.15.0-38-generic root=UUID=4e8bf64c-cc68-11e8-b9cd-080027a5bd3f ro maybe-u
biquity
invistd@server:/$
```

**Android Linux Kernel**
**Android is based on the Linux kernel.**




**Ubuntu is based on Linux kernel, too. What are the similarities, differences between Android and Ubuntu?**
Android and Ubuntu are Operating System (OS). They are both based on Linux kernel.

But, they include different software, such as graphic user interface (GUI), shell, tools and services.
Ubuntu contain many other pieces of software like GNU shell, X Window System, GNOME desktop, Firefox browser.
Rather than Ubuntu, Android includes Dalvik Virtual Machine to run application written in Java, and its own framework and libraries.

**Android Process**
Android Process is based on Linux process. However, it includes runtime execution environment such as Dalvik Virtual Machine.

**Process & Application Lifecycle**
*When does a process start?*
A process is created at the start of the application. At the start, OS will perform below steps:
- Create Linux process
- Create runtime execution environment
- Create *Application* instance
- The entry point is *onCreate()* method of Application object

*When does a process terminate?*
It is NOT controlled by the application itself. It is determined by system through some aspects:
- The important of process to user.
- The avaiable memory in device.
- If user forces stop application, process terminates.

**Android Thread**
Android delegates thread management to Linux kernel.
So, even if you create thread by Java or C/C++, it will be handled by the kernel.

**Main Thread**
The main thread is started on application start and stays alive during the lifetime of the process.
Main thread is in charge of updating Android components and UI elements.

There are two rules we need to be careful when working with main thread:

- If the main thread is blocked for more than 5 seconds, OS will hide your application and show dialog Application Not Responding (ANR)
- If UI element is access from outside the main thread, exception *CalledFromWrongThreadException* will be trigged.