



CS260 Intro to Java & Android

04.Android Details

Fall 2011

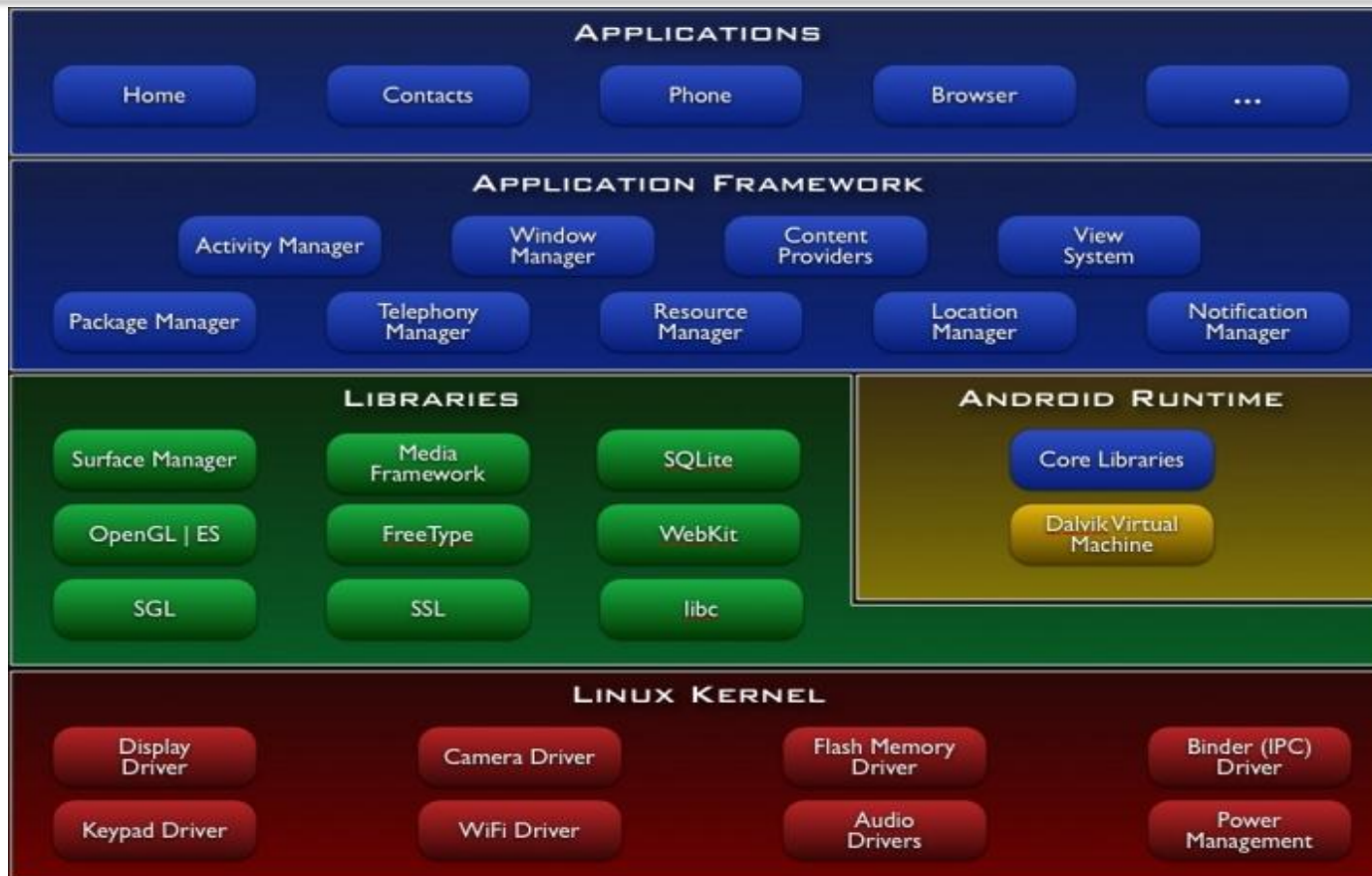
Important Android Dates

- Google acquires Android, August 2005
- Open Handset Alliance (OHA) announced, November 2007. OHA developed Android and is “...committed to commercially deploy handsets and services using the Android Platform.” [10]
- First Android Phone, G1, October 2008
- Android SDK 1.0, October 2008

What is Android?

- Android is a software stack (set of programs working together) for mobile devices that includes:
 - an operating system
 - middleware
 - applications

Android Architecture

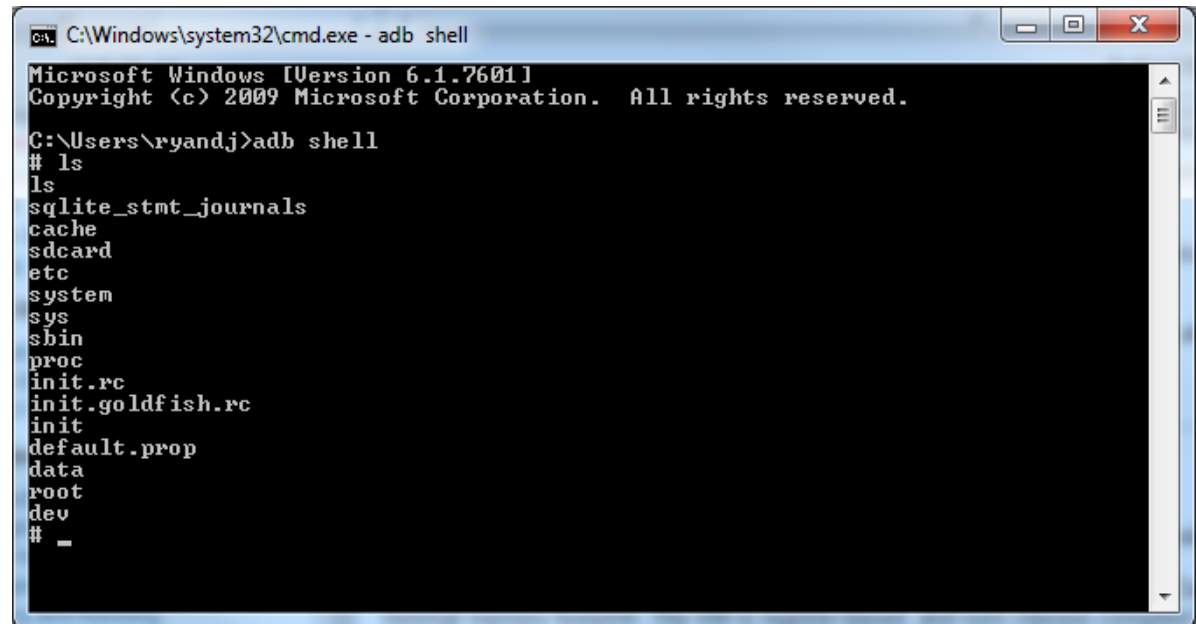


Linux Kernel

- Android relies on Linux version 2.6 for:
 - memory management
 - process management
 - security
 - networking
- You will not make Linux system calls
- Some utilities interact with Linux
 - e.g. adb shell

adb shell

- With an emulator running, open a Windows command shell
- Type adb shell
- Type ls



```
C:\Windows\system32\cmd.exe - adb shell
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\ryandj>adb shell
# ls
ls
sqlite_stmt_journals
cache
sdcard
etc
system
sys
sbin
proc
init.rc
init.goldfish.rc
init
default.prop
data
root
dev
# _
```

- Now you can examine the Linux file system of the phone which aids in debugging

Native Libraries

- The native libraries are written in C & C++
- The libraries are exposed through the Application framework

Application Framework

- Android developers have access to the same framework APIs use by the core applications
- Services and systems for applications include:
 - **Views** – including lists, grids, buttons,
 - **Content Providers** – methods for accessing data
 - **Resource Manager** – organizes non-code resources such as strings and layout files
 - **Notification Manager** – displays custom alerts
 - **Activity Manager** – manages lifecycle of applications

Android Runtime

Every Application:

- Runs in its own process space
- Has a separate instance of the Dalvik VM
 - The Dalvik VM uses the Linux kernel for functionality such as threading and low-level memory management
 - Dalvik VM \neq JVM
- All Android code is written in Java and run within the Dalvik VM

What is Dalvik?

- Dalvik is a VM optimized for low memory requirements
- Android code is compiled into bytecodes executed by the Dalvik VM
- bytecodes are machine-independent instructions

Android Applications

- Apps are written in Java
- Code is compiled into Android package (.apk file)
- All code (including data & resource files) in .apk is one application

Android Application Specifics

- Android is a multi-user Linux system where each application is a user
- Only one application is visible at a time
- Each process has its own VM running an application in isolation
- Two or more applications can share data
- Applications consist of one or more activities

What is an Activity?

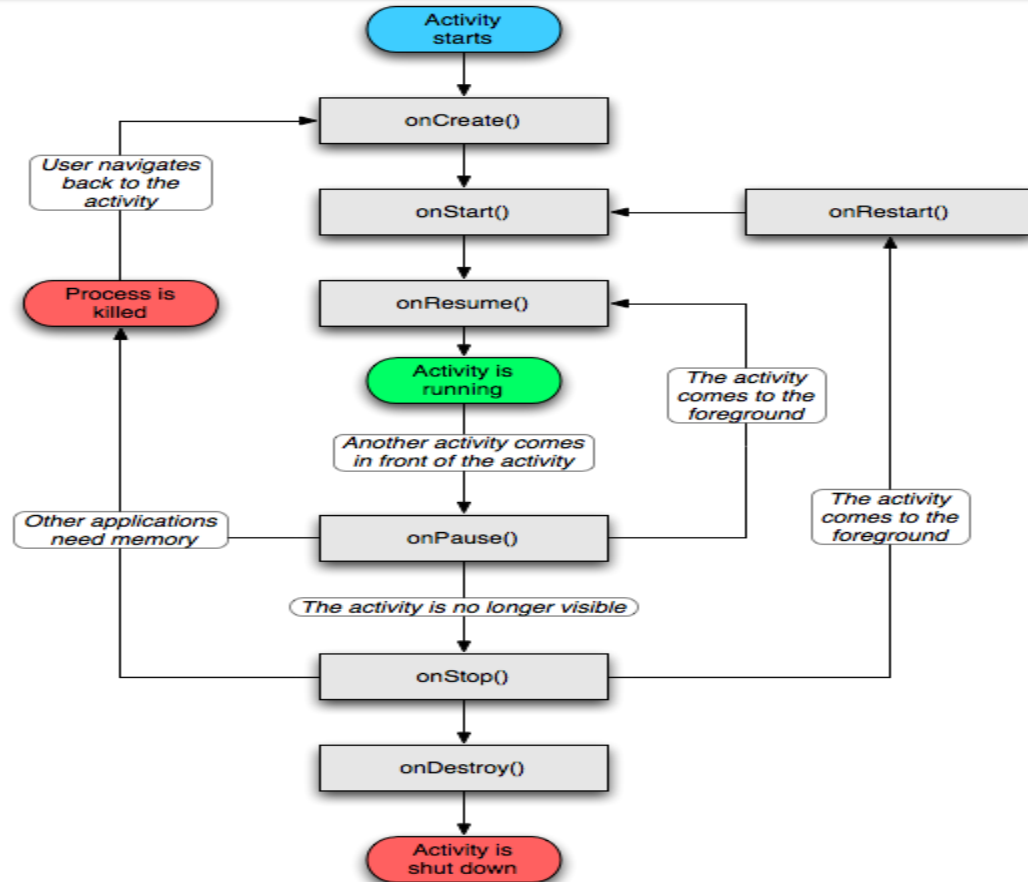
- An Activity represents a single screen with a UI
- Ex: Email Application consists of activities for
 - Showing list of emails
 - Composing an email
 - Reading an email
- Each activity is independent
- Other applications can use a particular activity if the email application gives permission to do so

Activity Lifecycle

Activity – a process that performs some specific action

- Every Android application is made up of one or more activities managed on an Activity Stack (AS).
- A new activity is always placed on top of the AS and then becomes the running activity.
- The older activity always remains below the running activity on the stack and will not come to the foreground until the current activity stops

Activity Lifecycle Visual



Activity States

An activity has essentially four states:

- **running** – in the foreground of the screen
- **paused** – lost focus but still visible with all state maintained
 - How? A new activity that is transparent or not full sized is running on top of the stack
- **stopped** – a new activity completely obscures another activity
 - The stopped activity is no longer visible
 - State is maintained
- **destroyed** – the activity must be completely restarted and the state information must be

ActivityLifecycleDemo Application

In CS260-01 Public is an Android Project called ActivityLifecycleDemo

1. Import project into your workspace
2. Let's take a look at the source code
3. Run the application

Q1: What is the difference between hitting the home button and back button?

