

# Android Applications

# Chapter 3 + Android Dev Guide Readings

## Creating Applications & Activities

What exactly is Android?

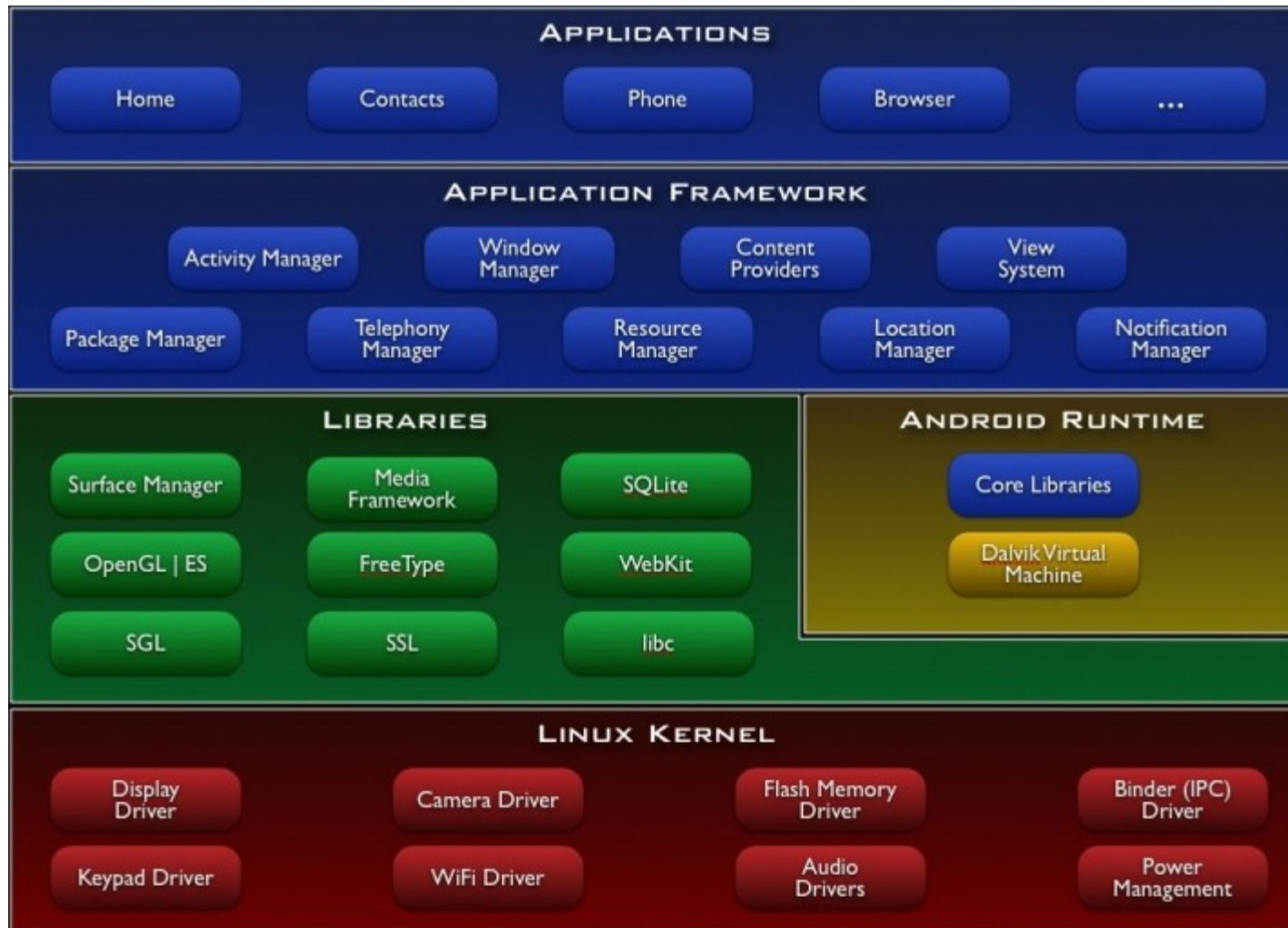
Answer: A software stack for mobile devices that includes:

- an OS
- middleware – software that connects applications or software components
- key applications – email client, browser, phone, contacts, calendar, maps, ..., all written in Java

# Features

- Application Framework
- Dalvik Virtual Machine
- Integrated Browser
- Optimized Graphics – 2D & 3D graphics with OpenGL ES 1.0 (OpenGL for Embedded Systems is a subset of the OpenGL 3D graphics API)
- SQLite
- Media Support for audio, video, still images
- GSM Telephony
- Bluetooth, EDGE, 3G, WiFi
- Camera, GPS, compass, accelerometer

# Android Architecture



<http://developer.android.com/guide/basics/what-is-android.html>

# Creating Applications & Activities

Android applications consist of six possible components:

- Activities
- Services
- Content Providers
- Intents
- Broadcast Receivers
- Notifications

The project manifest describes component specifics and component interaction

# Application Manifest

Each Android project has an `AndroidManifest.xml` (manifest file)

The manifest defines the applications components and structure as well as having metadata for specifying things like themes, icons used, ...

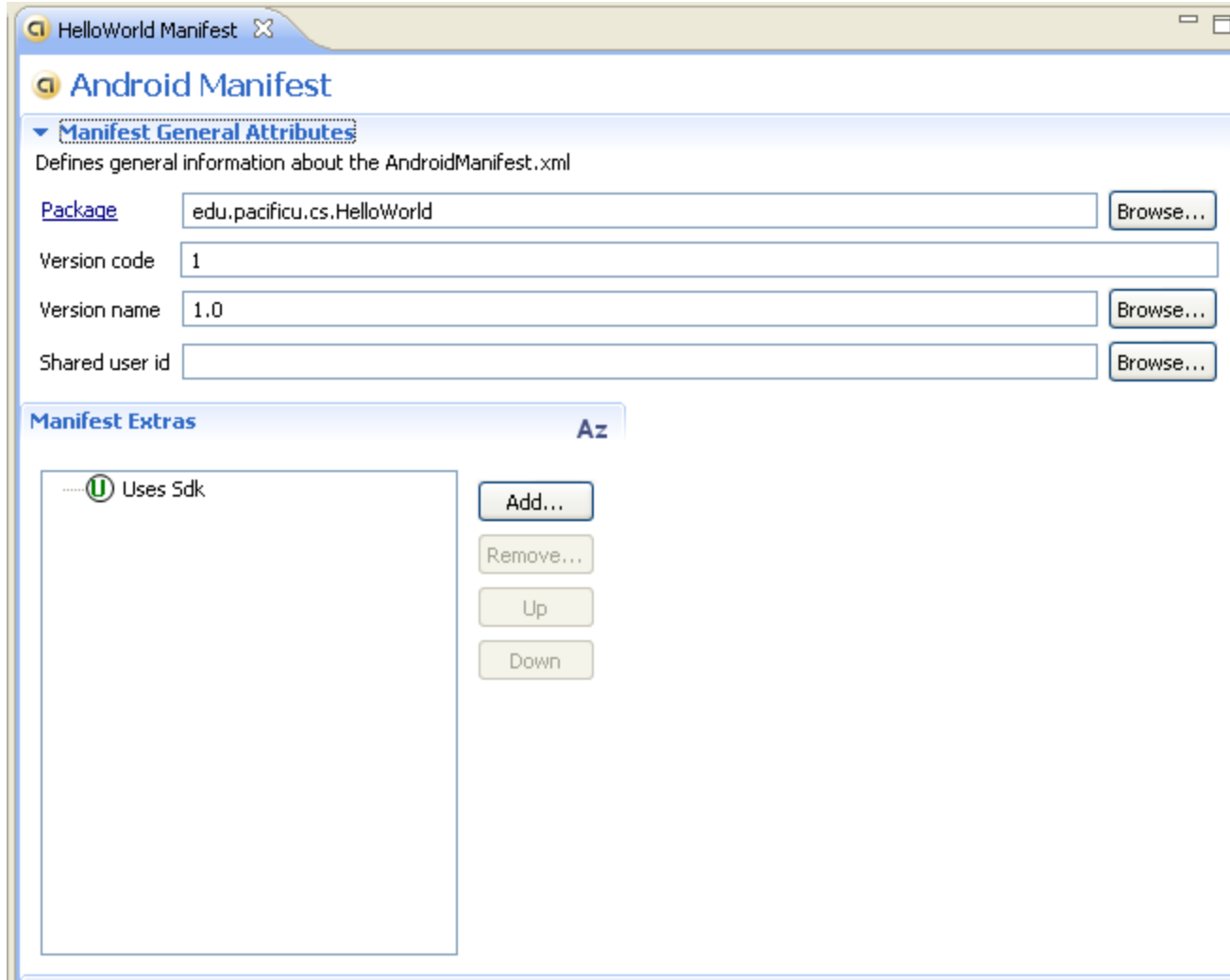
The manifest specifies application permissions and determines component interaction among other things

# HelloWorld AndroidManifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="edu.pacificu.cs.HelloWorld"
    android:versionCode="1"
    android:versionName="1.0">
    <application android:icon="@drawable/icon"
        android:label="@string/app_name">
        <activity android:name=".HelloWorld"
            android:label="@string/app_name">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>
    <uses-sdk android:minSdkVersion="2" />
</manifest>
```

# AndroidManifest.xml

Manifest  
Tab





# AndroidManifest.xml

Application  
Tab

Application Manifest Application

**Application Toggle**

- The [application](#) tag describes application-level components contained in the package, as well as general application attributes.
- Define an `<application>` tag in the `AndroidManifest.xml`

**Application Attributes**

Defines the attributes specific to the application.

Name	<input type="text"/>	Browse...	Allow task reparenting	<input type="text"/>	▼
Theme	<input type="text"/>	Browse...	Has code	<input type="text"/>	▼
Label	<input type="text" value="@string/app_name"/>	Browse...	Persistent	<input type="text"/>	▼
Icon	<input type="text" value="@drawable/icon"/>	Browse...	Enabled	<input type="text"/>	▼
Description	<input type="text"/>	Browse...	Debuggable	<input type="text"/>	▼
Permission	<input type="text"/>	▼	Manage space activity	<input type="text"/>	Browse...
Process	<input type="text"/>	Browse...	Allow clear user data	<input type="text"/>	▼
Task affinity	<input type="text"/>	Browse...			

**Application Nodes** [A] [R] [S] [M] [P] [U] [A] [Az]

- [A] .HelloWorld (Activity)

Buttons: Add..., Remove..., Up, Down

# Android Developers Guide

Let's go to:

<http://developer.android.com/guide/topics/manifest/manifest-element.html>

and examine

<manifest>

<application>

<activity>

<intent-filter>

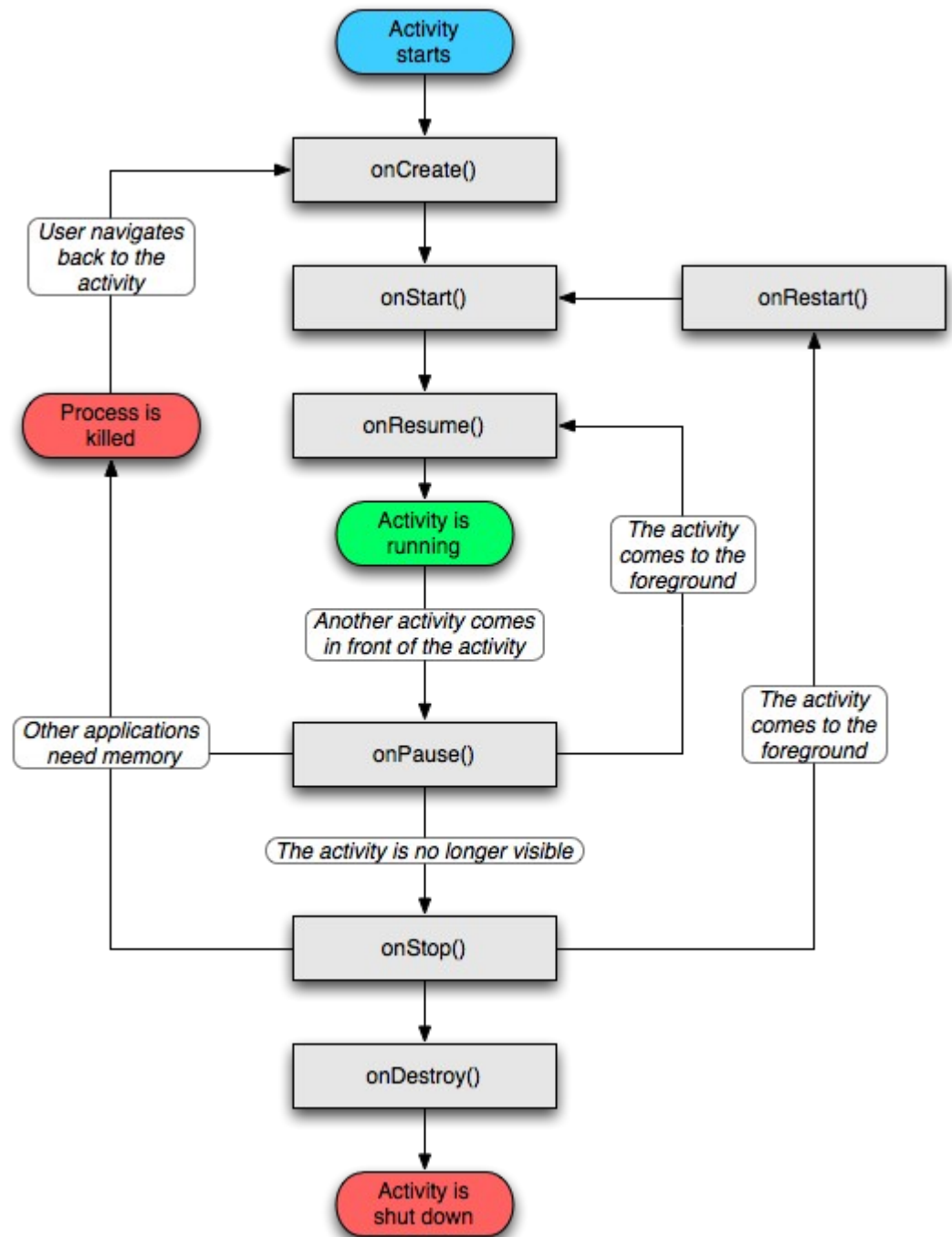
# Activity Lifecycle

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

Import  
05.code/ActivityLifecycleDemo



# 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