

ANDROID ARCHITECTURE

CMPT 381



Overview

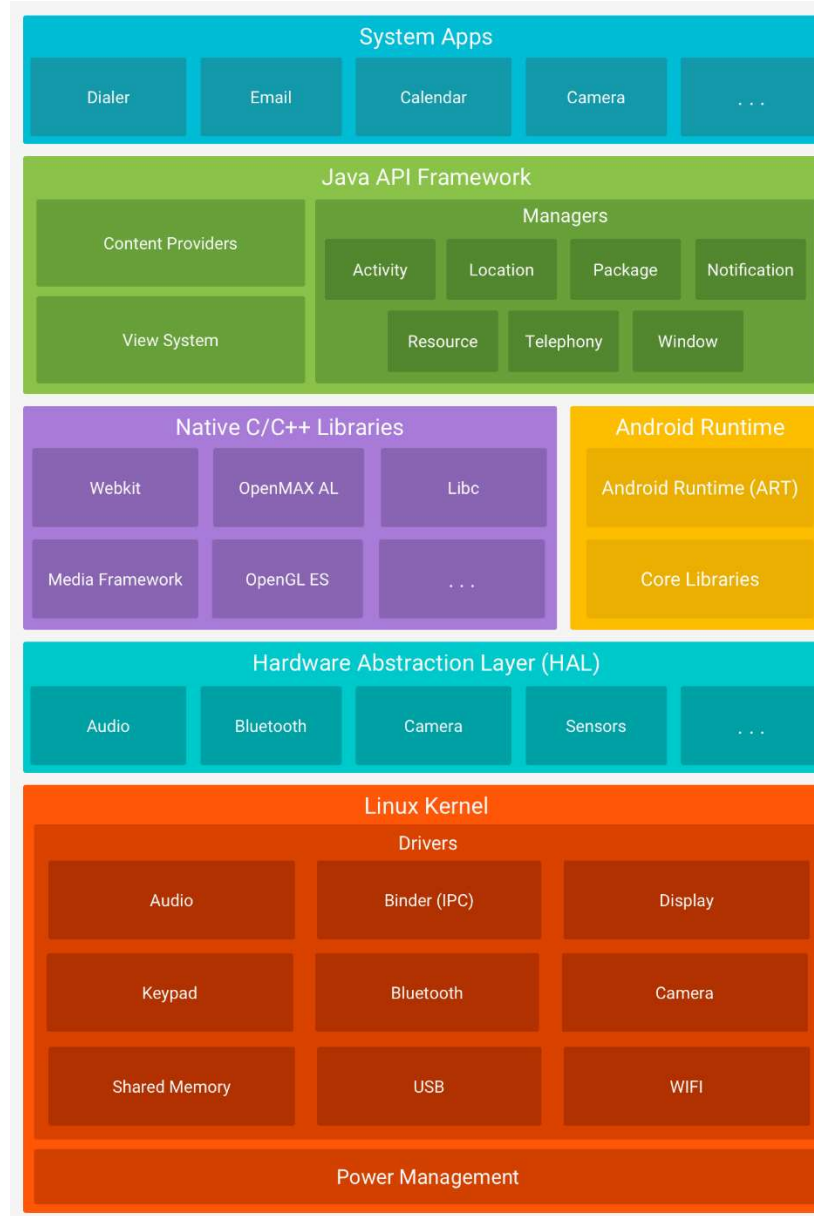
Android architecture

Android applications

Android application components

Activities and Intents

Android Platform Architecture



Android applications

- A combination of app components that can be invoked independently
 - Multiple entry points, no “main()” function
 - No single application class as in JavaFX
- Each app runs in its own Linux security sandbox
 - Limited access to system capabilities (Managers)
- Packaged for installation as an APK
- Architectural focus on dealing with limited memory

App components

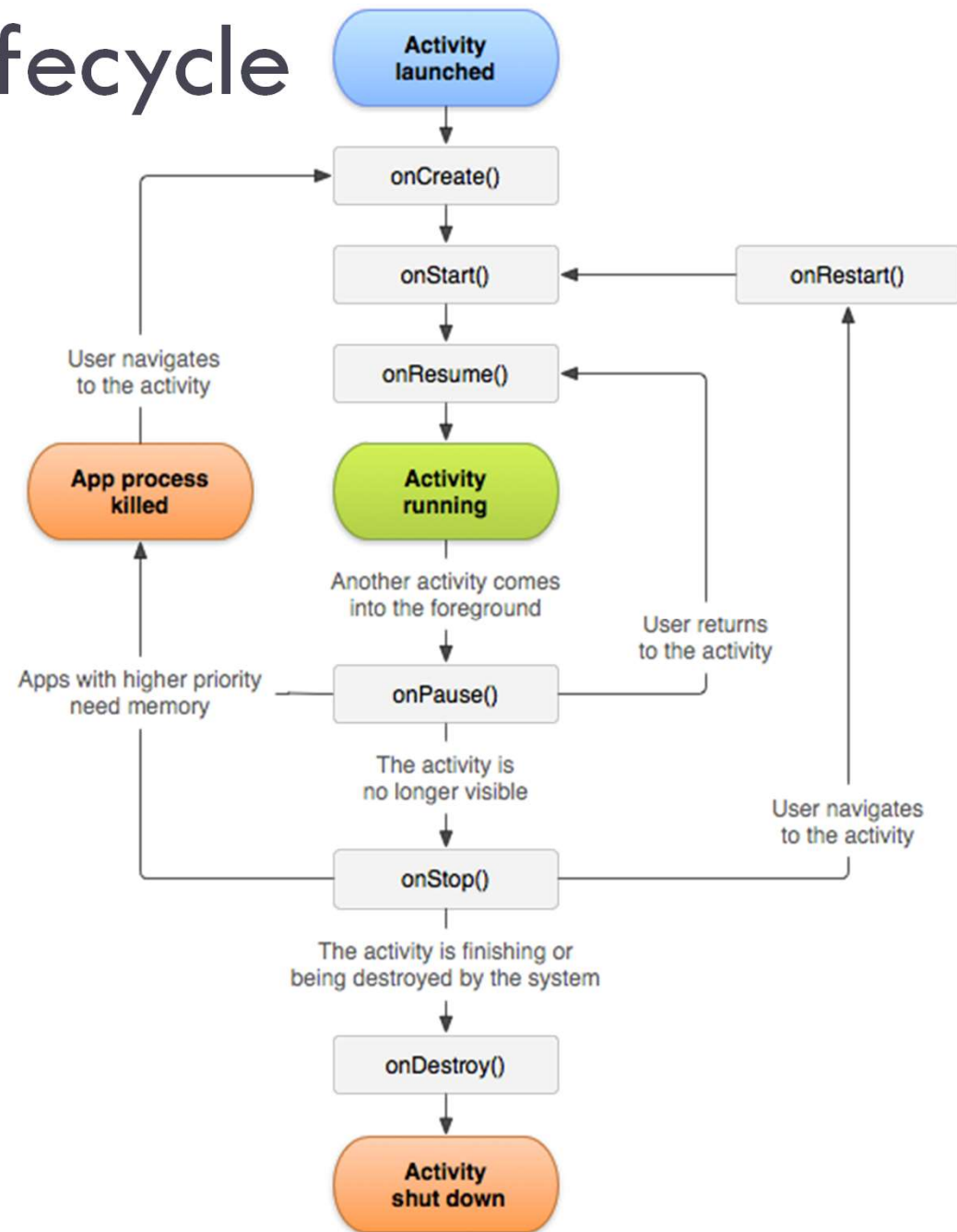
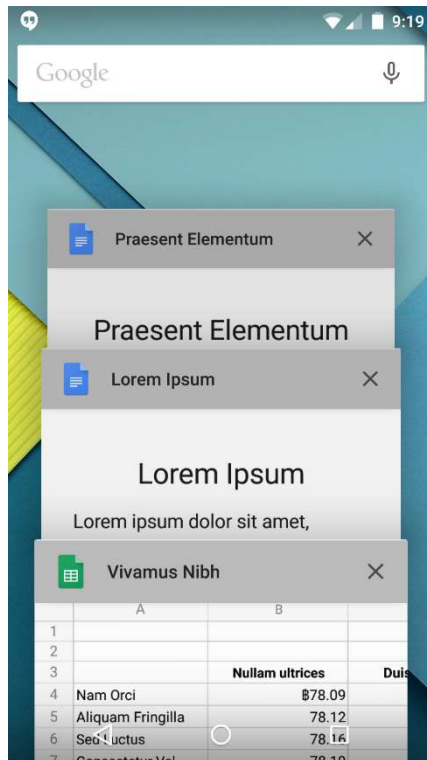
- Activities
 - Main UI component: “one Activity per screen”
 - Can be invoked by any app
- Services
 - Background process (e.g., downloading web page)
- Content providers
 - Provides data to applications (e.g., SQL, files, web)
- Broadcast receivers
 - Listeners for system events (e.g., power on, phone call)

Activities

- The main component for 381
- Provides a window for a UI
 - One activity, one screen
 - E.g., email list screen / email read screen
- Launched in response to an *Intent*
- Lifecycle:

onCreate()	onStart()	onResume()	onPause()
onStop()	onRestart()	onDestroy()	

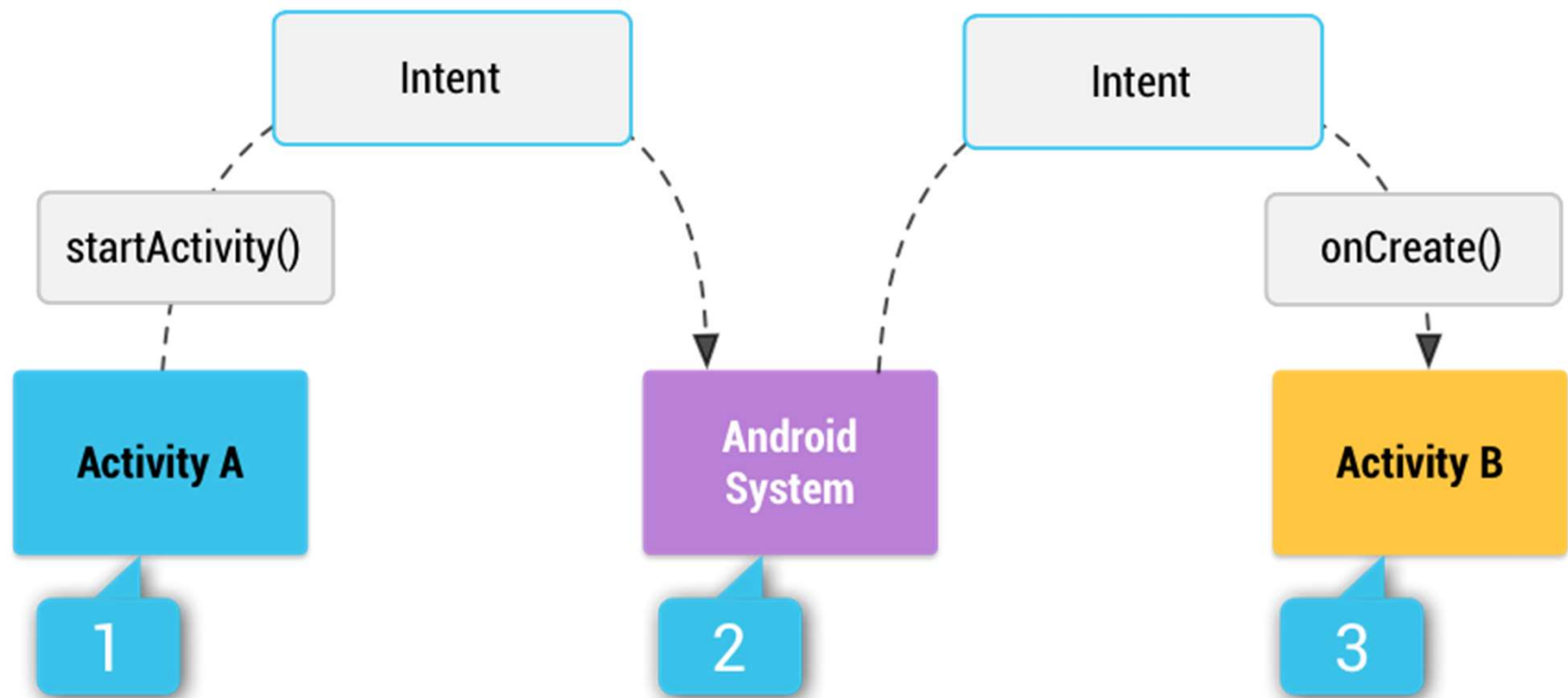
The activity lifecycle



Intents

- Object used as a message to request action from another app component
 - Starting an activity
 - Starting a service
 - Delivering a broadcast
- Types of intents
 - Explicit: used within your own app
 - Implicit: when you need general capabilities
- Intent filters
 - Part of each application's manifest file

Intents



[1] Activity A creates an [Intent](#) with an action description and passes it to [startActivity\(\)](#). **[2]** The Android System searches all apps for an intent filter that matches the intent. When a match is found, **[3]** the system starts the matching activity (Activity B) by invoking its [onCreate\(\)](#) method and passing it the [Intent](#).

Intent filter

```
<activity android:name="ShareActivity">
  <intent-filter>
    <action android:name="android.intent.action.SEND"/>
    <category android:name="android.intent.category.DEFAULT"/>
    <data android:mimeType="text/plain"/>
  </intent-filter>
</activity>
```

Using an intent to take a picture

```
static final int REQUEST_IMAGE_CAPTURE = 1;

private void dispatchTakePictureIntent() {
    Intent takePictureIntent = new Intent(MediaStore.ACTION_IMAGE_CAPTURE);
    if (takePictureIntent.resolveActivity(getPackageManager()) != null) {
        startActivityForResult(takePictureIntent, REQUEST_IMAGE_CAPTURE);
    }
}
```

App permissions

- Needed for every system service accessed
- Added to app's manifest file (manifest.xml)
- E.g., for access to the camera:

```
<manifest ... >
    <uses-feature android:name="android.hardware.camera"
                  android:required="true" />
    ...
</manifest>
```