

Introduction to Android

Dominic Duggan
Stevens Institute of Technology

1

Trends

- Consumerization
- Virtualization
- App-ification
 - From applications to apps
- Always-available self-service cloud
- Mobility shift
 - Wherever and whenever you want

2

Operating Systems for Cell Phones

Operating System	Market Share (Q3 2008)
Symbian	47%
iPhone	17%
RIM Blackberry	15%
Windows Mobile	14%
Linux	5%
Palm OS, BREW, J2ME	
Android	

3

Operating Systems for Cell Phones

Operating System	Market Share (Q3 2010)
Symbian	37%
iPhone	17%
RIM Blackberry	15%
Windows Mobile	3%
Linux	2%
Palm OS, BREW, J2ME	
Android	25%
Windows Phone	

4

Operating Systems for Cell Phones

Operating System	Market Share (Q3 2011)
Symbian	16.9%
iPhone	13.8%
RIM Blackberry	9.6%
Windows Phone	1.2%
Bada	2.2%
Others	0.9
Android	57.4%

5

Operating Systems for Cell Phones

Operating System	Market Share (Q3 2012)
Symbian	2.6%
iPhone	14.4%
RIM Blackberry	4.1%
Windows Phone	2.0%
Others	4.5%
Android	74.9%

6

Operating Systems for Cell Phones

Operating System	Market Share (Q3 2013)
iPhone	12.8%
RIM Blackberry	1.7%
Windows Phone	3.6%
Others	0.6%
Android	81.2%

7

Operating Systems for Cell Phones

Operating System	Market Share (Q3 2014)
iPhone	11.7%
RIM Blackberry	0.5%
Windows Phone	2.9%
Others	4.5%
Android	84.4%

8

Operating Systems for Cell Phones

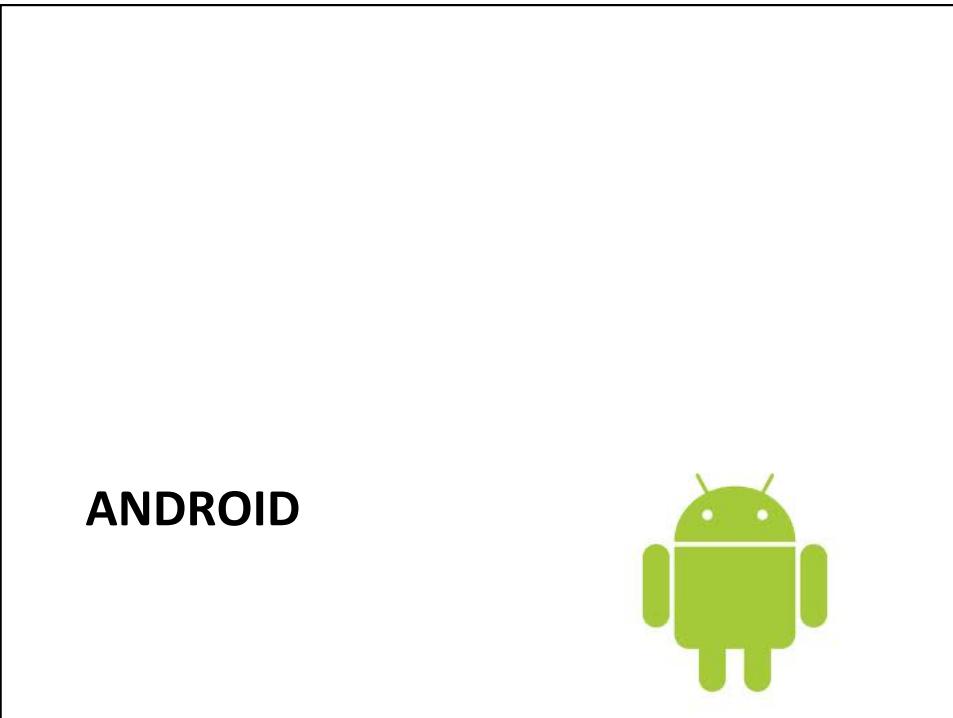
Operating System	Market Share (Q1 2016)
iOS	15.4%
Windows Phone	0.8%
Others	0.4%
Android	83.4%

9

Operating Systems for Cell Phones

Operating System	Market Share (Q1 2017)
iOS	14.7%
Windows Phone	0.1%
Others	0.1%
Android	85.0%

10

A screenshot of the Open Handset Alliance website. The header features the "open handset alliance" logo and a navigation bar with links for Home, Alliance, Android, Developers, Press, and Contact. A search bar is also present. The main content area has a blue background with a white mobile phone shape containing various icons like a video camera, a map, and a battery. The text "What would it take to build a better mobile phone?" is displayed above the phone icon. Below the phone, there is a paragraph about the alliance's commitment to openness and concrete plans. Another paragraph welcomes visitors to the Open Handset Alliance, mentioning the development of the Android platform. At the bottom, there are two buttons: "Develop for Android Get the SDK" and "Contribute to Android Get the Source".

open handset alliance

Home Alliance Android Developers Press Contact

What would it take to build a better mobile phone?

A commitment to openness, a shared vision for the future, and concrete plans to make the vision a reality.

Welcome to the Open Handset Alliance™, a group of 84 technology and mobile companies who have come together to accelerate innovation in mobile and offer consumers a richer, less expensive, and better mobile experience. Together we have developed Android™, the first complete, open, and free mobile platform.

We are committed to commercially deploy handsets and services using the Android Platform.

Develop for Android
Get the SDK

Contribute to Android
Get the Source

Android Motivation



Open architecture



All applications are equal



Breaking down boundaries



Fast & easy development

13

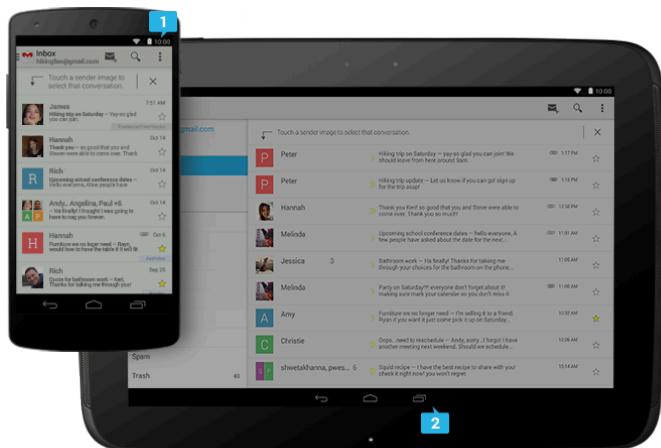
13

What is Android?

- A software stack for mobile devices.
 - Operating system
 - Middleware
 - Application framework
 - Key applications
- Java for app development

14

Not Just Smartphones



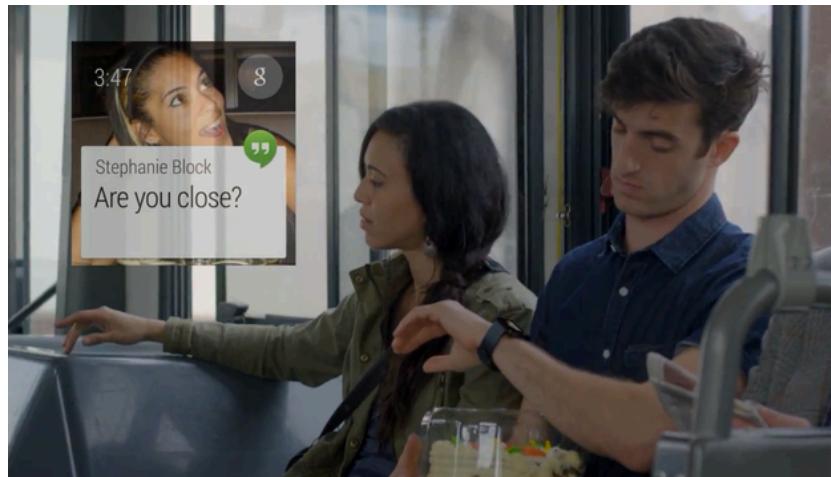
15

Not Just Smartphones



16

Not Just Smartphones

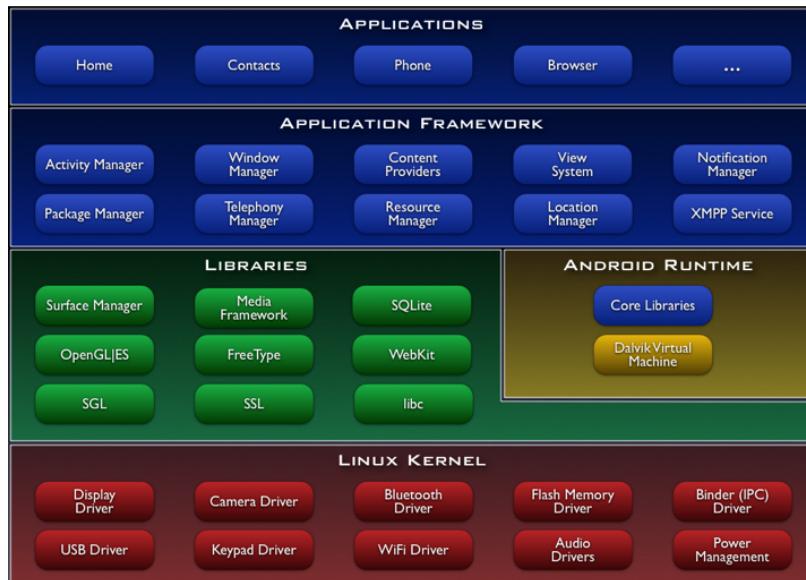


17

ANDROID ARCHITECTURE

18

Android Architecture



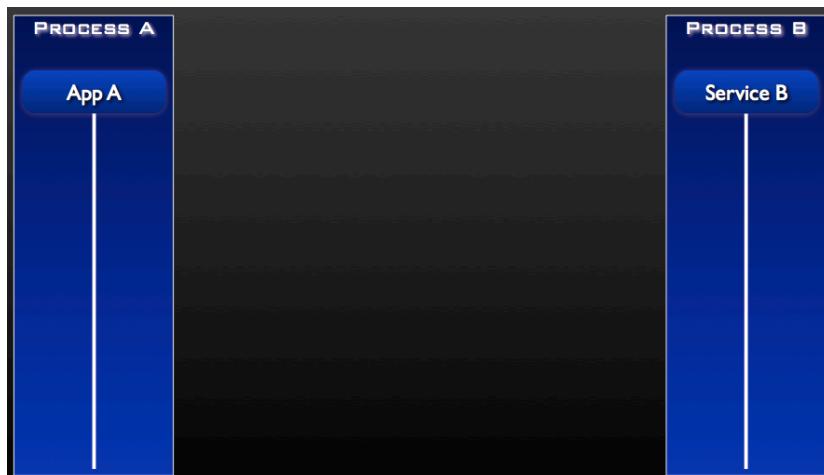
OS Kernel

- Linux 2.6.x
- Kernel enhancements:
 - Binder IPC: shared memory RPC
 - Wake locks (power management)
 - Alarms
 - Low memory killer



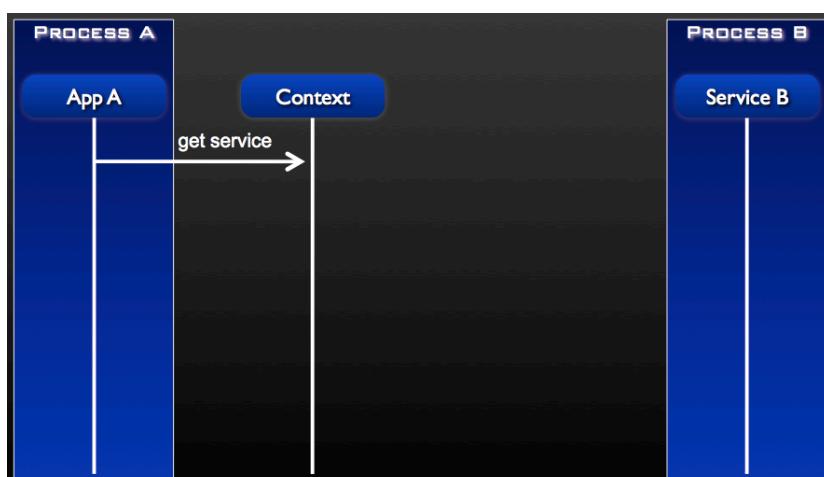
20

OS Kernel: Binder



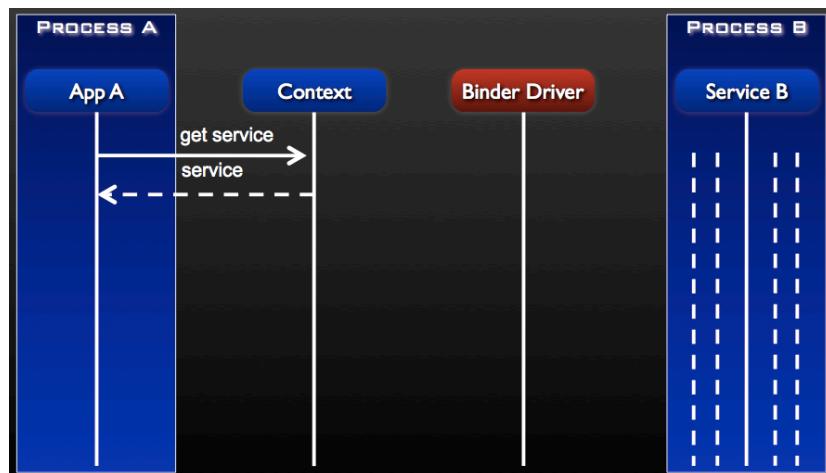
21

OS Kernel: Binder



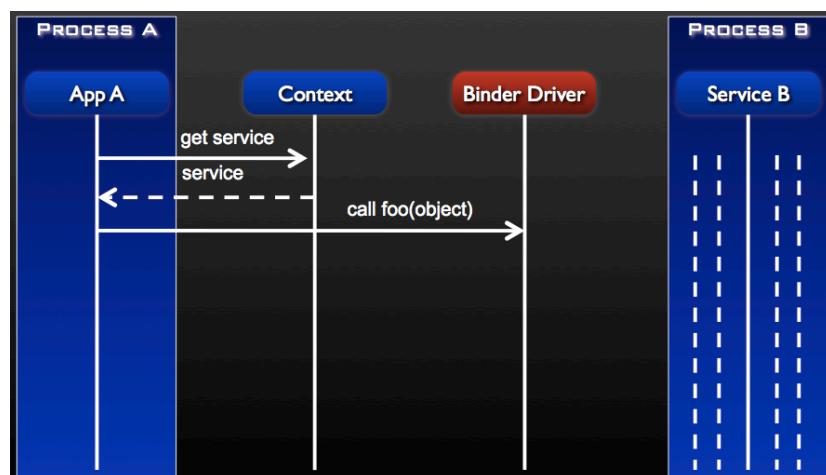
22

OS Kernel: Binder



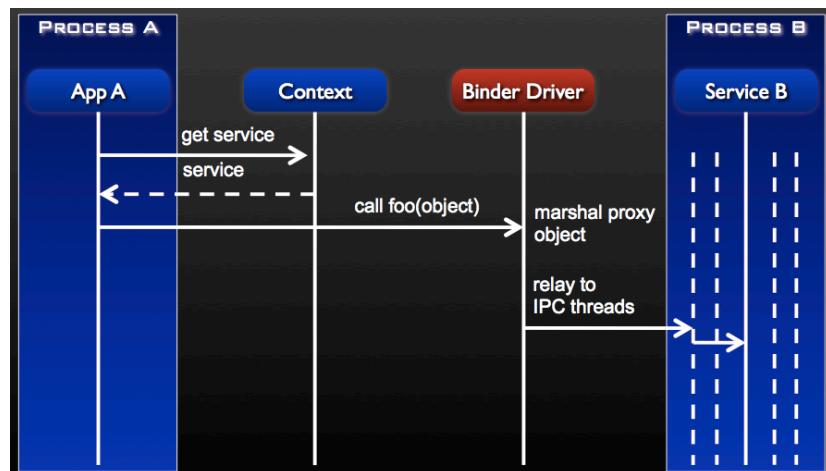
23

OS Kernel: Binder



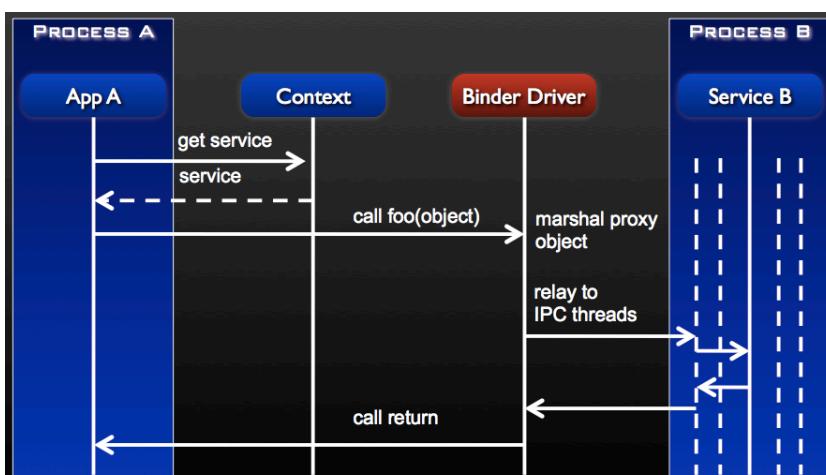
24

OS Kernel: Binder



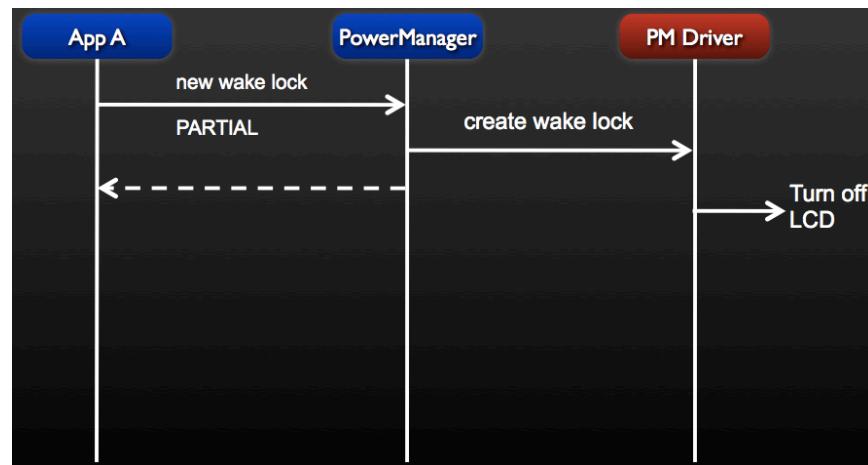
25

OS Kernel: Binder



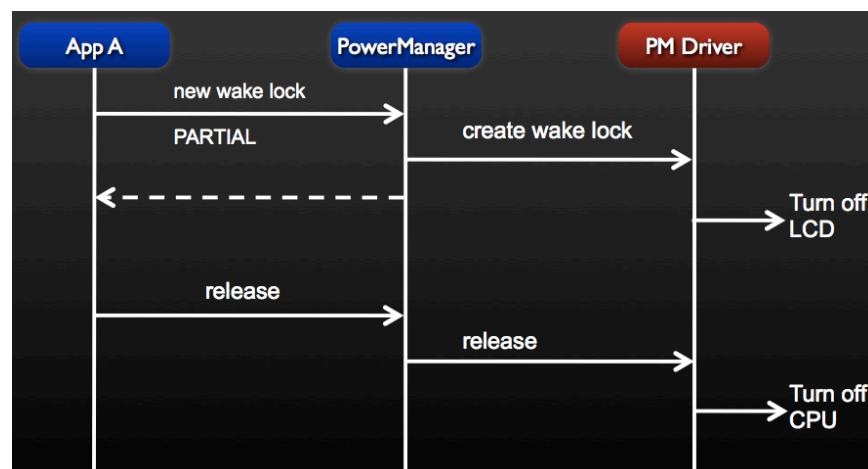
26

OS Kernel: Wake Locks



27

OS Kernel: Wake Locks



28

Runtime

- Linux processes
 - App separation
 - Permissions
- Dalvik VM: minimal memory footprint
 - Fast register-based execution (register-based)
 - DEX files (.class & .jar converted at run-time)
 - JIT compiler
 - Clean-room implementation (Not JVM...)



29

Runtime

- Linux processes
 - App separation
 - Permissions
- ~~Dalvik VM~~
- ART VM
 - Compile complete app (DEX) to native code
 - Efficient collector (smaller pause, parallelized)
 - Memory locking finer-grained



30

Libraries

- C/C++-implemented libraries
- Custom libc (small & fast, BSD)



31

Libraries

- C/C++-implemented libraries
- WebKit (browser toolkit)
 - Renders in full (desktop) view
 - Full CSS, Javascript, DOM, AJAX



32

Libraries

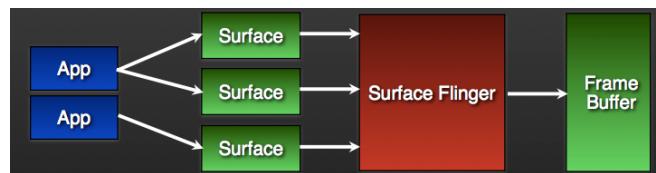
- C/C++-implemented libraries
- SQLite relational DBMS



33

Libraries

- Surface Flinger



34

Libraries

- Audio Flinger



35

Libraries

- User-Space Hardware Abstraction Layer (HAL)
 - Service-specific driver interfaces



36

Application Framework

- Content Providers Sharing data between applications
- Resource Manager External resources e.g. strings, graphics
- Notification Manager Signaling users
- Activity Manager Controls lifecycle of activities
- Views Build user interfaces



37

Application Framework

- Hardware Services
 - Access to low-level hardware API
 - Typically accessed through local Manager

```
LocationManager lm =  
    (LocationManager) Context  
    .getSystemService(Context.LOCATION_SERVICE);
```



38

Application Framework

- Hardware Services
 - Telephony
 - Location
 - Bluetooth
 - WiFi
 - USB
 - Sensor



39

Applications

- Email client
- SMS program
- Calendar
- Maps
- Browser and others...



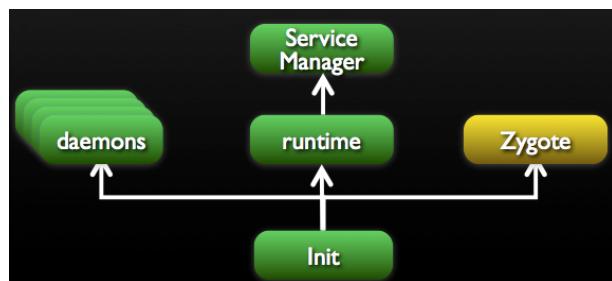
40

ANDROID RUNTIME

41

Android Runtime

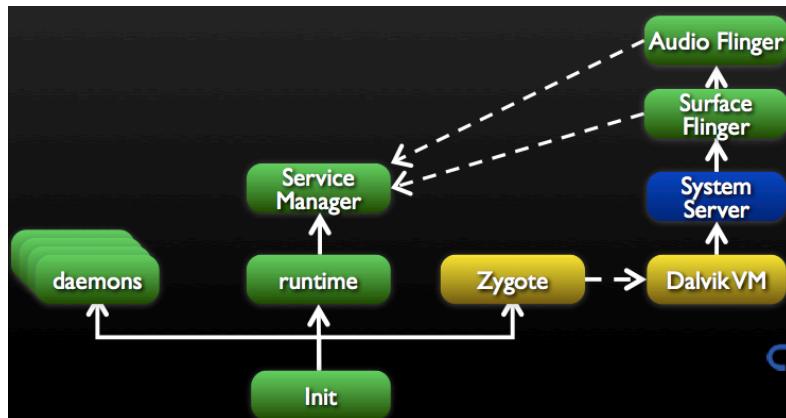
- Init process
 - Starts OS daemons
 - Zygote (to launch VM)
 - Service Manager (service registration)



42

Android Runtime

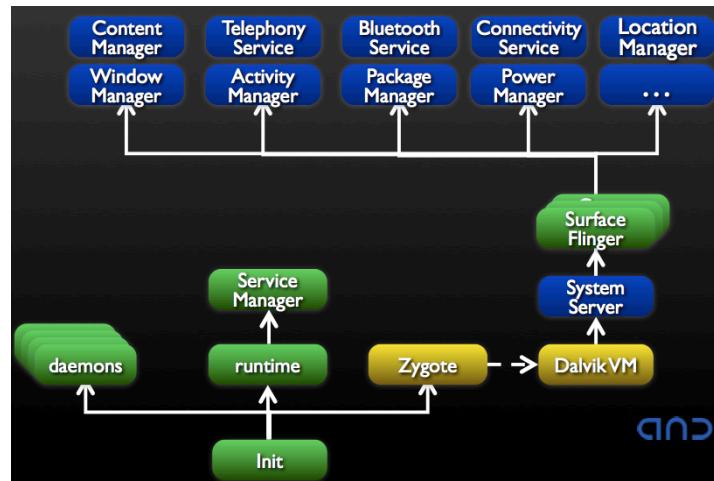
- Native System Services
 - Register for access via Binder IPC



43

Android Runtime

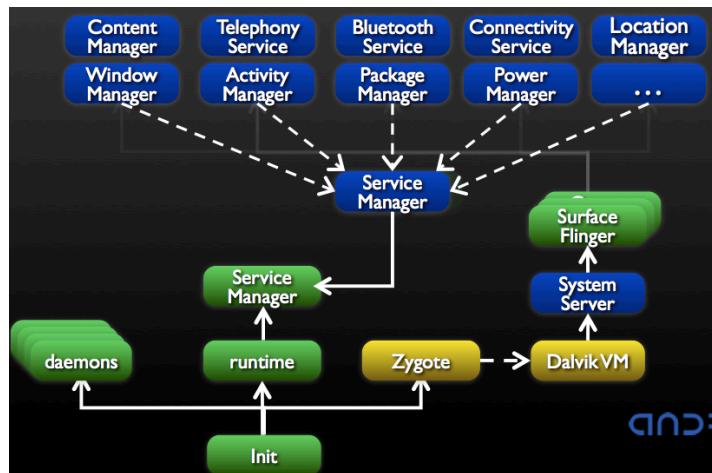
- Android-Managed Services



44

Android Runtime

- Android-Managed Services



45

Android Runtime

- System Processes



46

Android Runtime

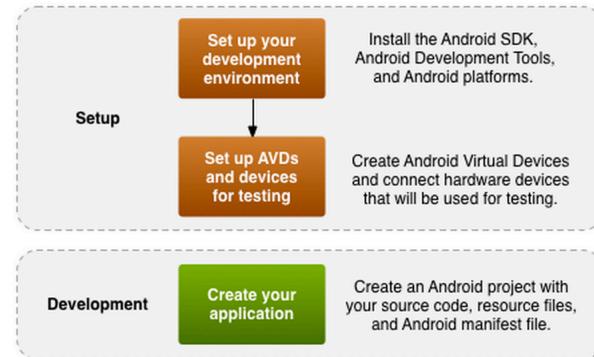
- System and App Processes



ANDROID DEVELOPMENT

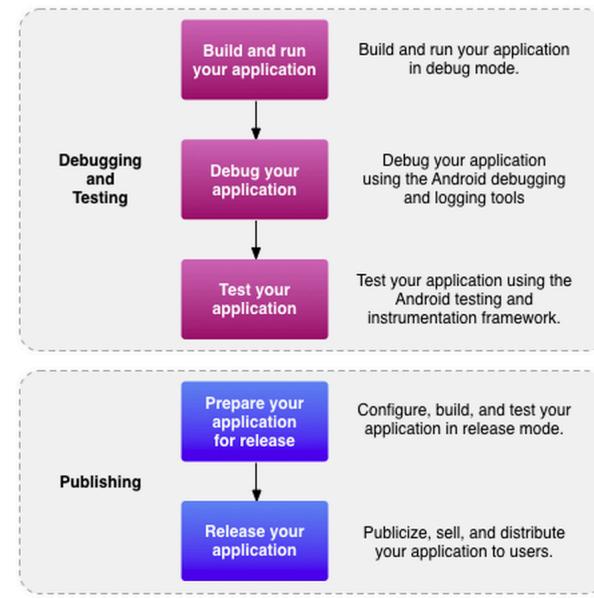
48

Development Process



49

Development Process



50

Programming Environment

- SDK available from Google
- Includes emulator
 - Not fully featured e.g. Bluetooth
- ~~Android Development Tools (ADT)~~
 - Eclipse plug-in
- Android Studio
 - IntelliJ-based

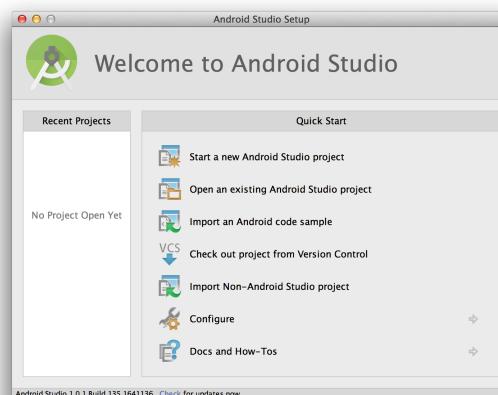


51

51

Android Studio

- Android Studio: <http://developer.android.com>
 - Better read the license agreement...
- Download & install
- SDK Manager: install Android platform(s)
- AVD Manager: create Android VM(s)



52

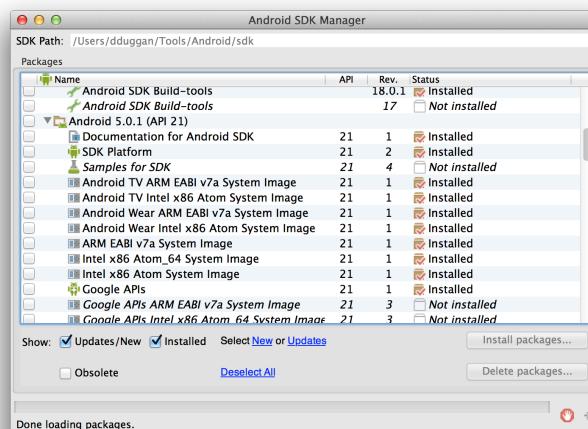
Android Studio

- Android Studio: <http://developer.android.com>
 - Better read the license agreement...
- Download & install
- SDK Manager: install Android platform(s)
- AVD Manager: create Android VM(s)



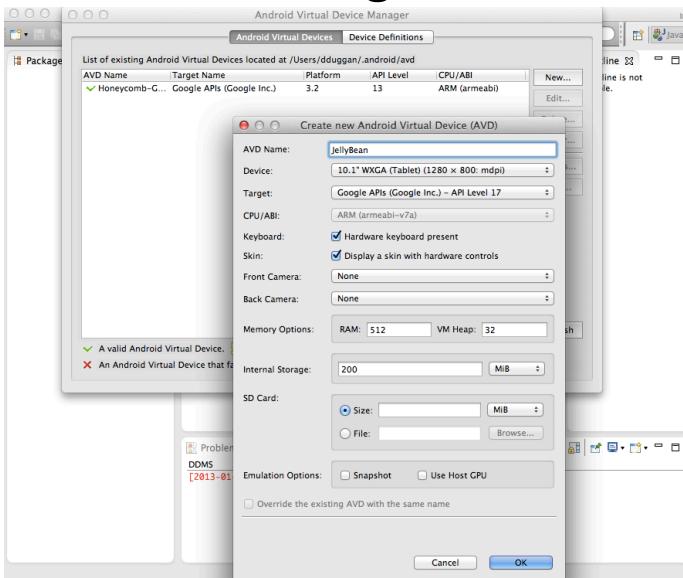
53

SDK Manager



54

AVD Manager



55

HELLO, WORLD

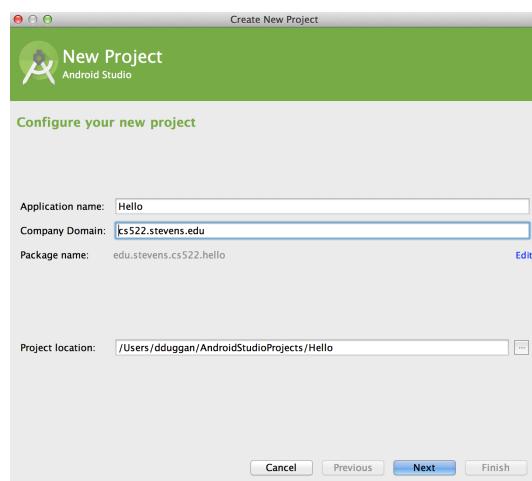
56

Hello, World



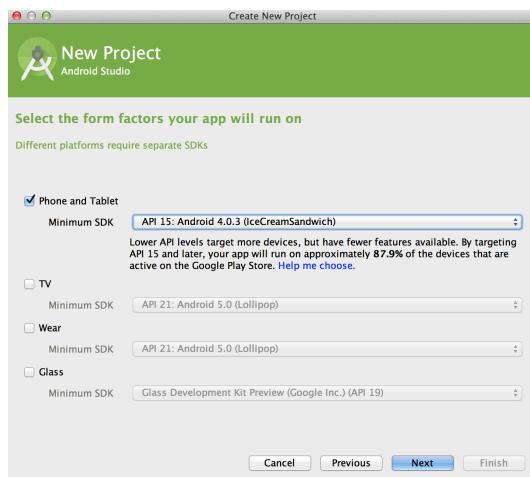
57

Hello, World



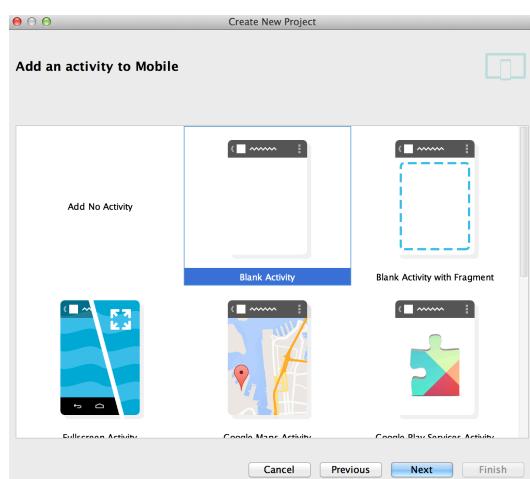
58

Hello, World

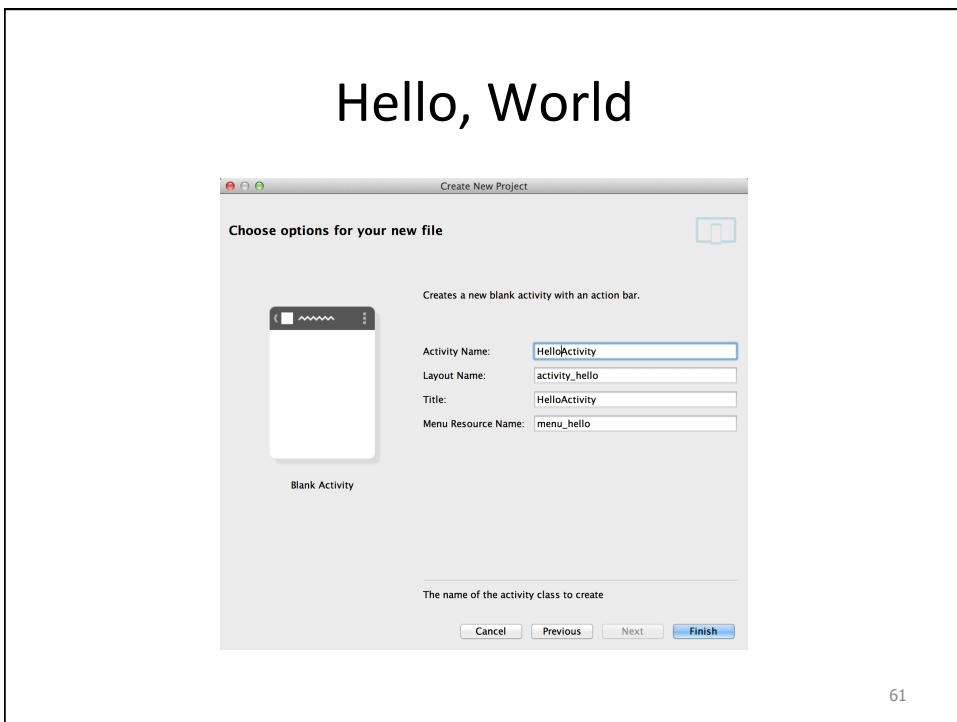


59

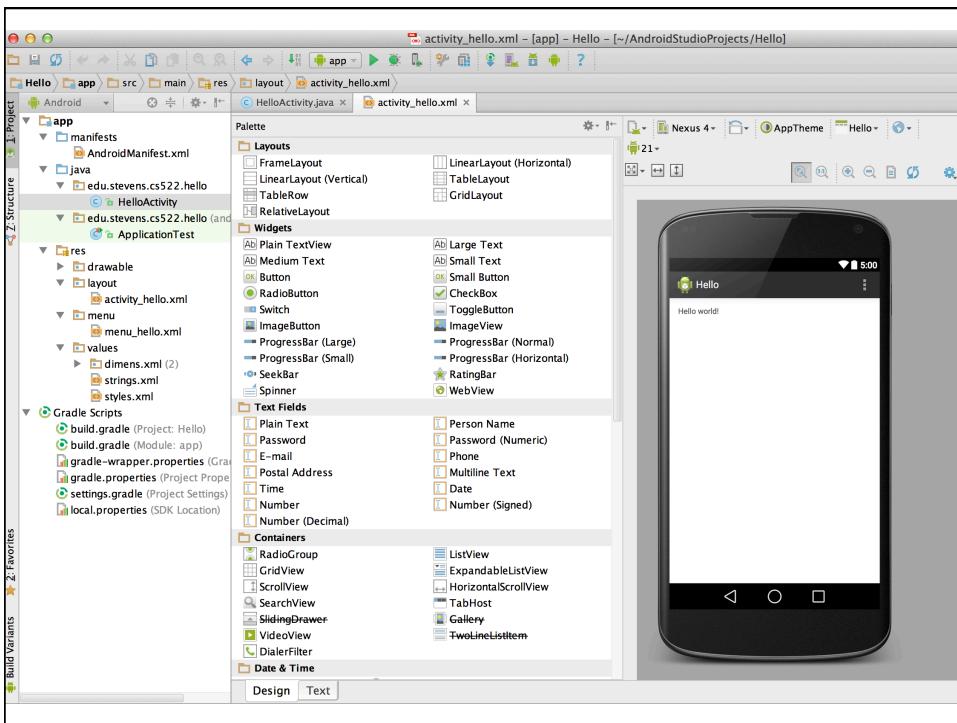
Hello, World



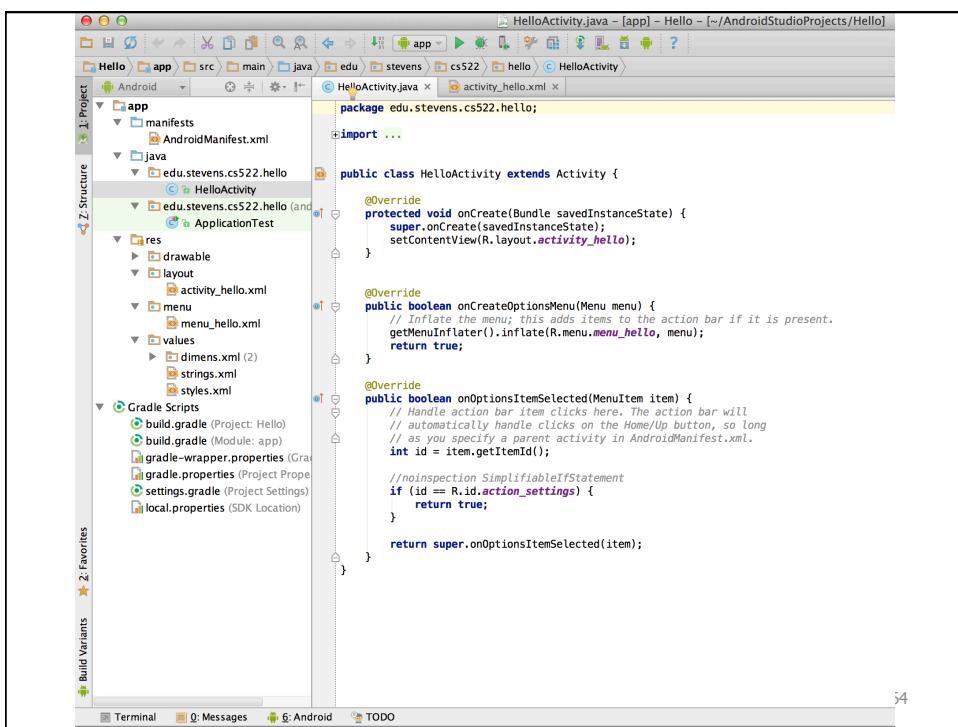
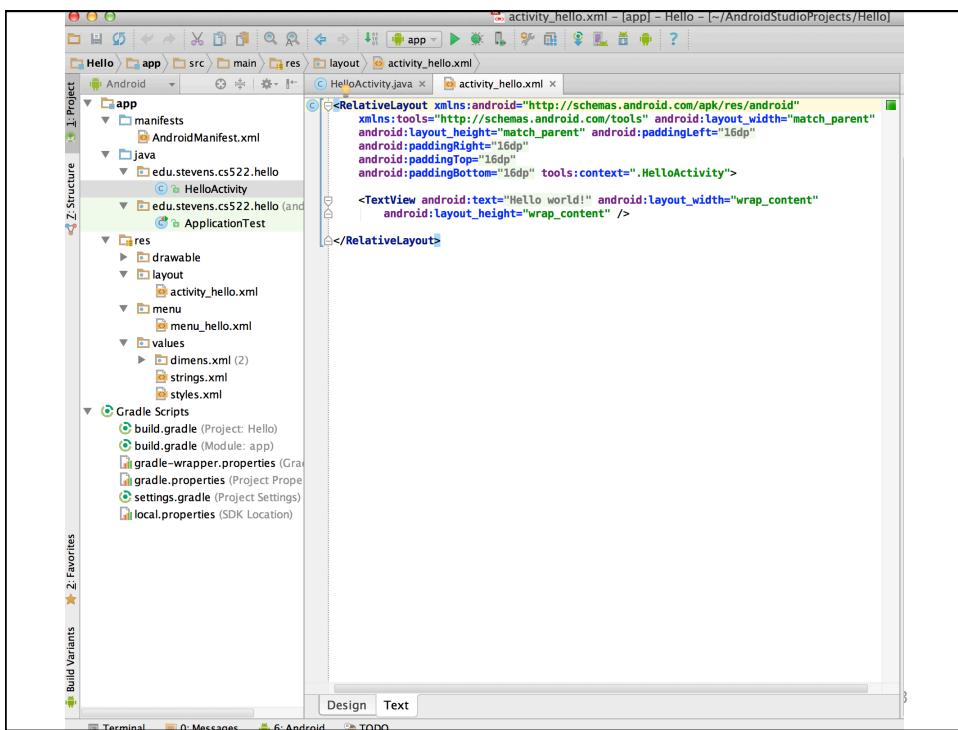
60



61



31



```
package edu.stevens.cs522.hello;

import android.app.Activity;
import android.os.Bundle;
import android.view.Menu;
import android.view.MenuItem;

public class HelloActivity extends Activity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_hello);
    }

    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        // Inflate the menu; this adds items to the action bar if it is present.
        getMenuInflater().inflate(R.menu.menu_hello, menu);
        return true;
    }

    @Override
    public boolean onOptionsItemSelected(MenuItem item) {...}
}
```

65

Programmatic UI

- Construct a TextView:

```
TextView tv = new TextView(this);
```

- Set string to display:

```
tv.setText("Hello, World");
```

- Tell system to display it:

```
setContentView(tv);
```

66

Layout-Based UI (Best Practice)

- Declare UI (`res/layout/activity_hello.xml`):

```
<TextView  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:layout_centerHorizontal="true"  
    android:layout_centerVertical="true"  
    android:text="@string/hello_world" />
```
- Declare message string (`res/values/strings.xml`):

```
<string name="hello_world">Hello world!</string>
```
- Tell system to display it:

```
setContentView(R.layout.activity_hello);
```

67

Layout

```
HelloActivity.java x  activity_hello.xml x  strings.xml x  
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"  
    xmlns:tools="http://schemas.android.com/tools" android:layout_width="match_parent"  
    android:layout_height="match_parent" android:paddingLeft="16dp"  
    android:paddingRight="16dp"  
    android:paddingTop="16dp"  
    android:paddingBottom="16dp" tools:context=".HelloActivity">  
    <TextView android:text="@string/hello_world" android:layout_width="wrap_content"  
        android:layout_height="wrap_content" />  
</RelativeLayout>
```

68

Layout & Strings

```
HelloActivity.java x activity_hello.xml x strings.xml x
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools" android:layout_width="match_parent"
    android:layout_height="match_parent" android:paddingLeft="16dp"
    android:paddingRight="16dp"
    android:paddingTop="16dp"
    android:paddingBottom="16dp" tools:context=".HelloActivity">

    <TextView android:text="@string/hello_world" android:layout_width="wrap_content"
        android:layout_height="wrap_content" />

</RelativeLayout>
```

```
HelloActivity.java x activity_hello.xml x strings.xml x
dit translations for all locales in the translations editor.
<?xml version="1.0" encoding="utf-8"?>
<resources>

    <string name="app_name">Hello</string>
    <string name="hello_world">Hello world!</string>
    <string name="action_settings">Settings</string>

</resources>
```

69

Layout & Strings

```
HelloActivity.java x activity_hello.xml x strings.xml x
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools" android:layout_width="match_parent"
    android:layout_height="match_parent" android:paddingLeft="16dp"
    android:paddingRight="16dp"
    android:paddingTop="16dp"
    android:paddingBottom="16dp" tools:context=".HelloActivity">

    <TextView android:text="@string/hello_world" android:layout_width="wrap_content"
        android:layout_height="wrap_content" />

</RelativeLayout>
```

```
HelloActivity.java x activity_hello.xml x
dit translations for all locales in the translations editor.
<?xml version="1.0" encoding="utf-8"?>
<resources>

    <string name="app_name">Hello</string>
    <string name="hello_world">Hello world!</string>
    <string name="action_settings">Settings</string>

</resources>
```



70

The screenshot shows the Android Studio interface with the project 'Hello' open. The left sidebar displays the project structure, including the app module with its Java, XML, and resources. The main editor window shows the HelloActivity.java code:

```
package edu.stevens.cs522.hello;

import android.app.Activity;
import android.os.Bundle;
import android.view.Menu;
import android.view.MenuItem;

public class HelloActivity extends Activity {

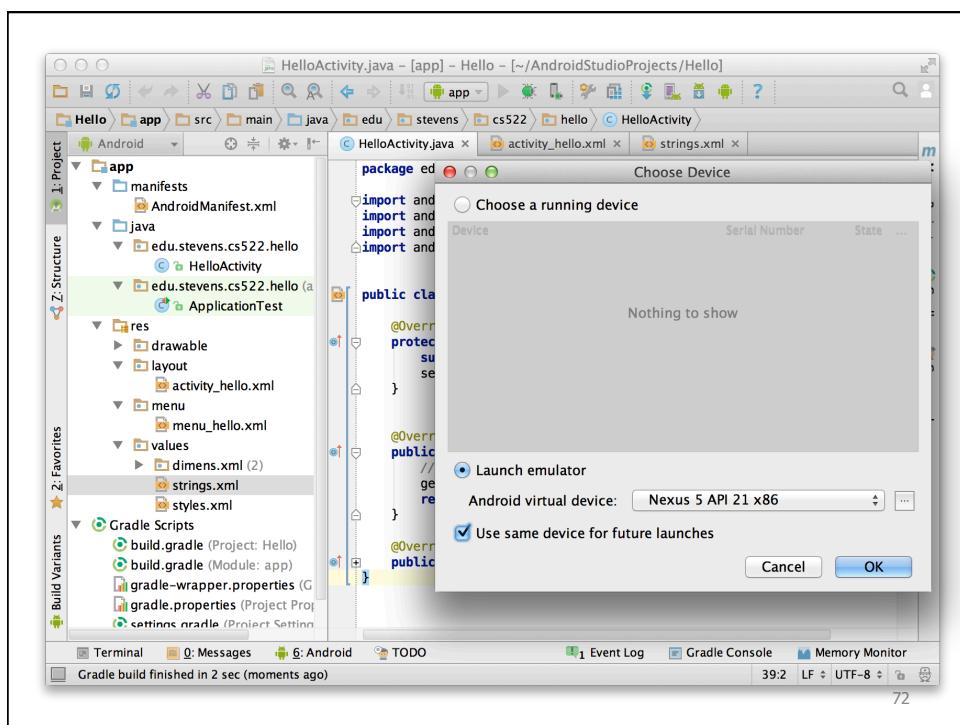
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_hello);
    }

    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        // Inflate the menu; this adds items to the action bar if it is present
        getMenuInflater().inflate(R.menu.menu_hello, menu);
        return true;
    }

    @Override
    public boolean onOptionsItemSelected(MenuItem item) { ... }
}
```

The code editor has syntax highlighting and code completion suggestions. The bottom status bar indicates a successful Gradle build.

71



72

The screenshot shows the Android Studio interface with the following details:

- Project Structure:** The project is named "Hello". The "app" module contains:
 - manifests:** AndroidManifest.xml
 - java:** edu.stevens.cs522.hello (selected), HelloActivity.java, ApplicationTest.java
 - res:** drawable, layout (activity_hello.xml), menu (menu_hello.xml), values
- Code Editor:** The HelloActivity.java file is open, showing Java code for an Activity. The line `setContentView(R.layout.activity_hello);` is highlighted.
- Build Log:** The "Run" tab shows the command run: `/Users/ddugan/Tools/Android/sdk/tools/emulator -avd Nexus_5_API_21_x86 -netspeed full -netdelay none`. The log output includes parameters for creating a filesystem, such as size, block size, and journal blocks.
- Bottom Bar:** Shows the status "Gradle build finished in 2 sec (a minute ago)" and the time "15:1".

73

The screenshot shows the Android Studio interface with the following details:

- Project Structure:** The project is named "Hello". The "app" module contains:
 - manifests:** AndroidManifest.xml
 - java:** edu.stevens.cs522.hello (selected), HelloActivity.java, ApplicationTest.java
 - res:** drawable, layout (activity_hello.xml), menu (menu_hello.xml), values
- Code Editor:** The HelloActivity.java file is open, showing Java code for an Activity. The line `setContentView(R.layout.activity_hello);` is highlighted.
- Logcat:** The "logcat" tab is selected, showing log entries from the emulator. The log entries include:
 - 01-21 20:10:01.121 2094-2111/edu.stevens.cs522.hello W/OpenGLRenderer: Failed
 - 01-21 15:10:05.507 2094-2111/edu.stevens.cs522.hello W/EGL_emulation: eglSurface
 - 01-21 15:10:05.507 2094-2111/edu.stevens.cs522.hello W/OpenGLRenderer: Failed
- Bottom Bar:** Shows the status "Gradle build finished in 2 sec (3 minutes ago)" and the time "11:14".

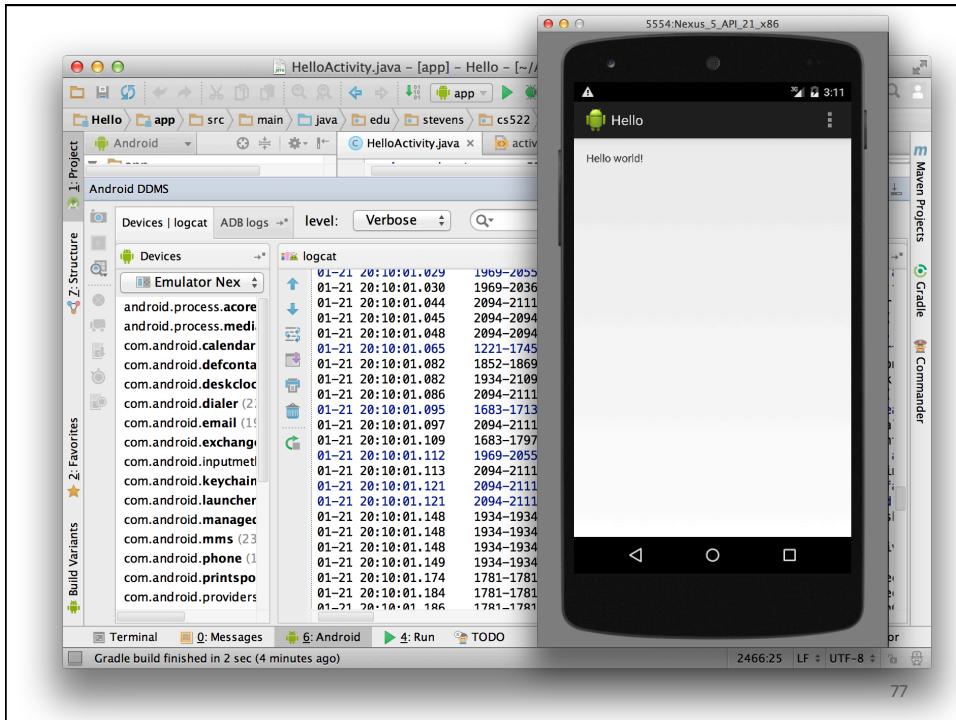
74

01-21 20:10:00.909 2094-2094/edu.stevens.cs522.hello I/art: Not late-enabling
 01-21 20:10:01.044 2094-2111/edu.stevens.cs522.hello D/OpenGLRenderer: Render
 01-21 20:10:01.045 2094-2094/edu.stevens.cs522.hello D/: HostConnection::get(
 01-21 20:10:01.048 2094-2094/edu.stevens.cs522.hello D/Atlas: Validating map.
 01-21 20:10:01.086 2094-2111/edu.stevens.cs522.hello D/: HostConnection::get(
 01-21 20:10:01.097 2094-2111/edu.stevens.cs522.hello I/OpenGLRenderer: Initia
 01-21 20:10:01.113 2094-2111/edu.stevens.cs522.hello D/OpenGLRenderer: Enabli
 01-21 20:10:01.121 2094-2111/edu.stevens.cs522.hello W/EGL_emulation: eglSurf
 01-21 20:10:01.121 2094-2111/edu.stevens.cs522.hello W/OpenGLRenderer: Failed
 01-21 15:10:05.507 2094-2111/edu.stevens.cs522.hello W/EGL_emulation: eglSurf
 01-21 15:10:05.507 2094-2111/edu.stevens.cs522.hello W/OpenGLRenderer: Failed

75

01-21 20:10:01.029 1969-2055/com.google.process.location W/Settings: Setting :
 01-21 20:10:01.030 1969-2036/com.google.process.location I/GCoreURL: Unbound
 01-21 20:10:01.044 2094-2111/edu.stevens.cs522.hello D/OpenGLRenderer: Render
 01-21 20:10:01.045 2094-2094/edu.stevens.cs522.hello D/: HostConnection::get(
 01-21 20:10:01.048 2094-2094/edu.stevens.cs522.hello D/Atlas: Validating map.
 01-21 20:10:01.065 1221-1745/system_process W/ActivityManager: Unable to star
 01-21 20:10:01.082 1852-1869/com.android.launcher I/art: Background sticky co
 01-21 20:10:01.082 1934-2189/com.android.exchange I/Exchange: RestartPingTask
 01-21 20:10:01.086 2094-2111/edu.stevens.cs522.hello D/: HostConnection::get(
 01-21 20:10:01.095 1683-1713/android.process.acore W/art: Suspending all thre
 01-21 20:10:01.097 2094-2111/edu.stevens.cs522.hello I/OpenGLRenderer: Initia
 01-21 20:10:01.109 1683-1797/android.process.acore I/ContactsFTS: Rebuild con
 01-21 20:10:01.112 1969-2055/com.google.process.location W/Settings: Setting :
 01-21 20:10:01.113 2094-2111/edu.stevens.cs522.hello D/OpenGLRenderer: Enabli
 01-21 20:10:01.121 2094-2111/edu.stevens.cs522.hello W/EGL_emulation: eglSurf
 01-21 20:10:01.121 2094-2111/edu.stevens.cs522.hello W/OpenGLRenderer: Failed
 01-21 20:10:01.148 1934-1934/com.android.exchange I/Exchange: RestartPingsTas
 01-21 20:10:01.148 1934-1934/com.android.exchange I/Exchange: PSS stopIdle
 01-21 20:10:01.149 1934-1934/com.android.exchange I/Exchange: PSS has no acti
 01-21 20:10:01.174 1934-1934/com.android.exchange I/Exchange: onDestory
 01-21 20:10:01.174 1781-1781/com.google.android.gms I/SystemUpdateService: re
 01-21 20:10:01.184 1781-1781/com.google.android.gms I/SystemUpdateService: re
 01-21 20:10:01.186 1781-1781/com.google.android.gms D/SystemUpdateService: on

76



77

Learning Android Studio

The screenshot shows a web browser window with the URL <https://developer.android.com/tools/studio/index.html>. The page title is "Android Studio Overview". The left sidebar contains a navigation menu with sections like "Download", "Android Studio", "Tips and Tricks", "Workflow", "Tools Help", "Build System", "Support Library", "Revisions", "NDK", "ADK", and "Eclipse with ADT". The main content area features a large heading "Android Studio Overview" and a detailed description of the IDE's features, including its integration with IntelliJ IDEA and various development tools. A sidebar on the right provides links to "IN THIS DOCUMENT" and "SEE ALSO".

78

Your First Assignment

The screenshot shows a web browser window with the title "Building Your First App" at the top. The URL in the address bar is <https://developer.android.com/training/basics/firstapp/index.html>. The left sidebar contains a navigation tree under "Getting Started" with "Building Your First App" expanded, showing sub-sections like "Creating an Android Project", "Running Your Application", and "Starting Another Activity". The main content area has a heading "Building Your First App" and a sub-section "Set Up Your Environment". It includes instructions for setting up the development environment, a note about using Android Studio or the command line, and a "GET STARTED" button. The bottom of the page shows the URL again and a "g+ 2.2" link.

79