

Android Studio... for Eclipse Users

Copyright © 2015 CommonsWare, LLC



The State of Eclipse

- Officially Disowned by Android Tools Team
 - ADT and such still available for download
 - ADT is not in especially good health, since has not been updated in many months
- Andmore Project
 - Eclipse Foundation
 - MOTODEV Studio bits plus other modern touches
 - Very much a work in progress



The State of Android Studio

- For Realz, Now
 - 1.0 in late 2014
 - The official IDE for Android, from the Tools team
- Net: Eclipse Users Need to Have a Plan
 - Migrate to Android Studio
 - Hold out for Andmore
 - Don't breathe, lest the vibrations somehow dislodge a critical bit and break a classic ADT setup...



Getting Going with Studio

- Download from Android Developer site
 - Options to automatically download and install SDK, initial emulator image, HAXM (Windows-only?)
- How to Get Updates
 - Welcome Dialog (tiny footer)
 - Help menu
 - Automatic checks on your desired channel
 - Default = stable





Welcome to Android Studio

Recent Projects

VideoBrowse

VideoBrowse

Pager

~/stuff/CommonsWare/projects/CWAC/Pager

My Application

/tmp/MyApplication

Decktastic

Decktastic

Provider

~/stuff/CommonsWare/projects/CWAC/Provider

Relative

Relative

VideoList

VideoList

MediaRouter

MediaRouter

Quick Start



Start a new Android Studio project



Open an existing Android Studio project



Import an Android code sample



Check out project from Version Control



Import Non-Android Studio project



Configure



Docs and How-Tos



Workspace, We Hardly Knew Ye

- Eclipse: Workspace and Projects
 - Workspace encapsulates plugins and kin
 - Projects did not have sub-projects
- Android Studio: Projects and Modules
 - Module == sub-project
 - No workspace, so each IDE window represents a project and its modules
 - Net: may wind up with a few IDE windows open at once



Escape From Eclipse

- Importing a Project
 - File > Import Project from main menu
- Import Types
 - If build.gradle exists, just opens the project and adds Android Studio files (.iml, .idea/)
 - If no build.gradle but has legacy project directory structure, copies and remodels



Escape from Eclipse

- Remodeling Work
 - Creates a new directory (at location you designate)
 - Sets up Studio and Gradle files
 - Puts your app code in an app/ module
 - Mostly in a src/main/ sourceset within there
 - Gives you a report of what moved where



Project Views: Old and New

- Android Project View
 - What you get by default when creating or importing an Android project
 - Synthetic merging of structures, notably resources
 - Benefit: maybe easier for you to find all variants of a resource
 - Cost: does not reflect actual filesystem

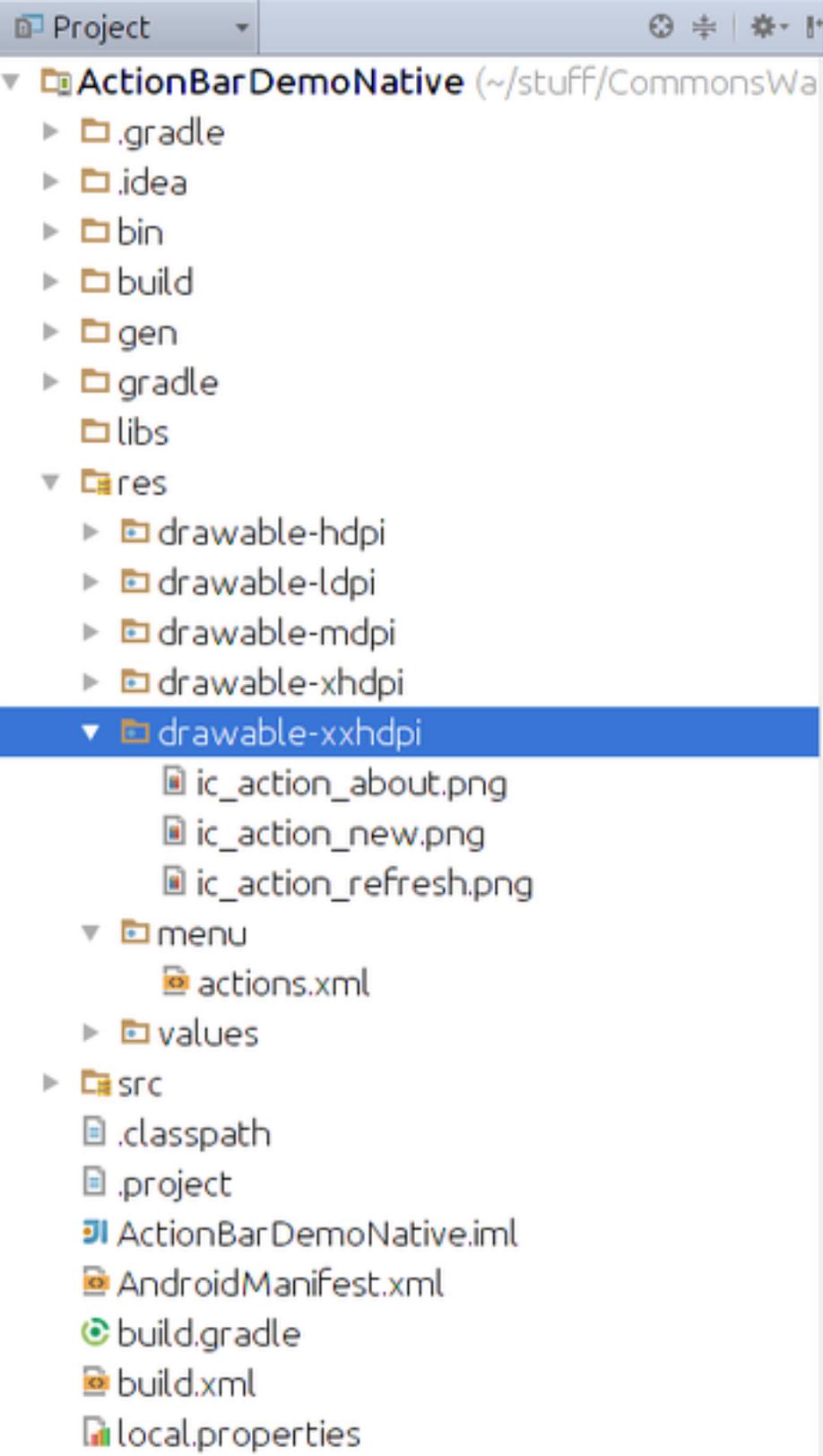


- ▼ **ActionBarDemoNative**
 - ▼ manifests
 - AndroidManifest.xml
 - ▼ java
 - ▼ com.commonware.android.inflation
 - ActionBarDemoActivity
 - ▼ res
 - ▼ drawable**
 - ▶ ic_action_about.png (4)
 - ▶ ic_action_new.png (4)
 - ▶ ic_action_refresh.png (4)
 - ▼ ic_launcher.png (4)
 - ic_launcher.png (hdpi)
 - ic_launcher.png (ldpi)
 - ic_launcher.png (mdpi)
 - ic_launcher.png (xhdpi)
 - ▶ menu
 - ▶ values
 - ▼ Gradle Scripts
 - build.gradle (Module: ActionBarDemoNative)
 - gradle-wrapper.properties (Gradle Version)
 - local.properties (SDK Location)

Project Views: Old and New

- Classic Project View
 - Switch drop-down above file tree to “Project”
 - Classic “these are the files in the filesystem” sort of view
 - Benefit: reflects reality
 - Cost: more tedious to find all variants of a resource





What You Don't Get

- Structured Editors
 - Only the drag-and-drop GUI builder plus a dubiously-useful Project Structure dialog
 - Resources (menus, strings, etc.) edit via raw XML
- Directly Integrated DDMS
 - Some DDMS features are baked into Android Studio, such as process list and LogCat
 - Others you will access via Android Device Monitor



Project Structure Dialog

- The Theory
 - Allows you to configure project settings
 - Reminiscent of Project > Properties > Android in Eclipse
 - Automatically updates Gradle build files that drive the build process



Project Structure Dialog

- The Reality
 - Works for simple projects
 - Starts to break down once you significantly tailor the Gradle build files, particularly with scripting
 - Example: automatically generating versionCode
 - Only really handle static properties
 - Does not handle everything that Gradle offers
 - Example: splits





SDK Location

Project

— Modules

app

SDK Location

Android SDK location:

The directory where the Android SDK is located. This location will be used for new projects, and for existing projects that do not have a local.properties file with a sdk.dir property.

**JDK location:**

The directory where the Java Development Kit (JDK) is located.



OK

Cancel

Apply

Project Structure

+ -

- SDK Location
- Project
- Modules
 - app

Gradle version	<input type="text" value="2.2.1"/>
Android Plugin Version	<input type="text" value="1.0.0"/>
Android Plugin Repository	<input type="text" value="jcenter"/>
Default Library Repository	<input type="text" value="jcenter"/>

OK Cancel Apply



- SDK Location
- Project
- Modules
- app

Compile Sdk Version	API 21: Android 5.0 (Lollipop)
Build Tools Version	21.1.2
Library Repository	
Ignore Assets Pattern	
Incremental Dex	
Source Compatibility	
Target Compatibility	

Project Structure

Properties Signing **Flavors** Build Types Dependencies

- SDK Location
- Project
- Modules
 - app

defaultConfig	Name:	defaultConfig
	Min Sdk Version	API 14: Android 4.0 (IceCreamSandwich)
	Application Id	com.commonware.myapplication
	Proguard File	<input type="text"/> ...
	Signing Config	<input type="text"/> ▼
	Target Sdk Version	API 21: Android 5.0 (Lollipop)
	Test Instrumentation Runner	<input type="text"/>
	Test Application Id	<input type="text"/>
	Version Code	1
	Version Name	1.0

OK Cancel Apply

Modules

- Scenarios
 - Public library with demo apps
 - App with private libraries
 - Forks of existing OSS libraries)
 - Isolating pure Java code for direct testing
- Adding a Module
 - File > New Module
 - Wizard UI



Running a Project

- Step #1: Pick Your Build Variant
 - Build type
 - Product flavor
- Step #2: Pick Your Run Configuration
 - Get one per app module “out of the box”, available in drop-down to the left of the green “run” toolbar button
 - Can add others (e.g., unit tests)



Build Variants



Module	Build Variant
 app 	debug



 Build Variants

Build Variants



Module

Build Variant

 app



debug

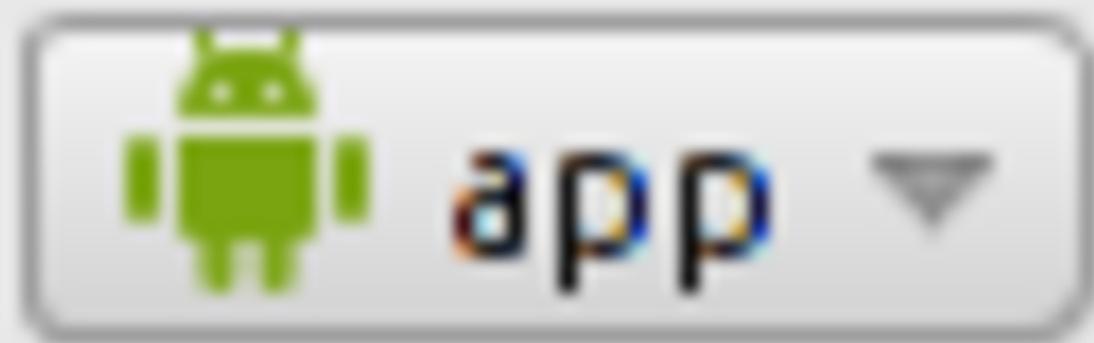
debug

release

Running a Project

- Step #3: Press that Run Toolbar Button
 - Build errors? They'll show up in Gradle Console
- Step #4: Choose a Device or Emulator
 - Existing connection
 - Start a fresh emulator instance
- Step #5: Run Successfully!
 - ...or die trying, using LogCat to see what went wrong







Choose Device

Choose a running device

Device	Serial Number	State	C...
Nothing to show			

Launch emulator

Android virtual device:

4.3-WVGA



Use same device for future launches

OK

Cancel

AVD Manager

- The New One
 - Facelift
 - Not all options from old AVD Manager available
 - And not everything configurable about an AVD was available in the old AVD Manager either...
- The Old One
 - Still available from command line, via **android avd**





Your Virtual Devices

Android Studio

Type	Name	Resolution	API	Target	CPU/ABI	Size on Disk	Actions
	Nexus 5 API 21 x86	1080 x 1920: xxhdpi	21	Google APIs	x86	750 MB	

[+ Create Virtual Device...](#)

OK

Cancel

DDMS and LogCat

- Android View
 - Docked by default in left side of lower edge
- Devices List
 - Drop-down of visible devices and running emulators
 - List of debuggable processes for selected device
 - Screenshot, video recording, process termination, GC, heap dumps, start/stop method tracing, start/stop allocation tracking



Android DDMS

Devices | logcat

ADB logs →



 Devices →

 Emulator 4.3-WVGA Ar ▼

com.commonware.empubl

android.process.acore (1343)

android.process.media (1308)

com.android.calendar (1568)

com.android.deskclock (149)

com.android.dialer (1523)

 4: Run

 TODO

 6: Android

DDMS and LogCat

- LogCat
 - Now a text area, not a scrolling list
 - Net: selection is an arbitrary chunk of text, not strictly by lines
 - Options to print and delete
 - Filtering
 - Automatic by last running app, though seems buggy
 - By log level
 - By arbitrary text
 - No save option – use clipboard



Log level: Verbose

Q*

No Filters

logcat

```
01-06 10:29:51.701 1138-1275/system_process D/Nat464Xlat: requiresClat: netType=5, hasIPv4Address=true
01-06 10:29:52.301 1138-1517/system_process D/ConnectivityService: [CheckMp] isMobileOk: timed out
01-06 10:29:52.301 1138-1517/system_process D/ConnectivityService: [CheckMp] isMobileOk: F stop hipri
01-06 10:29:52.301 1138-1517/system_process D/ConnectivityService: [CheckMp] isMobileOk: X result=0
01-06 10:29:52.301 1138-1157/system_process D/ConnectivityService: [CheckMp] onPostExecute: result=0
01-06 10:29:52.301 1138-1157/system_process D/ConnectivityService: CheckMp.onComplete: result=0
01-06 10:29:52.301 1138-1157/system_process D/ConnectivityService: CheckMp.onComplete: ignore, connected or no c
01-06 10:29:54.721 1138-1277/system_process D/Tethering: MasterInitialState.processMessage what=3
01-06 10:29:54.721 1138-1277/system_process D/Tethering: MasterInitialState.processMessage what=3
```

Standard IDE Stuff

- Debugger
 - Breakpoints in left column
 - Standard controls for stepping through code
- Refactoring, version control, navigation, etc.
- Mostly Shared with IntelliJ IDEA
 - Need to blend IDEA documentation in with Android Studio-specific material for complete picture



FAQs

- How Do I Change My Build Target?
 - Eclipse: Project > Properties > Android
 - Studio: Project Structure or build.gradle
- Where Did My <uses-sdk> Element Go?
 - See minSdkVersion and targetSdkVersion
 - defaultConfig in Project Structure or build.gradle
 - Can be overridden in product flavors
 - General: stuff that can be in Gradle or in manifest frequently is defined in Gradle



FAQs

- How Do I Mark a Project as a Library Project?
 - Eclipse: Project > Properties > Android
 - Studio
 - Set it as a library module at the outset
 - `com.android.library` plugin instead of `com.android.application`

