



Google
Developers



The New Android SDK Build System

google.com/+XavierDucrohet
Android SDK Tech Lead

Why A New Build System?

- Extensibility / Customization / Stable APIs
- Unified across CI Servers & IDEs
- Standard & Advanced Features



Why Gradle?

- Powerful Build System
 - Declarative, Flexible
 - Imperative customization
 - Tooling API
- Build System Toolkit
 - Plugins create own DSL, APIs, IDE integration
- Free / Open Source





Gradle Basics

Simple Java Project

```
apply plugin: 'java'
```

build.gradle



Simple Android Project

```
buildscript {  
    repositories { mavenCentral() }  
    dependencies { classpath 'com.android.tools.build:gradle:0.4' }  
}  
  
apply plugin: 'android'  
  
android {  
    compileSdkVersion 17  
    buildToolsVersion '16.1'  
}
```

build.gradle



Simple Android Project

```
buildscript {  
    repositories { mavenCentral() }  
    dependencies { classpath 'com.android.tools.build:gradle:0.4' }  
}
```

build.gradle

```
apply plugin: 'android'
```

```
android {  
    compileSdkVersion 17  
    buildToolsVersion '16.1'  
}
```



Simple Android Project

```
buildscript {  
    repositories { mavenCentral() }  
    dependencies { classpath 'com.android.tools.build:gradle:0.4' }  
}
```

build.gradle

```
apply plugin: 'android'
```

```
android {  
    compileSdkVersion 17  
    buildToolsVersion '16.1'  
}
```



Simple Android Project

```
buildscript {  
    repositories { mavenCentral() }  
    dependencies { classpath 'com.android.tools.build:gradle:0.4' }  
}
```

```
apply plugin: 'android'
```

```
android {  
    compileSdkVersion 17  
    buildToolsVersion '16.1'  
}
```

build.gradle



Conventions: Source Folders

src/main/java
src/main/resources

src/test/java
src/test/resources



Conventions: Source Folders

`src/main/AndroidManifest.xml`

`src/main/java`

`src/main/resources`

`src/main/res`

`src/main/assets`

`src/main/aidl`

`src/main/rs`

`src/main/jni`

`src/instrumentTest/java`

`src/instrumentTest/resources`

`src/instrumentTest/res`

`src/instrumentTest/assets`

`src/instrumentTest/aidl`

`src/instrumentTest/rs`

`src/instrumentTest/jni`



Configuration For Older Projects

build.gradle

```
android {
    sourceSets {
        main {
            manifest.srcFile 'AndroidManifest.xml'
            java.srcDirs ['src']
            res.srcDirs ['res']
            ...
        }
        instrumentTest.setRoot('tests')
    }
}
```



Configuration For Older Projects

build.gradle

```
android {  
    sourceSets {  
        main {  
            manifest.srcFile 'AndroidManifest.xml'  
            java.srcDirs ['src']  
            res.srcDirs ['res', 'res2']  
            ...  
        }  
        instrumentTest.setRoot('tests')  
    }  
}
```



Gradle Tasks

```
$ cd myproject/
```

```
$ gradle <taskName1> [<taskName2>...]
```

```
$ gradle tasks [--all]
```



Gradle Tasks

assemble

check

build

→ assemble + check

clean



Gradle Tasks

assemble

jar

check

test

build

→ assemble + check

clean



Gradle Tasks

assemble

jar

check

test

findBugsMain

findBugsTest

build

→ assemble + check

clean



Gradle Tasks

assemble
jar

check
test
findBugsMain
findBugsTest

build
→ assemble + check

clean

assemble

check
connectedCheck
deviceCheck

build
→ assemble + check

clean



Gradle Tasks

assemble

jar

check

test

findBugsMain

findBugsTest

build

→ assemble + check

clean

assemble

assembleDebug + assembleRelease

check

connectedCheck

deviceCheck

build

→ assemble + check

clean





Customization

Basic Customization

```
android {  
    defaultConfig {  
        versionCode 12  
        versionName '2.0'  
  
        minSdkVersion 14  
        targetSdkVersion 17  
    }  
}
```

build.gradle



Basic Customization

```
def getVersionCode() {  
    def code = ...  
    return code  
}  
  
android {  
    defaultConfig {  
        versionCode getVersionCode()  
        ...  
    }  
}
```

build.gradle



Build Types

- Build / Packaging customization
 - debuggable flag
 - ProGuard
 - signing configuration
 - source / resources overlay
- debug and release prebuilt



Build Types: DSL

```
android {  
    buildTypes {  
        debug {  
            packageNameSuffix '.debug'  
            versionNameSuffix '-DEBUG'  
        }  
        beta {  
            ...  
        }  
    }  
}
```

build.gradle



Build Types: Source Folders

src/main/AndroidManifest.xml

src/main/java

src/main/resources

src/main/res

src/main/assets

src/main/aidl

src/main/rs

src/main/jni

src/debug/...

src/release/...

src/beta/...

src/instrumentTest/java

src/instrumentTest/resources

src/instrumentTest/res

src/instrumentTest/assets

src/instrumentTest/aidl

src/instrumentTest/rs

src/instrumentTest/jni



Build Types: Tasks

assemble

check

build

→ assemble + check

clean

assemble

assembleDebug + assembleRelease +
assembleBeta

check

connectedCheck

deviceCheck

build

→ assemble + check

clean



Signing Configuration

build.gradle

```
android {  
    signingConfigs {  
        debug {  
            storeFile file('debug.keystore')  
        }  
    }  
  
    // buildTypes.debug already configured to  
    // use signingConfigs.debug  
}
```



Debuggable Release Builds

build.gradle

```
android {  
    buildTypes {  
        debugRelease.initWith(buildTypes.release)  
        debugRelease {  
            debuggable true  
            packageNameSuffix '.debugrelease'  
            signingConfig signingConfigs.debug  
        }  
    }  
    sourceSets.debugRelease.setRoot('src/release')  
}
```



Manipulating the Model

```
android {  
    ...  
}  
  
android.buildTypes.each { type ->  
    type.packageNameSuffix = ".$type.name"  
}
```

build.gradle





Dependencies

Dependency Scopes: Java Projects

compile

runtime

testCompile

testRuntime



Dependency Scopes: Android Projects

compile

runtime

testCompile

testRuntime

compile

package

instrumentTestCompile

instrumentTestPackage



Dependency Scopes: Android Projects

compile

runtime

testCompile

testRuntime

compile

package

instrumentTestCompile

instrumentTestPackage

debugCompile/Package

releaseCompile/Package

betaCompile/Package



Local Dependencies

```
dependencies {  
    compile fileTree(dir: 'libs', include: '*.jar')  
}
```

```
android {  
    ...  
}
```

build.gradle



Remote Dependencies

```
repositories { mavenCentral() }  
  
android {  
    ...  
}  
  
dependencies {  
    instrumentTestCompile 'com.squareup:fest-android:1.0.4'  
    betaCompile file('libs/protobuf.jar')  
}
```

build.gradle



Multi-Project Setup

```
MyProject/  
| settings.gradle  
+ app/  
  | build.gradle  
+ libraries/  
  + lib1/  
    | build.gradle  
  + lib2/  
    | build.gradle
```

settings.gradle

```
include ':app'  
include ':libraries:lib1'  
include ':libraries:lib2'
```



Project Dependencies

```
dependencies {  
    compile project(':libraries:lib1')  
}  
  
android {  
    ...  
}
```

build.gradle



Android Library Project

```
buildscript {  
    repositories { mavenCentral() }  
    dependencies { classpath 'com.android.tools.build:gradle:0.4' }  
}  
  
apply plugin: 'android-library'  
  
android {  
    compileSdkVersion 17  
    buildToolsVersion '16.1'  
}
```

build.gradle



Android Library

- Binary Bundle (.aar)
 - Uploadable to repositories
- Support for
 - assets
 - ProGuard rules
 - Custom Lint Rules
 - ...





Build Variants

Product Flavors

- Different versions of the same application
 - Paid vs Free
 - Multi-APK support in Google Play
- Different Dimension(s) than Build Types



Product Flavors DSL

build.gradle

```
android {  
    productFlavors {  
        flavor1 {  
            packageName 'com.example.flavor1'  
            minSdkVersion 14  
        }  
        flavor2 {  
            packageName 'com.example.flavor2'  
        }  
    }  
}
```



Product Flavors: Source Folders

src/main/AndroidManifest.xml

src/main/java

src/main/resources

src/main/res

src/main/assets

src/main/aidl

src/main/rs

src/main/jni

src/debug/...

src/release/...

src/beta/...

src/Flavor1/...

src/instrumentTest/java

src/instrumentTest/resources

src/instrumentTest/res

src/instrumentTest/assets

src/instrumentTest/aidl

src/instrumentTest/rs

src/instrumentTest/jni

src/instrumentTestFlavor1/...



Dependency Scopes: Product Flavors

compile

runtime

testCompile

testRuntime

compile

package

instrumentTestCompile

instrumentTestPackage

debugCompile/Package

releaseCompile/Package

betaCompile/Package

flavor1Compile/Package

instrumentTestFlavor1Compile/Package



Variant = Build Type + Product Flavor(s)

No Flavors:

	debug	release
-	debug	release
-	-	-



Variant = Build Type + Product Flavor(s)

No Flavors:

	debug	release
	debug	release
	-	-

With Flavors:

	debug	release
Flavor1	Flavor1Debug	Flavor1Release
Flavor2	Flavor2Debug	Flavor2Release



Multi-Flavors DSL

build.gradle

```
android {  
    flavorGroups 'abi', 'version'  
    productFlavors {  
        arm {  
            flavorGroup 'abi'  
        }  
        paid {  
            flavorGroup 'version'  
            packageName 'com.example.flavor2'  
        }  
    }  
}
```



Variant = Build Type + Product Flavors

		debug	release
paid	arm	paid-arm-debug	paid-arm-release
paid	x86	paid-x86-debug	paid-x86-release
paid	mips	paid-mips-debug	paid-mips-release
free	arm	free-arm-debug	free-arm-release
free	x86	free-x86-debug	free-x86-release
free	mips	free-mips-debug	free-mips-release



Build Variants: Source Code

Multiple Source Folders, Single Output

```
src/main/java  
src/free/java  
src/mips/java  
src/debug/java
```



Build Variants: Resources & Assets

Overlays

src/debug/res



src/paid/res



src/x86/res



src/main/res



Build Variants: Signing Configuration

Priority Order

`android.buildTypes.release.signingConfig`



`android.productFlavors.paid.signingConfig`



`android.productFlavors.arm.signingConfig`



`android.defaultConfig.signingConfig`



Build Variants: Package Name

Overlays + Suffix

`android.productFlavors.paid.packageName`



`android.productFlavors.x86.packageName`



`android.defaultConfig.packageName`



`src/main/AndroidManifest.xml`

+ `android.buildTypes.debug.packageNameSuffix`





Testing

Testing: Recap

- Test Source Sets: `src/instrumentTest`, `src/instrumentTestFlavor1`
- Dependency Scopes: `instrumentTestCompile`, `instrumentTestFlavor1Compile`
- Generated `AndroidManifest.xml`
- Only one Build Type is tested
- Library embedded into Test APK



Testing Scenarios

- check
- connectedCheck
- deviceCheck



Device Scenarios

- Delayed / Queued tests
 - Upload APKs to service
 - Queues tests
 - Remote report / dashboard



Device Scenarios

- Delayed / Queued tests
 - Upload APKs to service
 - Queues tests
 - Remote report / dashboard
- TestServer API
 - Upload APKs method
 - Plugin register itself to the android plugin
 - Support any number of Servers



Device Scenarios

- Devices Available on-demand
 - Direct connection to device
 - Local/Cloud emulators
 - Local report / dashboard



Device Scenarios

- Devices Available on-demand
 - Direct connection to device
 - Local/Cloud emulators
 - Local report / dashboard
- DeviceProvider / DeviceConnector API
 - Return list of DeviceConnectors
 - Connect, install, run tests, uninstall on each device
 - Support any number of Providers





Build Environment

Build Environment

Gradle Version

Use the Wrapper!



Build Environment

Gradle Version

Use the Wrapper!

Plugin Version

```
'com.android.tools.build:gradle:0.4'
```



Build Environment

Gradle Version

Use the Wrapper!

Plugin Version

```
'com.android.tools.build:gradle:0.4'
```

Build Tools Version

```
android.buildToolsVersion '16.1'
```



Build Environment

Gradle Version

Use the Wrapper!

Plugin Version

```
'com.android.tools.build:gradle:0.4'
```

Build Tools Version

```
android.buildToolsVersion '16.1'
```

IDE is not responsible for Builds



<Thank You!>

<http://tools.android.com>

adt-dev@googlegroups.com

<http://b.android.com>





Google
Developers