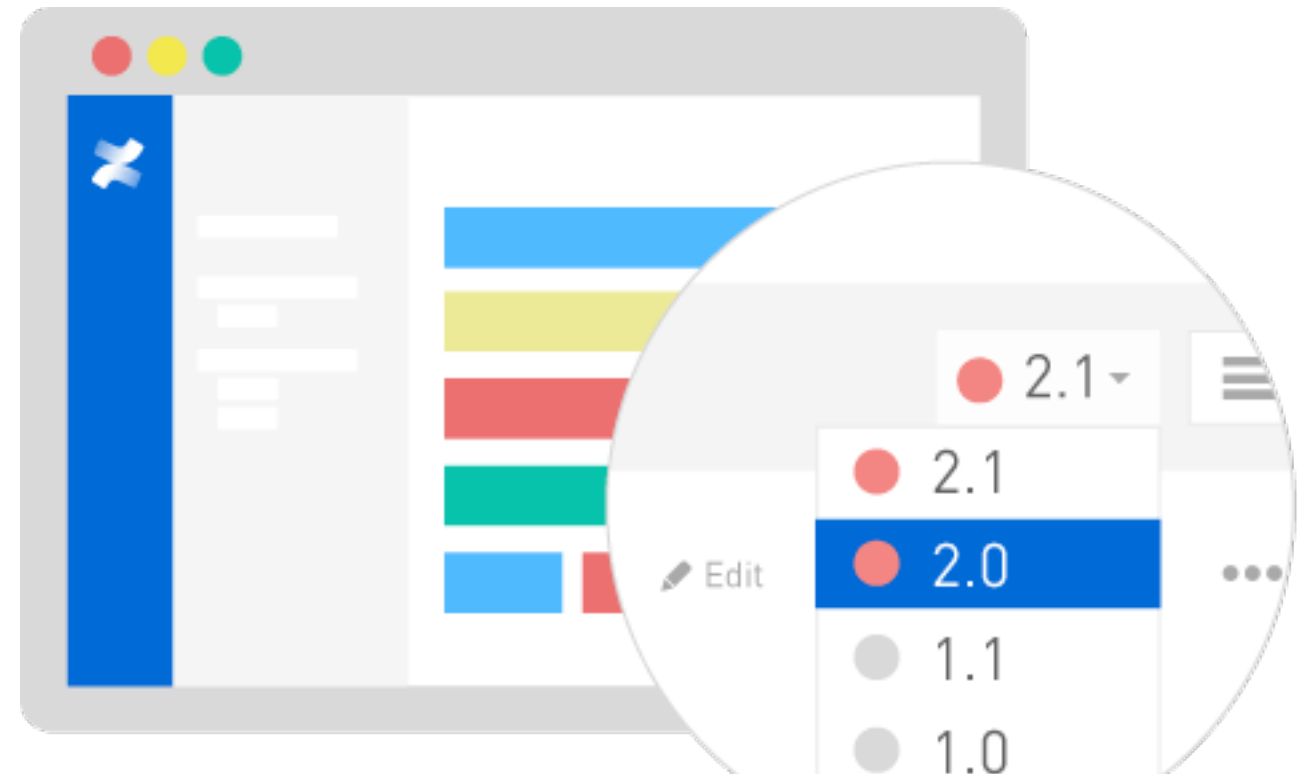


# Version Management

---



## Semantic Versioning

“In the world of software management there exists a dread place called “dependency hell.” The bigger your system grows and the more packages you integrate into your software, the more likely you are to find yourself, one day, in this pit of despair.”

<https://semver.org/>

“...Once you identify your public API, you communicate changes to it with specific increments to your version number. Consider a version format of X.Y.Z (Major.Minor.Patch). Bug fixes not affecting the API increment the patch version, backwards compatible API additions/changes increment the minor version, and backwards incompatible API changes increment the major version...”

“...I call this system “Semantic Versioning.” Under this scheme, version numbers and the way they change convey meaning about the underlying code and what has been modified from one version to the next...”

## Summary

Given a version number MAJOR.MINOR.PATCH,  
increment the:

1. **MAJOR** version when you make incompatible API changes,
2. **MINOR** version when you add functionality in a backwards-compatible manner, and
3. **PATCH** version when you make backwards-compatible bug fixes.

Additional labels for pre-release and build metadata are available as extensions to the MAJOR.MINOR.PATCH format.

# Project Gradle

```
buildscript {  
    ext.kotlin_version = '1.2.21'  
    repositories {  
        google()  
        jcenter()  
    }  
    dependencies {  
        classpath 'com.android.tools.build:gradle:3.0.1'  
        classpath "org.jetbrains.kotlin:kotlin-gradle-plugin:$kotlin_version"  
    }  
}  
  
allprojects {  
    repositories {  
        google()  
        jcenter()  
    }  
}  
  
task clean(type: Delete) {  
    delete rootProject.buildDir  
}
```

# Project Gradle

Kotlin  
Compiler

Gradle

```
buildscript {  
    ext.kotlin_version = '1.2.21'  
    repositories {  
        google()  
        jcenter()  
    }  
    dependencies {  
        classpath 'com.android.tools.build:gradle:3.0.1'  
        classpath "org.jetbrains.kotlin:kotlin-gradle-plugin:$kotlin_version"  
    }  
}  
  
allprojects {  
    repositories {  
        google()  
        jcenter()  
    }  
}  
  
task clean(type: Delete) {  
    delete rootProject.buildDir  
}
```

# Application Gradle

```
apply plugin: 'com.android.application'
apply plugin: 'kotlin-android'
apply plugin: 'kotlin-android-extensions'

android {
    compileSdkVersion 26
    defaultConfig {
        applicationId "org.wit.placemark"
        minSdkVersion 23
        targetSdkVersion 26
        versionCode 1
        versionName "1.0"
        testInstrumentationRunner "android.support.test.runner.AndroidJUnitRunner"
    }
    ...
}

dependencies {
    implementation fileTree(dir: 'libs', include: ['*.jar'])
    implementation "org.jetbrains.kotlin:kotlin-stdlib-jre7:$kotlin_version"
    implementation 'com.android.support:appcompat-v7:26.1.0'
    implementation 'com.android.support.constraint:constraint-layout:1.0.2'
    implementation 'com.android.support:design:26.1.0'
    implementation 'com.android.support:cardview-v7:26.1.0'

    implementation 'org.jetbrains.anko:anko:0.10.3'
    implementation 'org.jetbrains.anko:anko-commons:0.10.3'

    implementation 'com.google.android.gms:play-services-maps:11.6.2'
    implementation "com.google.android.gms:play-services-location:11.6.2"

    compile 'android.arch.persistence.room:runtime:1.0.0'
    kapt 'android.arch.persistence.room:compiler:1.0.0'

    ...
}
```

# Application Gradle

```
apply plugin: 'com.android.application'
apply plugin: 'kotlin-android'
apply plugin: 'kotlin-android-extensions'
```

```
android {
    compileSdkVersion 26
    defaultConfig {
        applicationId "org.wit.placemark"
        minSdkVersion 23
        targetSdkVersion 26
        versionCode 1
        versionName "1.0"
        testInstrumentationRunner "android.support.test.runner.AndroidJUnitRunner"
    }
    ...
}
```

Android  
Platform  
versions

```
dependencies {
    implementation fileTree(dir: 'libs', include: ['*.jar'])
    implementation "org.jetbrains.kotlin:kotlin-stdlib-jre7:$kotlin_version"
    implementation 'com.android.support:appcompat-v7:26.1.0'
    implementation 'com.android.support.constraint:constraint-layout:1.0.2'
    implementation 'com.android.support:design:26.1.0'
    implementation 'com.android.support:cardview-v7:26.1.0'

    implementation 'org.jetbrains.anko:anko:0.10.3'
    implementation 'org.jetbrains.anko:anko-commons:0.10.3'

    implementation 'com.google.android.gms:play-services-maps:11.6.2'
    implementation "com.google.android.gms:play-services-location:11.6.2"

    compile 'android.arch.persistence.room:runtime:1.0.0'
    kapt 'android.arch.persistence.room:compiler:1.0.0'

    ...
}
```

Android  
Application  
Libraries



# Android Application Libraries

```
implementation "org.jetbrains.kotlin:kotlin-stdlib-jre7:$kotlin_version"
implementation 'com.android.support:appcompat-v7:26.1.0'
implementation 'com.android.support.constraint:constraint-layout:1.0.2'
implementation 'com.android.support:design:26.1.0'
implementation 'com.android.support:cardview-v7:26.1.0'

implementation 'org.jetbrains.anko:anko:0.10.3'
implementation 'org.jetbrains.anko:anko-commons:0.10.3'

implementation 'com.google.android.gms:play-services-maps:11.6.2'
implementation 'com.google.android.gms:play-services-location:11.6.2'

compile 'android.arch.persistence.room:runtime:1.0.0'
kapt 'android.arch.persistence.room:compiler:1.0.0'
```

All libraries use semantic versioning  
Many libraries must be kept in sync on “MAJOR”  
version number

# Version Identifiers

## Project Gradle

```
ext.kotlin_version = '1.2.21'
```

Centralise all  
versions in  
Project  
Gradle

```
implementation "org.jetbrains.kotlin:kotlin-stdlib-jre7:$kotlin_version"
```

## Application Gradle

```
ext.kotlin_version = '1.2.31'
ext.gradle_version = '3.1.0'

ext.app_compat_version = '26.1.0'
ext.design_library_version = '26.1.0'
ext.support_library_version = '26.1.0'
ext.cardview_library_version = '26.1.0'

ext.constraint_layout_version = '1.0.2'

ext.anko_version = '0.10.4'
ext.anko_commons_version = '0.10.4'

ext.play_services_maps_version = '12.0.1'
ext.play_services_location_version = '12.0.1'

ext.room_version = '1.0.0'
```

## Project Gradle

```
implementation "org.jetbrains.kotlin:kotlin-stdlib-jre7:$kotlin_version"

implementation "com.android.support:appcompat-v7:$app_compat_version"
implementation "com.android.support.constraint:constraint-layout:$constraint_layout_version"
implementation "com.android.support:design:$design_library_version"
implementation "com.android.support:cardview-v7:$cardview_library_version"

implementation "org.jetbrains.anko:anko:$anko_version"
implementation "org.jetbrains.anko:anko-commons:$anko_commons_version"

implementation "com.google.android.gms:play-services-maps:$play_services_maps_version"
implementation "com.google.android.gms:play-services-location:$play_services_location_version"

implementation "android.arch.persistence.room:runtime:$room_version"
annotationProcessor "android.arch.persistence.room:compiler:$room_version"
kapt "android.arch.persistence.room:compiler:$room_version"
```

## Application Gradle

# Project Gradle

```
ext.kotlin_version = '1.2.31'
ext.gradle_version = '3.1.0'

ext.app_compat_version = '26.1.0'
ext.design_library_version = '26.1.0'
ext.support_library_version = '26.1.0'
ext.cardview_library_version = '26.1.0'

ext.constraint_layout_version = '1.0.2'

ext.anko_version = '0.10.4'
ext.anko_commons_version = '0.10.4'

ext.play_services_maps_version = '12.0.1'
ext.play_services_location_version = '12.0.1'

ext.room_version = '1.0.0'
```

Mutually dependent  
components aligned on  
Major (at least) number