

# Configurar objetivos para Kotlin Multiplatform

Puedes añadir objetivos al crear un proyecto con el asistente de proyectos. Si necesita agregar un objetivo más tarde, puede hacerlo manualmente utilizando los ajustes preestablecidos de destino para las plataformas compatibles.

Obtenga más información sobre la configuración adicional para los objetivos.

```
kotlin {
    jvm() // Create a JVM target with the default name 'jvm'

    linuxX64() {
        /* Specify additional settings for the 'linux' target here */
    }
}
```

Cada objetivo puede tener una o más compilaciones. Además de las compilaciones predeterminadas con fines de prueba y producción, puede crear compilaciones personalizadas.

## Distinguir varios objetivos para una plataforma

Puedes tener varios objetivos para una plataforma en una biblioteca multiplataforma. Por ejemplo, estos objetivos pueden proporcionar la misma API, pero utilizan diferentes bibliotecas durante el tiempo de ejecución, como marcos de prueba y soluciones de registro. Las dependencias de dicha biblioteca multiplataforma pueden no resolverse porque no está claro qué objetivo elegir.

Para resolver esto, marque los objetivos tanto en el lado del autor de la biblioteca como en el del consumidor con un atributo personalizado, que Gradle utiliza durante la resolución de dependencias.

Por ejemplo, considere una biblioteca de pruebas que admita tanto JUnit como TestNG en los dos objetivos. El autor de la biblioteca debe añadir un atributo a ambos objetivos de la siguiente manera:

```
val testFrameworkAttribute = Attribute.of("com.example.testFramework",
String::class.java)

kotlin {
    jvm("junit") {
        attributes.attribute(testFrameworkAttribute, "junit")
    }
    jvm("testng") {
        attributes.attribute(testFrameworkAttribute, "testng")
    }
}
```

El consumidor tiene que añadir el atributo a un solo objetivo donde surge la ambigüedad.