

---

# Gradle vs Maven

Luis Mateo Hincapié Martínez

Arquitectura de software  
Facultad de Ingeniería  
Universidad de Antioquia  
2020-1

Si es desarrollador, probablemente haya oído hablar o incluso utilizado **software de automatización de compilación**. El propósito de este software es **automatizar** algunas tareas asociadas con **la creación de software**. Por ejemplo, un software de automatización de compilación puede ayudarlo a compilar el código fuente de una aplicación, ejecutar pruebas de la aplicación, crear un instalador e incluso instalar la aplicación en un servidor remoto.

Las herramientas de compilación son una parte esencial del proceso DevOps. Ayudan a **ahorrar tiempo, resaltar problemas** potenciales y **facilitar el trabajo** de los desarrolladores. Hay decenas de software de automatización de compilación en el mercado. En este informe analizaremos dos de los más usados específicamente para el desarrollo en Java: **Gradle y Maven**.

## Gradle



Gradle es una herramienta de automatización de compilación y administración de dependencias que se basó en los conceptos de Ant y Maven. Una de las primeras cosas que podemos notar sobre Gradle es que no usa archivos XML, a diferencia de Maven.

Con el tiempo, los desarrolladores se interesaron cada vez más en tener y **trabajar con un lenguaje específico de dominio**, lo que, en pocas palabras, les permitiría resolver problemas en un dominio específico utilizando un lenguaje adaptado a ese dominio en particular.

Esto fue adoptado por Gradle, que **utiliza un DSL basado en Groovy o Kotlin**. Esto llevó a **archivos de configuración más pequeños con menos desorden**, ya que el lenguaje fue diseñado específicamente para resolver problemas de dominio específicos. El archivo de

---

configuración de Gradle se llama por convención *build.gradle* en Groovy, o *build.gradle.kts* en Kotlin.

Fue diseñado para admitir compilaciones de varios **proyectos** que se espera sean bastante **grandes**. Admite compilaciones incrementales, por lo que sabrá qué partes de su proyecto son actualizadas. Las tareas que dependen de piezas actualizadas ya no se vuelven a ejecutar. Por ahora, la última versión estable es la versión 6.6.1, que se lanzó en agosto de 2020. Es compatible con el desarrollo e implementación de **Java, Scala y Groovy**.

## Maven



Apache Maven es una herramienta de automatización de compilación y gestión de dependencias, que se utiliza principalmente para aplicaciones Java. Le ayuda a trazar un mapa de cómo se construye un software en particular, así como sus diferentes dependencias.

Utiliza un archivo **XML para describir el proyecto** que está creando, las dependencias del software con respecto a módulos y partes de terceros, el orden de construcción y los complementos necesarios. Hay objetivos predefinidos para tareas como empaquetar y compilar. Maven se basa en convenciones y proporciona comandos predefinidos (objetivos).

En pocas palabras, Maven nos permite centrarnos en lo que debería hacer nuestra construcción y **nos brinda el marco** para hacerlo. Otro aspecto positivo de Maven fue que proporcionó soporte integrado para la **gestión de dependencias**. Maven descargará bibliotecas y complementos de los diferentes repositorios y luego los pondrá todos en un caché en su máquina local. Si bien se usa principalmente para proyectos **Java**, puede usarlo para **Scala, Ruby y C #**, así como para una gran cantidad de otros lenguajes.

El archivo de configuración de Maven, que contiene instrucciones de administración de dependencia y compilación, se llama por convención *pom.xml*. Además, Maven también prescribe una estructura de proyecto estricta.

# Comparación

## Gradle

La herramienta Gradle utiliza el lenguaje específico de dominio basado en **Groovy** para desarrollar proyectos. La configuración del proyecto se mantiene por separado para los proyectos.

El objetivo definido es agregar una nueva funcionalidad en el proyecto.

El rendimiento de Gradle es mejor que Maven, ya que sólo rastrea la tarea en ejecución y no la tarea de entrada o salida.

Hay varias opciones presentes en términos de compatibilidad con IDE. Por lo tanto, hace que la herramienta sea altamente personalizable.

La compilación de java no es obligatoria.

Al ser más nueva, los usuarios de la herramienta son limitados en comparación con la herramienta Maven.

Tiene numerosas opciones para incluir las nuevas dependencias para el proyecto y el proceso de gestión de dependencias es fácil de comparar con el proyecto maven porque la herramienta evita el uso de archivos XML para la gestión de dependencias.

Empresas que lo usan:

- Netflix
- Lyft
- 9GAG

## Maven

El archivo **XML** se utiliza en la herramienta Maven para mantener la lista de componentes, dependencias del proyecto y otros requisitos.

El objetivo definido es completar el proyecto dentro del cronograma.

No utiliza el caché de compilación para la creación de proyectos, por lo que su tiempo de creación de proyectos es mayor en comparación con la herramienta Gradle.

No admite varios IDE, lo que hace que la personalización sin herramientas se compare con Gradle.

La compilación es obligatoria.

Es una herramienta conocida que hace que la herramienta esté fácilmente disponible para que los programadores desarrollen los nuevos proyectos.

También tiene la opción de agregar una nueva dependencia para la creación del proyecto, pero demasiadas dependencias pueden crear problemas para administrar la dependencia porque administrar el archivo XML es un poco complicado en comparación con la herramienta Gradle.

Empresas que lo usan:

- Zillow
- Intuit
- Zalando

---

Tanto Gradle como Maven son las herramientas de automatización que se utilizan para crear nuevos proyectos en varios lenguajes de programación que admiten múltiples entornos IDE. Tanto la herramienta tiene sus propias ventajas como desventajas y seleccionar la herramienta entre dos depende completamente de los requisitos del proyecto.

Gradle parece tener una curva de aprendizaje más pronunciada, especialmente si no está familiarizado con Groovy o Kotlin.

La "flexibilidad" es la razón principal por la que a los desarrolladores les gusta Gradle, mientras que la "gestión de dependencias" es la principal causa por la que los desarrolladores eligen Apache Maven.

Gradle y Apache Maven son herramientas de código abierto. Gradle con 9.23K GitHub estrellas y 2.7K forks en GitHub parece ser más popular que Apache Maven con 1.74K GitHub estrellas y 1.28K GitHub forks.

Netflix, Lyft y 9GAG son algunas de las empresas populares que utilizan Gradle, mientras que Apache Maven es utilizado por Zillow, Intuit y Zalando. Gradle tiene una aprobación más amplia, y se menciona en 465 pilas de empresas y 360 de desarrolladores; en comparación con Apache Maven, que figura en 305 pilas de empresas y 142 pilas de desarrolladores según un reporte de la página stackshare.

Al final, lo que elija dependerá principalmente de lo que necesite. Gradle es más poderoso. Sin embargo, hay ocasiones en las que realmente no necesitas la mayoría de las características y funcionalidades que ofrece. Maven puede ser mejor para proyectos pequeños, mientras que Gradle es mejor para proyectos más grandes.

---

## Referencias

[Top 10 Build Automation Software for 2019](#)

[Ant vs Maven vs Gradle](#)

[Gradle vs Maven | Top 11 Key Features of Apache Gradle vs Maven](#)

[Gradle vs Apache Maven | What are the differences?](#)

[Gradle Build Tool](#)

[Maven – Welcome to Apache Maven](#)

[5 Best Java build automation tools as of 2020](#)

[Gradle vs. Maven](#)