



# Tecnologías Web: Gestores de Paquetes

Cristian Augusto

[augustocristian@uniovi.es](mailto:augustocristian@uniovi.es)

Software Engineering Research Group

<http://giis.uniovi.es/>

Curso 2024-2025



# Desarrollo del software

- Compilación
- Empaquetado
- Pruebas
  - Pruebas Unitarias
  - Pruebas de Integración
  - Pruebas estáticas
  - Pruebas de sistema
- Despliegue



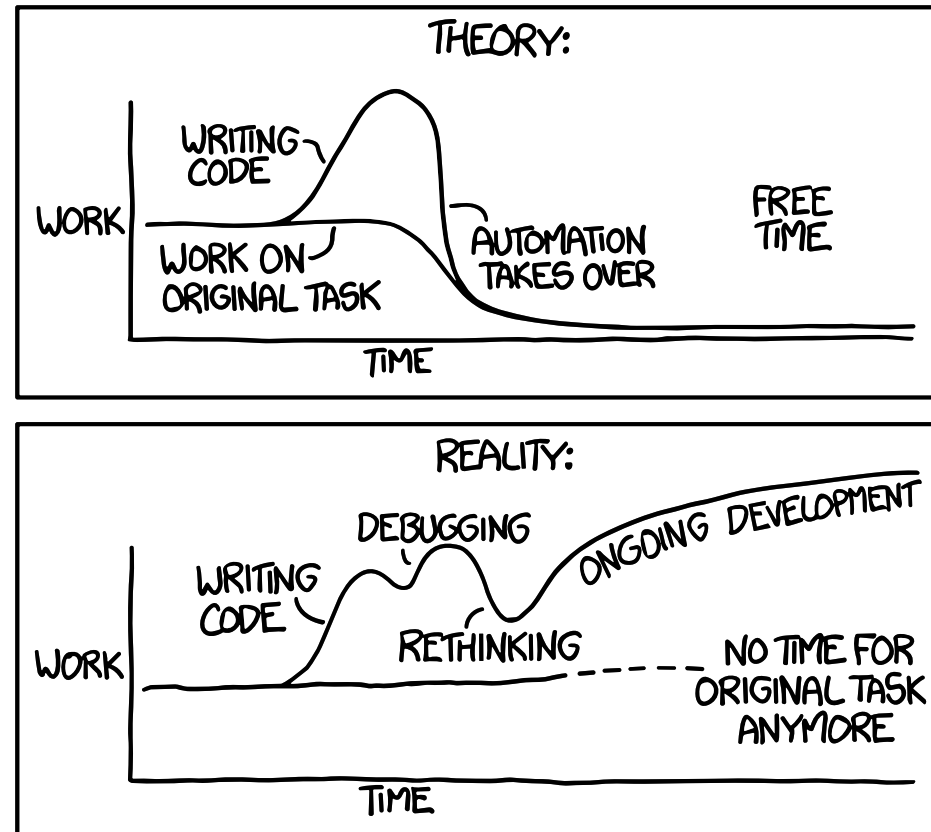
# Automatización: Reducir el “Toil”

- ¿Qué es el Toil [1]?
  - *“Any manual, structurally mandated operational task”*
- Beneficios de automatizar tareas en el desarrollo del software:
  - Reducir costes
  - Mejorar procesos y QA
  - Reducir release time (time-to-market) y retroalimentación
  - y mas...

[1] Betsy, B., Chris, J., Jennifer, P., & Niall Richard, M. (2016). *Site Reliability Engineering: How Google Runs Production Systems*. O'Reilly Media, Inc.

# ¿Todo debe de ser automatizado?

"I SPEND A LOT OF TIME ON THIS TASK.  
I SHOULD WRITE A PROGRAM AUTOMATING IT!"



Fuente: [xkcd.com/1319/](http://xkcd.com/1319/)

# ¿Qué es un gestor de paquetes?

- Es una herramienta que automatiza y simplifica la compilación y generación de ejecutables
- ANTES:
  - Estructuras compartidas para compilar → Cada proyecto era diferente → Curva de aprendizaje elevada
- CON UN GESTOR DE PAQUETES:
  - Todos los proyectos son compilados y ejecutados de la misma forma
- OBJETIVO: Automatizar la construcción del software

# Alternativas

- C, C++: Makefile
- Mundo Java: Ant, **Maven**, Gradle...
- Python: pip, poetry, pipenv
- .NET: NuGet
- Rust: Cargo
- Node: NPM
- Deno: JSR

# Maven

- Gestor de paquetes open-source que busca simplificar el proceso de construcción del software
- Lenguajes soportados:
  - Principalmente proyectos en Java
  - También soporta otros lenguajes:
    - C#
    - Ruby
    - Scala
- Dispone de un Ciclo de Vida que rige la construcción del software

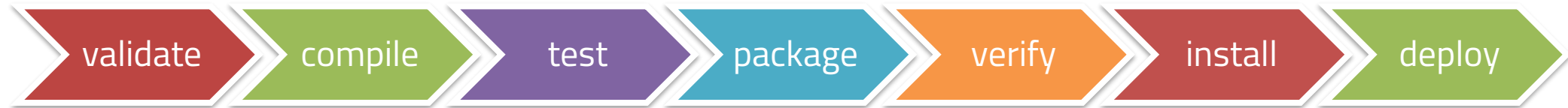
# Maven: Ciclo de vida

- Cada proyecto tiene tres ciclos y una serie de etapas
- Existen etapas “fuera del ciclo” que a su vez pueden tener su propio ciclo de vida. Ej:
  - **Clean:** limpieza de los .class y .jar generados.
  - **Assembly:** empaquetado de todo lo que se necesita para ejecutar el programa.



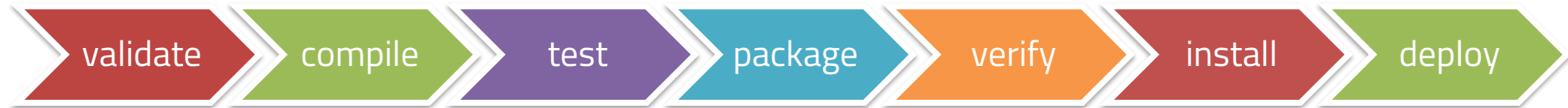


# Maven: Ciclo de vida



- **Validación:** Análisis estático y errores previos a compilación.
- **Compilación:** Se emplea el compilador para transformar el lenguaje en bytecode.
- **Test:** ejecución de las pruebas empleando el framework seleccionado.
- **Empaquetado:** Empaqueta en un .jar o .war el bytecode

# Maven: Ciclo de vida



- **Verificación:** pruebas de integración para asegurarse que cumplen los criterios de calidad.
- **Instalación:** instala los paquetes en el repositorio local para poder emplearlo en otros proyectos
- **Despliegue:** despliega en el entorno de compilación, copia el paquete final en el repositorio remoto
- Adicionalmente se pueden crear etapas adicionales **“goals”**

# Project Object Model (POM)

- Fichero en XML
- Contiene:
  - Fases y dependencias
  - Configuración en forma estructurada.
- POM mínimo:

```
<project>
```

```
  <modelVersion>4.0.0</modelVersion>
```

```
  <groupId>com.mycompany.app</groupId>
```

```
  <artifactId>my-app</artifactId>
```

```
  <version>1</version>
```

```
</project>
```

# Project Object Model (POM)

- Fichero en XML
- Contiene:
  - Fases y dependencias
  - Configuración en forma estructurada.
- POM mínimo:

```
<project>  
  <modelVersion>4.0.0</modelVersion>  
  <groupId>com.mycompany.app</groupId>  
  <artifactId>my-app</artifactId>  
  <version>1</version>  
</project>
```

modelversión:  
Por defecto 4.0.0.

groupid:  
Es el id del proyecto de grupo  
normalmente el nombre de la  
organización

artifactid:  
Es el id del proyecto de grupo  
normalmente el nombre de la  
organización

version: versión del artefacto

# Ejemplo de pom.xml

```
gestioneitor-springboot > pom.xml
1  <?xml version="1.0" encoding="UTF-8"?>
2  <project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.
3      xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 https://m
4      <modelVersion>4.0.0</modelVersion>
5      <parent>
6          <groupId>org.springframework.boot</groupId>
7          <artifactId>spring-boot-starter-parent</artifactId>
8          <version>3.1.4</version>
9          <relativePath/> <!-- lookup parent from repository -->
10     </parent>
11     <groupId>com.tew</groupId>
12     <artifactId>gestioneitor-springboot</artifactId>
13     <version>0.0.1-SNAPSHOT</version>
14     <packaging>war</packaging>
15     <name>gestioneitor-springboot</name>
16     <description>gestioneitor-springboot</description>
17     <properties>
18         <java.version>17</java.version>
19     </properties>
20     <dependencies>
21         <dependency>
22             <groupId>org.springframework.boot</groupId>
23             <artifactId>spring-boot-starter-jdbc</artifactId>
24         </dependency>
25         <dependency>
26             <groupId>org.springframework.boot</groupId>
27             <artifactId>spring-boot-starter-thymeleaf</artifactId>
28         </dependency>
29         <dependency>
30             <groupId>org.springframework.boot</groupId>
31             <artifactId>spring-boot-starter-web</artifactId>
32         </dependency>
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
        <dependency>
            <groupId>org.springframework.boot</groupId>
            <artifactId>spring-boot-devtools</artifactId>
            <scope>runtime</scope>
            <optional>true</optional>
        </dependency>
        <dependency>
            <groupId>org.hsqldb</groupId>
            <artifactId>hsqldb</artifactId>
            <version>2.7.2</version>
            <scope>runtime</scope>
        </dependency>
        <dependency>
            <groupId>org.springframework.boot</groupId>
            <artifactId>spring-boot-starter-tomcat</artifactId>
            <scope>provided</scope>
        </dependency>
        <dependency>
            <groupId>org.springframework.boot</groupId>
            <artifactId>spring-boot-starter-test</artifactId>
            <scope>test</scope>
        </dependency>
    </dependencies>
    <build>
        <plugins>
            <plugin>
                <groupId>org.springframework.boot</groupId>
                <artifactId>spring-boot-maven-plugin</artifactId>
            </plugin>
        </plugins>
    </build>
</project>
```



# Tecnologías Web: Gestores de Paquetes

Cristian Augusto

[augustocristian@uniovi.es](mailto:augustocristian@uniovi.es)

Software Engineering Research Group

<http://giis.uniovi.es/>

Curso 2024-2025

