

# JUG Py

Java User Group Paraguay

<http://jugpy.org/>

Agosto - 2015

Rafael Benegas

e-mail: [rafaelbenegas@gmail.com](mailto:rafaelbenegas@gmail.com)

# Construcción de proyectos

con



**¿Maven?**

Herramienta para la  
construcción, comprensión y  
administración de proyectos  
basados en Java.

# Instalación

- Versión actual 3.3
- Descargar y descomprimir
- Requiere mínimamente del JDK 1.7
- Setear variables de entorno `JAVA_HOME` y `PATH`

# **pom.xml**

Project Object Model

# Estructura mínima

```
<project>
```

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

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

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

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

```
</project>
```



**<groupId>:<artifactId>:<version>**

Fully qualified artifact name

# Super POM

<http://maven.apache.org/ref/3.0.4/maven-model-builder/super-pom.html>

# settings.xml

- `$M2_HOME/conf/settings.xml`
- `${user.home}/.m2/settings.xml`
- Más info: <http://maven.apache.org/settings.html>

# **settings.xml**

<http://maven.apache.org/guides/mini/guide-encryption.html>

# Uso del comando mvn

mvn [options] [<goal(s)>] [<phase(s)>]

# Dependencias

<project>

...

<dependencies>

<dependency>

<groupId>com.some.group</groupId>

<artifactId>dependency-artifact-id</artifactId>

<version>1.1.2</version>

<scope>provided</scope>

<dependency>

</dependencies>

# Dependencias transitivas

Es una característica que permite que se incluyan como dependencia de forma automática a las dependencias de dependencias directas.

El árbol de dependencias no tiene un límite en cuanto a su nivel, por ello Maven provee algunas características para lidiar con posibles problemas que puedan aparecer.



- Dependency mediation
- Dependency management
- Dependency scope
- Dependency exclusion
- Optional dependencies

# **Herencia y Agregación**

## **En la herencia se mezclan los siguientes elementos del proyecto:**

- dependencies
- developers and contributors
- plugins
- plugins executions con mismo id
- plugin configuration

# **Interpolación y variables**

En ocasiones se necesita usar un mismo valor en diversas partes del proyecto. Con la interpolación esos valores pueden definirse una sola vez

# Ejemplo:

```
<project>
```

```
...
```

```
  <build>
```

```
    <finalName>
```

```
      ${project.artifactId}-${project.version}
```

```
    </finalName>
```

```
  </build>
```

```
...
```

```
</project>
```

# Plugins

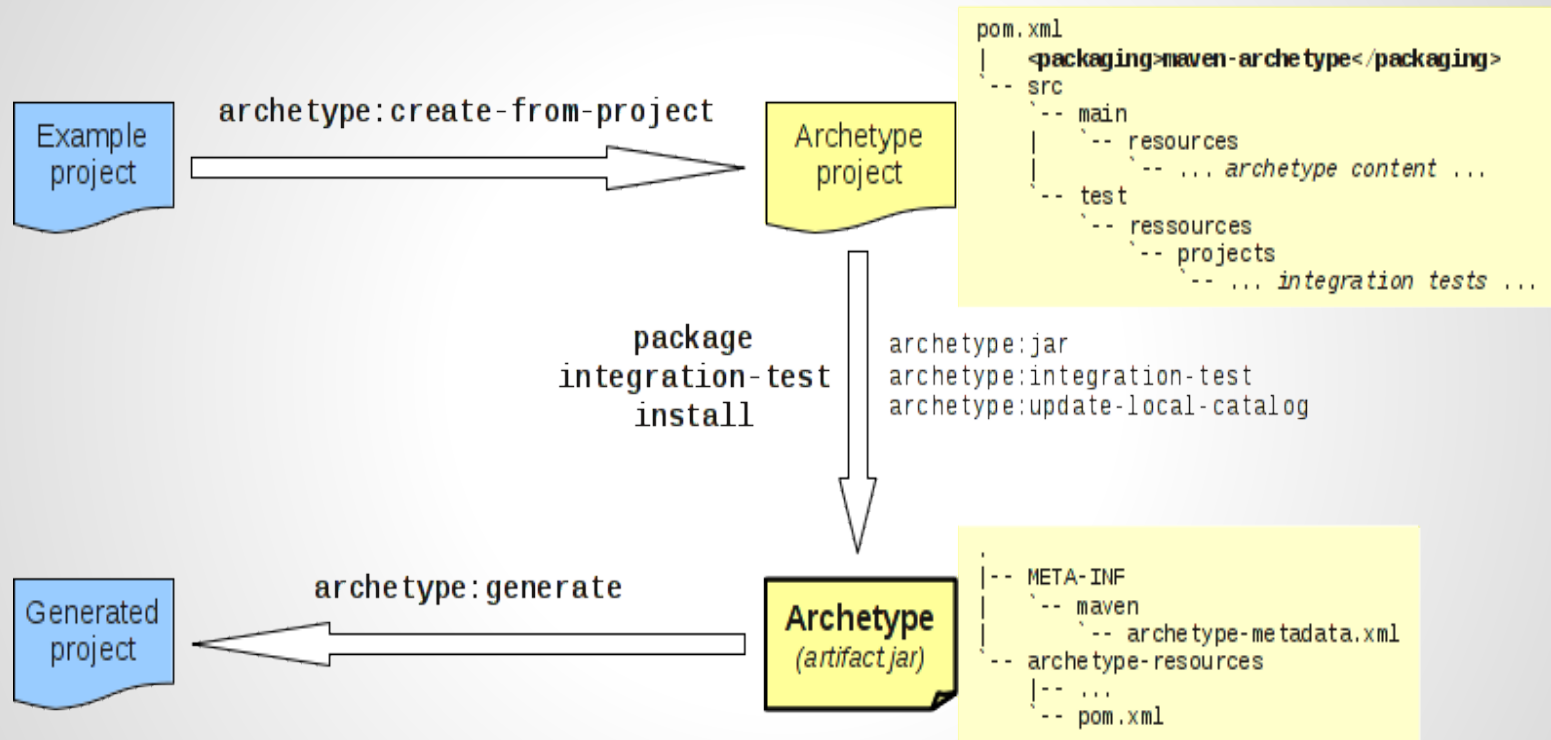
# Default plugins

<https://maven.apache.org/ref/3.2.2/maven-core/default-bindings.html>



# **Archetype plugin**

Permite reutilizar lógica de construcción.



# Checkstyle plugin

[http://checkstyle.sourceforge.net/google\\_style.html](http://checkstyle.sourceforge.net/google_style.html)

[http://checkstyle.sourceforge.net/sun\\_style.html](http://checkstyle.sourceforge.net/sun_style.html)

# Referencias

La más completa: <https://maven.apache.org/>

**Gracias por su atención! :-)**