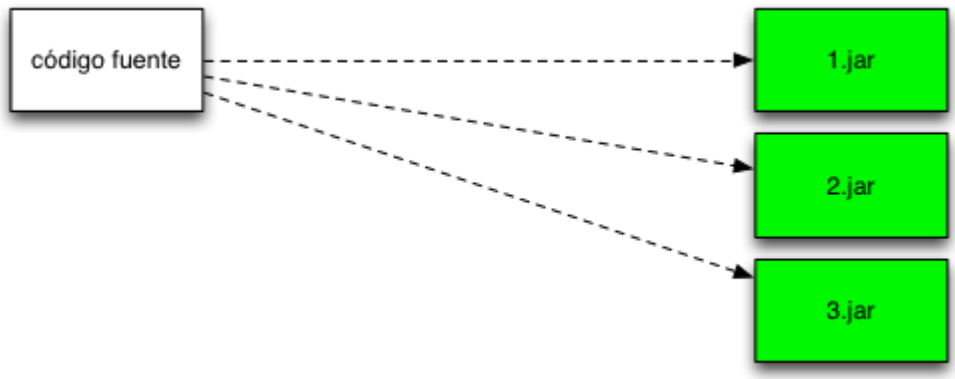
El concepto de Maven Tutorial es muy amplio ¿Qué es Maven?. Probablemente es difícil responder de forma sencilla a esta pregunta. Sin embargo si tuviera que decir algo diría que es una herramienta que controla el ciclo de desarrollo de software. Hemos visto en los post anteriores como se generan los módulos mas importantes de la plataforma JEE . Ademas hemos podido analizar su contenido .Ahora mismo con lo que sabemos nuestro ciclo de desarrollo de software se puede reducir a lo siguiente . Tenemos un código fuente y generamos un desplegable (JAR,WAR,EAR) es así de sencillo. Ahora bien si nos fijamos en la siguiente figura.

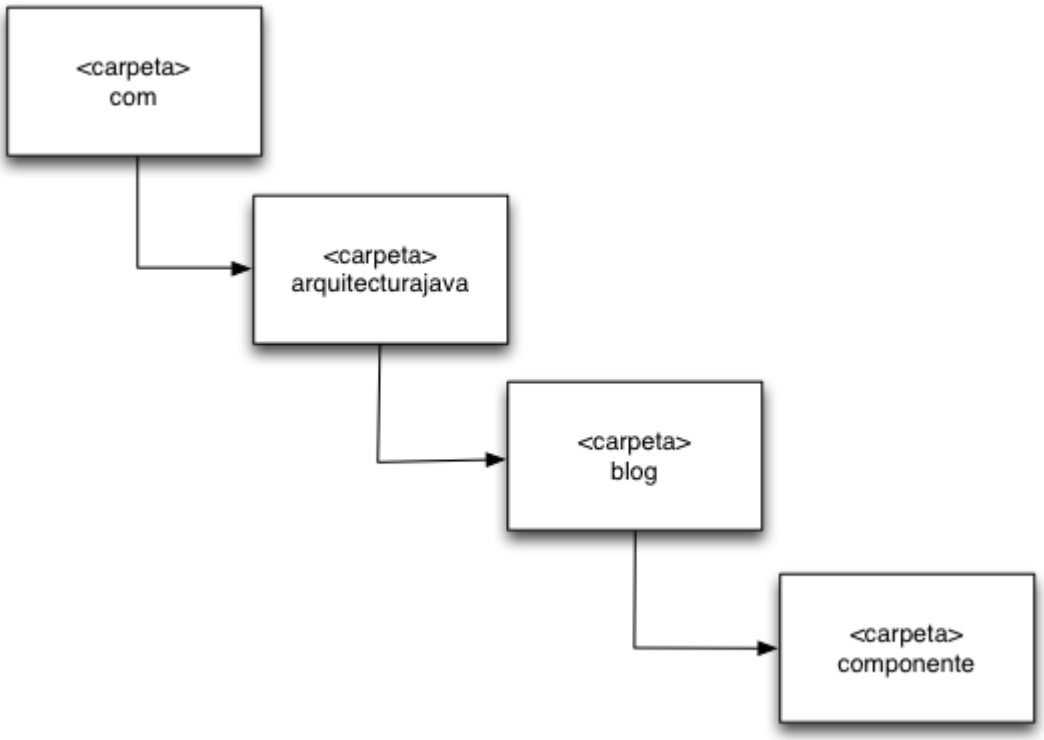


Es bastante claro que tenemos nuestro código fuente y generamos tres ficheros JAR .¿O quizás la figura quiere decir que tenemos 3 versiones del mismo fichero JAR ?.¿O Quizás se ha copiado y pegado el mismo JAR tres veces?. Las cosas no son tan sencillas como nos parecen a veces.La forma de empaquetar de Java deja muchas puertas abiertas a la flexibilidad pero también a la interpretación .Esto muchas veces genera quebraderos de cabeza ya que la gente tiene dudas sobre si la libreria que esta usando es la correcta o si su versión es la correcta. ¿Como podemos solventar estos problemas?. Hay dos cosas que necesitamos definir

- Nombre de la libreria :Especificar un nombre único de nuestra librería
- Versión de la libreria: Especificar una versión de la librería.

Si tuviéramos estas dos cosas podríamos trabajar con mayor comodidad.¿Ahora bien cómo definir el nombre de nuestra librería para que sea única a nivel mundial? .Para ello nos apoyaremos en las estructuras de paquetes Java . Recordemos que se recomienda el uso de

la URL de nuestra pagina web a la hora de definir la jerarquia de paquetes .Por ejemplo en mi caso la URL es “arquitecturajava.com”. Definiremos la estructura de paquetes como “com.arquitecturajava” . A partir de esa estructura nosotros podemos definir las subestructuras que deseemos por ejemplo com.arquitecturajava.blog.componente . En este caso la estructura de paquetes quedaría definida de la siguiente forma.



Maven Tutorial

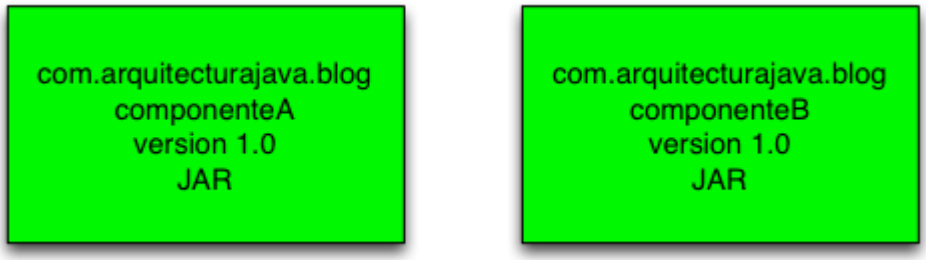
Así pues podemos tener ya claro cual es el nombre de nuestra librería “com.arquitecturajava.blog.componente” .Una vez realizado este primer paso deberemos también asignar una versión a nuestra librería . En este caso para no complicar demasiado las cosas vamos a suponer que la versión es la 1.0. A Ya tenemos dos cosas definidas el nombre (que es único) y la versión.



Ahora bien todavía tenemos que encajarlo de forma definitiva con la estructura de módulos de JEE . Con la información actual no sabemos si se trata de un JAR,WAR o EAR . Así pues deberemos añadir también que tipo de módulo JEE es nuestra librería.



Si tuviéramos varios componentes dentro de nuestro blog la estructura quedaría mas clara de la siguiente forma.



El concepto Java Artifact

Una librería normalmente tiene un nombre y una versión por lo tanto parece adecuado decir que estamos trabajando con una librería. Sin embargo también es evidente que un WAR o un EAR no encajan con el concepto de librería “standard” ya que contienen muchos ficheros que no son puramente Java .Así pues deberemos definir un concepto distinto en el cual estén incluidos el nombre, version y tipo del sotfware que nosotros estamos desarrollando. Este concepto es lo que en Maven se denomina Artefacto. Un artefacto incluye esta información y mucho más pero por ahora para quedarnos con una idea inicial es suficiente. Así pues lo que estamos definiendo es un nuevo Artefacto.

El fichero POM.xml

Ahora bien para poder trabajar con el Artefacto deberemos almacenar esta información en algún fichero.Este fichero se denomina en Maven POM.xml (Project Object Model) es el fichero fundamental de configuración y es el encargado de almacenar la información anteriormente expuesta.A continuación se muestra su contenido

```
<project xmlns="http://maven.apache.org/POM/4.0.0"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
https://maven.apache.org/xsd/maven-4.0.0.xsd">
        <modelVersion>4.0.0</modelVersion>
        <groupId>com.arquitecturajava</groupId>
        <artifactId>jpabasico</artifactId>
        <version>0.0.1-SNAPSHOT</version>
</proyect>
```

Acabamos de construir nuestro primer artefacto con su fichero POM.xml en este Maven Tutorial .La etiqueta modelVersion hace referencia a la propia estructura del fichero Maven (actualmente es la 4) y los demás elementos definen la versión , el nombre y el paquete del

artefacto.

- Maven Parent POM
- Maven Artifact
- POM.XML