



链滴

# Maven 与 Gradle 构建 Fat-Jar 工程

作者: [liumapp](#)

原文链接: <https://ld246.com/article/1557747389192>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

Maven与Gradle相信大家非常清楚，但是fat-jar是个什么鬼？长得比较胖的jar包吗？没错，就是比较胖的jar包...这篇文章介绍如何在Maven跟Gradle中构建这些胖子jar包

首先上项目源代码:

- [Gradle中构建Fat-Jar](#)
- [Maven中构建Fat-Jar](#)

## Fat-Jar

用过大红大紫的SpringBoot，就知道它的打包结果就是一个Fat-Jar：将项目代码以及所依赖的第三方jar包，全部打包到一个Jar包中即为Fat-Jar技术~~~~

具体使用哪种Fat-Jar实现方式，一般而言根据我们所使用的依赖管理工具来决定，这里我们举Gradle Maven这两个最常见的依赖管理工具来举例

## Gradle-plugin: shadowJar

如果我们使用的是Gradle来构建项目，那么可以使用shadowJar这一款plugin来将项目打包为一个"子jar包"

案例代码:

在项目的build.gradle中添加以下代码:

```
apply plugin: "com.github.johnrengelman.shadow"
```

```
jar.enabled = false  
assemble.dependsOn(shadowJar)
```

```
shadowJar {  
    baseName = 'dates-and-times-in-jdk1.8'  
    classifier = null  
    version = 'v1.0.0'  
    manifest {  
        attributes 'Main-Class': 'com.liumapp.tutorials.time.Console'  
    }  
}
```

具体使用请参考文章开头的源码链接

## Maven-plugin: maven-assembly-plugin

如果我们使用Maven来构建项目的话，那么可以使用maven-assembly-plugin来将项目打包为一个"胖子jar包"

在项目pom.xml文件中添加以下代码:

```
<plugin>  
    <groupId>org.apache.maven.plugins</groupId>  
    <artifactId>maven-assembly-plugin</artifactId>
```

```
<version>2.4.1</version>
<configuration>
  <!-- get all project dependencies -->
  <descriptorRefs>
    <descriptorRef>jar-with-dependencies</descriptorRef>
  </descriptorRefs>
  <!-- MainClass in manifest make a executable jar -->
  <archive>
    <manifest>
      <mainClass>com.liumapp.workable.converter.Console</mainClass>
    </manifest>
  </archive>

</configuration>
<executions>
  <execution>
    <id>make-assembly</id>
    <!-- bind to the packaging phase -->
    <phase>package</phase>
    <goals>
      <goal>single</goal>
    </goals>
  </execution>
</executions>
</plugin>
```

具体使用请参考文章开头的源码链接