**Maven 如何为不同的环境打包 ——开发、测试和生产环境**

一、配置profile

首先是profile配置，在pom.xml中添加如下profile的配置：

<profiles>

<profile>



</profile>

<profile>



</profile>

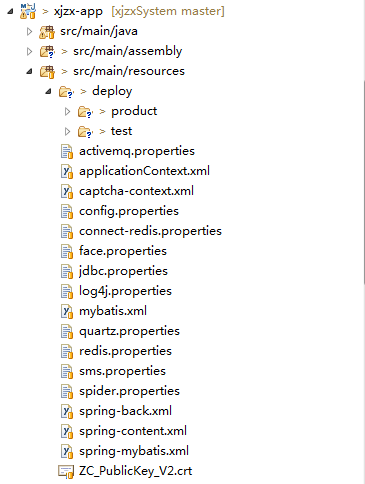
</profiles>

在每个profile中使用了maven-antrun-plugin插件，<phase>compile</phase>表示在编译阶段，<target>...</target>表示要执行的任务。即在编译是将dir="${basedir}/src/main/resources/deploy/product/"目录下的文件拷贝到todir="${basedir}/target/classes/"目录中，overwrite="true"表示如果文件重复则覆盖。

二、配置文件

针对不同的环境，我们定义不同的配置文件，而这些配置文件都做为资源文件放到maven工程的resources目录下，即src/main/resources目录下，

且各个环境的配置分别放到相应的目录下，直接放到src/main/resources/deploy目录下。如下图所示：



三、maven资源插件配置

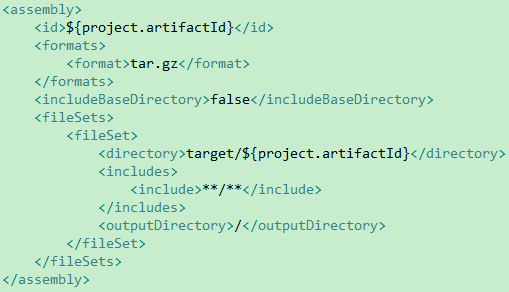
四、打包成tar.gz格式

java项目由maven管理构建时，最常见的是打成jar包或war包，但实际项目中，有可能构建出zip包或tar包，使用maven-assembly构建tar.gz文件形式。

使用到的配置：pom.xml, assembly.xml

调用方式：pom.xml文件中调用assembly.xml

首先定义assembly.xml:



然后定义pom.xml文件



五、打包

Maven打包使用mvn package命令，如果是打不同环境的包，可以加上参数-P加上profile的id，例如正式环境可以通过mvn package -P product来打包，测试环境可以通过mvn package -P test来打包，此时就会执行profile中配置的maven-antrun-plugin插件，完成配置文件的拷贝和覆盖，最终打出来的包中的配置文件是这个环境对应的配置。

开发环境：mvn clean package -Dmaven.test.skip=true

测试环境：mvn clean package -Ptest -Dmaven.test.skip=true

生成环境：mvn clean package -Pproduct -Dmaven.test.skip=true