#### 问题背景

由于console、self、report三个web项目有很多共同的依赖，所以做了bq-framework项目，该项目提供公共的框架功能，并引用了大量的第三方jar包。

部署时，三个web项目会被部署到同一个tomcat上，所以没必要分别在这三个项目的WEB-INF\lib下把bq-framework.jar及其依赖的大量第三方jar包含进来，只需要tomcat的类路径下包含bq-framework.jar及其依赖的大量第三方jar即可，其它项目依赖bq-framework.jar时应设置<scope>provided</scope>。

#### 开发环境方案

开发环境，在catalina.properties的common.loader属性后面加上如下配置：

|  |
| --- |
| ,"path/to/bq-framework/target/classes","path/to/bq-framework/target/lib/\*.jar" |

开发环境下，eclipse中的bq-framework项目下所有的类在经过编译后都会在"path/to/bq-framework/target/classes"下生成相应的.class文件，且对类进行了修改后相应的.class文件也会自动更新，这说明"path/to/bq-framework/target/classes"中的内容是不会过时的，这样的话，就算我们修改了bq-framework中类，也不需要对bq-framework进行package操作。

但是如果修改了bq-framework依赖的第三方jar，如新增了依赖jar或者修改了依赖jar的版本，那么"path/to/bq-framework/target/lib/\*.jar"下的内容就过时了，这时候就需要执行clean和package操作来更新一下"path/to/bq-framework/target/lib/\*.jar"下的内容了。

##### 开发环境部署web项目时不需要package依赖项目

bq-self-war除了依赖bq-framework外，还依赖了bq-self-framework与bq-self-service等。对于bq-framework及其依赖jar，可以在我们配置的类路径找到；而对于bq-self-framework与bq-self-service，部署之前也不需要进行package操作，你会发现部署后bq-self-war\WEB-INF\lib下会存在bq-self-framework.jar与bq-self-service.jar，以及两者的pom.xml中配置的所有第三方jar。（这个感觉很奇怪）

#### 生产环境方案

生产环境上，在tomcat下新建一个extlib目录，用于存放bq-framework.jar及其依赖的大量第三方jar，并在catalina.properties的common.loader属性后加上如下配置：

|  |
| --- |
| ,"${catalina.base}/extlib","${catalina.base}/extlib/\*.jar","${catalina.home}/extlib","${catalina.home}/extlib/\*.jar" |

#### package注意事项

在对bq-framework进行package时，出现如下错误：

|  |
| --- |
| Failed to execute goal on project bq-framework: Could not resolve dependencies for project com.yonyou.bg:bq-framework:jar:5.0.1-RELEASE: **Could not find artifact** com.yonyou.bg:bq-jsqlparser:jar:2.1.0-SNAPSHOT |

说明git仓库中缺少依赖项目bq-jsqlparser的jar，也就是说在对一个项目进行package前要先对其依赖的项目进行package。