

4.11 Maven高阶

4.11.2 Maven依赖传递

2.2.1 什么是依赖传递

在maven中，依赖是可以传递的，假设存在三个项目，分别是项目A，项目B以及项目C。假设C依赖B，B依赖A，那么我们可以根据maven项目依赖的特征不难推出项目C也依赖A。



4.11.3 Maven依赖冲突问题

由于依赖传递，一个工程中传递了不同版本的jar包，这样有可能造成不兼容的问题。

4.11.4 解决依赖冲突-依赖调节原则

第一声明者优先原则-不常用

根据在pom.xml文件中声明的依赖顺序为主

路径近者优先原则

直接依赖 > 依赖传递

4.11.5 解决依赖冲突-排除依赖

4.11.6 解决依赖冲突-版本锁定 最常用

版本锁定的使用方式：

第一步：在dependencyManagement标签中锁定依赖的版本-没有真正引入jar包

第二步：在dependencies标签中声明需要导入的maven坐标

4.11.7 properties标签的使用

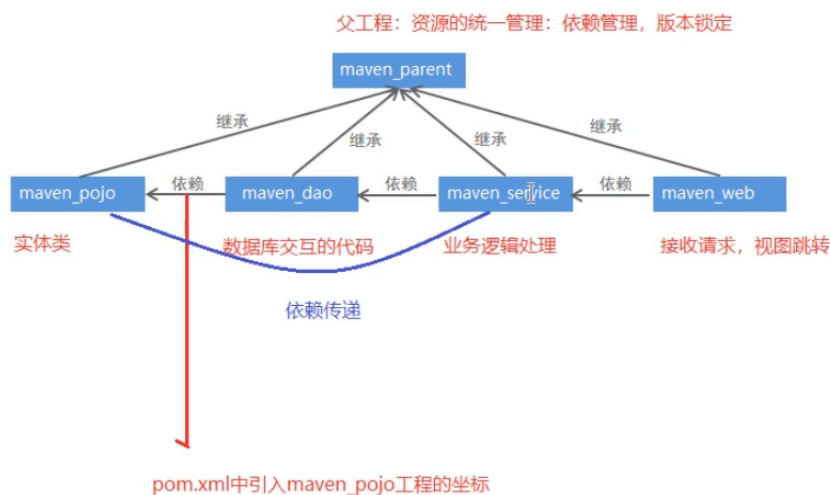
4.11.8 分模块构建Maven工程

常见的拆分方式有两种：

第一种：按照业务模块进行拆分，每个模块拆分成一个maven工程，例如将一个项目分为用户模块，订单模块，购物车模块等，每个模块对应就是一个maven工程

第二种：按照层进行拆分，例如持久层、业务层、表现层等，每个层对应就是一个maven工程

不管上面那种拆分方式，通常都会提供一个父工程，将一些公共的代码和配置提取到父工程中进行统一管理和配置。



4.11.9 Maven工程的继承

在Java语言中，类之间是可以继承的，通过继承，子类就可以引用父类中非private的属性和方法。同样，在maven工程之间也可以继承，子工程继承父工程后，就可以使用在父工程中引入的依赖。继承的目的是为了消除重复代码。

父工程：一般情况下不会编写具体的代码，主要用于公共资源的管理，例如依赖管理，版本锁定

被继承的Maven项目中的POM的部分定义是

```
<groupId>com.company</groupId>
<artifactId>company-project-parent</artifactId>
<version>0.0.1-SNAPSHOT</version>
<packaging>pom</packaging>
```

被继承的maven工程通常称为父工程，父工程的打包方式必须为pom，所以我们区分某个maven工程是否为父工程就看这个工程的打包方式是否为pom

继承的Maven项目中的POM的关键部分就是

```
<parent>
<groupId>com.taotao</groupId>
<artifactId>company-project-parent</artifactId>
<version>0.0.1-SNAPSHOT</version>
</parent>
<artifactId>company-project-children</artifactId>
```

继承其他maven父工程的工程通常称为子工程，在pom.xml文件中通过parent标签进行父工程的继承

4.11.10 Maven工程的聚合

在maven工程的pom.xml文件中可以使用标签将其他maven工程聚合到一起，聚合的目的是为了进行统一操作。

例如拆分后的maven工程有多个，如果要进行打包，就需要针对每个工程分别执行打包命令，操作起来非常繁琐。这时就可以使用标签将这些工程统一聚合到maven父工程中，需要打包的时候，只需要在此父工程中执行一次打包命令，其下被聚合的工程就都会被打包了。

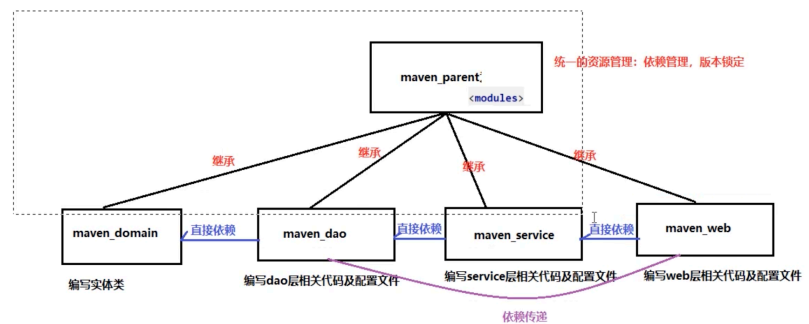
4.11.11 Maven思路分析

对构建maven聚合工程进行思路分析：

- 1) maven_parent为父工程，其余工程为子工程，都继承父工程maven_parent
- 2) maven_parent工程将其子工程都进行了聚合
- 3) 子工程之间存在依赖关系：

maven_dao依赖maven_domain
maven_service依赖maven_dao
maven_web依赖maven_service

拆分原则：按层进行拆分



依赖关系建立原则：当前工程需要用到那个工程的资源，就依赖该工程（直接依赖、间接依赖）

