1.Maven最核心的就是对依赖jar包的管理， Maven要求每一个jar包都必须明确定义自己的坐标，Maven就是通过这个坐标来查找管理这些jar包的。在Maven中，一个jar包的坐标是由它的groupId、artifactId、version这些元素来定义的。

**groupId：表明其所属组织或公司及其所属项目，命名规则为组织或公司域名反转加项目名称。**  
**artifactId：项目的模块名，通常与实际项目名称一致。模块的命名通常为项目名前缀加模块名。**  
**version：当前项目的版本号。**  
**packaging：定义项目的打包方式，可选值有jar、war、pom。默认为jar**

2.依赖关系

Maven的配置文件中dependencies元素包含了所有依赖的jar包，每一个jar包依赖使用dependency元素定义。

在声明一个jar包依赖时，除了指定groupId、artifactId、version这三项基本坐标外，还可以使用使用以下元素进行配置：

scope元素：指定依赖的范围

Maven有以下几种依赖范围：

compile：编译依赖范围(默认值），依赖在编译、测试、运行期间都有效。

test：测试依赖范围，只对测试的classpath有效，在编译或运行时无法使用指定为test的依赖包。

provided：已提供的依赖范围，只对编译和测试的classpath有效，运行期间不会使用这个依赖。例如servlet-api，在运行时容器已经提供，不需要再重复引入。

runtime：运行时依赖范围，编译时无效，只在测试和运行时使用这个依赖。

system：系统依赖范围，和provided范围一致，但是provided是使用容器提供依赖，system是使用系统提供依赖，需要指定依赖文件路径。

exclusions元素：排除传递性依赖

传递性依赖，是指依赖包对其他包的依赖，比如，我们依赖struts2-core，而strtus2-core需要依赖xwork-core、ognl等，Maven会将这些传递性依赖同时引入项目之中。这也是Maven的一大优点，简化了我们对jar包依赖的管理。而有时我们希望替换某个传递性依赖时，就需要使用exclusions排除掉这个传递性依赖，然后再添加我们自己要替换的依赖包。

3.聚合关系

用于将多个maven项目聚合为一个大的项目

比如目录结构如下：

.

|-- pom.xml

|-- module-a

`-- pom.xml

|-- module-b

`-- pom.xml

|-- module-c

`-- pom.xml

|-- foo-all

`-- pom.xml

那么总的pom.xml文件类似：

...

<modules>

<module>module-a</module>

<module>module-b</module>

<module>module-c</module>

<module>foo-all</module>

</modules>

</project>

<parent>

<groupId>com.huawei.inoc.wfm.task</groupId>

<artifactId>task-aggregator</artifactId>

<version>0.0.1-SNAPSHOT</version>

<relativePath>../task-aggregator</relativePath>

</parent>