

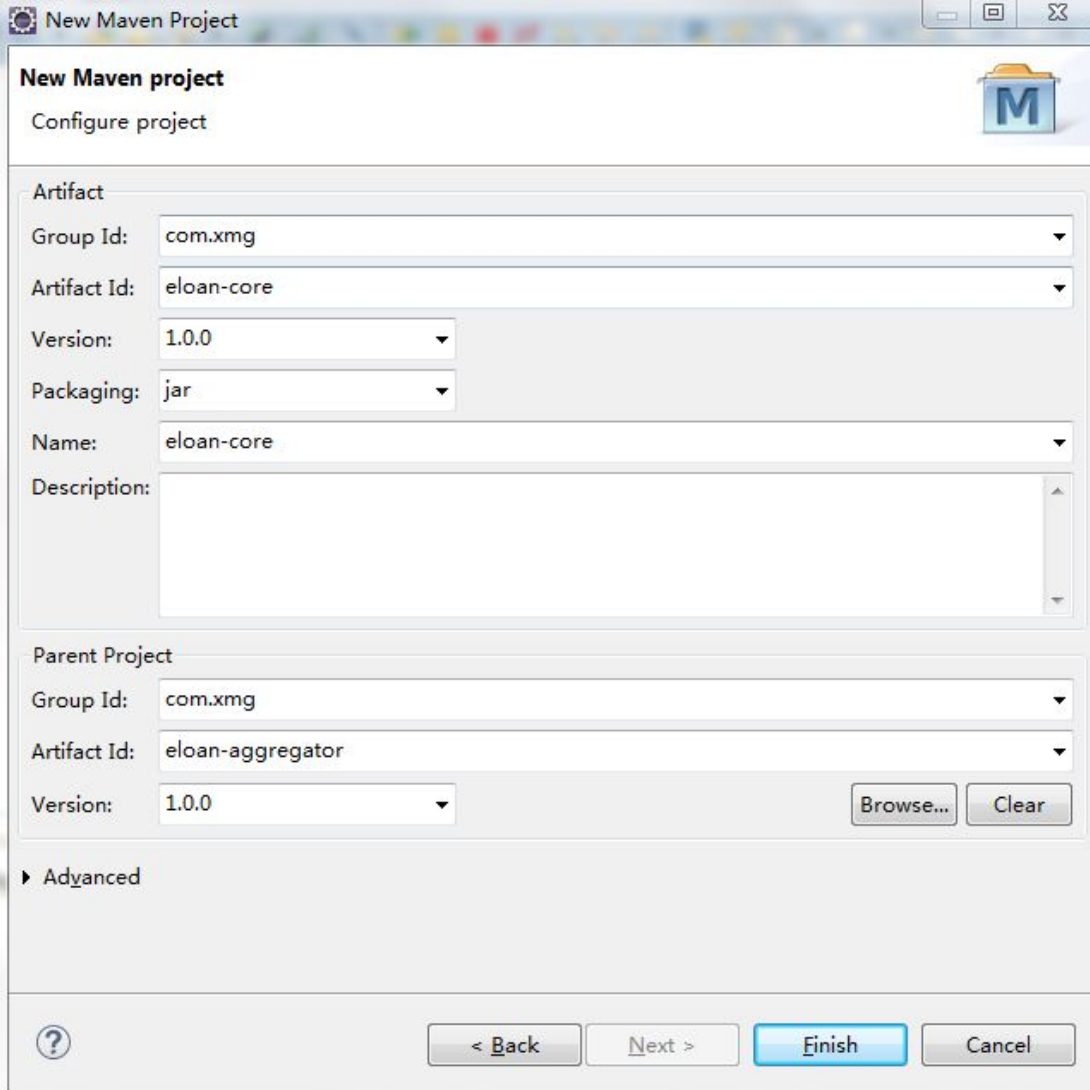
## Eloan 项目的 Maven 环境管理

创建主项目：

选择 maven 创建项目，注意在创建项目中，packing 选择 pom，代表这不是一个项目，只是一个 maven 的主项目，这个项目是用来集成多个子项目的；

给项目起名为：eloaan-aggregator

创建 eloaan-core 子项目：



注意：

- 1，在创建项目的时候，需要选择 Parent Project，定位到之前创建的 eloaan-aggregator，这样就可以让子项目的依赖继承于父项目；
- 2，在生成的 eloaan-core 项目中，我们可以清楚的看到相关 maven 继承的内容：

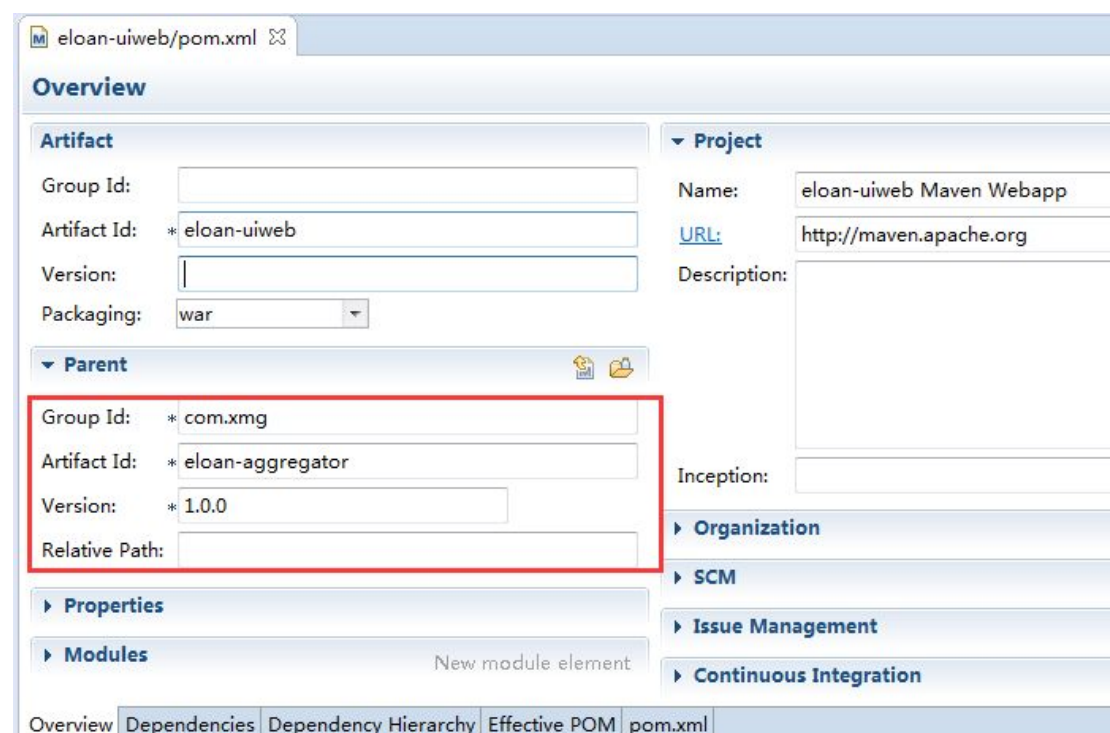
```
<parent>
  <groupId>com.xmg</groupId>
  <artifactId>eloaan-aggregator</artifactId>
```

```
<version>1.0.0</version>
</parent>
```

3, 注意, 如果已经选择了 parent 项目, 则子项目可以不需要写 groupId 和 version, 这两个属性可以通过父项目继承得到;

创建 eloan-uiweb 项目:

- 1, 创建一个 maven-web 项目;
- 2, 修改项目的 pom 文件:



同理, 按照创建 eloan-uiweb 的方式创建 eloan-mgrtool 项目;

## Maven 的依赖测试

在主项目的 pom 文件中添加:

```
<dependencies>
  <dependency>
    <groupId>junit</groupId>
    <artifactId>junit</artifactId>
    <version>4.12</version>
  </dependency>
</dependencies>
```

删除三个子项目中的所有对 junit 依赖;

刷新项目，可以看到三个子项目已经继承了主项目的 junit 配置；

## 项目环境配置

配置统一的编译版本：在主项目的 pom 中添加：

```
<build>
  <plugins>
    <plugin>
      <groupId>org.apache.maven.plugins</groupId>
      <artifactId>maven-compiler-plugin</artifactId>
      <configuration>
        <source>1.7</source>
        <target>1.7</target>
      </configuration>
    </plugin>
  </plugins>
</build>
```

刷新所有子项目即可；

配置统一编译字符集：在主项目 pom.xml 中添加

```
<properties>
  <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
</properties>
```