Maven的工程对象模型pom文件

Maven的pom.xml介绍

介绍： <http://blog.csdn.net/taiyangdao/article/details/52319644>

史上最全的maven的pom.xml文件详解：

<https://www.baidu.com/link?url=zzZQbpZNo88sj2u5oMgjyJpnQRZp8PPuihRSvArjl259rzXP2HaCgs0EUJV_6zYOAwg8nR33AnXegUAXSz2wL_&wd=&eqid=b4f5d4510002b213000000025a700b75>

# Maven的pom.xml文件

## pom工程对象模型

**POM (Project Object Model，工程对象模型)**，pom.xml文件是Maven项目的配置和管理核心。

## pom.xml文件的中配置信息

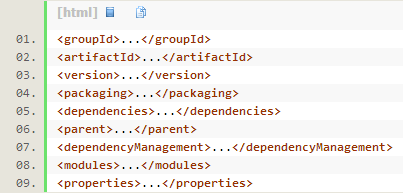
pom.xml文件包含大量配置信息，这些信息大致可以分为4类。

### POM的模型版本

**<modelVersion>4.0.0</modelVersion>**

说明：在Maven2和Maven3中，只支持4.0.0版本。

### 基本配置



说明：

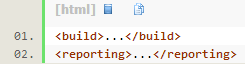
**groupId，构件所属的组织ID**

**artifactId，构件的ID**

**version，构件的当前版本**

其他配置参数将在后续文章中详述

### Build配置



### 环境配置



说明：

issueManagement，给出defect tracking system及其访问URL

system

url

ciManagement，给出Continuous Integration Management、其URL和notifier

system

url

notifiers，集成过程中发生事件，以某种方式（如mail）通知开发人员

scm，software configuration management

connection，用户使用的URI，能够只读地访问版本控制系统

developerConnection，开发人员使用URI，能够读写地访问版本控制系统

tag，项目当前的tag

url，可通过Web浏览器访问的公共网址

distributionManagement，构件的发布管理，详情见后续文章

prerequisites，POM执行的前提条件，目前只支持对Maven版本的要求

maven

mailingLists，开发人员或用户的邮件列表

mailingList

name

subscribe，订阅地址

unsubscribe，取消订阅地址

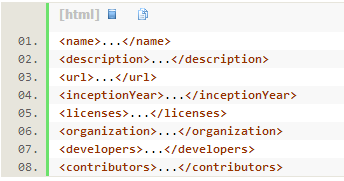
post，POST邮件的目的地址

archive，打包的邮件列表历史记录

otherArchives，镜像打包的邮件列表历史记录

其他参数将在后续文章中详述。

### 其他信息



说明：

name，项目的名称代号

description，项目的说明

url，项目的官网URL

inceptionYear，项目的开发年份

licenses，项目使用的License。其中可以包含多个license，license具体又包含如下子属性

name，license的名称

url，license可访问的URL地址

distribution，license发布的方式。repo表示可以直接从Maven库下载，manual表示必须手工安装

comments，对license的说明

organization，包含组织的name，组织的官网url

developers，其中的developer包含id, name, email, url, organization, organizationUrl, roles, timezone, properties属性

properties是可以自定义的各种必要属性

contributors，其中的contributor包含与developer基本相同的属性，除了没有id属性之外

# pom.xml中的标签

## <properties>标签：定义key-value

设定一些key-value值，其结构为<key>value</key>

<properties>

<project.build.version>2.5.5</project.build.version>

<profiles.dir>src/main/profiles</profiles.dir>

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

<spring-version>3.1.3.RELEASE</spring-version>

<lotus\_app-version>2.5.5</lotus\_app-version>

</properties>

在下面的dependency中就可以通过**${key}**调用：

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-beans</artifactId>

<version>**${spring-version}**</version>

<scope>compile</scope>

</dependency>

## </dependencies>中包括多个</dependency>

在出现jar包冲突的时候，可以在</dependency>中使用<exclusions>排除依赖，如

<dependency>

<groupId>com.wangyin.payment.risk</groupId>

<artifactId>portal-service</artifactId>

<version>1.3.9</version>

**<exclusions>**

**<exclusion>**

<groupId>org.apache.poi</groupId>

<artifactId>poi</artifactId>

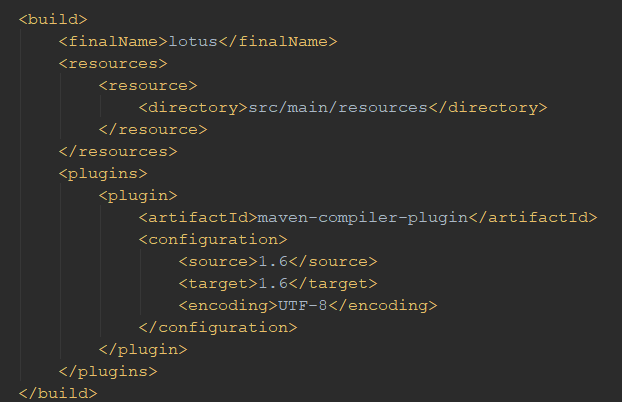
**</exclusion>**

**</exclusions>**

</dependency>

## build标签

配置项目的JRE编译版本、编码等。



## profiles详解

<https://blog.csdn.net/maozexijr/article/details/79074493>