

《开源软件设计与开发》课程总结

姓名：徐一文

学号：51195100020

对 Maven 的理解和实践

Maven 是 apache 的一个顶级项目，它的出现越来越影响着现在的众多的开源项目，不仅如此，很多公司的很多新项目都采用 Maven 提倡的方式进行管理。Maven 正逐渐侵入我们原先的管理项目的习惯，对于团队的管理，项目的构建，都是一种质的飞跃。利用 Maven 能帮助我们构建项目更科学、更有可维护性、更加模块化、模块功能更加职能单一、做出来的项目更加具有可复用性等等好处。当然我也是个刚学习 Maven 的人，我们一起来看看 Maven 到底为我们带来了什么好处。

1. Maven 的思想

目前我看到的 Maven 的主要功能是：

- 1) 项目构建；
- 2) 项目依赖管理；
- 3) 软件项目持续集成；
- 4) 版本管理；
- 5) 项目的站点描述信息管理；

由此可以看出 Maven 最主要的体现在了哪 2 个词？——“项目”和“管理”！加在一起就是项目管理。项目管理是一个很抽象的概念。项目管理既可以指技术上的管理手段，也可以指“以人为本”的非技术因素。诚然，无论是技术还是非技术，项目的成败最大的责任人其实就是项目经理。这里我们抛开人为因素不提，仅就技术上来说，Maven 提供了一种思想让团队更科学的管理、构建项目。用配置文件的方式对项目的描述、名称、版本号、项目依赖等等信息进行描述。使之项目描述结构清晰，任何人接手的成本比较低。在项目构建时，利用 Maven 的“约定大于配置”的思想，可以比 Ant 脚本构建项目省去不少配置文件的内容。而且一个项目可能依赖于其他的项目和第三方的组件才能顺利完成，Maven 提供了仓库的概念，让这些依赖项放进仓库中，项目想要从仓库中去取，其他项目组也需要，OK，从仓库中去取，不必每个人去开源项目的站点去苦苦搜寻了。如此人员的

成本、软件维护的成本、沟通的成本、硬件的成本都降下来了。客户笑了、公司老板也笑了、项目经理笑了、团队里面的人员也笑了、Apache 社区看到这么多人在用也笑了。给我的感觉是，现在的开源东西越来越向“敏捷开发”、“极限编程”的方向靠拢。通过 Maven 辅助管理项目，更能发现项目中哪些模块是重复的轮子。

2. 项目构建

Maven 这个基础设施落地下来，最重要的还是一个构建工具（虽然 Maven 创建者们不屑和 Ant 这样的构建工具相提并论）。所以介绍 Maven，必然还是需要先了解构建。

构建是什么呢？简单地说，构建就是软件项目生产的整个过程，比如这个过程应该包括：

- 文档和代码的生成（有些项目会使用代码自动生成工具，比如数据库访问代码的逆向工程）
- 代码的编译、测试和打包
- 打包好的代码进行分发或者部署

项目的构建可绝不仅仅是编译软件这件事情。除了写代码，在项目层面做的大部分工作，都包含在构建的过程中。有了 Maven，构建中的这些过程都能够进行良好的定义（模式、固化、共识，记住这些关键词哪），而且 Maven 能够帮我们串起来形成一个自动构建过程，这样比我们手动执行要高效得多。

3. 项目依赖管理

Java 最大的一个优势之一应该是整个生态中无数的框架和 API，我们创建实际的项目不可避免的都需要用到这些框架和 API，而它们通常都是以 JAR 包的形式提供。相信很多人都经历过 JAR Hell 的问题吧，事实上让一个项目所依赖的依赖的外部 jar 包保持正确的版本和最新的状态，是意见非常苦逼的事情。我们编译项目的时候，需要在 classpath 上存放依赖的 jar 包（不管直接使用 Eclipse

还是手动维护 Ant)。而且这些外部的 jar 包还会有其他依赖。你一定经历过递归地一个个去下载所有这些外部依赖，并且要确保下载的版本都是正确的，当项目越来越复杂的时候，这是意见极其麻烦的事情。

Maven 现在来拯救我们了，Maven 可以自动帮我们做依赖管理，我们需要做的就是 pom 文件里指定依赖 jar 包的名称、版本号，Maven 会自动下载，递归地去下载依赖的进一步依赖这件事情我们也不需要管了。

Maven 还提供一个非常方便的功能：快照依赖。快照依赖指的是那些还在开发中的内部依赖包。与其经常地更新版本号来获取最新版本，不如你直接依赖项目的快照版本。快照版本的每一个 build 版本都会被下载到本地仓库，即使该快照版本已经在本地仓库了。总是下载快照依赖可以确保本地仓库中的每一个 build 版本都是最新的。这对我们快速迭代开发是一个非常酷的特性。

课程反馈

《开源软件设计与开发》这门课程我觉得内容设计很充实，开源的方方面面都涉及到了，能很好的拓宽我们的视野。但是如果能增加课堂上与学生的互动，效果应该会更好。比如在课堂上打开一个开源项目，定位一个 issue，大家一起改 bug，然后提 pr，这样课堂的趣味性和参与感应该会提高。

最后，谢谢老师和助教这门课程和开源社区的付出！