



# Java私塾 《maven3》 ——深入浅出系列精品教程

10101010101010101010101010101



## 《maven3》——深入浅出系列精品教程

### maven初识—1

#### n maven是什么

maven这个词可以翻译为“知识的积累”，也可以翻译为“专家”或“内行”。作为apache组织中的一个颇为成功的开源项目，maven主要服务于基于java平台的**项目构建、依赖管理和项目信息管理**。

#### n maven能干什么

使项目构建构成更容易；  
提供统一构建系统(编译、测试、持续整合...);  
提供高质量的项目信息(依赖、报告、site...);  
提供开发的最佳实践指南；  
能无缝的加入新的特性；



## 《maven3》——深入浅出系列精品教程

### maven初识—2

#### n maven有什么【maven的核心概念】

项目对象模型(Project Object Model),

坐标(Coordinates),

项目生命周期(Project Lifecycle),

插件(plugin)和目标(goal),

依赖管理系统(Dependency Management System),

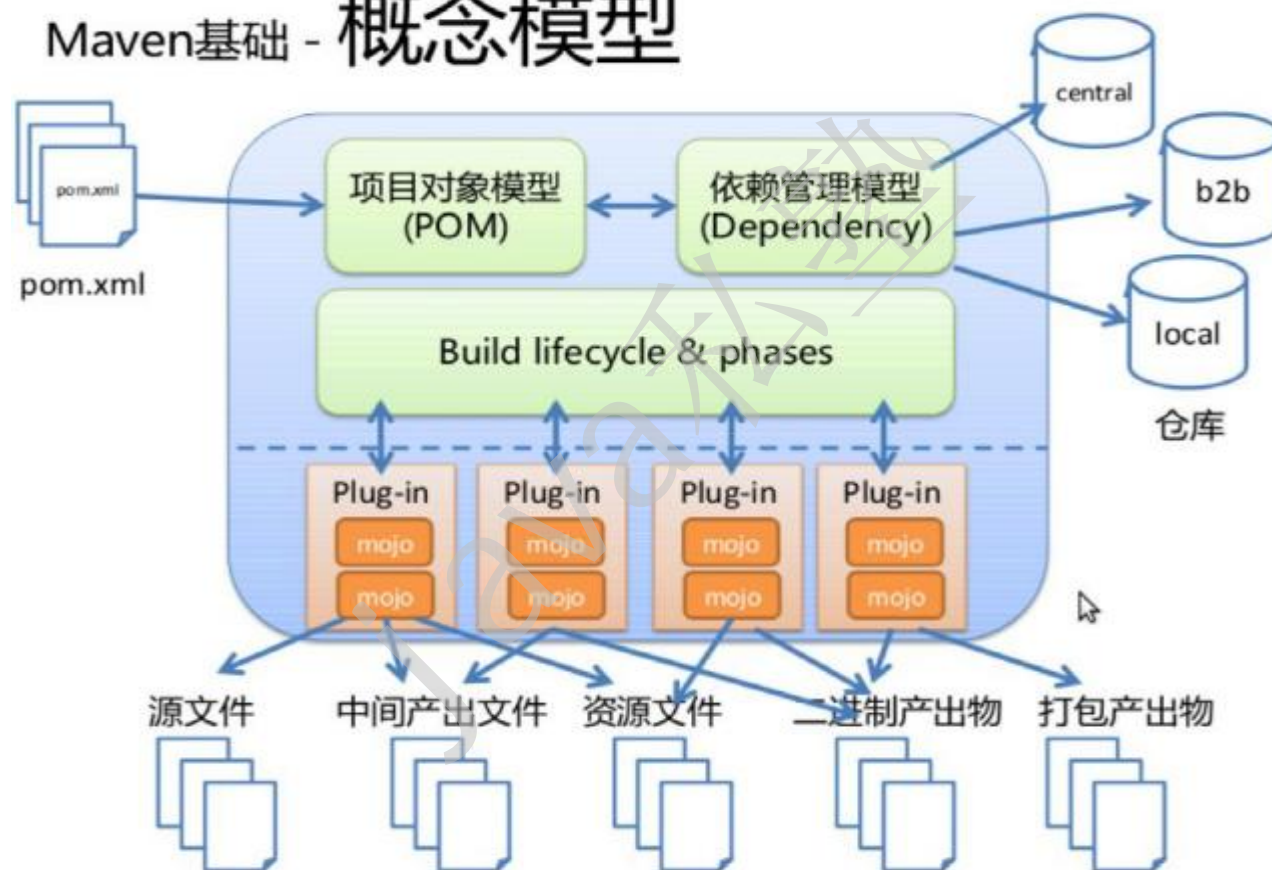
仓库管理(Repositories)。

真正高质量培训 签订就业协议

网 址: <http://www.javass.cn>  
咨询QQ: 460190900

## maven初识—3

### Maven基础 - 概念模型





## 《maven3》——深入浅出系列精品教程

### Hello world--1

#### n 准备我们需要的环境

下载maven的安装包apache-maven-3.0.3-bin.tar.gz，解压至任何目录。

设置环境变量M2\_HOME，设置为maven的安装路径；同时把maven的bin目录增加至环境变量path里。【我们可以看到，跟java的安装几乎一模一样】。

正常情况下，maven会到中央仓库去下载我们需要的构件或者插件；但是，现在在教室不能上网，所以需要将老师的私服加入到下载的配置中。建立C:\Documents and Settings\Administrator\.m2文件夹，其中Administrator为当前登录的用户。拷入准备好的settings.xml。其中

```
<profile>
    . . .
    <url>http://10.83.1.111:10080/nexus-webapp-1.9.1.1/content/groups/public/</url>
    <url>http://10.83.1.111:10080/nexus-webapp-1.9.1.1/content/groups/public/</url>
</profile>
```

代表老师的私服

真正高质量培训 签订就业协议

网 址: <http://www.javass.cn>  
咨询QQ: 460190900

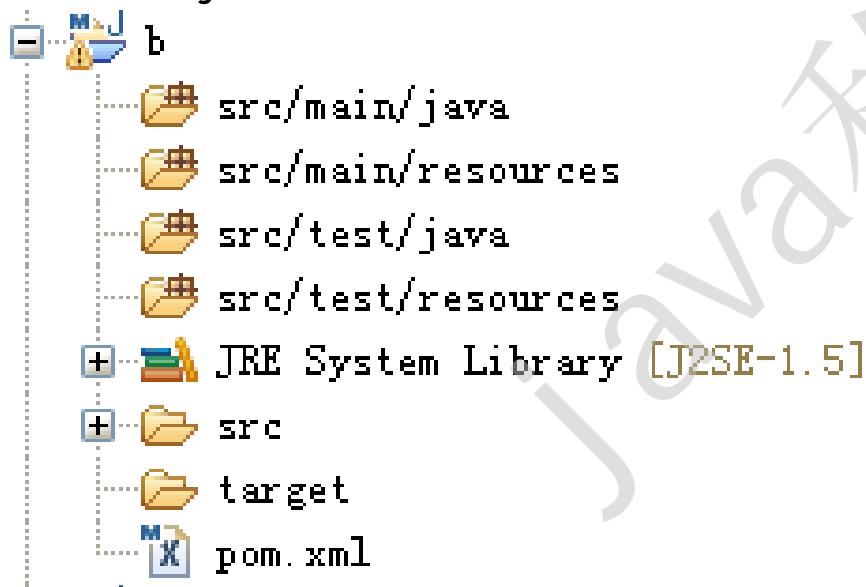


## 《maven3》——深入浅出系列精品教程

### HelloWorld--2

#### n 准备示例代码

在helloWorld的代码示例中，首先请大家**观察源代码的放置【代码的组织】**，其次请自己**观察pom文件**，至于代码里写的是什么，只是咱们原来学过的最简单的junit。





## 《maven3》——深入浅出系列精品教程

### Hello world--3

n 工程描述文件pom

```
<project xmlns="http://maven.apache.org/POM/4.0.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
    http://maven.apache.org/maven-v4_0_0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <groupId>cn.javass.study</groupId>
  <artifactId>hello-world</artifactId>
  <version>1.0-SNAPSHOT</version>
  <name>Maven Quick Start Archetype</name>
```

真正高质量培训 签订就业协议

网 址: <http://www.javass.cn>  
咨询QQ: 460190900





## 《maven3》——深入浅出系列精品教程

### Hello world--4

```
<dependencies>
  <dependency>
    <groupId>junit</groupId>
    <artifactId>junit</artifactId>
    <version>3.7.1</version>
    <scope>test</scope>
  </dependency>
</dependencies>
</project>
```





## 《maven3》——深入浅出系列精品教程

### Hello world--5

**n** 常见的mvn命令

mvn clean test

mvn clean package

mvn clean install

mvn是命令名

clean说明要清空所有的配置文件

test说明要运行单元测试

package说明要打包

install 安装到本地仓库

真正高质量培训 签订就业协议

网 址: <http://www.javass.cn>  
咨询QQ: 460190900



## 《maven3》——深入浅出系列精品教程

### 坐标

#### n 坐标

Maven的世界中拥有数量非常巨大的构件，也就是我们平时用的一些jar、war等文件。Maven定义了这样一组规则：世界上任何一个构件都可以使用Maven坐标唯一标识。Maven坐标的元素包括groupId、artifactId、version、packaging、classifier。

groupId：定义当前Maven项目隶属的实际项目。groupId的表示方式与java包名的表示方式类似，通常与域名反向一一对应。

artifactId：该元素定义实际项目中的一个Maven项目/模块。

version：版本【可以分成稳定版本和快照版本】。

packaging：打包方式。如：jar、war。

classifier：不能直接定义，用来表示构件到底用于何种jdk版本。



## 《maven3》——深入浅出系列精品教程

### pom、依赖

#### n pom

POM(Project Object Model):Maven的核心文件，位于每个工程的根目录中，指示Maven如何工作的元数据文件，类似于Ant中的build.xml文件。

#### n 依赖(Dependency)

为了能够构建或运行，Java工程一般会依赖其它的包。在Maven中，这些被依赖的包就被称为dependency。dependency一般是其它工程的坐标。

依赖具有传递性。



## 《maven3》——深入浅出系列精品教程

### mvn命令--1

#### n 生命周期

项目的生命周期是指软件开发人员每天都在对项目进行清理、编译以及部署。虽然大家都在不停的做构建工作，但公司 and 公司间、项目和项目间，往往使用不同的方式做类似的工作。

Maven的生命周期就是为了所有的构建过程进行抽象和统一。这个生命周期包含了项目的清理、初始化、编译、测试、打包、集成测试、验证、部署和站点生成等几乎所有构建步骤。

Maven拥有三套相互独立的生命周期，他们分别为clean、default和site。clean生命周期的目的是清理项目，default生命周期的目的是构建项目，而site生命周期的目的是建立项目站点。



## 《maven3》——深入浅出系列精品教程

### mvn命令--2

#### n 阶段【phase】

每个生命周期包含一些阶段，**这些阶段是有顺序的，并且后面的阶段依赖于前面的阶段**，用户和Maven最直接的交互方式就是调用这些生命周期阶段。

较之于生命周期阶段的前后依赖关系，三套生命周期本身是相互独立的，用户可以仅仅调用clean生命周期的某个阶段，或者仅仅调用default生命周期的某个阶段，而不会对其他生命周期产生任何影响。

clean生命周期包含三个阶段：

pre-clean、**clean**、post-clean

default生命周期包含很多阶段：

validate、initialize、generate-sources、process-sources、  
generate-resources、process-resources、**compile**、process-classes、  
generate-test-sources、process-test-sources、generate-test-resources、  
process-test-resources、test-compile、process-test-classes、**test**、  
prepare-package、**package**、pre-integration-test、integration-test、  
post-integration-test、verify、**install**、**deploy**



## 《maven3》——深入浅出系列精品教程

### mvn命令--3

site生命周期包含四个阶段：

pre-site、site、post-site、site-deploy

#### n 插件及其目标【goal】

Maven的核心仅仅定义了抽象的生命周期，具体的任务是交由插件完成的，插件以独立的构建形式存在。

对于插件本身，为了能够复用代码，它往往能够完成多个任务。例如maven-dependency-plugin插件，能够基于项目以来做很多事情。比如，能够分析项目依赖，找到无用的或者重复的依赖；还能够列出项目的依赖树。这些功能往往背后有很多可以复用的代码，因此，可以把这些功能聚集在一个插件里，每个功能就是一个插件目标。

我们原来仅仅学过通过前缀调用插件，现在可以用冒号来指定调用插件的某个具体目标了，比如：mvn dependency:tree。冒号前面是插件的前缀，冒号后是该插件的目标。



## 《maven3》——深入浅出系列精品教程

### 仓库

在Maven世界中，任何一个依赖、插件或者项目的构建输出，都可以称为构件。任何一个构件都有一组坐标唯一标识。

得益于坐标机制，任何Maven项目使用任何一个构件的方式都是完全相同的。在此基础上，Maven可以在某个位置统一存储所有Maven项目共享的构件，这个统一的位置就是仓库。

对于Maven来说，仓库只分为两大类：本地仓库和远程仓库。当Maven根据坐标寻找构件的时候，它首先会查看本地仓库，如果本地仓库存在此构件，则直接使用；如果本地仓库不存在此构件，Maven就会去远程仓库查找，发现需要的构件之后，下载到本地仓库再使用。

【以上概念，对比一下hibernate中的一级缓存。】

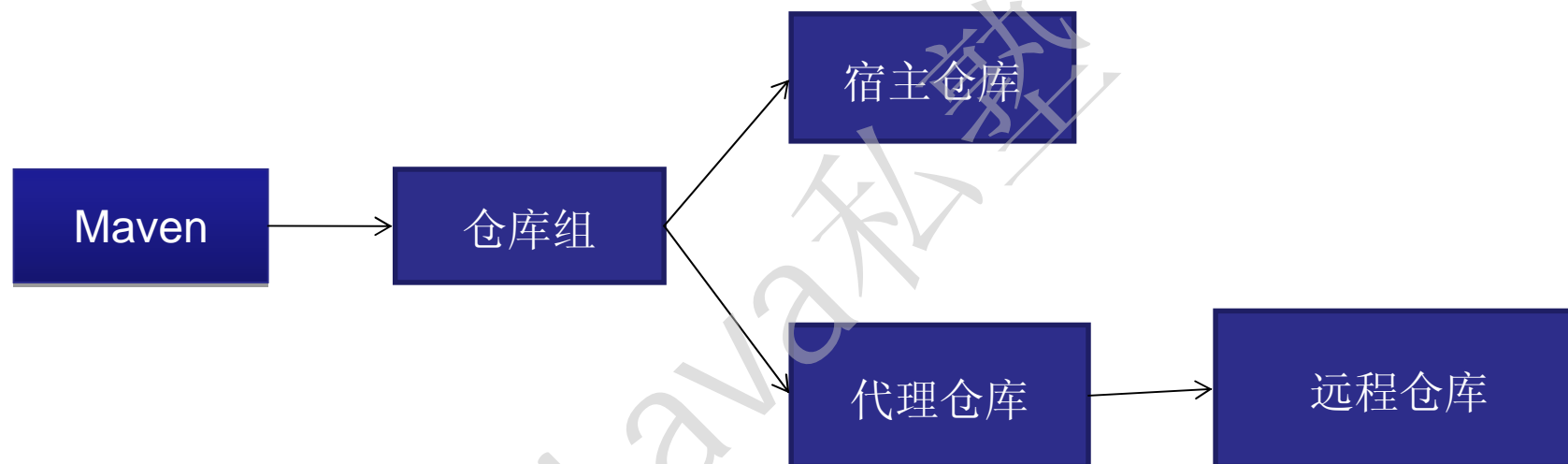
私服是一种特殊的远程仓库，它是架设在局域网内的仓库服务，私服代理广域网上的远程仓库，供局域网内的Maven用户使用。





## 《maven3》——深入浅出系列精品教程

私服--nexus



真正高质量培训 签订就业协议

网 址: <http://www.javass.cn>  
咨询QQ: 460190900



## 《maven3》——深入浅出系列精品教程

### 私服--nexus

首先从<http://nexus.sonatype.org/downloads/>下载最新版本的Nexus的war包形式版本。

登录时其默认的用户名为admin，密码为admin123。

使用nexus的时候，如果想支持搜索；可以从远程仓库下载其索引，这当然非常慢，我们可以使用gui的方式在下载好的构件基础上重建索引。

nexus支持非常全面的搜索方式。GAV搜索：通过GroupId、ArtifactId和Version进行搜索；全类名搜索；关键词搜索。

配置客户机使用nexus时，需要在settings.xml文件中加入相应的配置，从nexus的仓库组中下载构件和插件。【在helloworld中已经说明，这里不再举例】



## 《maven3》——深入浅出系列精品教程

### mvn的web项目

WTP项目【也就是咱们的eclipse下的dynamic web project】

src/main/java:源码目录

src/main/resources:资源目录(如存放log4j.properties)

src/main/webapp:web目录【它下面就是WEB-INF】

src/test/java:测试源码目录

src/test/resources:测试资源目录

target:编译结果目录

真正高质量培训 签订就业协议

网 址: <http://www.javass.cn>  
咨询QQ: 460190900



## 《maven3》——深入浅出系列精品教程

### m2eclipse

#### n 安装插件

将获得的m2eclipse.rar解压到eclipse的dropins文件夹下，重启eclipse，就可以看到maven插件了。

#### n 新建项目

选择【New】->【Project】->【Maven Project】，选择create a simple project，填入GAV，选择打包方式jar/war。

如果要新建web工程，请选择打包方式为war，为了让项目能在eclipse相关的tomcat下运行，还需要一点点麻烦的配置。选择项目的属性，添加dynamic web project支持【project facets】；再次选择项目的属性，重整项目的部署【deployment assembly】。这部分操作非常非常麻烦，请注意看老师的演示。

#### n 添加依赖

在项目上右击【maven】->【add dependency】。

#### n 运行命令行

在项目上右击【Run as】。



# Java私塾 《maven3》

## ——深入浅出系列精品教程

10101010101010101010101010101



## 《maven3》——深入浅出系列精品教程

### maven初识—1

#### n maven是什么

maven这个词可以翻译为“知识的积累”，也可以翻译为“专家”或“内行”。作为apache组织中的一个颇为成功的开源项目，maven主要服务于基于java平台的**项目构建、依赖管理和项目信息管理**。

#### n maven能干什么

使项目构建构成更容易；  
提供统一构建系统(编译、测试、持续整合...);  
提供高质量的项目信息(依赖、报告、site...);  
提供开发的最佳实践指南；  
能无缝的加入新的特性；



## 《maven3》——深入浅出系列精品教程

### maven初识—2

#### n maven有什么【maven的核心概念】

项目对象模型(Project Object Model),

坐标(Coordinates),

项目生命周期(Project Lifecycle),

插件(plugin)和目标(goal),

依赖管理系统(Dependency Management System),

仓库管理(Repositories)。

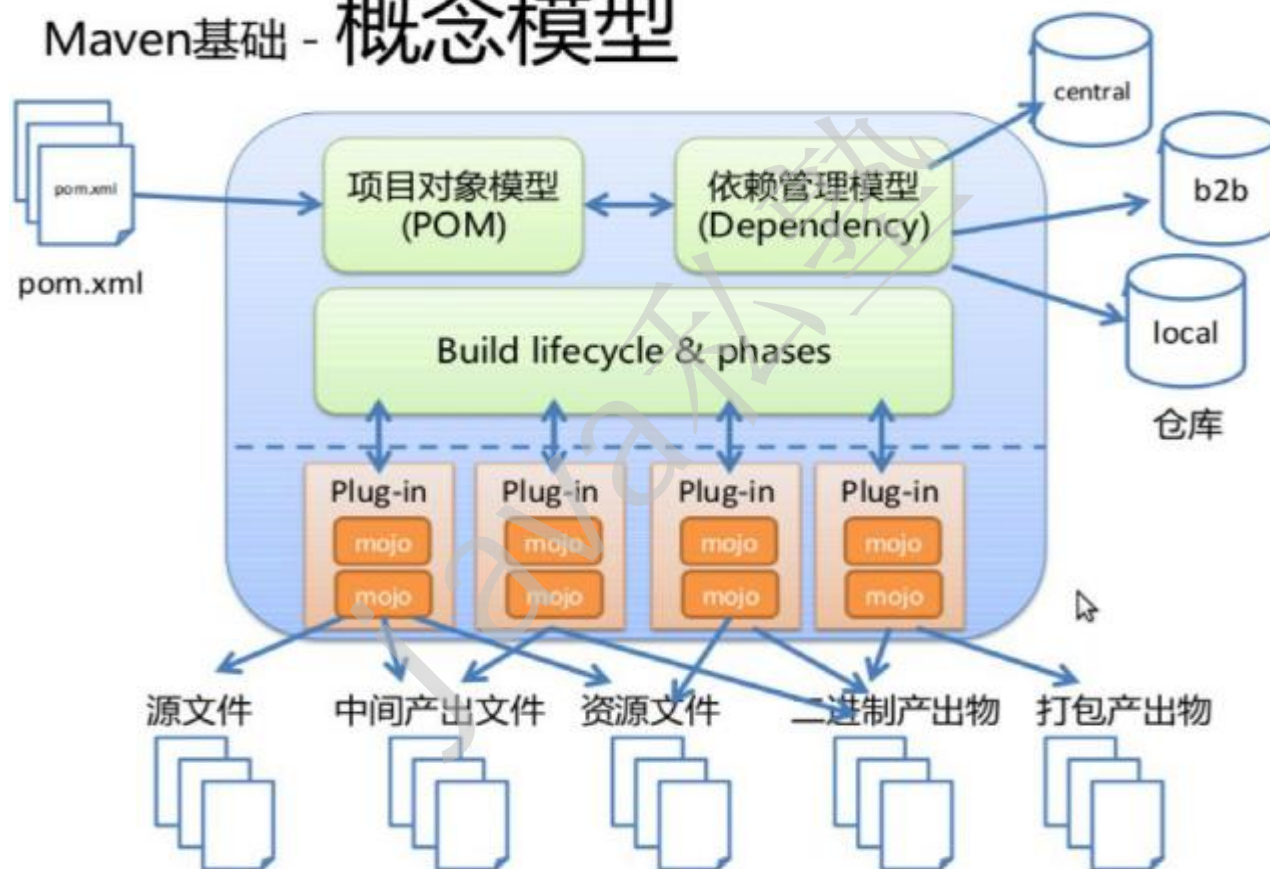
真正高质量培训 签订就业协议

网 址: <http://www.javass.cn>  
咨询QQ: 460190900



## maven初识—3

### Maven基础 - 概念模型





## 《maven3》——深入浅出系列精品教程

### Hello world--1

#### n 准备我们需要的环境

下载maven的安装包apache-maven-3.0.3-bin.tar.gz，解压至任何目录。

设置环境变量M2\_HOME，设置为maven的安装路径；同时把maven的bin目录增加至环境变量path里。【我们可以看到，跟java的安装几乎一模一样】。

正常情况下，maven会到中央仓库去下载我们需要的构件或者插件；但是，现在在教室不能上网，所以需要将老师的私服加入到下载的配置中。建立C:\Documents and Settings\Administrator\.m2文件夹，其中Administrator为当前登录的用户。拷入准备好的settings.xml。其中

```
<profile>
    . . .
    <url>http://10.83.1.111:10080/nexus-webapp-1.9.1.1/content/groups/public/</url>
    <url>http://10.83.1.111:10080/nexus-webapp-1.9.1.1/content/groups/public/</url>
</profile>
```

代表老师的私服

真正高质量培训 签订就业协议

网 址: <http://www.javass.cn>  
咨询QQ: 460190900

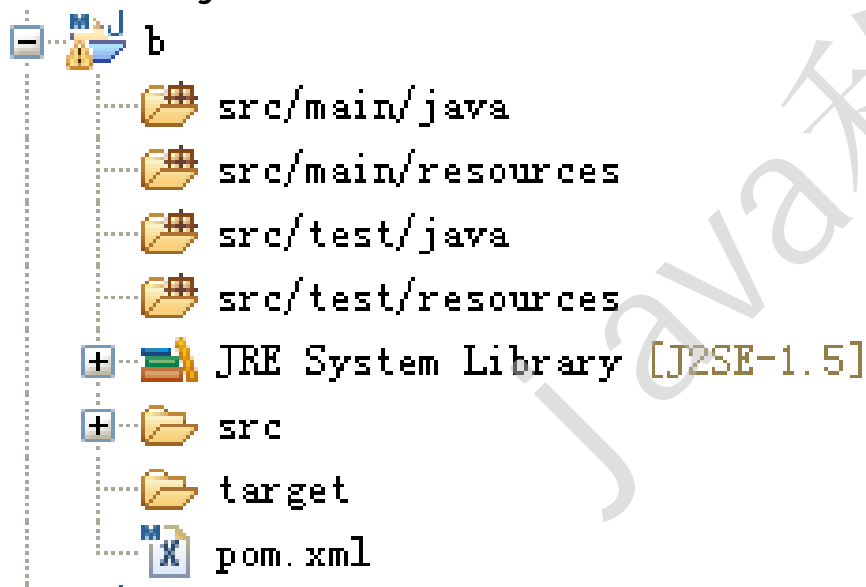


## 《maven3》——深入浅出系列精品教程

### HelloWorld--2

#### n 准备示例代码

在helloWorld的代码示例中，首先请大家**观察源代码的放置【代码的组织】**，其次请自己**观察pom文件**，至于代码里写的是什么，只是咱们原来学过的最简单的junit。





## 《maven3》——深入浅出系列精品教程

### Hello world--3

n 工程描述文件pom

```
<project xmlns="http://maven.apache.org/POM/4.0.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
    http://maven.apache.org/maven-v4_0_0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <groupId>cn.javass.study</groupId>
  <artifactId>hello-world</artifactId>
  <version>1.0-SNAPSHOT</version>
  <name>Maven Quick Start Archetype</name>
```

真正高质量培训 签订就业协议

网 址: <http://www.javass.cn>  
咨询QQ: 460190900



## 《maven3》——深入浅出系列精品教程

### Hello world--4

```
<dependencies>
  <dependency>
    <groupId>junit</groupId>
    <artifactId>junit</artifactId>
    <version>3.7.1</version>
    <scope>test</scope>
  </dependency>
</dependencies>
</project>
```



## 《maven3》——深入浅出系列精品教程

### Hello world--5

n 常见的mvn命令

mvn clean test

mvn clean package

mvn clean install

mvn是命令名

clean说明要清空所有的配置文件

test说明要运行单元测试

package说明要打包

install 安装到本地仓库

真正高质量培训 签订就业协议

网 址: <http://www.javass.cn>  
咨询QQ: 460190900



## 《maven3》——深入浅出系列精品教程

### 坐标

#### n 坐标

Maven的世界中拥有数量非常巨大的构件，也就是我们平时用的一些jar、war等文件。Maven定义了这样一组规则：世界上任何一个构件都可以使用Maven坐标唯一标识。Maven坐标的元素包括groupId、artifactId、version、packaging、classifier。

groupId：定义当前Maven项目隶属的实际项目。groupId的表示方式与java包名的表示方式类似，通常与域名反向一一对应。

artifactId：该元素定义实际项目中的一个Maven项目/模块。

version：版本【可以分成稳定版本和快照版本】。

packaging：打包方式。如：jar、war。

classifier：不能直接定义，用来表示构件到底用于何种jdk版本。





## 《maven3》——深入浅出系列精品教程

### pom、依赖

#### n pom

POM(Project Object Model):Maven的核心文件，位于每个工程的根目录中，指示Maven如何工作的元数据文件，类似于Ant中的build.xml文件。

#### n 依赖(Dependency)

为了能够构建或运行，Java工程一般会依赖其它的包。在Maven中，这些被依赖的包就被称为dependency。dependency一般是其它工程的坐标。

依赖具有传递性。



## 《maven3》——深入浅出系列精品教程

### mvn命令--1

#### n 生命周期

项目的生命周期是指软件开发人员每天都在对项目进行清理、编译以及部署。虽然大家都在不停的做构建工作，但公司 and 公司间、项目和项目间，往往使用不同的方式做类似的工作。

Maven的生命周期就是为了所有的构建过程进行抽象和统一。这个生命周期包含了项目的清理、初始化、编译、测试、打包、集成测试、验证、部署和站点生成等几乎所有构建步骤。

Maven拥有三套相互独立的生命周期，他们分别为clean、default和site。clean生命周期的目的是清理项目，default生命周期的目的是构建项目，而site生命周期的目的是建立项目站点。



## 《maven3》——深入浅出系列精品教程

### mvn命令--2

#### n 阶段【phase】

每个生命周期包含一些阶段，**这些阶段是有顺序的，并且后面的阶段依赖于前面的阶段**，用户和Maven最直接的交互方式就是调用这些生命周期阶段。

较之于生命周期阶段的前后依赖关系，三套生命周期本身是相互独立的，用户可以仅仅调用clean生命周期的某个阶段，或者仅仅调用default生命周期的某个阶段，而不会对其他生命周期产生任何影响。

clean生命周期包含三个阶段：

pre-clean、**clean**、post-clean

default生命周期包含很多阶段：

validate、initialize、generate-sources、process-sources、  
generate-resources、process-resources、**compile**、process-classes、  
generate-test-sources、process-test-sources、generate-test-resources、  
process-test-resources、test-compile、process-test-classes、**test**、  
prepare-package、**package**、pre-integration-test、integration-test、  
post-integration-test、verify、**install**、**deploy**



## 《maven3》——深入浅出系列精品教程

### mvn命令--3

site生命周期包含四个阶段：

pre-site、**site**、post-site、**site-deploy**

#### n 插件及其目标【goal】

Maven的核心仅仅定义了抽象的生命周期，具体的任务是交由插件完成的，插件以独立的构建形式存在。

对于插件本身，为了能够复用代码，它往往能够完成多个任务。例如maven-dependency-plugin插件，能够基于项目以来做很多事情。比如，能够分析项目依赖，找到无用的或者重复的依赖；还能够列出项目的依赖树。这些功能往往背后有很多可以复用的代码，因此，可以把这些功能聚集在一个插件里，每个功能就是一个插件目标。

我们原来仅仅学过通过前缀调用插件，现在可以用冒号来指定调用插件的某个具体目标了，比如：mvn dependency:tree。冒号前面是插件的前缀，冒号后是该插件的目标。



## 《maven3》——深入浅出系列精品教程

### 仓库

在Maven世界中，任何一个依赖、插件或者项目的构建输出，都可以称为构件。任何一个构件都有一组坐标唯一标识。

得益于坐标机制，任何Maven项目使用任何一个构件的方式都是完全相同的。在此基础上，Maven可以在某个位置统一存储所有Maven项目共享的构件，这个统一的位置就是仓库。

对于Maven来说，仓库只分为两大类：本地仓库和远程仓库。当Maven根据坐标寻找构件的时候，它首先会查看本地仓库，如果本地仓库存在此构件，则直接使用；如果本地仓库不存在此构件，Maven就会去远程仓库查找，发现需要的构件之后，下载到本地仓库再使用。

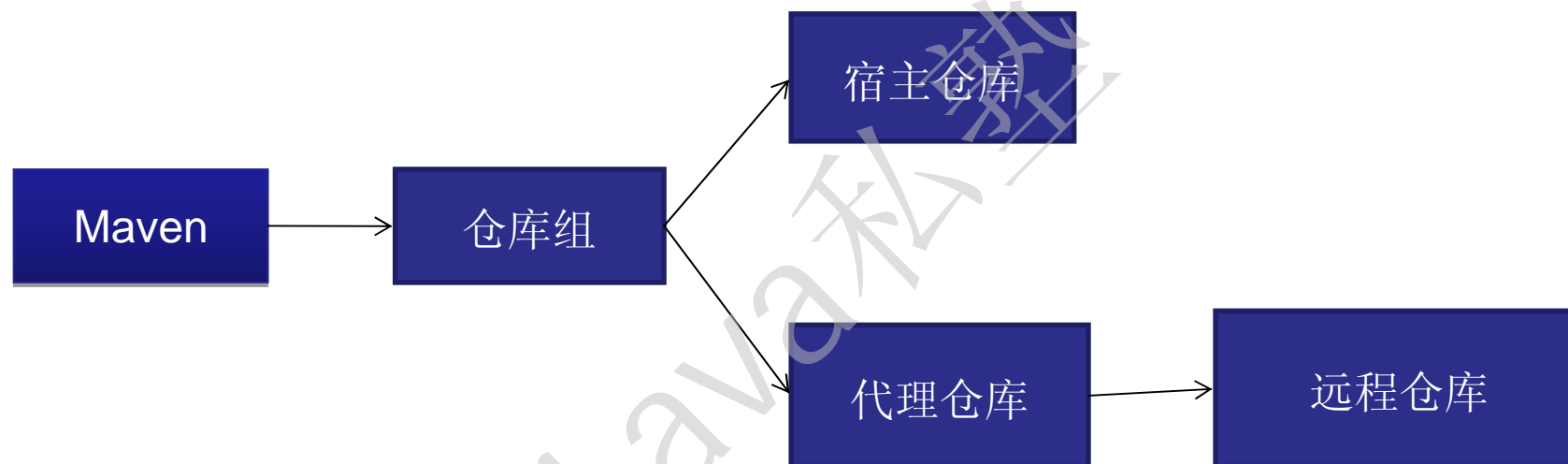
【以上概念，对比一下hibernate中的一级缓存。】

私服是一种特殊的远程仓库，它是架设在局域网内的仓库服务，私服代理广域网上的远程仓库，供局域网内的Maven用户使用。



## 《maven3》——深入浅出系列精品教程

私服--nexus



真正高质量培训 签订就业协议

网 址: <http://www.javass.cn>  
咨询QQ: 460190900



## 《maven3》——深入浅出系列精品教程

### 私服--nexus

首先从<http://nexus.sonatype.org/downloads/>下载最新版本的Nexus的war包形式版本。

登录时其默认的用户名为admin，密码为admin123。

使用nexus的时候，如果想支持搜索；可以从远程仓库下载其索引，这当然非常慢，我们可以使用gui的方式在下载好的构件基础上重建索引。

nexus支持非常全面的搜索方式。GAV搜索：通过GroupId、ArtifactId和Version进行搜索；全类名搜索；关键词搜索。

配置客户机使用nexus时，需要在settings.xml文件中加入相应的配置，从nexus的仓库组中下载构件和插件。【在helloworld中已经说明，这里不再举例】





## 《maven3》——深入浅出系列精品教程

### mvn的web项目

WTP项目【也就是咱们的eclipse下的dynamic web project】

src/main/java:源码目录

src/main/resources:资源目录(如存放log4j.properties)

src/main/webapp:web目录【它下面就是WEB-INF】

src/test/java:测试源码目录

src/test/resources:测试资源目录

target:编译结果目录

真正高质量培训 签订就业协议

网 址: <http://www.javass.cn>  
咨询QQ: 460190900



## 《maven3》——深入浅出系列精品教程

### m2eclipse

#### n 安装插件

将获得的m2eclipse.rar解压到eclipse的dropins文件夹下，重启eclipse，就可以看到maven插件了。

#### n 新建项目

选择【New】->【Project】->【Maven Project】，选择create a simple project，填入GAV，选择打包方式jar/war。

如果要新建web工程，请选择打包方式为war，为了让项目能在eclipse相关的tomcat下运行，还需要一点点麻烦的配置。选择项目的属性，添加dynamic web project支持【project facets】；再次选择项目的属性，重整项目的部署【deployment assembly】。这部分操作非常非常麻烦，请注意看老师的演示。

#### n 添加依赖

在项目上右击【maven】->【add dependency】。

#### n 运行命令行

在项目上右击【Run as】。