**传统项目开发**

1，模块多，手工管理

2，需要许多jar包，需要下载一堆jar包

3，还需要管理jar包的版本（版本不对，项目无法启动）

4，还需要管理jar包之间的依赖关系，a.jar中需要用到b.jar的类

那么我们必须先下载b.jar才可以使用a.jar

**Maven开发**

1，管理jar包

2，自动下载jar包和文档

3，管理jar包版本

4，管理jar包的依赖，我们要使用a.jar，但是a.jar又依赖b.jar

那么Maven下载a.jar时候，也会自动下载b.jar

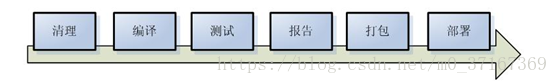
5，Maven可以编译文件，把java文件编译为class文件

6，Maven可以测试代码是否正确

7，打包文件，形成jar文件或war文件

8，帮助部署项目

**构建：项目的构建——Maven可以帮助我们进行项目构建**



Maven提供的构建功能——每个阶段一个命令就可以完成

1，清理：清楚项目残留的class文件（clean）

2，编译：Maven可以同时把上千个java文件编译为class文件（compile）

3，测试，Maven可以同时执行多个测试代码，测试功能

4，报告：Maven可以生成测试结果的文件

5，打包：Maven可以把项目中的所有class文件，配置文件放到一个压缩文件中（jar或war）（package）

6，部署：将项目部署到服务器上

**Maven的核心概念**

1，POM

Maven项目里面有一个pom.xml文件，pom.xml文件是整个项目对象的模型

Maven通过pom.xml来完成项目的构建，管理jar包

2，约定的项目目录结构

Maven项目的目录位置和文件位置都是有规定的

3，坐标

坐标是jar包的唯一标识，Maven通过pom.xml中的坐标在仓库中找到对应的jar包

4，依赖管理

管理项目用到的jar包

5，仓库管理

资源存放的位置

6，生命周期

Maven构建项目的过程

7，插件和目标

执行Maven构建用到的工具——插件

8，聚合

9，继承

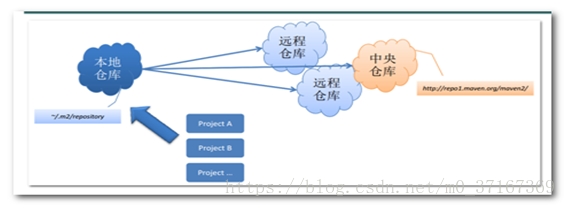
**Maven的目录**



**Maven仓库**

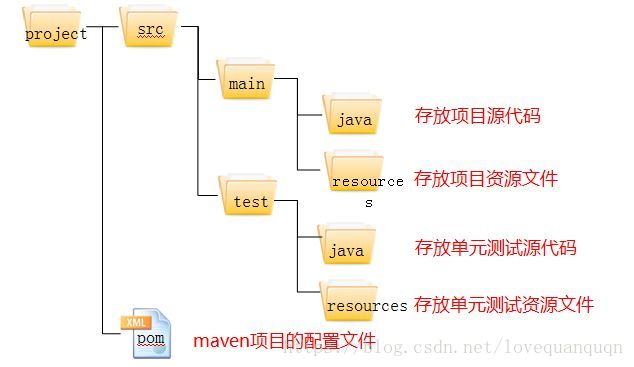
本地项目工作需要某个jar包时，会检查本地仓库有没有，如果没有，会让Maven从远程仓库下载jar包回来，放在本地仓库中

在maven软件中内置一个远程仓库地址http://repo1.maven.org/maven2 ，它是中央仓库，服务于整个互联网，它是由Maven团队自己维护，里面存储了非常全的jar包，它包含了世界上大部分流行的开源项目构件

默认的本地仓库如下图

、

**Maven约定的项目的目录结构**



src/main/java —— 存放项目的.java文件

src/main/resources —— 存放项目资源文件，如spring, hibernate配置文件

src/test/java —— 存放所有单元测试.java文件，如JUnit测试类

src/test/resources —— 测试资源文件

target —— 项目输出位置，编译后的class文件会输出到此目录

pom.xml——maven项目核心配置文件

**坐标**

在Maven中，坐标是Jar包的唯一标识，Maven通过pol.xml文件中的坐标在仓库中找到项目对应的Jar包

如下代码中，groupId和artifactId构成了一个Jar包的坐标。

**<dependency>**

**<groupId>cn.missbe.web.search</groupId>**

**<artifactId>resource-search</artifactId>**

**<packaging>jar</packaging>**

**<version>1.0-SNAPSHOT</version>**

**</dependency>**

groupId:所需Jar包的项目名

artifactId:所需Jar包的模块名

version:所需Jar包的版本号

**依赖冲突**

如果项目中多个Jar包引用了同一个Jar包，会产生依赖冲突，Maven采用了2种方法来避免依赖冲突——所以Maven中是不会有依赖冲突的

1，短路优先

本项目——>A.jar——>B.jar——C.jar

本项目——>C.jar——>X.jar

Maven只会引用路径最短的jar

2，声明优先

如果引用路径长度相同，那么谁在pom.xml中先被声明就是用谁

依赖冲突

若项目中多个Jar同时引用了相同的Jar时，会产生依赖冲突，但Maven采用了两种避免冲突的策略，因此在Maven中是不存在依赖冲突的。

**聚合**

将多个项目同时运行——聚合

通过在pom.xml中作出如下配置即可实现聚合

<modules>

<module>web-connection-pool</module>

<module>web-java-crawler</module>

</modules>

**继承**

聚合多个项目时，如果这些被聚合的项目中需要引入相同的jar包，那么我们将这些jar包写在父pom中，各个子项目的pom继承父pom即可

**父项目中的父pom配置：将需要被继承的Jar包的坐标放入标签即可。**

<dependencyManagement>

<dependencies>

<dependency>

<groupId>cn.missbe.web.search</groupId>

<artifactId>resource-search</artifactId>

<packaging>pom</packaging>

<version>1.0-SNAPSHOT</version>

</dependency>

</dependencies>

</dependencyManagement>

**子项目中的子pom配置：自动继承父pom中的jar包**

**<parent>**

**<groupId>父pom所在项目的groupId</groupId>**

**<artifactId>父pom所在项目的artifactId</artifactId>**

**<version>父pom所在项目的版本号</version>**

**</parent>**

<parent>

<groupId>cn.missbe.web.search</groupId>

<artifactId>resource-search</artifactId>

<version>1.0-SNAPSHOT</version>

</parent>

**pom详解**

pom.xml是Maven的核心，项目需要什么jar包就直接在pom.xml中配置，当编译项目时Maven读取pom.xml，根据jar包的坐标从仓库中加载jar包