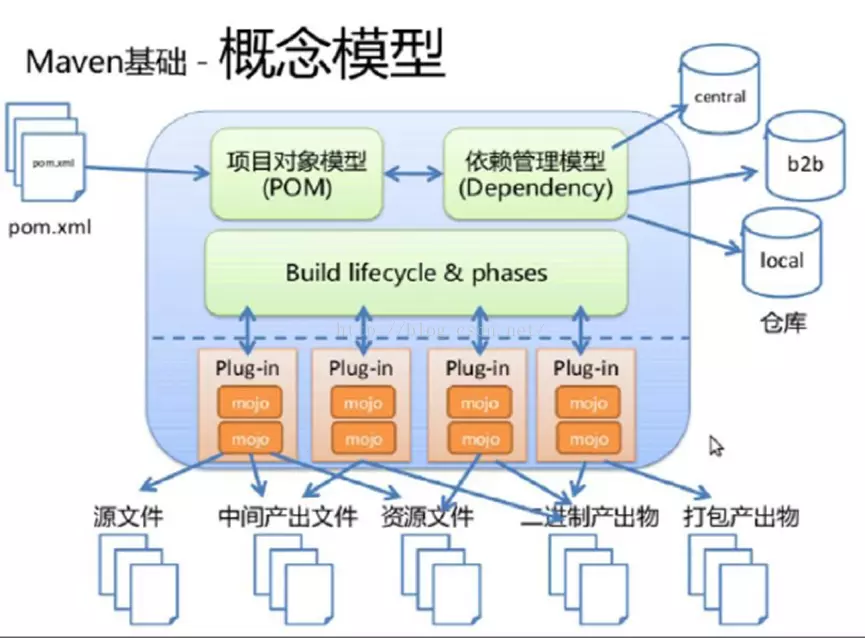
Maven本质是一个插件框架，它的核心并不执行任何具体的构建任务，所有这些任务都交给插件来完成。



Maven有以下三个标准的生命周期：

* Clean：项目清理的处理
* Default（或build）：项目部署的处理
* Site：项目站点文档创建的处理

每个生命周期中都包含着一系列的阶段（phase）。这些phase就相当于Maven提供的统一的接口，然后这些phase的实现由Maven的插件来完成。

命令mvn clean中，clean对应的就是Clean生命周期中的clean阶段。但是clean阶段的具体操作是由maven-clean-plugin来实现的。所以说Maven声明周期的每一个阶段的具体实现都是由Maven插件实现的。

Maven实际上是一个依赖插件执行的框架，Maven插件通常被用来：

* 创建jar文件
* 创建war文件
* 编译代码文件
* 代码单元测试
* 创建工程文档
* 创建工程报告

插件通常提供了一个目标的集合，可以使用下面的语法执行：

mvn [plugin-name]:[goal-name]

如编译命令：mvn compiler:compile

maven提供两种类型的插件：

|  |  |
| --- | --- |
| 类型 | 描述 |
| Build plugins | 在构建执行时，并在pom.xml的元素中配置。 |
| Reporting plugins | 在网站生成过程中执行，并在pom.xml的元素中配置。 |

常用插件列表：

|  |  |
| --- | --- |
| 插件 | 描述 |
| clean | 构建之后清理目标文件。删除目标文件。 |
| compiler | 编译Java源文件。 |
| surefile | 运行JUnit单元测试。创建测试报告。 |
| jar | 从当前工程中构建JAR文件。 |
| war | 从当前工程中构建WAR文件。 |
| javadoc | 为工程生成Javadoc。 |
| antrun | 从构建过程的任意一个阶段中运行一个ant任务集合。 |

除此之外，还有一些优秀的第三方插件，可以帮助快捷、方便的构架项目。当使用到某些功能或者特性的时候多加搜索。例如，项目中使用了Mybatis，就有一款插件，运行一个命令就可根据数据库的表，自动生成Mybatis的mapper配置文件以及DAO层接口模板。