

Maven

Maven

- Build automation and dependency management tool
- Command line tool
- Apache Software Foundation
- Created in 2003

Motivation

- Enterprise application
 - Source code
 - Dependencies (libraries)
- Automated/Repeatable way to assemble an application into deployable artifacts (JAR, WAR etc.)
- Deploy artifacts
- Doesn't depend on IDE
- Project metadata
 - SCM system
 - Reports
 - Documentation

Dependency management

- Central repository of artifacts (libraries)
- Locate and download dependencies
- Local (cache) or remote repositories
- First checks local repo (by default is in .m2 directory in user space)
- Private or public

Maven approach

- Uses "convention over configuration" approach
 - E.g., Source code and test code is in a certain directory, and you cannot redefine the location
 - Project build lifecycle consists of certain stages
 - Less flexible but standardized approach
 - You don't need to figure out how to organize your project and its build lifecycle for each application
 - Implemented by super pom.xml that is provided by maven itself
- Customizable by plugins

Project object model

- Describes maven project metadata and dependencies
- Specified in pom.xml file

Lifecycle

- Sequence of named phases
- Phases are executed sequentially
- Executing a phase executes all previous phases
- Lifecycles: clean, default and site
- Clean is the largest lifecycle

Plugins

- Plugin is a collection of goals
- Plugin is like a class and goal is like a method in Java
- Example: compiler plugin, has separate goals for compiling source code and test code
- Can be bound to lifecycle phase
- When a certain lifecycle phase is executed then all plugins and its goals tied to this phase are executed

Under the hood

